



TCPWave IP Address Management System®

CLI Reference Guide

Version 11.32P4

August 2022

TCPWave® Inc.

600 Alexander Road

Princeton, NJ 08540

USA

Phone: 888-831-8276

Email: support@tcpwave.com

Website: www.tcpwave.com

This manual is the proprietary and confidential property of TCPWave® Inc. All resulting rights, the rights of translation and duplication, are reserved and shall be subjected to a separate agreement. Do not share without prior approval.

TCPWave® Inc. reserves the right to modify the described product in compliance with technical progress at any time and without prior notice unless otherwise provided in the agreement.

Document Change History

Revision Date	Summary of Changes
August 2022	<p>Added the following CLI's:</p> <ul style="list-style-type: none">▪ <u>addipv6dnsreversezone</u>▪ <u>deletevrf</u>▪ <u>deleteipv6subnetmpl</u>▪ <u>deleteipv6dnsserver</u>▪ <u>deleteipv6dnsreversezone</u>▪ <u>editvrf</u>▪ <u>exportipv6dhcpserver</u>▪ <u>exportipv6dnsserver</u>▪ <u>editipv6dnsreversezone</u>▪ <u>enablefadmaccess</u>▪ <u>exportsubnettemplate</u>▪ <u>getipv6dnsserver</u>▪ <u>getipv6subnetmpl</u>▪ <u>getperfmetricstatistics</u>▪ <u>importipv6dhcpserver</u>▪ <u>importsubnettemplate</u>▪ <u>importipv6dnsserver</u>▪ <u>listvrf</u>▪ <u>listipv6subnetmpl</u>▪ <u>listipv6dnsserver</u>▪ <u>listipv6dnsreversezone</u>▪ <u>setipv6subnetmpl</u>▪ <u>setipv6remotemonitor</u>▪ <u>setipv6remotedebug</u>▪ <u>setipv6remotectrllog</u>▪ <u>setipv6dnsserver</u>▪ <u>syncipv6dnsserver</u>▪ <u>syncipv6dhcpserver</u>

Contents

Introduction	20
Executing Commands	20
Additions	20
acceptdevices	20
addasset.....	20
addadminrole	21
addadmin.....	22
addadminpermission	24
addmicrosoftdhcpserver.....	25
addmicrosoftdnsserver.....	25
addmicrosoftadserver	26
addmicrosoftadsite.....	27
addcloudprovider	27
addadmingroup	30
addappliancegroup.....	31
addasnumber.....	32
addcloudprovider	33
addcontact.....	36
addcustomfolder	36
adddhcption	37
adddnsforwarders	37
adddnsforwarderstmpl.....	38
addlogchannel	38
adddhcpfailoverpeer	40
adddhcpfingerprint.....	41
adddhcptionspace	42
adddnsreversezone	42
addipv6dnsreversezone.....	44
adddnsrootzone.....	46
adddnszone.....	48
adddnsview.....	53
adddnszonetmpl.....	54
adddomain.....	57

adddiscoverytmpl	57
adext	58
addipamtoha	59
addipv6network	59
addipv6object	60
addipv6scope	63
addipv6subnet	65
addipv6subnetgroup	66
addipv6block	67
addipv6pool	70
addlocation	72
addmirroredzone	73
addmicrosoftdhcpserver	73
addmicrosoftdnsserver	74
addnetwork	75
addobject	76
addobjecttype	81
addorg	81
addrpztmpl	82
addr	83
addscheduledjob	93
addscope	95
addsubnet	97
addsubnetgroup	101
addmultiarr	101
addawsimage	102
adddhcpssharednetwork	102
applybinlog	102
addmultiarr	103
addvrf	103
Deletions	104
deleteactivelease	104
deleteadminrole	105
deleteadmin	105
deleteadminpermission	105
deleteadmingroup	106

deleteappliancegroup.....	107
deleteawsimage.....	107
deleteasnumber.....	108
deletecontact.....	108
deletecustomfolder	109
deletedhcpfailoverpeer	109
deletedhcpfingerprint.....	110
deletedhcoption	110
deletedhcoptionspace	111
deletedhcpclass	111
deletedhcpserver.....	111
deletedhcpopttmpl.....	112
deletedhcppolicytmpl.....	113
deletediscoverytmpl	113
deletednsacl.....	113
deletednsforwarders	113
deletednsforwarderstmpl.....	114
deletednsopttmpl	114
deletednsreversezone	115
deletednsrootzone	115
deletednserver	116
deletednservertmpl.....	116
deletemicrosoftadserver	116
deletemicrosoftadsite.....	117
deletednsview.....	117
deletednszone	117
deletednszonetmpl.....	118
deletedomain.....	118
deletedump	119
deleteext.....	119
deleteextvalue	119
deletefirewalltmpl	120
deleteipamfromha.....	120
deleteipamappliance	120
deleteipv6dnsserver	121
deleteipv6dnsreversezone	121

deleteipv6dhcpserver	122
deleteipv6dhcptmpl.....	122
deleteipv6network	122
deleteipv6block	123
deleteipv6object.....	123
deleteipv6scope.....	124
deleteipv6subnet.....	124
deleteipv6subnettmpl	125
deleteipv6subnetgroup	126
deletelocation.....	126
deletelogchannel	127
deletemirroredzone.....	127
deletemicrosoftdhcpserver	128
deletemicrosoftdnsserver.....	128
deletenetwork	128
deleteobject.....	129
deleteobjecttype.....	130
deleteorg.....	130
deletepatch.....	130
deleterpztmpl	131
deleterr	131
deletescheduledjob	134
deletescope	134
deletescopeactivelease	135
deletesubnet.....	135
deletesubnettimpl	136
deletesubnetgroup	136
deleteasset.....	137
deletevrf	137
deploypatch	137
disablezonemonitor.....	138
deletediscovertask.....	139
deletedhcpsharednetwork	139
deletecloudprovider	139
deletensmtmpl.....	140
discover.....	140

displayzonedata	141
downloaddhcpconfig	141
downloaddnsconfig	141
dumpdb	142
Updates	142
editasset	142
editcloudprovider	144
editadminrole	146
editadminpermission	148
editadmin	149
editadmingroup	151
editappliancegroup	151
editawsimage	152
editasnumber	153
editcontact	154
editcustomfolder	155
editdhcpcfailoverpeer	155
editdhcpoptionspace	156
editdiscoverytmpl	157
editdhcpfingerprint	157
editdhcption	158
editdnsforwarders	159
editdnsforwarderstmpl	161
editdnsreversezone	161
editmicrosoftadserver	164
editdnsrootzone	165
editdnszone	168
editdnszonetmpl	172
editdomain	175
edittext	176
edittextvalue	177
editglobalopts	180
editipv6network	180
editipv6object	181
editipv6subnet	183
editipv6dnsreversezone	185

editipv6subnetgroup	187
editipv6block	187
editipv6pool.....	189
editlocation.....	189
editlogchannel.....	191
editmirroredzone.....	192
editmicrosoftdhcpserver	192
editmicrosoftdnsserver.....	193
editmicrosoftadsite.....	195
editnetwork	196
editobject.....	197
editobjecttype	201
editorg	201
editrpztmpl	202
editrr	203
editscheduledjob	211
editsubnet.....	213
editsubnetgroup	216
editdhcpsharednetwork	216
editcloudprovider	217
editnsmtmpl	220
editvrf	222
enablezonemonitor	222
exescheduledjob.....	223
Exports	223
exportadminrole	223
exportadmin	224
exportadmingroup.....	225
exportadminpermission.....	226
exportappliancegroup	227
exportasset	228
exportcontact	229
exportdhcpfailoverpeer.....	229
exportdhcptiontimpl	230
exportdhcppolicytmpl.....	234
exportdhcpserver	236

exportdnsacl	238
exportdnsoptiontmpl.....	239
exportdnsserver.....	241
exportdnsservertmpl	244
exportdomain	245
exportextension.....	245
exportipamappliance.....	246
exportipv6object.....	248
exportipv6objectrr.....	249
exportipv6subnet.....	250
exportipv6subnetgroup	251
exportipv6block	252
exportipv6pool.....	253
exportipv6reversezone.....	254
exportipv6dnsserver	255
exportipv6dhcpserver.....	261
exportlocation	267
exportlogchannel.....	268
exportnetwork	269
exportobject	270
exportobjecttype	271
exportobjectrr.....	271
exportorg	272
exportreversezonetmpl	272
exportrevzonerr	273
exportreversezone.....	274
exportrr	275
exportscope	275
exportsharednetwork	276
exportsubnet	277
exportsubnetdhcp	278
exportsubnetgroup	278
exportvrf	279
exportmirroredzone	280
exportmicrosoftdnsserver	280
exportzone.....	281

exportzonerr	282
exportsubnettemplate	283
exportzonetemplate	287
exportdnsforwarderstmpl	288
Gets.....	290
fetchquerylog.....	290
getipamappliance	291
getbgpconfig	291
getchanageticket	292
getdhcpolicytmpl	292
getdhcptmpl	295
getdhcpserver.....	299
getdnsacl.....	308
getdnsdebuglevel	308
getdnsopttmpl	308
getdnsserver	316
getdnsservertmpl.....	323
getdnszone	324
getfirewalltmpl	325
getipv6dnsserver	326
getipv6dhcpserver	337
getipv6dhcptmpl.....	338
getfreesubnetlist.....	340
getnextfreeip	340
getnonmanageddnsmaster.....	341
getobject.....	341
getrootaccessmgmt	342
getsubnet.....	342
getipv6subnettmpl	343
getperfmetricstatistics.....	344
getsubnettmpl	346
getzoneacl.....	346
getremotedebug.....	347
getnonmanageddnsmaster.....	348
Imports.....	348
importmicrosoftdnsserver	348

importadmin.....	349
importmirroredzone	350
importadminrole.....	351
importadmingroup	353
importadminpermission	354
importappliancegroup	355
importasset.....	356
importcloudinstances	357
importcontact.....	357
importdhcpfailoverpeer.....	358
importdhcptiontimpl.....	359
importdhcpolicytimpl	364
importdhcpserver	366
importipv6dhcpserver	369
importdnsacl.....	373
importdnsoptiontimpl	374
importipv6dnsserver	377
importdnsserver	384
importdnsservertimpl.....	387
importdomain.....	388
importextension	389
importipamappliance	390
importipv6object	392
importipv6objectrr	394
importipv6subnet	395
importipv6subnetgroup.....	396
importipv6block.....	398
importipv6pool	399
importlocation	401
importlogchannel	402
importnetwork.....	403
importobject	404
importobjectrr	406
importobjecttype.....	407
importorg.....	408
importreversezonetmpl.....	409

importrevzonerr	410
importreversezone	411
importipv6reversezone	412
importscope.....	414
importsharednetwork.....	415
importsubnet.....	416
importsubnettemplate	417
importsubnetdhcp	421
importsubnetgroup	421
importvrf.....	422
importzone	423
importzonerr.....	424
importzonetemplate.....	425
initdb.....	427
importlicense	427
importdnsforwarderstmpl.....	428
Lists	430
listactivelease	430
listavailablesubnet	430
listzonerr	431
listadminrole.....	431
listadmin	431
listadmingroup.....	432
listadminpermission	432
listappliancegroup	433
listpatch	433
listalerts	433
listawsimages.....	434
listawsinstance.....	434
listasnumber	435
listcloudprovider	435
listcloudprovidertypes	436
listcontact	436
listcustomfolder.....	436
listdhcpclass.....	437
listdhcfailoverpeer.....	437

listdhcpfingerprint	438
listhcoption.....	438
listhcpserver	438
listdhcpsharednetwork.....	439
listdhcptmpl.....	439
listdiscovertask	440
listdiscovertmpl.....	440
listdnsacl	440
listdnsforwarders.....	441
listdnsreversezone.....	442
listdnsopttmpl.....	442
listipv6dnsserver	442
listdnsserver.....	443
listdnsserververtmpl	443
listdnsview	443
listdnszone	444
listdnszonetmpl	444
listdomain	445
listext	445
listextvalue.....	446
listdumps	446
listfunctions	447
listipv6subnettmpl	448
listfirewalltmpl.....	448
listglobalopts.....	448
listhaipam	448
listhareremote	449
listipv6dhcpserver.....	449
listipv6dhcptmpl	450
listipv6network	450
listipv6object	450
listipv6scope	452
listipv6subnet	452
listipv6subnetgroup.....	453
listipv6block	453
listipv6pool	454

listipv6dnsreversezone	455
listlocation	455
listlogchannel.....	455
listmirroredzone	456
listmicrosoftdhcpserver.....	456
listmicrosoftdnsserver	456
listnetwork.....	457
listnsmtmpl	457
listmicrosoftadsites.....	458
listobject	458
listobjecttype	459
listorg	459
listroundrobin	459
listrpztmp.....	460
listrr.....	460
listrootaccessmgmt.....	461
listscheduledjob.....	461
listscope	462
listserversforzone	462
listsubnet	462
listsubnetgroup.....	464
listsubnetsforserver.....	464
listipv4subnettemplate.....	464
listrestricteddomain	465
listpolicycompliance	465
listnsmtmpl	466
listvrf.....	467
haipampreference	467
Other Commands.....	467
applydracton.....	467
applysnapshot.....	468
autologin.....	468
autologinmanager	469
addnsmtmpl.....	470
backupbinlog	471
changepassword	471

changereference.....	472
changerootpassword	472
clearaudit.....	473
clonednszone.....	473
configureha.....	473
configrootaccessmgmt	474
baselinepolicycompliance.....	474
changeschcedjobtype	476
enablefadmaccess	476
fetchnamedlog.....	477
login	477
logout.....	478
mergesubnet.....	479
moveobject.....	480
patchdeploymentinfo	481
purgeipamentity	481
promoteipamtomaster	482
rebuildsearch	483
redistributeremotes.....	483
resetdr	483
resethaconfig	483
restoredb	484
rndcflush	484
rollbackpatch	485
chkipexistence	486
chkobjnameexistence	486
rptadminaudit.....	486
rptdhcpactivelease	487
rptdhcpdnssvevents	488
rptdhcprotoptmplaudit	489
rptdhcppolicytmplaudit	490
rptdhcpsvraudit	490
rptdhcpsvrmanagedsnet.....	491
rptdhcprotopt.....	492
rptdhcptmplassociation.....	492
rptdhcptscopecount	493

rptdnsottmplaudit	493
rptdnssoa	494
rptdnssvraudit.....	495
rptdnssvrtmplaudit	495
rptgloballocbyobjtype.....	496
rptipameventaudit.....	497
rptloginactivity.....	497
rptmonitoringalerts	498
rptnetaudit.....	499
rptobjaudit.....	500
rptrraudit	500
rptschedeventsaudit.....	501
rptsnetaudit	502
rptsubnetauditbygroup	503
rptsubnetlistbygroup	504
rptsvrconfigaudit	504
rpttopalertproducers	506
rptv4netspaceutil.....	506
rptv4netspaceutil	507
rptzoneaudit	508
rptzonetmplaudit.....	508
redistributeremotes.....	509
runpolicycompliance.....	509
search.....	511
setbgpconfig	513
setchangeticket.....	515
setdhcpclass.....	515
setdhcppolicytmpl	516
setdhcpserver	518
setdhcptmpl.....	525
setdnsacl	529
setdnsdebuglevel.....	529
setdnsopttmpl	530
setdhcpopttmpl	538
setipv6remotemonitor	542
setdnsserver	543

setipv6dnsserver.....	550
setdnsservertmpl.....	557
setfirewalltmpl.....	558
setipv6dhcpserver	561
setipv6dhcptmpl.....	562
setremotenicsyncoption.....	564
setremotedebug	564
setipv6remotedebug	565
setipv6remotectrllog	565
setsubnettmpl.....	566
setipv6subnettmpl.....	569
setupdr.....	571
showconnected.....	571
showdefaultroute	572
showdevices.....	573
showdrconfig	573
showjobexehistory.....	573
showlicense	574
showsecuritylog.....	574
splitsubnet	574
syncdhcpserver.....	575
syncipv6dhcpserver	576
syncdnsserver	576
syncipv6dnsserver	577
syncmicrosoftdhcpserver	577
syncmicrosoftdnsserver.....	578
syncobject	578
synczone	579
syncactivelease	579
undolist	580
updatelicense.....	580
undoipamentiity	581
uploadpatch.....	582
zoneaxfrtotims.....	582
pauseschedjob	583
resumeschedjob	583

setipamappliance.....	583
-----------------------	-----

Introduction

This reference guide outlines command line interfaces (CLIs) into TCPWave IP Address Management (IPAM). Using CLIs extends the efficiency of the TCPWave IPAM Administrators, providing them the flexibility to run TCPWave IPAM functions from a command line. Use of CLIs can shorten the time needed to bulk import or export data or can allow for scheduling of tasks outside the TCPWave IPAM product using cron or Windows Task Scheduler.

Executing Commands

Additions

acceptdevices

NAME:

acceptdevices

DESCRIPTION:

Updates the TCPWave IPAM with the discovered objects on a given subnet.

ARGUMENTS:

--id

Command ID of the discovered subnet. Use the following command to see all the command IDs of discovered subnets: 'twc listdiscovertask --d=,[mandatory]

EXAMPLE USAGE:

twc acceptdevices --id=1391

addasset

DESCRIPTION:

Creates an asset data in the TCPWave IPAM

ARGUMENTS:

--service_tag

Service tag for the asset. [mandatory]

--serial_num

Unique serial number for the asset. [mandatory]

--vendor

Name of the vendor for the asset.

--model

Model value for the asset.

--name

Name for the asset.

--acquisition_type

Model acquisition type for the asset.

--maintenance_cost

Cost value of the asset for maintenance.

--cpu

cpu range of the asset.

--capacity

Capacity value for the asset.

--os_version

Supported os version for the asset.

--purchase_cost

Purchase value of the asset.

--disposal_reason

Reason for the disposal of asset.

--city

City name for the asset.

--green_zone

Green zone is to be set for asset.

--warranty_end_dt

Last Warranty date of the asset, date format is yyyy/mm/dd.

--purchase_dt

Purchase date of the asset, date format is yyyy/mm/dd.

--disposal_dt

Disposal date of the asset, date format is yyyy/mm/dd.

--description

Description for the asset.

EXAMPLE:

```
twc addasset --service_tag="Dell Inc." --serial_num=36906 --green_zone=1 --name=test
```

```
twc addasset --service_tag="Dell" --serial_num=36909 --purchase_dt=2020/04/12 --  
disposal_dt=2020/04/15 --warranty_end_dt=2020/04/20
```

addadminrole

NAME:

addadminrole

DESCRIPTION

Creates an administrator role in the TCPWave IPAM.

ARGUMENTS

--name
Name of the administrator role. [mandatory]

--functions
Name of the functions that accepts the comma separated function.

--desc
Description for administrator role.

EXAMPLE:

```
twc addadminrole --name=CADM --functions="Quick Tasks,Appliance Groups,Bulk Data Import"  
--desc="Custom Admin"
```

addadmin

NAME

addadmin - Creates an administrator in the TCPWave IPAM.

DESCRIPTION

The user is identified uniquely by the combination of First Name, Last Name, and Email Id. The user with administrator privileges are associated with a login name and password using which they can access TIMS IPAM.

Each administrator user is assigned with a role that controls their access to the system.

When an administrator user exceeds the maximum number of failed login attempts, his account will be suspended and has to be re-instated by a User administrator for further access to the system.

The password is subjected to the System level configuration rules which includes password ageing, password complexity, and password-reuse among others.

ARGUMENTS

--first_name
First name of the administrator. [mandatory]

--middle_name
Middle name of the administrator.

--last_name
Last name of the administrator. [mandatory]

--email
Email address of the administrator. [mandatory]

--phone
Phone number of the administrator.

--login_name

Login name of the administrator. [mandatory]

--groups

Administrator groups of the administrator. This is a comma separated list of administrator group.

--org

Default organization name in which the administrator has to be created. [mandatory]

--role

Default role of the administrator. [mandatory]

The following roles are the default roles supported by TCPWave IPAM
SADM - Super Administrator, has access to all the functionality of the system

FADM - Functional Administrator, Special administrator with functional privileges and valid for the special user 'twcadm' only. This role provides the privileges to switch authentication mechanisms and set system level parameters.

UADM - User Administrator, Has access to user administration functionality only

NADM - Normal Administrator, Has privileges only to create Objects and Scopes

PADM - Power Administrator, Has access to following IPAM entities Zone, Domain, Server, Network, Subnet, Scope, Template and Object

RADM - Read-only Administrator

--ext_attr

Comma separated list of extension attributes with their values in the format : extension_attribute_name/extension_attribute_value. Use the following command to see all the extension attributes applied to administrator :
'twc listext --entity=admin --d=.'

EXAMPLE

```
twc addadmin --first_name=John --last_name=Smith --email=john.smith@tcpwave.com --  
phone=920-310-5555 --login_name=jsmith --org=TCPWave --groups=default --role=NADM  
twc addadmin --first_name=John --middle_name=Fitzgerald --last_name=Kennedy --  
email=john.kennedy@tcpwave.com --phone=920-310-5555 --login_name=jkennedy --org=TCPWave --  
groups=default --role=SADM  
twc addadmin --first_name=John --last_name=Smith --email=john.smith@tcpwave.com --  
phone=920-310-5555 --login_name=jsmith --org=TCPWave --groups=default --role=PADM --  
ext_attr=ext_attr_1/value_1,ext_attr_2/value_2
```

addadminpermission

NAME:

addadminpermission

DESCRIPTION:

Defines an administrator/administrator group permissions in the TCPWave IPAM.

ARGUMENTS:

--level

It takes the input as admin or admin group, if the level is admin the input param of admin is mandatory otherwise admin group is mandatory. [mandatory]

-admin

Name of the admin.

--admin_group

Name of the admin group.

--org

Name of the organization. [mandatory]

--role

Name of the administrator role. [mandatory]

--function

Name of the administrator function. [mandatory]

--privilege

Name of the privilege, It takes the input as 'Read' or 'Write' or 'Deny'. [mandatory]

--select_all

It takes the input as '0' or '1'.

--list_entity

List of entity, It accepts the comma separated entity.

EXAMPLE

```
twc addadminpermission --level=Admin --admin=Test --org=TcpWave --role=EADM --function="IPv4 Networks" --privilege=Write --select_all=0 --list_entity=10.1.10.0
```

```
stwc addadminpermission --level="Admin Group" --admin_group=Test --org=TcpWave --role=EADM --function="TCPWave DHCP IPv4 Appliances" --privilege=Read --select_all=1
```

addmicrosoftdhcpserver

DESCRIPTION:

Creates Microsoft DHCP appliance in the TCPWave IPAM.

ARGUMENTS:

--ip

IP address of the appliance. [mandatory]

--org

Name of the organization. [mandatory]

--use_https

Takes '0' or '1' as input.

--user_name

User name for the Microsoft appliance. [mandatory]

--mac_exclusion_addr

Comma separated mac exclusion addresses without any spaces.

--desc

Description for the Microsoft DHCP appliance.

EXAMPLE:

```
twc addmicrosoftdhcpserver --ip=10.0.0.10 --org=TCPWave --user_name=Administrator --mac_exclusion_addr=AA:BB:CC:DD:EE:F5,AA:BB:CC:DD:EE:F9
```

```
twc addmicrosoftdhcpserver --ip=10.0.0.10 --org=TCPWave --use_https=1 --user_name=Administrator --desc="First Microsoft Appliance"
```

addmicrosoftdnsserver

DESCRIPTION:

Creates a Microsoft DNS appliance in the TCPWave IPAM.

ARGUMENTS:

--ip

IP address of the appliance. [mandatory]

--org

Name of the organization. [mandatory]

--use_https

Takes '0' or '1' as input

--user_name

User name for the Microsoft appliance. [mandatory]

--desc

Description for the Microsoft DNS appliance.

EXAMPLE

```
twc addmicrosoftdnsserver --ip=10.0.0.10 --org=TCPWave --user_name=profile
```

```
twc addmicrosoftdnsserver --ip=10.0.0.10 --org=TCPWave --use_https=1 --user_name=user --desc="First Microsoft Appliance"
```

addmicrosoftadserver

NAME

addmicrosoftadserver

DESCRIPTION

Creates Microsoft AD appliance in the TCPWave IPAM.

ARGUMENTS

--ip

IP address of the appliance. [mandatory]

--org

Name of the organization. [mandatory]

--use_https

Takes '0' or '1' as input.

--user_name

User name for the Microsoft appliance. [mandatory]

--desc

Description for the Microsoft AD appliance.

EXAMPLE

```
twc addmicrosoftadserver --ip=10.0.0.10 --org=TCPWave --user_name=Administrator
```

```
twc addmicrosoftadserver --ip=10.0.0.10 --org=TCPWave --use_https=1 --user_name=Administrator --desc="First Microsoft Appliance"
```

addmicrosoftadsite**NAME**

addmicrosoftadsite

DESCRIPTION

Creates Microsoft AD site in the TCPWave IPAM. You can enter site name up to 32 characters (alpha-numeric and hyphen). The system restricts you to enter a space between the words. At least one alphabet needs to be part of the name. The name cannot contain all the numerals.

ARGUMENTS**--name**

Name of the Microsoft AD site. [mandatory]

--subnet_group

Name of the subnet group, it accepts the comma separated subnet groups.

--ip

IP address of the appliance. [mandatory]

--org

Name of the organization. [mandatory]

--desc

Description for the Microsoft AD site.

EXAMPLE

```
tvc addmicrosoftadsite --name=UKSite --ip=10.0.0.10 --org=TCPWave
```

```
tvc addmicrosoftadsite --name=NJSite --subnet_group=NJGroup --ip=10.0.0.10 --org=TCPWave -  
-desc="NJ Site"
```

addcloudprovider**DESCRIPTION**

Creates a cloud provider in the TCPWave IPAM. Different type of cloud providers support different credentials. Follow the example section to add particular type of cloud provider.

ARGUMENTS**--org**

Organization name to be associated with the cloud provider. This argument is for users in FADM role to select appropriate organization to which the operation has to be applied. For users not in FADM role, the operation is by default applied to the organization that the user is associated with.

--provider_type

Type of the Cloud provider. In TCPWave IPAM provider type represents the

cloud service provider . TCPWave IPAM support following cloud providers:
'AKAMAI', 'AWS', 'AZURE', 'CLOUDFLARE', 'DYNDNS' and 'GOOGLE'
[mandatory].

--name

Name of the cloud provider [mandatory].

--user

User name of the cloud provider

--account

Valid AWS account number which manages the resource.

--iam_role

Use IAM Role mapped to the EC2 instance. It is applicable for AWS cloud provider.

--api_key

API key for the cloud provider. It is global API key for CLOUDFLARE provider.

--keystore_file

Key store file for the cloud provider. This key store file contains Secret access key. It is applicable only for AZURE provider type.

--application

Application ID for the cloud provider. It is applicable for AZURE cloud provider.

--service_account_id

Service Account ID for the GOOGLE cloud provider.

--p12file

p12file for the GOOGLE cloud provider.

--project_id

Project ID for the GOOGLE cloud provider.

--ad_tenant

Ad tenant ID for the AZURE cloud provider .

--resource_group

Resource Group for the AZURE cloud provider.

--customer_name

Customer name for the DYNDNS cloud provider.

--email

Email Address for the CLOUDFLARE provider.

--region

Region defines area of AWS cloud provider. For AWS Cloud provider
region can be one of the following-

US East (N. Virginia)

US East (Ohio)

US West (N. California)

US West (Oregon)

Asia Pacific (Mumbai)

Asia Pacific (Seoul)

Asia Pacific (Singapore)

Asia Pacific (Sydney)

Asia Pacific (Tokyo)

Canada (Central)

EU (Frankfurt)

EU (Ireland)

EU (London)

South America (Sao Paulo)

--desc

Description for the cloud provider.

EXAMPLE

```
twc addcloudprovider --provider_type=AWS --org=TCPWave --name="AWS-Provider50" --account=12 --user="AKIAINMQOEG7EBKXMTOQP" --region="EU (London)" --desc="AWS Cloud Provider"
```

```
twc addcloudprovider --provider_type=AWS --org=TCPWave --name="AWS-Provider50" --account=10 --iam_role=1 --region="EU (London)" --desc="AWS Cloud Provider"
```

```
twc addcloudprovider --provider_type=GOOGLE --org=TCPWave --name="Google-Provider22" --service_account_id="jhon@tcpwave-14981012.iam.gserviceaccount.com" --p12file="/tmp/tcpwave-2d185caa49dc1.p12" --project_id="tcpwave-14912810" --desc="Google Cloud Provider"
```

```
twc addcloudprovider --provider_type=AZURE --org=TCPWave --name="Azure-Provider03" --user="ppc0e31c0f-fdb0-438c-afff-6ea7600b0e61" --keystore_file="cloud_dns_app.pfx" --application="ebe1b568-5e63-46f0-9201-8a465cee092dqq" --ad_tenant="772a8482-16c9-4823-9f15-bd19827d23f111" --resource_group="tcpwave" --desc="Azure Cloud Provider"
```

```
twc addcloudprovider --provider_type=DYNDNS --org=TCPWave --name="DynDNS-Provider01" --user="jhon-smith" --customer_name="tcpwave01" --desc="DYNDNS Cloud Provider"
```

```
twc addcloudprovider --provider_type=CLOUDFLARE --org=TCPWave --name="CLOUDFLARE-Provider02" --email="jhon.tcpwave@tcpwave.com" --api_key="3cde9f553a9a21049e00046" --desc="CLOUDFLARE Cloud Provider"
```

```
twc addcloudprovider --provider_type=AKAMAI --org=TCPWave --name="Akamai-Provider06" --user="jhon.tcpwave@tcpwave.com" --api_key="client_secret" = xd3RTCMlmmZhdQ82LD34yAZUqOwc2DDt1ANgDAoc6iguY=host = akab-34nyw47p22fhpvptnu-v7ygacgwkb6cswza.luna.akamaiaapis.net access_token = akab-a24w5rojdc6lckdm-tcvscbkoo5ise5bw2 client_token = akab-sxdp7uvgkonm7jfu-w3phslypnzzv3llqv" --desc="AKAMAI Cloud Provider"
```

addadmingroup

NAME:

addadmingroup

DESCRIPTION:

Administrator Groups facilitate administrative simplicity by providing a mechanism to group administrator users. The grouping is adhoc and can be based on organization policies.

ARGUMENTS:

--name

Name of the administrator group. [mandatory]

--desc

Description for the administrator group.

--roles

Defines the role of the administrator group.

Accepts multiple roles with comma separated pairs of role and organization as shown:

Example: SADM,TcpWave;CADM,Internal

Note: FADM and UADM roles are not organization specific.

EXAMPLE:

```
twc addadmingroup --name="default-admin-group" --desc="Default Admin Group" --  
roles="SADM,TcpWave;CADM,Internal"
```

addappliancegroup

NAME:

addappliancegroup

DESCRIPTION:

Creates an appliance group in the TCPWave IPAM.

ARGUMENTS:

--name

Name of the appliance group being created. [mandatory]

--org

Name of the organization where the operation must be performed. This argument is mandatory if the user is FADM.

--desc

Description of the appliance group.

--parent

Name of the parent appliance group.

EXAMPLE USAGE:

```
twc addappliancegroup --name=IT_AG --desc="IT appliance group" --org=TCPWave
```

```
twc addappliancegroup --name=IT_AG --desc="IT appliance group" --org=TCPWave --  
parent=IT_PG_1
```

addasnumber**NAME**

addasnumber

DESCRIPTION

Creates a Autonomous System Number in the TCPWave IPAM.

ARGUMENTS

--org

Organization name under which the Autonomous System Number is being created. [mandatory]

--name

Name of the Autonomous System Number. [mandatory]

--as_number

Autonomous System number. It accepts up to 5 digits of the number.
[mandatory]

--email

Email address for the Autonomous System Number.

--desc

Description for the Autonomous System Number..

EXAMPLE

```
twc addasnumber --name="ASN" --org=TCPWave --as_number=151 --desc="TCPWave Autonomous System Number."
```

```
twc addasnumber --name="TCPWave-ASN" --org=TCPWave --as_number=151 --email=jsmith@tcpwave.com
```

addcloudprovider**NAME:**

addcloudprovider

DESCRIPTION:

Creates a cloud provider in the TCPWave IPAM. Different type of cloud providers supports different credentials. Follow the example section to add particular type of cloud provider.

ARGUMENTS:

--org

Organization name to be associated with the cloud provider. This argument is for users in FADM role to select appropriate organization to which the operation must be applied. For users not in FADM role, the operation is by default applied to the organization that the user is associated with.

--provider_type

Type of the Cloud provider. In TCPWave IPAM provider type represents the cloud service provider. TCPWave IPAM support following cloud providers:

'AKAMAI', 'AWS', 'AZURE', 'CLOUDFLARE', 'DYNDNS' and 'GOOGLE'[mandatory].

--name

Name of the cloud provider [mandatory].

--user

Username of the cloud provider

--password

API key for the cloud provider. It is Secret access key for AWS, and it is global API key for CLOUDFLARE provider.

--keystore_file

Key store file for the cloud provider. This key store file contains Secret access key. It is applicable only for AZURE provider type.

--application

Application ID for the cloud provider. It is applicable for AZURE type cloud provider.

--password

Password for the cloud provider. It is applicable for DYNDNS type cloud provider.

--service_account_id

Service Account ID for the GOOGLE type cloud provider.

--p12file

p12file for the GOOGLE type cloud provider.

--project_id

Project ID for the GOOGLE type cloud provider.

--ad_tenant

Ad tenant ID for the AZURE type cloud provider.

--resource_group

Resource Group for the AZURE type cloud provider.

--customer_name

Customer name for the DYNDNS type cloud provider type.

--email

Email Address for the CLOUDFLARE provider.

--region

Region defines area of AWS type cloud provider. For AWS Cloud provider region can be one of the following-

- US East (N. Virginia)
- US East (Ohio)
- US West (N. California)
- US West (Oregon)
- Asia Pacific (Mumbai)
- Asia Pacific (Seoul)
- Asia Pacific (Singapore)
- Asia Pacific (Sydney)
- Asia Pacific (Tokyo)
- Canada (Central)
- EU (Frankfurt)
- EU (Ireland)
- EU (London)
- South America (Sao Paulo)

--desc

Description for the cloud provider.

EXAMPLE USAGE:

```
twc addcloudprovider --provider_type=AWS --org=TCPWave --name="AWS-Provider50" --
user="AKIAINLQMEG7EBWXMTOQP" --
api_key="E52BxojR5f2hM802hG+Zl8Z4boxzlZRNcnPpaii1+" --region="EU (London)" --
desc="AWS Cloud Provider"
```

```
twc addcloudprovider --provider_type=GOOGLE --org=TCPWave --name="Google-Provider22" --
service_account_id="jhon@tcpwave-14981012.iam.gserviceaccount.com" --
p12file="/tmp/tcpwave-2d185caa49dc1.p12" --project_id="tcpwave-14912810" -desc="Google
Cloud Provider"
```

```
twc addcloudprovider --provider_type=AZURE --org=TCPWave --name="Azure-Provider03" --
user="ppc0e31c0f-fdb0-438c-afff-6ea7600b0e61" --keystore_file="cloud_dns_app.pfx" --
application="ebe1b568-5e63-46f0-9201-8a465cee092dqq" --ad_tenant="772a8482-16c9-4823-
9f15-bd19827d23f111" --resource_group="tcpwave" --password="abc123" --desc="Azure Cloud
Provider"
```

```
twc addcloudprovider --provider_type=DYNDNS --org=TCPWave --name="DynDNS-Provider01" --
user="jhon-smith" --password="123CO2zbCJ6Qb" --customer_name="tcpwave01" --
desc="DYNDNS Cloud Provider"
```

```
twc addcloudprovider --provider_type=CLOUDFLARE --org=TCPWave --name="CLOUDFLARE-
Provider02" --email="jhon.tcpwave@tcpwave.com" --api_key="3cde9f553a9a21049e00046" --
desc="CLOUDFLARE Cloud Provider"
```

```
twc addcloudprovider --provider_type=AKAMAI --org=TCPWave --name="Akamai-Provider06" --
user="jhon.tcpwave@tcpwave.com" --password="Glider0N123#" --api_key="client_secret =
xd3RTCMImmZhdQ82LD34yAZUqOwc2DDt1ANgDAoc6iguY=host = akab-
```

```
34nyw47p22fhpvptnu-v7ygacgwkb6cswza.luna.akamaiapis.net    access_token = akab-
a24w5rojdc6lckdmt-cvscbkoo5ise5bw2 client_token = akab-sxdp7uvgkonm7jfu-
w3phslypnzzv3llqv" --desc="AKAMAI Cloud Provider"
```

addcontact**NAME:**

addcontact

DESCRIPTION:

Create a contact for a given organization in the TCPWave IPAM.

ARGUMENTS:

- org Organization name for which the contact is being created. [mandatory]
- first_name First name field of the contact information. [mandatory]
- middle_name Middle name field of the contact information.
- last_name Last name field of the contact information. [mandatory]
- email Email id field of the contact information. [mandatory]
- phone Phone number field of the contact information.

EXAMPLE USAGE:

```
twc addcontact --first_name=John --last_name=Smith --org=TCPWave --  
email=john.smith@tcpwave.com --phone=920-310-5555
```

```
twc addcontact --first_name=James --middle_name=Francis --last_name=Stuart --org=TCPWave  
--email=james.stuart@tcpwave.com --phone=920-310-5556
```

addcustomfolder**NAME:**

addcustomfolder

DESCRIPTION:

Custom folders help to organize the user defined DHCP Options based on the users' preference.

ARGUMENTS:

- name Name of the custom folder. [mandatory]

EXAMPLE USAGE:

```
twc addcustomfolder --name="voip-options"
```

adddhcpoption**NAME:**

adddhcpoption

DESCRIPTION:

Creates a user defined DHCP option in the TCPWave IPAM.

ARGUMENTS:`--name`

User defined name of the option [mandatory]

`--option_code`

Name of the unused option code [mandatory]

`--option_type`

Name of the option type. Takes 'custom' or 'sub-option' as input [mandatory]

`--data_type`

Data type of the option. Takes one of the following values

'IP ADDRESS','IP ADDRESS LIST','STRING','DOMAIN','TEXT' or 'BOOLEAN'
[mandatory]`--group_name`

Name of the DHCP custom folder or DHCP option space [mandatory]

EXAMPLE USAGE:

```
twc adddhcption --name=ip-map --option_code="option 130" --option_type=custom --  
data_type="IP ADDRESS" --group_name="voip-options"
```

adddnsforwarders**NAME**

adddnsforwarders

DESCRIPTION

Creates a DNS forwarder that is used to resolve a DNS zone that is not managed by the TCPWave. Forwarders can be created on an internal 'BIND CACHE' or 'UNBOUND' or 'BIND AUTH' with recursion enabled DNS appliance in the TCPWave IPAM.

ARGUMENTS`--ip`

IP Address of the DNS internal cache appliance [mandatory]

`--appliance_type`

Type of the DNS appliance. Takes 'BIND CACHE' or 'UNBOUND' or 'BIND AUTH' with recursion enabled DNS appliance [mandatory]

`--zone_name`

Name of the DNS forward Zone [mandatory]

--fwd_ipv4

Semicolon separated list of IPv4 addresses for forwarding the requests.

--fwd_ipv6

Semicolon separated list of IPv6 addresses for forwarding the requests.

--desc

Description for the DNS forward zone.

EXAMPLE

```
twc      adddnsforwarders    --ip=10.1.10.29    --appliance_type="BIND"    CACHE"    --zon
e_name="tcpwave.com"    --fwd_ipv4="10.1.10.204;10.1.10.10"    --desc="Default"    Forward
zone"
```

adddnsforwarderstmpl

NAME

adddnsforwarderstmpl

DESCRIPTION

Creates a DNS forwarder template with forwarder zones which are not managed by TCPWave IPAM.

ARGUMENTS

--group_name

Name of the DNS forwarders group. [mandatory]

--forwarders

Specify the forwarders in the below format. [mandatory]

'Zone Name|Semicolon Separated IPv4 Address|Semicolon Separated IPv6 Address'

if need to add second entry separate with comma.

Example: test.com|10.1.10.12;10.1.10.13,zone.com|10.1.10.15;10.1.10.16

--desc

Description for the forwarders group.

EXAMPLE

```
twc adddnsforwarderstmpl --group_name=Test --
forwarders="test.com|10.1.10.12;10.1.10.13,zone.com|10.1.10.15;10.1.10.16" --desc="Description
for the forwarders"
```

addlogchannel

NAME:

addlogchannel

DESCRIPTION:

Creates a DNS log channel in the TCPWave IPAM.

ARGUMENTS:

- name
Name of the DNS log channel [mandatory]
- type
DNS log channel type. Takes 'FILE', 'SYSLOG', 'STDERR' or 'NULL' [mandatory]
- print_time
Takes '1' or '0'. '1' enables print time. '0' disables print time.
- print_sev
Takes '1' or '0'. '1' enables print severity. '0' disables print severity.
- print_cat
Takes '1' or '0'. '1' enables print log category. '0' disables print log category.
- severity
Takes 'dynamic','debug','info','notice','warning','error' or 'critical'. [mandatory]
- file_path
Full path the file to which log is written to.
- versions
Number of versions of the log file to be preserved before purging older versions.
- size
Size of the log file before a new version of the file is created.
- facility
Syslog facility name. Should be specified, for channel type 'SYSLOG'.
- debug_level
Debug level value. Should be specified, for severity as 'debug'.

EXAMPLE USAGE:

```
twc addlogchannel --name=testlogchannel --type=FILE --severity=dynamic --  
file_path=/tmp/log/test.log --versions=10 --size=1024 --print_time=1 --print_sev=1 --print_cat=1
```

```
twc addlogchannel --name=testlogchannel --type=SYSLOG --severity=debug --debug_level=1 --  
print_time=1 --print_sev=1 --print_cat=1
```

```
twc addlogchannel --name=testlogchannel --type=STDERR --severity=dynamic --print_time=1 --  
print_sev=1 --print_cat=1
```

addhcpfailoverpeer**NAME:**

addhcpfailoverpeer

DESCRIPTION:

Creates a DHCP failover peer in the TCPWave IPAM.

ARGUMENTS:**--name**

Name of the DHCP failover peer. [mandatory]

--org

Organization name in which the DHCP failover peer is being created. This argument is mandatory if user is FADM.

--primary_appliance

IP address of the primary appliance. [mandatory]

--failover_appliance

IP address of the failover appliance. [mandatory]

--primary_port

Port number of the primary appliance. [mandatory]

--failover_port

Port number of the failover appliance. [mandatory]

--max_resp_delay

Maximum response delay value. [mandatory]

--mclt

Maximum client lead time value. [mandatory]

--max_unacked_updates

Maximum unacked updates value. [mandatory]

--split

SPLIT value. [mandatory]

--load_bal_max_sec

Load balance maximum second value. [mandatory]

--desc

Description for the DHCP failover peer being created.

--message_auth

Message authentication takes '0' or '1' as an input.

--shared_secret

Shared secret is mandatory when message_auth is specified as '1'.

--sso_interval

State switch over interval takes the input in seconds.

EXAMPLE USAGE:

```
tvc adddhcpfailoverpeer --name=dhcp-failover-peer-1 --org=TCPWave --  
primary_appliance=10.1.10.180 --failover_appliance=10.1.10.185 --primary_port=647 --  
failover_port=647 --max_resp_delay=30 --mclt=1800 --max_unacked_updates=30 --split=50 --  
load_bal_max_sec=3 --desc="DHCP Failover Peer 1"
```

```
tvc adddhcpfailoverpeer --name=dhcp-failover-peer-1 --org=TCPWave --  
primary_appliance=10.1.10.180 --failover_appliance=10.1.10.185 --appliance_type=msdhcp --  
mclt=1800
```

```
--split=50 --message_auth=1 --shared_secret=abc@123 --sso_interval=100 --desc="DHCP  
Failover Peer 1"
```

adddhcpfingerprint

NAME:

adddhcpfingerprint

DESCRIPTION:

Creates DHCP finger print data in the TCPWave IPAM

ARGUMENTS:

--mac_vendor

Name of the MAC vendor. [mandatory]

--mac_bits

First six bits of the MAC address. [mandatory]

--device_profile

Name of the device profile. [mandatory]

--os

Name of the operating system or version info.

--option_sequence

DHCP option sequence. [mandatory]

--user_agent

Name of the user agent.

--certainty_index

Certainty index value.

EXAMPLE USAGE:

```
tvc adddhcpfingerprint --mac_vendor="Dell Inc." --mac_bits=3690e6 --device_profile=profile --  
os=window --option_sequence=1,2,6,5,4,7 --user_agent=agent --certainty_index=1235
```

```
twc adddhcpfingerprint --mac_vendor="Microsoft Corp." --mac_bits=3695e6 --
device_profile=profile --option_sequence=1,2,6,5,4,7
```

adddhcoptionspace

NAME:

adddhcoptionspace

DESCRIPTION:

Creates a DHCP option space in the TCPWave IPAM.
Option space is a collection of the DHCP options with
different data types.

To create the sub-option in the defined option space, use the adddhcoption
CLI and mention option_type as sub-option.

ARGUMENTS:

```
--name
      Name of the DHCP option space. [mandatory]
--vendor_class
      Name of the vendor class.

--desc
      Description for DHCP option space.
```

EXAMPLE USAGE:

```
twc adddhcoptionspace --name=space
```

```
twc adddhcoptionspace --name=space --vendor_class="vendor class" --desc=description
```

adddnsreversezone

NAME:

adddnsreversezone

DESCRIPTION:

Creates a DNS reverse zone in the TCPWave IPAM. IP address, mask length (an integer between 8 and 32) and zone template name are mandatory fields to be given as input to create a DNS reverse zone for a subnet.

ARGUMENTS:

```
--ip
      IP address of the subnet. [mandatory]
--mask
      Mask length of the subnet (an integer between 8 and 32) [mandatory]
--org
      Organization name to be associated with the DNS reverse zone. [mandatory]
```

--zone_tmpl

Zone template name to be associated with the DNS reverse zone

--dnssec

'1' indicates that DNSSEC must be enabled for the zone. '0' indicates that DNSSEC must be disabled. If the argument is not specified, the value is defaulted to '0'.

--nsec_opt

NSEC option for the reverse zone. Takes 'NSEC' or 'NSEC3' as values.

--monit

'1' indicates that the monitoring must be enabled for this zone. '0' indicates that the monitoring must be disabled for this zone. If the argument is not specified, the value is defaulted to '1'.

--dmz_visible

'1' indicates that the zone must be visible to the cache server rooted at a public internet root server. '0' indicates that the zone must not be visible. If this argument is not specified the value is defaulted to '0'.

--ext_attr

Comma separated list of extension attributes with their values in the format : extension_attribute_name/extension_attribute_value. Use the following command to see all the extension attributes applied to zone :

'twc listext --entity=revzone --d=,'

--views

Comma separated list of DNS view names to be associated with DNS reverse zone.

--contact_fname

First name of the contact to be associated with the DNS reverse zone.

--contact_mname

Middle name of the contact to be associated with the DNS reverse zone.

--contact_lname

Last name of the contact to be associated with the DNS reverse zone.

--contact_email

Email Id of the contact to be associated with the DNS reverse zone.

--custom_allow_ns

Custom allow NS should be specified as FQDN. It accepts the server name and IP address by separating them with comma. It accepts multiple values by separating them with pipe symbol.

Example: test1.com.,10.1.10.1|test2.com.,192.168.0.0

--tsig_key_names

It accepts the multiple TSIG key values by separating with comma. Custom allow NS cannot be null to specify TSIG key names.

Example: key1,key2

--desc

Description for the DNS reverse Zone.

EXAMPLE:

```
twc adddnsreversezone --ip=10.0.0.0 --org=TCPwave --mask=16 --
zone_tmpl="TestZoneTemplate" --desc="Reverse Zone"
```

```
twc adddnsreversezone --ip=10.0.0.0 --org=TCPwave --mask=16 --dnssec=1 --nsec_opt=NSEC3 -
-zone_tmpl="TestZoneTemplate" --dmz_visible=1 --desc="Reverse Zone"
```

```
twc adddnsreversezone --ip=10.0.0.0 --org=TCPwave --mask=16 --
zone_tmpl="TestZoneTemplate" --desc="Reverse Zone" --
ext_attr=ext_attr_1/value_1,ext_attr_2/value_2
```

```
twc adddnsreversezone --ip=10.0.0.0 --org=TCPwave --mask=16 --
zone_tmpl="TestZoneTemplate" --dmz_visible=1 --contact_fname=John --contact_lname=Smith --
contact_email=john.smith@tcpwave.com --desc="TCPWave Reverse Zone"
```

```
twc adddnsreversezone --ip=10.0.0.0 --org=TCPwave --mask=16 --
zone_tmpl="TestZoneTemplate" --views=view1/view2 --desc="Reverse Zone"
```

```
twc adddnsreversezone --ip=10.0.0.0 --org=TCPwave --mask=16 --
zone_tmpl="TestZoneTemplate" --
custom_allow_ns="TCPWave.com.,10.1.10.1|NS1.com.,192.168.0.0" --tsig_key_names=key1,key2 --
desc="TCPWave Reverse Zone"
```

addipv6dnsreversezone

NAME

addipv6dnsreversezone

DESCRIPTION

Creates an IPv6 DNS reverse zone in the TCPWave IPAM. IPv6 address, mask length (an integer between 4 and 128) and organization name are mandatory fields to be given as input to create a DNS reverse zone
for a subnet.

ARGUMENTS:

--ip

IPv6 address of the subnet. [mandatory]

--name

Name of IPv6 DNS reverse zone. [mandatory]

--mask

Mask length of the subnet (an integer between 4 and 128). [mandatory]

--org

Organization name to be associated with the DNS reverse zone. [mandatory]

--zone_tmpl

Zone template name to be associated with the DNS reverse zone

--dnssec

'1' indicates that DNSSEC must be enabled for the zone. '0' indicates that DNSSEC must be disabled. If the argument is not specified, the value is defaulted to '0'.

--nsec_opt

NSEC option for the reverse zone. Takes 'NSEC' or 'NSEC3' as values.

--monit

'1' indicates that the monitoring must be enabled for this zone. '0' indicates that the monitoring must be disabled for this zone. If the argument is not specified, the value is defaulted to '1'.

--dmz_visible

'1' indicates that the zone must be visible to the cache server rooted at a public internet root server. '0' indicates that the zone must not be visible. If this argument is not specified the value is defaulted to '0'.

--ms_ad_integrate

Microsoft AD integration applicable only when the selected DNS zone template has Microsoft DNS appliance as master. It accepts '1' or '0' as input. '1' indicates zone on the Microsoft DNS Appliance will be changed to AD integrated zone. '0' indicates zone on the Microsoft DNS appliance will be changed to Standard Zone.

--views

Comma separated list of DNS view names to be associated with DNS reverse zone.

--contact_fname

First name of the contact to be associated with the DNS reverse zone.

--contact_mname

Middle name of the contact to be associated with the DNS reverse zone.

--contact_lname

Last name of the contact to be associated with the DNS reverse zone.

--contact_email

Email Id of the contact to be associated with the DNS reverse zone.

--custom_allow_ns

Custom allow NS should be specified as FQDN. It accepts the server name and IP address by separating them with comma. It accepts multiple values by separating them with pipe symbol.

Example: test1.com.,5455::4|test2.com.,3333::3

--is_tsig

'1' indicates tsig is enabled.'0' indicates tsig is disabled.

--desc

Description for the DNS reverse Zone.

EXAMPLE:

```
twc addipv6dnsreversezone --ip=5000:: --name=0.0.0.0.5.ip6.arpa --org=TCPwave --mask=16  
--zone_tmpl="TestZoneTemplate" --desc="Reverse Zone"
```

```
twc addipv6dnsreversezone --ip=5000:: --name=0.0.0.0.5.ip6.arpa --org=TCPwave --mask=16  
--dnssec=1 --nsec_opt=NSEC3 --zone_tmpl="TestZoneTemplate" --dmz_visible=1 --desc="Reverse  
Zone"
```

```
twc addipv6dnsreversezone --ip=5000:: --name=0.0.0.0.5.ip6.arpa --org=TCPwave --mask=16  
--zone_tmpl="TestZoneTemplate" --desc="Reverse Zone" --dnssec=1
```

```
twc addipv6dnsreversezone --ip=5000:: --name=0.0.0.0.5.ip6.arpa --org=TCPwave --mask=16  
--zone_tmpl="TestZoneTemplate" --dmz_visible=1 --contact_fname=John --contact_lname=Smith  
--contact_email=john.smith@tcpwave.com --desc="TCPWave Reverse Zone"
```

```
twc addipv6dnsreversezone --ip=5000:: --name=0.0.0.0.5.ip6.arpa --org=TCPwave --mask=16  
--zone_tmpl="TestZoneTemplate" --views=view1,view2 --desc="Reverse Zone"
```

```
twc addipv6dnsreversezone --ip=5000:: --name=0.0.0.0.5.ip6.arpa --org=TCPwave --mask=16  
--zone_tmpl="TestZoneTemplate"  
custom_allow_ns="TCPWave.com.,10.1.10.1|NS1.com.,192.168.0.0" --is_tsig=1 --desc="TCPWave  
Reverse Zone"
```

adddnsrootzone

NAME:

adddnsrootzone

DESCRIPTION:

Create a DNS root zone in the TCPWave IPAM. Time formats are specified, as integer/time_unit. Time_unit can be one of the following values: S, MIN, H,D,W,MON,Y representing seconds, minutes, hours, days, weeks, months, years
Example: 84600/S or 30/D

ARGUMENTS:**--org**

Organization name under which the DNS root zone is being created.
[mandatory]

--desc

Description for the DNS root zone.

--dnssec

Takes '1' or '0'. '1' indicates that DNSSEC should be enabled for the root zone. '0' indicates that DNSSEC is not enabled. [mandatory]

--nsec_opt

NSEC option for the root zone. Takes 'NSEC' or 'NSEC3' as values. This argument should be specified, if DNSSEC is enabled.

--default_ttl

Default TTL for the root zone. Should be specified, in time format as described in the description.

--soa_email

Email id associated with the SOA record for the DNS root zone.[mandatory]

--soa_refresh

Refresh time associated with the SOA record for the DNS root zone. Should be specified, in time format asdescribed in the description.

--soa_retry

Retry time associated with the SOA record for the DNS root zone. Should be specified, in time format asdescribed in the description.

--soa_expiry

Expiry time associated with the SOA record for the DNS root zone. Should be specified, in time format asdescribed in the description.

--soa_negcache

Negative Cache time associated with the SOA record for the DNS root zone. Should be specified, in time format asdescribed in the description.

--allow_query

ACL for query operation. Takes a comma separated list of ACL elements in one of the following formats:

IP Address/permission (192.168.0.1/Allow)
 ACL-name/permission (internal/Deny)
 IP Address/mask/permission (192.168.0.0/24/Allow)

--masters

Comma separated list of IP addresses of authoritative servers acting as masters for the DNS root zone [mandatory]

--is_proxy

DNS Proxy root zone flag. It takes '0' or '1'. If it is specified, as '1' proxy root zone is created. If it is not specified, or specified, as '0' root zone is created.

--contact_first_name

First name field of the associated contact information for the root/proxy root zone.

--contact_middle_name

Middle name field of the associated contact information for the root/proxy root zone.

--contact_last_name

Last name field of the associated contact information for the root/proxy root zone.

--contact_email

Email ID field of the associated contact information for the root/proxy root zone.

EXAMPLE USAGE:

```
twc adddnsrootzone --org=TCPWave --default_ttl=84600/S --dnssec=1 --nsec_opt=NSEC3 --
soa_email=john.smith@tcpwave.com      --soa_refresh=21600/S      --soa_retry=3600/S      --
soa_expiry=604800/S      --soa_negcache=86400/S      --masters=192.168.1.10,192.168.1.11      --
allow_query="192.168.0.1/Allow,192.168.1.0/24/Deny"      --contact_first_name=John      --
contact_last_name=Smith --contact_email=john.smith@tcpwave.com --desc="TCPWave root zone" --
-is_proxy=0
```

```
twc adddnsrootzone --org=TCPWave --default_ttl=84600/S --soa_email=john.smith@tcpwave.com --
soa_refresh=21600/S      --soa_retry=3600/S      --soa_expiry=604800/S      --soa_negcache=86400/S      --
masters=192.168.1.10,192.168.1.11      --allow_query="192.168.0.1/Allow,192.168.1.0/24/Deny"      --
contact_first_name=John --contact_last_name=Smith --contact_email=john.smith@tcpwave.com --desc="TCPWave root zone" --is_proxy=1
```

adddnszone

NAME

adddnszone

DESCRIPTION

Creates a DNS Zone in the TCPWave IPAM.

ARGUMENTS

--name

Name of the DNS Zone [mandatory]

--org

Organization name associated with the DNS zone. [mandatory]

--zone_tmpl

Zone template name associated with the DNS zone. [mandatory]

--dnssec

1 indicates that DNSSEC should be enabled for the zone. 0 indicates that DNSSEC is not enabled. [mandatory]

--nsec_opt

NSEC option for the zone. Takes 'NSEC' or 'NSEC3' as values.

--ad_upd

'1' indicates Active Directory updates are enabled for this zone. '0' indicates Active Directory updates are disabled for this zone. If this argument is not specified the value is defaulted to '0'.

--dc_ip

Comma separated values of IPs of domain controllers applicable for this zone. This argument should be specified if ad_upd argument is specified as '1'.

--ad_sec

'1' indicates Active Directory secure updates are enabled for this zone. '0' indicates Active Directory secure updates are disabled for this zone. If this argument is not specified the value is defaulted to '0'.

--ad_forest

Indicates active directory forest. It takes either 'parent' or 'child'. 'parent' indicates active directory zone is a standalone parent forestzone. 'child' indicates active directory zone is a child forest zone.

--ms_ad_integrate

Microsoft AD integration applicable only when the selected DNS zone template has Microsoft DNS appliance as master. It accepts '1' or '0' as input. '1' indicates zone on the Microsoft DNS Appliance will be changed to AD integrated zone. '0' indicates zone on the Microsoft DNS appliance will be changed to Standard Zone.

--parent_forest

Indicates active directory parent forest for the child forest. This argument is mandatory if ad_forest argument is specified as 'child'.

--monit

'1' indicates monitoring is enabled for this zone. '0' indicates monitoring is disabled for this zone. If this argument is not specified the value is defaulted to '1'.

--dmz_visible

'1' indicates that the zone is visible to cache server rooted at a public internet root server. '0' indicates that the zone is not visible. If this argument is not specified the value is defaulted to '0'.

--acl

Comma separated list of ACL names. This argument should be specified if ad_upd argument is specified as '1' and ad_sec is 0.

--is_restricted

Restricted zone flag. It takes '1' or '0'. '1' indicates that the zone is created as restricted zone. '0' indicates zone is created as a non-restricted zone.

--views

Comma separated list of DNS view names to be associated with this zone.

--ext_attr

Comma separated list of extension attributes with their values in the format : extension_attribute_name/extension_attribute_value. Use the following command to see all the extension attributes applied to zone :

```
'twc listext --entity=zone --d=,'
```

--import_cloud_rr

Import cloud DNS resource records flag. It takes '1' or '0'. If it is '1' and specified zone template is associated with cloud provider(s), all the cloud managed resource records will be imported to IPAM if the zone already exists in the cloud provider(s).

--contact_fname

First name field of the associated contact information for the DNS zone.

--contact_mname

Middle name field of the associated contact information for the DNS zone.

--contact_lname

Last name field of the associated contact information for the DNS zone.

--contact_email

Email ID field of the associated contact information for the DNS zone.

--custom_allow_ns

Custom allow NS should be specified as FQDN. It accepts the server name and IP address by separating them with comma. It accepts multiple values by separating them with pipe symbol.

Example: test1.com.,10.1.10.1|test2.com.,192.168.0.0

--tsig_key_names

It accepts the multiple TSIG key values by separating with comma. Custom allow NS cannot be null to specify TSIG key names.

Example: key1,key2

--desc

Description for the DNS Zone.

EXAMPLE:

```
twc adddnszone --name="tcpwave.com" --org=TCPWave --zone_tmpl="base-zone-template" --dnssec=1 --nsec_opt=NSEC --ad_upd=0 --monit=0 --dmz_visible=0 --is_restricted=0 --views=view1/view2 --desc="TCPWave Zone"
```

```
twc adddnszone --name="tcpwave.com" --org=TCPWave --zone_tmpl="base-zone-template" --dnssec=1 --nsec_opt=NSEC --ad_upd=1 --ad_forest=parent --dc_ip=10.1.10.172,10.0.10.50 --acl=none,any --ad_sec=0 --monit=0 --is_restricted=0 --desc="TCPWave Zone"
```

```
twc adddnszone --name="tcpwave.com" --org=TCPWave --zone_tmpl="base-zone-template" --dnssec=1 --nsec_opt=NSEC --ad_upd=0 --monit=0 --dmz_visible=0 --is_restricted=0 --views=view1/view2 --ext_attr=ext_attr_1/value_1,ext_attr_2/value_2 --desc="TCPWave Zone"
```

```
twc adddnszone --name="tcpwave.com" --org=TCPWave --zone_tmpl="base-zone-template" --dnssec=0 --ad_upd=0 --monit=0 --is_restricted=0 --import_cloud_rr=1 --desc="TCPWave Zone"
```

```
twc adddnszone --name="tcpwave.com" --org=TCPWave --zone_tmpl="base-zone-template" --dnssec=1 --nsec_opt=NSEC --ad_upd=1 --ad_forest=parent --dc_ip=10.1.10.173 --acl=acl1 --ad_sec=0 --monit=0 --is_restricted=0 --desc="TCPWave Zone"
```

```
twc adddnszone --name="dev.tcpwave.com" --org=TCPWave --zone_tmpl="base-zone-template" --dnssec=1 --nsec_opt=NSEC --ad_upd=1 --ad_forest=child --parent_forest=tcpwave.com --dc_ip=10.1.10.172,10.0.10.50 --ad_sec=1 --monit=0 --is_restricted=0 --desc="TCPWave Zone"
```

```
twc adddnszone --name="tcpwave.com" --org=TCPWave --zone_tmpl="base-zone-template" --dnssec=1 --nsec_opt=NSEC --ad_upd=0 --monit=0 --dmz_visible=0 --is_restricted=0 --
```

```
contact_fname=John --contact_lname=Smith --contact_email=john.smith@tcpwave.com --
desc="TCPWave Zone"
```

```
twc adddnszone --name="tcpwave.com" --org=TCPWave --zone_tmpl="base-zone-template" --
dnssec=1 --nsec_opt=NSEC --ad_upd=0 --monit=0 --dmz_visible=0 --is_restricted=0 --
contact_fname=John --contact_lname=Smith --contact_email=john.smith@tcpwave.com --
custom_allow_ns="TCPWave.com.,10.1.10.1|NS1.com.,192.168.0.0" --tsig_key_names=key1,key2 --
desc="TCPWave Zone"
```

adddnsview

NAME:

adddnsview - Creates a DNS view in the TCPWave IPAM.

DESCRIPTION:

Creates a DNS view in the TCPWave IPAM.

ARGUMENTS:

--org

Organization name under which the DNS view is being created. This argument is mandatory if the user is FADM.

--name

Name of the DNS view. [mandatory]

--geoip_type

Takes the input as '0' or '1'. It's default value is '0'.

--match_clients

Comma separated list of match clients in one of the following formats:

IPAddress/permission (192.168.0.1/Allow)

ACL-name/permission (internal/Deny)

Takes the input as comma separated list of match clients in the below format when the geoip_type=1.

country-code/permission (AD/Allow)

--match_destinations

Comma separated list of destination in one of the following formats:

IPAddress/permission (192.168.0.1/Allow)

ACL-name/permission (internal/Deny)

EXAMPLE:

```
twc adddnsview --org=TCPWave --name=view1 --match_clients="10.0.0.10/Allow,none/Deny" --
geoip_type=0
twc adddnsview --org=TCPWave --name=view1 --match_clients="AD/Allow,IN/Deny,AE/Deny" --
geoip_type=1
twcadddnsview--org=TCPWave--name=view1
match_destinations="10.0.0.10/Allow,10.0.0.10/Deny,internal/Deny"
adddnszonetmpl
```

NAME:**adddnszonetmpl****DESCRIPTION:**

Create a DNS zone template in the TCPWave IPAM. Time formats are specified as integer/time_unit. time_unit can be one of the following values: S,MIN,H,D,W,MON,Y representing seconds, minutes, hours, days, weeks, months, years Example: 84600/S or 30/D

ARGUMENTS:**--name**

Name of the DNS zone template. [mandatory]

--org

Organization name associated with the zone template. [mandatory]

--default_ttl

Default TTL for the zone. Should be specified in time format as described in the description.

--soa_email

Email id associated with the SOA record for the zone. [mandatory]

--soa_refresh

Refresh time associated with the SOA record for the zone. Should be specified in time format as described in the description.

--soa_retry

Retry time associated with the SOA record for the zone. Should be specified in time format as described in the description.

--soa_expiry

Expiry time associated with the SOA record for the zone. Should be specified in time format as described in the description.

--soa_negcache

Negative Cache time associated with the SOA record for the zone. Should be specified in time format as described in the description.

--allow_query

ACL for query operation. Takes a comma separated list of ACL elements in one of the following formats:

IP Address/permission (192.168.0.1/Allow)

ACL-name/permission (internal/Deny)

IP Address/mask/permission (192.168.0.0/24/Allow)

--allow_update

ACL for custom allow update operation. Takes a comma separated list of ACL elements in one of the following formats:

Ex: localhost,localnets

--allow_transfer

ACL for custom allow transfer operation. Takes a comma separated list of ACL elements in one of the following formats:

Ex: none,any

--also_notify

Input for custom also notify. Takes a comma separated list of IPV4 address in one of the following formats:

IP Address/permission (192.168.0.1/Allow)

IP Address/mask/permission (192.168.0.0/24/Allow)

--masters

Comma separated list of IP addresses of authoritative servers acting as masters for the zone. [mandatory]

--slaves

Comma separated list of IP addresses of authoritative servers acting as slaves for the zone.

--ms_dns_master

Comma separated list of IP addresses of Microsoft DNS Master appliances acting as masters for the zone.

--empty_forwarders

Takes '1' or '0'. '1' indicates that zones will be generated with empty forwarders in the zone sections. '0' indicates that no empty forwarders for zones. Default 0.

--desc

Description for the DNS zone template.

EXAMPLE:

```
twc adddnszonetmpl --name="base-zone-template-1" --desc="base zone template" --soa_email=jsmith@tcpwave.com --soa_refresh=21600/S --soa_retry=3600/S --soa_expiry=604800/S --soa_negcache=86400/S --allow_query=192.168.0.1/Allow --allow_update=localhost,localnets --masters=10.1.10.190 --slaves=10.1.10.188 --org=Internal --empty_forwarders=1
```

```
twc adddnszonetmpl --name="base-zone-template-2" --desc="base zone template" --soa_email=john.smith@tcpwave.com --soa_refresh=21600/S --soa_retry=3600/S --soa_expiry=604800/S --soa_negcache=86400/S --masters=192.168.1.10,192.168.1.11 --slaves=192.168.1.12,192.168.1.13 --org=TCPWave
```

```
twc adddnszonetmpl --name="base-zone-template-3" --desc="base zone template" --soa_email=jsmith@tcpwave.com --soa_refresh=21600/S --soa_retry=3600/S --soa_expiry=604800/S --soa_negcache=86400/S --allow_transfer=test,any --ms_dns_master=10.1.10.190 --org=Internal --empty_forwarders=1
```

adddomain**NAME:**

adddomain

DESCRIPTION:

Creates a DNS domain in the TCPWave IPAM.

On the Internet, a domain is part of every network address, including website addresses, email addresses, and addresses for other Internet protocols such as FTP, IRC, and SSH. All devices sharing a common part of an address, or URL, are said to be in the same domain.

ARGUMENTS:

--org

Organization name under which the domain is being created. This argument is mandatory if user is FADM.

--name

Name of the domain being created. [mandatory]

--desc

Description for the domain being created.

EXAMPLE USAGE:

```
twc adddomain --org=TCPWave --name="tcpwave.com" --desc="TCPWave Primary Domain"
```

adddiscoverytmpl**NAME:**

adddiscoverytmpl

DESCRIPTION:

Creates a discovery template in the TCPWave IPAM.

ARGUMENTS:

--discovery_tmpl

Name of the discovery template. [mandatory]

--org

Name of the organization. [mandatory]

--discovery_method

Discovery method takes the input

"PING,SNMP,NMAP,Reverse DNS,NetBios,Switch,VMWare" as comma separated.

[mandatory]

--snmp_req_retries

SNMP request retries. takes '1' to '5' only.

--snmp_res_timeout
SNMP Response Timeout (In milliseconds).

--rev_dnslookup_timeout
Reverse DNS Lookup Timeout (In milliseconds).

--non_conflicting_objects
Add non conflicting objects, specifies 'yes' or 'no'. [mandatory]

--conflicting_objpreference
Conflicting Object Preferences takes the input
'1' = Flag For Manual Reconciliation.
'2' = Ignore The Conflicting Discovered Object.
'3' = Overwrite with Discovered Object. [mandatory]

--reclaim_grace_duration
Reclaim Duration (In days).

--description
Description of the discovery template.

EXAMPLE USAGE:

```
twc adddiscoverytmpl --discovery_tmpl=TestTmpl --org=TCPWave --
discovery_method="PING,SNMP,NMAP,Reverse DNS" --snmp_req_retries=2 --
snmp_res_timeout=2000 --rev_dnslookup_timeout=2000 --non_conflicting_objects=yes --
conflicting_objpreference=2 --reclaim_grace_duration=2 --description="Discovery template"
```

adext**NAME:**

adext

DESCRIPTION:

Creates an extended attribute in the TCPWave IPAM.

ARGUMENTS:

--name
Name of the extension attribute. [mandatory]

--type
Data type of the extension attribute. It takes one of 'string', 'date', 'numeric', 'ip' or 'list'. [mandatory]

--list_values
Values of the list type extension attribute. It takes comma separated list of strings. This argument is applicable only when type argument is 'list'.

--entities

Entities for which extension attribute to be applied. It takes comma separated list of entity in format: entity/show_grid_flag. Allowable entities are 'admin', 'network', 'subnet', 'object', 'zone'. show_grid_flag takes 1 or 0. Examples are admin/1, network/0

--mandatory

It takes '1' or '0'. '1' indicates that extension attribute value is mandatory while creating the target entities. '0' indicates that extension attribute value is not mandatory while creating the target entities.

--desc

Description of the extension attribute being created.

EXAMPLE USAGE:

```
twc addext --name=RITS_ID --type=numeric --entities=admin/1 --mandatory=0 --desc="RIT ID admin"
```

```
twc addext --name=CHG_TKT --type=string --entities=admin/1, network/0 --mandatory=0
```

addipamtoha**NAME**

addipamtoha

DESCRIPTION

Adds a TCPWave IPAM to high availability cluster.

ARGUMENTS:**--ip**

IP address of the target IPAM. [mandatory]

--login_name

Login name of FADM user on the target IPAM. [mandatory]

EXAMPLE USAGE

```
twc addipamtoha --ip=172.16.0.172 --login_name=twcadm
```

addipv6network**NAME**

addipv6network

DESCRIPTION

Creates an IPv6 network in the TCPWave IPAM.

ARGUMENTS:**--network**

IPv6 address of the target network. [mandatory]

--mask
Mask length of the network prefix (an integer between 4 and 64).
[mandatory]

--name
Name of the network. [mandatory]

--org
Organization name under which the network is being created. This argument is mandatory if the user is FADM.

--desc
Description of the network.

--dnssec
1 indicates that dnssec should be enabled for the reverse zone. 0 indicates that dnssec is not enabled.

--nsec_opt
NSEC option for the reverse zone. Takes 'NSEC' or 'NSEC3' as values.

--zone_name_tmpl
Zone template name associated with the reverse zone.

--dmz_visible
'1' indicates that the zone is visible to the cache server rooted at a public internet root server. '0' indicates that the zone is not visible. If this argument is not specified, the value is defaulted to '0'.

EXAMPLE USAGE:

```
twc addipv6network --network=2000:: --name="TCPWave network" --org=TCPwave --mask=16 - -desc="TCPWave Network"
```

```
twc addipv6network --network=ff00:: --name="TCPWave network" --org=TCPwave --mask=16 --dnssec=1 --nsec_opt=NSEC --zone_name_tmpl="TestZoneTemplate" --dmz_visible=1 --desc="TCPWave Network"
```

addipv6object**NAME:**

addipv6object

DESCRIPTION:

Creates an IPv6 object in the TCPWave IPAM. The syntax of this command is shown below. The user invoking this command is expected to have authentication permission and should be authorized to perform add object. The TCPWave IPAM audits this action. Successful completion of this command exits with a status code 0.

ARGUMENTS:

--object
IPv6 address of the object to be created. [mandatory]

--name
Name of the object. [mandatory]

--org
Organization name under which the object is being created. This argument is mandatory if user is FADM.

--subnet
IPv6 address of the subnet in which user wants to create the object. [mandatory]

--desc
Description for the target object.

--alloc_type
Address allocation type. It takes one of 'Static', 'Manual', 'Dynamic' or 'Auto'.

--class_code
Class code of the target object [mandatory]

--domain
Domain name to be associated with the object. [mandatory]

--mac
MAC address of the target object.

--ttl
Time-to-Live in seconds for the target object.

--opt_tmpl
IPv6 Option Template Name associated with the target object. Should be specified, for objects of allocation type manual or dynamic.

--dhcp_appliance
IPv6 DHCP Server Name associated with the target object. Should be specified, for objects of allocation type manual or dynamic.

--ns_a
Takes '0' or '1'. 1 indicates that the corresponding A resource record must be updated in the name service zone file. 0 indicates that the corresponding A resource record will not be updated in the name service zone file. A default value of 1 will be used if not specified.,

--ns_ptr
Takes '0' or '1'. 1 indicates that the corresponding PTR resource record must be updated in the name service zone file. 0 indicates that the corresponding PTR resource record will not be updated in the name

service zone file. A default value of 1 will be used if not specified.,

--ddns_a

Takes '0' or '1'. 1 indicates that dynamic DNS updates are allowed to the corresponding A resource record in the name service zone file. 0 indicates that dynamic DNS updates are not allowed to the corresponding A resource record in the name service zone file. A default value of 1 will be used if not specified.,

--ddns_ptr

Takes '0' or '1'. 1 indicates that dynamic DNS updates are allowed to the corresponding PTR resource record in the name service zone file. 0 indicates that dynamic DNS updates are not allowed to the corresponding PTR resource record in the name service zone file. A default value of 1 will be used if not specified.,

--ddns cname

Takes '0' or '1'. 1 indicates that dynamic DNS updates are allowed to the corresponding CNAME resource record in the name service zone file. 0 indicates that dynamic DNS updates are not allowed to the corresponding CNAME resource record in the name service zone file. A default value of 1 will be used if not specified.,

--ddns_mx

Takes '0' or '1'. 1 indicates that dynamic DNS updates are allowed to the corresponding MX resource record in the name service zone file. 0 indicates that dynamic DNS updates are not allowed to the corresponding MX resource record in the name service zone file. A default value of 1 will be used if not specified.,

--contact_first_name

First Name field of the associated contact information for the object.

--contact_last_name

Last Name field of the associated contact information for the object.

--contact_email

Email Id field of the associated contact information for the object.

--f

Flag for the Ping operation before object creation. Takes '0' or '1'. If it is not specified, or specified, as '0', a warning will be given if a live object is present before creating object. If it is specified, as '1' ping operation will not be performed.

EXAMPLE USAGE:

```
twc addipv6object --object=2001::36 --name="PC-273565" --subnet=2001:: --class_code=PC --domain=tcpwave.com --alloc_type=static --org=TCPWave --mac=01:23:45:67:89:ab --ttl=300 --ns_a=1 --ns_ptr=1 --ddns_a=1 --ddns_ptr=1 --ddns_cname=1 --ddns_mx=1
```

```
tvc addipv6object --object=2001::36 --name="PC-273565" --subnet=2001:: --class_code=PC --domain=tcpwave.com --alloc_type=Dynamic --org=TCPWave --opt_tmpl=Generic --dhcp_appliance="nusalx-trv10-sl098" --mac=01:23:45:67:89:ab --ttl=300 --ns_a=1 --ns_ptr=1 --ddns_a=1 --ddns_ptr=1 --ddns_cname=1 --ddns_mx=1
```

```
tvc addipv6object --object=2001::36 --name="Server-2733664" --subnet=2001:: --class_code="TCPWave Remote" --domain=tcpwave.com --org=TCPWave --alloc_type=static --mac=01:23:45:67:89:ad --desc="Internal DNS Server" --ttl=300 --ns_a=1 --ns_ptr=1 --ddns_a=1 --ddns_ptr=1 --ddns_cname=1 --ddns_mx=1
```

addipv6scope

NAME

addipv6scope

DESCRIPTION

Creates an IPv6 DHCP scope in the TCPWave IPAM.

ARGUMENTS

--subnet

IPv6 subnet address where the scope is to be created [mandatory]

--scope_size

Scope size to be created [mandatory]

--no_of_scope

Number of scope to be created [mandatory]

--obj_type

Valid class code in TCPWave IPAM to be associated with the DHCP scope [mandatory]

--ns_aaaa

Takes 0 | 1. 1 indicates that the corresponding A resource record must be updated in the name service zone file. 0 indicates that the corresponding A resource record will not be updated in the name service zone file. A default value of 1 will be used if not specified.,

--ns_ptr

Takes 0 | 1. 1 indicates that the corresponding PTR resource record must be updated in the name service zone file. 0 indicates that the corresponding PTR resource record will not be updated in the name service zone file. A default value of 1 will be used if not specified.,

--ddns_aaaa

Takes 0 | 1. 1 indicates that dynamic DNS updates are allowed to the corresponding A resource record in the name service zone file. 0 indicates that dynamic DNS updates are not allowed to the corresponding A resource record in the name service zone file. A default value of 1 will be used if not specified.,

--ddns_ptr

Takes 0|1. 1 indicates that dynamic DNS updates are allowed to the corresponding PTR resource record in the name service zone file.
 0 indicates that dynamic DNS updates are not allowed to the corresponding PTR resource record in the name service zone file.
 A default value of 1 will be used if not specified.,

--ddns cname

Takes 0|1. 1 indicates that dynamic DNS updates are allowed to the corresponding CNAME resource record in the name service zone file.
 0 indicates that dynamic DNS updates are not allowed to the corresponding CNAME resource record in the name service zone file.
 A default value of 1 will be used if not specified.,

--ddns_mx

Takes 0|1. 1 indicates that dynamic DNS updates are allowed to the corresponding MX resource record in the name service zone file.
 0 indicates that dynamic DNS updates are not allowed to the corresponding MX resource record in the name service zone file.
 A default value of 1 will be used if not specified.,

--allow_client

DHCP client class for which leases can be granted from this scope.

--allow_user

DHCP user class for which leases can be granted from this scope.

--allow_vendor

DHCP vendor class for which leases can be granted from this scope.

--deny_client

DHCP client class for which leases are not granted from this scope.

--deny_user

DHCP user class for which leases are not granted from this scope.

--deny_vendor

DHCP vendor class for which leases are not granted from this scope.

--desc

Description for the scope.

--org

Name of the organization to which the objects belongs. This argument is mandatory if user is 'FADM'.

EXAMPLE USAGE:

```
twc addipv6scope --subnet=5000:0:0:20:: --scope_size=4 --no_of_scope=4 --obj_type="3G Phone"
--ns_aaaa=1 --ns_ptr=1 --ddns_aaaa=1 --ddns_ptr=1 --ddns_cname=1 --ddns_mx=1 --
```

```
allow_client=clientclassX --allow_user=userclassX --allow_vendor=vendorclassX --
deny_client=clientclassY --deny_user=userclassY --deny_vendor=vendorclassY --org=TCPWave
```

addipv6subnet

NAME:

addipv6subnet

DESCRIPTION:

Creates a ipv6 subnet in the TCPWave the IPAM.

ARGUMENTS:

- subnet
IPv6 address of the subnet. [mandatory]
- mask
Mask length of the subnet. [mandatory]
- network
Start address of the associated network. [mandatory]
- org
Organization name of the associated Network. [mandatory]
- name
Name of the subnet being created.
- subnet_groupname
Name of the associated subnet group.
- domain
Domain to be associated with this subnet. [mandatory]
- router_addr
IPv6 address of the router associated with the subnet being created in case of a single subnet creation.
- street1
Street1 part of the location information.
- street2
Street2 part of the location information.
- city
City part of the location information.
- state
State part of the location information.
- zip
Zip code part of the location information.

--dhcp_tmpl

Template name specifying the DHCP options for the subnet. This argument is mandatory if dhcp_server argument is specified.,

--dhcp_appliance

Primary DHCP server for the subnet. This argument is mandatory if dhcp_tmpl argument is specified.,

--desc

Description text for the subnet.

EXAMPLE USAGE:

```
twc addipv6subnet --subnet=2001:0:0:b000:: --mask=64 --network=2001:: --name="Subnet-0001" --org=TCPWave --domain=tcpwave.com --router_addr=2001:0:0:b000::1
```

```
twc addipv6subnet --subnet=2001:0:0:f000:: --mask=64 --network=2001:: --name="Subnet-0001" --org=TCPWave --subnet_groupname=subnet-group --street1="600 ALEXANDER ROAD" --city="PRINCETON" --state=NJ --country=USA --zip=08540 --domain=tcpwave.com --dhcp_tmpl=dhcp-option-temp-1 --dhcp_appliance=dhcp-1 --router_addr=2001:0:0:f000::1 --desc="TCPWave Subnet"
```

addipv6subnetgroup**NAME**

addipv6subnetgroup

DESCRIPTION

Creates a subnet group in the TCPWave IPAM.

To create the IPv6 subnet group needs to provide the name of the template and organization.

ARGUMENTS**--name**

Name of the subnet group being created. [mandatory]

--org

Organization name of the subnet group. [mandatory]

--desc

Description of the subnet group.

EXAMPLE

```
twc addipv6subnetgroup --name=IT_SG --desc="IT subnet group" --org=TCPWave
```

addipv6block

NAME

addipv6block

DESCRIPTION

Creates an IPv6 address block in an IPv6 address pool defined in the TCPWave IPAM.

ARGUMENTS

--pool_ip

Address of the IPv6 address pool in which block has to be created. [mandatory]

--name

Name of the IPv6 address block. [mandatory]

--ip

Address of the target IPv6 address block. [mandatory]

--mask

Mask length of the IPv6 address block (an integer between 1 and 128). [mandatory]

--org

Organization name to which the IPv6 address block is being created [mandatory]

--zone_tmpl

Zone template name to be associated with the block reverse zone.

--dmz_visible

DMZ visibility flag. '1' indicates that the reverse zone is visible to the cache server rooted at a public internet root server. '0' indicates that the zone is not visible. If this argument is not specified the value is defaulted to '0'.

--desc

Description of the IPv6 address block.

--dnssec

DNSSEC flag. 1 indicates that DNSSEC will be enabled for the reverse zone. 0 indicates that DNSSEC is not enabled.

--nsec_opt

NSEC option for the reverse zone. Takes 'NSEC' or 'NSEC3' as values.

--contact_fname

First name field of the associated contact information for the IPv6 block.

--contact_mname

Middle name field of the associated contact information for the IPv6 block.

--contact_lname

Last name field of the associated contact information for the IPv6 block.

--contact_email

Email ID field of the associated contact information for the IPv6 block.

--discovery_tmpl

Discovery template name to be associated with the IPV6 block.

--vrf

Virtual routing and forwarding name to be associated with the IPV6 block.

--cloud_region

Cloud region for IPV6 block.

EXAMPLE

```
twc addipv6block --pool_ip=2001:db8:: --org=TCPwave --name="block_1" --ip=2001:db8:: --mask=64 --desc="TCPWave IPv6 address pool"
```

```
twc addipv6block --pool_ip=ff00:: --org=TCPwave --name="block_2" --ip=ff00:: --mask=72 --zone_tmpl="TestZoneTemplate" --dmz_visible=1 --dnssec=1 --nsec_opt=NSEC --desc="TCPWave IPv6 address block"
```

addipv6pool**NAME**

addipv6pool

DESCRIPTION

Creates an IPv6 address pool in the TCPWave IPAM.

ARGUMENTS

--region

Region of the IPv6 pool. [mandatory]

--name

Name of the IPv6 pool. [mandatory]

--ip

Address of the target IPv6 pool. [mandatory]

--mask

Mask length of the IPv6 pool (an integer between 1 and 128). [mandatory]

--org

Organization name to which the IPv6 pool is being created. [mandatory]

--zone_tmpl

Zone template name to be associated with the pool reverse zone.

--dmz_visible

DMZ visibility flag. '1' indicates that the reverse zone is visible to the cache server rooted at a public internet root server. '0' indicates that the zone is not visible. If this argument is not specified the value is defaulted to '0'.

--desc

Description of the IPv6 pool.

--dnssec

DNSSEC flag. 1 indicates that DNSSEC will be enabled for the reverse zone. 0 indicates that DNSSEC is not enabled.

--nsec_opt

NSEC option for the reverse zone. Takes 'NSEC' or 'NSEC3' as values.

--contact_fname

First name field of the associated contact information for the IPv6 pool.

--contact_mname

Middle name field of the associated contact information for the IPv6 pool.

--contact_lname

Last name field of the associated contact information for the IPv6 pool.

--contact_email

Email ID field of the associated contact information for the IPv6 pool.

--discovery_tmpl

Discovery template name to be associated with the IPv6 pool.

--vrf

Virtual routing and forwarding name to be associated with the IPv6 pool.

--cloud_region

Cloud region for IPv6 pool.

EXAMPLE

```
twc addipv6pool --region=USA --name="pool1" --ip=2001:db8:: --org=TCPwave --mask=32 --
desc="TCPWave IPv6 address pool"
```

```
twc addipv6pool --region=USA --name="pool2" --ip=ff00:: --org=TCPwave --mask=64 --
zone_tmpl="TestZoneTemplate" --dmz_visible=1 --dnssec=1 --nsec_opt=NSEC --desc="TCPWave IPv6
address pool"
```

```
twc addipv6pool --region=USA --name="pool3" --ip=2002:db8:: --org=TCPwave --mask=32 --
discovery_tmpl="descTemp" --vrf=CUCM1 --cloud_region="us-east-1"
```

addlocation

NAME:

addlocation

DESCRIPTION:

Creates a location for a given organization in the TCPWave IPAM.
Target organization must be specified, by name using --org parameter
All address fields except for Street2 (--street2) are mandatory.

ARGUMENTS:

--street1

Street1 field of the location address [mandatory]

--street2

Street2 field of the location address.

--city

City field of the location address [mandatory]

--state

State field of the location address [mandatory]

--country

Country field of the location address [mandatory]

--zip

Zip code field of the location address [mandatory]

--org

Organization name for which the location is being created [mandatory]

EXAMPLE USAGE:

```
twc addlocation --street1="600 ALEXANDER ROAD" --city="PRINCETON" --state=NJ --
country=USA --zip=08540 --org=TCPWave
```

```
twc addlocation --street1="600 ALEXANDER ROAD" --street2="Building 1" --city="PRINCETON" -
-state=NJ --country=USA --zip=08540 --org=TCPWave
```

addmirroredzone**NAME:**

addmirroredzone

DESCRIPTION:

Mirrored zone contains the same data as of its managed DNS zone in the IPAM. Number of mirrored zones to each managed DNS zone are limited to the value assigned to the global option.

ARGUMENTS:**--org**

Organization name under which the mirrored zones are being created.
This argument is mandatory if the user is FADM.

--zone_name

Name of the DNS zone for which the mirrored zones are being created.
[mandatory]

--name

Comma separated list of domain names to be created as mirrored zones.
[mandatory]

--desc

Description for the mirrored zones.

EXAMPLE USAGE:

```
tvc addmirroredzone --org=TCPWave --zone_name=tcpwave.com --  
name=tcpwave1.com,tcpwave2.com --desc="Mirrored zone of tcpwave.com"
```

```
tvc addmirroredzone --org=TCPWave --zone_name=tcpwave.com --name=tcpwave3.com --  
desc="Mirrored zone of tcpwave.com"
```

addmicrosoftdhcpserver**NAME:**

addmicrosoftdhcpserver

DESCRIPTION:

Creates Microsoft DHCP appliance in the TCPWave IPAM.

ARGUMENTS:**--addr**

IP address of the appliance. [mandatory]

--org

Name of the organization. [mandatory]

--use_https

Takes '0' or '1' as input.

```
--user_name
    User name for the Microsoft appliance. [mandatory]

--password
    Password for Microsoft appliance. [mandatory]

--mac_exclusion_addr
    Comma separated mac exclusion addresses without any spaces.

--desc
    Description for the Microsoft DHCP appliance.
```

EXAMPLE USAGE:

```
twc addmicrosoftdhcpserver --addr=10.0.0.10 --org=TCPWave --user_name=Administrator --
password=Apple!23 --mac_exclusion_addr=AA:BB:CC:DD:EE:F5,AA:BB:CC:DD:EE:F9
```

```
twc addmicrosoftdhcpserver --addr=10.0.0.10 --org=TCPWave --use_https=1 --
user_name=Administrator --password=Apple!23 --desc="First Microsoft Appliance"
```

addmicrosoftdnsserver**NAME:**

addmicrosoftdnsserver - Creates a Microsoft DNS appliance in the TCPWave IPAM.

DESCRIPTION:

Creates a Microsoft DNS appliance in the TCPWave IPAM.

ARGUMENTS:

```
--addr
    IP address of the appliance. [mandatory]

--org
    Name of the organization. [mandatory]

--use_https
    Takes '0' or '1' as input.

--user_name
    User name for the Microsoft appliance. [mandatory]

--password
    Password for Microsoft appliance. [mandatory]

--desc
    Description for the Microsoft DNS appliance.
```

EXAMPLE USAGE:

```
twc addmicrosoftdnsserver --addr=10.0.0.10 --org=TCPWave --user_name=profile --
```

password=window

```
twc addmicrosoftdnsserver --addr=10.0.0.10 --org=TCPWave --use_https=1 --user_name=user --password=abc1234 --desc="First Microsoft Appliance"
```

addnetwork

NAME:

addnetwork

DESCRIPTION:

Creates a network in the TCPWave IPAM.

ARGUMENTS:

--network

IP Address of the network. [mandatory]

--mask

Mask length of the network prefix (an integer between 8 and 32) [mandatory]

--org

Organization name under which the network is being created. This argument is mandatory if the user is FADM.

--name

Name of the network [mandatory]

--desc

Description of the network.

--dnssec

1 indicates that dnssec should be enabled for the reverse zone. 0 indicates that dnssec is not enabled.

--nsec_opt

NSEC option for the reverse zone. Takes 'NSEC' or 'NSEC3' as values.

--zone_tmpl

Zone template name associated with the reverse zone.

--dmz_visible

'1' indicates that the zone is visible to the cache server rooted at a public internet root server. '0' indicates that the zone is not visible. If this argument is not specified, the value is defaulted to '0'.

--ext_attr

Comma separated list of extension attributes with their values in the format : extension_attribute_name/extension_attribute_value. Use the following command to see all the extension attributes applied to network : 'twc listext --entity=network --d=,'

--monitoring

Enable monitoring services flag. Takes '1' or '0'. '1' indicates that monitoring service should be enabled for the network. '0' indicates that monitoring is not enabled.

--enable_discovery

Enable discovery flag. Takes '1' or '0'. '1' indicates that discovery should be enabled for the network. '0' indicates that discovery is not enabled.

--discovery_tmpl

Name of the discovery template. It is mandatory to specify the discovery template name if discovery is enabled.

--contact_first_name

First name field of the associated contact information for the network.

--contact_middle_name

Middle name field of the associated contact information for the network.

--contact_last_name

Last name field of the associated contact information for the network.

--contact_email

Email ID field of the associated contact information for the network.

EXAMPLE USAGE:

```
twc addnetwork --network=80.0.0.0 --name="TCPWave network" --org=TCPwave --mask=16 --dnssec=0 --desc="TCPWave Network"
```

```
twc addnetwork --network=80.0.0.0 --name="TCPWave network" --org=TCPwave --mask=16 --dnssec=1 --nsec_opt=NSEC --zone_tmpl="TestZoneTemplate" --dmz_visible=1 --desc="TCPWave Network"
```

```
twc addnetwork --network=80.0.0.0 --name="TCPWave network" --org=TCPwave --mask=16 --dnssec=0 --desc="TCPWave Network" --ext_attr=ext_attr_1/value_1,ext_attr_2/value_2
```

```
twc addnetwork --network=80.0.0.0 --name="TCPWave network" --org=TCPwave --mask=16 --dnssec=0 --monitoring=1 --enable_discovery=1 --discovery_tmpl="Discovery-Template" --desc="TCPWave Network"
```

```
twc addnetwork --network=80.0.0.0 --name="TCPWave network" --org=TCPwave --mask=16 --dnssec=1 --nsec_opt=NSEC --zone_tmpl="TestZoneTemplate" --dmz_visible=1 --contact_first_name=John --contact_last_name=Smith --contact_email=john.smith@tcpwave.com --desc="TCPWave Network"
```

addobject

NAME

addobject

DESCRIPTION

The twc addobject CLI command is used to add an object in the TCPWave IPAM. The syntax of this command is shown below. The user invoking this command is expected to have authentication permission and should be authorized to perform add object. The TCPWave IPAM audits this action. Successful completion of this command exits with a status code 0.

ARGUMENTS

--object

IP address of the target object. [mandatory]

--name

Name of the target object. [mandatory]

--org

Organization name under which the object is being created. [mandatory]

--alloc_type

Address allocation type. It takes one of 'Static', 'Manual', 'Dynamic', 'Reserved', or 'Auto'.

--class_code

Class code of the target object. [mandatory]

--domain

Domain name associated with the target object. [mandatory]

--mac

MAC address of the target object.

--ttl

Time-to-Live in seconds for the target object.

--view

Name of the DNS view. It accepts the comma separated DNS view names.

--opt_tmpl

Option Template Name associated with the target object. Should be specified for objects of allocation type manual, dynamic or auto.

--ns_a

Takes '0' or '1'. 1 indicates that the corresponding A resource record has to be updated in the name service zone file. 0 indicates that the corresponding A resource record will not be updated in the name service zone file. A default value of 1 will be used if not specified.

--ns_ptr

Takes '0' or '1'. 1 indicates that the corresponding PTR resource record has to be updated in the name service zone file. 0 indicates that the corresponding PTR resource record will not be updated in the name service zone file. A default value of 1 will be used if not specified.

--ddns_a

Takes '0' or '1'. 1 indicates that dynamic DNS updates are allowed to the corresponding A resource record in the name service zone file. 0 indicates that dynamic DNS updates are not allowed to the corresponding A resource record in the name service zone file. A default value of 1 will be used if not specified.

--ddns_ptr

Takes '0' or '1'. 1 indicates that dynamic DNS updates are allowed to the corresponding PTR resource record in the name service zone file. 0 indicates that dynamic DNS updates are not allowed to the corresponding PTR resource record in the name service zone file. A default value of 1 will be used if not specified.

--ddns_cname

Takes '0' or '1'. 1 indicates that dynamic DNS updates are allowed to the corresponding CNAME resource record in the name service zone file. 0 indicates that dynamic DNS updates are not allowed to the corresponding CNAME resource record in the name service zone file. A default value of 1 will be used if not specified.

--ddns_mx

Takes '0' or '1'. 1 indicates that dynamic DNS updates are allowed to the corresponding MX resource record in the name service zone file. 0 indicates that dynamic DNS updates are not allowed to the corresponding MX resource record in the name service zone file. A default value of 1 will be used if not specified.

--contact_fname

First Name field of the associated contact information for the object.

--contact_mname

Middle Name field of the associated contact information for the object.

--contact_lname

Last Name field of the associated contact information for the object.

--contact_email

Email Id field of the associated contact information for the object.

--street1

Street1 part of the location information.

--street2

Street2 part of the location information.

--city

City part of the location information.

--state

State part of the location information.

--zip

Zip code part of the location information.

--country

Country part of the location information.

--flag

Flag for the Ping operation before object creation. Takes '0' or '1'. If it is not specified or specified as '0', a warning will be given if a live object is present before creating object. If it is specified as '1' ping operation will not be performed.

--ext_attr

Comma separated list of extension attributes with their values in the format : extension_attribute_name/extension_attribute_value. Use the following command to see all the extension attributes applied to object:

'twc listext --entity=object --d=.'

--expiry_date

Expiry date applicable for Reserved Objects.

--cloud_instance_tmpl

Name of the cloud instance provisioning template.

--room

Room information for object location attribute.

--floor

Floor information for object location attribute.

--terminal_server_kvm

Terminal server kvm for the object.

--switch

Switch for the object.

--port

Port for the object.

--duplex

Duplex for the object. It accepts the input as any digit along with Mbps or Gbps, Ex: 100Mbps.

--vmware_attributes

VMWare Attributes are mandatory for VMWare ESXi and VMWare vCenter object types. It accepts port number, user name and password by separating with '|' symbol. Example: 7443|tcpwave|abc12345

--validate

To validate the VMWare attribute. It accepts '1' or '0'. '1' indicates to validate the VMWare Attributes. '0' indicates don't validate the VMWare Attributes.

--desc

Description for the target object.

EXAMPLE

```
twc addobject --object=10.0.0.10 --name="PC-273565" --class_code=PC --domain=tcpwave.com
--alloc_type=Dynamic --opt_tmpl=Generic --mac=01:23:45:67:89:ab --ttl=300 --ns_a=1 --ns_ptr=1 --
ddns_a=1 --ddns_ptr=1 --ddns_cname=1 --ddns_mx=1 --org=TCPWave
```

```
twc addobject --object=10.0.0.10 --name="Server-2733663" --class_code="TCPWave Remote" --
domain=tcpwave.com --alloc_type=static --mac=01:23:45:67:89:ac --desc="Internal DNS Server" --
ttl=300 --ns_a=1 --ns_ptr=1 --ddns_a=1 --ddns_ptr=1 --ddns_cname=1 --ddns_mx=1 --org=TCPWave
```

```
twc addobject --object=10.0.0.10 --name="Server-2733664" --class_code="TCPWave Remote" --
domain=tcpwave.com --alloc_type=static --mac=01:23:45:67:89:ad --desc="Internal DNS Server" --
ttl=300 --ns_a=1 --ns_ptr=1 --ddns_a=1 --ddns_ptr=1 --ddns_cname=1 --ddns_mx=1 --org=TCPWave
```

```
twc addobject --object=10.0.0.10 --name="Server-2733664" --class_code="TCPWave Remote" --
domain=tcpwave.com --alloc_type=static --room="Room-1" --floor=second --switch=test --port=8808
--duplex=10mbps --org=TCPWave
```

```
twc addobject --object=10.0.0.10 --name="PC-273565" --class_code=PC --domain=tcpwave.com
--alloc_type=Dynamic --opt_tmpl=Generic --mac=01:23:45:67:89:ab --ttl=300 --ns_a=1 --ns_ptr=1 --
ddns_a=1     --ddns_ptr=1     --ddns_cname=1     --ddns_mx=1     --org=TCPWave     --
ext_attr=ext_attr_1/value_1,ext_attr_2/value_2
```

```
twc addobject --object=10.0.0.10 --name="3G121Phone" --class_code="3G Phone" --
domain=tcpwave.com --alloc_type=Reserved --expiry_date="07/1/2018" --mac=01:23:45:67:89:ad --
desc="Reserved IP" --ttl=300 --org=TCPWave
```

```
twc addobject --object=10.0.0.10 --name="AWSInstance0001" --class_code="AWS Instance" --
cloud_instance_tmpl=AWS-Instance-Template --domain=tcpwave.com --alloc_type=static --
mac=01:23:45:67:f9:ac --desc="AWS DNS Instance" --ttl=5000 --ns_a=1 --ns_ptr=1 --ddns_a=1 --
ddns_ptr=1 --ddns_cname=1 --ddns_mx=1 --org=TCPWave
```

```
twc addobject --object=10.0.0.10 --name="Server-2733663" --class_code="TCPWave Remote" --
domain=inter.com --org=Internal --street1="600 ALEXANDER ROAD" --city="PRINCETON" --state=NJ --
country=USA      --zip=08540      --contact_fname=John      --contact_lname=Smith      --
contact_email=john.smith@tcpwave.com
```

addobjecttype**NAME:**

addobjecttype

DESCRIPTION:

Network objects are used to categorize IP addresses into different types of network entities.

Object type is category to define the type of an object or group of objects.
In TCPWave IPAM user can use custom object type or can create user defined object type.

ARGUMENTS:

--code	Unique code for the object type [mandatory]
--logo	Optional logo to display on the GUI for the object type.
--prefix	Prefix of the object type. Example: 3G
--suffix	Suffix of the object type. Example: Phone
--sequence	Initial sequence number for the object type [mandatory]
--prefix_zeros	Set zeros as prefix to initial sequence number. It takes as 'yes' or 'No' [mandatory]
--desc	Description of the object type.

EXAMPLE USAGE:

```
twc addobjecttype --code="3G Phone" --prefix=3G --suffix=Phone --sequence=1 --
prefix_zeros=yes --desc="A 3G Phone"
```

addorg**NAME:**

addorg

DESCRIPTION:

In TCPWave IPAM organization is used to maintain users and networks.
Each organization can be associated with multiple networks.
Root zone of an organization can be enabled or disabled.

ARGUMENTS:

--name
Name of the organization being created. [mandatory]

--desc
Description of the organization being created.

--enable_root_zone
Takes '1' or '0'. Root zone can be created for the organization when the value is '1' and cannot be created for the organization when the value is '0'.

EXAMPLE USAGE:

```
twc addorg --name="TCPWave" --desc="TCPWave organization" --enable_root_zone=1  
addrpztmpl
```

NAME:

addrpztmpl

DESCRIPTION:

Creates a DNS Response policy zone(RPZ) template in the TCPWave IPAM.

ARGUMENTS:

--name
Name of the DNS Response policy zone(RPZ) template to be created in the TCPWave IPAM. [mandatory]

--org
Organization in which Response policy zone(RPZ) template to be created.
This argument is mandatory if user is FADM.

--zone_name
Zone name used in Response policy zone(RPZ) template. [mandatory]

--policy_file
Policy rules file for the RPZ template.

--certificate_file
Path of the certificate file for the Response policy zone(RPZ) data feed from third party.

--url
Response policy zone(RPZ) data feed URL of third party.

--auto_xfr

Flag to indicate whether the zone data feed will be done from an external DNS server or local data. Takes '0' or '1'. Default value is '0'. If this argument is specified, as '1' the zone data will be feed from an external DNS Server using zone transfer.

--master_server

IP address of the server used for RPZ data feed. This argument is mandatory if --auto_xfr is specified, as '1'.

--comm_key_name

Server zone transfer key name for RPZ feed. This argument is mandatory if --auto_xfr is specified, as '1'.

--comm_key_value

Server zone transfer key value for RPZ feed. This argument is mandatory if --auto_xfr is specified, as '1'.

--desc

Description of the RPZ template to be created in the TCPWave IPAM.

EXAMPLE USAGE:

```
twc addrpztmpl --org=TCPWave --name="RPZ-Template" --zone_name=rpzzone.com --  
policy_file=RpzPolicyFile --desc="RPZ template description"
```

```
twc addrpztmpl --org=TCPWave --name="RPZ-Template" --zone_name=rpzzone.com --  
auto_xfr=1 --master_server=10.1.10.26 --comm_key_name="CommunicationKeyName" --  
comm_key_value="CommunicationKeyValue" --desc="RPZ auto feed template"
```

addr

NAME:

addr

DESCRIPTION:

Creates a DNS resource record in 'object', 'zone' or 'network' scopes.

ARGUMENTS:

--rr_scope

Takes 'object', 'zone' or 'revzone'. Defines the context in which the resource record is being added.

--ipv4

IP address of the target object in TCPWave IPAM when defining resource record of type 'A'.

--zone_name

Zone name of the target zone in TCPWave IPAM when rr_scope argument is specified as 'zone'.

--addr

IP Address of the reverse zone in TCPWave IPAM when rr_scope argument is specified as 'revzone'.

--type

Indicates the type of the resource record. Takes one of 'A','CNAME', 'MX','SRV','NS','TXT','NAPTR','PTR','AAAA','DNAME','HINFO','CAA','LOC', 'TLSA' or 'DS'. [mandatory]

--class

Indicates the class of the resource record. Support only 'IN' currently
[mandatory]

--ttl

Indicates the time-to-live value specified in number of seconds for the resource record.

--owner

Owner name of the resource record.

Should be a valid domain name for records of type 'A'.

Should be a valid alias for records of type CNAME

Should be a valid IP Address for records of type PTR

Should be a valid domain name for records of type NS

Should be a valid domain name for record of type DS

[mandatory]

--cname

CNAME data part of a CNAME record.

--domain

Domain name in data part of a PTR resource record.

--host

Host name in data part of a PTR resource record.

--name_server

Name Server or data part a NS resource record.

--org

Organization name to be specified for resource records. [mandatory]

--prefnum

Preference number associated with a MX resource record.

--mail_host

Name of the server hosting the mail service associated with a MX resource record.

--service

Service name associated with a SRV resource record.

--protocol

Protocol associated with a SRV or TLSA resource record.

--priority

Priority number associated with a SRV resource record.

--weight

Weight associated with a SRV resource record.

--port

Port number associated with a SRV or TLSA resource record.

--target

Name of the server hosting the service associated with an SRV record.

Should point to a valid A record for records of type 'SRV'.

--svc_subtype

Service subtype takes the value as '1' or '2'.

--txt

Text associated with a TXT resource record.

--order

Order number associated with a NAPTR resource record.

--flag

Flag value associated with a NAPTR or CAA resource record.

--params

Params value associated with a NAPTR resource record.

--regexp

Regexp value associated with a NAPTR resource record.

--replace

Replace field associated with a NAPTR resource record.

--desc

Description for the resource record.

--external_rr

Takes '0' or '1'. If this argument is specified as '1' resource record being added will be added as an external resource record. This argument is applicable when --rr_scope=zone else it will be ignored.

--is_proxy

DNS Proxy root zone flag. It takes '0' or '1'. If it is specified as '1' resource record being added will be added as a proxy root zone resource record. If it is specified as '0' resource record being added will be added as a root zone resource record. This argument is applicable when --rr_scope=zone and --zone_name=.(dot).

--view

DNS view name in which resource record is being created. This argument is applicable when --rr_scope is zone or object or revzone when type is PTR.

--ipv6

IPv6 address associated with an AAAA resource record.

--redir_name

Redirection name associated with a DNAME resource record.

--hardware

Hardware associated with a HINFO resource record.

--os

OS associated with a HINFO resource record.

--tag

Tag associated with a CAA resource record. It takes one of 'issue', 'issuewild' or 'iodef'

--value

Value associated with a CAA resource record.

--latitude

Latitude value associated with a LOC resource record. Value should be in the form of "<Degree>:<Minutes>:<Seconds>:<N/S>". Example:
"52:22:23.000:N"

--longitude

Longitude value associated with a LOC resource record. Value should be in the form of "<Degree>:<Minutes>:<Seconds>:<E/W>". Example:
"4:53:32.000:E"

--altitude

Altitude value associated with a LOC resource record. Value should be in the form of "<altitude>:<Size>:<Horizontal Precision>:<Vertical Precision>". Example: "-2.00:0.00:10000:10"

--cert_usage

Certificate usage associated with a TLSA resource record.

--selector

Selector associated with a TLSA resource record.

--match_type

Matching type associated with a TLSA resource record.

--cert_data

Certificate association data associated with a TLSA resource record.

--key_tag

Key Tag data associated with a DS resource record. It should be positive integer value. Example: 100.

--algorithm

Algorithm data associated with a DS or SSHFP resource record.

It should be positive value for DS resource record integer value.

Example: 100.

Algorithm is mandatory for SSHFP resource record, it accepts the below algorithm types.

Example: RSA, DSA, ECDSA, Ed25519.

--digest_type

Digest type data associated with a DS resource record. It should be positive integer value. Example: 100.

--key_digest

Key Digest data associated with a DS resource record. It should be hexadecimal key.

--ext_attr

Comma separated list of extension attributes with their values in the format : extension_attribute_name/extension_attribute_value. Use the following command to see all the extension attributes applied to zone :
'twc listext --entity=revzone --d=,'

--mask

Mask length of the network.

--fprint_type

Fingerprint type data associated with SSHFP resource record.
It should accept 'SHA-1' or 'SHA-256' value.

--fprint

Fingerprint data associated with SSHFP resource record.

--public_key

Public Key data associated with a DKIM resource record.The key should not have spaces.

--test_mode

Test mode data associated with a DKIM resource record. It should accepts only 'Only domain' or 'Domain and sub-domains' value.

EXAMPLE:

```
twc addrr --type=A --class=IN --ttl=5000 --owner=www --ipv4=10.0.0.1 --rr_scope=object --org=TCPWave
```

```
twc addrr --type=CNAME --class=IN --ttl=5000 --owner=ftp --ipv4=10.0.0.1 --cname=www --rr_scope=object --org=TCPWave
```

```
twc addrr --type=MX --class=IN --ttl=5000 --owner=www --ipv4=10.0.0.1 --prefnum=10 --mail_host=mail --rr_scope=object --org=TCPWave
```

```
twc addrr --type=SRV --class=IN --ttl=5000 --owner=www --ipv4=10.0.0.1 --service=ldap --protocol=tcp --priority=1 --weight=10 --port=7001 --target=ldapserver.tcpwave.com. --rr_scope=object --org=TCPWave
```

```
twc addrr --type=NAPTR --class=IN --ttl=5000 --owner=www --order=30 --prefnum=100 --flag=U --params="E2U+email" --regexp="^.+\$\\!info@tcpwave.com\\li" --replace=. --rr_scope=object --ipv4=10.0.0.5 --org=TCPWave
```

```
twc addrr --type=TXT --class=IN --ttl=5000 --owner=text --ipv4=10.0.0.5 --txt="spf1 a:mail.tcpwave.com -all" --rr_scope=object --org=TCPWave
```

```
twc addrr --type=PTR --class=IN --ttl=5000 --owner=10.0.0.5 --host=dev --domain=tcpwave.com --rr_scope=revzone --addr=10.0.0.0 --mask=28 --org=TCPWave --desc=description
```

```
twc addrr --type=PTR --class=IN --ttl=1200 --owner=10.0.0.5 --addr=10.0.0.0 --host=dev --domain=tcpwave.com --rr_scope=revzone --mask=28 --org=TCPWave --ext_attr=ext_attr_1/value_1,ext_attr_2/value_2
```

```
twc addrr --type=CNAME --class=IN --ttl=5000 --rr_scope=revzone --zone_name=10.in-addr.arpa --addr=10.10.1.1 --cname=tcpwave123 --org=TCPWave --mask=8 --owner=TCPWave.com.
```

```
twc addrr --type=NS --class=IN --ttl=5000 --rr_scope=revzone --zone_name=10.in-addr.arpa --addr=10.10.1.1 --name_server=tcpwave123. --org=TCPWave --mask=8 --owner=0-9.10.in-addr.arpa.
```

```
twc addrr --type=A --class=IN --ttl=5000 --owner=www --ipv4=10.0.0.1 --rr_scope=zone --zone_name=tcpwave.com --org=TCPWave
```

```
twc addrr --type=CNAME --class=IN --ttl=5000 --owner=ftp --cname=www --zone_name=tcpwave.com --rr_scope=zone --org=TCPWave
```

```
twc addrr --type=MX --class=IN --ttl=5000 --owner=www --prefnum=10 --mail_host=mail --rr_scope=zone --zone_name=tcpwave.com --org=TCPWave
```

```
twc addrr --type=SRV --class=IN --ttl=5000 --owner=www --service=sip --protocol=tcp --priority=1 --weight=10 --port=7001 --target=sipserver.tcpwave.com. --rr_scope=zone --zone_name=tcpwave.com --org=TCPWave
```

```
twc addrr --type=TXT --class=IN --ttl=5000 --owner=text --txt="spf1 a:mail.tcpwave.com -all" --zone_name=tcpwave.com --rr_scope=zone --org=TCPWave
```

```
twc addrr --type=NAPTR --class=IN --ttl=5000 --owner=www --order=30 --prefnum=100 --flag=U --params="E2U+email" --regexp="\!^.+\$\\!info@tcpwave.com\\!i" --replace=. --zone_name=tcpwave.com --rr_scope=zone --org=TCPWave
```

```
twc addrr --type=NS --class=IN --ttl=5000 --owner=ns.external --name_server=ns.tcpwave.com. --zone_name=tcpwave.com --rr_scope=zone --org=TCPWave --external_rr=1
```

```
twc addrr --type=DS --class=IN --ttl=5000 --owner=ns.external --key_tag=10 --algorithm=11 --digest_type=13 --key_digest=23 --zone_name=tcpwave.com --rr_scope=zone --org=TCPWave --external_rr=1
```

```
twc addrr --type=A --class=IN --ttl=5000 --owner=www --ipv4=10.0.0.1 --rr_scope=zone --zone_name=tcpwave.com --org=TCPWave --external_rr=1
```

```
twc addrr --type=NS --class=IN --ttl=5000 --owner=test.tcpwave.com. --name_server=ns.tcpwave.com. --zone_name=. --rr_scope=zone --org=TCPWave
```

```
twc addrr --type=A --class=IN --ttl=5000 --owner=www.tcpwave.com. --ipv4=10.0.0.1 --
rr_scope=zone --zone_name=. --org=TCPWave
```

```
twc addrr --type=A --class=IN --ttl=5000 --owner=www.tcpwave.com. --ipv4=10.1.5.1 --
rr_scope=zone --zone_name=. --is_proxy=1 --org=TCPWave
```

```
twc addrr --type=TLSA --class=IN --ttl=5000 --owner=www.tcpwave.com. --port=9443 --protocol=tcp --
cert_usage=2 --selector=0 --match_type=2 --
cert_data=0ff0ebee2e9be02487662a6caa238f9c329344a9c0e146dccc74b1bbb84a51e6f762fa9e33b
a6d6acd86581184f97c18ca885b753a6bb42f918ff6b6a17801e1 --rr_scope=object --ipv4=172.13.2.13
--org=TCPWave
```

```
twc addrr --type=AAAA --class=IN --ttl=5000 --owner=www.tcpwave.com. --ipv6=5000::1 --
rr_scope=zone --zone_name=www.tcpwave.com --org=TCPWave
```

```
twc addrr --type=DNAME --class=IN --ttl=5000 --redir_name=example.com --rr_scope=zone --
zone_name=www.tcpwave.com --org=TCPWave
```

```
twc addrr --type=HINFO --class=IN --ttl=5000 --owner=www.tcpwave.com. --hardware="PC-Intel-
700mhz" --os="Redhat Linux 7.1" --rr_scope=zone --zone_name=www.tcpwave.com --org=TCPWave
```

```
twc addrr --type=CAA --class=IN --ttl=5000 --flag=0 --tag=issue --value=example.com --
rr_scope=zone --zone_name=www.tcpwave.com --org=TCPWave
```

```
twc addrr --type=LOC --class=IN --ttl=5000 --owner=www.tcpwave.com. --
latitude="52:22:23.000:N" --longitude="4:53:32.000:E" --altitude="-2.00:0.00:10000:10" --
rr_scope=zone --zone_name=www.tcpwave.com --org=TCPWave
```

```
twc addrr --type=TLSA --class=IN --ttl=5000 --port=9443 --protocol=tcp --cert_usage=2 --
selector=0 --match_type=2 --
cert_data=0ff0ebee2e9be02487662a6caa238f9c329344a9c0e146dccc74b1bbb84a51e6f762fa9e33b
a6d6acd86581184f97c18ca885b753a6bb42f918ff6b6a17801e1 --rr_scope=zone --
zone_name=www.tcpwave.com --org=TCPWave
```

```
twc addrr --type=URI --class=IN --ttl=5000 --owner=www --service=sip --protocol=tcp --priority=1  
--weight=10 --target=sipserver.tcpwave.com. --rr_scope=zone --zone_name=tcpwave.com --  
org=TCPWave
```

```
twc addrr --type=SSHFP --class=IN --ttl=5000 --owner=tcpwave.com. --rr_scope=zone --  
zone_name=tcpwave.com --org=TCPWave --fprint_type=SHA-1 --fprint=TCPWave --algorithm=RSA
```

```
twc addrr --type=AFSDB --class=IN --ttl=5000 --owner=tcpwave.com --srvc_subtype=1 --host=arr.  
--rr_scope=zone --zone_name=tcpwave.com --org=TCPWave
```

```
twc addrr --type=DKIM --class=IN --ttl=5000 --owner=tcpwave.com --rr_scope=zone --  
zone_name=tcpwave.com --org=TCPWave --test_mode="Only domain" --public_key=12345
```

```
twc addrr --type=DKIM --class=IN --ttl=5000 --owner=tcpwave.com --rr_scope=zone --  
zone_name=tcpwave.com --org=TCPWave --test_mode="Domain and sub-domains" --  
public_key=12345
```

addscheduledjob

NAME:

addscheduledjob

DESCRIPTION:

Creates a scheduled job in the TCPWave IPAM.

ARGUMENTS:

--job_id

Id of the scheduled job. [mandatory]

--job_type

Type of the scheduled job. It takes one of 'script', 'event-handler' or
'callback-handler'. [mandatory]

--event_handler

Name of the event handler. This argument is applicable if the job_type
is 'event-handler'.

--callback_handler

Name of the callback handler. This argument is applicable if the
job_type is 'callback-handler'.

--file_name

Full path of the script file. This argument is applicable if the
job_type is 'script'.

--args

Space separated list of required argument if any.

--repeat_type

Repeat type of the scheduled job. It takes one of 'daily', 'weekly', 'monthly', 'repetitive' or 'none'. [mandatory]

--exe_date

Execution date and time on which the scheduled job is to be executed. This argument is applicable if the repeat_type is 'none'. Date and time format is "yyyy-MM-dd hh:mm:ss".

--start_date

Start date and time from which the scheduled job is to be applied. This argument is applicable if the repeat_type is one of 'daily', 'weekly' or 'monthly'. Date and time format is "yyyy-MM-dd hh:mm:ss".

--end_date

End date and time till which the scheduled job is to be applied. This argument is applicable if the repeat_type is one of 'daily', 'weekly' or 'monthly'. Date and time format is "yyyy-MM-dd hh:mm:ss".

--exe_at

Execution time of the scheduled job. This argument is applicable if the repeat_type is one of 'daily', 'weekly' or 'monthly'. Time format is "hh:mm:ss".

--day_of_week

Day of the month if repeat_type is weekly. Takes number from 1 to 7.

--day_of_month

Day of the month if repeat_type is monthly. Takes number from 1 to 31.

--repeat_interval

Repeat interval of the scheduled job in minutes. This argument is applicable when repeat_type is 'repetitive'.

--repeat_count

Repeat count of the scheduled job. This argument is applicable when repeat_type is 'repetitive'.

--desc

Description of the scheduled job.

EXAMPLE USAGE:

```
twc addscheduledjob --job_id=ScheduledJobId --job_type=event-handler --  
event_handler=RemoteCheckoutOperation --repeat_type=daily --start_date="2017-10-25  
12:00:00" --end_date="2018-10-25 00:00:00" --exe_at=12:00:00 --desc="The IPAM Checkout  
operation"
```

```
twc addscheduledjob --job_id=ScheduledJobId --job_type=event-handler --
```

```
event_handler=RemoteCheckoutOperation --repeat_type=weekly --start_date="2017-10-25 12:00:00" --end_date="2019-10-25 12:00:00" --exe_at=12:00:00 --day_of_week=1 --desc="The IPAM Checkout operation"
```

```
tvc addscheduledjob --job_id=ScheduledJobId --job_type=callback-handler -- callback_handler=RemoteCheckoutOperation --repeat_type=monthly --start_date="2017-10-25 12:00:00" --end_date="2019-10-25 12:00:00" --exe_at=12:00:00 --day_of_month=1 --desc="The IPAM Checkout operation"
```

```
tvc addscheduledjob --job_id=ScriptSchedjob --job_type=script --file_name=Script_File_Name -- repeat_type=repetitive --start_date="2017-12-25 12:00:00" --end_date="2018-12-25 00:00:00" -- repeat_interval=10 --repeat_count=5 --desc="Script type Repetitive scheduled Job"
```

addscope

NAME:

addscope

DESCRIPTION:

Add a DHCP scope specified, by --start_ip and --end_ip arguments in the TCPWave IPAM.

ARGUMENTS:

--start_ip

Start IP address of the DHCP scope. [mandatory]

--end_ip

End IP address of the DHCP scope. [mandatory]

--obj_type

Valid class code in TCPWave IPAM to be associated with the DHCP scope. [mandatory]

--ttl

Time-to-Live in seconds for the target scopes.

--ns_a

Takes 0 | 1. 1 indicates that the corresponding A resource record must be updated in the name service zone file. 0 indicates that the corresponding A resource record will not be updated in the name service zone file. A default value of 1 will be used if not specified.,

--ns_ptr

Takes 0 | 1. 1 indicates that the corresponding PTR resource record must be updated in the name service zone file. 0 indicates that the corresponding PTR resource record will not be updated in the name service zone file. A default value of 1 will be used if not specified.,

--ddns_a

Takes 0 | 1. 1 indicates that dynamic DNS updates are allowed to the corresponding A resource record in the name service zone file.

0 indicates that dynamic DNS updates are not allowed to the corresponding A resource record in the name service zone file.
A default value of 1 will be used if not specified.,

--ddns_ptr

Takes 0 | 1. 1 indicates that dynamic DNS updates are allowed to the corresponding PTR resource record in the name service zone file.
0 indicates that dynamic DNS updates are not allowed to the corresponding PTR resource record in the name service zone file.
A default value of 1 will be used if not specified.,

--ddns cname

Takes 0 | 1. 1 indicates that dynamic DNS updates are allowed to the corresponding CNAME resource record in the name service zone file.
0 indicates that dynamic DNS updates are not allowed to the corresponding CNAME resource record in the name service zone file.
A default value of 1 will be used if not specified.,

--ddns_mx

Takes 0 | 1. 1 indicates that dynamic DNS updates are allowed to the corresponding MX resource record in the name service zone file.
0 indicates that dynamic DNS updates are not allowed to the corresponding MX resource record in the name service zone file.
A default value of 1 will be used if not specified.,

--allow_client

DHCP client class for which leases can be granted from this scope.

--allow_user

DHCP user class for which leases can be granted from this scope.

--allow_vendor

DHCP vendor class for which leases can be granted from this scope.

--deny_client

DHCP client class for which leases are not granted from this scope.

--deny_user

DHCP user class for which leases are not granted from this scope.

--deny_vendor

DHCP vendor class for which leases are not granted from this scope.

--desc

Description for the scope.

--org

Name of the organization to which the objects belongs. This argument is mandatory if user is 'FADM'.

EXAMPLE USAGE:

```
twc addscope --start_ip=9.12.0.13 --end_ip=9.12.0.17 --obj_type="3G Phone" --ttl=1200 --ns_a=1 --ns_ptr=1 --ddns_a=1 --ddns_ptr=1 --ddns_cname=1 --ddns_mx=1 --allow_client=clientclassX --allow_user=userclassX --allow_vendor=vendorclassX --deny_client=clientclassY --deny_user=userclassY --deny_vendor=vendorclassY --org=TCPWave
```

addsubnet**NAME****addsubnet****DESCRIPTION**

Creates a subnet in the TCPWave the IPAM.

ARGUMENTS**--name**

Name of the subnet being created.

--subnet

IP Address of the subnet. [mandatory]

--mask

Mask length of the subnet. [mandatory]

--type

Type of the subnet. Takes 'Non-DHCP', 'DHCP-Enabled' or 'Cloud-Hosted'. [mandatory]

--network

Start Address of the associated network. [mandatory]

--org

Organization name under which the subnet is being created. [mandatory]

--subnet_group

Name of the associated subnet group.

--domain

Domain to be associated with this subnet. [mandatory]

--enable_discovery

Enable discovery option for the subnet as 'yes' or 'no'.

--discovery_tmpl

Discovery template name. Accepted only when enable discovery option is set to 'yes'.

--enable_reclaim

If enabled, reclaim the eligible objects in the subnet based on the discovery result. Enable discovery option and discovery template name is mandatory when it is set to 'yes'.

--desc

Description text for the subnet.

--street1
Street1 part of the location information.

--street2
Street2 part of the location information.

--city
City part of the location information.

--state
State part of the location information.

--zip
Zip code part of the location information.

--secondary_domains
Name of the secondary domains to be associated with subnet. It accepts up to 50 secondary domains by separating with comma. Example: "tcp.com,tcpwave.com,tcpzone.com"

--option
Takes 'single', 'multi' or 'all'. When 'multi' is specified, the list of IP Addresses of the subnets to be created should be specified as a comma separated values using --subnet argument When 'all' is specified, there is no need to specify an IP Address. All available subnets for the given mask length shall be created.

--router_addr
IP Address of the router associated with the subnet being created in case of a single subnet creation.

--router_opt
Takes 'first' or 'last' or 'none' In case of multi subnet creation, 'first' indicates that the router IP Address will be the first address of the subnet address range. 'last' indicates that the router IP Address will be the last address of the subnet address range. 'none' indicates that not to specify the router when the subnet type is 'Non-DHCP'.

--dhcp_tmpl
Template name specifying the DHCP options for the subnet. This argument is mandatory if dhcp_server argument is specified.

--dhcp_appliance
Primary DHCP appliance for the subnet. This argument is mandatory if dhcp_tmpl argument is specified.

--dhcp_failover_peer
Name of the DHCP failover peer.

--domain_server

IP address of the DNS appliances. It accepts the comma separated DNS appliances, this argument is applicable only when type is specified as 'DHCP-Enabled'.

--shared_network

Name of the shared network. This argument is only applicable when type is specified as 'DHCP-Enabled'.

--views

Comma separated list of DNS view names to be associated with this subnet. Specified DNS views must be available for the primary domain.

--vlan

VLAN to be associated with this subnet.

--vrf_name

Name of the VRF to be associated with this subnet.

--cloud_provider

Name of the cloud provider to be associated with this subnet.

--ext_attr

Comma separated list of extension attributes with their values in the format : extension_attribute_name/extension_attribute_value. Use the following command to see all the extension attributes applied to subnet : 'twc listext --entity=subnet --d=,'

--subnet_tmpl

Subnet template name used to create subnet directly. Subnet template contains all the information to creates a subnet and defines offset to create objects and router in that subnet.

--contact_fname

First name field of the associated contact information for the subnet.

--contact_mname

Middle name field of the associated contact information for the subnet.

--contact_lname

Last name field of the associated contact information for the subnet.

--contact_email

Email ID field of the associated contact information for the subnet.

EXAMPLE

```
twc addsubnet --subnet=10.0.0.0 --mask=24 --network=10.0.0.0 --name="IT-Subnet" --  
type=DHCP-Enabled --subnet_group=IT_SG --street1="600 ALEXANDER ROAD" --city="PRINCETON" --  
state=NJ --country=USA --zip=08540 --domain=tcpwave.com --dhcp_tmpl=Generic --
```

```
dhcp_appliance=nusalx-trv10-sl0984 --option=single --router_addr=10.0.0.1 --org=TCPWave --
views=view1,view2 --desc="TCPWave IT Subnet"

tgc addsubnet --subnet=192.168.0.80 --mask=28 --network=192.168.0.0 --
domain_server=192.168.0.8,12.168.0.20 --shared_network=tcpsharednetwork --name="IT-Subnet" --
-type=DHCP-Enabled --domain=inter.com --dhcp_tmpl=Generic --
dhcp_appliance=TCPWave01Remote --option=single --router_addr=192.168.0.88 --org=Internal --
desc="TCPWave IT Subnet"

tgc addsubnet --subnet=10.0.0.0,10.128.0.0 --mask=9 --network=10.0.0.0 --type=DHCP-Enabled
--subnet_group=IT_SG --street1="600 ALEXANDER ROAD" --city="PRINCETON" --state=NJ --
country=USA --zip=08540 --domain=tcpwave.com --dhcp_tmpl=Generic --dhcp_appliance=nusalx-
trv10-sl0984 --option=multi --router_opt=last --org=TCPWave --desc="TCPWave IT Subnet"

tgc addsubnet --subnet=10.0.0.0 --mask=16 --network=10.0.0.0 --type=Non-DHCP --
subnet_group=IT_SG --street1="600 ALEXANDER ROAD" --city="PRINCETON" --state=NJ --
country=USA --zip=08540 --domain=tcpwave.com --option=all --router_opt=none --org=TCPWave --
desc="TCPWave IT Subnet"

tgc addsubnet --subnet=10.0.0.0 --mask=24 --network=10.0.0.0 --name="IT-Subnet" --
type=Cloud-Hosted --cloud_provider=AWS_Provider --subnet_group=IT_SG --street1="600
ALEXANDER ROAD" --city="PRINCETON" --state=NJ --country=USA --zip=08540 --
domain=tcpwave.com --option=single --router_addr=10.0.0.1 --org=TCPWave --views=view1,view2 --
desc="TCPWave IT Subnet" --ext_attr=ext_attr_1/value_1,ext_attr_2/value_2

tgc addsubnet --subnet=10.0.0.0 --mask=24 --network=10.0.0.0 --option=single --org=Internal --
desc="TCPWave IT Subnet" --subnet_tmpl="sub-1" --enable_discovery=no

tgc addsubnet --subnet=10.1.0.0 --mask=24 --network=10.1.0.0 --option=single --org=Internal --
desc="TCPWave IT Subnet" --subnet_tmpl="sub-1" --enable_discovery=yes --
discovery_tmpl="Discovery Template1" --enable_reclaim=yes

tgc addsubnet --subnet=10.1.0.0 --mask=24 --network=10.1.0.0 --option=single --
router_addr=10.1.0.1 --type=DHCP-Enabled --domain=tcpwave.com --dhcp_tmpl=Generic --
dhcp_appliance=nusalx-trv10-sl0984 --dhcp_failover_peer=dhcp-failover-peer-1 --org=TCPWave --
desc="TCPWave IT Subnet"

tgc addsubnet --subnet=10.1.0.0 --mask=24 --network=10.1.0.0 --option=single --
router_addr=10.1.0.1 --type=DHCP-Enabled --domain=tcpwave.com --dhcp_tmpl=Generic --
dhcp_appliance=nusalx-trv10-sl0984 --dhcp_failover_peer=dhcp-failover-peer-1 --
contact_fname=John --contact_lname=Smith --contact_email=john.smith@tcpwave.com --
org=TCPWave --desc="TCPWave IT Subnet"
```

```
twc addsubnet --subnet=10.1.0.0 --mask=24 --network=10.1.0.0 --type=Non-DHCP --
domain=tcpwave.com --option=single --router_addr=10.1.0.1 --org=TCPWave
```

```
twc addsubnet --subnet=10.1.0.0 --mask=24 --network=10.1.0.0 --type=Non-DHCP --
domain=tcpwave.com --option=single --router_addr=10.1.0.1 --org=TCPWave --
secondary_domains="tcpwave.com,tcpwave1.com"
```

addsubnetgroup

NAME:

addsubnetgroup

DESCRIPTION:

Creates a subnet group in the TCPWave IPAM.

ARGUMENTS:

--name	Name of the subnetgroup being created [mandatory]
--org	Organization name of the subnetgroup [mandatory]
--desc	Description of the subnetgroup.

EXAMPLE USAGE:

```
twc addsubnetgroup --name=IT_SG --desc="IT subnet group" --org=TCPWave
```

addmultiarr

DESCRIPTION:

Adds A records into the zone in the given organization in the TCPWave IPAM.

ARGUMENTS:

--zone	Name of the zone. [mandatory]
--ip	Comma separated IP address. [mandatory]
--org	Name of the organization. [mandatory]
--owner	Name of the owner. [mandatory]
--proxy	proxy of the A resource record take input as '0' or '1'.

EXAMPLE:

```
twc      addmultiarr    --zone=tcpwave.com      --org=TCPWave      --owner=newzone      --
ip=10.1.10.22,10.1.10.23 --proxy=0
```

addawsimage**NAME:**

addawsimage

DESCRIPTION:

Creates an AWS Machine image in the TCPWave IPAM.

ARGUMENTS:

--provider_name	Name	of	the	cloud	provider.	[mandatory]
--image_id	ID	of	the	AWS	Machine	Image.
--image_name	Name	of	the	AWS	Machine	Image.
--desc	Description of the AWS Machine Image.					

EXAMPLE USAGE:

```
twc addawsimage --provider_name="AWS" --image_name="AWS Image" --image_id="ami-a4c7edb2-test" --desc="AWS Machine Image-1"
```

adddhcpsharednetwork**NAME:**

Adddhcpsharednetwork

DESCRIPTION:

Creates a DHCP shared network in the TCPWave IPAM.

ARGUMENT:

--name	Name of the shared network. [mandatory]
--ip	IP address of the DHCP primary appliance. [mandatory]
--org	Name of the organization. [mandatory]
--desc	Description of the shared network.

EXAMPLE USAGE:

```
twc      adddhcpsharednetwork      --name=TestNet      --ip=10.0.0.213      --org=TCPWave      --
desc="Tcpwave Shared Network"
```

applybinlog**NAME:**

applybinlog

DESCRIPTION:

Loads database incremental changes into recovery database. If value for --dump_dir argument is not specified, the changes will be applied from the most recent dump directory.

ARGUMENTS:**--dump_dir**

Dump directory from which incremental changes are to be applied.

EXAMPLE USAGE:

```
twc applybinlog --dump_dir=Dump_1408948935
```

addmultiarr**ARGUMENTS:****--zone**

Name of the zone. [mandatory]

--ip

Comma separated IP address. [mandatory]

--org

Name of the organization. [mandatory]

--owner

Name of the owner. [mandatory]

--proxy

proxy of the A resource record take input as '0' or '1'.

EXAMPLE:

```
twc      addmultiarr      --zone=tcpwave.com      --org=TCPWave      --owner=newzone      --
ip=10.1.10.22,10.1.10.23 --proxy=0
```

addvrf**NAME****addvrf****DESCRIPTION**

Creates a VRF in the TCPWave IPAM.

ARGUMENTS:**--name**

Name of VRF. [mandatory]

--org

Name of the organization. [mandatory]

--router_distinguisher

Enter the AS number or IP address of the route distinguisher of the discovered VRF.

--interface

VRF can be assigned to any interface loopback or VLAN. Example: f0/0.82

--import_target

Imports routing information from the target extended community.

--export_target

Exports routing information to the target extended community.

--desc

Description of the VRF.

EXAMPLE:

```
twc addvrf --name=testVrf --org=TCPWave --router_distinguisher=100:30 --interface=1/1 --  
import_target=10.1.1.10 --export_target=10.1.1.20 --desc="Test VRF"
```

Deletions

deleteactivelease

NAME:

deleteactivelease

DESCRIPTION:

Deletes the active lease from a DHCP server defined in the TCPWave IPAM.

ARGUMENTS:

--dhcp_appliance

IP address of the DHCP server. [mandatory]

-ip

IP Address of the active lease to be deleted. [mandatory]

EXAMPLE USAGE:

```
twc deleteactivelease --dhcp_appliance=10.1.10.180 --ip=10.1.10.103
```

deleteadminrole**NAME:**

```
deleteadminrole
```

DESCRIPTION:

Deletes an administrator role from the TCPWave IPAM. An administrator role can be deleted by specifying name of the administrator role.

ARGUMENTS:

```
--name
```

Name of the administrator role. [mandatory]

EXAMPLE:

```
twc deleteadminrole --name=CADM
```

deleteadmin**NAME:**

```
deleteadmin
```

DESCRIPTION:

Deletes an administrator from the TCPWave IPAM. The user is identified uniquely by the login name.

ARGUMENTS:

```
--login_name
```

Login name of the administrator. [mandatory]

EXAMPLE USAGE:

```
twc deleteadmin --login_name=john
```

deleteadminpermission**NAME**

```
deleteadminpermission
```

DESCRIPTION

Deletes an administrator/administrator group permission from the TCPWave IPAM.

ARGUMENTS

```
--level
```

It takes the input as admin or admin group, if the level is admin the input param of admin is mandatory otherwise admin group is mandatory. [mandatory]

--admin

Name of the admin.

--admin_group

Name of the admin group.

--org

Name of the organization. [mandatory]

--role

Name of the administrator role. [mandatory]

--privilege

Name of the privilege, It takes the input as 'Read' or 'Write' or 'Deny'. [mandatory]

--function

Name of the administrator function. [mandatory]

--function_value

Value of the administrator function. [mandatory]

--select_all

It takes the input as '0' or '1'. [mandatory]

EXAMPLE:

```
twc deleteadminpermission --level=Admin --admin=Test --function="IPv4 Networks" --  
privilege=Write --function_value=10.1.10.0 --org=TcpWave --role=EADM --select_all=0
```

```
twc deleteadminpermission --level="Admin Group" --admin_group="Test Group" --  
function="TCPWave DHCP IPv4 Appliances" --privilege=Read --  
function_value=TCPWave00001Remote --org=TcpWave --role=CADM --select_all=1
```

deleteadmingroup

NAME:

deleteadmingroup

DESCRIPTION:

Deletes an administrator group from the TCPWave IPAM.

An administrator group can be deleted by specifying name of the administrator group.

ARGUMENTS:

--name

Name of the administrator group. [mandatory]

--org

Organization name associated with the administrator group. This argument is for users in FADM role to select appropriate organization to which the operation must be applied. For users not in FADM role, the operation is by default applied to the organization that the user is associated with.

EXAMPLE USAGE:

twc deleteadmingroup --org=TCPWave --name="default-admin-group"

twc deleteadmingroup --name="default-admin-group"

deleteappliancegroup

NAME:

deleteappliancegroup

DESCRIPTION:

Deletes an appliance group from the TCPWave IPAM.

ARGUMENTS:

--name

Name of the appliance group being deleted. [mandatory]

--org

Name of the organization where the operation must be performed. This argument is mandatory if the user is FADM. [mandatory]

EXAMPLE USAGE:

twc deleteappliancegroup --name=IT_SG --org=TCPWave

deleteawsimage

NAME:

deleteawsimage

DESCRIPTION:

Deletes an AWS machine Image from the TCPWave IPAM.

ARGUMENTS:

--aws_image_id

ID of the AWS machine image. [mandatory]

EXAMPLE USAGE:

twc deleteawsimage --aws_image_id="ami-a4c7edb2"

deleteasnumber**NAME**

Deleteasnumber

DESCRIPTION

Deletes a Autonomous System Number from the TCPWave IPAM.

ARGUMENTS

--org

Organization name from which the Autonomous System Number has to be deleted. [mandatory]

--name

Name of the Autonomous System Number. [mandatory]

EXAMPLE

```
twc deleteasnumber --name="TCPWave-ASN" --org=TCPWave
```

```
twc deleteasnumber --name="ASN" --org=TCPWave
```

deletecontact**NAME:**

deletecontact

DESCRIPTION:

Deletes a contact from the TCPWave IPAM. Target organization must be specified, by name using --org parameter if user is the FADM. Target contact must be specified, in terms of the mandatory contact information fields.

ARGUMENTS:

--org

Organization name for which the contact is being deleted. This argument is mandatory if the user is FADM.

--first_name

First name field of the contact information [mandatory]

--middle_name

Middle name field of the contact information if any.

```
--last_name
    Last name field of the contact information. [mandatory]

--email
    Email id field of the contact information. [mandatory]

--error_file
    Path of the file to write the contact references, on the target IPAM
    server. If the file path is not specified, the error output is written
    to the standard output.
```

EXAMPLE USAGE:

```
twc deletecontact --first_name=John --last_name=Smith --email=john.smith@tcpwave.com --
org=TCPWave
```

```
twc deletecontact --first_name=John --last_name=Smith --email=john.smith@tcpwave.com --
org=TCPWave --error_file=/tmp/referenced_rerords.txt
```

```
twc deletecontact --first_name=James --middle_name=Francis --last_name=Stuart --
email=james.stuart@tcpwave.com --org=TCPWave
```

deletecustomfolder**NAME:**

```
deletecustomfolder
```

DESCRIPTION:

Deletes a DHCP options custom folder from the TCPWave IPAM.
Note:- A custom folder cannot be deleted if it has custom DHCP option defined.

ARGUMENTS:

```
--name
    Name of the DHCP custom folder. [mandatory]
```

EXAMPLE USAGE:

```
twc deletecustomfolder --name=voip-options
```

deletedhcpfailoverpeer**NAME:**

```
deletedhcpfailoverpeer
```

DESCRIPTION:

Deletes a DHCP failover peer from the TCPWave IPAM.

ARGUMENTS:

```
--name
    Name of the DHCP failover peer. [mandatory]
```

EXAMPLE USAGE:

```
twc deletedhcpfailoverpeer --name=dhcp-failover-peer-1
```

deletedhcpfingerprint**NAME:**

deletedhcpfingerprint

DESCRIPTION:

Deletes DHCP finger print data in the TCPWave IPAM
A DHCP finger print can be deleted by specifying MAC bits and option sequence of the DHCP finger print.

ARGUMENTS:

--mac_vendor
Name of the MAC vendor.

--mac_bits
First six bits of the MAC address. [mandatory]

--option_sequence
DHCP option sequence. [mandatory]

EXAMPLE USAGE:

```
twc deletedhcpfingerprint --mac_vendor=DELL --mac_bits=3690e6 --option_sequence=1,2,6,5,4,7
```

```
twc deletedhcpfingerprint --mac_bits=3695e6 --option_sequence=1,2,6,7,9
```

deletedhcpoption**NAME:**

deletedhcpoption

DESCRIPTION:

Deletes a user defined DHCP option from the TCPWave IPAM.
option code used by this option will be unused after successfully execution of this operation.

ARGUMENTS:

--name
Name of the DHCP custom option [mandatory]

EXAMPLE USAGE:

```
twc deletedhcpoption --name=ip-map
```

deletedhcpoptionspace**NAME:**

deletedhcpoptionspace

DESCRIPTION:

Deletes a DHCP option space from the TCPWave IPAM.
Specify the DHCP option space name to delete the option space.

ARGUMENTS:**--name**

Name of the DHCP option space. [mandatory]

EXAMPLE USAGE:

twc deletedhcpoptionspace --name=space

deletedhcpclass**NAME:**

deletedhcpclass

DESCRIPTION:

Deletes a DHCP Client Class or User Class or Vendor Class from the TCPWave IPAM.

ARGUMENTS:**--name**

Name of the DHCP class [mandatory]

--type

Type of the DHCP class. Takes 'user','vendor' or 'client' [mandatory]

EXAMPLE USAGE:

twc deletedhcpclass --name=vendor-class --type=vendor

twc deletedhcpclass --name=user-class --type=user

twc deletedhcpclass --name=client-class --type=client

deletedhcpserver**NAME:**

deletedhcpserver

DESCRIPTION:

Deletes a DHCP Server from the TCPWave IPAM.

ARGUMENTS:

--ip

IP Address of the DHCP Server. [mandatory]

EXAMPLE USAGE:

twc deletedhcpserver --ip=192.168.0.238

deletedhcpopttmpl

NAME

deletedhcpopttmpl

DESCRIPTION

Deletes a DHCP option template from the TCPWave IPAM.

ARGUMENTS

--name

Name of the DHCP option template to be deleted [mandatory]

--org

Organization name associated with the template. This argument is mandatory for the users in FADM role to select appropriate organization to which the operation must be applied. For users not in FADM role, the operation is applied to the organization that the user is associated with.

EXAMPLE

twc deletedhcpopttmpl --name="DHCP-Option-Template" --org=TCPWave

deletedhcppolicytmpl**NAME**

deletedhcppolicytmpl

DESCRIPTION

Deletes a DHCP policy template from the TCPWave IPAM.

ARGUMENTS

--name

Name of the DHCP policy template to be deleted [mandatory]

--org

Organization name associated with the template. The operation is applied to the organization that the user is associated with. [mandatory]

EXAMPLE

```
twc deletedhcppolicytmpl --name="DHCP-Oplicy-Template" --org=TCPWave
```

deletediscoverytmpl**NAME:**

deletediscoverytmpl

DESCRIPTION:

Deletes a discovery template from the TCPWave IPAM.

ARGUMENTS:

--discovery_tmpl

Name of the discovery template. [mandatory]

--org

Name of the organization. [mandatory]

EXAMPLE USAGE:

```
twc deletediscoverytmpl --discovery_tmpl=TestTmpl --org=TCPWave
```

deletednsacl**NAME:**

deletednsacl

DESCRIPTION:

Deletes a DNS ACL from the TCPWave IPAM.

ARGUMENTS:

--name

Name of the DNS ACL to be deleted [mandatory]

EXAMPLE USAGE:

```
twc deletednsacl --name=Internal-ACL
```

deletednsforwarders

NAME:

deletednsforwarders

DESCRIPTION:

Deletes a DNS forwarder from the TCPWave IPAM which is used to resolve a DNS zone that is not managed by the TCPWave. Forwarders exist on an internal 'BIND CACHE' or 'UNBOUND' DNS server in the TCPWave IPAM.

ARGUMENTS:

--appliance_ip

IP Address of the DNS internal cache server [mandatory]

--appliance_type

Type of the DNS server. Takes 'BIND CACHE' or 'UNBOUND' [mandatory]

--zone

Name of the DNS forward Zone [mandatory]

EXAMPLE USAGE:

```
twc deletednsforwarders --appliance_ip=10.1.10.29 --appliance_type="BIND CACHE" --zone="tcpwave.com"
```

deletednsforwarderstmpl**NAME**

deletednsforwarderstmpl

DESCRIPTION

Deletes a DNS forwarder template with forwarder zones which are not managed by TCPWave IPAM.

ARGUMENTS

--group_name

Name of the DNS forwarders group. [mandatory]

EXAMPLE

```
twc deletednsforwarderstmpl --group_name=Test
```

deletednsopttmpl**NAME**

deletednsopttmpl

DESCRIPTION

Deletes a DNS Option Template from the TCPWave IPAM.

ARGUMENTS:

--name

Name of the DNS option template to be deleted [mandatory]

--org

Organization name associated with the template. This argument is mandatory for the users in FADM role to select appropriate organization to which the operation must be applied. For users not in FADM role, the operation is applied to the organization that the user is associated with.

EXAMPLE USAGE:

```
twc deletednsopttmpl --name="Bind-Auth-Template" --org=TCPWave
```

```
twc deletednsopttmpl --name="Bind-Auth-Template"
```

deletednsreversezone**NAME:**

deletednsreversezone

DESCRIPTION:

Deletes a DNS reverse zone from the TCPWave IPAM.

A DNS reverse zone can be deleted by specifying IP address, mask length and the organization name of the DNS reverse zone.

ARGUMENTS:**--ip**

IP address of the DNS reverse zone to be deleted [mandatory]

--mask

Mask length of the reverse zone to be deleted [mandatory]

--org

Organization name associated with the reverse zone [mandatory]

EXAMPLE USAGE:

```
twc deletednsreversezone --ip="1.0.0.0" --mask=24 --org=TCPWave
```

deletednsrootzone**NAME:**

deletednsrootzone

DESCRIPTION:

Deletes a DNS root zone from the TCPWave IPAM.

ARGUMENTS:**--org**

Organization name associated with the DNS root zone. [mandatory]

--is_proxy

DNS proxy root zone flag. It takes '0' or '1'. If it is specified, as '1' proxy root zone is deleted. If it is not specified, or specified, as '0'

root zone is deleted.

EXAMPLE USAGE:

```
twc deletednsrootzone --org=TCPWave --is_proxy=0
```

```
twc deletednsrootzone --org=TCPWave --is_proxy=1
```

deletednsserver**NAME:**

deletednsserver

DESCRIPTION:

Deletes a DNS Server from the TCPWave IPAM.

ARGUMENTS:

--ip

IP address of the DNS Server to be deleted [mandatory]

EXAMPLE USAGE:

```
twc deletednsserver --ip=10.1.10.181
```

deletednsservertmpl**NAME:**

deletednsservertmpl

DESCRIPTION:

Deletes a DNS Server Template from the TCPWave IPAM.

ARGUMENTS:

--name

Name of the DNS Server Template to be deleted [mandatory]

EXAMPLE USAGE:

```
twc deletednsservertmpl --name="DNS Appliance Template"
```

deletemicrosoftadserver**NAME**

deletemicrosoftadserver

DESCRIPTION

Deletes a Microsoft AD appliance from the TCPWave IPAM.

ARGUMENTS

--ip

IP address of the appliance. [mandatory]

EXAMPLE

```
twc deletemicrosoftadserver --ip=10.0.0.10
```

deletemicrosoftadsite**NAME**

deletemicrosoftadsite

DESCRIPTION

Deletes Microsoft AD site from the TCPWave IPAM.

ARGUMENTS

--name

Name of the Microsoft AD site. [mandatory]

--ip

IP address of the appliance. [mandatory]

EXAMPLE

twc deletemicrosoftadsite --name=NJSite --ip=10.0.0.10

deletednsview**NAME:**

deletednsview

DESCRIPTION:

Deletes a DNS view from the TCPWave IPAM.

ARGUMENTS:

--org

Name of the organization DNS views associated with. This argument is mandatory if the user is FADM.

--name

Name of the DNS view to be deleted. [mandatory]

EXAMPLE USAGE:

twc deletednsview --name=view1

twc deletednsview --org=TCPWave --name=view1

deletednszone**NAME:**

deletednszone

DESCRIPTION:

Deletes a DNS Zone from the TCPWave IPAM.

ARGUMENTS:

--name

Name of the DNS Zone to be deleted [mandatory]

--org

Organization name associated with the zone. If this argument is not specified, organization associated with the user will be used.

EXAMPLE USAGE:

```
twc deletednszone --name="tcpwave.com" --org=TCPWave
```

```
twc deletednszone --name="tcpwave.com"
```

deletednszonetmpl

NAME:

deletednszonetmpl

DESCRIPTION:

Deletes a DNS Zone Template from the TCPWave IPAM.

ARGUMENTS:

--name

Name of the DNS zone template to be deleted [mandatory]

--org

Organization name associated with the template. This argument is mandatory for the users in FADM role to select appropriate organization to which the operation must be applied. For users not in FADM role, the operation is applied to the organization that the user is associated with.

EXAMPLE USAGE:

```
twc deletednszonetmpl --name="base-zone-template" --org=TCPWave
```

```
twc deletednszonetmpl --name="base-zone-template"
```

deletedomain

NAME:

deletedomain

DESCRIPTION:

Deletes a DNS Domain from the TCPWave IPAM.

A DNS domain can be deleted by specifying name of the DNS domain.

Note:- Before deleting a DNS domain it is mandatory to delete all the associated IPAM entities of the target DNS domain.

ARGUMENTS:

--name

Name of the target domain being deleted. [mandatory]

--org

Organization name associated with the domain.

This argument is for users in FADM role to select appropriate

organization to which the operation must be applied.
For users not in FADM role, the operation is applied to the organization
that the user is associated with.

EXAMPLE USAGE:

```
twc deletedomain --name="tcpwave.com" --org=TCPWave
```

```
twc deletedomain --name="tcpwave.com"
```

deletedump**NAME:**

```
deletedump
```

DESCRIPTION:

Deletes a database dump from the TCPWave IPAM.

ARGUMENTS:

```
--dump_dir [mandatory]
```

Path of the dump directory to be deleted. [mandatory]

EXAMPLE USAGE:

```
twc deletedump --dump_dir=/tmp/dumps/Dump_1408948935
```

deleteext**NAME:**

```
deleteext
```

DESCRIPTION:

Deletes an extended attribute from the TCPWave IPAM.

ARGUMENTS:

```
--name
```

Name of the extension attribute. [mandatory]

EXAMPLE USAGE:

```
twc deleteext --name=RITS_ID
```

deleteextvalue**NAME**

```
deleteextvalue
```

DESCRIPTION

Deletes the extended attribute value of the specified key defined in the TCPWave IPAM.
Applicable entities are 'admin', 'network', 'subnet', 'object' and 'zone'. If entity is 'network',
'subnet' or 'object' key must be IP address. If entity is admin key is the admin name. If entity is
zone, then key is domain name.

ARGUMENTS

--name
 Name of the extended attribute. [mandatory]
--entity
 Entity of the extended attribute. [mandatory]
--key
 Key of the extended attribute. [mandatory]
--org
 Name of the organization. [mandatory]

EXAMPLE

twc deleteextvalue --name="CHG_TKT" --entity=zone --key=tcpwave.com --or g=TCPWave
deletefirewalltmpl

NAME:

deletefirewalltmpl

DESCRIPTION:

Deletes the specified, firewall template from the TCPWave IPAM.

ARGUMENTS:

--name
 Name of the firewall template. [mandatory]

EXAMPLE USAGE:

twc deletefirewalltmpl --name="DNS-Firewall-Template"
deleteipamfromha

NAME:

deleteipamfromha

DESCRIPTION:

Removes a TCPWave IPAM from high availability cluster.

ARGUMENTS

--ip
 IP address of the IPAM. [mandatory]

EXAMPLE USAGE

twc deleteipamfromha --ip=172.16.0.172
deleteipamappliance

NAME

deleteipamappliance

DESCRIPTION

Deletes an IPAM appliance from the TCPWave IPAM. An IPAM appliance can be deleted by specifying IP address of the appliance.

ARGUMENTS

--ip

IPAM appliance IP address. [mandatory]

EXAMPLE

```
twc deleteipamappliance --ip=10.1.10.243
```

deleteipv6dnsserver**NAME**

deleteipv6dnsserver

DESCRIPTION

Deletes an IPv6 DNS appliance from the TCPWave IPAM.

ARGUMENTS

--ip

IPv6 address of the DNS appliance to be deleted [mandatory]

EXAMPLE

```
twc deleteipv6dnsserver --ip=5000::2
```

deleteipv6dnsreversezone**NAME**

deleteipv6dnsreversezone

DESCRIPTION

Deletes an IPv6 DNS reverse zone from the TCPWave IPAM. An IPv6 DNS reverse zone can be deleted by specifying IP address, mask length and the organization name of the DNS reverse zone.

ARGUMENTS

--ip

IPv6 address of the DNS reverse zone to be deleted. [mandatory]

--name

Name of the IPv6 DNS reverse zone to be deleted. [mandatory]

--mask

Mask length of the reverse zone to be deleted. [mandatory]

--org

Organization name associated with the reverse zone. [mandatory]

EXAMPLE

```
twc deleteipv6dnsreversezone --ip="1236::" --name="0.0.6.3.2.1.ip6.arpa" --mask=24 --  
org=TCPWave
```

deleteipv6dhcpserver**NAME:**

deleteipv6dhcpserver

DESCRIPTION:

Deletes an IPv6 DHCP Server from the TCPWave IPAM.

ARGUMENTS:

--ip

IPv6 Address of the DHCP Server. [mandatory]

EXAMPLE USAGE:

```
twc deleteipv6dhcpserver --ip=2001:db8::4
```

deleteipv6dhcptmpl**NAME:**

deleteipv6dhcptmpl

DESCRIPTION:

Deletes an IPv6 DHCP option template from the TCPWave IPAM.

ARGUMENTS:

--name

Name of the IPv6 DHCP option template to be deleted. [mandatory]

--org

Organization name associated with the template. This argument is mandatory for the users in FADM role to select appropriate organization to which the operation must be applied. For users not in FADM role, the operation is applied to the organization that the user is associated with.

EXAMPLE USAGE:

```
twc deleteipv6dhcptmpl --name="voip devices template" --org=TCPWave
```

deleteipv6network**NAME:**

deleteipv6network

DESCRIPTION:

Deletes an IPv6 network from the TCPWave IPAM.

ARGUMENTS:

```
--network
    Network start address [mandatory]

--org
    Name of the organization. This argument is required if the user is
    'FADM'.
```

EXAMPLE USAGE:

```
twc deleteipv6network --network=2000:: --org=TCPWave
```

deleteipv6block**NAME**

deleteipv6block - Deletes an IPv6 address block from the TCPWave IPAM.

DESCRIPTION

Deletes an IPv6 address block from an IPv6 address pool defined in the TCPWave IPAM.

ARGUMENTS

```
--ip
    Address of the IPv6 address block. [mandatory]

--org
    Name of the organization in which IPv6 address block exist. [mandatory]
```

EXAMPLE

```
twc deleteipv6block --ip=2000:: --org=TCPWave
```

deleteipv6object**NAME:**

deleteipv6object

DESCRIPTION:

Deletes an IPv6 object from the TCPWave IPAM.

ARGUMENTS:

```
--object
    IPv6 address of the target object to be deleted. [mandatory]
```

--org

Name of the organization in which the operation must be performed.

This argument is mandatory if the user is 'FADM'.

--reference_delete

Deletes all the referenced resource records of the object. It takes 0 or

1. If it is specified, as 1 all the referenced resource records will be deleted. Default is 0.

--error_file

Path on the target IPAM server to the file to write the referenced resource records. If the file path is not specified, the error output is written to the standard output.

EXAMPLE USAGE:

```
twc deleteipv6object --object=2000:0:0:a000::3 --org=TCPWave --reference_delete=1
```

```
twc deleteipv6object --object=2000:0:0:a000::3 --reference_delete=0 --  
error_file=/tmp/referenced_rerords.txt  
twc deleteipv6object --object=2000:0:0:a000::3
```

deleteipv6scope

NAME:

deleteipv6scope

DESCRIPTION:

Deletes an IPv6 DHCP scope specified, by --start_ip and --end_ip arguments from the TCPWave IPAM .

ARGUMENTS:

--start_ip

Start IPv6 address of the DHCP scope [mandatory]

--end_ip

End IPv6 address of the DHCP scope [mandatory]

--org

Name of the organization to which the scope belongs. This argument is mandatory if the user is 'FADM'.

EXAMPLE USAGE:

```
twc deleteipv6scope --start_ip=5000:0:0:20::4 --end_ip=5000:0:0:20::9 --org=TCPWave
```

```
twc deleteipv6scope --start_ip=5000:0:0:20::10 --end_ip=5000:0:0:20::30
```

deleteipv6subnet

NAME:

deleteipv6subnet

DESCRIPTION:

Deletes an IPv6 subnet from the TCPWave IPAM.

ARGUMENTS:

--subnet

IPv6 address of the target subnet. [mandatory]

--org

Name of the organization in which the operation must be performed.
This argument is required if the user is 'FADM'.

EXAMPLE USAGE:

```
twc deleteipv6subnet --subnet=5001:0:0:3000::
```

```
twc deleteipv6subnet --subnet=5001:0:0:5000:: --org=TCPWave
```

deleteipv6subnetmpl

NAME

deleteipv6subnetmpl

DESCRIPTION

Deletes a IPv6 subnet template data from the TCPWave IPAM. A IPv6 subnet template can be deleted by specifying subnet template name and organization name.

ARGUMENTS

--name

Name of the IPv6 subnet template [mandatory].

--org

Name of the organization [mandatory].

EXAMPLE

```
twc deleteipv6subnetmpl --name=FirstTemplate --org=Internal
```

deleteipv6subnetgroup**NAME**

deleteipv6subnetgroup

DESCRIPTION

Deletes an IPv6 subnet group from the TCPWave IPAM.

ARGUMENTS

--name

Name of the IPv6 subnet group being deleted. [mandatory]

--org

Organization name of the IPv6 subnet group. [mandatory]

--disassociate_reference

Disassociates IPv6 subnet group references if set to yes.

EXAMPLE

```
twc deleteipv6subnetgroup --name=Sales_SG1 --org=EARTH --disassociate_reference=no
```

deletelocation**NAME:**

deletelocation

DESCRIPTION:

Deletes a location from the TCPWave IPAM. Target organization must be specified, by name using --org parameter if the user is FADM.
Target location must be specified, in terms of the mandatory address fields

ARGUMENTS:

--street1

Street1 field of the location address [mandatory]

-
- street2
Street2 field of the location address.
 - city
City field of the location address [mandatory]
 - state
State field of the location address [mandatory]
 - country
Country field of the location address [mandatory]
 - zip
Zip code field of the location address [mandatory]
 - org
Organization name to which the location is associated. This argument is mandatory if the user is FADM. [mandatory]

EXAMPLE USAGE:

```
twc deletelocation --street1="600 ALEXANDER ROAD" --city="PRINCETON" --state=NJ --country=USA --zip=08540 --org=TCPWave
```

```
twc deletelocation --street1="600 ALEXANDER ROAD" --city="PRINCETON" --state=NJ --country=USA --zip=08540
```

deletelogchannel**NAME:**

deletelogchannel

DESCRIPTION:

Deletes a DNS log channel from the TCPWave IPAM.

ARGUMENTS:

- name
Name of the Log channel to be deleted [mandatory]

EXAMPLE USAGE:

```
twc deletelogchannel --name=queries
```

deletemirroredzone**NAME:**

deletemirroredzone

DESCRIPTION:

Deletes a DNS managed mirrored Zone from the TCPWave IPAM.
Deleting a mirrored zone will not affect the parent zone.

ARGUMENTS:

--org
Organization name from which the mirrored zones is being delete. This argument is mandatory if the user is FADM.

--name
Name of the mirrored zone. [mandatory]

--zone_name
Name of the DNS zone associated with the mirrored zone. [mandatory]

EXAMPLE USAGE:

twc deletemirroredzone --org=TCPWave --name=tcpwave1.com --zone_name=tcpwave.com
deletemicrosoftdhcpserver

NAME:

deletemicrosoftdhcpserver

DESCRIPTION:

Deletes Microsoft DHCP appliance from the TCPWave IPAM.

ARGUMENTS:

--addr
IP address of the appliance. [mandatory]

EXAMPLE USAGE:

twc deletemicrosoftdhcpserver --addr=10.0.0.10
deletemicrosoftdnsserver

NAME:

deletemicrosoftdnsserver

DESCRIPTION:

Deletes Microsoft DNS appliance from the TCPWave IPAM.

ARGUMENTS:

--addr
IP address of the appliance. [mandatory]

EXAMPLE USAGE:

twc deletemicrosoftdnsserver --addr=10.0.0.10
deletenetwork

NAME:

deletenetwork

DESCRIPTION:

Deletes a network from the TCPWave IPAM.

ARGUMENTS:**--network**

Network start address [mandatory]

--org

Name of the organization. This argument is required if the user is 'FADM'.

EXAMPLE USAGE:

twc deletenetwork --network=80.0.0.0 --org=TCPWave

twc deletenetwork --network=80.0.0.0

deleteobject**NAME:**

deleteobject

DESCRIPTION:

The twc deleteobject CLI command is used to delete a specified, object from the TCPWave IP Address Management system. The syntax of this command is shown below. The user invoking this command is expected to have authentication permission and should be authorized to perform delete object. The TCPWave IPAM audits this action. Successful completion of this command exits with a status code 0.

Target object should be specified, using --ip parameter.

ARGUMENTS:**--object**

IP address of the object to be deleted. [mandatory]

--org

Name of the organization to which the object belongs. This argument is mandatory if user is 'FADM'.

--rr_reference_delete

Deletes all the referenced resource records of the object. It takes 0 or 1. If it is specified, as 1 all the referenced resource records will be deleted. Default is 0.

--error_file

Path on the target IPAM server to the file to write the referenced resource records. If the file path is not specified, the error output is written to the standard output.

EXAMPLE USAGE:

twc deleteobject --object=10.1.0.4 --org=TCPWave --rr_reference_delete=1

twc deleteobject --object=10.1.0.5 --org=TCPWave --rr_reference_delete=0 --error_file=/tmp/referenced_rerords.txt

```
twc deleteobject --object=10.1.0.6 --org=TCPWave
```

deleteobjecttype**NAME:****deleteobjecttype****DESCRIPTION:**

Deletes an object type from the TCPWave IPAM.

An object type can be deleted by specifying unique code of the object type.

Note:- For a successful delete operation it is mandatory that object type is not referenced to any object.

ARGUMENTS:**--code**

Unique code of the object type to be deleted [mandatory]

EXAMPLE USAGE:

```
twc deleteobjecttype --code="3G Phone"
```

deleteorg**NAME:****deleteorg****DESCRIPTION:**

Deletes an organization from the TCPWave IPAM.

An organization can be deleted by specifying name of the organization.

Note:- Before deleting an organization it is mandatory to delete all the associated IPAM entities of the target organization.

ARGUMENTS:**--name**

Name of the target organization being deleted [mandatory]

EXAMPLE USAGE:

```
twc deleteorg --name="TCPWave"
```

deletepatch**NAME:****deletepatch****DESCRIPTION:**

Deletes a patch from the TCPWave IPAM.

ARGUMENTS:

--package_name
Name of the package. [mandatory]

--major_version
Major version of the TIMS. [mandatory]

--minor_version
Minor version of the TIMS. [mandatory]

--patch_name
Name of the patch. [mandatory]

--appliance_type
Type of the server. [mandatory]

--patch_level
Level of the patch. [mandatory]

--patch_status
Status of the patch. [mandatory]

EXAMPLE USAGE:

twc deletepatch --package_name=TCPWaveIPAM --major_version=11 --minor_version=27 --patch_name="Test patch" --appliance_type=IPAM --patch_level=2 --patch_status=0
deletepztmpl

NAME:

deletepztmpl - Deletes a DNS Response policy zone(RPZ) template from the TCPWave IPAM.

DESCRIPTION:

Deletes a DNS Response policy zone(RPZ) template from the TCPWave IPAM.

ARGUMENTS:

--name
Name of the DNS Response policy zone(RPZ) template to be deleted from the TCPWave IPAM. [mandatory]

EXAMPLE USAGE:

twc deletepztmpl --name="RPZ-Template"

deleterr**NAME:**

deleterr

DESCRIPTION:

Deletes the DNS resource records in 'object', 'zone' or 'reverse zone' scopes.

ARGUMENTS:**--rr_scope**

Takes 'object', 'zone' or 'revzone'. Defines the context in which the resource record is being added [mandatory]

--obj_addr

IP address of the target object in TCPWave IPAM rr_scope argument is specified, as 'object'.

--zone_name

Zone name of the target zone in TCPWave IPAM when rr_scope argument is specified, as 'zone'.

--addr

IP Address of the reverse zone in TCPWave IPAM when rr_scope argument is specified, as 'revzone'.

--mask

Mask length of the reverse zone in TCPWave IPAM when rr_scope argument is specified, as 'revzone'.

--type

Indicates the type of the resource record. Takes one of 'A','CNAME', 'MX','SRV','NS','TXT','NAPTR','PTR','AAAA','DNAME','HINFO','CAA','LOC' or 'TLSA'. [mandatory]

--owner

Should be a valid FQDN [mandatory]

--data

Type specific data part of the resource record [mandatory]

--org

Organization name to be specified, for resource records of type NS. If this argument is omitted then the root zone will be selected from organization that the user is associated with.

--is_proxy

DNS Proxy root zone flag. It takes '0' or '1'. If it is specified, as '1' resource record being deleted will be deleted from proxy root zone. If it is specified, as '0' resource record being deleted will be deleted from root zone. This argument is applicable when --rr_scope=zone and --zone_name=(dot).

EXAMPLE USAGE:

```
twc deleterr --rr_scope=zone --zone_name=tcpwave.com --type=A --owner=ftp.tcpwave.com. --data=192.168.1.113 --org=TCPWave
```

```
twc deleterr --rr_scope=zone --zone_name=tcpwave.com --type=CNAME --
```

```
owner=ftp.tcpwave.com. --data=www.tcpwave.com. --org=TCPWave

tvc deleterr --rr_scope=zone --zone_name=tcpwave.com --type=MX --owner=mailtcpwave.com. --data="10 www.tcpwave.com." --org=TCPWave

tvc deleterr --rr_scope=zone --zone_name=tcpwave.com --type=SRV --owner="_sipinternalts._tcp.tcpwave.com." --data="0 100 3268 cdmdev.tcpwave.com." --org=TCPWave

tvc deleterr --rr_scope=zone --zone_name=tcpwave.com --type=NAPTR --owner="naptr.tcpwave.com." --data="30 100 \"U\" \"E2U+email\" \"\!^.*\$\!info@tcpwave.com\!i\" ." --org=TCPWave

tvc deleterr --rr_scope=zone --zone_name=tcpwave.com --type=NS --owner="oxf.tcpwave.com." --data="ns81-qrs01.apac.tcpwave.com." --org=TCPWave

tvc deleterr --rr_scope=zone --zone_name=tcpwave.com --type=TXT --owner=text.tcpwave.com. --data="spf1 a:mail.tcpwave.com -all" --org=TCPWave

tvc deleterr --rr_scope=object --obj_addr=192.168.1.113 --type=A --owner=dev.tcpwave.com. --data=192.168.1.113 --org=TCPWave

tvc deleterr --rr_scope=object --obj_addr=192.168.1.113 --type=CNAME --owner=ftp.tcpwave.com. --data=www.tcpwave.com. --org=TCPWave

tvc deleterr --rr_scope=object --obj_addr=192.168.1.113 --type=MX --owner=mailtcpwave.com. --data="10 www.tcpwave.com." --org=TCPWave

tvc deleterr --rr_scope=object --obj_addr=192.168.1.113 --type=SRV --owner="_sipinternalts._tcp.tcpwave.com." --data="0 100 3268 cdmdev.tcpwave.com." --org=TCPWave

tvc deleterr --rr_scope=object --obj_addr=192.168.1.113 --type=NAPTR --owner="naptr.tcpwave.com." --data="30 100 \"U\" \"E2U+email\" \"\!^.*\$\!info@tcpwave.com\!i\" ." --org=TCPWave

tvc deleterr --rr_scope=revzone --addr=192.168.1.0 --mask=28 --type=PTR --owner=ftp.tcpwave.com. --data=192.168.1.113 --org=TCPWave

tvc deleterr --rr_scope=zone --zone_name=. --type=CNAME --owner=ftp.tcpwave.com. --data=www.tcpwave.com. --is_proxy=0 --org=TCPWave

tvc deleterr --rr_scope=zone --zone_name=. --type=CNAME --owner=ftp.tcpwave.com. --data=www.tcpwave.com. --is_proxy=1 --org=TCPWave

tvc deleterr --rr_scope=object --obj_addr=172.13.2.13 --type=TLSA --owner="_9443._tcp.tcpwave.com." --data="2:0:2:0ff0ebee2e9be02487662a6caa238f9c329344a9c0e146dccc74b1bbb84a51e6f762fa9e33ba6d6acd86581184f97c18ca885b753a6bb42f918ff6b6a17801e1" --org=TCPWave

tvc deleterr --rr_scope=zone --zone_name=tcpwave.com --type=AAAA --
```

```
owner=dev.tcpwave.com. --data="5000::1" --org=TCPWave

tvc deleterr --rr_scope=zone --zone_name=tcpwave.com --type=DNAME --owner=tcpwave.com.
--data=tcpwave1.com. --org=TCPWave

tvc deleterr --rr_scope=zone --zone_name=tcpwave.com --type=HINFO --
owner=dev.tcpwave.com. --data="PC-Intel-700mhz:Redhat Linux 7.1" --org=TCPWave

tvc deleterr --rr_scope=zone --zone_name=tcpwave.com --type=CAA --owner=tcpwave.com. --
data="0:issue:example.com" --org=TCPWave

tvc deleterr --rr_scope=zone --zone_name=tcpwave.com --type=LOC --
owner=arec.tcpwave.com. --data="52:22:23.000:N:4:53:32.000:E:-2.00:0.00:10000:10" --
org=TCPWave
```

deletescheduledjob

NAME:

deletescheduledjob

DESCRIPTION:

Deletes a scheduled job from the TCPWave IPAM.

ARGUMENTS:

--job_id
Id of the scheduled job. [mandatory]

EXAMPLE USAGE:

```
tvc deletescheduledjob --job_id=ScheduledJobId
```

deletescope

NAME:

deletescope

DESCRIPTION:

Deletes a DHCP scope specified, by --start_ip and --end_ip arguments from the TCPWave IPAM

ARGUMENTS:

--start_ip
Start IP address of the DHCP scope [mandatory]

--end_ip

End IP address of the DHCP scope [mandatory]

--org

Name of the organization to which the scope belongs. This argument is mandatory if user is 'FADM'.

EXAMPLE USAGE:

```
twc deletescope --start_ip=10.1.10.51 --end_ip=10.1.10.142 --org=TCPWave
```

```
twc deletescope --start_ip=10.1.10.51 --end_ip=10.1.10.142
```

deletescopeactivelease

NAME:

deletescopeactivelease - Deletes an active lease for a given scope range from the TCPWave IPAM.

DESCRIPTION:

Deletes an active lease for a given scope range from the TCPWave IPAM.

ARGUMENTS:

--start_addr

Start address of the scope [mandatory]

--end_addr

End address of the scope [mandatory]

--org

Name of the organization from where the scope of the active leases to be deleted [mandatory]

EXAMPLE:

```
twc deletescopeactivelease --start_addr=10.1.10.105 --end_addr=10.1.10.110 --org=TCPWave
```

deletesubnet

NAME:

deletesubnet

DESCRIPTION:

Deletes a subnet from the TCPWave IPAM.

ARGUMENTS:

--subnet

IP Address of the subnet to be deleted.[mandatory]

--org

Name of the organization to which subnet belongs. This argument is mandatory if the user is 'FADM'.

--reference_delete
Deletes all the referenced resource records in subnet.

EXAMPLE USAGE:

twc deletesubnet --subnet=50.0.0.0 --org=TCPWave --reference_delete=1

twc deletesubnet --subnet=50.0.0.0 --reference_delete=1

deletesubnettmpl

NAME:

deletesubnettmpl

DESCRIPTION:

Deletes a subnet template data from the TCPWave IPAM.

A subnet template can be deleted by specifying subnet template name and organization name.

ARGUMENTS:

--name
Name of the subnet template. [mandatory]

--org
Name of the organization. [mandatory]

EXAMPLE USAGE:

twc deletesubnettmpl --name=FirstTemplate --org=Internal

deletesubnetgroup

NAME:

deletesubnetgroup

DESCRIPTION:

Deletes a subnet group from the TCPWave IPAM.

ARGUMENTS:

--name
Name of the subnet group being deleted [mandatory]

--org
Organization name of the subnet group [mandatory].

--disassociate_reference
Disassociates subnet group references if set to yes.

EXAMPLE USAGE:

twc deletesubnetgroup --name=Sales_SG1 --org=EARTH --disassociate_reference=no

deleteasset**DESCRIPTION:**

Deletes an asset data form a TCPWave IPAM.

ARGUMENTS:

--service_tag

Service tag for the asset. [mandatory]

EXAMPLE:

```
twc deleteasset --service_tag="Test"
```

```
twc deleteasset --service_tag="One,two"
```

deletevrf**NAME**

deletevrf

DESCRIPTION

Deletes a VRF from the TCPWave IPAM.

ARGUMENTS

--name

Name of the VRF. [mandatory]

--org

Name of the organization. [mandatory]

EXAMPLE:

```
twc deletevrf --name=testVrf --org=TCPWave
```

deploypatch**NAME:**

deploypatch

DESCRIPTION:

Deploy a patch in the TCPWave IPAM.

Uploaded patch can only be deploy in the TCPWave IPAM.

ARGUMENTS:

--file_name

Name of the zip file. [mandatory]

--package_name

Name of the package. [mandatory]

--major_version
Major version of the TIMS. [mandatory]

--minor_version
Minor version of the TIMS. [mandatory]

--patch_name
Name of the patch. [mandatory]

--appliance_type
Type of the appliance. [mandatory]

--patch_level
Level of the patch. [mandatory]

--schd_time
Specifies the schedule time, time format should be "YYYY-MM-DD HH:MM:SS".

EXAMPLE USAGE:

```
twc deploypatch --file_name=Patch_IPAM_11.27_2_6301.zip --package_name=TCPWaveIPAM --  
major_version=11 --minor_version=27 --patch_name="Test patch" --appliance_type=IPAM --  
patch_level=2
```

```
twc deploypatch --file_name=Patch_IPAM_11.27_2_6301.zip --package_name=TCPWaveIPAM --  
major_version=11 --minor_version=27 --patch_name="Test patch" --appliance_type=IPAM --  
patch_level=2 --schd_time="2019-05-08 03:12:00"
```

disablezonemonitor**NAME:**

disablezonemonitor

DESCRIPTION:

Disables the monitoring on a given list of zones from the TCPWave IPAM.

ARGUMENTS:

--zone_list
Takes comma separated list of zone names. [mandatory]

--org
Name of the organization to which the specified, zones belongs. This argument is mandatory if the user is FADM.

EXAMPLE USAGE:

```
twc disablezonemonitor --zone_list=dev.tcpwave.com,tcpwave.com --org=TCPWave
```

```
twc disablezonemonitor --zone_list=dev.tcpwave.com,tcpwave.com
```

deletediscovertask**NAME:**

deletediscovertask

DESCRIPTION:

Deletes the results of a given subnet discovery task from the TCPWave IPAM.

ARGUMENTS:

--id

Command ID of the discovered subnet. Use the following command to see all the command IDs of discovered subnets: 'twc listdiscovertask --d='.
[mandatory]

EXAMPLE USAGE:

twc deletediscovertask --id=1391

deletedhcpsharednetwork**NAME:**

deletedhcpsharednetwork

DESCRIPTION:

Deletes DHCP shared network in the TCPWave IPAM.

ARGUMENTS:

--name

Name of the shared network. [mandatory]

--org

Name of the organization. [mandatory]

EXAMPLE:

twc deletedhcpsharednetwork --name=TestNet --org=TcpWave

deletecloudprovider**NAME:**

deletecloudprovider

DESCRIPTION:

Deletes cloud provider from the TCPWave IPAM.

ARGUMENTS:

--name

Name of the cloud provider. [mandatory]

--org

Name of the organization. [mandatory]

EXAMPLE:

twc deletecloudprovider --name=aws --org= TcpWave

deletensmtmpl

NAME

deletensmtmpl

DESCRIPTION

Deletes NSM template data form a TCPWave IPAM.

ARGUMENTS

--org

Organization of the NSM Template. [mandatory]

--tmpl_name

NSM Template name. [mandatory]

EXAMPLE

twc deletensmtmpl --org="TCPWave" --tmpl_name=FirstTemplate

discover

NAME:

discover

DESCRIPTION:

Discovers a specified, element in the TCPWave IPAM.

TCPWave IPAM Discovery engine can perform deep network and port scans on a network and discover network devices in them. The discovery process can be triggered on any network in the IPAM and the results are stored into the database. Users can either reject or accept the discovered IP Addresses into the IPAM. This automatic discovery process is an easy and fast way to add existing IP Addresses into the IPAM. The discovery process can be scheduled to run repetitively or at a particular time in the future.

ARGUMENTS:

--element_type

Element type to be discovered. It takes 'subnet' or 'object'.[mandatory]

--ip

IP address of the 'subnet' or 'object'. If it is a 'subnet' it should contain IP address along with mask separated by '/'. [mandatory]

--org

Name of the organization to which specified, subnet/object belongs. This argument is mandatory if the user is 'FADM'.

--apply

Apply the changes discovered for object discovery. (Applicable only for objects)

EXAMPLE USAGE:

```
twc discover --element_type=subnet --ip=10.1.10.0/24 --org=TCPWave
```

```
twc discover --element_type=object --ip=10.1.10.3 --org=TCPWave --apply=1/0
```

displayzonedata**NAME****displayzonedata****DESCRIPTION**

Displays the contents of a DNS zone from the TCPWave IPAM.

ARGUMENTS**--zone_name**

Zone name for which the zone contents are to be displayed. [mandatory]

--org

Organization name in which zone exists. [mandatory]

--d

Delimiter character separating the columns. If this argument is not specified then comma will be used as a delimiter.

EXAMPLE

```
twc displayzonedata --zone_name=tcpwave.com --org=TCPWave --d=,
```

downloaddhcpconfig**NAME****downloaddhcpconfig****DESCRIPTION**

Downloads the configuration of a DHCP Server in the TCPWave IPAM.

ARGUMENTS**--ip**

IP Address of the DHCP server.[mandatory]

--output_file

Full path to the output file to which the server configuration is to be written.[mandatory]

EXAMPLE USAGE:

```
twc downloaddhcpconfig --ip=10.1.10.180 --output_file=/tmp/dhcpconfig
```

downloaddnsconfig**NAME:****downloaddnsconfig**

DESCRIPTION:

Downloads the configuration of a DNS Server in the specified, zip file in the TCPWave IPAM.

ARGUMENTS:

--ip
IP Address of the DNS server.[mandatory]

--output_file
Full path to the output zip file to which the server configuration files are to be compressed.[mandatory]

EXAMPLE USAGE:

twc downloaddnsconfig --ip=10.1.10.180 --output_file=/tmp/dnsconfig.zip

dumpdb**NAME:**

dumpdb

DESCRIPTION:

Dumps a snapshot of TCPWave IPAM database. A new dump directory Dump_<timestamp> is created under a pre-configured parent directory. Snapshots are placed under Dump_<timestamp>/snapshot. Incremental changes are placed under Dump_<timestamp>/binlogs

ARGUMENTS:

- NA

- EXAMPLE USAGE:

twc dumpdb

Updates**editasset****DESCRIPTION:**

Updates an asset data in the TCPWave IPAM

ARGUMENTS:

--service_tag
Service tag for the asset. [mandatory]

--new_service_tag
New service tag for the asset.

--serial_num
Unique serial number for the asset.

--vendor

Name of the vendor for the asset.

--model

Model value for the asset.

--name

Name for the asset.

--acquisition_type

Model acquisition type for the asset.

--maintenance_cost

Cost value of the asset for maintenance.

--cpu

cpu range of the asset.

--capacity

Capacity value for the asset.

--os_version

Supported os version for the asset.

--purchase_cost

Purchase value of the asset.

--disposal_reason

Reason for the disposal of asset.

--city

City name for the asset.

--green_zone

Green zone is to be set for asset.

--warranty_end_dt

Last Warranty date of the asset, date format is yyyy/mm/dd.

--purchase_dt

Purchase date of the asset, date format is yyyy/mm/dd.

--disposal_dt

Disposal date of the asset, date format is yyyy/mm/dd.

--description

Description for the asset.

EXAMPLE:

```
twc editasset --service_tag="Dell Inc." --serial_num=36906 --green_zone=1 --name=test
```

editcloudprovider

DESCRIPTION: Updates a cloud provider in the TCPWave IPAM. Different type of cloud providers support different credentials. Follow the example section to edit particular type of cloud provider

ARGUMENTS:

--org

Organization name to be associated with the cloud provider. This argument is for users in FADM role to select appropriate organization to which the operation has to be applied. For users not in FADM role, the operation is by default applied to the organization that the user is associated with.

--provider_type

Type of the Cloud provider. In TCPWave IPAM provider type represents the cloud service provider . TCPWave IPAM support following cloud providers: 'AKAMAI', 'AWS', 'AZURE', 'CLOUDFLARE', 'DYNDNS' and 'GOOGLE' [mandatory].

--name

Name of the cloud provider [mandatory].

--new_name

New name of the cloud provider.

--user

User name of the cloud provider.

--account

Valid AWS account number which manages the resource.

--iam_role

Use IAM Role mapped to the EC2 instance. It is applicable for AWS cloud provider.

--api_key

API key for the cloud provider. It is Secret access key for AWS and it is global API key for CLOUDFLARE provider .

--keystore_file

Key store file for the cloud provider. This key store file contains Secret access key. It is applicable only for AZURE provider type.

--application

Application ID for the cloud provider. It is applicable for AZURE cloud

provider.

--service_account_id

Service Account ID for the GOOGLE cloud provider .

--p12file

p12file for the GOOGLE cloud provider .

--project_id

Project ID for the GOOGLE cloud provider .

--ad_tenant

Ad tenant ID for the AZURE cloud provider .

--update_password

Update the AWS, Azure and DYNDNS cloud providers password, takes input as '1' to update the password.

--resource_group

Resource Group for the AZURE cloud provider.

--customer_name

Customer name for the DYNDNS cloud provider .

--email

Email Address for the CLOUDFLARE provider .

--region

Region defines area of AWS cloud provider. For AWS Cloud provider region can be one of the following-

US East (N. Virginia)

US East (Ohio)

US West (N. California)

US West (Oregon)

Asia Pacific (Mumbai)

Asia Pacific (Seoul)

Asia Pacific (Singapore)

Asia Pacific (Sydney)

Asia Pacific (Tokyo)

Canada (Central)

EU (Frankfurt)

EU (Ireland)

EU (London)

South America (Sao Paulo)

--desc

Description for the cloud provider .

EXAMPLE:

```
twc editcloudprovider --name="AWS-Provider50" --org=TCPWave --
user="AKIAINLQMOG7KBWXMTOQP" --update_password=1 --region="EU (London)" --desc="AWS
Cloud Provider"
```

```
twc editcloudprovider --name="AWS-Provider50" --org=TCPWave --iam_role=1 --region="EU
(London)" --desc="AWS Cloud Provider"
```

```
twc editcloudprovider --name="Google-Provider22" --org=TCPWave --
service_account_id="jhon@tcpwave-14981012.iam.gserviceaccount.com" --p12file="/tmp/tcpwave-
2d185caa49dc1.p12" --project_id="tcpwave-14912810" -desc="Google Cloud Provider"
```

```
twc editcloudprovider --name="Azure-Provider03" --org=TCPWave --user="ppc0e31c0f-fdb0-
438c-afff-6ea7600b0e61" --keystore_file="cloud_dns_app.pfx" --application="ebe1b568-5e63-46f0-
9201-8a465cee092dqq" --ad_tenant="772a8482-16c9-4823-9f15-bd19827d23f111" --
resource_group="tcpwave" --desc="Azure Cloud Provider"
```

```
twc editcloudprovider --name="DynDNS-Provider01" --org=TCPWave --user="jhon-smith" --
customer_name="tcpwave01" --update_password=1 --desc="DYNDNS Cloud Provider"
```

```
twc editcloudprovider --name="CLOUDFLARE-Provider02" --org=TCPWave --
email="jhon.tcpwave@tcpwave.com" --api_key="3cde9f553a9a21049e00046" --
desc="CLOUDFLARE Cloud Provider"
```

```
twc editcloudprovider --name="Akamai-Provider06" --org=TCPWave --
user="jhon.tcpwave@tcpwave.com" --api_key="client_secret" =
xd3RTCMlmmZhdQ82LD34yAZUqOwc2DDt1ANgDAoc6iguY=host = akab-34nyw47p22fhpvptnu-
v7ygacgwkb6cswza.luna.akamaiaapis.net access_token = akab-a24w5rojdc6lckdm-
cvscbkoo5ise5bw2 client_token = akab-sxdp7uvgkonm7jfu-w3phslypnzzv3llqv" --desc="AKAMAI
Cloud Provider"
```

editadminrole**NAME:**

editadminrole

DESCRIPTION:

Updates an administrator role in the TCPWave IPAM.

ARGUMENTS:

--name

Name of the administrator role. [mandatory]

--new_name

New name of the administrator role.

--functions

Name of the functions that accepts the comma separated function.

--desc

Description for an administrator role.

EXAMPLE:

```
twc editadminrole --name=CADM --functions="Quick Tasks,Appliance Groups,Bulk Data Import" --desc="Custom Administrator"
```

```
twc editadminrole --name=CADM --new_name=EADM --functions="Appliance Groups,Bulk Data Import"
```

editadminpermission**NAME:**

editadminpermission

DESCRIPTION:

Updates an administrator/administrator group permission in the TCPWave IPAM.

ARGUMENTS:

--level

It takes the input as admin or admin group, if the level is admin the input param of admin is mandatory otherwise admin group is mandatory.

--admin

Name of the admin.

--admin_group

Name of the admin group.

--org

Name of the organization. [mandatory]

--role

Name of the administrator role. [mandatory]

--privilege

Name of the privilege, It takes the input as 'Read' or 'Write' or 'Deny'. [mandatory]

--function

Name of the administrator function. [mandatory]

--function_value

Value of the administrator function. [mandatory]

--select_all

It takes the input as '0' or '1'. [mandatory]

EXAMPLE:

```
twc editadminpermission --level=Admin --admin=Test --privilege=Write --function="IPv4 Networks" --function_value=10.1.10.0 --org=TcpWave --role=EADM --select_all=0
```

```
twc editadminpermission --level="Admin Group" --admin_group="Test Group" --privilege=Deny --function="TCPWave DHCP IPv4 Appliances" --function_value=TCPWave00001Remote --org=TcpWave --role=CADM --select_all=1
```

editadmin**NAME**

editadmin

DESCRIPTION

Updates the profile of the TCPWave IPAM administrator. The user is identified uniquely by the combination of First Name, Last Name, and Email Id. Each administrator user is assigned with a role that controls their access to the system. When an administrator user exceeds the maximum number of failed login attempts, his account will be suspended and has to be re-instated by a User administrator for further access to the system.

ARGUMENTS**--first_name**

First name of the administrator. [mandatory]

--last_name

Last name of the administrator. [mandatory]

--middle_name

Middle name of the administrator.

--email

Email address of the administrator. [mandatory]

--new_first_name

This argument is used if first name is to be updated.

--new_middle_name

This argument is used if middle name is to be updated.

--new_last_name

This argument is used if last name is to be updated.

--new_email

This argument is used if email is to be updated.

--phone

This argument is used if phone number is to be updated.

--login

This argument is used if login name is to be updated.

--groups

This argument is used if administrator groups of the administrator are to be updated. This is a comma separated list of administrator groups.

--org

Default name the organization to which administrator is associated.

--role

Default role of the administrator. [mandatory]

The following roles are the default roles supported by TCPWave IPAM

- SADM - Super Administrator, has access to all the functionality of the system
- FADM - Functional Administrator, Special administrator with functional privileges and valid for the special user 'twcadm' only. This role provides the privileges to switch authentication mechanisms and set system level parameters.
- UADM - User Administrator, Has access to user administration functionality only
- NADM - Normal Administrator, Has privileges only to create Objects and Scopes
- PADM - Power Administrator, Has access to following IPAM entities Zone, Domain, Server, Network, Subnet, Scope, Template and Object
- RADM - Read-only Administrator

--ext_attr

Comma separated list of extension attributes with their values in the

format : extension_attribute_name/extension_attribute_value. Use the following command to see all the extension attributes applied to admin :

```
'twc listext --entity=admin --d=,'
```

EXAMPLE:

```
twc editadmin --first_name=John --last_name=Smith --email=john.smith@tcpwave.com --  
phone=920-310-5555 --login=jsmith --org=TCPWave --groups=default --role=NADM
```

```
twc editadmin --first_name=John --last_name=Smith --email=john.smith@tcpwave.com --  
org=TCPWave --ext_attr=ext_attr_1/value_1,ext_attr_2/value_2
```

```
twc editadmin --first_name=John --middle_name=Fitzgerald --last_name=Kennedy --  
email=john.kennedy@tcpwave.com --phone=920-310-5555 --org=TCPWave --login=jkennedy
```

editadmingroup**NAME**

editadmingroup

DESCRIPTION

Updates an administrator group in the TCPWave IPAM. An administrator group can be updated with a new name, roles, and description.

ARGUMENTS**--name**

Name of the existing administrator group. [mandatory]

--new_name

New name for the administrator group if the name needs to be updated.

--roles

Defines the role of the admin group. Accepts multiple roles with comma separated pairs of role and organization as shown:

Example: SADM,TcpWave;CADM,Internal

Note: FADM and UADM roles are not organization specific.

--desc

Description for the administrator group.

EXAMPLE:

```
twc editadmingroup --name="default-admin-group" --new_name="default-group" --desc="Default Admin Group"
```

```
twc editadmingroup --name="default-admin-group" --roles="SADM,TcpWave;CADM,Internal"
```

editappliancegroup**NAME:**

editappliancegroup

DESCRIPTION:

Updates an appliance group in the TCPWave IPAM.

ARGUMENTS:**--name**

Name of the appliance group being updated. [mandatory]

--org

Name of the organization where the operation must be performed. This argument is mandatory if the user is FADM.

--desc

Description of the appliance group. [updatable field]

--new_name

New name for the appliance group if the name needs to be updated.

--parent

Name of the parent appliance group.

EXAMPLE USAGE:

```
tvc editappliancegroup --name=IT_SG --org=TCPwave --new_name=Sales_SG --desc="IT  
appliance group"
```

```
tvc editappliancegroup --name=IT_SG --org=TCPwave --new_name=Sales_SG --desc="IT  
appliance group" --parent=IT_PG_1
```

editawsimage

NAME:

editawsimage

DESCRIPTION:

Updates an AWS Machine image in the TCPWave IPAM.

ARGUMENTS:

--provider_name

Name of the cloud provider. [mandatory]

--image_id

ID of the AWS Machine Image. [mandatory]

--image_name

Name of the AWS Machine Image.

--desc

Description of the AWS Machine Image.

EXAMPLE USAGE:

```
tvc editawsimage --provider_name="AWS" --image_id="ami-a4c7edb2-test" --  
image_name="AWS Image" --desc="AWS Machine Image-1"
```

editasnumber**NAME**

editasnumber

DESCRIPTION

Updates a Autonomous System Number in the TCPWave IPAM.

ARGUMENTS

--org

Name of the organization to which the Autonomous System Number is associated.

[mandatory]

--name

Name of the Autonomous System Number. [mandatory]

--new_name

New name for the Autonomous System Number.

--as_number

Autonomous System number. It accepts up to 5 digits of the number.

[mandatory]

--email

Email address for the Autonomous System Number.

--desc

Description for the Autonomous System Number..

EXAMPLE

```
twc editasnumber      --name="TCPWave-ASN"    --org=TCPWave   --as_number=1501   --
new_name="TCPWave1" --desc="TCPWave Autonomous System Number."
```

```
twc      editasnumber          --name="ASN"      --org=TCPWave      --as_number=11      --
email="jsmith@tcpwave.com"
```

editcontact

NAME:

editcontact

DESCRIPTION

Update a contact for a given organization in TCPWave IPAM. The target organization has to be specified using --org parameter Target contact has to be specified in terms of the mandatory contact information fields.

ARGUMENTS

--org

Name of the organization to which the contact is associated. [mandatory]

--first_name

First name field of the contact information. [mandatory]

--middle_name

Middle name field of the contact information if any.

--last_name

Last name field of the contact information. [mandatory]

--email

Email ID field of the contact information. [mandatory]

--firstname_new

New first name field of the contact information.

--middlename_new

New middle name field of the contact information.

--lastname_new

New last name field of the contact information.

--email_new

New email id field of the contact information.

--phone_new

New phone number field of the contact information.

EXAMPLE

```
twc editcontact --first_name=John --last_name=Smith --email=john.smith@tcpwave.com --
firstname_new=Mary           --lastname_new=Walker           --org=TCPWave        --
email_new=mary.walker@tcpwave.com --phone_new=920-310-5554
```

```
twc editcontact --first_name=James --middle_name=Francis --last_name=Stuart --
email=james.stuart@tcpwave.com --firstname_new=Mary --middlename_new=del --
lastname_new=Walker --org=TCPWave --email_new=mary.walker@tcpwave.com --phone_new=920-
310-5554
```

editcustomfolder**NAME:**

editcustomfolder

DESCRIPTION:

Only name of the custom folder can be updated so it is mandatory to specify both name and new name.

ARGUMENTS:

--name
Name of the DHCP custom folder. [mandatory]

--new_name
New name of the DHCP custom folder. [mandatory]

EXAMPLE USAGE:

```
twc editcustomfolder --name=voip-options --new_name=voip-option-folder
```

editdhcpfailoverpeer**NAME:**

editdhcpfailoverpeer

DESCRIPTION:

Updates a DHCP failover peer in the TCPWave IPAM.

ARGUMENTS:

--name
Name of the DHCP failover peer. [mandatory]

--org
Organization name in which the DHCP failover peer is being updated. This argument is mandatory if user is FADM.

--new_name
New name of the DHCP failover peer.

--primary_appliance
IP address of the primary appliance.

--failover_appliance
IP address of the failover appliance.

--primary_port

Port number of the primary appliance.

--failover_port
Port number of the failover appliance.

--max_resp_delay
Maximum response delay value.

--mclt
Maximum client lead time value.

--max_unacked_updates
Maximum unacked updates value.

--split
SPLIT value.

--load_bal_max_sec
Load balance maximum second value.

--desc
Description for the DHCP failover peer being updated.

--message_auth
Message authentication takes '0' or '1' as an input.

--shared_secret
Shared secret is mandatory when message_auth is specified as '1'.

--sso_interval
State switch over interval takes the input in seconds.

EXAMPLE USAGE:

```
twc editdhcpfailoverpeer --name=dhcp-failover-peer-1 --org=TCPWave --  
primary_appliance="10.1.10.180" --failover_appliance=10.1.10.185 --primary_port=647 --  
failover_port=647 --max_resp_delay=30 --mclt=1800 --max_unacked_updates=30 --split=120 --  
load_bal_max_sec=3 --desc="DHCP Failover Peer 1"
```

```
twc editdhcpfailoverpeer --name=dhcp-failover-peer-1 --org=TCPWave --new_name=dhcp-  
failover-peer-2
```

```
twc editdhcpfailoverpeer --name=dhcp-failover-peer-1 --org=TCPWave --  
primary_appliance=10.1.10.180 --failover_appliance=10.1.10.185 --appliance_type=msdhcp --  
mclt=1800
```

```
--split=50 --message_auth=1 --shared_secret=abc@123 --sso_interval=100 --desc="DHCP  
Failover Peer 1"
```

editdhcoptionspace

NAME:

editdhcoptionspace

DESCRIPTION:

Updates a DHCP option space in the TCPWave IPAM.
Name of the DHCP option space is mandatory. Name of the DHCP, vendor class and description are editable.

ARGUMENTS:

- name
Name of the DHCP option space. [mandatory]
- new_name
New name of the DHCP option space.
- vendor_class
Name of the vendor class.
- desc
Description for DHCP option space.

EXAMPLE USAGE:

```
twc editdhcoptionspace --name=space --new_name="Option space"
```

```
twc editdhcoptionspace --name=space --vendor_class="vender class" --desc=description  
editdiscoverytmp
```

NAME:

deletediscoverytmpl

DESCRIPTION:

Deletes a discovery template from the TCPWave IPAM.

ARGUMENTS:

- discovery_tmpl
Name of the discovery template. [mandatory]
- org
Name of the organization. [mandatory]

EXAMPLE USAGE:

```
twc deletediscoverytmpl --discovery_tmpl=TestTmpl --org=TCPWave  
editdhcpfingerprint
```

NAME:

editdhcpfingerprint

DESCRIPTION:

Updates DHCP finger print data in the TCPWave IPAM

ARGUMENTS:

- mac_vendor
Name of the MAC vendor.

--mac_bits
First six bits of the MAC address. [mandatory]

--mac_bits_new
First six bits of the MAC address.

--device_profile
Name of the device profile.

--os
Name of the operating system or version info.

--option_sequence
DHCP option sequence. [mandatory]

--option_sequence_new
DHCP option sequence.

--user_agent
Name of the user agent.

--certainty_index
Certainty index value.

EXAMPLE USAGE:

```
tvc editdhcpfingerprint --mac_vendor=Dell --mac_bits=3690e6 --device_profile=profile --os=window --option_sequence=1,2,6,5,4,7 --user_agent=agent --certainty_index=1235 --option_sequence_new=1,4,6,7,8,9,7
```

```
tvc editdhcpfingerprint --mac_vendor=Microsoft --mac_bits=3695e6 --device_profile=profile --option_sequence=1,2,6,5,4,7 --mac_bits_new=6753r4
```

editdhcpoption**NAME:**

editdhcpoption

DESCRIPTION:

Updates a user defined DHCP option in the TCPWave IPAM.

ARGUMENTS:

--name
Name of the DHCP custom option [mandatory]

--new_name
New name of the DHCP custom option.

--option_type

Name of the option type. Takes 'custom' or 'sub-option' as input
[mandatory]

--data_type

New Data type of the option. Updatable field Takes one of the following values

'IP ADDRESS','IP ADDRESS LIST','STRING','DOMAIN','TEXT' or 'BOOLEAN'

--group_name

Name of the DHCP custom folder. Updatable field.

EXAMPLE USAGE:

```
twc editdhcpoption --name=ip-map --new_name=ip-list --option_type=custom --data_type="IP ADDRESS LIST"
```

```
twc editdhcpoption --name=ip-map --option_type=sub-option --group_name="voip-option-folder"
```

editdnsforwarders

NAME:

editdnsforwarders

zone that is not managed by the TCPWave.

DESCRIPTION:

Updates a DNS forwarder that is used to resolve a DNS zone that is not managed by the TCPWave. Forwarders exist on an internal 'BIND CACHE' or 'UNBOUND' DNS server in the TCPWave IPAM.

ARGUMENTS:

--appliance_ip

IP Address of the DNS internal cache server [mandatory]

--appliance_type

Type of the DNS server. Takes 'BIND CACHE' or 'UNBOUND' [mandatory]

--zone

Name of the DNS forward Zone [mandatory]

--fwd_ipv4

Semicolon separated list of ipv4 addresses for forwarding the requests
[updatable]

--fwd_ipv6

Semicolon separated list of ipv6 addresses for forwarding the requests
[updatable]

--desc

Description for the DNS forward zone [updatable]

EXAMPLE USAGE:

```
twc editdnsforwarders --appliance_ip=10.1.10.29 --appliance_type="BIND CACHE" --
zone="tcpwave.com" --fwd_ipv4="10.1.10.204;10.1.10.10" --desc="Default Forward zone"
```

editdnsforwarderstmpl**NAME**

editdnsforwarderstmpl

DESCRIPTION

Updates a DNS forwarder template with forwarder zones which are not managed by TCPWave IPAM.

ARGUMENTS

--group_name

Name of the DNS forwarders group. [mandatory]

--new_group_name

New name for the DNS forwarders group.

--add_forwarders

Specify the forwarders to be added in the below format.

'Zone Name | Semicolon Separated IPV4 Address | Semicolon Separated IPV6 Address'

if need to add second entry separate with comma.

Example: test.com|10.1.10.12;10.1.10.13,zone.com|10.1.10.15;10.1.10.16

--delete_forwarders

Specify the forwarders to be deleted in the below format.

Example: one.com,two.com

--desc

Description for the forwarders group.

EXAMPLE

```
twc editdnsforwarderstmpl --group_name=Test --  
add_forwarders="test.com|10.1.10.12;10.1.10.13,zone.com|10.1.10.15;10.1.10.16" --  
desc="Description for the forwarders"
```

```
twc editdnsforwarderstmpl --group_name=Test --delete_forwarders="one.com,two.com" --  
desc="Description for the forwarders"
```

editdnsreversezone**NAME**

editdnsreversezone

DESCRIPTION

Updates a DNS reverse zone in the TCPWave IPAM. IP address of the reverse zone, mask length (an integer between 8 and 32) and organization name are mandatory fields to be given as input to update a DNS reverse zone.

ARGUMENTS:

--ip

IP address of a DNS reverse zone. [mandatory]

--mask

Mask length (an integer between 8 and 32) [mandatory]

--org

Organization name associated with the reverse zone. [mandatory]

--zone_tmpl

New zone template name to be associated with the DNS reverse zone.

--dnssec

'1' indicates that DNSSEC should be enabled for the reverse zone. '0' indicates that DNSSEC is not enabled.

--nsec_opt

NSEC option for the reverse zone. Takes 'NSEC' or 'NSEC3' as values.

--monit

'1' indicates monitoring is enabled for this reverse zone. '0' indicates monitoring is disabled for this reverse zone.

--dmz_visible

'1' indicates that the reverse zone is visible to cache server rooted at a public internet root server. '0' indicates that the reverse zone is visible. If this argument is not specified the value is defaulted to '0'.

--ext_attr

Comma separated list of extension attributes with their values in the format : extension_attribute_name/extension_attribute_value. Use the following command to see all the extension attributes applied to reverse

```
zone : 'twc listext --entity=zone --d=,'
```

--views

Comma separated list of DNS view names to be associated with the DNS reverse zone. This argument takes 'None' to dissociate the views from specified zone.

--contact_fname

First name of the new contact to be associated with the DNS reverse zone.

--contact_mname

Middle name of the new contact to be associated with the DNS reverse zone.

--contact_lname

Last name of the new contact to be associated with the DNS reverse zone.

--contact_email

Email Id of the new contact to be associated with the DNS reverse zone.

--custom_allow_ns

Custom allow NS should be specified as FQDN. It accepts the server name and IP address by separating them with comma. It accepts multiple values by separating them with pipe symbol.

Example: test1.com.,10.1.10.1|test2.com.,192.168.0.0

--tsig_key_names

It accepts the multiple TSIG key values by separating with comma. Custom allow NS cannot be null to specify TSIG key names.

Example: key1,key2

--desc

New description for the DNS reverse zone.

EXAMPLE:

```
twc editdnsreversezone --ip=10.1.10.0 --mask=24 --org=TCPWave --desc="TCPWave Reverse Zone" --zone_tmpl="base-zone-template" --monit=0
```

```
twc editdnsreversezone --ip=10.1.10.0 --mask=24 --org=TCPWave --desc="TCPWave Reverse Zone" --ext_attr=ext_attr_1/value_1,ext_attr_2/value_2
```

```
twc editdnsreversezone --ip=10.1.10.0 --mask=24 --org=TCPWave --desc="TCPWave Reverse Zone" --zone_tmpl="base-zone-template" --dnssec=1 --nsec_opt=NSEC3
```

```
twc editdnsreversezone --ip=10.1.10.0 --mask=24 --org=TCPWave --contact_fname=John --contact_lname=Smith --contact_email=john.smith@tcpwave.com
```

```
twc editdnsreversezone --ip=10.1.10.0 --mask=24 --org=TCPWave --views=view1/view2 --custom_allow_ns="TCPWaveNs.,10.1.10.1|NS1.,192.168.0.0" --tsig_key_names=key1,key2  
editmicrosoftadserver
```

NAME

editmicrosoftadserver

DESCRIPTION

Updates the existing Microsoft AD appliance in the TCPWave IPAM.

ARGUMENTS

--ip

IP address of the appliance. [mandatory]

--org

Name of the organization. [mandatory]

--new_addr

New IP address of the appliance.

--use_https

Takes '0' or '1' as input.

--user_name

User name for the Microsoft appliance.

--desc

Description for the Microsoft AD appliance.

EXAMPLE

```
twc editmicrosoftadserver --ip=10.0.0.10 --new_addr=10.0.0.12 --org=TCPWave --
user_name=profile
```

```
twc editmicrosoftadserver --ip=10.0.0.10 --org=TCPWave --use_https=1 --user_name=user --
desc="First Microsoft Appliance"
```

editdnsrootzone

NAME:

editdnsrootzone

DESCRIPTION:

Updates a DNS root zone in the TCPWave IPAM. Time formats are specified, as integer/time_unit. Time_unit can be one of the following values: S,MIN,H,D,W,MON,Y representing seconds, minutes, hours, days, weeks, months, years Example: 84600/S or 30/D

ARGUMENTS:

--org

Organization name under which the DNS root zone is being updated.
[mandatory]

--desc

Description	for	the	DNS	root	zone.
-------------	-----	-----	-----	------	-------

--dnssec

Takes '1' or '0'. '1' indicates that DNSSEC should be enabled for the root zone. '0' indicates that DNSSEC is not enabled.

--nsec_opt

NSEC option for the root zone. Takes 'NSEC' or 'NSEC3' as values. This argument should be specified, if DNSSEC is enabled.

--default_ttl

Default TTL for the root zone. Should be specified, in time format as described in the description.

--soa_email

Email id associated with the SOA record for the DNS root zone.

--soa_refresh

Refresh time associated with the SOA record for the DNS root zone.

Should be specified, in time format as described in the description.

--soa_retry

Retry time associated with the SOA record for the DNS root zone. Should

be specified, in time format as described in the description.

--soa_expiry

Expiry time associated with the SOA record for the DNS root zone. Should

be specified, in time format as described in the description.

--soa_negcache

Negative Cache time associated with the SOA record for the DNS root zone.

Should be specified, in time format as described in the description.

--allow_query

ACL for query operation. Takes a comma separated list of ACL elements
in one of the following formats:

IP Address/permission (192.168.0.1/Allow)

ACL-name/permission (internal/Deny)

IP Address/mask/permission (192.168.0.0/24/Allow)

--masters

Comma separated list of IP addresses of authoritative servers acting as
masters for the DNS root zone.

--is_proxy

DNS Proxy root zone flag. It takes '0' or '1'. If it is specified, as '1'
proxy root zone is updated. If it is not specified, or specified, as '0'
root zone is updated.

```
--contact_first_name
First name field of the associated contact information for the
root/proxy                                              root                                              zone.

--contact_middle_name
Middle name field of the associated contact information for the
root/proxy                                              root                                              zone.

--contact_last_name
Last name field of the associated contact information for the
root/proxy                                              root                                              zone.

--contact_email
Email ID field of the associated contact information for the root/proxy
root                                              zone.
```

EXAMPLE USAGE:

```
twc editdnsrootzone --org=TCPWave --default_ttl=84600/S --dnssec=1 --nsec_opt=NSEC3 --
soa_email=john.smith@tcpwave.com      --soa_refresh=21600/S      --soa_retry=3600/S      --
soa_expiry=604800/S      --soa_negcache=86400/S      --masters=192.168.1.10,192.168.1.11      --
allow_query="192.168.0.1/Allow,192.168.1.0/24/Deny,internal/Deny" --desc="TCPWave root zone" -
-is_proxy=0
```

```
twc editdnsrootzone --org=TCPWave --default_ttl=84600/S --soa_email=john.smith@tcpwave.com --
soa_refresh=21600/S --soa_retry=3600/S --soa_expiry=604800/S --soa_negcache=86400/S      --
masters=192.168.1.10,192.168.1.11      --
allow_query="192.168.0.1/Allow,192.168.1.0/24/Deny,internal/Deny" --desc="TCPWave proxy root
zone" --is_proxy=1
```

editdnsview

NAME:

editdnsview

DESCRIPTION:

Updates a DNS view in the TCPWave IPAM.

ARGUMENTS:

--org

Organization name from which the DNS view belongs. This argument is mandatory if the user is FADM.

--name

Name of the DNS view. [mandatory]

--geoip_type

Takes the input as '0' or '1'.

--match_clients

Comma separated list of match clients in one of the following formats:

IPAddress/permission (192.168.0.1/Allow)

ACL-name/permission (internal/Deny)

Takes the input as comma separated list of match clients in the below format when the geoip_type=1.

country-code/permission (AD/Allow)

--match_destinations

Comma separated list of destinations in one of the following formats:

IPAddress/permission (192.168.0.1/Allow)

ACL-name/permission (internal/Deny)

EXAMPLE USAGE:

```
twc editdnsview --org=TCPWave --name=view1 --match_destinations="173.0.2.5/Allow,none/Deny"
```

```
twc editdnsview --org=TCPWave --name=view1 --match_clients="AD/Allow,IN/Deny,AE/Deny" --geoip_type=1
```

```
twc editdnsview --org=TCPWave --name=view1 --match_clients="173.0.2.5/Allow,none/Deny"
```

editdnszone**NAME****editdnszone**

DESCRIPTION

Updates a DNS Zone in the TCPWave IPAM.

ARGUMENTS

--name

Name of the DNS Zone [mandatory]

--new_name

New name for the DNS Zone if the name is to be updated.

--org

Organization name associated with the zone. [mandatory]

--zone_tmpl

Zone template name associated with the DNS zone.

--dnssec

1 indicates that DNSSEC should be enabled for the zone. 0 indicates that DNSSEC is not enabled.

--nsec_opt

NSEC option for the zone. Takes 'NSEC' or 'NSEC3' as values.

--cascade

Takes 0 or 1. 0 indicates that any name change is not propagated to the subzones. 1 indicates that any name change should be propagated to the subzones. Should be specified when --new_name argument is specified.

--ad_upd

'1' indicates Active Directory updates are enabled for this zone. '0' indicates Active Directory updates are disabled for this zone. If this argument is not specified the value is defaulted to '0'.

--dc_ip

comma separated values of IPs of domain controllers applicable for this zone. This argument should be specified if ad_upd argument is specified as '1'.

--ad_sec

'1' indicates Active Directory secure updates are enabled for this zone. '0' indicates Active Directory secure updates are disabled for this zone. If this argument is not specified the value is defaulted to '0'.

--ad_forest

Indicates active directory forest. It takes either 'parent' or 'child'. 'parent' indicates active directory zone is a standalone parent forest zone. 'child' indicates active directory zone is a child forest zone.

--ms_ad_integrate

Microsoft AD integration applicable only when the selected DNS zone template has Microsoft DNS appliance as master. It accepts '1' or '0' as input. '1' indicates zone on the Microsoft DNS Appliance will be changed to AD integrated zone. '0' indicates zone on the Microsoft DNS appliance will be changed to Standard Zone.

--parent_forest

Indicates active directory parent forest for the child forest.

--monit

'1' indicates monitoring is enabled for this zone. '0' indicates monitoring is disabled for this zone.

--dmz_visible

'1' indicates that the zone is visible to cache server rooted at a public internet root server. '0' indicates that the zone is visible. If this argument is not specified the value is defaulted to '0'.

--acl

Comma separated list of ACL names. This argument should be specified if ad_upd argument is specified as '1' and ad_sec is '0'.

--is_restricted

Restricted zone flag. It takes '1' or '0'. '1' indicates that the zone is updated as a restricted zone. '0' indicates zone is updated as a non-restricted zone.

--views

Comma separated list of DNS view names to be associated with this zone. This argument takes 'None' to dissociate the views from specified zone.

--ext_attr

Comma separated list of extension attributes with their values in the format : extension_attribute_name/extension_attribute_value. Use the following command to see all the extension attributes applied to zone : 'twc listext --entity=zone --d=,'

--contact_fname

First name field of the associated contact information for the DNS zone.

--contact_mname

Middle name field of the associated contact information for the DNS zone.

--contact_lname

Last name field of the associated contact information for the DNS zone.

--contact_email

Email ID field of the associated contact information for the DNS zone.

--custom_allow_ns

Custom allow NS should be specified as FQDN. It accepts the server name and IP address by separating them with comma. It accepts multiple values by separating them with pipe symbol.

Example: test1.com.,10.1.10.1|test2.com.,192.168.0.0

--tsig_key_names

It accepts the multiple TSIG key values by separating with comma. Custom allow NS cannot be null to specify TSIG key names.

Example: key1,key2

--desc

Description for the DNS Zone.

EXAMPLE

```
tvc editdnszone --name="tcpwave.com" --org=TCPWave --zone_tmpl="base-zone-template" --dnssec=1 --nsec_opt=NSEC3 --ad_upd=1 --dc_ip="10.1.10.172,10.0.10.50" --acl=none,any --ad_sec=0 --monit=0 --views=view1,view2 --desc="TCPWave Zone"
```

```
tvc editdnszone --name="tcpwave.com" --new_name=dev.tcpwave.com --org=TCPWave --cascade=1 --ad_upd=0 --is_restricted=0 --desc="TCPWave Zone"
```

```
tvc editdnszone --name="tcpwave.com" --org=TCPWave --ext_attr=ext_attr_1/value_1,ext_attr_2/value_2 --desc="TCPWave Zone"
```

```
tvc editdnszone --name="dev.tcpwave.com" --org=TCPWave --zone_tmpl="base-zone-template" --dnssec=1 --nsec_opt=NSEC3 --ad_upd=1 --ad_forest=child --parent_forest=tcpwave.com --dc_ip="10.1.20.175,10.3.10.52" --ad_sec=1 --desc="TCPWave Zone"
```

```
tvc editdnszone --name="tcpwave.com" --org=TCPWave --contact_fname=John --contact_lname=Smith --contact_email=john.smith@tcpwave.com
```

```
tvc editdnszone --name="tcpwave.com" --org=TCPWave --zone_tmpl="base-zone-template" --dnssec=1 --nsec_opt=NSEC3 --custom_allow_ns="TCPWaveNs.,10.1.10.1|NS1.,192.168.0.0" --tsig_key_names=key1,key2 --desc="TCPWave Zone"
```

editdnszonetmpl

NAME:

editdnszonetmpl

DESCRIPTION:

Update a DNS zone template in TCPWave IPAM. Time formats are specified as integer/time_unit. time_unit can be one of the following values: S,MIN,H,D,W,MON,Y representing seconds, minutes, hours, days, weeks, months, years
Example: 84600/S or 30/D

ARGUMENTS:

--name

Name of the DNS zone template [mandatory]

--new_name

New name of the DNS zone template if the name is to be updated.

--org

Organization name associated with the zone template.

--default_ttl

Default TTL for the zone. Should be specified in time format as described in the description.

--soa_email

Email id associated with the SOA record for the zone.

--soa_refresh

Refresh time associated with the SOA record for the zone. Should be specified in time format as described in the description.

--soa_retry

Retry time associated with the SOA record for the zone. Should be specified in time format as described in the description.

--soa_expiry

Expiry time associated with the SOA record for the zone. Should be specified in time format as described in the description.

--soa_negcache

Negative Cache time associated with the SOA record for the zone. Should be specified in time format as described in the description.

--allow_query

ACL for query operation. Takes a comma separated list of ACL elements in one of the following formats:

IP Address/permission (192.168.0.1/Allow)

ACL-name/permission (internal/Deny)

IP Address/mask/permission (192.168.0.0/24/Allow)

--allow_update

ACL for custom allow update operation. Takes a comma separated list of ACL elements in one of the following formats:

Ex: localhost,localnets

--allow_transfer

ACL for custom allow transfer operation. Takes a comma separated list of ACL elements in one of the following formats:

Ex: none,any

--also_notify

Input for custom also notify. Takes a comma separated list of IPV4 address in one of the following formats:

IP Address/permission (192.168.0.1/Allow)

IP Address/mask/permission (192.168.0.0/24/Allow)

--masters

Comma separated list of IP addresses of authoritative servers acting as masters for the zone.

--slaves

Comma separated list of IP addresses of authoritative servers acting as slaves for the zone.

--ms_dns_master

Comma separated list of IP addresses of Microsoft DNS Master appliances acting as masters for the zone.

--empty_forwarders

Takes '1' or '0'. '1' indicates that zones will be generated with empty forwarders in the zone sections. '0' indicates that no empty forwarders for zones. Default 0.

--desc

Description for the DNS zone template.

EXAMPLE:

```
twc editdnszonetmpl --name="base-zone-template" --org=TCPWave --desc="base zone template" --soa_email=john.smith@tcpwave.com --allow_update=localhost,localnets --allow_transfer=none,any --masters=192.168.1.10,192.168.1.11 --slaves=192.168.1.12,192.168.1.13 --empty_forwarders=1
```

```
twc editdnszonetmpl --name="base-zone-template" --org=TCPWave --soa_email=john.smith@tcpwave.com --masters=192.168.1.10 --slaves=192.168.1.12
```

```
twc editdnszonetmpl --name="base-zone-template" --org=TCPWave --soa_email=john.smith@tcpwave.com --ms_dns_master=192.168.1.12 --empty_forwarders=1
```

editdomain**NAME:**

editdomain

DESCRIPTION:

Updates a DNS domain in the TCPWave IPAM.

A DNS domain can be updated with a new name and description.

ARGUMENTS:**--name**

Name of the domain being updated. [mandatory]

--org

Organization name associated with the domain.

This argument is for users in FADM role to select appropriate organization to which the operation must be applied.

For users not in FADM role, the operation is by default applied to the organization that the user is associated with.

--new_name

New name for the domain if the name of the domain is being updated.

--desc

Description for the domain being updated.

--cascade

Takes 0 or 1. 0 indicates that any name change is not propagated to the subdomains. 1 indicates that any name change should be propagated to the subdomains. Should be specified, when --new_name argument is specified.,

EXAMPLE USAGE:

```
tvc editdomain --name="tcpwave.com" --desc="TCPWave Primary Domain" --  
new_name=tcpwave1.com --cascade=1 --org=TCPWave  
tvc editdomain --name="tcpwave.com" --new_name=tcpwave1.com --cascade=1 --  
org=TCPWave
```

```
tvc editdomain --name="QAZone1.com" --new_name=QAZone.com --desc="QA Domain" --  
org=QAOrg --cascade=0
```

editext**NAME:**

editext

DESCRIPTION:

Updates an extended attribute in the TCPWave IPAM.

ARGUMENTS:**--name**

Name of the extension attribute. [mandatory]

--new_name

New name of the extension attribute.

--type

Data type of the extension attribute. It takes one of 'string', 'date', 'numeric', 'ip' or 'list'.

--list_values

Values of the list type extension attribute. It takes comma separated list of strings. This argument is applicable only when type argument is 'list'.

--entities

Entities for which extension attribute to be applied. It takes comma separated list of entity in format : entity/show_grid_flag. Allowable entities are 'admin', 'network', 'subnet', 'object', 'zone'.

show_grid_flag takes 1 or 0. Examples admin/1, network/0

--mandatory

It takes '1' or '0'. '1' indicates that extension attribute value is mandatory while creating the target entities. '0' indicates that extension attribute value is not mandatory while creating the target entities.

--desc

Description of the extension attribute being updated.

EXAMPLE USAGE:

```
tvc editext --name=RITS_ID --type=numeric --entities=admin/1,network/0 --mandatory=0 --desc="RITS ID administrator"
```

```
tvc editext --name=CHG_TKT --new_name=RITS_ID --mandatory=1
```

```
tvc editext --name=OS_TYPE --type=list --entities=object/1 --list_values=Windows,Linux,Mac --mandatory=0
```

editextvalue**NAME**

editextvalue

DESCRIPTION

Updates the extended attribute value of specified key defined in the TCPWave IPAM. Applicable entities are 'admin', 'network', 'subnet', 'object' and 'zone'. If entity is 'network', 'subnet' or 'object' key must be an IP address. If entity is admin, key is the login-name of admin. If entity is zone, then key is domain name.

ARGUMENTS

--name

Name of the extended attribute. [mandatory]

--entity

Entity of the extended attribute. [mandatory]

--key

Key of the extended attribute. [mandatory]

--value

Value of the extended attribute for specified key. [mandatory]

--org

Name of the organization. [mandatory]

EXAMPLE

```
twc editextvalue --name="CHG_TKT" --entity=zone --key=tcpwave.com --value=tkt --
org=TCPWave
```


editglobalopts**NAME:**

editglobalopts

DESCRIPTION:

Updates a Global Option in the TCPWave IPAM.

ARGUMENTS:

- option
 Option to edit [mandatory]
- value
 Value of the associated option to be set. Updatable field.

EXAMPLE USAGE:

```
twc editglobalopts --option=ENABLE_DEBUG_LOGGING --value=Yes
```

editipv6network**NAME:**

editipv6network

DESCRIPTION:

Updates an IPv6 network in the TCPWave IPAM.

ARGUMENTS:

- network
 IPv6 address of the target network. [mandatory]
- org
 Organization name under which the network is being created. This argument is mandatory if the user is FADM.
- name
 New name of the network to be updated.
- desc
 Description for the network to be updated.
- dnssec
 1 indicates that dnssec should be enabled for the reverse zone. 0 indicates that dnssec is not enabled.
- nsec_opt
 NSEC option for the reverse zone. Takes 'NSEC' or 'NSEC3' as values.
- zone_tmpl
 Zone template name to be associated with the reverse zone.

--dmz_visible

'1' indicates that the zone is visible to the cache server rooted at a public internet root server. '0' indicates that the zone is not visible. If this argument is not specified, the value is defaulted to '0'.

EXAMPLE USAGE:

```
tvc editipv6network --network=2000:: --org=TCPwave --desc="TCPWave Network"
```

```
tvc editipv6network --network=2000:: --name=NewName --org=TCPwave --desc="TCPWave Network" --dnssec=1 --nsec_opt=NSEC --zone_tmpl="TestZoneTemplate" --dmz_visible=1
```

editipv6object**NAME:**

editipv6object

DESCRIPTION:

Updates an IPv6 object in the TCPWave IPAM.

ARGUMENTS:**--object**

IPv6 address of the target object. [mandatory]

--org

Organization name in which the operation must be performed. This argument is mandatory if the user is FADM. [mandatory]

--name

Name of the target object.

--alloc_type

Address allocation type (Static | Manual | Dynamic | Auto).

--class_code

Class code of the target object.

--mac

MAC address of the target object.

--desc

Description for the target object.

--ttl

Time-to-Live in seconds for the target object.

--opt_tmpl

Option Template Name associated with the target object.

--dhcp_appliance

DHCP Server Name associated with the specified, Option Template.

--ns_aaaa

Takes 0 | 1. 1 indicates that the corresponding AAAA resource record must be updated in the name service zone file. 0 indicates that the corresponding A resource record will not be updated in the name service zone file.

--ns_ptr

Takes 0 | 1. 1 indicates that the corresponding PTR resource record must be updated in the name service zone file. 0 indicates that the corresponding PTR resource record will not be updated in the name service zone file.

--ddns_aaaa

Takes 0 | 1. 1 indicates that dynamic DNS updates are allowed to the corresponding AAAA resource record in the name service zone file. 0 indicates that dynamic DNS updates are not allowed to the corresponding A resource record in the name service zone file.

--ddns_ptr

Takes 0 | 1. 1 indicates that dynamic DNS updates are allowed to the corresponding PTR resource record in the name service zone file. 0 indicates that dynamic DNS updates are not allowed to the corresponding PTR resource record in the name service zone file.

--ddns_cname

Takes 0 | 1. 1 indicates that dynamic DNS updates are allowed to the corresponding CNAME resource record in the name service zone file. 0 indicates that dynamic DNS updates are not allowed to the corresponding CNAME resource record in the name service zone file.

--ddns_mx

Takes 0 | 1. 1 indicates that dynamic DNS updates are allowed to the corresponding MX resource record in the name service zone file. 0 indicates that dynamic DNS updates are not allowed to the corresponding MX resource record in the name service zone file.

--contact_first_name

First Name field of the associated contact information for the object.

--contact_last_name

Last Name field of the associated contact information for the object.

--contact_email

Email Id field of the associated contact information for the object.

--street1

Street1 field of the ipv6 object address [mandatory]

--street2

Street2 field of the ipv6 object address.
`--city` City field of the ipv6 object address [mandatory]
`--state` State field of the ipv6 object address [mandatory]
`--country` Country field of the ipv6 object address [mandatory]
`--zip` Zip code field of the ipv6 object address [mandatory]

EXAMPLE USAGE:

```
twc editipv6object --object='2001::a:0:0:0' --org=TCPWave --desc='testing' --name=Laptop-001 -  
-class_code=Laptop --mac=11:11:22:33:aa:bb --ttl=100 --ns_aaaa=1 --ns_ptr=1 --ddns_aaaa=1 --  
ddns_ptr=1 --ddns_cname=1 --ddns_mx=1 --room=231 --end_of_life=2000 --floor=10
```

```
twc editipv6object --object='2001::a:0:0:0' --org=TCPWave --alloc_type=Dynamic --  
opt_tmpl=Generic --dhcp_appliance='DHCP-server'
```

```
twc editipv6object --object='2001::a:0:0:0' --org=TCPWave --contact_first_name=John10 --  
contact_last_name=Smith10 --contact_email=john.smith10@tcpwave.com
```

editipv6subnet**NAME:**

`editipv6subnet`

DESCRIPTION:

Updates an IPv6 subnet in the TCPWave IPAM.

ARGUMENTS:

`--subnet` IPv6 address of the target subnet. [mandatory]

`--name` Name of the subnet. [Updatable field]

`--org` Name of the organization to which subnet belongs. This argument is
mandatory if user is 'FADM'.

`--subnet_groupname` Name of the associated subnet group. If this argument is specified, as
'None' this will dissociate subnet group from subnet. [Updatable field]

`--domain` Domain to be associated with this IPv6 subnet. [Updatable field]

`--router_addr` IPv6 address of the router to be associated with the subnet.
[Updatable field]

--dhcp_tmpl

Template name specifying the IPv6 DHCP options for the subnet. If this argument is specified, as 'None' this will dissociate DHCP option Template from subnet. [Updatable field]

--dhcp_appliance

Primary DHCP server address for the subnet. This argument is mandatory if dhcp_tmpl argument is specified. If this argument is specified, as 'None' this will dissociate DHCP server from subnet. [Updatable field]

--desc

Description text for the IPv6 subnet. [Updatable field]

--street1

Street1 part of the location information. Should be specified, along with other mandatory location fields if location is to be updated.

--street2

Street2 part of the location information. Should be specified, along with other mandatory location fields if location is to be updated.

--city

City part of the location information. Should be specified, along with other mandatory location fields if location is to be updated.

--state

State part of the location information. Should be specified, along with other mandatory location fields if location is to be updated.

--country

Country part of the location information. Should be specified, along with other mandatory location fields if location is to be updated.

--zip

Zip code part of the location information. Should be specified, along with other mandatory location fields if location is to be updated.

EXAMPLE USAGE:

```
twc editipv6subnet --subnet=2001::f000:0:0:0 --org=TCPWave --domain=123.com --
router_addr=2001:0:0:f000::2 --desc="Description" --dhcp_tmpl=DHCP-Option-Template --
dhcp_appliance=DHCP-Server-006
twc editipv6subnet --subnet=2001::f000:0:0:0 --org=TCPWave --street1="600 ALEXANDER
ROAD" --city="PRINCETON" --state=NJ --country=USA --zip=08540 --desc="Location updated"
```

editipv6dnsreversezone**NAME**

editipv6dnsreversezone

DESCRIPTION

Updates a DNS reverse zone in the TCPWave IPAM. IP address of the reverse zone, mask length (an integer between 4 and 128) and organization name are mandatory fields to be given as input to update a DNS reverse zone.

ARGUMENTS

--ip

IPv6 address of a DNS reverse zone. [mandatory]

--name

Name of the IPv6 DNS reverse zone. [mandatory]

--mask

Mask length (an integer between 4 and 128) [mandatory]

--org

Organization name associated with the reverse zone. [mandatory]

--zone_tmpl

New zone template name to be associated with the DNS reverse zone.

--dnssec

'1' indicates that DNSSEC should be enabled for the reverse zone. '0' indicates that DNSSEC is not enabled.

--nsec_opt

NSEC option for the reverse zone. Takes 'NSEC' or 'NSEC3' as values.

--monit

'1' indicates monitoring is enabled for this reverse zone. '0' indicates monitoring is disabled for this reverse zone.

--dmz_visible

'1' indicates that the reverse zone is visible to cache server rooted at a public internet root server. '0' indicates that the reverse zone is visible. If this argument is not specified the value is defaulted to '0'.

--ms_ad_integrate

Microsoft AD integration applicable only when the selected DNS zone template has Microsoft DNS appliance as master. It accepts '1' or '0' as input. '1' indicates zone on the Microsoft

DNS Appliance will be changed to AD integrated zone. '0' indicates zone on the Microsoft DNS appliance will be changed to Standard Zone.'

--views

Comma separated list of DNS view names to be associated with the DNS reverse zone. This argument takes 'None' to dissociate the views from specified zone.

--contact_fname

First name of the new contact to be associated with the DNS reverse zone.

--contact_mname

Middle name of the new contact to be associated with the DNS reverse zone.

--contact_lname

Last name of the new contact to be associated with the DNS reverse zone.

--contact_email

Email Id of the new contact to be associated with the DNS reverse zone.

--custom_allow_ns

Custom allow NS should be specified as FQDN. It accepts the server name and IP address by separating them with comma. It accepts multiple values by separating them with pipe symbol.

Example: test1.com.,3111::3|test2.com.,3000::3

--is_tsig

1 indicates TSIG is enabled. 0 indicates TSIG is disabled.

--desc

New description for the DNS reverse zone.

EXAMPLE

```
twc editipv6dnsreversezone --ip=5000:: --name=0.0.0.0.5.ip6.arpa --mask=24 --org=TCPWave -  
-desc="TCPWave Reverse Zone" --zone_tmpl="base-zone-template" --monit=0
```

```
twc editipv6dnsreversezone --ip=5000:: --name=0.0.0.0.5.ip6.arpa --mask=24 --org=TCPWave -  
-desc="TCPWave Reverse Zone" --dnssec=1
```

```
twc editipv6dnsreversezone --ip=5000:: --name=0.0.0.0.5.ip6.arpa --mask=24 --org=TCPWave -  
-desc="TCPWave Reverse Zone" --zone_tmpl="base-zone-template" --dnssec=1 --nsec_opt=NSEC3
```

```
twc editipv6dnsreversezone --ip=5000:: --name=0.0.0.0.5.ip6.arpa --mask=24 --org=TCPWave -  
-contact_fname=John --contact_lname=Smith --contact_email=john.smith@tcpwave.com
```

```
twc editipv6dnsreversezone --ip=5000:: --name=0.0.0.0.5.ip6.arpa --mask=24 --org=TCPWave -  
-views=view1,view2 --custom_allow_ns="TCPWaveNs.,10.1.10.1|NS1.,192.168.0.0" --is_tsig=0
```

editipv6subnetgroup

NAME

editipv6subnetgroup

DESCRIPTION

Updates an IPv6 subnet group in the TCPWave IPAM.

ARGUMENTS

--name

Name of the IPv6 subnet group being updated. [mandatory]

--org

Organization name of the IPv6 subnet group. [mandatory]

--new_name

New name for the IPv6 subnet group if the name needs to be updated.

--desc

Description of the IPv6 subnet group.

EXAMPLE

```
twc editipv6subnetgroup --name=IT_SG --new_name=Sales_SG --desc="IT subnet group" --  
org=TCPWave
```

editipv6block

NAME

editipv6block

DESCRIPTION

Updates an IPv6 address block in the IPv6 address pool defined in the TCPWave IPAM.

ARGUMENTS

--name

New name of the IPv6 address block.

--ip

Address of the target IPv6 address block. [mandatory]

--org

Organization name to which the IPv6 address block exist. [mandatory]

--zone_tmpl

Zone template name to be associated with the block reverse zone.

--dmz_visible

DMZ visibility flag. '1' indicates that the reverse zone is visible to the cache server rooted at a public internet root server. '0' indicates that the zone is not visible. If this argument is not specified the value is defaulted to '0'.

--desc

Description of the IPv6 address block.

--dnssec

DNSSEC flag. 1 indicates that DNSSEC will be enabled for the reverse zone. 0 indicates that DNSSEC is not enabled.

--nsec_opt

NSEC option for the reverse zone. Takes 'NSEC' or 'NSEC3' as values.

EXAMPLE

```
twc editipv6block --ip=2000:: --org=TCPwave --name=new_name --desc="TCPWave IPv6 address block"
```

```
twc editipv6block --ip=2000:: --name=NewName --org=TCPwave --desc="TCPWave IPv6 address block" --dnssec=1 --nsec_opt=NSEC --zone_tmpl="TestZoneTemplate" --dmz_visible=1
```

editipv6pool**NAME**

editipv6pool

DESCRIPTION

Updates an IPv6 address pool in the TCPWave IPAM.

ARGUMENTS**--ip**

Address of the target IPv6 address pool. [mandatory]

--org

Organization name of the existing IPv6 address pool.

--name

New name of the IPv6 address pool to be updated.

--desc

Description for the IPv6 address pool to be updated.

--dnssec

1 indicates that dnssec should be enabled for the reverse zone. 0 indicates that dnssec is not enabled.

--nsec_opt

NSEC option for the reverse zone. Takes 'NSEC' or 'NSEC3' as values.

--zone_tmpl

Zone template name to be associated with the reverse zone.

--dmz_visible

'1' indicates that the zone is visible to the cache server rooted at a public internet root server. '0' indicates that the zone is not visible. If this argument is not specified the value is defaulted to '0'.

EXAMPLE

```
twc editipv6pool --ip=2000:: --org=TCPwave --desc="TCPWave IPv6 address pool"
```

```
twc editipv6pool --ip=2000:: --name=NewName --org=TCPwave --desc="TCPWave IPv6 address pool" --dnssec=1 --nsec_opt=NSEC --zone_tmpl="TestZoneTemplate" --dmz_visible=1
```

editlocation**NAME:**

editlocation

DESCRIPTION:

Update a location for a given organization in the TCPWave IPAM.
The target organization must be specified, using --org parameter
Target location must specify in terms of the mandatory address fields.

ARGUMENTS:

- street1
Street1 field of the location address [mandatory]
- street2
Street2 field of the location address.
- city
City field of the location address [mandatory]
- state
State field of the location address [mandatory]
- country
Country field of the location address [mandatory]
- zip
Zip code field of the location address [mandatory]
- org
Organization name for which the location is being updated [mandatory]
- new_street1
New Street1 field of the location address.
- new_street2
New Street2 field of the location address.
- new_city
New City field of the location address.
- new_state
New State field of the location address.
- new_country
New Country field of the location address.
- new_zip
New Zip code filed of the location address.

EXAMPLE USAGE:

```
twc editlocation --org=TCPWave --street1="600 ALEXANDER ROAD" --city="PRINCETON" --  
state=NJ --country=USA --zip=08540  
--new_street1="10 Main street" --new_street2="lane park"" --new_city="Atlantic city" --  
new_state=princeton --new_country=NJ --new_zip=08401
```

editlogchannel**NAME:**

editlogchannel - Updates a DNS log channel in the TCPWave IPAM.

DESCRIPTION:

Updates a DNS log channel in the TCPWave IPAM.

ARGUMENTS:

--name

Name of the DNS log channel [mandatory]

--new_name

New name of the DNS log channel.

--type

DNS log channel type. Takes 'FILE', 'SYSLOG', 'STDERR' or 'NULL'.

--print_time

Takes '1' or '0'. '1' enables print time. '0' disables print time.

--print_sev

Takes '1' or '0'. '1' enables print severity. '0' disables print severity.

--print_cat

Takes '1' or '0'. '1' enables print log category. '0' disables print log category.

--severity

Takes 'dynamic','debug','info','notice','warning','error' or 'critical'.

--file_path

Full path the file to which log is written to.

--versions

Number of versions of the log file to be preserved before purging older versions.

--size

Size of the log file before a new version of the file is created.

--facility

Syslog facility name. Should be specified, for channel type 'SYSLOG'.

--debug_level

Debug level value. Should be specified, if severity is 'debug'.

EXAMPLE USAGE:

```
tvc editlogchannel --name=testlogchannel --type=FILE --severity=dynamic --
file_path=/var/log/test.log --versions=10 --size=1024 --print_time=1 --print_sev=1 --print_cat=1

tvc editlogchannel --name=testlogchannel --new_name=defaultChannel
```

editmirroredzone

NAME:

editmirroredzone

DESCRIPTION:

Updates a DNS managed mirrored zone in the TCPWave IPAM.
Name, description for a given mirrored zone can be updated and one of them
must be specified.,

ARGUMENTS:

--org	Organization name under which the mirrored zones is being updated. This argument is mandatory if the user is FADM.
--name	Name of the existing mirrored zone. [mandatory]
--zone_name	Name of the DNS zone associated with the mirrored zone. [mandatory]
--new_name	New name of the mirrored zone.
--desc	Description for the mirrored zone.

EXAMPLE USAGE:

```
tvc editmirroredzone --org=TCPWave --name=tcpwave.com --zone_name=tcpwave1.com --
new_name=tcpwave2.com --desc="Mirrored zone of tcpwave.com"
```

editmicrosoftdhcpserver

NAME:

editmicrosoftdhcpserver

DESCRIPTION:

Updates the existing Microsoft DHCP appliance in the TCPWave IPAM.

ARGUMENTS:

--addr	IP address of the appliance. [mandatory]
--------	--

```
--org          Name of the organization. [mandatory]
--new_addr     New IP address of the appliance.
--use_https    Takes '0' or '1' as input.
--user_name    User name for the Microsoft DHCP appliance.
--mac_exclusion_addr  Comma separated mac exclusion addresses without any spaces.
--desc         Description for the Microsoft DHCP appliance.
```

EXAMPLE USAGE:

```
twc editmicrosoftdhcpserver --addr=10.0.0.10 --new_addr=10.0.0.12 --org=TCPWave --
user_name=profile --mac_exclusion_addr=AA:BB:CC:DD:EE:F5,AA:BB:CC:DD:EE:F9
```

```
twc editmicrosoftdhcpserver --addr=10.0.0.10 --org=TCPWave --use_https=1 --user_name=user --
-desc="First Microsoft Appliance"
```

editmicrosoftdnsserver**NAME:**

editmicrosoftdnsserver

DESCRIPTION:

Updates the existing Microsoft DNS appliance in the TCPWave IPAM.

ARGUMENTS:

```
--addr          IP address of the appliance. [mandatory]
--org           Name of the organization. [mandatory]
--new_addr      New IP address of the appliance.
--use_https     Takes '0' or '1' as input.
--user_name     Username for the Microsoft appliance.
--desc
```

Description for the Microsoft DNS appliance.

EXAMPLE USAGE:

```
tvc editmicrosoftdnsserver --addr=10.0.0.10 --new_addr=10.0.0.12 --org=TCPWave --  
user_name=profile
```

```
tvc editmicrosoftdnsserver --addr=10.0.0.10 --org=TCPWave --use_https=1 --user_name=user --  
desc="First Microsoft Appliance"
```

editmicrosoftadsite**NAME**

editmicrosoftadsite

DESCRIPTION

Updates Microsoft AD site in the TCPWave IPAM. You can enter site name up to 32 characters (alpha-numeric and hyphen). The system restricts you to enter a space between the words. At least one alphabet needs to be part of the name. The name cannot contain all the numerals.

ARGUMENTS**--name**

Name of the Microsoft AD site. [mandatory]

--new_name

New name of the Microsoft AD site.

--subnet_group

Name of the subnet group, it accepts the comma separated subnet groups.

--ip

IP address of the appliance. [mandatory]

--org

Name of the organization. [mandatory]

--desc

Description for the Microsoft AD site.

EXAMPLE

```
twc editmicrosoftadsite --name=UKSite --ip=10.0.0.10 --org=TCPWave --desc="Updated UK Site"
twc editmicrosoftadsite --name=UKSite --new_name=NJSite --ip=10.0.0.10 --org=TCPWave
twc editmicrosoftadsite --name=NJSite --subnet_group=NJGroup --ip=10.0.0.10 --org=TCPWave
twc editmicrosoftadsite --name=NJSite --subnet_group=TestGroup --ip=10.0.0.10 --org=TCPWave
--desc="Updated NJ Site"
```

editnetwork**NAME:**

editnetwork

DESCRIPTION:

Updates a network in the TCPWave IPAM.

ARGUMENTS:**--network**

Address of the network. [mandatory]

--name

Name of the network.

--desc

Description of the network.

--dnssec

Enable DNSSEC flag. Takes '1' or '0'. '1' indicates that DNSSEC should be enabled for the reverse zone. '0' indicates that DNSSEC is not enabled.

--nsec_opt

NSEC option for the reverse zone. Takes 'NSEC' or 'NSEC3' as values.

--zone_tmpl

Name of the zone template to be associated with the reverse zone.

--dmz_visible

'1' indicates that the zone is visible to the cache server rooted at a public internet root server. '0' indicates that the zone is not visible. If this argument is not specified, the value is defaulted to '0'.

--org

Name of the organization to which network belongs. This argument is mandatory if user is 'FADM'.

--ext_attr

Comma separated list of extension attributes with their values in the format: extension_attribute_name/extension_attribute_value. Use the following command to see all the extension attributes applied to network: 'tvc listext --entity=network --d=.'

--monitoring

Enable monitoring services flag. Takes '1' or '0'. '1' indicates that monitoring service should be enabled for the network. '0' indicates that monitoring is not enabled.

--enable_discovery

Enable discovery flag. Takes '1' or '0'. '1' indicates that discovery

should be enabled for the network. '0' indicates that discovery is not enabled.

--discovery_tmpl

Name of the discovery template. It is mandatory to specify the discovery template name if discovery is enabled.

--contact_first_name

First name field of the associated contact information for the network.

--contact_middle_name

Middle name field of the associated contact information for the network.

--contact_last_name

Last name field of the associated contact information for the network.

--contact_email

Email ID field of the associated contact information for the network.

EXAMPLE USAGE:

```
twc editnetwork --network=50.0.0.0 --org=TCPWave --name="TCPWave Network" --desc="TCPWave Network"
```

```
twc editnetwork --network=50.0.0.0 --org=TCPWave --name="TCPWave Network" --desc="TCPWave Network" --dnssec=1 --nsec_opt=NSEC --zone_tmpl="TestZoneTemplate" --dmz_visible=1
```

```
twc editnetwork --network=50.0.0.0 --org=TCPWave --name="TCPWave Network" --desc="TCPWave Network" --ext_attr=ext_attr_1/value_1,ext_attr_2/value_2
```

```
twc editnetwork --network=50.0.0.0 --org=TCPWave --name="TCPWave Network" --monitoring=1 --enable_discovery=1 --discovery_tmpl="Discovery-Template" --desc="TCPWave Network"
```

```
twc editnetwork --network=80.0.0.0 --org=TCPwave --contact_first_name=John --contact_last_name=Smith --contact_email=john.smith@tcpwave.com --desc="TCPWave Network"
```

editobject

NAME

editobject

DESCRIPTION

The twc editobject CLI command is used to update object in TCPWave IP Address Management system. The syntax of this command is shown below. The user invoking this command is expected to have authentication permission and should be authorized to perform edit object. The TCPWave IPAM audits this action. Successful completion of this command exits with a status code 0. Target object should be specified in terms of --object parameter

ARGUMENTS

--object
IP Address of the target object. [mandatory]

--name
Name of the target object.

--alloc_type
Address allocation type. (Static | Manual | Dynamic | Reserved | Auto)

--class_code
Class code of the target object.

--domain
Domain name associated with the target object.

--org
Name of the organization to which specified object belongs. [mandatory]

--mac
MAC address of the target object.

--desc
Description for the target object.

--ttl
Time-to-Live in seconds for the target object.

--view
Name of the DNS view. It accepts the comma separated DNS view names.

--opt_tmpl
Option Template Name associated with the target object.

--ns_a
Takes 0|1. 1 indicates that the corresponding A resource record has to be updated in the name service zone file. 0 indicates that the corresponding A resource record will not be updated in the name service zone file.

--ns_ptr
Takes 0|1. 1 indicates that the corresponding PTR resource record has to be updated in the name service zone file. 0 indicates that the corresponding PTR resource record will not be updated in the name service zone file.

--ddns_a
Takes 0|1. 1 indicates that dynamic DNS updates are allowed to the corresponding A resource record in the name service zone file. 0 indicates that dynamic DNS updates are not allowed to the corresponding A resource record in the name service zone file.

--ddns_ptr

Takes 0|1. 1 indicates that dynamic DNS updates are allowed to the corresponding PTR resource record in the name service zone file.

0 indicates that dynamic DNS updates are not allowed to the corresponding PTR resource record in the name service zone file.

--ddns cname

Takes 0|1. 1 indicates that dynamic DNS updates are allowed to the corresponding CNAME resource record in the name service zone file. 0 indicates that dynamic DNS updates are not allowed to the corresponding CNAME resource record in the name service zone file.

--ddns_mx

Takes 0|1. 1 indicates that dynamic DNS updates are allowed to the corresponding MX resource record in the name service zone file. 0 indicates that dynamic DNS updates are not allowed to the corresponding MX resource record in the name service zone file.

--contact_fname

First Name field of the associated contact information for the object.

--contact_mname

Middle Name field of the associated contact information for the object.

--contact_lname

Last Name field of the associated contact information for the object.

--contact_email

Email Id field of the associated contact information for the object.

--street1

Street1 part of the location information.

--street2

Street2 part of the location information.

--city

City part of the location information.

--state

State part of the location information.

--zip

Zip code part of the location information.

--country

Country part of the location information.

--room

Room information for object location attribute.

--floor

Floor information for object location attribute.

--terminal_server_kvm

Terminal server kvm for the object.

--switch

Switch for the object.

--port

Port for the object.

--duplex

Duplex for the object. It accepts the input as any digit along with Mbps or Gbps, Ex: 100Mbps.

--ext_attr

Comma separated list of extension attributes with their values in the format : extension_attribute_name/extension_attribute_value. Use the following command to see all the extension attributes applied to object : 'twc listext --entity=object --d=.'

--vmware_attributes

VMWare Attributes are mandatory for VMWare ESXi and VMWare vCenter object types. It accepts port number, user name and password by separating with '|' symbol.

Example: 7443|tcpwave|abc12345

--validate

To validate the VMWare attribute. It accepts '1' or '0'. '1' indicates to validate the VMWare Attributes. '0' indicates don't validate the VMWare Attributes.

EXAMPLE

```
twc editobject --object=10.20.0.5 --org=Internal --name="PC-2733663" --class_code=PC --domain=tcpwave.com --mac=01:23:45:67:89:ab --desc="General use PC" --ttl=300 --ns_a=1 --ns_ptr=1 --ddns_a=1 --ddns_ptr=1 --ddns_cname=1 --ddns_mx=1 --alloc_type=Dynamic --opt_tmpl=Generic
```

```
twc editobject --object=10.20.0.5 --org=Internal --name="PC-2733663" --class_code=PC --domain=tcpwave.com --mac=01:23:45:67:89:ab --desc="General use PC" --ext_attr=ext_attr_1/value_1,ext_attr_2/value_2
```

```
twc editobject --object=10.20.0.5 --org=Internal --name="9G1880000Phone" --class_code="3G Phone" --domain=angels.com --alloc_type=Reserved --expiry_date="06/1/2018" --mac=29:23:46:97:89:7d --ttl=300 --desc="Reserved IP Updated"
```

```
twc editobject --object=10.20.0.5 --org=Internal --name="Server-2733663" --
class_code="TCPWave Remote" --domain=tcpwave.com --street1="600 ALEXANDER ROAD" --
city="PRINCETON" --state=NJ --country=USA --zip=08540 --contact_fname=John --
contact_lname=Smith --contact_email=john.smith@tcpwave.com

twc editobject --object=10.20.0.5 --org=Internal --room="Room-1" --floor=second --switch=test --
port=8808 --duplex=10mbps
```

editobjecttype**NAME:**

editobjecttype

DESCRIPTION:

Updates an object type in the TCPWave IPAM.

An object type can be updated with a new prefix, suffix and description.

ARGUMENTS:

--code	Unique code of the object type. [mandatory]
--prefix	Prefix of the object type. Example: 3G
--suffix	Suffix of the object type. Example: Phone
--prefix_zeros	Set zeros as prefix to initial sequence number. It takes as 'yes' or 'No'.
--desc	description of the object type.

EXAMPLE USAGE:

```
twc editobjecttype --code="Server" --prefix="server" --suffix="type-object" --prefix_zeros=yes --
desc="A Generic Web Server"
twc editobjecttype --code="Web Server" --prefix="web-server" --suffix="type-object" --
prefix_zeros=no --desc="A Generic Web Server"
```

editorg**NAME:**

editorg

DESCRIPTION:

Updates an organization in the TCPWave IPAM.

An organization can be updated with a new name and description. Also, a root zone can be enabled or disabled.

ARGUMENTS:

--name
Name of the target organization that is being updated [mandatory]

--new_name
New name for the target organization.

--new_desc
New description for the target organization.

--enable_root_zone
Takes '1' or '0'. Root zone can be created for the organization when the value is '1' and cannot be created for the organization when the value is '0'.

EXAMPLE USAGE:

```
twc editorg --name="TCPWave" --new_name="TCPWave1" --new_desc="TCPWave Organization" --enable_root_zone=1
```

editrpztmp**NAME:**

editrpztmp

DESCRIPTION:

Updates a DNS Response policy zone (RPZ) template in the TCPWave IPAM.

ARGUMENTS:

--name
Name of the DNS Response policy zone (RPZ) template to be updated in the TCPWave IPAM. [mandatory]

--new_name
New name of the DNS Response policy zone(RPZ) template.

--zone_name
Zone name used in Response policy zone(RPZ) template.

--policy_file
Policy rules file for the RPZ template.

--certificate_file
Path of the certificate file for the Response policy zone(RPZ) data feed from third party.

--url
Response policy zone(RPZ) data feed URL of third party.

--auto_xfr

Flag to indicate whether the zone data feed will be done from an external DNS server or local data. Takes '0' or '1'. Default value is '0'. If this argument is specified, as '1' the zone data will be feed from an external DNS Server using zone transfer.

--master_server

IP address of the server used for RPZ data feed. This argument is mandatory if --auto_xfr is specified, as '1'.

--comm_key_name

Server zone transfer key name for RPZ feed. This argument is mandatory if --auto_xfr is specified, as '1'.

--comm_key_value

Server zone transfer key value for RPZ feed. This argument is mandatory if --auto_xfr is specified, as '1'.

--desc

Description of the RPZ template to be updated in TCPWave IPAM.

EXAMPLE USAGE:

```
twc editrpztmpl --name="RPZ-Template" --new_name="New-RPZ-Template"
```

```
twc editrpztmpl --name="RPZ-Template" --zone_name=newzone.com --  
policy_file=RpzPolicyFile --auto_xfr=0
```

```
twc editrpztmpl --name="RPZ-Template" --master_server=10.1.10.251
```

```
twc editrpztmpl --name="RPZ-Template" --comm_key_name="NewCommunicationKeyName" --  
comm_key_value="NewCoomunicationKeyValue"
```

editrr**NAME:**

editrr

DESCRIPTION:

Updates a DNS resource record in 'object', 'zone' or 'revzone' scopes.

ARGUMENTS:**--owner**

Name part of the existing resource record. Should be a valid FQDN.
[mandatory]

--data

Data part of the existing resource record. [mandatory]

--rr_scope

Takes 'object', 'zone' or 'revzone'. Defines the context in which the resource record exist. [mandatory]

--type

Indicates the type of the resource record. Takes one of 'A', 'CNAME', 'MX', 'SRV', 'NS', 'TXT', 'NAPTR', 'PTR', 'DS', 'AAAA', 'DNAME', 'HINFO', 'CAA', 'LOC', 'LSA'. [mandatory]

--class

Indicates the class of the resource record. Support only 'IN' currently.

Valid class is 'IN'.

[mandatory]

--zone_name

Zone name of the target zone in TCPWave IPAM when rr_scope argument is specified as 'zone'.

--ipv4

IP address of the target object in TCPWave IPAM when defining resource record of type 'A'. This argument also represents IP address part when resource record type is 'PTR'.

--addr

IP Address of the reverse zone in TCPWave IPAM when rr_scope argument is specified as 'revzone'.

--mask

Mask length of the reverse zone in TCPWave IPAM when rr_scope argument is specified as 'revzone'.

--ttl

Indicates the time-to-live value specified in number of seconds for the resource record.

--new_owner

New owner name of the resource record.

Should be a valid domain name for records of type 'A'.

Should be a valid alias for records of type CNAME

Should be a valid domain name for records of type NS

Should be a valid domain name for record of type DS

--cname

CNAME data part of a CNAME record.

--domain

Domain name in data part of a PTR resource record.

--host

Host name in data part of a PTR resource record.

--name_server

Name Server or data part a NS resource record.

--org

Organization name to which the specified scope belongs. [mandatory]

--view

DNS view name to which resource record is being updated. This argument is applicable when --rr_scope is zone or object or revzone when type is PTR.

--prefnum

Preference number associated with an MX resource record.

--mail_host

Name of the server hosting the mail service associated with an MX resource record.

--service

Service name associated with an SRV resource record.

--protocol

Protocol associated with an SRV resource record.

--priority

Priority number associated with an SRV resource record.

--weight

Weight associated with an SRV resource record.

--port

Port number associated with an SRV resource record.

--target

Name of the server hosting the service associated with an SRV record.

Should point to a valid A record for records of type 'SRV'.

--txt

Text associated with a TXT resource record.

--order

Order number associated with an NAPTR resource record.

--flag

Flag value associated with an NAPTR resource record.

--params

Params value associated with an NAPTR resource record.

--regexp

Regexp value associated with an NAPTR resource record.

--replace

Replace field associated with an NAPTR resource record.

--key_tag

Key Tag data associated with a DS resource record. It should be positive integer value. Example: 100.

--algorithm

Algorithm data associated with a DS or SSHFP resource record.

It should be positive value for DS resource record integer value.

Example: 100.

Algorithm is mandatory for SSHFP resource record, it accepts the below algorithm types.

Example: RSA, DSA, ECDSA, Ed25519.

--digest_type

Digest type data associated with a DS resource record. It should be positive integer value. Example: 100.

--key_digest

Key Digest data associated with a DS resource record. It should be hexadecimal key.

--svc_subtype

Service subtype takes the value as '1' or '2'.

--desc

Description for the resource record.

--external_rr

Takes '0' or '1'. If this argument is specified as '1' resource record will be updated as an external resource record. This argument is applicable when --rr_scope=zone else it will be ignored.

--is_proxy

DNS Proxy root zone flag. It takes '0' or '1'. If it is specified as '1' resource record being updated will be treated as a proxy root zone resource record. If it is specified as '0' resource record being updated will be treated as a root zone resource record. This argument is applicable when --rr_scope=zone and --zone_name=.(dot).

--fprint_type

Fingerprint type data associated with SSHFP resource record. It should accept 'SHA-1' or 'SHA-256' value.

--fprint

Fingerprint data associated with SSHFP resource record.

--public_key

Public Key data associated with a DKIM resource record. The key should not have spaces.

--test_mode

Test mode data associated with a DKIM resource record. It should accept only 'Only domain' or 'Domain and sub-domains' value.

EXAMPLE:

```
twc editrr --owner=www.tcpwave.com. --data=10.0.0.1 --type=A --class=IN --rr_scope=object --  
ipv4=10.0.0.1 --ttl=5000 --new_owner=www --org=TCPWave --desc="changed the owner and TTL "
```

```
twc editrr --owner=ftp.tcpwave.com. --data=www.tcpwave.com. --type=CNAME --class=IN --  
rr_scope=object --ipv4=10.0.0.1 --cname=www1 --org=TCPWave --desc="Updated pointing A record"
```

```
twc editrr --owner=ftp.tcpwave.com. --data="10 www.tcpwave.com." --type=MX --class=IN --
ipv4=10.0.0.1 --rr_scope=object --prefnum=10 --org=TCPWave --desc="Updated preference number"
```

```
twc editrr --owner="_ldap._udp.www.tcpwave.com." --data="10 20 7443 lookup" --type=SRV --
class=IN --rr_scope=object --ipv4=10.0.0.1 --service=sip --protocol=udp --new_owner=www1 --
org=TCPWave --desc="Updated service, protocol and owner"
```

```
twc editrr --owner="_ldap._udp.www.tcpwave.com." --data="10 20 7443 lookup" --type=SRV --
class=IN --rr_scope=object --ipv4=10.0.0.1 --priority=1 --weight=10 --port=7001 --target=lookup --
org=TCPWave --desc="Updated priority, weight, port and service host"
```

```
twc editrr --owner="hostname.tcpwave.com." --data=10.0.0.4 --type=PTR --class=IN --
rr_scope=revzone --addr=10.0.0.0 --mask=24 --host=www --ttl=5000 --domain=tcpwave.com --
org=TCPWave --desc="Updated TTL and domain name"
```

```
twc editrr --owner="www.inter.com." --data=192.168.0.11 --type=PTR --host=www --class=IN --
rr_scope=revzone --mask=24 --addr=192.168.0.0 --ttl=5000 --domain=inter.com --org=Internal --
desc="ex" --ext_attr=test/value_1,rr/value_2 --ipv4=192.168.0.10
```

```
twc editrr --owner=www.tcpwave.com. --data=10.0.0.4 --type=A --class=IN --rr_scope=zone --
zone_name=tcpwave.com --ipv4=10.0.0.9 --org=TCPWave --desc="Updated IP Address"
```

```
twc editrr --owner=ftp.tcpwave.com. --data=www.tcpwave.com. --type=CNAME --class=IN --
rr_scope=zone --zone_name=tcpwave.com --cname=www1 --org=TCPWave --desc="Updated
pointing A record"
```

```
twc editrr --owner=text.tcpwave.com. --data="spf1 a:mail.tcpwave.com -all" --type=TXT --
class=IN --new_owner=text --rr_scope=zone --zone_name=tcpwave.com --txt="spf1
a:ftp.tcpwave.com -all" --org=TCPWave --desc="Updated txt data"
```

```
twc editrr --owner=ns.external.tcpwave.com. --data=ns.tcpwave.com. --type=NS --class=IN --
rr_scope=zone --zone_name=tcpwave.com --name_server=ns1.tcpwave.com. --org=TCPWave --
desc="Updated pointing Name Server"
```

```
twc editrr --owner=ns.internal.tcpwave.com. --data=ns.tcpwave.com. --type=NS --class=IN --rr_scope=zone --zone_name=. --new_owner=test.tcpwave.com. --ttl=5000 --org=TCPWave --desc="Updated TTL and owner name"
```

```
twc editrr --owner=ns.internal.tcpwave.com. --data="8 12 15 42" --type=DS --class=IN --rr_scope=zone --zone_name=tcpwave.com --new_owner=test --ttl=5000 --org=TCPWave --desc="Updated TTL and owner name"
```

```
twc editrr --owner=www.tcpwave.com. --data=10.0.0.1 --type=A --class=IN --rr_scope=zone --zone_name=. --ipv4=10.0.0.8 --org=TCPWave --desc="Updated IP Address of root zone A type RR"
```

```
twc editrr --owner=www.tcpwave.com. --data=10.0.1.12 --type=A --class=IN --rr_scope=zone --zone_name=. --is_proxy=1 --ttl=5000 --org=TCPWave --desc="Updated TTL of proxy root zone A type RR"
```

```
twc editrr --owner=www.tcpwave.com. --class=IN --type=URI --ttl=5000 --service=http --protocol=tcp --new_owner=www1 --priority=1 --weight=10 --target=lookup --org=TCPWave --desc="Owner is updated" --data="1 10 lookup" --rr_scope=zone --zone_name=www.tcpwave.com
```

```
twc editrr --owner=www.tcpwave.com. --class=IN --type=AFSDB --ttl=5000 --rr_scope=zone --zone_name=www.tcpwave.com --new_owner=www.tcpwave.com. --srvc_subtype=2 --org=TCPWave --desc="RR is updated" --host=www --data="1 www"
```

```
twc editrr --owner=www.tcpwave.com. --type=MX --class=IN --rr_scope=zone --zone_name=www.tcpwave.com --prefnum=20 --org=TCPWave --data="10 www.tcpwave.com." --new_owner=www.tcpwave1 --mail_host=www --desc="Updated preference number"
```

```
twc editrr --owner="_sip._tcp.www.tcpwave.com." --data="1 10 7001 www.tcpwave.com." --type=SRV --class=IN --rr_scope=zone --zone_name=www.tcpwave.com --service=sip1 --protocol=udp --org=TCPWave --new_owner=www.tcpwave1.com --priority=2 --port=7443 --weight=20 --target=www. --desc="Updated service, protocol"
```

```
twc editrr --type=SSHFP --class=IN --ttl=2100 --owner=tcpwave.com. --rr_scope=zone --
zone_name=tcpwave.com --org=TCPWave --fprint_type=SHA-1 --fprint=TCPWave --algorithm=RSA --
desc="Updated ttl" --new_owner=www.tcpwave1.com --data="1 1 TCPWave"
```

```
twc editrr --type=AFSDB --class=IN --ttl=5000 --owner=tcpwave.com. --svc_subtype=2 --host=arr.
--rr_scope=zone      --zone_name=tcpwave.com      --org=TCPWave      --data="1      arr."      --
new_owner=tcpwave.com
```

```
twc editrr --type=DKIM --class=IN --ttl=5000 --owner=_domainkey.tcpwave.com. --rr_scope=zone
--zone_name=tcpwave.com  --org=TCPWave  --test_mode="Only domain"  --public_key=145  --
data="v=DKIM1;t=s;p=12345" --new_owner=tcpwave1
```

editscheduledjob

NAME:

editscheduledjob

DESCRIPTION:

Updates a scheduled job in the TCPWave IPAM.

ARGUMENTS:

--job_id

Id of the scheduled job. [mandatory]

--repeat_type

Repeat type of the scheduled job. It takes one of 'daily', 'weekly', 'monthly', 'repetitive' or 'none'.

--exe_date

Execution date and time on which the scheduled job is to be executed. This argument is applicable if the repeat_type is 'none'. Date and time format is "yyyy-MM-dd hh:mm:ss".

--start_date

Start date and time from which the scheduled job is to be applied. This argument is applicable if the repeat_type is one of 'daily', 'weekly' or 'monthly'. Date and time format is "yyyy-MM-dd hh:mm:ss".

--end_date

End date and time till which the scheduled job is to be applied. This argument is applicable if the repeat_type is one of 'daily', 'weekly' or 'monthly'. Date and time format is "yyyy-MM-dd hh:mm:ss".

--exe_at

Execution time of the scheduled job. This argument is applicable if the repeat_type is one of 'daily', 'weekly' or 'monthly'. Time format is "hh:mm:ss".

-
- day_of_week
 - Day of the month if repeat_type is weekly. Takes number from 1 to 7.
 - day_of_month
 - Day of the month if repeat_type is monthly. Takes number from 1 to 31.
 - repeat_interval
 - Repeat interval of the scheduled job in minutes. This argument is applicable when repeat_type is 'repetitive'.
 - repeat_count
 - Repeat count of the scheduled job. This argument is applicable when repeat_type is 'repetitive'.
 - desc
 - Description of the scheduled job.

EXAMPLE USAGE:

```
twc editscheduledjob --job_id=ScheduledJobId --end_date="2018-10-25 00:00:00" --desc="Updated the ending date of the job"
```

```
twc editscheduledjob --job_id=ScheduledJobId --repeat_type=weekly --start_date="2017-12-25 12:00:00" --end_date="2018-12-25 00:00:00"
```

```
twc editscheduledjob --job_id=ScheduledJobId --repeat_type=monthly --start_date="2017-12-25 12:00:00" --end_date="2018-12-25 00:00:00" --exe_at=12:00:00 --day_of_month=1
```

```
twc editscheduledjob --job_id=ScheduledJobId --repeat_type=daily --exe_at=12:00:00 --start_date="2017-12-25 12:00:00" --desc="Updated monthly job into daily job"
```

```
twc editscheduledjob --job_id=ScriptJobId --repeat_interval=100 --repeat_count=6 --desc="Updated the repeat interval and repeat count"
```

editsubnet**NAME**

editsubnet

DESCRIPTION

Updates a subnet in the TCPWave IPAM.

ARGUMENTS**--subnet**

IP address of the target subnet. [mandatory]

--name

Name of the target subnet.

--type

Type of the subnet. Takes 'Non-DHCP', 'DHCP-Enabled' or 'Cloud-Hosted'.

--org

Name of the organization to which subnet belongs. [mandatory]

--subnet_group

Name of the associated subnet group. If this argument is specified as 'None' this will dissociate subnet group from the subnet.

--domain

Domain to be associated with this subnet.

--router_addr

IP address of the router associated with the subnet.

--dhcp_tmpl

Template name specifying the DHCP options for the subnet.

--dhcp_appliance

Primary DHCP appliance address for the subnet.

--dhcp_failover_peer

Name of the DHCP failover peer.

--domain_server

IP address of the DNS appliances. It accepts the comma separated DNS appliances, this argument is applicable only when type is specified as 'DHCP-Enabled'.

--shared_network

Name of the shared network. This argument is only applicable when type is specified as 'DHCP-Enabled'.

--enable_discovery

Enable discovery option for the subnet accepted as 'yes' or 'no'. Mandatory when enabled reclaim option is set to 'yes'.

--discovery_tmpl

Discovery template name for a subnet. Accepted only when enable discovery option is set to 'yes'

--enable_reclaim

If enabled, reclaim the eligible objects in the subnet based on the discovery result. Enable discovery option and discovery template name is mandatory when it is set to 'yes'.

--desc

Description text for the subnet.

--street1

Street1 part of the location information. Should be specified along with other mandatory location fields if the location has to be updated.

--street2

Street2 part of the location information. Should be specified along with other mandatory location fields if the location has to be updated.

--city

City part of the location information. Should be specified along with other mandatory location fields if the location has to be updated.

--state

State part of the location information. Should be specified along with other mandatory location fields if the location has to be updated.

--country

Country part of the location information. Should be specified along with other mandatory location fields if the location has to be updated.

--zip

Zip code part of the location information. Should be specified along with other mandatory location fields if the location has to be updated.

--secondary_domains

Name of the secondary domains to be associated with subnet. It accepts up to 50 secondary domains by separating with comma.

Example: "tcp.com,tcpwave.com,tcpzone.com"

--remove_location

Takes '0' or '1'. '1' indicates that the location information should be removed for the subnet. '0' indicates that the location information for the subnet remains unchanged or can be updated using the location arguments mentioned above. If this argument is not specified, it takes a default value of '0'.

--views

Comma-separated list of DNS view names to be associated with this subnet. Specified DNS views must be available for the primary domain.

--vlan

VLAN to be associated with this subnet.

--vrf_name

Name of the VRF to be associated with this subnet.

--cloud_provider

Name of the cloud provider to be associated with this subnet.

--ext_attr

Comma separated list of extension attributes with their values in the format : extension_attribute_name/extension_attribute_value. Use the following command to see all the extension attributes applied to subnet : 'twc listext --entity=subnet --d=,'

--contact_fname

First name field of the associated contact information for the subnet.

--contact_mname

Middle name field of the associated contact information for the subnet.

--contact_lname

Last name field of the associated contact information for the subnet.

--contact_email

Email ID field of the associated contact information for the subnet.

EXAMPLE

```
twc editsubnet --subnet=10.0.10.0 --subnet_group=IT-SG --desc="IT Subnet" --  
domain=dev.tcpwave.com --type=DHCP-Enabled --dhcp_tmpl=Generic --dhcp_appliance=10.0.10.180  
--router_addr=10.0.10.1 --org=TCPWave --views=view1,view2
```

```
twc editsubnet --subnet=10.0.10.0 --type=Non-DHCP --remove_location=1 --org=TCPWave
```

```
twc editsubnet --subnet=10.0.10.0 --subnet_group=IT-SG --desc="IT Subnet" --  
domain=dev.tcpwave.com --type=DHCP-Enabled --dhcp_tmpl=Generic --dhcp_appliance=10.0.10.180  
--router_addr=10.0.10.1 --org=TCPWave --ext_attr=ext_attr_1/value_1,ext_attr_2/value_2
```

```

tvc editsubnet --subnet=10.0.10.0 --name=sub-1 --org=Internal --enable_discovery=yes --
discovery_tmpl=tera

tvc editsubnet --subnet=10.0.10.0 --name=sub-1 --org=Internal --enable_reclaim=yes --
enable_discovery=yes --discovery_tmpl=tera

tvc editsubnet --subnet=10.14.0.0 --domain=dev.tcpwave.com --type=DHCP-Enabled --
dhcp_tmpl=Generic --dhcp_appliance=10.14.0.13 --dhcp_failover_peer=dhcp-failover-peer-1 --
router_addr=10.14.0.1 --org=TCPWave

tvc editsubnet --subnet=10.0.10.0 --contact_fname=John --contact_lname=Smith --
contact_email=john.smith@tcpwave.com --org=TCPWave

tvc editsubnet --subnet=10.0.10.0 --contact_fname=John --contact_lname=Smith --
contact_email=john.smith@tcpwave.com --org=TCPWave --
secondary_domains="tcpwave.com,tcpwave1.com"

```

editsubnetgroup

NAME:

editsubnetgroup

DESCRIPTION:

Updates a subnet group in the TCPWave IPAM.

ARGUMENTS:

--name

Name of the subnet group being updated. [mandatory]

--org

Organization name of the subnet group. [mandatory]

--new_name

New name for the subnet group if the name needs to be updated. [updatable field]

--desc

Description of the subnet group. [updatable field]

EXAMPLE USAGE:

```
tvc editsubnetgroup --name=IT_SG --new_name=Sales_SG --desc="IT subnet group" --
org=Internal
```

editdhcpsharednetwork

NAME:

editdhcpsharednetwork

DESCRIPTION:

Updates a DHCP shared network in the TCPWave IPAM.

ARGUMENTS:

- name
Name of the shared network. [mandatory]
- new_name
New name of the shared network.
- ip
IP address of the DHCP primary appliance. [mandatory]
- new_ip
New IP address of the DHCP primary appliance.
- desc
Description of the shared network.

EXAMPLE:

```
twc editdhcpsharednetwork --name=TestNet --new_name=TcpSNetwork --ip=10.0.0.213 --desc="Tcpwave Shared Network"
```

```
twc editdhcpsharednetwork --name=TestNet --ip=10.0.0.213 --new_ip=10.0.0.215 --desc="Tcpwave Shared Network"
```

editcloudprovider

NAME:

editcloudprovider

DESCRIPTION:

Updates a cloud provider in the TCPWave IPAM. Different type of cloud providers support different credentials. Follow the example section to edit particular type of cloud provider.

ARGUMENTS:

- org
Organization name to be associated with the cloud provider. This argument is for users in FADM role to select appropriate organization which the operation has to be applied. For users not in FADM role operation is by default applied to the organization that the user is associated with.
- provider_type
Type of the Cloud provider. In TCPWave IPAM provider type represents the cloud service provider . TCPWave IPAM support following cloud providers.
'AKAMAI', 'AWS', 'AZURE', 'CLOUDFLARE', 'DYNDNS' and 'GOOGLE' [mandatory].
- name
Name of the cloud provider [mandatory].
- new_name
New name of the cloud provider.
- user
User name of the cloud provider.

--api_key

API key for the cloud provider. It is Secret access key for AWS and it is global API key for CLOUDFLARE provider.

--keystore_file

Key store file for the cloud provider. This key store file contains secret access key. It is applicable only for AZURE provider type

--application

Application ID for the cloud provider. It is applicable for AZURE type cloud provider.

--password

Password for the cloud provider. It is applicable for DYNDNS type cloud provider.

--service_account_id

Service Account ID for the GOOGLE type cloud provider.

--p12file

p12file for the GOOGLE type cloud provider.

--project_id

Project ID for the GOOGLE type cloud provider.

--ad_tenant

Ad tenant ID for the AZURE type cloud provider.

--resource_group

Resource Group for the AZURE type cloud provider.

--customer_name

Customer name for the DYNDNS type cloud provider type.

--email

Email Address for the CLOUDFLARE provider.

--region

Region defines area of AWS type cloud provider. For AWS Cloud provider region can be one of the following-

US East (N. Virginia)

US East (Ohio)

US West (N. California)

US West (Oregon)

Asia Pacific (Mumbai)

Asia Pacific (Seoul)

Asia Pacific (Singapore)

Asia Pacific (Sydney)

Asia Pacific (Tokyo)

Canada (Central)

EU (Frankfurt)
EU (Ireland)
EU (London)
South America (Sao Paulo)

--desc
Description for the cloud provider.

EXAMPLE:

```
twc editcloudprovider --name="AWS-Provider50" --org=TCPWave --user="AK
IAINLQMEG7EBWXMTQOP" --api_key="E52BxojR5f2hM802hG+Zl8Z4boxzlZRNCnPpaii1+" --
region="EU (London)" --desc="AWS Cloud Provider"
```

```
twc editcloudprovider --name="Google-Provider22" --org=TCPWave --service
e_account_id="jhon@tcpwave-14981012.iam.gserviceaccount.com" --p12file="/tmp/tc
pwave-2d185caa49dc1.p12" --project_id="tcpwave-14912810" --desc="Google Cloud Pro
vider"
```

```
twc editcloudprovider --name="Azure-Provider03" --org=TCPWave --user="p
pc0e31c0f-fdb0-438c-afff-6ea7600b0e61" --keystore_file="cloud_dns_app.pfx" --app
lication="ebe1b568-5e63-46f0-9201-8a465cee092dqq" --ad_tenant="772a8482-16c9-482
3-9f15-bd19827d23f111" --resource_group="tcpwave" --password="abc123" --desc="Az
ure Cloud Provider"
```

```
twc editcloudprovider --name="DynDNS-Provider01" --org=TCPWave --user="jhon-
smith" --password="123CO2zbCJ6Qb" --customer_name="tcpwave01" --desc="DYNDNS Cloud
Provider"
```

```
twc editcloudprovider --name="CLOUDFLARE-Provider02" --org=TCPWave --em
ail_addr="jhon.tcpwave@tcpwave.com" --api_key="3cde9f553a9a21049e00046" --
desc="CLOUDFLARE Cloud Provider"
```

```
twc editcloudprovider --name="Akamai-Provider06" --org=TCPWave --user="jhon.tcpwave@tcpwave.com" --password="Glider0N123#" --api_key="client_secret = x
d3RTCMImmZhdQ82LD34yAZUqOwc2DDt1ANgDAoc6iguY=host = akab-34nyw47p22fhpvptnu-
v7ygacgwkb6cswza.luna.akamaapis.netaccess_token = akab-a24w5rojdc6lckdmt-
cvscbkoo5ise5bw2 client_token = akab-sxdp7uvgkonm7jfu-w3phslypnzzv3llqv" --desc="AKAMAI
Cloud Provider"
```

editnsmtmp**NAME****editnsmtemplate****DESCRIPTION**

Edit a NSM template for a given organization in the TCPWave IPAM.

ARGUMENTS**--org**

Organization name for which NSM template is being edited. [mandatory]

--old_tmpl_name

Name of the NSM template. [mandatory]

--new_tmpl_name

New name of the NSM template.

--desc

Description for the NSM template.

--network_interface

Network interface for NSM template. [mandatory]

--anomaly_detection

Takes 'true' or 'false'. If it is true, Anomaly detection will enable on the NSM template.

If it is 'false', Anomaly detection will disable on the NSM template.

--ml_model

It accepts the numeric value from '0' to '7'.

--intrusion_detection

It takes 'true' or 'false'. If it is true, Intrusion detection will enable on the NSM template.

If it is 'false', Intrusion detection will disable on the NSM template.

--intrusion_prevention

It takes 'true' or 'false'. If it is true, Intrusion prevention will enable on the NSM template.

If it is 'false', Intrusion prevention will disable on the NSM template.

--ips_rules

It takes multiple rules by separating with '|' symbol.

--rule

It takes address or port, name and value for the rule variable by separating with comma. It can accept multiple values by separating with pipe symbol.

Example: address,HOME,10.1.10.1|port,HOME_NET,123

EXAMPLE

```
twc      editnsmtmpl      --org=TCPWave      --old_tmpl_name=FirstTemplate      --
new_tmpl_name=FirstTemplate1 --network_interface=eth0 --anomaly_detection=true  --ml_model
=1 --intrusion_detection=true --intrusion_prevention=true --ips_rules="alert dns any any -> any
any(msg:TCPWAVE DNS TITAN This is a alert test for Example;dns_query;content:www.example.com;
depth:14;fast_pattern;  endswith;  nocase;  classtype :pup-activity;  sid:9999991;  rev:33;)" --
rule="address,HOME_NET,10.1.10.1|port,NET,123" --desc="TCPWave NSM Template"
```

```
twc      editnsmtmpl      --org=TCPWave      --old_tmpl_name=FirstTemplate      --
new_tmpl_name=FirstTemplate1      --network_interface=eth0      --anomaly_detection=false      --
intrusion_detection=false --intrusion_prevention=false --desc="TCPWave NSM Template"
```

editvrf**NAME**

editvrf

DESCRIPTION

Updates a VRF in the TCPWave IPAM.

ARGUMENTS**--name**

Name of VRF. [mandatory]

--org

Name of the organization. [mandatory]

--new_name

New name of VRF.

--router_distinguisher

Enter the AS number or IP address of the route distinguisher of the discovered VRF.

--interface

VRF can be assigned to any interface loopback or VLAN. Example: f0/0.82

--import_target

Imports routing information from the target extended community.

--export_target

Exports routing information to the target extended community.

--desc

Description of the VRF.

EXAMPLE

```
twc editvrf --name=testVrf --new_name=newVrf --org=TCPWave --router_distinguisher=100:30 -  
-interface=1/1 --import_target=10.1.1.10 --export_target=10.1.1.20 --desc="Test VRF"
```

enablezonemonitor**NAME:**

enablezonemonitor

DESCRIPTION:

Enables the monitoring on a given list of zones in the TCPWave IPAM.

ARGUMENTS:

--zone_list

Takes comma separated list of zone names. [mandatory]

--org

Name of the organization to which the specified zones belong. This argument is mandatory if the user is FADM.

EXAMPLE USAGE:

```
twc enablezonemonitor --zone_list=dev.tcpwave.com, tcpwave.com --org=TCPWave
```

```
twc enablezonemonitor --zone_list=dev.tcpwave.com,tcpwave.com
```

exescheduledjob

NAME:

exescheduledjob

DESCRIPTION:

Executes a scheduled job in the TCPWave IPAM.

ARGUMENTS:

--job_id

Id of the scheduled job. [mandatory]

EXAMPLE USAGE:

```
twc exescheduledjob --job_id=RemoteMonitStatsOperation
```

Exports

exportadminrole

NAME

exportadminrole

DESCRIPTION

Exports the administrator roles from the TCPWave IPAM as a csv list into a specified output file.

ARGUMENTS

--output_file

full path to the output file to which administrators are exported.

[mandatory]

EXAMPLE

```
twc exportadminrole --output_file=/tmp/output.txt
```

IMPORT FILE

The information to create an administrator role should be in the order below

```
[administrator-roles]
NAME= Name of the role
INTERNAL= Type of the role (False/True)
DESCRIPTION= Description of the role
[functions]
NAME=Name of the function
DESCRIPTION= Description
GRANULAR_SUPPORT= False/True
```

EXAMPLE DATA:

```
[administrator-roles]
NAME=QADM
INTERNAL=false
DESCRIPTION=quality checks
[functions]
NAME=Quick Tasks
DESCRIPTION=
GRANULAR_SUPPORT=false
[functions]
NAME=Bulk Data Export
DESCRIPTION=
GRANULAR_SUPPORT=false
[functions]
NAME=AWS Images
DESCRIPTION=
GRANULAR_SUPPORT=true
[administrator-roles]
NAME=RADM
INTERNAL=true
DESCRIPTION=Read-only Admin
exportadmin
NAME
```

exportadmin

DESCRIPTION

Exports the administrators from the TCPWave IPAM as a csv list into a specified output file.

ARGUMENTS

--output_file

full path to the output file to which administrators are exported. [mandatory]

--org

Name of the organization from which administrators are to be exported. If this argument is not specified, administrators from all organizations will be exported for FADM user, administrators from user's organization will be exported for non-FADM user.

EXAMPLE:

twc exportadmin --output_file=/tmp/output.txt

twc exportadmin --org=TCPWave --output_file=/tmp/output.txt

EXPORT FILE

The information in the output file is a comma separated list of fields specified in the order below

ORG_NAME, FIRST_NAME, MIDDLE_NAME, LAST_NAME, EMAIL, PHONE, LOGIN_NAME,
ADMIN_ROLE, ADMIN_GROUPS, XTN_ADMINEXTENSION

EXAMPLE DATA

"TCPWave","John","","Smith","jsmith@tcpwave.com","9000102010","jsmith","NADM","default","","

exportadmingroup

NAME

exportadmingroup

DESCRIPTION

Exports the administrator groups from the TCPWave IPAM to a csv file.

ARGUMENTS

--output_file

Full path to the output file to which administrator groups are exported. [mandatory]

EXAMPLE

twc exportadmingroup --output_file=/tmp/output.txt

EXPORT FILE:

The information in the output file list of fields as specified in the order below

[admin-group]

NAME=Name of the admin group

DESCRIPTION=Description for admin group

[role]

ROLE=Name of the role

ORGANIZATION=Name of the organization

EXAMPLE DATA:

[admin-group]

NAME=CADM-Test

DESCRIPTION=Group with Quick Tasks,Architecture Overview

[role]

ROLE=CADM-First

ORGANIZATION=EARTH

[role]

ROLE=RADM

ORGANIZATION=Internal

[admin-group]

NAME=Default EARTH RADM Group

DESCRIPTION=Default EARTH RADM Group

[admin-group]

NAME=CADM-lrMQMF-group

DESCRIPTION=Group with Quick Tasks,Administrator Groups

[role]

ROLE=CADM-One

ORGANIZATION=EARTH

[role]

ROLE=RADM

ORGANIZATION=Internal

[admin-group]

NAME=Default Internal RADM Group

DESCRIPTION=Default Internal RADM Group

exportadminpermission

NAME

exportadminpermission

DESCRIPTION

Exports administrator permissions from the TCPWave IPAM as a csv list into the specified output file

ARGUMENTS

--output_file

Full path to the output file to which administrator permissions are exported. [mandatory]

--org

Name of the organization from which administrator permissions are to be exported. If this argument is not specified, administrator permissions from all organizations will be exported for FADM user, administrator permissions from user's organization will be exported for non-FADM user.

EXAMPLE

```
twc exportadminpermission --output_file=/tmp/output.txt
```

```
twc exportadminpermission --org=TCPWave --output_file=/tmp/output.txt
```

EXPORT FILE

The information in the output file is a comma separated list of fields as specified in the order below

"PERMISSION_LEVEL","ROLE","FUNCTION","VALUE","SELECT_ALL","ADMIN_GROUP","ADMIN","ORG_NAME","PRIVILEGE"

EXAMPLE DATA:

```
"Admin","PADM","IPv4 Subnets","","Yes","","wfpadm","Internal","Write"
```

```
"Admin","PADM","TCPWave DHCP IPv4 Appliances","","Yes","","wfpadm","Internal","Write"
```

```
"Admin Group","NADM","IPv4 Objects","","Yes","Default Internal NADM Group","","Internal","Write"
```

exportappliancegroup

NAME:

exportappliancegroup

DESCRIPTION:

Exports the appliance groups from the TCPWave IPAM as a csv list into the specified, output file.

ARGUMENTS:

--output_file

Full path to the output file to which appliance groups are to be exported. [mandatory]

--org

Name of the organization from which appliance groups are to be exported. If this argument is not specified, appliance groups from all organizations will be exported for FADM user, appliance groups from user's organization will be exported for non-FADM user.

EXAMPLE USAGE:

```
twc exportappliancegroup --output_file=/tmp/output.txt
```

```
twc exportappliancegroup --org=TCPWave --output_file=/tmp/output.txt
```

EXPORT FILE:

The information in the output file is a comma separated list of fields as specified, in the order below.

"NAME","ORG_NAME","DESCRIPTION"

EXAMPLE DATA:

"NAME","ORG_NAME","DESCRIPTION"
 "app-group1234","Internal","testing qa done"
 "app-group-1256","Internal","testing qa"
 "TCPWave-Remote-Group","TCPWave Organization","TCPWave-Remote-Group"

exportasset**NAME:**

exportasset

DESCRIPTION:

Export assets from the TCPWave IPAM as a csv list into the specified, output file.

ARGUMENTS:

--export_level

Export level. It takes 'asset'. [mandatory]

--output_file

Full path to the output file to which assets are to be exported. [mandatory]

--org

Name of the organization from which assets are to be exported. If this argument is not specified, assets from all organizations will be exported for FADM user, assets from user's organization will be exported for non-FADM user.

EXAMPLE USAGE:

twc exportasset --export_level=asset --output_file=/tmp/output.txt

twc exportasset --export_level=asset --org=TCPWave --output_file=/tmp/output.txt

EXPORT FILE:

The information in the output file is a comma separated list of fields as specified, in the order below

"SERVICE_TAG","SERIAL_NUM","GREEN_ZONE","VENDOR","MODEL","NAME","DESCRIPTION","PURCHASE_COST","PURCHASE_DATE","ACQUISITION_TYPE","MAINTENANCE_COST","MAINTENANCE_END_DATE","WARRANTY_END_DATE","CPU","CAPACITY","OS_VERSION","DISPOSAL_DATE","CITY"

EXAMPLE DATA:

"temptest","1-2-3-4-5","1","temp","temp2","temp","temp","2","2017-02-15 00:00:00","","","","","2017-

02-09 00:00:00","temp","temp","temp","2017-02-01 00:00:00","temp"

exportcontact

NAME:

exportcontact

DESCRIPTION:

Exports the contacts from TCPWave IPAM as a CSV list into the specified, output file.

ARGUMENTS:

--output_file

Full path to the output file to which contacts are exported. [mandatory]

--org

Name of the organization from which contacts are to be exported. If this argument is not specified, contacts from all organizations will be exported for FADM user, contacts from user's organization will be exported for non-FADM user.

EXAMPLE USAGE:

twc exportcontact --output_file=/tmp/output.txt

twc exportcontact --org=TCPWave --output_file=/tmp/output.txt

EXPORT FILE:

The information in the output file is a comma separated list of fields as specified, in the order below

"FIRST_NAME","LAST_NAME","EMAIL_ID","PHONE_NUM","ORG_NAME"

EXAMPLE DATA:

"John","Smith","john.smith@tcpwave.com","920-310-5555","TCPWave"

exportdhcpfailoverpeer

NAME:

exportdhcpfailoverpeer

DESCRIPTION:

Exports the DHCP failover peers from the TCPWave IPAM as a csv list into the specified, output file.

ARGUMENTS:

--output_file

Full path to the output file to which DHCP failover peers are to be exported. [mandatory]

--org

Name of the organization from which DHCP failover peers are to be

exported. If this argument is not specified, DHCP failover peers from all organizations will be exported for FADM user, DHCP failover peers from user's organization will be exported for non-FADM user.

EXAMPLE USAGE:

```
twc exportdhcpfailoverpeer --output_file=/tmp/output.txt
```

```
twc exportdhcpfailoverpeer --org=TCPWave --output_file=/tmp/output.txt
```

EXPORT FILE:

The information in the output file is a comma separated list of fields as specified, in the order below.

```
"NAME","ORGANIZATION_NAME","PRIMARY_APPLIANCE_IP","FAILOVER_APPLIANCE_IP","PRIMARY_APPLIANCE_PORT","FAILOVER_APPLIANCE_PORT","MCLT","SPLIT","LOAD_BALANCE_MAX_SECONDS","MAX_RESPONSE_DELAY","MAX_UNACKED_UPDATES","PRIMARY_APPLIANCE_NAME","FAILOVER_APPLIANCE_NAME","DESCRIPTION"
```

EXAMPLE DATA:

```
"NAME","ORGANIZATION_NAME","PRIMARY_APPLIANCE_IP","FAILOVER_APPLIANCE_IP","PRIMARY_APPLIANCE_PORT","FAILOVER_APPLIANCE_PORT","MCLT","SPLIT","LOAD_BALANCE_MAX_SECONDS","MAX_RESPONSE_DELAY","MAX_UNACKED_UPDATES","PRIMARY_APPLIANCE_NAME","FAILOVER_APPLIANCE_NAME","DESCRIPTION"
```

```
"dhcp-failover-peer-
```

```
1","TCPWave","16.0.0.2","15.0.0.2","647","647","1800","120","3","30","30","DHCP-16.0.0.2","DHCP-15.0.0.2","","
```

```
"dhcp-failover-peer-
```

```
5","TCPWave","163.35.7.57","172.175.231.5","647","648","1600","128","2","50","40","dhcp-server01-sl0984","dhcp-server01-sl0984","","
```

```
"demo-peer-
```

```
1","TCPWave","172.181.11.243","172.175.156.133","647","647","1800","128","3","30","30","nhkna81-qrs01-sl0984","dhcp-server02-sl0984","","
```

```
"demo-peer-
```

```
2","TCPWave","9.0.3.4","9.0.1.2","647","647","1800","192","3","30","30","TemDHCPServer","dhcp-server-2","","
```

exportdhcptiontmp

NAME:

`exportdhcptiontmp`

DESCRIPTION:

Exports the DHCP option templates from the TCPWave IPAM to a name-value pair format file.

ARGUMENTS:

`--output_file`

Full path to the output file to which DHCP option templates are exported.
[mandatory]

--org

Name of the organization from which DHCP option templates are to be exported. If this argument is not specified, DHCP option templates from all organizations will be exported for FADM user, DHCP option templates from user's organization will be exported for non-FADM user.

EXAMPLE USAGE:

```
twc exportdhcptiontmpl --output_file=/tmp/output.txt
```

```
twc exportdhcptiontmpl --org=TCPWave --output_file=/tmp/output.txt
```

EXPORT FILE:

The information in the output file in a format as described below

Each DHCP option template starts with a section [dhcp-option-template] followed by various DHCP parameters in the format <param-name>=<param-value> one per each line.

TemplateName is the name of the DHCP option template and is mandatory

Other DHCP option parameters take the GUI display name as the <param-name>

SECTIONS & CONFIGURATION PARAMETERS:**[dhcp-option-template]**

TemplateName: Defines the name of the DHCP option template.

Organization: Defines the name of organization name template is associated.

Description: Display description of DHCP option template.

User Authentication Servers: Display the name of the User Authentication Servers.

Default TCP TTL: Defines the default time-to-live value in seconds.

Keepalive Time: Defines the client waiting time for sending alive messages in seconds.

Keepalive Data: Display true if sends live messages with an octet compatibility else false.

Service Location Protocol Directory Agent: Defines true/false with SLP agent IP address.

SLP Service Scope: Defines true/false with a list of service scopes for SLP.

Domain Search: Defines the domain name.

Subnet Mask: Defines the input as 'Same as in subnet profile' only.

Time Offset: Defines the Time Offset value in seconds.

Router: Defines the input as 'Same as in subnet profile' only.

Time Server: Defines the comma separated list of valid IPV4 addresses of Time servers.

Name Server: Defines the comma separated list of valid IPV4 addresses of Name servers.

Domain Name Server: Defines the comma separated list of valid IPV4 addresses of Domain Name servers.

Log Server: Defines the comma separated list of valid IPV4 addresses of Log servers.

Quotes Server: Defines the comma separated list of valid IPV4 addresses of Quotes servers.

LPR Server: Defines the comma separated list of valid IPV4 addresses of LPR servers.

Impress Server: Defines the comma separated list of valid IPV4 addresses of Impress servers.

RLP Server: Defines the comma separated list of valid IPV4 addresses of RLP servers.

Hostname: Defines the name of the client.

Boot File Size: Defines the boot file size.

Merit Dump File: Defines the path of Merit dump file for the DHCP option template.

Domain Name: Defines the input as 'Same as the primary domain in subnet profile' only.

Swap Server: Defines the IPV4 address for Swap server.

Root Path: Defines the path of root disk for the DHCP option template.

Extension File: Defines the name of Extension file for the DHCP option template.

NetWare/IP Domain: Defines the name of NetWare/IP domain for the client to use.

NetWare/IP Options-nwip.nsq-broadcast: Defines true to use the NetWare Nearest Server Query to locate a NetWare/IP server else false.

NetWare/IP Options-nwip.preferred-dss: Defines the comma separated list of valid IPV4 addresses of NetWare Domain SAP/RIP servers.

NetWare/IP Options-nwip.nearest-nwip-server: Defines the comma separated list of valid IPV4 addresses of NetWare servers.

NetWare/IP Options-nwip.autoretries: Defines the valid Integer for the number of times that a NetWare/IP client should attempt to communicate with a given DSS server at startup.

NetWare/IP Options-nwip.autoretry-secs: Defines the number of seconds that a NetWare/IP client should wait between retries when attempting to establish communications with a DSS server.

NetWare/IP Options-nwip.nwip-1-1: Defines true for NetWare/IP client to support NetWare/IP version 1.1 compatibility else false.

NetWare/IP Options-nwip.primary-dss: Defines the valid IPV4 address of the Primary Domain SAP/RIP Service server.

NDS Servers: Defines the comma separated list of valid IPV4 addresses of NDS servers.

NDS Tree Name: Defines the name of NDS Tree for DHCP option template.

NDS Context: Defines the name of the initial NetWare Directory Service for a NDS client.

Address Request: Defines the comma separated list of valid IPV4 addresses to be used by the client in a DHCP discover to request that a particular IP addresses.

DHCP Message Type: Defines the type of DHCP message.

Parameter List: Defines the comma separated list of valid DHCP options for the client to request DHCP Appliance to return on request.

DHCP Max Msg Size: Defines the maximum size of response that the appliance sends to the client.

Home Agent Addresses: Defines the comma separated list of valid IPV4 addresses for mobile home agents.

User Class: Defines the name of User class to be specified, for DHCP appliance.

Netinfo Address: Defines the comma separated list of Valid NetInfo IPV4 address.

Netinfo Tag: Defines the name of the NetInfo tag for DHCP option template.

Default URL: Defines the value for Default URL.

Vendor Identified Vendor-Specific Information: Defines the Vendor class name associated with DHCP appliance.

Client FQDN: Defines a FQDN for the client to use.

MTU Subnet: Display true if takes the same MTU for all the subnets of the IP network else false.

Trailers: Display true if the client uses trailers else false.

ARP Timeout : Display time in seconds for ARP cache entries.

Ethernet: Display true if the client uses Ethernet Version 2 (RFC 894) else false.

Forward On/Off: Display true if the client configures its IP layer for packet forwarding else false.

Source Routing: Display true if the client configures its IP layer to allow forwarding of datagrams with non-local source routes else false.

Policy Filter: Defines the comma separated list of valid IPV4 addresses for non-local source routing.

Max Datagram Size: Defines the maximum size of the datagram that client should be prepared to reassemble.

Default IP TTL: Defines the time-to-live in seconds for the client to use on outgoing datagrams.

MTU Timeout: Defines the MTU Timeout in seconds.

MTU Plateau: Defines an Integer for MTU plateau to use when performing Path MTU Discovery.

Mask Discovery: Display true if the client perform Mask discovery using ICMP else false.

Mask Supplier: Display true if the client respond to subnet mask requests using ICMP else false.

Router Discovery: Display true if the client perform Router discovery else false.

Router Request: Defines an IPV4 address to which the client transmits router solicitation requests.

Static Route: Defines the comma separated list of valid IPV4 addresses that client should install in its routing cache.

MTU Interface: Defines the valid Integer for the MTU Interface.

Broadcast Address: Defines the valid IPV4 address.

Address Time: Defines the client request lease time in seconds.

Overload: Defines the valid Integer for DHCP appliance to insert if the returned parameters will exceed the usual space allotted for options.

Vendor Class Id: Defines the value for Vendor class Id.

Client Id: Defines the value for Client Id.

Server Name: Defines the name of the Server to identify a TFTP server.

Bootfile Name: Defines the name of Bootfile to be used by the client.

NETBIOS Dist Server: Defines the comma separated list of valid IPV4 addresses for NETBIOS Dist servers.

NETBIOS Node Type: Defines the valid NetBIOS node type which allows NetBIOS over TCP/IP clients to configure as per RFC 1001/1002.

NETBIOS Scope: Defines the value for NetBIOS scope to specifies the NetBIOS over TCP/IP scope parameter for the client.

X Window Font: Defines the comma separated list of valid IPV4 addresses of X Window System Font servers.

X Window Manager: Defines the comma separated list of valid IPV4 addresses of X Window Manager servers.

NIS+ Domain Name: Defines the name for NIS domain.

NIS+ Server Address: Defines the comma separated list of valid IPV4 addresses of NIS servers.

SMTP Server: Defines the comma separated list of valid IPV4 addresses of SMTP servers.

POP3 Server: Defines the comma separated list of valid IPV4 addresses of POP3 servers.

NNTP Server: Defines the comma separated list of valid IPV4 addresses of NNTP servers.

WWW Server: Defines the comma separated list of valid IPV4 addresses of WWW servers.

Finger Server: Defines the comma separated list of valid IPV4 addresses of Finger servers.

IRC Server: Defines the comma separated list of valid IPV4 addresses of IRC servers.

StreetTalk Server: Defines the comma separated list of valid IPV4 addresses of StreetTalk servers.

StreetTalk Directory Assistance (STDA) Server : Defines the comma separated list of valid IPV4 addresses of STDA servers.

BCMCS Controller IPv4 address option: Defines the comma separated list of valid IPV4

addresses of BCMCS servers.

NIS Domain: Defines the value for NIS domain.

NIS Servers: Defines the comma separated list of valid IPV4 addresses of NIS servers.

NTP Servers: Defines the comma separated list of IPV4 address of NTP servers.

Vendor Specific: Defines the value for vendor specific name.

NETBIOS Name Server: Defines the comma separated list of valid IPV4 addresses of NETBIOS Name servers.

EXAMPLE DATA:

```
[dhcp-option-template]
```

```
TemplateName=OptionTemplate10
```

```
Organization=Internal
```

```
Description=
```

```
Subnet Mask=Same as in subnet profile
```

```
Router=Same as in subnet profile
```

```
Domain Name=Same as the primary domain in subnet profile
```

```
[dhcp-userdefined-option]
```

```
OPTION, GROUP, DATA_TYPE
```

```
[dhcp-option-template]
```

```
TemplateName=OptionTemplate11
```

```
Organization=Internal
```

```
Description=
```

```
Subnet Mask=Same as in subnet profile
```

```
Router=Same as in subnet profile
```

```
Domain Name=Same as the primary domain in subnet profile
```

```
[dhcp-userdefined-option]
```

```
OPTION, GROUP, DATA_TYPE
```

exportdhcpolicytmpl

NAME:

`exportdhcpolicytmpl`

DESCRIPTION:

Exports the DHCP policy template from the TCPWave IPAM to a name-value pair format file.

ARGUMENTS:

`--output_file`

Full path to the output file to which DHCP policy templates are exported.
[mandatory]

`--org`

Name of the organization from which DHCP policy templates are to be exported. If this argument is not specified, DHCP policy templates from all organizations will be exported for FADM user, DHCP policy templates from user's organization will be exported for non-FADM user.

EXAMPLE USAGE:

```
twc exportdhcpolicytmpl --output_file=/tmp/output.txt
```

tvc exportdhcpolicytmpl --org=TCPWave --output_file=/tmp/output.txt

EXPORT FILE:

The information in the output file in a format as described below

Each DHCP policy template starts with a section [dhcp-policy-template] followed by various DHCP policy parameters in the format
<param-name>=<param-value> one per each line.

TemplateName is the name of the DHCP policy template and is mandatory

Other DHCP policy parameters take the GUI display names as the <param-name>

SECTIONS & CONFIGURATION PARAMETERS:

[dhcp-policy-template]

TemplateName	Defines name of the DHCP option template.
Organization	Defines the name of organization name template is associated.
Description	Defines the description of DHCP policy template.
Authoritative	Display 'yes' or 'no'.
DB Time Format	Defines default/local value for DB time format to format timestamp in lease information. This takes 'default' or 'local'.
Local Port	Defines an Integer value as the port number on which DHCP appliances receives messages.
Local Address	Defines an IPV4 address on which the DHCP Appliance gets DHCP messages.
Log Facility	Defines the Log Facility name.
Always Broadcast	Displays true if DHCP Appliance always broadcasts its responses to clients within the scope of the parameter defined else false.
Always reply RFC1048	Displays true if DHCP Appliance format options in RFC 1048 format else false.
Min Secs	Defines the minimum value in seconds for DHCP Appliance to process the message.
Remote Port	Defines an Integer value which override default port number on which DHCP messages are sent to clients.
Stash Agent Options	Displays true if the DHCP appliance stores the DHCP relay agent information else false.
Adaptive Lease Time Percentage	Defines an Integer value for the DHCP appliance to automatically decrease lease time for new clients to min-lease-time when the allocated leases as a percentage of pool capacity exceed given percent.
Boot Unknown Clients	Displays true if the DHCP appliance offers IPV4 addresses for the clients which are not declared with a host declaration format else false.
Default Lease time	Defines an number of seconds provided to clients that do not request for given specified, time.
Get Lease Host Names	Displays true if DHCP server lookup the hostname corresponding to the assigned IP address and set the resolved hostname in the DHCP hostname option else false.
Infinite is reserved	Displays on if the Client is getting infinite lease time else off.
Max Lease Time	Defines the maximum lease time in seconds.
Min Lease Time	Defines the minimum lease time in seconds.
Next Server	Defines the Name server from which the client obtains its boot file.

One Lease Per Client	Displays true if DHCP appliance assigns the requested address and free any other leases associated with the client else false.
Ping Check	Displays true if the DHCP appliance ping the address before issuing the offer to client else false.
Ping Timeout	Defines ping timeout value in seconds.
Use Lease Addr For Default Route	Displays true if the router option is same IP address as that offered by the client else false.
Server Identifier	Defines an IPV4 address for the Server Identifier.
Server Name	Defines a FQDN name for Server name.
Site Option Space	Defines the name of Site option space.
Vendor Option Space	Defines the name of Vendor option space.

EXAMPLE DATA:

```
[dhcp-policy-template]
TemplateName=policy1
Organization=QAOrg
Description=test
Default Lease time=3600
```

exportdhcpserver**NAME:**

exportdhcpserver

DESCRIPTION:

Exports the DHCP servers from the TCPWave IPAM to a name-value pair format file.

ARGUMENTS:**--output_file**

Full path to the output file to which DHCP servers are exported.
[mandatory]

--org

Name of the organization from which DHCP servers are to be exported. If this argument is not specified, DHCP servers from all organizations will be exported for FADM user, DHCP servers from user's organization will be exported for non-FADM user.

EXAMPLE USAGE:

```
twc exportdhcpserver --output_file=/tmp/dhcpserver.txt
```

```
twc exportdhcpserver --org=TCPWave --output_file=/tmp/dhcpserver.txt
```

FILE FORMAT:

The output file format is as follows:

Each server starts with a [dhcp-server] section

Each configuration section begins with a [<section name>] field followed by <name>=<value> pairs one per line.

SECTIONS & CONFIGURATION PARAMETERS:

[dhcp-server]

IP_ADDRESS IP Address of the DHCP server
ORGANIZATION_NAME Organization Name of the DHCP server
POLICY_TEMPLATE Policy template name for the DHCP server
APPLIANCE_GROUP Name of the Appliance group to be associated
ENABLE_MONIT '0' to enable monitoring and '1' to disable monitoring
TIME_ZONE Time zone
DESCRIPTION Description of the DHCP Server.

[ntp]

NTP_SERVERS comma separated list of IP addresses of NTP servers

[snmp]

TRAP_SINK_1 IP address of SNMP trap sink
TRAP_SINK_2 IP address of SNMP trap sink
COMMUNITY_STRING Community string for SNMP
SYSTEM_LOCATION System Location
SYSTEM_CONTACT System contact
PROCESS_LIST comma separated list of processes to be monitored. The following is a valid list of processes:
 ntpd, dns, bgpd, zebra, crond, sshd, monit, syslog-ng,dhcpd

[snmpv3users]

ENABLE_SNMPV3 Takes 'true' or 'false' to enable or disable SNMPv3 respectively
SNMPV3_USERS Takes list of JSON objects to define SNMPv3 users. Example
 [{"userName":"Smith","password":"abcd1234","authentication_protocol":"SHA","encryption_protocol":"AES"}]

[macexclusions]

MAC MAC Address to be exclude
DESCRIPTION Description of the MAC address exclusion

[tacacs]

TACACS_PASSKEY TACACS passkey
TACACS_SERVERS Comma separated list of TACACS servers.

SAMPLE FILE CONTENTS:

```
[dhcp-server]
[dhcp-server]
IP_ADDRESS=10.1.10.86
ORGANIZATION_NAME=TCPWave
POLICY_TEMPLATE=policy
APPLIANCE_GROUP=ApplianceGroup1
ENABLE_MONIT=1
TIME_ZONE=GMT (GMT)
DESCRIPTION=
```

```
[ntp]
NTP_SERVERS=192.168.1.1,192.168.1.2,192.168.1.3,192.168.1.4,
[snmp]
TRAP_SINK_1=1.1.1.1
TRAP_SINK_2=1.1.1.2
COMMUNITY_STRING=sph1nkx5
SYSTEM_LOCATION=
SYSTEM_CONTACT=
PROCESS_LIST=ntpd,dns,sshd,monit,syslog-ng,dhcpcd,
[snmpv3users]
ENABLE_SNMPv3=false
SNMPv3_USERS=
[maceclusions]
MAC=04:a1:51:8d:f6:96
DESCRIPTION=Detected as abusive DHCP client
[maceclusions]
MAC=04:a1:51:8d:f6:97
DESCRIPTION=Detected as abusive DHCP client
```

exportdnsacl

NAME:

exportdnsacl

DESCRIPTION:

Exports DNS ACLs from the TCPWave IPAM as a csv list into the specified, output file.

ARGUMENTS:

- output_file
 - Full path to the output file to which ACLs are exported. [mandatory]
- org
 - Name of the organization from which ACLs are to be exported. If this argument is not specified, ACLs from all organizations will be exported for FADM user, ACLs from user's organization will be exported for non-FADM user.

EXAMPLE USAGE:

```
twc exportdnsacl --output_file=/tmp/output.txt
```

```
twc exportdnsacl --org=TCPWave --output_file=/tmp/output.txt
```

EXPORT FILE:

The information in the output file is a comma separated list of fields as specified, in the order below

NAME, DESCRIPTION, ACL

FIELD FORMATS:

ACL is a comma separated list of ACL elements in one of the following formats:

IPAddress/permission (192.168.0.1/Allow)
 ACL-name/permission (internal/Deny)
 IPAddress/mask/permission (192.168.0.0/24/Allow)

EXAMPLE DATA:

"external","external servers","172.0.0.1/24/Allow,172.0.0.2/Deny,internal/Deny"

exportdnsoptiontmpl**NAME:**

`exportdnsoptiontmpl`

DESCRIPTION:

Exports the DNS option templates from the TCPWave IPAM to a name-value pair format file.

ARGUMENTS:**--output_file**

Full path to the output file to which DNS option templates are exported.
 [mandatory]

--org

Name of the organization from which DNS option templates are to be exported. If this argument is not specified, DNS option templates from all organizations will be exported for FADM user, DNS option templates from user's organization will be exported for non-FADM user.

EXAMPLE USAGE:

`tvc exportdnsoptiontmpl --output_file=/tmp/output.txt`

`tvc exportdnsoptiontmpl --org=TCPWave --output_file=/tmp/output.txt`

EXPORT FILE:

The information in the output file is in a format as described below

Each DNS option template starts with a section [dns-option-template] followed by various DNS parameters in the format <param-name>=<param-value> one per each line.

TemplateName is the name of the DNS option template and is mandatory.

SECTIONS & CONFIGURATION PARAMETERS:

TemplateName Name of the DNS option template.

Organization Name of the organization where template is defined.

Description Description for the DNS option template.

Type Takes 'BIND AUTH' or 'BIND CACHE' or 'UNBOUND' or 'NSD' or 'DNS PROXY'

allow-query Defines an address match list of IP address(es) which are allowed to issue queries to the server.

allow-recursion Defines an address match list of IP address(es) which are allowed to issue recursive queries to the server.

allow-transfer Defines an address match list e.g. IP address(es) that are allowed to transfer (copy) the zone information from the server

blackhole Defines an address match list of hosts that the server will NOT respond to, or answer queries for

lame-ttl Defines the number of seconds to cache lame delegations or lame servers, that is, servers which should be authoritative (obtained via a referral or delegation from a parent) but do not respond as authoritative.

max-ncache-ttl Sets the maximum time (in seconds) for which the server will cache negative (NXDOMAIN) answers (positives are defined by max-cache-ttl)

tcp-clients The tcp-clients allows the user to define the maximum number of TCP connections to be supported.

responses-per-second This parameter defines the number of identical responses per second allowed from any given source IP address and lies in the range 0 to 1000.

window Default is no. If set to yes, then the rate limiting function will not be performed will log when the rate-limit function would have been invoked.

transfers-in Only used by slave zones. It determines the number of concurrent inbound zone transfers. Default is 10.

transfers-out Only used by master zones. It determines the number of concurrent outbound zone transfers. Default is 10.

transfers-per-ns Only used by slave zones. It determines the number of concurrent inbound zone transfers for any zone. Default is 2.

directory It is a quoted string defining the absolute path for the server e.g. "/var/named". All subsequent relative paths use this base directory.

statistics-file the pathname of the file the server appends statistics to when instructed to do so using rndc stats. If not specified, the default is named.stats in the server's current directory.

dump-file It is a quoted string defining the absolute path where BIND dumps the database (cache) in response to a rndc dumpdb.

pid-file It is a quoted string which allows to define where the pid (Process Identifier) used by BIND is written.

session-keyfile The pathname of the file into which to write a TSIG session key generated by named for use by nsupdate.

rrset-order It defines the order in which multiple records of the same type are returned.

check-srv-cname If check-integrity is set then fail, warn or ignore SRV records that refer to CNAMES. The default is to warn.

check-mx-cname If check-integrity is set then fail, warn or ignore MX records that refer to CNAMES. The default is to warn.

check-mx Check whether the MX record appears to refer to an IP address. The default is to warn. Other possible values are fail and ignore.

check-names The check-names statement will cause any host name for the zone to be checked for compliance with RFC 952 and RFC 1123 and take the defined action.

recursion If recursion is set to 'yes' the server will always provide recursive query behaviour if requested by the client. If set to 'no' the server will only provide iterative query behaviour - normally resulting in a referral.

empty-zones-enable By default empty-zones-enable is set to yes which means that reverse queries for IPv4 and IPv6 addresses covered by RFCs 1918, 4193, 5737 and 6598 but which is not covered by a locally defined zone clause will automatically return an NXDOMAIN response from the local name server.

listen-on-v6 It turns on BIND to listen for IPv6 queries.

version It specifies the string that will be returned to a version.bind query when using the chaos class only.

dnssec-enable It indicates that a secure DNS service is being used which may be one, or

more, of TSIG, SIG(0) or DNSSEC.

dnssec-validation It indicates that a resolver (a caching or caching-only name server) will attempt to validate replies from DNSSEC enabled (signed) zones.

minimal-responses If yes the server will only add NS resource records to the Authority section and A or AAAA resource records to the Additional sections of a query response when they are required by the protocol, for instance, delegations and negative responses.

zone-statistics If zone-statistics is 'yes', the server will collect statistical data on all zones.

EXAMPLE DATA:

```
[dns-option-template]
TemplateName=BIND AUTH Template
Organization=TCPWave
Description=
Type=BIND AUTH
allow-query=any/Allow;
allow-recursion=any/Allow;
allow-transfer=none/Allow;
blackhole=23259
lame-ttl=0
max-nocache-ttl=60
tcp-clients=500
responses-per-second=0
window=15
transfers-in=10
transfers-out=10
transfers-per-ns=2
directory=/
statistics-file=/var/named/log/named.stats
dump-file=/var/named/log/named_dump.db
pid-file=/var/run/named/named.pid
session-keyfile=/var/run/named/session.key
rrset-order=cyclic
check-srv-cname=ignore
check-mx-cname=ignore
check-mx=ignore
check-names=master ignore,response ignore
recursion=no
empty-zones-enable=no
listen-on-v6=none
version=TCPWave DNS Server
dnssec-enable=yes
dnssec-validation=yes
minimal-responses=yes
zone-statistics=yes
```

exportdnsserver

NAME:

`exportdnsserver`

DESCRIPTION:

Exports the DNS servers from the TCPWave IPAM to a name-value pair format file.

ARGUMENTS:**--output_file**

Full path to the output file to which DNS servers are exported
[mandatory]

--appliance_type

Takes 'auth' or 'cache' as value. If the value is specified, as 'auth' then the command exports all the Authoritative DNS Servers from into the output file. If the value is specified, as 'cache' then the command exports all the Cache DNS Servers into the output file. [mandatory]

--org

Name of the organization from which DNS servers are to be exported. If this argument is not specified, DNS servers from all organizations will be exported for FADM user, DNS servers from user's organization will be exported for non-FADM user.

EXAMPLE USAGE:

```
twc exportdnsserver --output_file=/tmp/auth_dns_server.txt --appliance_type=auth
```

```
twc exportdnsserver --output_file=/tmp/cache_dns_server.txt --appliance_type=cache
```

```
twc exportdnsserver --org=TCPWave --output_file=/tmp/auth_dns_server.txt --appliance_type=auth
```

FILE FORMAT:

The output file format is as follows:

Each server starts with a [dns-server] section

Each configuration section begins with a [<section name>] field followed by <name>=<value> pairs one per line.

SECTIONS & CONFIGURATION PARAMETERS:

[dns-server]

TYPE Takes 'BIND AUTH' or 'BIND CACHE' or 'UNBOUND' or 'NSD' or 'DNS PROXY'

OPTION_TEMPLATE DNS Option template name

SERVER_TEMPLATE DNS server template

IP_ADDRESS IP address of the server

ORGANIZATION_NAME Organization Name of the server

ENABLE_MONIT '0' to enable monitoring and '1' to disable monitoring

INTERNAL_CACHE Applicable for servers of type 'BIND CACHE'. '0' indicates that the server is rooted at an internal root server. '1' indicates that the server is rooted at public internet root server

DMZ_VISIBLE When a cache server is root to a public internet root server '1' indicates visibility of internal zones, '0'

indicates internal zone are not visible. This flag is not applicable for cache servers rooted at an internal root

server
DESCRIPTION DNS server description
TIME_ZONE Time zone

[ntp]
NTP_SERVERS Comma separated list of IP addresses of NTP servers

[snmp]
TRAP_SINK_1 IP address of SNMP trap sink
TRAP_SINK_2 IP address of SNMP trap sink
COMMUNITY_STRING Community string for SNMP
SYSTEM_LOCATION System Location
SYSTEM_CONTACT System contact
PROCESS_LIST comma separated list of processes to be monitored. The following is a valid list of processes:
ntpd, dns, bgpd, zebra, crond, sshd, monit, syslog-ng,dhcpd
[snmpv3users]
ENABLE_SNMPv3 Takes 'true' or 'false' to enable or disable SNMPv3 respectively
SNMPv3_USERS Takes list of JSON objects to define SNMPv3 users. Example
[{"userName":"Smith","password":"abcd1234","authentication_protocol":"SHA","encryption_protocol":"AES"}]
[tacacs]
TACACS_PASSKEY TACACS passkey
TACACS_SERVERS Comma separated list of TACACS servers.

SAMPLE FILE CONTENTS:

```
[dns-server]
TYPE=BIND AUTH
OPTION_TEMPLATE=testdns
SERVER_TEMPLATE=ISC BIND Authoritative Appliance Template
IP_ADDRESS=10.1.10.201
ORGANIZATION_NAME=TCPWave
ENABLE_MONIT=1
DESCRIPTION=Root
TIME_ZONE=America/New_York (Eastern Time)
[ntp]
NTP_SERVERS=17.253.68.253,17.253.16.243,17.253.80.243,17.253.6.243,
[snmp]
TRAP_SINK_1=194.41.67.51
TRAP_SINK_2=194.41.65.177
COMMUNITY_STRING=sph1nkx5
SYSTEM_LOCATION=Datacenter for systematic trading infrastructure
SYSTEM_CONTACT=GNCC +1 877 462 2284
PROCESS_LIST=ntpd,dns,bgpd,zebra,crond,
[snmpv3users]
ENABLE_SNMPv3=false
```

SNMPv3_USERS=

exportdnsservertmpl

NAME:

exportdnsservertmpl

DESCRIPTION:

Exports the DNS server templates from the TCPWave IPAM as a csv list into the specified, output file.

ARGUMENTS:

--output_file

Full path to the output file to which DNS server templates are exported.
[mandatory]

EXAMPLE USAGE:

tvc exportdnsservertmpl --output_file=/tmp/output.txt

EXPORT FILE:

The information in the output file is a comma separated list of fields as specified, in the order below

TYPE, NAME, DESCRIPTION, EMAIL, DYN_UPD, ALGORITHM, LOGGER

FIELD FORMATS:

TYPE is one of the following values: 'BIND AUTH', 'BIND CACHE',
'UNBOUND'

DYN_UPD '1' indicates dynamic updates must be enabled.

'0' indicates dynamic updates are disable

ALGORITHM is a comma separated list of algorithm specification as follows:

Algorithm: Bit_size

Algorithm should be one of the valid TSIG algorithms. Bit_size should be between minimum bit size and maximum bit size specified, for that algorithm

Example: "HMAC-SHA1:150,HMAC-SHA256:200"

LOGGER is a comma separated list of logger specification as follows:

LogCategory/LogChannels

LogChannels is a colon separated list of log channels

Example: "client/default_stderr:default_debug:default_syslog"

EXAMPLE DATA:

"BIND
AUTH","TestBindAuthTemplate","TestBindAuthTemplate","admin@tcpwave.com","1","HMAC-SHA512:512","client/default_stderr:default_debug:default_syslog"

exportdomain

NAME:

exportdomain

DESCRIPTION:

Exports the DNS Domains from the TCPWave IPAM as a csv list into the specified, output file.

ARGUMENTS:

--output_file

Full path to the output file to which domains are to be exported.
[mandatory]

--org

Name of the organization from which domains are to be exported. If this argument is not specified, domains from all organizations will be exported for FADM user, domains from user's organization will be exported for non-FADM user.

EXAMPLE USAGE:

twc exportdomain --output_file=/tmp/output.txt

twc exportdomain --org=TCPWave --output_file=/tmp/output.txt

EXPORT FILE:

The information in the output file is a comma separated list of fields as specified, in the order below

DOMAIN,ORGANIZATION,DESCRIPTION

EXAMPLE DATA:

"tcpwave.com","TCPWave","TCPwave Domain"

exportextension

NAME:

exportextension

DESCRIPTION:

Exports the extended attributes from the TCPWave IPAM as a csv list into the specified, output file.

ARGUMENTS:

--output_file

Full path to the output file to which extended attributes are to be exported. [mandatory]

--org

Name of the organization from which extended attributes are to be exported. If this argument is not specified, extended attributes from all organizations will be exported for FADM user, extended attributes from user's organization will be exported for non-FADM user.

EXAMPLE USAGE:

```
twc exportextension --output_file=/tmp/output.txt
```

```
twc exportextension --org=TCPWave --output_file=/tmp/output.txt
```

EXPORT FILE:

The information in the output file is a comma separated list of fields as specified, in the order below

```
"NAME","DESCRIPTION","EXTENSION_TYPE","CONSTRAINTS","LOW","HIGH","IS_UNIQUE","FLAG"
```

EXAMPLE DATA:

```
"NAME","DESCRIPTION","EXTENSION_TYPE","CONSTRAINTS","LOW","HIGH","IS_UNIQUE","FLAG"
```

```
"nextension","","STRING",\N,\N,\N,\N,\N
```

```
"MPLList","","LIST","LIST_VALUES=domain,network,subnet",\N,\N,\N,\N
```

```
"subbaiah","","STRING",\N,\N,\N,\N,\N
```

```
"First_Seen_In_Cloud","DO NOT DELETE THIS ATTRIBUTE OR EDIT THE VALUE OF IT.",STRING",\N,\N,\N,\N,"1
```

```
"Last_Seen_In_Cloud","DO NOT DELETE THIS ATTRIBUTE OR EDIT THE VALUE OF IT.",STRING",\N,\N,\N,\N,"1
```

```
"testxtn","","STRING",\N,\N,\N,\N,\N
```

```
"Subnet_ext","test","STRING","mandatory",\N,\N,\N,\N
```

exportipamappliance**NAME:**

exportipamappliance

DESCRIPTION:

Exports the eIPAM appliances from the TCPWave IPAM as a csv list into the specified, output file.

ARGUMENTS:

--output_file

Full path to the output file to which IPAM appliances are to be exported.
[mandatory]

EXAMPLE USAGE:

```
twc exportipamappliance --output_file=/tmp/output.txt
```

SECTIONS & CONFIGURATION PARAMETERS:

[ipam-appliance]

NAME	Name of the IPAM Appliance.
------	-----------------------------

IP_ADDRESS IP Address of the IPAM Appliance.
 TYPE Type of the IPAM Appliance.
 BANNER_COLOR Color of the banner in the Appliance.
 BANNER_TITLE Title of the banner in the Appliance.
 DESCSCRIPTION Description for the IPAM Appliance.

[ntp]
 NTP_SERVERS comma separated list of IP addresses of NTP servers

[snmp]
 TRAP_SINK_1 IP address of SNMP trap sink
 TRAP_SINK_2 IP address of SNMP trap sink
 COMMUNITY_STRING Community string for SNMP
 SYSTEM_LOCATION System Location
 SYSTEM_CONTACT System contact
 PROCESS_LIST comma separated list of processes to be monitored. The following is a valid list of processes:
 ntpd, dns, bgpd, zebra, crond, sshd, monit, syslog-ng, dhcpcd

[snmpv3users]
 ENABLE_SNMPV3 Takes 'true' or 'false' to enable or disable SNMPv3 respectively
 SNMPv3_USERS Takes list of JSON objects to define SNMPv3 users. Example
 [{"userName":"Smith","password":"abcd1234","authentication_protocol":"SHA","encryption_protocol":"AES"}]

[tacacs]
 ENABLE_TACACS Takes '0' or '1'. '1' indicates TACACS+ configuration should be enabled for this server. '0' indicates TACACS+ configuration should be disabled
 TACACS_PASSKEY TACACS passkey
 TACACS_SERVERS Comma separated list of TACACS servers.

SAMPLE OUTPUT FILE CONTENTS:

```
[ipam-appliance]
NAME=COMMON-DEVELOPER-IPAM
IP_ADDRESS=10.1.10.240
TYPE=Production
BANNER_COLOR=red
BANNER_TITLE=This is a Production Server. All actions are audited. Please do not make any unauthorized changes.
DESCSCRIPTION=
[ntp]
NTP_SERVERS=172.253.172.253,172.253.172.254,
[snmp]
TRAP_SINK_1=10.1.10.1
TRAP_SINK_2=10.1.10.2
COMMUNITY_STRING=E61B8541B79BC35E8E5BEFBE908EB050
```

```
SYSTEM_LOCATION=
SYSTEM_CONTACT=
PROCESS_LIST=ntpd,sshd,de,cli,search,tims,timsscheduler,mysql,
[snmpv3users]
ENABLE_SNMPv3=false
SNMPv3_USERS=
```

exportipv6object

NAME

exportipv6object

DESCRIPTION

Exports the IPv6 objects from the TCPWave IPAM as a csv list into the specified output file.

ARGUMENTS

--output_file

Full path to the output file to which IPv6 objects are to be exported. [mandatory]

--org

Name of the organization from which IPv6 objects are to be exported.

EXAMPLE:

```
twc exportipv6object --output_file=/tmp/output.txt
```

```
twc exportipv6object --org=TCPWave --output_file=/tmp/output.txt
```

EXPORT FILE:

The information in the output file is a comma separated list of fields as specified in the order below

"ADDRESS","ORGANIZATION","NAME","DOMAIN","OBJECT_TYPE","ALLOCATION_TYPE","MAC","OPTION_TEMPLATE","TTL","NS_A","NS_PTR","DDNS_A","DDNS_PTR","DDNS_CNAME","DDNS_MX","CONTACT_FIRST_NAME","CONTACT_MIDDLE_NAME","CONTACT_LAST_NAME","CONTACT_EMAIL","DESCRIPTION","MANAGED_BY","MONITORED_BY","CHANGE_CONTROL_TICKET","TERMINAL_SERVER_KVM","END_OF_LIFE","ROOM","FLOOR"

EXAMPLE DATA:

"6002::12","Internal","Access00001Router","tcp.com","Access
Router","1","","","","1200","1","1","1","1","1","","","","","","6002::12","","","","","","","","","

exportipv6objectrr

NAME:

exportipv6objectrr

DESCRIPTION:

Exports the IPv6 objects resource records from the TCPWave IPAM as a csv list into the specified, output file.

ARGUMENTS:

--output_file

Full path to the output file to which IPv6 objects resource records are exported. [mandatory]

--org

Name of the organization from which IPv6 objects resource records are to be exported. If this argument is not specified, IPv6 objects resource records from all organizations will be exported for FADM user, IPv6 objects resource records from user's organization will be exported for non-FADM user.

EXAMPLE USAGE:

twc exportipv6objectrr --output_file=/tmp/output.txt

twc exportipv6objectrr --org=TCPWave --output_file=/tmp/output.txt

EXPORT FILE:

The information in the output file is a comma separated list of fields as

specified, in the order below

"IP_ADDRESS","ORGANIZATION_NAME","OWNER","TTL","CLASS","TYPE","DATA"

EXAMPLE DATA:

"6001::2","TCPWave","www.tcpwave.com.", "600","IN","AAAA","6001::2"

"6001::2","TCPWave","dev.tcpwave.com.", "300","IN","CNAME","www.tcpwave.com."

exportipv6subnet

NAME

exportipv6subnet

DESCRIPTION

Exports the IPv6 subnets from the TCPWave IPAM as a csv list into the specified output file.

ARGUMENTS

--output_file

Full path to the output file to which IPv6 subnets are exported. [mandatory]

--org

Name of the organization from which IPv6 subnets are to be exported. If this argument is not specified, IPv6 subnets from all organizations will be exported for FADM user, IPv6 subnets from user's organization will be exported for non-FADM user.

EXAMPLE:

twc exportipv6subnet --output_file=/tmp/output.txt

twc exportipv6subnet --org=TCPWave --output_file=/tmp/output.txt

IMPORT FILE:

The information in the output file is a comma separated list of fields as specified in the order below

"ADDRESS","MASK","BLOCK_ADDRESS","NAME","ORGANIZATION","DOMAIN","SUBNET_GROUP","ROUTER_OUTER","DHCP_TEMPLATE_NAME","PRIMARY_DHCP_SERVER","STREET_1","STREET_2","CITY","STATE","COUNTRY","ZIP","DESC","CONTACT_F_NAME","CONTACT_M_NAME","CONTACT_L_NAME","CONTACT_EMAIL","VLAN","VRF","DISCOVERY_TEMPLATE"

EXAMPLE DATA:

"6001:10::","27","6001::","Internal-463-
498","Internal","inter.com"localhostlocalhostlocalhostlocalhostlocalhostlocalhost

exportipv6subnetgroup

NAME:

exportipv6subnetgroup

DESCRIPTION:

Exports the IPv6 subnet groups from the TCPWave IPAM as a csv list into the specified, output file.

ARGUMENTS:

--output_file

Full path to the output file to which IPv6 subnet groups are to be exported. [mandatory]

--org

Name of the organization from which IPv6 subnet groups are to be exported. If this argument is not specified, IPv6 subnet groups from all organizations will be exported for FADM user, IPv6 subnet groups from user's organization will be exported for non-FADM user.

EXAMPLE USAGE:

```
twc exportipv6subnetgroup --output_file=/tmp/output.txt
```

```
twc exportipv6subnetgroup --org=TCPWave --output_file=/tmp/output.txt
```

EXPORT FILE:

The information in the output file is a comma separated list of fields as specified, in the order below

"NAME","ORG_NAME","DESCRIPTION"

EXAMPLE DATA:

"NAME","ORG_NAME","DESCRIPTION"
"v6Sub_gr1","Internal",""
"v6sg","EARTH","test"

exportipv6block**NAME**

exportipv6block

DESCRIPTION

Exports the IPv6 block from the TCPWave IPAM as a csv list into the specified output file.

ARGUMENTS

--output_file

Full path to the output file to which IPv6 pool are exported. [mandatory]

--org

Name of the organization from which IPv6 blocks are to be exported. If this argument is not specified, IPv6 blocks from all organizations

EXAMPLE

```
twc exportipv6block --output_file=/tmp/output.txt
```

```
twc exportipv6block --org=TCPWave --output_file=/tmp/output.txt
```

EXPORT FILE

The information in the output file is a comma separated list of fields as specified in the order below

"ADDRESS","MASK","POOL_ADDRESS","NAME","ORG_NAME","DNSSEC","NSEC_OPT","ZONE_TEMPLATE","DMZ_VISIBLE","DESCRIPTION","CONTACT_F_NAME","CONTACT_M_NAME","CONTACT_L_NAME","CONTACT_EMAIL","CLOUD_REGION","DISCOVERY_TEMPLATE","VRF"

EXAMPLE DATA

```
"8001::","24","8000::","Test","Internal","0","NSEC3","","0","","","","","","","","","","","",""
```

exportipv6pool

NAME

exportipv6pool

DESCRIPTION

Exports the IPv6 pool from the TCPWave IPAM as a csv list into the specified output file.

ARGUMENTS

--output_file

Full path to the output file to which IPv6 pool are exported. [mandatory]

--org

Name of the organization from which IPv6 pools are to be exported. If this argument is not specified, IPv6 pools from all organizations

EXAMPLE

```
twc exportipv6pool --output_file=/tmp/output.txt
```

```
twc exportipv6pool --org=TCPWave --output_file=/tmp/output.txt
```

EXPORT FILE

The information in the output file is a comma separated list of fields as specified in the order below

"IP_ADDR","MASK","ORG_NAME","REGION","NAME","DNSSEC","NSEC_OPT","ZONE_TEMPLATE","DMZ_VISIBLE","DESCRIPTION","CONTACT_F_NAME","CONTACT_M_NAME","CONTACT_L_NAME","CONTACT_EMAIL","CLOUD_REGION","DISCOVERY_TEMPLATE","VRF"

EXAMPLE DATA

"8001::", "48", "TCPWave", "USA", "pool1", "0", "", "", "0", "TCPWave	IPv6	address
pool", "", "", "", "", "", "", "		

exportipv6reversezone

NAME

`exportipv6reversezone`

DESCRIPTION

Exports the DNS IPv6 reverse zones from the TCPWave IPAM in CSV format into the specified output file.

ARGUMENTS

`--output_file`

Full path of the output file to which zones must be exported. [mandatory]

`--org`

Name of the organization from which DNS IPv6 reverse zones have to be exported. If this argument is not specified, DNS IPv6 reverse zones from all organizations will be exported for FADM user whereas DNS IPv6 reverse zones from user's organization will be exported for non-FADM user.

EXAMPLE

`twc exportipv6reversezone --output_file=/tmp/output.txt`

```
twc exportipv6reversezone --org=TCPWave --output_file=/tmp/output.txt
```

EXPORT FILE

The rows in the output file are comma separated values of fields as specified in the order below

```
"NAME","ORG_NAME","TMPL_NAME","DNSSEC","NSEC_OPT","MONIT","DESCRIPTION","DMZ_VISIB  
LE","CONTACT_F_NAME","CONTACT_M_NAME","CONTACT_L_NAME","CONTACT_EMAIL","ADDRES  
S","MASK_LENGTH"
```

FIELD FORMATS

DNSSEC '1' indicates that DNSSEC should be enabled for the zone. '0' indicates that DNSSEC is not enabled
NSEC_OPT 'NSEC' or 'NSEC3'

MONIT '1' indicates monitoring is enabled for this zone. '0' indicates monitoring is disabled for this zone.

EXAMPLE DATA

```
"0.0.0.0.0.5.ip6.arpa.", "Internal", "", "0", "NSEC3", "1", "", "0", "", "", "", "5000::", "24"  
exportipv6dnsserver
```

NAME

exportipv6dnsserver

DESCRIPTION

Exports the IPv6 DNS appliances from the TCPWave IPAM to a name-value pair format file.

ARGUMENTS

--output_file

Full path to the output file to which DNS appliances are exported [mandatory]

--appliance_type

Takes 'auth' or 'cache' as value. If the value is specified as 'auth' then the command exports all the Authoritative DNS appliances from into the output file. If the value is specified as 'cache' then the command exports all the Cache DNS appliances into the output file. [mandatory]

--org

Name of the organization from which DNS appliances are to be exported. If this argument is not specified, DNS appliances from all organizations will be exported for FADM user, DNS servers from user's organization will be exported for non-FADM user.

EXAMPLE

```
twc exportipv6dnsserver --output_file=/tmp/auth_dns_server.txt --appliance_type=auth
```

```
twc exportipv6dnsserver --output_file=/tmp/cache_dns_server.txt --appliance_type=cache
```

```
twc exportipv6dnsserver --org=TCPWave --output_file=/tmp/auth_dns_server.txt --appliance_type=auth
```

FILE FORMAT

The output file format is as follows:

Each server starts with a [dns-server] section

Each configuration section begins with a [<section name>] field followed by <name>=<value> pairs one per line.

SECTIONS & CONFIGURATION PARAMETERS

[dns-server]

TYPE Takes 'BIND AUTH' or 'BIND CACHE' or 'UNBOUND' or 'NSD' or 'DNS PROXY'

OPTION_TEMPLATE DNS Option template name

SERVER_TEMPLATE DNS server template

IPV6_ADDRESS IPv6 address of the appliance

ORGANIZATION_NAME Organization Name of the appliance

ENABLE_MONIT '0' to enable monitoring and '1' to disable monitoring

INTERNAL_CACHE Applicable for appliances of type 'BIND CACHE'. '0' indicates that the appliance is rooted at an internal root server. '1' indicates that the server is rooted at public internet root appliance

NSM_TEMPLATE NSM Template is applicable if the selected appliance type is ISC BIND Cache appliance, Unbound Cache Appliance, or recursion enabled ISC BIND Authoritative appliance.

DESCRIPTION DNS appliance description

TIME_ZONE Time zone

[ntp]

IPV6_NTP_SERVERS Comma separated list of IP addresses of NTP appliances

[snmp]

IPV6_TRAP_SINK_1 IPv6 address of SNMP trap sink
IPV6_TRAP_SINK_2 IPv6 address of SNMP trap sink
COMMUNITY_STRING Community string for SNMP
SYSTEM_LOCATION System Location
SYSTEM_CONTACT System contact
PROCESS_LIST comma separated list of processes to be monitored. The following is a valid list of processes:
ntpd, dns, bgpd, zebra, crond, sshd, monit, syslog-ng,dhcpd

[snmpv3users]

ENABLE_SNMPV3 Takes 'true' or 'false' to enable or disable SNMPv3 respectively
SNMPv3_USERS Takes list of JSON objects to define SNMPv3 users. Example

`[{"userName":"Smith","password":"abcd1234","authentication_protocol":"SHA","encryption_protocol":"AES"}]`

[tacacs]

TACACS_PASSKEY TACACS passkey
IPV6_TACACS_SERVERS Comma separated list of TACACS appliances.

[syslogng-global-options]

TIME_REOPEN The time to wait in seconds before a dead connection is reestablished. Takes a value less than or equal to 32767.

TIME_REAP If no new messages are written to a destination within the specified time in seconds, the connection will be closed, and its state will be freed. Takes a value less than or equal to 32767.

FLUSH_LINES Specifies how many lines are flushed to a destination at a time. Takes a value less than or equal to 32767.

STATS_FREQ Syslog-NG OSE periodically sends a log statistics message. Takes a value less than or equal to 32767.

LOG_FIFO_SIZE The number of messages that the output queue can store. Takes a value less than or equal to 32767.

LOG_MSG_SIZE The maximal length of the log messages is limited by this option. It is not recommended to set the option value higher than 10 MiB. Takes a value less than or equal to

32767.

KEEP_TIMESTAMP Specifies whether syslog-ng should accept the timestamp received from the sending application or client.
Takes value 'Yes' or 'No'.

[syslogng-source]

SOURCE_NAME Name of the Source
INTERNAL_MSG Internal syslog-NG message, takes input values as '0' or '1'.
default value is '1'.
SYSTEM_MSG System specific log message, takes input value as '0' or '1'.
MSG_TXT_FILE Message from text file, takes the file name as input.
MSG_MULTI_TXT_FILE Message from multiple text files, takes input '0' or '1'.
if this flag is '1' need to specify the **FILE_PATH** and
FILE_PATTERN.
FILE_PATH File patch to the multiple text file.
FILE_PATTERN File Name pattern.
SYSLOG_SERVER Syslog-NG sever, takes the input as '0' or '1'.
IP_ADDRESS IP address of the syslog server.
PORT Port number of the syslog server.
NETWORK_PROTOCOL Network protocol, supports 'UDP' and 'TCP'.

[syslogng-filter]

FILTER_NAME Name of the Filter.
CONDITION Takes the input as 'complex' or 'simple'.
FACILITIES Allow values are one or more comma separated option given below.
auth, authpriv, cron, deamon, kern, lpr, mail,mark, news,
syslog, user, uucp, local0, local1,local2, local3,
local4, local5, local6, local7.
PRIORITIES Allow values are one or more comma separated option given below.
info, notice, warning, err, crit, alert, emerg.
HOST_NAME Name of the host.
IP_NETWORK IP address with mask length.
MATCH_EXPRESSION Match expression.
PROGRAM Program.

[syslogng-destination]

DESTINATION_NAME Name of the destination.
TYPE_SNG Type of the destination. takes the value between 1 to 5.
'1'= File
'2'= Named pipe
'3'= Local Users
'4'= All logged-in users
'5'= Syslog server.

LOG_FILE_NAME File name to log the message, mandatory when TYPE_SNG is specified as '1'.
NAMED_PIPE_NAME Named pipe name, mandatory when TYPE_SNG is specified as '2'.
LOCAL_USERS Local users, mandatory when TYPE_SNG is specified as '3'.
SYSLOG_SERVER IP address of the syslog server, mandatory when TYPE_SNG is specified as '5'.
PORT Port number of the syslog server, mandatory when TYPE_SNG is specified as '5'.
NETWORK_PROTOCOL Network protocol, supports 'UDP' and 'TCP', mandatory when TYPE_SNG is specified as '5'.

[syslogng-target]

SOURCE Name of the source.
FILTER Name of the filter.
DESTINATION Name of the destination.

SAMPLE FILE CONTENTS:

[dns-server]

TYPE=BIND AUTH
OPTION_TEMPLATE=BIND AUTH Default Template
SERVER_TEMPLATE=BIND AUTH Default Server Template
IPV6_ADDRESS=2000::3
ORGANIZATION_NAME=TCPWave
ENABLE_MONIT=1
DESCRIPTION>Edit
TIME_ZONE=GMT (GMT)
ENABLE_RECURSION=no
INTERNAL_CACHE=0

[ntp]

IPV6_NTP_SERVERS=5000::3,5000::4,

[dns_resolver]

IPV6_NAME_APPLIANCES=4000::3
SEARCH_SUFFIXES=

[snmp]

IPV6_TRAP_SINK_1=1000::1
IPV6_TRAP_SINK_2=1000::2
COMMUNITY_STRING=sph1nkx5

```
SYSTEM_LOCATION=
SYSTEM_CONTACT=
PROCESS_LIST=ntpd,dns,sshd,monit,syslog-ng,dhcpd,
```

```
[snmpv3users]
```

```
ENABLE_SNMPv3=false
SNMPv3_USERS=[]
```

```
[syslogng-options]
```

```
TIME_REOPEN=60
TIME_REAP=60
FLUSH_LINES=60
STATS_FREQ=600
LOG_FIFO_SIZE=1000
LOG_MSG_SIZE=2048
KEEP_TIMESTAMP=Yes
```

```
[syslogng-source]
```

```
SOURCE_NAME=s_sys
INTERNAL_MSG=YES
SYSTEM_MSG=YES
```

```
[syslogng-filter]
```

```
FILTER_NAME=f_default
CONDITION=complex
COMPLEX_CONDITION=level(info..emerg) and not (facility(mail) or facility(authpriv) or facility(cron))
```

```
[syslogng-filter]
```

```
FILTER_NAME=f_cron
CONDITION=complex
COMPLEX_CONDITION=facility(cron)
```

```
[syslogng-destination]
```

```
DESTINATION_NAME=d_mesg
TYPE_SNG=File
LOG_FILE_NAME=messages
ENABLE_SYNC=Yes
```

```
[syslogng-destination]
```

```
DESTINATION_NAME=d_cron
TYPE_SNG=File
```

```
LOG_FILE_NAME=cron
ENABLE_SYNC=Yes
```

```
[syslogng-target]
```

```
SOURCE=s_sys
FILTER=f_default
DESTINATION=d_mesg
```

```
[syslogng-target]
```

```
SOURCE=s_sys
FILTER=f_cron
DESTINATION=d_cron
exportipv6dhcpserver
```

NAME

exportipv6dhcpserver

DESCRIPTION

Exports the IPv6 DHCP servers from the TCPWave IPAM to a name-value pair format file.

ARGUMENTS

```
--output_file
```

Full path to the output file to which IPv6 DHCP servers are exported. [mandatory]

```
--org
```

Name of the organization from which IPv6 DHCP servers are to be exported. If this argument is not specified, DHCP servers from all organizations will be exported for FADM user, DHCP servers from user's organization will be exported for non-FADM user.

EXAMPLE

```
twc exportipv6dhcpserver --output_file=/tmp/dhcpserver.txt
```

```
twc exportipv6dhcpserver --org=TCPWave --output_file=/tmp/dhcpserver.txt
```

FILE FORMAT

The output file format is as follows:

Each server starts with a [dhcp-server] section

Each configuration section begins with a [<section name>] field followed

by <name>=<value> pairs one per line.

SECTIONS & CONFIGURATION PARAMETERS

[dhcp-server]

IP_ADDRESS IPv6 Address of the DHCP server
ORGANIZATION_NAME Organization Name of the DHCP server
POLICY_TEMPLATE Policy template name for the DHCP server
USAGE Name of the Appliance group to be associated
ENABLE_MONIT '0' to enable monitoring and '1' to disable monitoring
TIME_ZONE Time zone
DHCPD_INTERFACES Specify DHCPD Interfaces.
DESCRIPTION Description of the DHCP Server.
DISCOVERY '0' to enable discovery and '1' to disable discovery

[ntp]

NTP_SERVERS comma separated list of IP addresses of NTP servers

[dns_resolver]

NAME_APPLIANCES IP address of Name Appliances
SEARCH_SUFFIXES Specify search suffixes

[snmpv3users]

ENABLE_SNMPV3 Takes 'true' or 'false' to enable or disable SNMPv3 respectively
SNMPV3_USERS Takes list of JSON objects to define SNMPv3 users. Example

`[{"userName":"Smith","password":"abcd1234","authentication_protocol":"SHA","encryption_protocol":"AES"}]`

[syslogng-options]

TIME_REOPEN It is the time to wait in seconds before a dead connection is reestablished, and the default value is 60.

TIME_REAP It is the time to wait before closing idle connections. The default value is 60.

FLUSH_LINES It specifies the number of lines flushed to a destination at a time. The default value is 60.

STATS_FREQ It is the time to wait between statistics messages in seconds. The default value is 60.

LOG_FIFO_SIZE It specifies the number of messages that the output queue can store. The default value is 1000.

LOG_MSG_SIZE It specifies the maximum length of a message in bytes.

KEEP_TIMESTAMP It specifies whether Syslog-ng should accept the timestamp received from the sending application or client.

The default value is Yes.

[syslogng-source]

SOURCE_NAME Enter the Source Name of the Syslog-NG sources.

INTERNAL_MSG By default, this field is enabled. All messages generated internally by Syslog-NG use the source driver internal().

SYSTEM_MSG Syslog-NG automatically collects the system-specific log messages of the host on several platforms using the system() driver.

[syslogng-filter]

FILTER_NAME Enter the name of the Syslog_NG filter.

CONDITION Simple or Complex.

COMPLEX_CONDITION On selecting, complex condition, you must provide a valid filter string to add to the configuration file.

[syslogng-filter]

FILTER_NAME Enter the name of the Syslog_NG filter.

CONDITION Simple or Complex.

COMPLEX_CONDITION On selecting, complex condition, you must provide a valid filter string to add to the configuration file.

[syslogng-destination]

DESTINATION_NAME Enter the Name of the Destination file.

TYPE_SNG Type of SNG.

LOG_FILE_NAME Name of the log file.

ENABLE_SYNC Specify 'yes' or 'no' to enable sync.

[syslogng-destination]

DESTINATION_NAME Enter the Name of the Destination file.

TYPE_SNG Type of SNG.

LOG_FILE_NAME Name of the log file.

ENABLE_SYNC Specify 'yes' or 'no' to enable sync.

[syslogng-target]

SOURCE Name of the destination file.

FILTER Select a value from the dropdown.

DESTINATION Enter the log file path.

[syslogng-target]

SOURCE Name of the destination file.

FILTER Select a value from the dropdown.

DESTINATION Enter the log file path.

SAMPLE FILE CONTENTS

[dhcp-server]

```
IP_ADDRESS=9000::  
ORGANIZATION_NAME=EARTH  
POLICY_TEMPLATE=DEF-POLICY  
USAGE=Primary  
ENABLE_MONIT=1  
TIME_ZONE=GMT (GMT)  
DHCPD_INTERFACES=  
DESCRIPTION=Testing  
DISCOVERY=0
```

[ntp]

```
NTP_SERVERS=5000::2,
```

[dns_resolver]

```
NAME_APPLIANCES=5000::2  
SEARCH_SUFFIXES=
```

[snmpv3users]

```
ENABLE_SNMPv3=false  
SNMPv3_USERS=[]
```

[syslogng-options]

```
TIME_REOPEN=60  
TIME_REAP=60
```

```
FLUSH_LINES=60
STATS_FREQ=600
LOG_FIFO_SIZE=1000
LOG_MSG_SIZE=65536
KEEP_TIMESTAMP=Yes
```

```
[syslogng-source]
```

```
SOURCE_NAME=s_sys
INTERNAL_MSG=YES
SYSTEM_MSG=YES
```

```
[syslogng-filter]
```

```
FILTER_NAME=f_default
CONDITION=complex
COMPLEX_CONDITION=level(info..emerg) and not (facility(mail) or facility(authpriv) or
facility(cron))
```

```
[syslogng-filter]
```

```
FILTER_NAME=f_cron
CONDITION=complex
COMPLEX_CONDITION=facility(cron)
```

```
[syslogng-destination]
```

```
DESTINATION_NAME=d_mesg
TYPE_SNG=File
LOG_FILE_NAME=messages
ENABLE_SYNC=Yes
```

[syslogng-destination]

```
DESTINATION_NAME=d_cron  
TYPE_SNG=File  
LOG_FILE_NAME=cron  
ENABLE_SYNC=Yes
```

[syslogng-target]

```
SOURCE=s_sys  
FILTER=f_default  
DESTINATION=d_mesg
```

[syslogng-target]

```
SOURCE=s_sys  
FILTER=f_cron  
DESTINATION=d_cron
```

exportlocation

NAME:

exportlocation

DESCRIPTION:

Exports the locations from the TCPWave IPAM as a csv list into the specified, output file.

ARGUMENTS:

--output_file

Full path to the output file to which locations are exported. [mandatory]

--org

Name of the organization from which locations are to be exported. If this argument is not specified, locations from all organizations will be exported for FADM user, locations from user's organization will be exported for non-FADM user.

EXAMPLE USAGE:

```
twc exportlocation --output_file=/tmp/output.txt
```

```
twc exportlocation --org=TCPWave --output_file=/tmp/output.txt
```

EXPORT FILE:

The information in the output file is a comma separated list of fields as specified, in the order below

```
"STREET1","STREET2","CITY","STATE","ZIP","COUNTRY","ORG_NAME"
```

EXAMPLE DATA:

```
"600 ALEXANDER ROAD","","PRINCETON","NJ","08540","USA","TCPWave"
```

exportlogchannel**NAME:**

exportlogchannel

DESCRIPTION:

Exports the DNS Log Channels from the TCPWave IPAM as a csv list into the specified, output file.

ARGUMENTS:

--output_file

Full path to the output file to which log channels are exported.
[mandatory]

--org

Name of the organization from which DNS Log Channels are to be exported.
If this argument is not specified, DNS Log Channels from all organizations will be exported for FADM user, DNS Log Channels from user's organization will be exported for non-FADM user.

EXAMPLE USAGE:

```
twc exportlogchannel --output_file=/tmp/output.txt
```

```
twc exportlogchannel --org=TCPWave --output_file=/tmp/output.txt
```

EXPORT FILE:

The information in the output file is a comma separated list of fields as specified, in the order below

```
NAME, TYPE, FILE_PATH, VERSION, SIZE, FACILITY, SEVERITY, DEBUG_LEVEL, PRINT_TIME,  
PRINT_SEVERITY, PRINT_CATEGORY
```

FIELD FORMATS:

NAME is the name of the DNS Log Channel

TYPE takes 'FILE', 'SYSLOG', 'STDERR' or 'NULL'

FILE_PATH a valid file path when TYPE is 'FILE'

VERSION is a valid integer indicating the maximum number of log file versions to be retained on disk during log rotation before purging the oldest log file.

SIZE is a valid integer in bytes indicating the maximum size of a log file before a new log file is created during log rotation.

FACILITY is the facility name when TYPE is 'SYSLOG'

SEVERITY takes 'dynamic', 'debug', 'info', 'notice', 'warning', 'error' or 'critical'.

DEBUG_LEVEL is a valid integer indicating the debug level when SEVERITY is specified, as 'debug'.

PRINT_TIME, PRINT_SEVERITY, PRINT_CATEGORY takes '0' or '1'

EXAMPLE DATA:

```
"NAME","TYPE","FILE_PATH","VERSION","SIZE","FACILITY","SEVERITY","DEBUG_LEVEL","PRINT_TIME","PRINT_SEVERITY","PRINT_CATEGORY"  
"queries","FILE","/var/named/log/query.log","6","1024000","","dynamic","","0","0","0"  
"named","FILE","/var/named/log/named.log","6","1024000","","dynamic","","1","0","1"
```

exportnetwork

NAME:

exportnetwork

DESCRIPTION:

Exports the networks from the TCPWave IPAM as a csv list into the specified, output file.

ARGUMENTS:

--output_file

Full path to the output file to which networks are exported. [mandatory]

--org

Name of the organization from which networks are to be exported. If this argument is not specified, networks from all organizations will be exported for FADM user, networks from user's organization will be exported for non-FADM user.

EXAMPLE USAGE:

```
tvc exportnetwork --output_file=/tmp/output.txt
```

```
tvc exportnetwork --org=TCPWave --output_file=/tmp/output.txt
```

EXPORT FILE:

The information in the output file is a comma separated list of fields as specified, in the order below

IP_ADDR,MASK,ORG_NAME,NAME,EMAIL,PERCENTAGE_FULL,EMAIL_CHECK,SNMP_CHECK,L
OG_CHECK,DESCRIPTION

EXAMPLE DATA:

"60.0.0.0","24","TCPWave","TCPWave
Network","admin@tcpwave.com","30","0","1","0","TCPWave Network"

exportobject

NAME:

exportobject

DESCRIPTION:

Exports the objects from the TCPWave IPAM as a csv list into the specified,
output file.

ARGUMENTS:

--output_file

Full path to the output file to which objects are to be exported.
[mandatory]

--subnet

IP address of the subnet. If this argument is specified, objects within
specified, subnet will be exported.

--org

Name of the organization from which objects are to be exported. If this
argument is not specified, objects from all organizations will be
exported for FADM user, objects from user's organization will be
exported for non-FADM user.

EXAMPLE USAGE:

twc exportobject --output_file=/tmp/output.txt

twc exportobject --output_file=/tmp/output.txt --subnet=192.168.1.0

twc exportobject --org=TCPWave --output_file=/tmp/output.txt

EXPORT FILE:

The information in the output file is a comma separated list of fields as
specified, in the order below

"ADDRESS","ORGANIZATION","NAME","DOMAIN","OBJECT_TYPE","ALLOCATION_TYPE","MAC",
"OPTION_TEMPLATE","TTL","NS_A","NS_PTR","DDNS_A","DDNS_PTR","DDNS_CNAME","DDNS_
MX","CONTACT_FIRST_NAME","CONTACT_MIDDLE_NAME","CONTACT_LAST_NAME","CONTAC
T_EMAIL","DESCRIPTION","MANAGED_BY","MONITORED_BY","CHANGE_CONTROL_TICKET","T
ERMINAL_SERVER_KVM","END_OF_LIFE","ROOM","FLOOR"

EXAMPLE DATA:

90.0.0.1,TCPWave,"Server-2733663",tcpwave.com,"3G Phone",4,01:23:45:67:89:ab,"Generic-
template",30,0,0,1,0,1,0,John,Francis,Smith,john.smith@tcpwave.com,"TCPwave Internal
Server","","","","","","","","","

exportobjecttype**NAME:**

exportobjecttype

DESCRIPTION:

Exports the object types from the TCPWave IPAM as a csv list into the specified, output file.

ARGUMENTS:

--output_file

Full path to the output file to which object types are to be exported.
[mandatory]

EXAMPLE USAGE:

```
tvc exportobjecttype --output_file=/tmp/output.txt
```

EXPORT FILE:

The information in the output file is a comma separated list of fields as specified, in the order below

```
"CODE","DESCRIPTION","LOGO_PATH","USER_DEFINED","PREFIX","SUFFIX","INITIAL_SEQUENCE_NUMBER","PREFIX_ZEROS"
```

EXAMPLE DATA:

```
"CODE","DESCRIPTION","LOGO_PATH","USER_DEFINED","PREFIX","SUFFIX","INITIAL_SEQUENCE_NUMBER","PREFIX_ZEROS"  
"3G Phone","Smart Phone","","N","3G","Phone","1","Yes"  
"Access Router","A Multiservice Router","","N","Access","Router","1","Yes"  
"Audio MCU","Audio MCU bridges for IP and ISDN-based  
videoconferencing","","N","Audio","MCU","1","Yes"  
"Bridge","A Generic Bridge","","N","Bri","dge","1","Yes"
```

exportobjectrr**NAME:**

exportobjectrr

DESCRIPTION:

Exports the objects resource records from the TCPWave IPAM as a csv list into the specified, output file.

ARGUMENTS:

--output_file

Full path to the output file to which objects resource records are exported. [mandatory]

--org

Name of the organization from which objects resource records are to be exported. If this argument is not specified, objects resource records from all organizations will be exported for FADM user, objects resource

records from user's organization will be exported for non-FADM user.

EXAMPLE USAGE:

```
twc exportobjectrr --output_file=/tmp/output.txt
```

```
twc exportobjectrr --org=TCPWave --output_file=/tmp/output.txt
```

EXPORT FILE:

The information in the output file is a comma separated list of fields as specified, in the order below

"IP_ADDRESS","ORGANIZATION_NAME","OWNER","TTL","CLASS","TYPE","DATA"

EXAMPLE DATA:

```
"10.1.10.14","TCPWave","www.tcpwave.com.", "600", "IN", "A", "10.1.10.14"
```

```
"10.1.10.14","TCPWave","dev.tcpwave.com.", "300", "IN", "CNAME", "www.tcpwave.com."
```

exportorg**NAME:**

exportorg

DESCRIPTION:

Exports the organizations and its components from the TCPWave IPAM to a zip file.

ARGUMENTS:**--org**

Name of the organization from which organization and its components are to be exported. If this argument is not specified, components from all the organizations will be exported.

EXAMPLE USAGE:

```
twc exportorg
```

```
twc exportorg --org=TCPWave
```

EXPORT FILE:

Exports the organization and its components from the TCPWave IPAM to a zip file.

exportreversezonetmpl**NAME:**

exportreversezonetmpl

DESCRIPTION:

Exports the network to reverse zone template associations from the TCPWave IPAM to csv file.

ARGUMENTS:**--output_file**

Full path to the output file to which network to reverse zone template associations are exported. [mandatory]

--org

Name of the organization from which network to reverse zone template associations are to be exported. If this argument is not specified, reverse zone template from all organizations will be exported for FADM user, reverse zone template from user's organization will be exported for non-FADM user.

EXAMPLE USAGE:

```
twc exportreversezonetmpl --output_file=/tmp/output.txt
```

```
twc exportreversezonetmpl --org=TCPWave --output_file=/tmp/output.txt
```

EXPORT FILE:

The information in the output file is a comma separated list of fields as specified, in the order below

```
"IP_ADDR","ZONE_TEMPLATE","ORGANIZATION","MASK_LENGTH"
```

EXAMPLE DATA:

```
"192.193.219.0","TCPWave Default Zone Template","TCPWave","24"
```

exportrevzonerr**NAME:**

```
exportrevzonerr
```

DESCRIPTION:

Exports the reverse zone resource records from the TCPWave IPAM to csv file.

ARGUMENTS:**--output_file**

Full path to the output file to which reverse zone resource records are exported. [mandatory]

--org

Name of the organization from which reverse zone resource records are to be exported. If this argument is not specified, reverse zone resource records from all organizations will be exported for FADM user, reverse zone resource records from user's organization will be exported for non-FADM user.

EXAMPLE USAGE:

```
twc exportrevzonerr --output_file=/tmp/output.txt
```

```
twc exportrevzonerr --org=TCPWave --output_file=/tmp/output.txt
```

EXPORT FILE:

The information in the output file is a comma separated list of fields as specified, in the order below

"NETWORK_IP","ORGANIZATION_NAME","OWNER","TTL","CLASS","TYPE","DATA"

EXAMPLE DATA:

"80.0.0.0","TCPWave","www.tcpwave.com.", "1200", "IN", "PTR", "80.0.1.5"

exportreversezone

NAME:

exportreversezone

DESCRIPTION:

Exports the DNS reverse zones from the TCPWave IPAM in CSV format into the specified, output file.

ARGUMENTS:

--output_file

Full path of the output file to which zones must be exported. [mandatory]

--org

Name of the organization from which DNS reverse zones have to be exported.

If

this argument is not specified, DNS reverse zones from all organizations will be exported for FADM user whereas DNS reverse zones from user's organization

will be

exported for non-FADM user.

EXAMPLE USAGE:

twc exportreversezone --output_file=/tmp/output.txt

twc exportreversezone --org=TCPWave --output_file=/tmp/output.txt

EXPORT FILE:

The rows in the output file are comma separated values of fields as specified, in the order below

"NAME","ORG_NAME","TMPL_NAME","DNSSEC","NSEC_OPT","MONIT","DESCRIPTION","DMZ_V_ISIBLE","CONTACT_F_NAME","CONTACT_M_NAME","CONTACT_L_NAME","CONTACT_EMAIL","ADDRESS","MASK_LENGTH"

FIELD FORMATS:

DNSSEC '1' indicates that DNSSEC should be enabled for the zone. '0' indicates that DNSSEC is not enabled

NSEC_OPT 'NSEC' or 'NSEC3'

MONIT '1' indicates monitoring is enabled for this zone. '0' indicates monitoring is disabled for this zone.

EXAMPLE DATA:

```
"1.0.1.in-addr.arpa","tcpwave","TestZoneTemplate","0","NSEC","0","test reverse  
zone","1","John","","Smith","ohn.smith@tcpwave.com","1.0.2.16","28"
```

exportrr**NAME:**

exportrr

DESCRIPTION:

Exports the objects resource records of a network or subnet from the TCPWave IPAM as a csv list into the specified, output file.

ARGUMENTS:**--export_level**

Export level. It takes one of 'network' or 'subnet'. [mandatory]

--ip

IP address of the network or subnet for which you want to export object resource records. [mandatory]

--default_arr

Default A record flag. It takes '1' or '0'. If this argument is specified, as '1', object's default A record will also be exported. If it is specified, as '0', object's default A record will not be exported.

--output_file

Full path to the output file to which objects resource records are to be exported. [mandatory]

EXAMPLE USAGE:

```
twc exportrr --export_level=subnet --ip=10.1.10.0 --default_arr=1 --output_file=/tmp/output.txt
```

```
twc exportrr --export_level=network --ip=10.1.0.0 --output_file=/tmp/output.txt
```

EXPORT FILE:

The information in the output file is a comma separated list of fields as specified, in the order below

"IP_ADDRESS","ORGANIZATION_NAME","OWNER","TTL","CLASS","TYPE","DATA"

EXAMPLE DATA:

```
"10.1.10.14","TCPWave","www.tcpwave.com.", "600", "IN", "A", "10.1.10.14"
```

```
"10.1.10.14","TCPWave","dev.tcpwave.com.", "300", "IN", "CNAME", www.tcpwave.com.
```

exportscope**NAME:**

exportscope

DESCRIPTION:

Exports the scopes from the TCPWave IPAM as a csv list into the specified, output file.

ARGUMENTS:

--output_file

full path to the output file to which scopes are exported. [mandatory]

--org

Name of the organization from which scopes are to be exported. If this argument is not specified, scopes from all organizations will be exported for FADM user, scopes from user's organization will be exported for non-FADM user.

EXAMPLE USAGE:

tvc exportscope --output_file=/tmp/output.txt

tvc exportscope --org=TCPWave --output_file=/tmp/output.txt

EXPORT FILE:

The information for a DHCP scope are exported as a comma separated values as specified, in the order below

START_IP, END_IP, OBJECT_TYPE, OPTION_TEMPLATE, DHCP_SERVER, NS_A, NS_PTR, DDNS_A, DDNS_PTR, DDNS_CNAME, DDNS_MX, ALLOW_CLIENT, ALLOW_VENDOR, ALLOW_USER, DENY_CLIENT, DENY_VENDOR, DENY_USER, TTL, DESCRIPTION, ORGANIZATION

FIELD FORMATS:

START_IP, END_IP IPv4 address format

NS_A, NS_PTR, DDNS_A, DDNS_PTR, DDNS_CNAME, DDNS_MX take '0' or '1'

ALLOW_CLIENT, ALLOW_VENDOR, ALLOW_USER, DENY_CLIENT, DENY_VENDOR, DENY_USER

These flags determine allow or deny DHCP classes for client, vendor and user classes

EXAMPLE DATA:

"10.0.2.44","10.0.2.48","3G Phone","Generic Template","nusalx-trv10-sl0984","1","1","1","1","1","","","","","","","1200","Test Scope48","TCPWave"

exportsharednetwork

NAME:

exportsharednetwork

DESCRIPTION:

Exports the shared networks from the TCPWave IPAM as a csv list into the specified, output file.

ARGUMENTS:**--output_file**

Full path to the output file to which shared networks are to be exported.
[mandatory]

--org

Name of the organization from which shared networks are to be exported.
If this argument is not specified, shared networks from all organizations
will be exported for FADM user, shared networks from user's organization
will be exported for non-FADM user.

EXAMPLE USAGE:

```
twc exportsharednetwork --output_file=/tmp/output.txt
```

```
twc exportsharednetwork --org=TCPWave --output_file=/tmp/output.txt
```

EXPORT FILE:

The information in the output file is a comma separated list of fields as
specified, in the order below

"NAME","ORGANIZATION_NAME","DESCRIPTION","APPLIANCE_NAME","APPLIANCE_IP"

EXAMPLE DATA:

"NAME","ORGANIZATION_NAME","DESCRIPTION","APPLIANCE_NAME","APPLIANCE_IP"
"TCPWave shared Network","TCPWave","","dns-server02-sl0984","172.186.214.222"

exportsubnet**NAME:**

```
exportsubnet
```

DESCRIPTION:

Exports the subnets from the TCPWave IPAM as a csv list into the specified,
output file.

ARGUMENTS:**--output_file**

Full path to the output file to which subnets are exported. [mandatory]

--org

Name of the organization from which subnets are to be exported. If this
argument is not specified, subnets from all organizations will be
exported for FADM user, subnets from user's organization will be
exported for non-FADM user.

EXAMPLE USAGE:

```
twc exportsubnet --output_file=/tmp/output.txt
```

```
twc exportsubnet --org=TCPWave --output_file=/tmp/output.txt
```

IMPORT FILE:

The information in the output file is a comma separated list of fields as

specified, in the order below

ADDRESS,MASK,NETWORK_ADDRESS,NAME,DOMAIN,SUBNET_GROUP,ROUTER_ADDRESS,DHCP_TEMPLATE_NAME,PRIMARY_DHCP_SERVER,STREET_1,STREET_2,CITY,STATE,COUNTRY,ZIP,DESCRIPTION

The DOMAIN field is a quoted comma separated values of domains. The first value is considered as primary domain and rest are considered as secondary domains for the subnet. Eg: "tcpwave.com,dev.tcpwave.com"

EXAMPLE DATA:

106.0.0.0,24,106.0.0.0,"import-sn-test-1","tcpwave.com,dev.tcpwave.com",IT-SG,106.0.0.1,"","","600 ALEXANDER ROAD","","PRINCETON","NJ","USA","08540","sn-grp-test"

exportsubnetdhcp

NAME:

`exportsubnetdhcp`

DESCRIPTION:

Exports the subnet to DHCP Server associations from the TCPWave IPAM to csv file.

ARGUMENTS:

`--output_file`

Full path to the output file to which subnet to DHCP Server associations are to be exported. [mandatory]

`--org`

Name of the organization from which subnet to DHCP Server associations are to be exported. If this argument is not specified, DHCP Server associations from all organizations will be exported for FADM user, DHCP Server associations from user's organization will be exported for non-FADM user.

EXAMPLE USAGE:

`twc exportsubnetdhcp --output_file=/tmp/output.txt`

`twc exportsubnetdhcp --org=TCPWave --output_file=/tmp/output.txt`

EXPORT FILE:

The information in the output file is a comma separated list of fields as specified, in the order below

"ADDRESS","DHCP_TEMPLATE_NAME","PRIMARY_DHCP_SERVER"

EXAMPLE DATA:

"14.0.0.0","Generic","10.1.10.180"

exportsubnetgroup

NAME:

`exportsubnetgroup`

DESCRIPTION:

Exports the subnet groups from the TCPWave IPAM as a csv list into the specified, output file

ARGUMENTS:

--output_file

full path to the output file to which subnet groups are exported.
[mandatory]

--org

Name of the organization from which subnet groups are to be exported. If this argument is not specified, subnet groups from all organizations will be exported for FADM user, subnet groups from user's organization will be exported for non-FADM user.

EXAMPLE USAGE:

```
twc exportsubnetgroup --output_file=/tmp/output.txt
```

```
twc exportsubnetgroup --org=TCPWave --output_file=/tmp/output.txt
```

IMPORT FILE:

The information in the output file is a comma separated list of fields as specified, in the order below

"NAME","ORGANIZATION","DESCRIPTION"

EXAMPLE DATA:

"Generic Subnet Group","TCPWave","A Subnet Group for Generic Use"

exportvrf

NAME:

exportvrf

DESCRIPTION:

Exports the VRFs from the TCPWave IPAM as a csv list into the specified, output file.

ARGUMENTS:

--output_file

Full path to the output file to which VRFs are to be exported.
[mandatory]

--org

Name of the organization from which VRFs are to be exported. If this argument is not specified, VRFs from all organizations will be exported for FADM user, VRFs from user's organization will be exported for non-FADM user.

EXAMPLE USAGE:

```
twc exportvrf --output_file=/tmp/output.txt
```

```
twc exportvrf --org=TCPWave --output_file=/tmp/output.txt
```

EXPORT FILE:

The information in the output file is a comma separated list of fields as specified, in the order below

"NAME","ORG_NAME","DESCRIPTION"

EXAMPLE DATA:

"NAME","ORG_NAME","DESCRIPTION"

"VRF1","TCPWave","Virtual routing and forwarding"

exportmirroredzone**DESCRIPTION:**

Exports the mirrored zone from the TCPWave IPAM to a CSV file

ARGUMENTS:

--output_file

Full path to the output file to which mirrored zone are exported. [mandatory]

--org

Name of the organization from which mirrored zone are to be exported.

If this argument is not specified, mirrored zone all organizations will be exported for FADM user, DNS appliance from user's organization wil be exported for non-FADM user.

EXAMPLE:

```
twc exportmirroredzone --output_file=/tmp/mirroredzone.txt
```

```
twc exportmirroredzone --org=TCPWave --output_file=/tmp/mirroredzone.txt
```

EXPORT FILE:

The information in the output file is a comma separated list of fields as Specified in the order below

"NAME","ORG_NAME","ZONE_NAME","DESCRIPTION"

EXAMPLE DATA:

"Test","TcpWave","first.com","Test mirrored zone"

exportmicrosoftdnsserver**DESCRIPTION:**

Exports the microsoft DNS appliance from the TCPWave IPAM to a CSV file

ARGUMENTS:

--output_file

Full path to the output file to which microsoft DNS appliance are exported. [mandatory]

--org

Name of the organization from which microsoft DNS appliance are to be exported. If this argument is not specified, DNS appliance from all organizations will be exported for FADM user, DNS appliance from user's organization will be exported for non-FADM user .

EXAMPLE:

```
twc exportmicrosoftdnsserver --output_file=/tmp/msdnsserver.txt
```

```
twc exportmicrosoftdnsserver --org=TCPWave --output_file=/tmp/msdnsserver.txt
```

EXPORT FILE:

The information in the output file is a comma separated list of fields as specified in the order below

"NAME","ORG_NAME","IP_ADDR","USER_NAME","PASSWORD","DESCRIPTION","IS_HTTPS"

EXAMPLE DATA:

"Microsoft00001Remote","TcpWave","10.0.0.230,"Test","1FC23EBE134B12B","Test appliance","0"

exportzone

NAME:

exportzone

DESCRIPTION:

Exports the DNS zones from the TCPWave IPAM as a csv list into the specified, output file.

ARGUMENTS:

--output_file

full path to the output file to which zones are exported. [mandatory]

--org

Name of the organization from which DNS zones are to be exported. If this argument is not specified, DNS zones from all organizations will be exported for FADM user, DNS zones from user's organization will be exported for non-FADM user .

EXAMPLE USAGE:

```
twc exportzone --output_file=/tmp/output.txt
```

```
twc exportzone --org=TCPWave --output_file=/tmp/output.txt
```

EXPORT FILE:

The information in the output file is a comma separated list of fields as specified, in the order below

NAME, ORG_NAME, TMPL_NAME, DNSSEC, NSEC_OPT, AD_UPDATES ,DC_IP, AD_SEC, MONIT, DESCRIPTION, DMZ_VISIBLE, ACL

FIELD FORMATS:

DNSSEC 1 indicates that DNSSEC should be enabled for the zone. 0 indicates that DNSSEC is not enabled

NSEC_OPT 'NSEC' or 'NSEC3'

AD_UPDATES '1' indicates Active Directory updates are enabled for this zone.
'0' indicates Active Directory updates are disabled for this zone.

DC_IP is a comma separated values of IPs of domain controllers applicable for this zone.

AD_SEC '1' indicates Active Directory secure updates are enabled for this zone. '0' indicates Active Directory secure updates are disabled for this zone.

MONIT '1' indicates monitoring is enabled for this zone. '0' indicates monitoring is disabled for this zone.

ACL is a comma separated names of ACL applicable for this zone.

EXAMPLE DATA:

```
"tcpwave.com","TCPWave","TestZoneTemplate","0","NSEC","0","192.168.1.10","1","1","test zone  
for tcpwave.com","0",""
```

exportzonerr**NAME:**

exportzonerr

DESCRIPTION:

Exports the DNS zones resource records from the TCPWave IPAM as a csv list into the specified, output file.

ARGUMENTS:

--output_file

Full path to the output file to which zones resource records are exported. [mandatory]

--org

Name of the organization from which zones resource records are to be exported. If this argument is not specified, zones resource records from all organizations will be exported for FADM user, zones resource records from user's organization will be exported for non-FADM user.

EXAMPLE USAGE:

```
twc exportzonerr --output_file=/tmp/output.txt
```

```
twc exportzonerr --org=TCPWave --output_file=/tmp/output.txt
```

EXPORT FILE:

The information in the output file is a comma separated list of fields as specified, in the order below

"ZONE_NAME","ORGANIZATION_NAME","OWNER","TTL","CLASS","TYPE","DATA","EXTERNAL"

EXAMPLE DATA:

"tcpwave.com","TCPWave","www30.tcpwave.com.", "1200","IN","A","25.6.67.7","0"

exportsubnettemplate**NAME**

exportsubnettemplate

DESCRIPTION

Exports the Subnet templates from the TCPWave IPAM as a name-value pair into the specified output file.

ARGUMENTS:**--output_file**

Full path to the output file to which subnet templates are exported.
[mandatory]

--org

Name of the organization from which subnet templates are to be exported. If this argument is not specified, subnet templates from all organizations will be exported for FADM user. For non-FADM users subnet templates will be exported based on the user's permissions.

EXAMPLE

```
twc exportsubnettemplate --output_file=/tmp/output.txt
```

```
twc exportsubnettemplate --org=TCPWave --output_file=/tmp/output.txt
```

EXPORT FILE

The information in the output file is a name-value pair list of fields as specified in the order below

NAME, ORGANIZATION_NAME, PRIMARY_DOMAIN, ROUTER_OPTION, DESCRIPTION,
LOCATION_NAME, CONTACT_NAME, SUBNET_TYPE, SUBNET_GROUP, VLAN_ID, VRF,
ENABLE_DISCOVERY, ENABLE_RECLAIM, DISCOVERY_TEMPLATE, DHCP_OPTION_TEMPLATE,
DHCP_APPLIANCE, DHCP_APPLIANCE_ADDRESS, DHCP_FAILOVER_PEER, SHARED_NETWORK,
DNS_APPLIANCES, CLOUD_PROVIDER, SECONDARY_DOMAINS, START_OFFSET, END_OFFSET,
CLASS_CODE, ALLOCATION_TYPE, DOMAIN, CLIENT_CLASS_ALLOW, CLIENT_CLASS_DENY,
USER_CLASS_ALLOW, USER_CLASS_DENY, VENDOR_CLASS_ALLOW, VENDOR_CLASS_DENY, TTL,
RES_EXPIRY_DATE

SECTIONS & CONFIGURATION PARAMETERS:

[subnet-template]

NAME Name of the subnet template.
ORGANIZATION_NAME Organization Name of the subnet template.
PRIMARY_DOMAIN Primary domain for the subnet created using this template
ROUTER_OPTION Select the first or last object as a router or give the router object offset.
DESCRIPTION Description of the subnet template.
LOCATION_NAME Location of the subnet template.
CONTACT_NAME Contact for the subnet template.
SUBNET_TYPE Non-DHCP, DHCP-Enabled or Cloud Hosted

(If you select the subnet type as DHCP-Enabled, then the system displays the DHCP Attributes fields.

If you select the subnet type as Cloud Hosted, then the system displays the Cloud Attributes fields.)

SUBNET_GROUP Subnet group for the subnet template.
VLAN_ID ID of VLAN.
VRF Select VRF.
ENABLE_DISCOVERY Takes 'true' or 'false' to enable or disable discovery respectively.
ENABLE_RECLAIM Takes 'true' or 'false' to enable or disable reclaim respectively.
DISCOVERY_TEMPLATE Name of discovery template.
DHCP_OPTION_TEMPLATE Name of DHCP Option Template.
DHCP_APPLIANCE Name of DHCP Appliance.

DHCP_APPLIANCE_ADDRESS IP of the DHCP Appliance.
DHCP_FAILOVER_PEER Specify the DHCP failover peer.
SHARED_NETWORK Specify the shared networks.
DNS_APPLIANCES IP of DNS Appliances.
CLOUD_PROVIDER Specify cloud provider.
SECONDARY_DOMAINS Specify the secondary domains.

[address-allocations]

START_OFFSET Enter the start offset.
END_OFFSET Enter the end offset.
CLASS_CODE Select object type.
ALLOCATION_TYPE Specify if static, dynamic or reserved.
DOMAIN Name of the domain.
CLIENT_CLASS_ALLOW Specify the allowed client classes.
CLIENT_CLASS_DENY Specify the denied client classes.
USER_CLASS_ALLOW Specify the allowed user classes.
USER_CLASS_DENY Specify the user client classes.
VENDOR_CLASS_ALLOW Specify the allowed vendor classes.
VENDOR_CLASS_DENY Specify the denied vendor classes.
TTL Specify the time to live.
RES_EXPIRY_DATE Specify the expiry date for reserved objects.

EXAMPLE DATA:

[subnet-template]

NAME=v4-template
ORGANIZATION_NAME=Internal
PRIMARY_DOMAIN=tcpwave.com

```
ROUTER_OPTION=first
DESCRIPTION=
LOCATION_NAME=test    hyd    telangana    India
CONTACT_NAME=ramya    bali    ramya.bali@tcpwave.com
SUBNET_TYPE=Non-DHCP
SUBNET_GROUP=subnet-test-group
VLAN_ID=100
VRF=test-vrf
ENABLE_DISCOVERY=yes
ENABLE_RECLAIM=yes
DISCOVERY_TEMPLATE=Default Discovery Template
DHCP_OPTION_TEMPLATE=
DHCP_APPLIANCE=
DHCP_APPLIANCE_ADDRESS=
DHCP_FAILOVER_PEER=null
SHARED_NETWORK=null
DNS_APPLIANCES=
CLOUD_PROVIDER=
SECONDARY_DOMAINS=null
```

[address-allocations]

```
START_OFFSET=10
END_OFFSET=1000
CLASS_CODE=3G Phone
LOCATION_TYPE=Static
DOMAIN=tcpwave.com
CLIENT_CLASS_ALLOW=
CLIENT_CLASS_DENY=
USER_CLASS_ALLOW=
USER_CLASS_DENY=
```

```
VENDOR_CLASS_ALLOW=
VENDOR_CLASS_DENY=
TTL=1200
RES_EXPIRY_DATE=
```

EXAMPLE DATA:

```
"TestSubetTemplate","tcpwave","tcpwave.in","first","","","","Non-
DHCP","","","","no","","","","","","","","","10","100","3G-
Phone","Static","afghanisthan.af","","","","","","","1200",""
```

exportzonetemplate**NAME:**

exportzonetemplate

DESCRIPTION:

Exports the DNS zone templates from the TCPWave IPAM as a csv list into the specified, output file.

ARGUMENTS:**--output_file**

Full path to the output file to which zone templates are exported.
[mandatory]

--org

Name of the organization from which zone templates are to be exported.
If this argument is not specified, zone templates from all organizations will be exported for FADM user, zone templates from user's organization will be exported for non-FADM user.

EXAMPLE USAGE:

```
twc exportzonetemplate --output_file=/tmp/output.txt
```

```
twc exportzonetemplate --org=TCPWave --output_file=/tmp/output.txt
```

EXPORT FILE:

The information in the output file is a comma separated list of fields as specified, in the order below

```
NAME, ORG_NAME, DEFAULT_TTL, SOA_EMAIL, SOA_REFRESH, SOA_RETRY, SOA_EXPIRY,
SOA_NEGCACHE, SOA_SERIAL, ALLOW_NOTIFY, ALLOW_QUERY, ALLOW_XFR,
ALLOW_UPDATE, FORWARD, FWD_IPV4, FWD_IPV6, MASTERS, SLAVES, DESCRIPTION
```

FIELD FORMATS:

DEFAULT_TTL, SOA_REFRESH, SOA_RETRY, SOA_EXPIRY, SOA_NEGCACHE are time

format as follows:

Time formats are specified, as integer/time_unit. time_unit can be one of the following values: S,MIN,H,D,W,MON,Y representing seconds, minutes, hours, days, weeks, months, years

Example: 84600/S or 30/D

SOA_SERIAL is 'DATE' or 'NODATE'

ALLOW_NOTIFY, ALLOW_QUERY, ALLOW_XFR, ALLOW_UPDATE accept one of the following ACL formats

IPAddress/permission (192.168.0.1/Allow)

ACL-name/permission (internal/Deny)

IPAddress/mask/permission (192.168.0.0/24/Allow)

MASTERS, SLAVES are comma separated lists of IP addresses of authoritative servers acting as slaves for the reverse zone

FORWARD takes 'first' or 'only' as values

FWD_IPV4 semicolon separated list of ipv4 addresses

FWD_IPV6 semicolon separated list of ipv6 addresses

EXAMPLE DATA:

```
"TestZoneTemplate","tcpwave","52000/S","admin@tcpwave.com","","","","","","","any/Allow,192.168.1.4/Deny","any/Allow,192.168.1.3/Deny","any/Allow,192.168.1.2/Deny","any/Allow,192.168.1.1/Deny","","","","","192.168.1.102,192.168.1.107","","Test zone template"
```

exportdnsforwarderstmpl

NAME

exportdnsforwarderstmpl

DESCRIPTION

Exports the DNS forwarder template with forwarder zones from the TCPWave IPAM to a csv file.

ARGUMENTS

--output_file

Full path to the output file to which DNS forwarder template with forwarder zones are exported. [mandatory]

EXAMPLE

```
twc exportdnsforwarderstmpl --output_file=/tmp/forwarders.txt
```

EXPORT FILE

The information in the output file is specified in the order below.

EXAMPLE DATA:

```
[dns-forwarders]
NAME=TestForwarder
DESCRIPTION=Description for the forwarders
[forwarder]
ZONENAME=test1.com
FORWARD=only
FORWARDERSIPV4=10.1.10.12;10.1.10.13
FORWARDERSIPV6=
ISPRESENT=true
[forwarder]
ZONENAME=zone.com
FORWARD=only
FORWARDERSIPV4=10.1.10.15;10.1.10.16
FORWARDERSIPV6=
ISPRESENT=true
[dns-forwarders]
NAME=TestTwo
DESCRIPTION=Description for the forwarders
[forwarder]
ZONENAME=test2.com
FORWARD=only
FORWARDERSIPV4=10.1.10.18;10.1.10.19
FORWARDERSIPV6=
ISPRESENT=true
[forwarder]
ZONENAME=zone3.com
FORWARD=only
FORWARDERSIPV4=10.1.10.20;10.1.10.21
FORWARDERSIPV6=
ISPRESENT=true
```

Gets

fetchquerylog

NAME

fetchquerylog

DESCRIPTION

Fetch the DNS server query logs from the TCPWave IPAM. DNS server should be 'BIND AUTH','BIND CACHE' or 'UNBOUND'.

ARGUMENTS

--ip

IP Address of the DNS server. [mandatory]

--start_date

Start date for logs generation. Date format is mm/dd/yyyy. [mandatory]

--end_date

End date for logs generation. Date format is mm/dd/yyyy. [mandatory]

--output_file

Full path to the output file to which query logs are to be written. If the file path is not specified, the output is written to the standard output.

--count

Number of lines to be fetched from the query log. If this argument is not specified, 100 lines will be fetched.

EXAMPLE

```
twc fetchquerylog --ip=10.1.10.190 --output_file=/tmp/querylog.txt --count=200 --
start_date=08/12/2021 --end_date=08/15/2021
```

getipamappliance**NAME:**

getipamappliance - Get the configuration for an IPAM appliance in the TCPWave IPAM.

DESCRIPTION:

Get the configuration of an IPAM appliance from the TCPWave IPAM and write the contents to a file.

ARGUMENTS:

--ip

IP address of the IPAM appliance. [mandatory]

--output_file

Full path to the output file to which the appliance configuration is to be written. If the file path is not specified, the output is written to the standard output.

EXAMPLE:

```
twc getipamappliance --ip=10.1.10.174 --output_file=/tmp/ipamappliance.txt
```

getbgpconfig**NAME:**

getbgpconfig

DESCRIPTION:

Displays the BGP configuration of a DNS server from the TCPWave IPAM.

ARGUMENTS:

--ip

IP Address of the DNS server. [mandatory]

--type

Server type. It takes 'BIND CACHE' or 'UNBOUND'. [mandatory]

--output_file

Full path to the output file to which the BGP configuration is to be written. If the file path is not specified, the output is written to the standard out.

EXAMPLE USAGE:

```
twc getbgpconfig --ip=10.1.10.190 --type='BIND CACHE' --output_file=/tmp/bgpconfig.txt
```

SAMPLE OUTPUT CONTENTS:

[Basic_Configuration]

ASN=64881

Router_ID=192.168.1.80

Debug_BGP_Events=true

Debug_BGP_Updates=true

```
Debug_BGP_Filters=true
[BGP_Timer]
Keep_Alive=4
Hold_down=16
[Networks]
Network_List=192.193.215.64/30,192.193.215.68/30,192.193.215.72/30,192.168.1.80/32
[Prefix_List]
Name=DNS
Sequence=5
IP=192.193.215.64/30
Prefix_length=
permission=permit
[Prefix_List]
Name=DEFAULT
Sequence=5
IP=0.0.0.0/0
Prefix_length=
permission=permit
[Neighbor_Group]
Name=EBGP-PEERS
Remote ASN=64881
Route_Map_In=
Route_Map_Out=
Prefix_List_In=DEFAULT
Prefix_List_out=DNS
[Neighbor]
Peer=10.1.10.253
Peer_Group=EBGP-PEERS
Description=GSS-PEER-IP1-DESCRIPTION
```

getchangeticket

NAME:

getchangeticket

DESCRIPTION:

Displays the change ticket associated with the current session in the TCPWave IPAM.

ARGUMENTS:

- NA

EXAMPLE USAGE:

twc getchangeticket

getdhcpolicytmpl

NAME:

getdhcpolicytmpl

DESCRIPTION:

Get a DHCP policy template specified, by template name from the TCPWave IPAM

and write the contents to a file.

ARGUMENTS:**--name**

Name of the DHCP policy template to be retrieved from TCPWave IPAM
[mandatory]

--org

Name of the organization.

--output_file

Full path to the output file to which the template contents are to be written. If the file path is not specified, the output is written to the standard out.

--display

Takes 'all' or 'used'. 'used' will display/output only used options.
'all' will display all the options including unused options. [mandatory]

EXAMPLE USAGE:

```
twc getdhcpolicytmpl --name="base policy template" --org=TcpWave --  
output_file="/tmp/dhcpcpolicytmpl.txt" --display=all
```

SAMPLE OUTPUT:

```
# Global Parameters
```

```
Authoritative(authoritative)=  
DB Time Format(db-time-format )=  
Lease File Time(lease-file-name)=  
Local Port(local-port )=  
Log Facility(log-facility)=  
OMAPI Port(omapi-port)=  
PID File Name(pid-file-name )=  
Server DUID(server-duid)=
```

```
# DNS Update Parameters
```

```
DDNS Hostname(ddns-hostname)=  
DDNS Domainname(ddns-domainname)=  
DDNS Rev Domain Name(ddns-rev-domainname)=  
DDNS Update Style(ddns-update-style)=  
DDNS Updates(ddns-updates)=  
Do Forward Updates(do-forward-updates)=  
Update Conflict Detection(update-conflict-detection)=  
Update Optimization(update-optimization)=  
Update Static Leases(update-static-leases)=  
Use Host Decl Names(use-host-decl-names)=
```

```
# DHCP Server-Client Communications Parameters
```

Always Broadcast(always-broadcast)=
Always reply RFC1048(always-reply-rfc1048)=
Min Secs(min-secs)=
Remote Port(remote-port)=
Stash Agent Options(stash-agent-options)=

Client Handling Parameters

Adaptive Lease Time Percentage(adaptive-lease-time-percentage)=
Boot Unknown Clients(boot-unknown-clients)=
Default Lease time(default-lease-time)=
File Name(filename)=
Fixed Address(fixed-address)=
Get Lease Host Names(get-lease-hostnames)=
Hardware(hardware)=
Host Identifier(host-identifier)=
Infinite is reserved(infinite-is-reserved)=
Max Lease Time(max-lease-time)=
Min Lease Time(min-lease-time)=
Next Server(next-server)=
One Lease Per Client(one-lease-per-client)=
Ping Check(ping-check)=
Ping Timeout(ping-timeout)=
Preferred Lifetime(preferred-lifetime)=
Server Identifier(server-identifier)=
Server Name(server-name)=
Site Option Spaceuse Lease Addr For Default Route(site-option-spaceuse-lease-addr-for-default-route)=
Limit Addrs Per Ia(limit-addrs-per-ia)=
Vendor Option Space(vendor-option-space)=

getdhcptmpl**NAME**

getdhcptmpl

DESCRIPTION

Displays the contents of a DHCP option template from the TCPWave IPAM.

ARGUMENTS

--name

name of the DHCP option template to be retrieved from TCPWave IPAM
[mandatory]

--org

Name of the organization. [mandatory]

--output_file

full path to the output file to which the template contents are to be written. If the file path is not specified, the output is written to the standard out.

--display

takes 'all' or 'used'. 'used' will display/output only used options.
'all' will display all the options including unused options. [mandatory]

EXAMPLE:

```
twc getdhcptmpl --name="voip devices template" --org=TcpWave --  
output_file="/tmp/dhcpoPTIONtmpl.txt" --display=all
```

SAMPLE OUTPUT

```
# User Authentication Protocol Options
```

```
User Authentication Servers (98)=
```

```
# TCP Parameters
```

```
Default TCP TTL (37)=125
```

```
Keepalive Time (38)=
```

```
Keepalive Data (39)=
```

```
# SLP Protocol Options
```

```
Service Location Protocol Directory Agent (78)=
```

```
SLP Service Scope (79)=
```

RFC 3397 Options

Domain Search (119)=

RFC 1497 Vendor Extensions

Subnet Mask (1)=Same as in subnet profile

Time Offset (2)=

Router (3)=Same as in subnet profile

Time Server (4)=

Name Server (5)=

Domain Name Server (6)=

Log Server (7)=

Quotes Server (8)=

LPR Server (9)=

Impress Server (10)=

RLP Server (11)=

Hostname (12)=

Boot File Size (13)=

Merit Dump File (14)=

Domain Name (15)=

Swap Server (16)=

Root Path (17)=

Extension File (18)=

Novell Options

Netware/IP Domain (62)=

Netware/IP Options-nwip.nsq-broadcast (63)=

Netware/IP Options-nwip.preferred-dss (63)=

Netware/IP Options-nwip.nearest-nwip-server (63)=

Netware/IP Options-nwip.autoretries (63)=

Netware/IP Options-nwip.autoretry-secs (63)=

Netware/IP Options-nwip.nwip-1-1 (63)=

Netware/IP Options-nwip.primary-dss (63)=

NDS Servers (85)=

NDS Tree Name (86)=
NDS Context (87)=
Miscellaneous
Netinfo Address (112)=
Netinfo Tag (113)=
Default URL (114)=
Vendor Identified Vendor-Specific Information (125)=
MTU Subnet (27)=
Parameter List (55)=
DHCP Max Msg Size (57)=
DHCP Renewal time (58)=
DHCP Rebinding time (59)=
Home Agent Addresses (68)=
User Class (77)=
Agent/Circuit Id (82)=
Agent/Remote Id (82)=
Link Layer Parameters per interface
Trailers (34)=
ARP Timeout (35)=
Ethernet (36)=
IP Layer Params Per Host
Forward On/Off (19)=
Source Routing (20)=
Policy Filter (21)=
Max Datagram Size (22)=
Default IP TTL (23)=
MTU Timeout (24)=
MTU Plateau (25)=
IP Layer Parameters per interface
MTU Interface (26)=
Broadcast Address (28)=

Mask Discovery (29)=
Mask Supplier (30)=
Router Discovery (31)=
Router Request (32)=
Static Route (33)=
DHCP Extensions
Address Time (51)=
Overload (52)=
Vendor Class Id (60)=
Client Id (61)=
Server Name (66)=
Bootfile Name (67)=
Application and Service Parameters
NIS Domain (40)=
NIS Servers (41)=
NTP Servers (42)=
Vendor Specific (43)=
NETBIOS Name Server (44)=
NETBIOS Dist Server (45)=
NETBIOS Node Type (46)=
NETBIOS Scope (47)=
X Window Font (48)=
X Window Manager (49)=
NIS+ Domain Name (64)=
NIS+ Server Address (65)=
SMTP Server (69)=
POP3 Server (70)=
NNTP Server (71)=
WWW Server (72)=
Finger Server (73)=
IRC Server (74)=

StreetTalk Server (75)=
StreetTalk Directory Assistance (STDA) Server (76)=
BCMCS Controller Domain Name (88)=
BCMCS Controller IPv4 address option (89)=
voip-options (CUSTOM OPTIONS)
ip-map (130)=
options (OPTION SPACES)
string (1)=
You have new mail in /var/spool/mail/root

getdhcpserver

NAME:

getdhcpserver

DESCRIPTION:

Get the configuration of a DHCP server from the TCPWave IPAM and write the contents to a file.

ARGUMENTS:

--ip
IP address of the DHCP appliance.

--output_file
Full path to the output file to which the server configuration is to be written. If the file path is not specified, the output is written to the standard output.

EXAMPLE USAGE:

tvc getdhcpserver --ip=10.1.10.180 --output_file=/tmp/dhcpserver.txt

FILE FORMAT:

The output file format is as follows:

Each configuration section begins with a [<section name>] field followed by <name>=<value> pairs one per line.

SECTIONS & CONFIGURATION PARAMETERS:

[dhcp-server]

IP_ADDRESS IP address of the appliance.

ORGANIZATION_NAME Organization name associated with the DHCP appliance.

POLICY_TEMPLATE Policy Template associated with the DHCP appliance.

APPLIANCE_GROUP Appliance group associated with the DHCP appliance.

ENABLE_MONIT '0' to enable monitoring and '1' to disable monitoring

TIME_ZONE Time zone

[ntp]

NTP_SERVERS Comma separated list of IP addresses of NTP servers

[dns-resolver]

NAME_APPLIANCES comma separated list of IP addresses of DNS appliance. Maximum allowed DNS appliance is four.

SEARCH_SUFFIXES comma separated list of domain names. Maximum allowed search suffix is six.

[snmp]

TRAP_SINK_1 IP address of SNMP trap sink

TRAP_SINK_2 IP address of SNMP trap sink

COMMUNITY_STRING Community string for SNMP

SYSTEM_LOCATION System Location

SYSTEM_CONTACT System contact

PROCESS_LIST Comma separated list of processes to be monitored. The following is a valid list of processes:

ntpd, dns, bgpd, zebra, crond, sshd, monit, syslog-ng,dhcpd

ENABLE_SNMPV3 Takes 'true' or 'false'. 'true' indicates that SNMPv3 is enable. 'false' indicates that SNMPv3 is disable.

[snmpv3]

USER_NAME User name of SNMPv3

PASSWORD Password of the specified, user

AUTHENTICATION_PROTOCOL Authentication protocol

ENCRYPTION_PROTOCOL Encryption protocol

[tacacs]

ENABLE_TACACS Takes '0' or '1'. '1' indicates TACACS+ configuration should be enabled for this server. '0' indicates TACACS+ configuration should be disabled

TACACS_PASSKEY TACACS passkey

TACACS_SERVERS Comma separated list of TACACS servers.

[syslogng-global-options]

TIME_REOPEN The time to wait in seconds before a dead connection is reestablished. Takes a value less than or equal to 32767.

TIME_REAP If no new messages are written to a destination within the specified, time in seconds, the connection will be closed, and its state will be freed. Takes a value less than or equal to 32767.

FLUSH_LINES Specifies how many lines are flushed to a destination at a time. Takes a value less than or equal to 32767.

STATS_FREQ Syslog-NG OSE periodically sends a log statistics message. Takes a value less than or equal to 32767.

LOG_FIFO_SIZE The number of messages that the output queue can store.

Takes a value less than or equal to 32767.

LOG_MSG_SIZE The maximal length of the log messages is limited by this option. It is not recommended to set the option value higher than 10 MiB. Takes a value less than or equal to 32767.

KEEP_TIMESTAMP Specifies whether syslog-ng should accept the timestamp received from the sending application or client.

Takes value 'Yes' or 'No'.

[syslogng-source]

SOURCE_NAME Name of the Source

INTERNAL_MSG Internal syslog-NG message, takes input values as '0' or '1'. default value is '1'.

SYSTEM_MSG System specific log message, takes input value as '0' or '1'.

MSG_TXT_FILE Message from text file, takes the file name as input.

MSG_MULTI_TXT_FILE Message from multiple text files, takes input '0' or '1'. if this flag is '1' need to specify the **FILE_PATH** and **FILE_PATTERN**.

FILE_PATH File patch to the multiple text file.

FILE_PATTERN File Name pattern.

SYSLOG_SERVER Syslog-NG sever, takes the input as '0' or '1'.

IP_ADDRESS IP address of the syslog server.

PORT Port number of the syslog server.

NETWORK_PROTOCOL Network protocol, supports 'UDP' and 'TCP'.

[syslogng-filter]

FILTER_NAME Name of the Filter.

CONDITION Takes the input as 'complex' or 'simple'.

FACILITIES Allow values are one or more comma separated option given below.

auth, authpriv, cron, deamon, kern, lpr, mail,mark, news, syslog, user, uucp, local0, local1,local2, local3, local4, local5, local6, local7.

PRIORITIES Allow values are one or more comma separated option given below.

info, notice, warning, err, crit, alert, emerg.

HOST_NAME Name of the host.

IP_NETWORK IP address with mask length.

MATCH_EXPRESSION Match expression.

PROGRAM Program.

[syslogng-destination]

DESTINATION_NAME Name of the destination.

TYPE_SNG Type of the destination. takes the value between 1 to 5.

'1'= File

'2'= Named pipe

'3'= Local Users

'4'= All logged-in users

'5'= Syslog server.

LOG_FILE_NAME File name to log the message, mandatory when **TYPE_SNG** is

specified, as '1'.

NAMED_PIPE_NAME Named pipe name, mandatory when TYPE_SNG is specified, as '2'.

LOCAL_USERS Local users, mandatory when TYPE_SNG is specified, as '3'.

SYSLOG_SERVER IP address of the syslog server, mandatory when TYPE_SNG is specified, as '5'.

PORT Port number of the syslog server, mandatory when TYPE_SNG is specified, as '5'.

NETWORK_PROTOCOL Network protocol, supports 'UDP' and 'TCP', mandatory when TYPE_SNG is specified, as '5'.

[syslogng-target]

SOURCE Name of the source.

FILTER Name of the filter.

DESTINATION Name of the destination.

[macexclusions]

MAC MAC address of the device exclusion list

DESCRIPTION Description for the MAC address exclusion

SAMPLE FILE CONTENTS:

[dhcp-server]

IP_ADDRESS=10.1.10.180

ORGANIZATION_NAME=TCPWave

POLICY_TEMPLATE=Clone-Policy-1

APPLIANCE_GROUP=Appliance-Group

ENABLE_MONIT=1

TIME_ZONE=GMT (GMT)

[ntp]

NTP_SERVERS=10.1.10.10,10.1.10.11,10.1.10.12,10.1.10.13,

[dns-resolver]

NAME_APPLIANCES=8.8.8.8

SEARCH_SUFFIXES=tcpwave.com

[snmp]

TRAP_SINK_1=10.1.10.15

TRAP_SINK_2=10.1.10.18

COMMUNITY_STRING=public

SYSTEM_LOCATION=

SYSTEM_CONTACT=

PROCESS_LIST=dns,

ENABLE_SNMPV3=true

[snmpv3]

```
USER_NAME=admin
PASSWORD=abc123
AUTHENTICATION_PROTOCOL=SHA
ENCRYPTION_PROTOCOL=AES
[syslogng-global-options]
TIME_REOPEN=60
TIME_REAP=60
FLUSH_LINES=60
STATS_FREQ=600
LOG_FIFO_SIZE=1000
LOG_MSG_SIZE=2048
KEEP_TIMESTAMP=Yes
```

[syslogng-source]

```
SOURCE_NAME=s_sys
INTERNAL_MSG=1
SYSTEM_MSG=1
MSG_TXT_FILE=/var/tmp/mft.txt
MSG_MULTI_TXT_FILE=1
FILE_PATH=mfts.txt
FILE_PATTERN=/var/tmp
SYSLOG_SERVER=1
IP_ADDRESS=192.168.0.2
PORT=53
NETWORK_PROTOCOL=UDP
```

[syslogng-filter]

```
FILTER_NAME=f_kernel
CONDITION=complex
COMPLEX_CONDITION=facility(kern)
```

[syslogng-filter]

```
FILTER_NAME=f_default
CONDITION=complex
COMPLEX_CONDITION=level(info..emerg) and not (facility(mail) or facility(authpriv) or
facility(cron))
```

[syslogng-filter]

```
FILTER_NAME=f_auth
CONDITION=complex
COMPLEX_CONDITION=facility(authpriv)
```

[syslogng-filter]

```
FILTER_NAME=f_mail
```

```
CONDITION=complex  
COMPLEX_CONDITION=facility(mail)
```

```
[syslogng-filter]
```

```
FILTER_NAME=f_emergency  
CONDITION=complex  
COMPLEX_CONDITION=level(emerg)
```

```
[syslogng-filter]
```

```
FILTER_NAME=f_news  
CONDITION=complex  
COMPLEX_CONDITION=facility(uucp) or (facility(news) and level(crit..emerg))
```

```
[syslogng-filter]
```

```
FILTER_NAME=f_boot  
CONDITION=complex  
COMPLEX_CONDITION=facility(local7)
```

```
[syslogng-filter]
```

```
FILTER_NAME=f_cron  
CONDITION=complex  
COMPLEX_CONDITION=facility(cron)
```

```
[syslogng-filter]
```

```
FILTER_NAME=filter  
CONDITION=simple  
FACILITIES=auth,authpriv,cron  
PRIORITIES=info,notice  
HOST_NAME=local  
IP_NETWORK=192.166.0.2/24  
MATCH_EXPRESSION=exp  
PROGRAM=prog
```

```
[syslogng-destination]
```

```
DESTINATION_NAME=d_cons  
LOG_FILE_NAME=console  
TYPE_SNG=1  
ENABLE_SYNC=Yes
```

```
[syslogng-destination]
```

```
DESTINATION_NAME=d_mesg  
LOG_FILE_NAME=messages  
TYPE_SNG=1  
ENABLE_SYNC=Yes
```

```
[syslogng-destination]
```

```
DESTINATION_NAME=d_auth
LOG_FILE_NAME=secure
TYPE_SNG=1
ENABLE_SYNC=Yes
```

```
[syslogng-destination]
```

```
DESTINATION_NAME=d_mail
LOG_FILE_NAME=maillog
TYPE_SNG=1
ENABLE_SYNC=Yes
```

```
[syslogng-destination]
```

```
DESTINATION_NAME=d_spooler
LOG_FILE_NAME=spooler
TYPE_SNG=1
ENABLE_SYNC=Yes
```

```
[syslogng-destination]
```

```
DESTINATION_NAME=d_boot
LOG_FILE_NAME=boot.log
TYPE_SNG=1
ENABLE_SYNC=Yes
```

```
[syslogng-destination]
```

```
DESTINATION_NAME=d_cron
LOG_FILE_NAME=cron
TYPE_SNG=1
ENABLE_SYNC=Yes
```

```
[syslogng-destination]
```

```
DESTINATION_NAME=d_kern
LOG_FILE_NAME=kern
TYPE_SNG=1
ENABLE_SYNC=Yes
```

```
[syslogng-destination]
```

```
DESTINATION_NAME=d_mlal
TYPE_SNG=4
```

```
[syslogng-destination]
```

```
DESTINATION_NAME=ttt
```

```
TYPE_SNG=4
```

```
[syslogng-destination]
```

```
DESTINATION_NAME=port
```

```
TYPE_SNG=5
```

```
SYSLOG_SERVER=192.166.0.2
```

```
NETWORK_PROTOCOL=UDP
```

```
PORT=514
```

```
[syslogng-target]
```

```
SOURCE=s_sys
```

```
FILTER=f_kernal
```

```
DESTINATION=d_kern
```

```
[syslogng-target]
```

```
SOURCE=s_sys
```

```
FILTER=f_default
```

```
DESTINATION=d_mesg
```

```
[syslogng-target]
```

```
SOURCE=s_sys
```

```
FILTER=f_auth
```

```
DESTINATION=d_auth
```

```
[syslogng-target]
```

```
SOURCE=s_sys
```

```
FILTER=f_mail
```

```
DESTINATION=d_mail
```

```
[syslogng-target]
```

```
SOURCE=s_sys
```

```
FILTER=f_emergency
```

```
DESTINATION=d_mal
```

```
[syslogng-target]
```

```
SOURCE=s_sys
```

```
FILTER=f_news
```

```
DESTINATION=d_spol
```

```
[syslogng-target]
```

```
SOURCE=s_sys
```

```
FILTER=f_boot
```

```
DESTINATION=d_boot
```

[syslogng-target]

SOURCE=s_sys
FILTER=f_cron
DESTINATION=d_cron
[macexclusions]

MAC=8E-2C-E7-88-53-7A
DESCRIPTION=

[macexclusions]

MAC=E0:8F:8D:59:CF:60
DESCRIPTION=

getdnsacl**NAME:**

getdnsacl

DESCRIPTION:

Displays a BIND DNS ACL definition from the TCPWave IPAM.

ARGUMENTS:**--name**

name of the DNS ACL [mandatory]

--output_file

full path to the file where the contents of the ACL are written to. If this argument is not specified, the contents are displayed on the standard output.

--formatTakes 'csv' or 'prop'. Specifies the format of the output. 'csv' displays the contents as a comma separated list of values. 'prop' displays the contents in the format name=value.
'csv' is the default format if this argument is not specified.,**EXAMPLE USAGE:**

twc getdnsacl --name=external --format=prop

twc getdnsacl --name=external --format=csv --output_file=/tmp/output.csv

getdnsdebuglevel**NAME:**

getdnsdebuglevel

DESCRIPTION:Displays the debug level of a DNS server in the TCPWave IPAM.
DNS server should be 'BIND AUTH','BIND CACHE','UNBOUND' or 'NSD'.**ARGUMENTS:****--ip**

IP address of the DNS server [mandatory]

EXAMPLE USAGE:

twc getdnsdebuglevel --ip=10.1.10.240

getdnsopttmp1**NAME**

getdnsopttmp1

DESCRIPTION

Get a DNS option template specified by template name from the TCPWave IPAM and write the contents to a file.

ARGUMENTS

--name

Name of the DNS option template to be retrieved from TCPWave IPAM
[mandatory]

--type

DNS server type. Takes one of the following values: 'BIND AUTH', 'NSD'
'BIND CACHE', 'UNBOUND', or 'DNS PROXY'. [mandatory]

--org

Name of the organization from which the DNS option template to be
retrieved. [mandatory]

--output_file

Full path to the output file to which the template contents are to be
written. If the file path is not specified, the output is written to
the standard output.

EXAMPLE

```
twc getdnsopttmpl --name="BIND AUTH Default Template" --type="BIND AUTH" --org=TCPWave  
--output_file=/tmp/bindtemplate.txt
```

SAMPLE OUTPUT FILE CONTENTS FOR BIND AUTH TYPE

```
Directory(directory)=/  
Allow Query(allow-query)=any/Allow;  
Allow Recursion(allow-recursion)=any/Allow;  
Allow Transfer(allow-transfer)=none/Allow;  
Blackhole(blackhole)=none/Allow;  
DNSSEC Enable(dnssec-enable)=yes  
DNSSEC Validation(dnssec-validation)=yes  
Listen On v6(listen-on-v6)=none  
Check SRV CNAME(check-srv-cname)=ignore  
Check MX CNAME(check-mx-cname)=ignore  
Check MX(check-mx)=ignore
```

Check Names(check-names)=master ignore,response ignore
Dump File(dump-file)=/var/named/log/named_dump.db
Lame TTL(lame-ttl)=0
Max Negative Cache TTL(max-nocache-ttl)=60
Minimal Responses(minimal-responses)=yes
PID File(pid-file)=/var/run/named/named.pid
Recursion(recursion)=no
Session Key File(session-keyfile)=/var/run/named/session.key
Statistics File(statistics-file)=/var/named/log/named.stats
TCP Clients(tcp-clients)=500
Version(version)=TCPWave DNS Server
Zone Statistics(zone-statistics)=yes
Empty Zones Enable(empty-zones-enable)=no
Responses Per Second(responses-per-second)=0
Referrals Per Second(referrals-per-second)=0
NODATA Per Second(nodata-per-second)=0
NXDOMAINs Per Second(nxdomains-per-second)=0
Errors Per Second(errors-per-second)=0
All Per Second(all-per-second)=0
Window(window)=15
QPS Scale(qps-scale)=
IPv4 Prefix Length(ipv4-prefix-length)=24
IPv6 Prefix Length(ipv6-prefix-length)=56
Slip(slip)=0
Log Only(log-only)=yes
Exempt Clients(exempt-clients)=none;
Max Table Size(max-table-size)=500
Min Table Size(min-table-size)=500
RRSet Order(rrset-order)=cyclic
Sort List(sortlist)={192.168.2.23;{212.1.2.0/24;};{1.1.1.1;{24.234.4.56;12.34.67.0/24;}};}
Transfers In(transfers-in)=10

Transfers Out(transfers-out)=10
Transfers Per NS(transfers-per-ns)=2
Recursive Clients(recursive-clients)=20000
Forward(forward)=first
Forwarders(forwarders)=
Custom Parameters(custom-params)=
DNS Cookies(cookie-enabled)=yes

SAMPLE OUTPUT FILE CONTENTS FOR BIND CACHE TYPE:

Directory(directory)=/
Allow Query(allow-query)=any/Allow;
Allow Recursion(allow-recursion)=any/Allow;
Recursive Clients(recursive-clients)=20000
Allow Transfer(allow-transfer)=none/Allow;
Blackhole(blackhole)=none/Allow;
DNSSEC Enable(dnssec-enable)=yes
DNSSEC Validation(dnssec-validation)=yes
Listen On v6(listen-on-v6)=none
Check SRV CNAME(check-srv-cname)=ignore
Check MX CNAME(check-mx-cname)=ignore
Check MX(check-mx)=ignore
Check Names(check-names)=master ignore,response ignore
Dump File(dump-file)=/var/named/log/named_dump.db
Lame TTL(lame-ttl)=600
Max Negative Cache TTL(max-nocache-ttl)=10800
Minimal Responses(minimal-responses)=yes
PID File(pid-file)=/var/run/named/named.pid
Recursion(recursion)=yes
Session Key File(session-keyfile)=/var/run/named/session.key
Statistics File(statistics-file)=/var/named/log/named.stats
TCP Clients(tcp-clients)=150
Version(version)=TCPWave DNS Server

Zone Statistics(zone-statistics)=yes
Empty Zones Enable(empty-zones-enable)=no
Forward(forward)=first
Forwarders(forwarders)=
Responses Per Second(responses-per-second)=0
Referrals Per Second(referrals-per-second)=0
NODATA Per Second(nodata-per-second)=0
NXDOMAINs Per Second(nxdomains-per-second)=0
Errors Per Second(errors-per-second)=0
All Per Second(all-per-second)=0
Window(window)=15
QPS Scale(qps-scale)=
IPv4 Prefix Length(ipv4-prefix-length)=24
IPv6 Prefix Length(ipv6-prefix-length)=56
Slip(slip)=0
Log Only(log-only)=no
Exempt Clients(exempt-clients)=none;
Max Table Size(max-table-size)=500
Min Table Size(min-table-size)=500
Custom Parameters(custom-params)=
DNS Cookies(cookie-enabled)=no

SAMPLE OUTPUT FILE CONTENTS FOR UNBOUND TYPE:

Extended Statistics(extended-statistics)=yes
Interface(interface)=0.0.0.0
Outgoing Number of TCP(outgoing-num-tcp)=50
Incoming Number of TCP(incoming-num-tcp)=50
SO_RCVBUF(so-rcvbuf)=4m
EDNS Buffer Size(edns-buffer-size)=4096
Access Control(access-control)=0.0.0.0/0 allow
Message Buffer Size(msg-buffer-size)=65552
Message Cache Size(msg-cache-size)=8m

Number Of Queries Per Thread(num-queries-per-thread)=1024

Do IP4(do-ip4)=yes

Do IP6(do-ip6)=no

Do UDP(do-udp)=yes

Do TCP(do-tcp)=yes

Do Daemonize(do-daemonize)=yes

CHROOT(chroot)=/opt/tcpwave/etc/unbound

Username(username)=twcadmin

Directory(directory)=/opt/tcpwave/etc/unbound

Use Syslog(use-syslog)=no

Log File(logfile)=/var/log/twcdns.log

PID File(pidfile)=/opt/tcpwave/etc/unbound/unbound.pid

Root Hints(root-hints)=/opt/tcpwave/etc/unbound/db.cache

Hide Version(hide-version)=yes

Harden Glue(harden-glue)=yes

Log Time ASCII(log-time-ascii)=yes

Private Address(private-address)=1.0.0.0/8

Local Zone(local-zone)=10.in-addr.arpa nodefault

Cache Max Negative TTL(cache-max-negative-ttl)=3600

Module Config(module-config)=validator iterator

Module Config(module-config)=first

Module Config(module-config)=

Custom Parameters(custom-params)=

DNS Cookies(cookie-enabled)=no

SAMPLE OUTPUT FILE CONTENTS FOR NSD TYPE:

Server Count(server-count)=1

Hide Version(hide-version)=no

Version(version)=NSD

Identity(identity)=unidentified server

NSID(nsid)=aabbccdd

TCP Count(tcp-count)=100

TCP Query Count(tcp-query-count)=0
TCP Timeout(tcp-timeout)=120
IPv4 EDNS Size(ipv4-edns-size)=4096
Transfer Reload Timeout(xfrd-reload-timeout)=1
Ascii Log Time(log-time-ascii)=yes
Round Robin(round-robin)=no
Zone Files Check(zonefiles-check)=yes
Zone Files Write Seconds(zonefiles-write)=3600
RRL Size(rrl-size)=1000000
RRL Rate Limit(rrl-ratelimit)=200
RRL Slip(rrl-slip)=2
RRL IPv4 Prefix Length(rrl-ipv4-prefix-length)=24
RRL IPv6 Prefix Length(rrl-ipv6-prefix-length)=64
RRL Whitelist Rate limit(rrl-whitelist-ratelimit)=2000
Custom Parameters(custom-params)=
DNS Cookies(cookie-enabled)=no

SAMPLE OUTPUT FILE CONTENTS FOR DNS PROXY TYPE:

Directory(directory)=/
Allow Query(allow-query)=any/Allow;
Allow Recursion(allow-recursion)=any/Allow;
Allow Transfer(allow-transfer)=none/Allow;
Blackhole(blackhole)=none/Allow;
Listen On(listen-on)=127.0.0.1
Check SRV CNAME(check-srv-cname)=ignore
Check MX CNAME(check-mx-cname)=ignore
Check MX(check-mx)=ignore
Check Names(check-names)=master ignore,response ignore
Responses Per Second(responses-per-second)=0
Window(window)=15
Dump File(dump-file)=/var/named/log/named_dump.db
Lame TTL(lame-ttl)=600

Max Negative Cache TTL(max-nocache-ttl)=10800
Minimal Responses(minimal-responses)=yes
PID File(pid-file)=/var/run/named/named.pid
Recursion(recursion)=no
Session Key File(session-keyfile)=/var/run/named/session.key
Statistics File(statistics-file)=/var/named/log/named.stats
TCP Clients(tcp-clients)=150
Version(version)=TCPWave DNS Server
Zone Statistics(zone-statistics)=yes
Empty Zones Enable(empty-zones-enable)=no
Custom Parameters(custom-params)=
DNS Cookies(cookie-enabled)=no

getdnsserver**NAME**

getdnsserver

DESCRIPTION

Get the configuration of a DNS appliance from the TCPWave IPAM and write the contents to a file.

ARGUMENTS**--ip**

IP address of the DNS appliances. [mandatory]

--output_file

Full path to the output file to which the appliance configuration is to be written. If the file path is not specified, the output is written to the standard output.

EXAMPLE

```
twc getdnsserver --ip=10.1.10.174 --output_file=/tmp/dnsserver.txt
```

FILE FORMAT

The output file format is as follows:

Each configuration section begins with a [<section name>] field followed by <name>=<value> pairs one per line.

SECTIONS & CONFIGURATION PARAMETERS:**[dns-server]**

TYPE Type is one of 'BIND AUTH' or 'BIND CACHE' or 'UNBOUND' or 'NSD' or 'DNS PROXY'

OPTION_TEMPLATE DNS Option template name

APPLIANCE_TEMPLATE DNS appliance template

IP_ADDRESS IP address of the appliance

ORGANIZATION_NAME Organization Name of the DNS appliance

APPLIANCE_GROUP Name of the appliance group

ENABLE_MONIT '0' to enable monitoring and '1' to disable monitoring

INTERNAL_CACHE Applicable for appliances of type 'BIND CACHE' and 'BIND AUTH'. '0' indicates that the appliance is rooted at an internal root appliance. '1' indicates that the appliance is rooted at public internet root appliance.

DMZ_VISIBLE When a cache appliance is root to a public internet root appliance '1' indicates visibility of internal zones, '0' indicates internal zones are not visible. This flag is not applicable for cache appliances rooted at an internal root appliance

FIREWALL_TEMPLATE Firewall template name

DESCRIPTION	DNS appliance description
TIME_ZONE	Time zone
STEALTH_APPLIANCE	Applicable for appliances of type 'BIND AUTH'. Accepts 1 or 0 only. '1' indicates that the appliance can act as a stealth appliance. This option cannot be enabled when ENABLE_RECURSION option is set to 'yes'.
ENABLE_RECURSION	Applicable for appliances of type 'BIND AUTH'. 'yes' indicates that the appliance will act as a recursive appliance. This option cannot be enabled when STEALTH_APPLIANCE option is set to '1'.
RPZ_TEMPLATE	Response policy zone(RPZ) template name. Applicable when INTERNAL_CACHE is '0' and TYPE is 'BIND CACHE'.
[ntp]	
NTP_SERVERS	comma separated list of IP addresses of NTP servers
UPSTREAM	To authenticate with the NTP Server, user need to enable Upstream Authentication and fill the following details in the given format.
<IP>-<Key>-<SHA1>,<IP>-<Key>-<SHA1>	
Ex: 192.168.0.10-1-zxcvqwer,192.168.0.11-2-asdflkjhg	
DOWNSTREAM	Comma separated NTP Keys and the sha1, sha1 will auto generate if not specified. Keys of downstream should not be same in the upstream authentication key.
<key>-<sha1>,<key>-<sha1>,<key>	
EX: 2-262f8ff934271eea15f68b5c7481935e5f00fb3b, 3-595c0bcd44c76232315a9bd6b5cd0de1cd78d40a,5	
[dns-resolver]	
NAME_APPLIANCES	comma separated list of IP addresses of DNS appliances
SEARCH_SUFFIXES	comma separated list of domain names
[snmp]	
TRAP_SINK_1	IP address of SNMP trap sink
TRAP_SINK_2	IP address of SNMP trap sink
COMMUNITY_STRING	Community string for SNMP
SYSTEM_LOCATION	System Location
SYSTEM_CONTACT	System contact
PROCESS_LIST	comma separated list of processes being monitored. The following is a valid list of processes:
ntpd, dns, bgpd, zebra, crond, sshd, monit, syslog-ng,dhcpd	

ENABLE_SNMPV3 Takes 'true' or 'false'. 'true' indicates that SNMPv3 is enable. 'false' indicates that SNMPv3 is disabled.

FIREWALL_SNMP_ACL Name of the SNMP ACL.

[snmpv3]

USER_NAME User name of SNMPv3

AUTHENTICATION_PASSWORD Authentication password of the specified user

APPROVE_PASSWORD Approve password of the specified user

AUTHENTICATION_PROTOCOL Authentication protocol

ENCRYPTION_PROTOCOL Encryption protocol

[ldap-ssh]

ENABLE_LDAP_SSH Takes '0' or '1'. '1' indicates that enable LDAP Authentication on appliance. '0' indicates that disable LDAP Authentication on appliance.

[tacacs]

ENABLE_TACACS Takes '0' or '1'. '1' indicates TACACS+ configuration should be enabled for this appliance. '0' indicates TACACS+ configuration should be disabled

TACACS_PASSKEY TACACS passkey

TACACS_SERVERS Comma separated list of TACACS servers.

[syslogng-global-options]

TIME_REOPEN The time to wait in seconds before a dead connection is reestablished. Takes a value less than or equal to 32767.

TIME_REUSE If no new messages are written to a destination within the specified time in seconds, the connection will be closed, and its state will be freed. Takes a value less than or equal to 32767.

FLUSH_LINES Specifies how many lines are flushed to a destination at a time. Takes a value less than or equal to '32767'.

STATS_FREQ Syslog-*ng* OSE periodically sends a log statistics message. Takes a value less than or equal to 32767.

LOG_FIFO_SIZE The number of messages that the output queue can store. Takes a value less than or equal to 32767.

LOG_MSG_SIZE The maximal length of the log messages is limited by this option It is not recommended to set the option value higher than 10 MiB. Takes a value less than or equal to 32767.

KEEP_TIMESTAMP Specifies whether Syslog-*ng* should accept the timestamp received from the sending application or client. Takes value 'Yes' or 'No'.

[syslogng-source]

SOURCE_NAME	Name of the Source
INTERNAL_MSG	Internal syslog-NG message, takes input values as '0' or '1'. Default value is '1'.
SYSTEM_MSG	System specific log message, takes input value as '0' or '1'.
MSG_TXT_FILE	Message from text file, takes the file name as input.
MSG_MULTI_TXT_FILE	Message from multiple text files, takes input '0' or '1'. If this flag is '1' need to specify the FILE_PATH and FILE_PATTERN.
FILE_PATH	File path to the multiple text file.
FILE_PATTERN	File name pattern.
SYSLOG_SERVER	Syslog-NG sever, takes the input as '0' or '1'.
IP_ADDRESS	IP address of the syslog server.
PORT	Port number of the syslog server.
NETWORK_PROTOCOL	Network protocol, supports 'UDP' and 'TCP'.
[syslogng-filter]	
FILTER_NAME	Name of the Filter.
CONDITION	Takes the input as 'complex' or 'simple'.
FACILITIES	Allow values are one or more comma separated option given below. auth, authpriv, cron, deamon, kern, lpr, mail,mark, news, syslog, user, uucp, local0, local1,local2, local3, local4, local5, local6, local7.
PRIORITIES	Allow values are one or more comma separated option given below. info, notice, warning, err, crit, alert, emerg.
HOST_NAME	Name of the host.
IP_NETWORK	IP address with mask length.
MATCH_EXPRESSION	Match expression.
PROGRAM	Program.
[syslogng-destination]	
DESTINATION_NAME	Name of the destination.
TYPE_SNG	Type of the destination. Takes the value between 1 to 5 '1'= File '2'= Named pipe '3'= Local Users '4'= All logged-in users '5'= Syslog server.

LOG_FILE_NAME File name to log the message, mandatory when TYPE_SNG is specified as '1'.
NAMED_PIPE_NAME Named pipe name, mandatory when TYPE_SNG is specified as '2'.
LOCAL_USERS Local users, mandatory when TYPE_SNG specified as '3'.
SYSLOG_SERVER IP address of the syslog server, mandatory when TYPE_SNG is specified as '5'.
PORT Port number of the syslog server, mandatory when TYPE_SNG is specified as '5'.
NETWORK_PROTOCOL Network protocol, supports 'UDP' and 'TCP', mandatory when TYPE_SNG is specified as '5'.

[syslogng-target]

SOURCE Name of the source.
FILTER Name of the filter.
DESTINATION Name of the destination.
[view]
NAMES Comma separated list of DNS view names. Sequence of views are ordered from left to right in ascending order

[banner]

Banner title of the appliance.

SAMPLE FILE CONTENTS:

```
[dns-server]
TYPE=BIND AUTH
OPTION_TEMPLATE=testdns
APPLIANCE_TEMPLATE=ISC BIND Authoritative Appliance Template
IP_ADDRESS=10.1.10.201
ORGANIZATION_NAME=TCPWave
APPLIANCE_GROUP=Appliance-Group
ENABLE_MONIT=1
FIREWALL_TEMPLATE=Default_Firewall
DESCRIPTION=Root
TIME_ZONE=America/New_York (Eastern Time)
```

[ntp]

```
NTP_SERVERS=17.253.68.253,17.253.16.243,17.253.80.243,17.253.6.243,
```

UPSTREAM=

DOWNSTREAM=

[dns-resolver]

NAME_APPLIANCES=8.8.8.8

SEARCH_SUFFIXES=tcpwave.com

[snmp]

TRAP_SINK_1=194.41.67.51

TRAP_SINK_2=194.41.65.177

COMMUNITY_STRING=sph1nkx5

SYSTEM_LOCATION=Datacenter for systematic trading infrastructure

SYSTEM_CONTACT=GNCC +1 877 462 2284

PROCESS_LIST=ntpd,dns,bgpd,zebra,crond,

ENABLE_SNMPV3=true

FIREWALL_SNMP_ACL=TestAcl

[snmpv3]

USER_NAME=admin

AUTHENTICATION_PASSWORD=zxcv1234

APPROVE_PASSWORD=abc1234567

AUTHENTICATION_PROTOCOL=SHA

ENCRYPTION_PROTOCOL=AES

[ldap-ssh]

ENABLE_LDAP_SSH=1

[tacacs]

ENABLE_TACACS=1

```
TACACS_PASSKEY=abc123
TACACS_SERVERS=10.1.10.173,10.1.10.172,1.2.3.4,2.3.4.5,
```

```
[syslogng-global-options]
```

```
TIME_REOPEN=60
TIME_REAP=60
FLUSH_LINES=60
STATS_FREQ=600
LOG_FIFO_SIZE=1000
LOG_MSG_SIZE=2048
KEEP_TIMESTAMP=Yes
```

```
[syslogng-source]
```

```
SOURCE_NAME=s_sys
INTERNAL_MSG=1
SYSTEM_MSG=1
MSG_TXT_FILE=/var/tmp/mft.txt
MSG_MULTI_TXT_FILE=1
FILE_PATH=mfts.txt
FILE_PATTERN=/var/tmp
SYSLOG_SERVER=1
IP_ADDRESS=192.168.0.2
PORT=53
NETWORK_PROTOCOL=UDP
```

```
[syslogng-filter]
```

```
FILTER_NAME=f_default
CONDITION=complex
COMPLEX_CONDITION=level(info..emerg) and not (facility(mail) or facility(authpriv) or facility(cron))
```

```
[syslogng-destination]
```

```
DESTINATION_NAME=d_mesg  
LOG_FILE_NAME=messages  
TYPE_SNG=1  
ENABLE_SYNC=Yes
```

[syslogng-target]

```
SOURCE=s_sys  
FILTER=f_default  
DESTINATION=d_mesg
```

[view]

```
NAMES=view1,view2
```

[banner]

The default value of Banner configuration for DNS and DHCP appliances.

getdnsservermpl

NAME

getdnsservermpl

DESCRIPTION

Displays the contents of a DNS appliance template from the TCPWave IPAM.

ARGUMENTS

--name

Name of the DNS appliance template in TCPWave IPAM [mandatory]

--output_file

Full path to the file where the contents of the server template are written to. If this argument is not specified the contents are displayed on the standard output.

--format

Takes 'csv' or 'prop'. Specifies the format of the output. 'csv' displays the contents as a comma separated list of values. 'prop' displays the contents in the format name=value. 'csv' is the default format if this argument is not specified.

EXAMPLE

```
twc getdnsservertmpl --name="Bind Auth Server Template" --format=prop --  
output_file="/tmp/template.out"
```

```
twc getdnsservertmpl --name="Bind Auth Server Template" --format=csv --  
output_file="/tmp/template.out"
```

```
twc getdnsservertmpl --name="Bind Auth Server Template" --format=csv
```

getdnszone

NAME

getdnszone

DESCRIPTION

Displays the contents of a DNS zone definition from the TCPWave IPAM.

ARGUMENTS

--name

Name of the DNS zone in TCPWave IPAM. [mandatory]

--org

Name of the organization to which specified DNS zone belongs. [mandatory]

--output_file

Full path to the file where the contents of the DNS zone are written to. If this argument is not specified the contents are displayed on the standard output.

--format

Takes 'csv' or 'prop'. Specifies the format of the output. 'csv' displays the contents as a comma separated list of values. 'prop' displays the contents in the format name=value. 'csv' is the default format if this argument is not specified.

EXAMPLE

```
twc      getdnszone      --name=tcpwave.com      --org=TCPWave      --format=prop      --  
output_file="/tmp/dnszone.out"
```

```
twc getdnszone --name=tcpwave.com --org=TCPWave --format=csv
```

getfirewalltmpl**NAME:**

getfirewalltmpl

DESCRIPTION:

Displays the contents of a firewall template from the TCPWave IPAM.

ARGUMENTS:

--name

Name of the firewall template. [mandatory]

--org

Organization name under which the firewall template is being created.
This argument is mandatory if user is FADM.

--output_file

Full path to the output file to which the firewall template contents are to be written. If the file path is not specified, the output is written to the standard output.

EXAMPLE USAGE:

```
twc getfirewalltmpl --name=Defaults_Firewall --output_file=/tmp/firewalltmpl.txt --  
org=TCPWave
```

FILE FORMAT:

The output file format is as follows:

Each configuration section begins with a [<section name>] field followed by <name>=<value> pairs one per line

SAMPLE FILE CONTENTS:

[Firewall-Template]

Name = TCPWave-Firewall-Tmpl

Organization = TCPWave

Description = "TCPWave Default Firewall Template"

[Rule]

Name = Default

Action = ACCEPT

Chain = INPUT

Protocol = udp

Source = 192.168.1.4

Invert_Source = false

Source_Port = 1122

Invert_Source_Port = true

Destination =

Invert_Destination = false

Destination_Port =

Invert_Destination_Port = false

Incoming_Interface =

```
Outgoing_Interface =
Ethernet_Address =
Fragment = false
DNS = true
DNS_Query = true
DNS_Response = false
DNS_Query_Type = A
EDNS0 = true
EDNS0_Buffer_Size = 4321
Match_String =
Extension =
[Rule]
Name =
Action = ACCEPT
Chain = INPUT
Protocol = udp
Source = 10.1.10.0/24
Invert_Source = true
Source_Port =
Invert_Source_Port = false
Destination =
Invert_Destination = false
Destination_Port =
Invert_Destination_Port = false
Incoming_Interface =
Outgoing_Interface =
Ethernet_Address =
Fragment = false
DNS = true
DNS_Query = false
DNS_Response = false
DNS_Query_Type = ANY
EDNS0 = false
EDNS0_Buffer_Size =
Match_String =
Extension =
```

getipv6dnsserver

NAME

getipv6dnsserver

DESCRIPTION

Get the configuration of a IPv6 DNS appliance from the TCPWave IPAM and write the contents to a file.

ARGUMENTS

--ip

IPv6 address of the DNS appliances. [mandatory]

--output_file

Full path to the output file to which the appliance configuration is to be written. If the file path is not specified, the output is written to the standard output.

EXAMPLE:

```
twc getipv6dnsserver --ip=5000::2 --output_file=/tmp/ipv6dnsserver.txt
```

FILE FORMAT:

The output file format is as follows:

Each configuration section begins with a [<section name>] field followed by <name>=<value> pairs one per line.

SECTIONS & CONFIGURATION PARAMETERS:

[dns-server]

TYPE Type is one of 'BIND AUTH' or 'BIND CACHE' or 'UNBOUND' or
'NSD' or 'DNS PROXY'

OPTION_TEMPLATE DNS Option template name

APPLIANCE_TEMPLATE DNS appliance template name

IPV6_ADDRESS IPv6 address of the appliance

ORGANIZATION_NAME Organization Name of the DNS appliance

APPLIANCE_GROUP Name of the appliance group

ENABLE_MONIT '0' to enable monitoring and '1' to disable monitoring

INTERNAL_CACHE Applicable for appliances of type 'BIND CACHE' and
'BIND AUTH'.

'0' indicates that the appliance is rooted at an internal root appliance.

'1' indicates that the appliance is rooted at public internet root appliance.

DMZ_VISIBLE When a cache appliance is root to a public internet root appliance '1' indicates visibility of internal zones, '0' indicates internal zones are not visible. This flag is not applicable for cache appliances rooted at an internal root appliance.

FIREWALL_TEMPLATE Firewall template name

DESCRIPTION DNS appliance description

TIME_ZONE Time zone

STEALTH_APPLIANCE Applicable for appliances of type 'BIND AUTH'. Accepts 1 or 0 only.

'1' indicates that the appliance can act as a stealth appliance.

This option can not be enabled when **ENABLE_RECURSION** option is set to 'yes'.

ENABLE_RECURSION Applicable for appliances of type 'BIND AUTH'. 'yes' indicates that the appliance will act as a recursive appliance.

This option can not be enabled when **STEALTH_APPLIANCE** option is set to '1'.

RPZ_TEMPLATE Response policy zone(RPZ) template name. Applicable when INTERNAL_CACHE is '0' and TYPE is 'BIND CACHE', 'BIND AUTH + CACHE", UNBOUND.

NSM_TEMPLATE Network Security Monitoring (NSM) template name. Applicable for 'BIND CACHE', 'BIND AUTH + CACHE", UNBOUND appliances.

[ntp]

IPV6_NTP_SERVERS Comma separated list of IPv6 addresses of NTP servers

UPSTREAM To authenticate with the NTP Server, user needs to enable Upstream Authentication and fill the following details in the given format.

<IP>-<Key>-<SHA1>,<IP>-<Key>-<SHA1>

Ex: 5000::2-zxcvqwer,5000::5-2-asdflkjhg

DOWNSTREAM Comma separated NTP Keys and the sha1, sha1 will auto generate if not specified.

Keys of downstream should not be same in the upstream authentication key.

<key>-<sha1>,<key>-<sha1>,<key>

EX: 2-262f8ff934271eea15f68b5c7481935e5f00fb3b,

3-595c0bcd44c76232315a9bd6b5cd0de1cd78d40a,5

[dns-resolver]

IPV6_NAME_APPLIANCES Comma separated list of IPv6 addresses of DNS appliances

SEARCH_SUFFIXES Comma separated list of domain names

[snmp]

IPV6_TRAP_SINK_1 IPv6 address of SNMP trap sink

IPV6_TRAP_SINK_2 IPv6 address of SNMP trap sink

COMMUNITY_STRING Community string for SNMP

SYSTEM_LOCATION System Location

SYSTEM_CONTACT System contact

PROCESS_LIST comma separated list of processes being monitored. The following is a valid list of processes:

ntpd, dns, bgpd, zebra, crond, sshd, monit, syslog-ng,dhcpd

ENABLE_SNMPV3 Takes 'true' or 'false'. 'true' indicates that SNMPv3 is enabled. 'false' indicates that SNMPv3 is disabled.

FIREWALL_SNMP_ACL Name of the SNMP ACL.

[snmpv3]

USER_NAME User name of SNMPv3
AUTHENTICATION_PASSWORD Authentication password of the specified user
APPROVE_PASSWORD Approve password of the specified user
AUTHENTICATION_PROTOCOL Authentication protocol
ENCRYPTION_PROTOCOL Encryption protocol

[ldap-ssh]

ENABLE_LDAP_SSH Takes '0' or '1'. '1' indicates that enable LDAP Authentication on appliance. '0' indicates that disable LDAP Authentication on appliance.

[tacacs]

ENABLE_TACACS Takes '0' or '1'. '1' indicates TACACS+ configuration should be enabled for this appliance. '0' indicates TACACS+ configuration should be disabled
TACACS_PASSKEY TACACS passkey
IPV6_TACACS_SERVERS Comma separated list of TACACS servers.

[syslogng-global-options]

TIME_REOPEN The time to wait in seconds before a dead connection is reestablished. Takes a value less than or equal to 32767.

TIME_REAP If no new messages are written to a destination within the specified time in seconds, the connection will be

closed, and its state will be freed. Takes a value less than or equal to 32767.

FLUSH_LINES Specifies how many lines are flushed to a destination at a time. Takes a value less than or equal to '32767'.

STATS_FREQ Syslog-ng OSE periodically sends a log statistics message. Takes a value less than or equal to 32767.

LOG_FIFO_SIZE The number of messages that the output queue can store. Takes a value less than or equal to 32767.

LOG_MSG_SIZE The maximal length of the log messages is limited by this option It is not recommended to set the option value higher than 10 MiB. Takes a value less than or equal to 32767.

KEEP_TIMESTAMP Specifies whether Syslog-ng should accept the timestamp received from the sending application or client.

Takes value 'Yes' or 'No'.

[syslogng-source]

SOURCE_NAME Name of the Source

INTERNAL_MSG Internal syslog-NG message, takes input values as '0' or '1'. Default value is '1'.

SYSTEM_MSG System specific log message, takes input value as '0' or '1'.

MSG_TXT_FILE Message from text file, takes the file name as input.

MSG_MULTI_TXT_FILE Message from multiple text files, takes input '0' or '1'. If this flag is '1' need to specify the **FILE_PATH** and **FILE_PATTERN**.

FILE_PATH File patch to the multiple text file.

FILE_PATTERN File name pattern.

SYSLOG_SERVER Syslog-NG sever, takes the input as '0' or '1'.

IP_ADDRESS IP address of the syslog server.

PORT Port number of the syslog server.

NETWORK_PROTOCOL Network protocol, supports 'UDP' and 'TCP'.

[syslogng-filter]

FILTER_NAME Name of the Filter.

CONDITION Takes the input as 'complex' or 'simple'.

FACILITIES Allow values are one or more comma separated option given below.

auth, authpriv, cron, deamon, kern, lpr, mail,mark, news,
syslog, user, uucp, local0, local1,local2, local3,
local4, local5, local6, local7.

PRIORITIES Allow values are one or more comma separated option given below.

info, notice, warning, err, crit, alert, emerg.

HOST_NAME Name of the host.

IP_NETWORK IP address with mask length.

MATCH_EXPRESSION Match expression.

PROGRAM Program.

[syslogng-destination]

DESTINATION_NAME Name of the destination.

TYPE_SNG Type of the destination. Takes the value between 1 to 5

'1'= File

'2'= Named pipe

'3'= Local Users

'4'= All logged-in users

'5'= Syslog server.

LOG_FILE_NAME File name to log the message, mandatory when TYPE_SNG is specified as '1'.

NAMED_PIPE_NAME Named pipe name, mandatory when TYPE_SNG is specified

as '2'.

LOCAL_USERS Local users, mandatory when TYPE_SNG specified as '3'.

SYSLOG_SERVER IP address of the syslog server, mandatory when TYPE_SNG is specified as '5'.

PORT Port number of the syslog server, mandatory when TYPE_SNG is specified as '5'.

NETWORK_PROTOCOL Network protocol, supports 'UDP' and 'TCP', mandatory when TYPE_SNG is specified as '5'.

[syslogng-target]

SOURCE Name of the source.

FILTER Name of the filter.

DESTINATION Name of the destination.

[view]

NAMES Comma separated list of DNS views names. Sequence of views are ordered from left to right in ascending order

[banner]

Banner title of the appliance.

SAMPLE FILE CONTENTS:

[dns-server]

TYPE=BIND AUTH

OPTION_TEMPLATE=testdns

```
APPLIANCE_TEMPLATE=ISC BIND Authoritative Appliance Template
IPV6_ADDRESS=5000::3
ORGANIZATION_NAME=TCPWave
APPLIANCE_GROUP=Appliance-Group
ENABLE_MONIT=1
FIREWALL_TEMPLATE=Default_Firewall
NSM_TEMPLATE=TCPWAVE-NSM
DESCRIPTION=Root
TIME_ZONE=America/New_York (Eastern Time)
```

[ntp]

```
IPV6_NTP_SERVERS=5000::2,3000::5
UPSTREAM=
DOWNSTREAM=
```

[dns-resolver]

```
IPV6_NAME_APPLIANCES=5000::2
SEARCH_SUFFIXES=tcpwave.com
```

[snmp]

```
IPV6_TRAP_SINK_1=5000::2
IPV6_TRAP_SINK_2=3000::5
COMMUNITY_STRING=sph1nkx5
SYSTEM_LOCATION=
SYSTEM_CONTACT=
PROCESS_LIST=ntpd,dns,bgpd,zebra,crond,
ENABLE_SNMPV3=true
FIREWALL_SNMP_ACL=TestAcl
```

[snmpv3]

```
USER_NAME=admin  
AUTHENTICATION_PASSWORD=zxcv1234  
APPROVE_PASSWORD=abc1234567  
AUTHENTICATION_PROTOCOL=SHA  
ENCRYPTION_PROTOCOL=AES
```

[ldap-ssh]

```
ENABLE_LDAP_SSH=1
```

[tacacs]

```
ENABLE_TACACS=1  
TACACS_PASSKEY=abc123  
IPV6_TACACS_SERVERS=3000::2,5000::5
```

[syslogng-global-options]

```
TIME_REOPEN=60  
TIME_REAP=60  
FLUSH_LINES=60  
STATS_FREQ=600  
LOG_FIFO_SIZE=1000  
LOG_MSG_SIZE=2048  
KEEP_TIMESTAMP=Yes
```

[syslogng-source]

```
SOURCE_NAME=s_sys
INTERNAL_MSG=1
SYSTEM_MSG=1
MSG_TXT_FILE=/var/tmp/mft.txt
MSG_MULTI_TXT_FILE=1
FILE_PATH=mfts.txt
FILE_PATTERN=/var/tmp
SYSLOG_SERVER=1
IP_ADDRESS=5000::3
PORT=53
NETWORK_PROTOCOL=UDP
```

[syslogng-filter]

```
FILTER_NAME=f_default
CONDITION=complex
COMPLEX_CONDITION=level(info..emerg) and not (facility(mail) or facility(authpriv) or facility(cron))
```

[syslogng-destination]

```
DESTINATION_NAME=d_mesg
LOG_FILE_NAME=messages
TYPE_SNG=1
ENABLE_SYNC=Yes
```

[syslogng-target]

```
SOURCE=s_sys
FILTER=f_default
DESTINATION=d_mesg
```

[view]

NAMES=view1,view2

[banner]

The default value of Banner configuration for DNS and DHCP appliances.

getipv6dhcpserver

NAME:

getipv6dhcpserver

DESCRIPTION:

Displays the configuration for an IPv6 DHCP Server in the TCPWave IPAM.

ARGUMENTS:

--ip

IPv6 Address of the IPv6 DHCP server. [mandatory]

--output_file

Full path to the output file to which the server configuration is to be written. If the file path is not specified, the output is written to the standard output.

EXAMPLE USAGE:

tvc getipv6dhcpserver --ip=2001:db8::4 --output_file=/tmp/dhcpserver.txt

SAMPLE OUTPUT FILE CONTENTS:

[dhcp-server]

IPv6_ADDRESS=2001:db8::4

ORGANIZATION_NAME=TCPWave

POLICY_TEMPLATE=Clone-Policy-1

IPv4_DHCP_APPLIANCE=10.1.10.181

ENABLE_MONIT=1

TIME_ZONE=GMT (GMT)

[ntp]

NTP_SERVERS=10.1.10.10,10.1.10.11,10.1.10.12,10.1.10.13,

[snmp]

TRAP_SINK_1=10.1.10.15

TRAP_SINK_2=10.1.10.18

COMMUNITY_STRING=public

SYSTEM_LOCATION=

SYSTEM_CONTACT=

PROCESS_LIST=dns,

[syslog]

DESTINATION_TYPE=F

DESTINATION=/var/log/messages

ACTIVE=1

```
SELECTORS=*.info;mail.none;authpriv.none;cron.none
[syslog]
DESTINATION_TYPE=F
DESTINATION=/var/log/secure
ACTIVE=1
SELECTORS=authpriv.*
[syslog]
DESTINATION_TYPE=F
SYNC_EACH_MESSAGE=1
DESTINATION=/var/log/maillog
ACTIVE=1
SELECTORS=mail.*
[syslog]
DESTINATION_TYPE=F
DESTINATION=/var/log/cron
ACTIVE=1
SELECTORS=cron.*
[syslog]
DESTINATION_TYPE=AU
DESTINATION=
ACTIVE=1
SELECTORS=*.emerg
[syslog]
DESTINATION_TYPE=F
DESTINATION=/var/log/spooler
ACTIVE=1
SELECTORS=uucp,news.crit
[syslog]
DESTINATION_TYPE=F
DESTINATION=/var/log/boot.log
ACTIVE=1
SELECTORS=local7.*
[macexclusions]
MAC=8E-2C-E7-88-53-7A
DESCRIPTION=
[macexclusions]
MAC=E0:8F:8D:59:CF:60
DESCRIPTION=
```

getipv6dhcptmpl

NAME:

getipv6dhcptmpl

DESCRIPTION:

Displays the contents of an IPv6 DHCP option template from the TCPWave IPAM.

ARGUMENTS:

--name

name of the IPv6 DHCP option template to be retrieved from TCPWave IPAM
[mandatory]

--org

Name of the organization. [mandatory]

--output_file

full path to the output file to which the template contents are to be written. If the file path is not specified, the output is written to the standard output.

--display

takes 'all' or 'used'. 'used' will display/output only used options.
'all' will display all the options including unused options. [mandatory]

EXAMPLE USAGE:

```
twc getipv6dhcptmpl --name="voip devices template" --org= TcpWave --  
output_file="/tmp/getipv6dhcptmpl.txt" --display=all
```

SAMPLE OUTPUT:

CLIENTID = SN0001

SERVERID = SVRNO002

SIP SERVER D = abc.com

NIS DOMAIN NAME = tcpwave.com

IA NA =

IA TA =

IAADDR =

ORO =

PREFERENCE =

UNICAST =

RAPID COMMIT =

VENDOR OPTS =

INTERFACE ID =

RECONF ACCEPT =

SIP SERVER A =

DNS SERVERS =

DOMAIN LIST =

NIS SERVERS =

NISP SERVERS =

NISP DOMAIN NAME =

SNTP SERVERS =

INFORMATION REFRESH TIME =

BCMCS SERVER D =

BCMCS SERVER A =

SUBSCRIBER ID =

CLIENT FQDN =

getfreesubnetlist**NAME**

getfreesubnetlist

DESCRIPTION

Lists all the free subnets within a given network in the TCPWave IPAM.

ARGUMENTS**--ip**

IP address of the network for which subnets are to be listed.[mandatory]

--org

Organization name associated with network IP. [mandatory]

--d

Delimiter character separating the columns. If this argument is not specified, then comma will be used as a delimiter.

EXAMPLE

```
twc getfreesubnetlist --ip=10.0.10.0 --org=Internal --d=,
```

getnextfreeip**NAME**

getnextfreeip

DESCRIPTION

Displays the next available free IP address from a given subnet in the TCPWave IPAM. To get the free IP address from the specific range provide start_ip and end_ip address.

ARGUMENTS**--subnet**

Subnet address to get the next available free IP from [mandatory]

--start_ip

Start IP address to get the next available free IP.

--end_ip

End IP address to get the next available free IP.

--org

Name of the organization to which specified subnet belongs. This argument is mandatory if the user is 'FADM'.

EXAMPLE

```
twc getnextfreeip --subnet=10.1.10.0 --org=TCPWave
```

```
twc getnextfreeip --subnet=10.1.10.0 --start_ip=10.1.10.5 --end_ip=10.1.10.10 --org=TCPWave
```

getnonmanageddnsmaster**NAME**

getnonmanageddnsmaster

DESCRIPTION

Displays the content of a non-managed DNS master from the TCPWave IPAM. Non-Managed DNS Master is a Name to IP and IP to Name resolution service. Non-Managed DNS Master is partially managed by TCPWave IPAM.

ARGUMENTS

--ip

IP address of the non-managed DNS master. [mandatory]

--type

Type of the non-managed DNS master. It takes input as 'Power DNS' or 'External DNS'. [mandatory]

--org

Organization name associated with the non-managed DNS master. [mandatory]

--output_file

Full path to the output file to which the appliance configuration is to be written. If the file path is not specified, the output is written to the standard output.

--format

Takes 'csv' or 'prop'. Specifies the format of the output. 'csv' displays the contents as a comma separated list of values. 'prop' displays the contents in the format name=value. 'csv' is the default format if this argument is not specified.

EXAMPLE

```
twc getnonmanageddnsmaster --ip=10.0.0.123 --type='External DNS' --org=TCPWave --format=prop --output_file=/tmp/externaldnsmaster.txt
```

```
twc getnonmanageddnsmaster --ip=10.0.0.123 --type='External DNS' --org=TCPWave --format=csv --output_file=/tmp/externaldnsmaster.txt
```

getobject**NAME:**

getobject

DESCRIPTION:

Displays the contents of an object definition from the TCPWave IPAM.

ARGUMENTS:

--object

IP Address of the object in TCPWave IPAM [mandatory]

--org

Name of the organization to which specified, object belongs. This argument is mandatory if the user is 'FADM'.

--output_file

Full path to the file where the contents of the object are written to. If this argument is not specified, the contents are displayed on the standard output.

--format

Takes 'csv' or 'prop'. Specifies the format of the output. 'csv' displays the contents as a comma separated list of values. 'prop' displays the contents in the format name=value.
'csv' is the default format if this argument is not specified.

Name

getobject CLI to search on object name

EXAMPLE USAGE:

```
twc getobject --object="9.0.0.4" --org=TCPWave --format=prop --output_file="/tmp/object.out"
```

```
twc getobject --object="9.0.0.4" --org=TCPWave --format=csv --output_file="/tmp/object.out"
```

```
twc getobject --object="9.0.0.4" --org=TCPWave --format=csv
```

getrootaccessmgmt

NAME:

getrootaccessmgmt

DESCRIPTION:

Gets the Vault Type configuration details set up in root access management in the TCPWave IPAM.

ARGUMENTS:

- NA

EXAMPLE USAGE:

```
twc getrootaccessmgmt --d=,
```

getsubnet

NAME:

getsubnet

DESCRIPTION:

Displays the details of a given subnet in the TCPWave IPAM.

ARGUMENTS:

--subnet

IP address of the subnet. [mandatory]

--org

Name of the organization to which specified, subnet belongs. This argument is mandatory if the user is 'FADM'.

--output_file

Full path to the file where the contents of the subnet are written to. If this argument is not specified, the contents are displayed on the standard output.

--format

Takes 'csv' or 'prop'. Specifies the format of the output. 'csv' displays the contents as a comma separated list of values. 'prop' displays the contents in the format name=value.
'csv' is the default format if this argument is not specified.,

EXAMPLE USAGE:

```
twc getsubnet --subnet="9.1.0.0" --org=TCPWave --format=prop --  
output_file="/tmp/subnet.prop"
```

```
twc getsubnet --subnet="9.1.0.0" --org=TCPWave --format=csv --output_file="/tmp/subnet.csv"
```

```
twc getsubnet --subnet="9.1.0.0" --org=TCPWave --format=csv
```

getipv6subnettmpl

NAME

getipv6subnettmpl

DESCRIPTION

Gets the IPv6 subnet template details created in the TCPWave IPAM. To get the IPv6 subnet template user needs to provide the name of the subnet template and organization.

ARGUMENTS

--name

Name of the IPv6 subnet template [mandatory].

--org

Name of the organization [mandatory].

--output_file

Full path to the output file to which the subnet template configuration

is to be written. If the file path is not specified, the output is written to the standard output.

EXAMPLE

```
twc getipv6subnettmpl --name="Subnet Template" --org="Internal" --output_file=/tmp/subnettmpl.txt
```

getperfmetricstatistics

NAME

getperfmetricstatistics

DESCRIPTION

Gets the performance metrics statistics details from the TCPWave IPAM.

ARGUMENTS

--ip

IP address of the appliance for which the performance metrics statistics are to be listed.
[mandatory]

--type

Type of the appliance, It accepts the DNS or DHCP or IPAM. [mandatory]

--metric_name

Name of the performance metric, Below are supported metric names

Basic Properties

CPU Utilization, Memory Utilization, Disk Utilization, Swap Memory, ICMP and Heart Beat.

DNS Query

A, AAAA, CNAME, SOA, MX, PTR, SRV and NS.

DNS Response

SUCCESS, SRVFAIL, FORMERR, NXDOMAIN and RECURSION.

NTP

NTP Delay and NTP Offset.

DHCP Query

Requests, Offer, Release, Inform, Decline, Discover and Acknowledge.

DHCP Response

DHCP Response.

Heap

Heap.

Database Summary

Opened Tables, Slow Queries, Open Tables, Queries and Questions.

Database Threads

Connected, Running, Created and Cached.

Database Connections

Used Connections, Aborted Clients and Aborted Connections.

Database Network Traffic

RX and TX.

Database Handler

Commit, Read First, Read Key, Read Rnd, Read Next, Rollback, Write and Created.

Database Row Operations

Deleted, Insert, Read and Update.

Database Table Locks

Immediate, Waited and Slow Queries.

--duration

Type of the duration from which the performance metrics statistics are to be listed.

It accepts the 'd' or 'w' or 'm'. [mandatory]

i.e d = Daily, w = weekly and m = monthly.

EXAMPLE:

```
twc getperfmetricstatistics --ip=50.0.0.0 --type=DNS --metric_name="CPU Utilization" --duration=d
```

```
twc getperfmetricstatistics --ip=50.0.0.0 --type=DHCP --metric_name="DHCP Response" --duration=d
```

```
twc getperfmetricstatistics --ip=10.0.1.24 --type=IPAM --metric_name="Heap" --duration=d
```

getsubnetmpl

NAME

getsubnetmpl

DESCRIPTION

Gets the IPv4 subnet template details created in the TCPWave IPAM. To get the IPv4 subnet template user needs to provide name of the subnet template and organization.

ARGUMENTS

--name

Name of the IPv4 subnet template.

--org

Name of the organization.

--output_file

Full path to the output file to which the subnet template configuration is to be written. If the file path is not specified, the output is written to the standard output.

EXAMPLE

```
twc      getsubnetmpl      --name="Subnet      Template"      --org="Internal"      --
output_file=/tmp/subnettmpl.txt
```

getzoneacl

NAME:

getzoneacl

DESCRIPTION:

Gets the ACL list assigned to the zone in the TCPWave IPAM.
To get the ACL list assigned to the zone, user needs to provide name of the zone and organization.

ARGUMENTS:

- zone_name
Name of the zone.
- org
Name of the organization.

EXAMPLE USAGE:

```
twc getzoneacl --zone_name="tcp.com" --org="Internal"
```

```
twc getzoneacl --zone_name="168.192.in-addr.arpa" --org="Internal"
```

getremotedebug**NAME**

getremotedebug

DESCRIPTION

Displays the DNS or DHCP remote debugging status from the TCPWave IPAM.

ARGUMENTS:

- ip
IP address of the DNS or DHCP appliance. [mandatory]
- type
Type of the appliance. It takes 'DHCP' or 'DNS' as a input. [mandatory]

EXAMPLE

```
twc getremotedebug --ip=10.1.10.240 --type=DNS
```

```
twc getremotedebug --ip=10.1.10.240 --type=DHCP
```

getnonmanageddnsmaster**NAME**

getnonmanageddnsmaster

DESCRIPTION

Displays the content of a non-managed DNS master from the TCPWave IPAM. Non-Managed DNS Master is a Name to IP and IP to Name resolution service. Non-Managed DNS Master is partially managed by TCPWave IPAM.

ARGUMENTS

--ip

IP address of the non-managed DNS master. [mandatory]

--type

Type of the non-managed DNS master. It takes input as 'Power DNS' or 'External DNS'. [mandatory]

--org

Organization name associated with the non-managed DNS master. [mandatory]

--output_file

Full path to the output file to which the appliance configuration is to be written. If the file path is not specified, the output is written to the standard output.

--format

Takes 'csv' or 'prop'. Specifies the format of the output. 'csv' displays the contents as a comma separated list of values. 'prop' displays the contents in the format name=value. 'csv' is the default format if this argument is not specified.

EXAMPLE

```
twc getnonmanageddnsmaster --ip=10.0.0.123 --type='External DNS' --org=TCPWave --format=prop --output_file=/tmp/externaldnsmaster.txt
```

```
twc getnonmanageddnsmaster --ip=10.0.0.123 --type='External DNS' --org=TCPWave --format=csv --output_file=/tmp/externaldnsmaster.txt
```

Imports***importmicrosoftdnsserver*****DESCRIPTION:**

Imports the Microsoft DNS appliance from a CSV file into the TCPWave IPAM

ARGUMENTS:

--input_file

Path on the target IPAM server to the input csv file to import the

Microsoft DNS appliance from. [Mandatory]

--output_file

Path on the target IPAM server to the output file to write the results
Of the import. [Mandatory]

--error_file

Path on the target IPAM server to the file to write the records that
Failed to import. [Mandatory]

--max_errors

Maximum permissible errors. The import is aborted if the error count
Reaches this value. [Mandatory]

EXAMPLE:

```
twc importmicrosoftdnsserver --input_file=/tmp/msdnsserver.txt --output_file=/tmp/output --error_file=/tmp/error --max_errors=10
```

IMPORT FILE:

The information to create a microsoft DNS appliance should be specified as a
comma separated values as specified in the order below.

"NAME","ORG_NAME","IP_ADDR","USER_NAME","PASSWORD","DESCRIPTION","IS_HTTPS"

EXAMPLE DATA:

"Microsoft00001Remote","TcpWave","1.0.0.20","Twcadm","49C5ECBC1DA7F46085CB73F1B5BF1B
6000","",""1"

importadmin

NAME:

importadmin

DESCRIPTION:

Imports the administrators from a csv file in the TCPWave IPAM.

ARGUMENTS:

--input_file

Path on the target IPAM server to the input csv file to import the
administrators from. [mandatory]

--output_file

Path on the target IPAM server to the output file to write the results
of the import. [mandatory]

--error_file

Path on the target IPAM server to the file to write the records that
failed to import. [mandatory]

--max_errors

Maximum permissible errors. The import is aborted if the error count reaches this value. [mandatory]

EXAMPLE USAGE:

```
twc importadmin --input_file=/tmp/admin.csv --output_file=/tmp/output --error_file=/tmp/error  
--max_errors=100
```

IMPORT FILE:

The information to create an administrator should be specified, as a comma separated values as specified, in the order below

"ORG_NAME","FIRST_NAME","MIDDLE_NAME","LAST_NAME","EMAIL","PHONE","LOGIN_NAME","ADMIN_ROLE","ADMIN_GROUPS"

Append comma separated list of applicable extended attributes at the end.

Extended attribute column name format should be in the format:

XTN_<Extended attribute name in capital letters>.

Example: If extended attribute name is Ip, column name should be XTN_IP.

Below is the header example with two extended attributes called Obj and Zone

"ORG_NAME","FIRST_NAME","MIDDLE_NAME","LAST_NAME","EMAIL","PHONE","LOGIN_NAME","ADMIN_ROLE","ADMIN_GROUPS","XTN_OBJ","XTN_ZONE"

FIELD FORMATS:

ADMIN_ROLE takes SADM, UADM, NADM, PADM, RADM

ADMIN_GROUPS comma separated list of admin groups

EXAMPLE DATA:

"TCPWave","John","","Smith","jsmith@tcpwave.com","9000102010","jsmith","NADM","default","","
importmirroredzone

DESCRIPTION:

Imports the mirrored zone from a CSV file into the TCPWave IPAM

ARGUMENTS:

--input_file

Path on the target IPAM server to the input csv file to import the mirrored zone from. [mandatory]

--output_file

Path on the target IPAM server to the output file to write the results of the import. [mandatory]

--error_file

Path on the target IPAM server to the file to write the records that failed to import. [mandatory]

--max_errors

Maximum permissible errors. The import is aborted if the error count reaches this value. [mandatory]

EXAMPLE:

```
twc importmirroredzone --input_file=/tmp/mirroredzone.txt --output_file=/tmp/output --error_file=/tmp/error --max_errors=10
```

IMPORT FILE

The information to create a mirrored zone should be specified as a comma separated values as specified in the order below.

"NAME","ORG_NAME","ZONE_NAME","DESCRIPTION"

EXAMPLE DATA:

"Test","TcpWave","first.com","Test mirrored zone"

importadminrole

NAME

importadminrole

DESCRIPTION

Imports the administrator roles from a csv file in the TCPWave IPAM.

ARGUMENTS

--input_file

Path on the target IPAM server to the input csv file to import the administrators from. [mandatory]

--output_file

Path on the target IPAM server to the output file to write the results of the import. [mandatory]

--error_file

Path on the target IPAM server to the file to write the records that failed to import. [mandatory]

--max_errors

Maximum permissible errors. The import is aborted if the error count reaches this value. [mandatory]

--failed_entries_file

Path on the target IPAM server to the file to write the only failed record entries.

EXAMPLE

```
twc importadminrole --input_file=/tmp/adminrole.csv --output_file=/tmp/output --error_file=/tmp/error --max_errors=100 --failed_entries_file=/tmp/failed_entries
```

IMPORT FILE

The information to create an administrator role should be in the order below

[administrator-roles]

NAME= Name of the role

INTERNAL= Type of the role (False/True)

DESCRIPTION= Description of the role

[functions]

NAME=Name of the function

DESCRIPTION= Description

GRANULAR_SUPPORT=False/True

EXAMPLE DATA:

[administrator-roles]

NAME=QADM

INTERNAL=false

DESCRIPTION=quality checks

[functions]

NAME=Quick Tasks

DESCRIPTION=

GRANULAR_SUPPORT=false

[functions]

NAME=Bulk Data Export

DESCRIPTION=

GRANULAR_SUPPORT=false

[functions]

NAME=AWS Images

DESCRIPTION=

GRANULAR_SUPPORT=true

[administrator-roles]

NAME=RADM

INTERNAL=true

DESCRIPTION=Read-only administrator

importadmingroup**NAME****importadmingroup****DESCRIPTION**

Imports the admin groups from a CSV file into the TCPWave IPAM.

ARGUMENTS**--input_file**

Path on the target IPAM server to the input csv file to import the admin groups from [mandatory]

--output_file

Path on the target IPAM server to the output file to write the results of the import [mandatory]

--error_file

Path on the target IPAM server to the file to write the records that failed to import [mandatory]

--max_errors

Maximum permissible errors. The import is aborted if the error count reaches this value [mandatory]

--failed_entries_file

Path on the target IPAM server to the file to write the only failed record entries.

EXAMPLE

```
twc importadmingroup --input_file=/tmp/admingroup.csv --output_file=/tmp/output --error_file=/tmp/error --max_errors=10 --failed_entries_file=/tmp/failed_entries
```

EXPORT FILE

The information in the output file list of fields as specified in the order below [admin-group]

NAME=Name of the admin group

DESCRIPTION=Description for admin group

[role]

ROLE=Name of the role

ORGANIZATION=Name of the organization

EXAMPLE DATA

[admin-group]

NAME=CADM-Test

DESCRIPTION=Group with Quick Tasks,Architecture Overview

[role]

```
ROLE=CADM-First
ORGANIZATION=EARTH
[role]
ROLE=RADM
ORGANIZATION=Internal
[admin-group]
NAME=Default EARTH RADM Group
DESCRIPTION=Default EARTH RADM Group
[admin-group]
NAME=CADM-lrMQMF-group
DESCRIPTION=Group with Quick Tasks,Administrator Groups
[role]
ROLE=CADM-One
ORGANIZATION=EARTH
[role]
ROLE=RADM
ORGANIZATION=Internal
[admin-group]
NAME=Default Internal RADM Group
DESCRIPTION=Default Internal RADM Group
importadminpermission
NAME
importadminpermission
DESCRIPTION
Imports the administrator permissions from a CSV file into the TCPWave IPAM.
ARGUMENTS
--input_file
    Path on the target IPAM server to the input csv file to import the administrator
permissions from [mandatory]
--output_file
    Path on the target IPAM server to the output file to write the results of the import
[mandatory]
--error_file
    Path on the target IPAM server to the file to write the records that failed to import
[mandatory]
--max_errors
    Maximum permissible errors. The import is aborted if the error count reaches this
```

value [mandatory]

--failed_entries_file

Path on the target IPAM server to the file to write the only failed record entries.

EXAMPLE:

```
twc importadminpermission --input_file=/tmp/adminperm.csv --output_file=/tmp/output --error_file=/tmp/error --max_errors=10 --failed_entries_file=/tmp/failed_entries
```

IMPORT FILE:

The information to create an administrator permissions should be specified as a comma separated values as specified in the order below

"PERMISSION_LEVEL","ROLE","FUNCTION","VALUE","SELECT_ALL","ADMIN_GROUP","ADMIN","ORG_NAME","PRIVILEGE"

FIELD FORMATS:

PERMISSION_LEVEL takes 'Admin' or 'Admin Group'

ROLE accepts valid name of the role

FUNCTION accepts valid name of the functions

VALUE is the entity value of specified functions

SELECT_ALL to select the all value of the function

ADMIN_GROUP Name of the administrator group if any

ADMIN Name of the administrator

ORG_NAME Name of the organizations

PRIVILEGE takes 'Write' or 'Read' or 'Deny'

EXAMPLE DATA

"Admin","PADM","IPv4 Subnets","","","Yes","","wfpadm","Internal","Write"

"Admin","PADM","IPv4 Networks","10.1.10.0","No","","wfpadm","Internal","Write"

"Admin Group","NADM","IPv4 Objects","","Yes","Default Internal NADM Group","","Internal","Write"

importappliancegroup

NAME:

importappliancegroup

DESCRIPTION:

Imports the appliance groups from a CSV file into the TCPWave IPAM.

ARGUMENTS:

--input_file

Path on the target IPAM server to the input csv file to import the appliance groups from. [mandatory]

--output_file

Path on the target IPAM server to the output file to write the results

of the import. [mandatory]

--error_file
Path on the target IPAM server to the file to write the records that failed to import. [mandatory]

--max_errors
Maximum permissible errors. The import is aborted if the error count reaches this value. [mandatory]

EXAMPLE USAGE:

```
twc importappliancegroup --input_file=/tmp/input.txt --output_file=/tmp/output.txt --error_file=/tmp/error.txt --max_errors=4
```

IMPORT FILE:

The information to create an appliance group should be specified, as a comma separated values as specified, in the order below.

"NAME","ORG_NAME","DESCRIPTION"

EXAMPLE DATA:

```
"app-group1234","Internal","testing qa done"  
"app-group-1256","Internal","testing qa"  
"TCPWave-Remote-Group","TCPWave Organization","TCPWave-Remote-Group"
```

importasset**NAME:**

importasset

DESCRIPTION:

Imports the assets from a CSV file into the TCPWave IPAM.

ARGUMENTS:

--input_file
Path on the target IPAM server to the input csv file to import the assets from. [mandatory]

--output_file
Path on the target IPAM server to the output file to write the results of the import [mandatory]

--error_file
Path on the target IPAM server to the file to write the records that failed to import [mandatory]

--max_errors
Maximum permissible errors. The import is aborted if the error count reaches this value [mandatory]

EXAMPLE USAGE:

```
twc importasset --input_file=/tmp/asset.txt --output_file=/tmp/output --error_file=/tmp/error --max_errors=10
```

IMPORT FILE:

The information to create an asset should be specified, as a comma separated values as specified, in the order below

```
"SERVICE_TAG","SERIAL_NUMBER","GREEN_ZONE","VENDOR","MODEL","NAME","DESCRIPTION","PURCHASE_COST","PURCHASE_DATE","ACQUISITION_TYPE","MAINTENANCE_COST","MAINTENANCE_END_DATE","WARRANTY_END_DATE","CPU","CAPACITY","OS_VERSION","DISPOSAL_DATE","DISPOSAL_REASON","CITY"
```

EXAMPLE DATA:

```
"tag-123","12345","1","vender","model-123","test","","2","2017-02-15 00:00:00","","","","","","2017-02-09 00:00:00","4","500","7.0","2017-02-01 00:00:00","",""
```

importcloudinstances**NAME**

importcloudinstances

DESCRIPTION

Import device instances from cloud to TCPWave IPAM.

ARGUMENTS

--subnet

IP Address of the cloud-hosted subnet with the mask to be imported. [mandatory]

--org

Name of the organization to which subnet belongs. [mandatory]

EXAMPLE

```
twc importcloudinstances --subnet=10.1.10.0/24 --org=TCPWave
```

importcontact

NAME:

importcontact

DESCRIPTION:

Imports the contacts from a CSV file into the TCPWave IPAM.

ARGUMENTS:

--input_file

Path on the target IPAM server to the input csv file to import the contacts from. [mandatory]

--output_file

Path on the target IPAM server to the output file to write the results of the import. [mandatory]

--error_file

Path on the target IPAM server to the file to write the records that failed to import. [mandatory]

--max_errors

Maximum permissible errors. The import is aborted if the error count reaches this value. [mandatory]

EXAMPLE USAGE:

```
twc importcontact --input_file=/tmp/contacts.txt --output_file=/tmp/output --  
error_file=/tmp/error --max_errors=10
```

IMPORT FILE:

The information to create a contact should be specified, as a comma separated values as specified, in the order below.

"FIRST_NAME","MIDDLE_NAME","LAST_NAME","EMAIL_ID","PHONE_NUM","ORG_NAME"

EXAMPLE DATA:

"John","","Smith","john.smith@tcpwave.com","920-310-5555","TCPWave"

importdhcpfailoverpeer

NAME:

importdhcpfailoverpeer

DESCRIPTION:

Imports the DHCP failover peers from a CSV file into the TCPWave IPAM.

ARGUMENTS:

--input_file

Path on the target IPAM server to the input csv file to import the DHCP failover peers from. [mandatory]

--output_file

Path on the target IPAM server to the output file to write the results of the import. [mandatory]

--error_file

Path on the target IPAM server to the file to write the records that failed to import. [mandatory]

--max_errors

Maximum permissible errors. The import is aborted if the error count reaches this value. [mandatory]

EXAMPLE USAGE:

```
twc importdhcpfailoverpeer --input_file=/tmp/input.txt --output_file=/tmp/output.txt --error_file=/tmp/error.txt --max_errors=4
```

IMPORT FILE:

The information to create a DHCP failover peer should be specified, as a comma separated values as specified, in the order below.

```
"NAME","ORGANIZATION_NAME","PRIMARY_APPLIANCE_IP","FAILOVER_APPLIANCE_IP","PRIMARY_APPLIANCE_PORT","FAILOVER_APPLIANCE_PORT","MCLT","SPLIT","LOAD_BALANCE_MAX_SECONDS","MAX_RESPONSE_DELAY","MAX_UNACKED_UPDATES","PRIMARY_APPLIANCE_NAME","FAILOVER_APPLIANCE_NAME","DESCRIPTION"
```

EXAMPLE DATA:

```
"NAME","ORGANIZATION_NAME","PRIMARY_APPLIANCE_IP","FAILOVER_APPLIANCE_IP","PRIMARY_APPLIANCE_PORT","FAILOVER_APPLIANCE_PORT","MCLT","SPLIT","LOAD_BALANCE_MAX_SECONDS","MAX_RESPONSE_DELAY","MAX_UNACKED_UPDATES","PRIMARY_APPLIANCE_NAME","FAILOVER_APPLIANCE_NAME","DESCRIPTION"
```

"dhcp-failover-peer-

```
1","TCPWave","16.0.0.2","15.0.0.2","647","647","1800","120","3","30","30","DHCP-16.0.0.2","DHCP-15.0.0.2","",""
```

"dhcp-failover-peer-

```
5","TCPWave","163.35.7.57","172.175.231.5","647","648","1600","128","2","50","40","dhcp-server01-sl0984","dhcp-server01-sl0984","",""
```

"demo-peer-

```
1","TCPWave","172.181.11.243","172.175.156.133","647","647","1800","128","3","30","30","nhkna81-qrs01-sl0984","dhcp-server02-sl0984","",""
```

"demo-peer-

```
2","TCPWave","9.0.3.4","9.0.1.2","647","647","1800","192","3","30","30","TemDHCPServer","dhcp-server-2","",""
```

importdhcptiontmp

NAME:

importdhcptiontmp

DESCRIPTION:

Imports the DHCP option templates from name-value pair file into the TCPWave IPAM.

ARGUMENTS:**--input_file**

Path on the target IPAM server to the input file to import the DHCP option templates from [mandatory]

--output_file

Path on the target IPAM server to the output file to write the results of the import [mandatory]

--error_file

Path on the target IPAM server to the file to write the records that failed to import [mandatory]

--max_errors

Maximum permissible errors. The import is aborted if the error count reaches this value [mandatory]

EXAMPLE USAGE:

```
twc importdhcoptiontmpl --input_file=/tmp/dhcoptiontmpl.txt --output_file=/tmp/output --error_file=/tmp/error --max_errors=100
```

IMPORT FILE:

The information in the input file is in a format as described below

Each DHCP option template starts with a section [dhcp-option-template] followed by various DHCP parameters in the format <param-name>=<param-value> one per each line.

User defined DHCP options start with a section [dhcp-userdefined-option] followed by the section [dhcp-option-template] for a DHCP option template.

TemplateName is the name of the DHCP option template and is mandatory

SECTIONS & CONFIGURATION PARAMETERS:**[dhcp-option-template]**

TemplateName

Enter name of the DHCP option template.

Organization

Enter the organization name where template must be

create.

Description

Enter description for DHCP option template.

User Authentication Servers

Enter name of the User Authentication Servers.

Default TCP TTL

Enter default time-to-live value in seconds.

Keepalive Time
seconds.

Enter client waiting time to send alive messages in

Keepalive Data
compatibility else false.

Enter true to send live messages with an octet

Service Location Protocol Directory Agent

Enter true/false with SLP agent IP address.

SLP Service Scope

Enter true/false with a list of service scopes for SLP.

Domain Search

Enter a domain name.

Subnet Mask

Takes input as 'Same as in subnet profile' only.

Time Offset

Enter the Time Offset value in seconds.

Router

Takes input as 'Same as in subnet profile' only.

Time Server Time servers.	Enter the comma separated list of valid IPV4 addresses of Time servers.
Name Server Name servers.	Enter the comma separated list of valid IPV4 addresses of Name servers.
Domain Name Server addresses of Domain Name servers.	Enter the comma separated list of valid IPV4 addresses of Domain Name servers.
Log Server Log servers.	Enter the comma separated list of valid IPV4 addresses of Log servers.
Quotes Server Quotes servers.	Enter the comma separated list of valid IPV4 addresses of Quotes servers.
LPR Server LPR servers.	Enter the comma separated list of valid IPV4 addresses of LPR servers.
Impress Server Impress servers.	Enter the comma separated list of valid IPV4 addresses of Impress servers.
RLP Server RLP servers.	Enter the comma separated list of valid IPV4 addresses of RLP servers.
Hostname	Enter name of the client.
Boot File Size	Enter boot file size.
Merit Dump File template.	Enter the path of Merit dump file for the DHCP option template.
Domain Name profile' only.	Takes input as 'Same as the primary domain in subnet
Swap Server	Enter the IPV4 address for Swap server.
Root Path	Enter the path of root disk for the DHCP option template.
Extension File template.	Enter the name of Extension file for the DHCP option template.
NetWare/IP Domain use.	Enter the name of NetWare/IP domain for the client to use.
NetWare/IP Options-nwip.nsq-broadcast Query to locate a NetWare/IP server else false.	Enter true to use the NetWare Nearest Server
NetWare/IP Options-nwip.preferred-dss addresses of NetWare Domain SAP/RIP servers.	Enter the comma separated list of valid IPV4 addresses of NetWare Domain SAP/RIP servers.
NetWare/IP Options-nwip.nearest-nwip-server IPV4 addresses of NetWare servers.	Enter the comma separated list of valid IPV4 addresses of NetWare servers.
NetWare/IP Options-nwip.autoretries	Enter the valid Integer for the number of times that a NetWare/IP client should attempt to communicate with a given DSS server at startup.
NetWare/IP Options-nwip.autoretry-secs	Enter the number of seconds that a NetWare/IP client should wait between retries when attempting to establish communications with a DSS server.
NetWare/IP Options-nwip.nwip-1-1 NetWare/IP version 1.1 compatibility else false.	Enter true for NetWare/IP client to support NetWare/IP version 1.1 compatibility else false.
NetWare/IP Options-nwip.primary-dss Domain SAP/RIP Service server.	Enter the valid IPV4 address of the Primary Domain SAP/RIP Service server.
NDS Servers NDS servers.	Enter the comma separated list of valid IPV4 addresses of NDS servers.
NDS Tree Name	Enter the name of NDS Tree for DHCP option template.
NDS Context for a NDS client.	Enter the name of the initial NetWare Directory Service for a NDS client.
Address Request to be used by the client in a DHCP discover to request that a particular IP addresses.	Enter the comma separated list of valid IPV4 addresses to be used by the client in a DHCP discover to request that a particular IP addresses.
DHCP Message Type	Enter the type of DHCP message.

Parameter List	Enter the comma separated list of valid DHCP options for the client to request DHCP Appliance to return on request.
DHCP Max Msg Size	Enter the maximum size of response that the appliance sends to the client.
Home Agent Addresses	Enter the comma separated list of valid IPV4 addresses for mobile home agents.
User Class	Enter the name of User class to be specified, for DHCP appliance.
Netinfo Address	Enter the comma separated list of Valid NetInfo IPV4 address.
Netinfo Tag	Enter the name of the NetInfo tag for DHCP option template.
Default URL	Enter the value for Default URL.
Vendor Identified Vendor-Specific Information	Enter the Vendor class name to be specified, for DHCP appliance.
Client FQDN	Enter the Valid FQDN for the client to use.
MTU Subnet	Enter true to take the same MTU for all the subnets of the IP network else false.
Trailers	Enter true for the client to use trailers else false.
ARP Timeout	Enter time in seconds for ARP cache entries.
Ethernet	Enter true for the client to use Ethernet Version 2 (RFC 894) else false.
Forward On/Off	Enter true for the client to configure its IP layer for packet forwarding else false.
Source Routing	Enter true for the client to configure its IP layer to allow forwarding of datagrams with non-local source routes else false.
Policy Filter	Enter the comma separated list of valid IPV4 addresses for non-local source routing.
Max Datagram Size	Enter the maximum size of the datagram that client should be prepared to reassemble.
Default IP TTL	Enter the valid time-to-live in seconds for the client to use on outgoing datagrams.
MTU Timeout	Enter the valid MTU Timeout in seconds.
MTU Plateau	Enter the valid Integer for MTU plateau to use when performing Path MTU Discovery.
Mask Discovery	Enter true for the client to perform Mask discovery using ICMP else false.
Mask Supplier	Enter true for the client to respond to subnet mask requests using ICMP else false.
Router Discovery	Enter true for the client to perform Router discovery else false.
Router Request	Enter a valid IPV4 address to which the client should transmit router solicitation requests.
Static Route	Enter the comma separated list of valid IPV4 addresses that client should install in its routing cache.
MTU Interface	Enter the valid Integer for the MTU Interface.
Broadcast Address	Enter the valid IPV4 address.
Address Time	Enter the client request lease time in seconds.
Overload	Enter the valid Integer for DHCP appliance to insert if the returned parameters will exceed the usual space allotted for options.
Vendor Class Id	Enter the value for Vendor class Id.

Client Id	Enter the value for Client Id.
Server Name	Enter the name of the Server to identify a TFTP server.
Bootfile Name	Enter the name of Bootfile to be used by the client.
NETBIOS Dist Server for NETBIOS Dist servers.	Enter the comma separated list of valid IPV4 addresses
NETBIOS Node Type	Enter the valid NetBIOS node type which allows NetBIOS over TCP/IP clients to configure as per RFC 1001/1002.
NETBIOS Scope	Enter the value for NetBIOS scope to specifies the NetBIOS over TCP/IP scope parameter for the client.
X Window Font	Enter the comma separated list of valid IPV4 addresses of X Window System Font servers.
X Window Manager	Enter the comma separated list of valid IPV4 addresses of X Window Manager servers.
NIS+ Domain Name	Enter the name for NIS domain.
NIS+ Server Address of NIS servers.	Enter the comma separated list of valid IPV4 addresses
SMTP Server SMTP servers.	Enter the comma separated list of valid IPV4 addresses of
POP3 Server POP3 servers.	Enter the comma separated list of valid IPV4 addresses of
NNTP Server NNTP servers.	Enter the comma separated list of valid IPV4 addresses of
WWW Server of WWW servers.	Enter the comma separated list of valid IPV4 addresses
Finger Server Finger servers.	Enter the comma separated list of valid IPV4 addresses of
IRC Server IRC servers.	Enter the comma separated list of valid IPV4 addresses of
StreetTalk Server	Enter the comma separated list of valid IPV4 addresses of StreetTalk servers.
StreetTalk Directory Assistance (STDA) Server	Enter the comma separated list of valid IPV4 addresses of STDA servers.
BCMCS Controller IPv4 address option addresses of BCMCS servers.	Enter the comma separated list of valid IPV4
NIS Domain	Enter the value for NIS domain.
NIS Servers NIS servers.	Enter the comma separated list of valid IPV4 addresses of
NTP Servers	Enter the comma separated list of IPV4 address of NTP servers.
Vendor Specific	Enter the value for vendor specific name.
NETBIOS Name Server addresses of NETBIOS Name servers.	Enter the comma separated list of valid IPV4

EXAMPLE DATA:

```
[dhcp-option-template]
TemplateName=OptionTemplate10
Organization=Internal
Description=
Subnet Mask=Same as in subnet profile
Router=Same as in subnet profile
Domain Name=Same as the primary domain in subnet profile
```

```
[dhcp-userdefined-option]
OPTION,GROUP,DATA_TYPE
[dhcp-option-template]
TemplateName=OptionTemplate11
Organization=Internal
Description=
Subnet Mask=Same as in subnet profile
Router=Same as in subnet profile
Domain Name=Same as the primary domain in subnet profile
[dhcp-userdefined-option]
OPTION,GROUP,DATA_TYPE
```

importdhcppolicytmpl**NAME:**

importdhcppolicytmpl

DESCRIPTION:

Imports the DHCP policy templates from name-value pair file into the TCPWave IPAM.

ARGUMENTS:**--input_file**

Path on the target IPAM server to the input file to import the DHCP policy templates from [mandatory]

--output_file

Path on the target IPAM server to the output file to write the results of the import [mandatory]

--error_file

Path on the target IPAM server to the file to write the records that failed to import [mandatory]

--max_errors

Maximum permissible errors. The import is aborted if the error count reaches this value [mandatory]

EXAMPLE USAGE:

```
twc importdhcppolicytmpl --input_file=/tmp/dhcппolicytmpl.txt --output_file=/tmp/output --error_file=/tmp/error --max_errors=100
```

IMPORT FILE:

The information in the input file is in a format as described below

Each DHCP policy template starts with a section [dhcp-policy-template] followed by various DHCP policy parameters in the format
<param-name>=<param-value> one per each line.

TemplateName is the name of the DHCP policy template and is mandatory

SECTIONS & CONFIGURATION PARAMETERS:

[dhcp-policy-template]

TemplateName	Enter name of the DHCP option template.
Organization	Enter the organization name where template must be create.
Description	Enter description for DHCP policy template.
Authoritative	Enter the value of Authoritative network information to eliminate the issue of sending DHCP NAK for legitimate clients. This takes 'yes' or 'no'.
DB Time Format	Enter default/local value for DB time format to format time-stamp in lease information. This takes 'default' or 'local'.
Local Port	Enter the Valid Integer as the port number on which DHCP appliances will receives messages.
Local Address	Enter the Valid IPV4 address on which the DHCP Appliance will get DHCP messages.
Log Facility	Enter the Log Facility name.
Always Broadcast	Enter true for DHCP Appliance to always broadcast its responses to clients within the scope of the parameter defined else false.
Always reply RFC1048	Enter true for DHCP Appliance to format options in RFC 1048 format else false.
Min Secs	Enter the minimum value in seconds for DHCP Appliance to process the message.
Remote Port	Enter a valid Integer to override default port number on which DHCP messages are sent to clients.
Stash Agent Options	Enter true for the DHCP appliance to store the DHCP relay agent information else false.
Adaptive Lease Time Percentage	Enter a valid Integer for the DHCP appliance to automatically decrease lease time for new clients to min-lease-time when the allocated leases as a percentage of pool capacity exceed given percent.
Boot Unknown Clients	Enter true for the DHCP appliance to offer IPV4 addresses for the clients which are not declared with a host declaration format else false.
Default Lease time	Enter a valid Integer in seconds to be provided for clients that do not request for given specified, time.
Get Lease Host Names	Enter true for DHCP server to lookup the hostname corresponding to the assigned IP address and set the resolved hostname in the DHCP hostname option else false.
Infinite is reserved	Enter on for the Client to get infinite lease time else off.
Max Lease Time	Enter a valid Integer for maximum lease time in seconds.
Min Lease Time	Enter a valid Integer for minimum lease time in seconds.
Next Server	Enter the value for Name server from which the client will obtain its boot file.
One Lease Per Client	Enter true for DHCP appliance to assign the requested address and free any other leases associated with the client else false.
Ping Check	Enter true for the DHCP appliance to ping the address before issuing the offer to client else false.
Ping Timeout	Enter a valid ping timeout in seconds.
Use Lease Addr For Default Route	Enter true to set the router option to the same IP address as that offered to the client else false.
Server Identifier	Enter a valid IPV4 address for the Server Identifier.
Server Name	Enter a valid FQDN name for Server name.
Site Option Space	Enter the name of Site option space.
Vendor Option Space	Enter the name of Vendor option space.

EXAMPLE DATA:

```
[dhcp-policy-template]
TemplateName=policy1
Organization=QAOrg
Description=test
Default Lease time=3600
```

importdhcpserver**NAME:**

importdhcpserver

DESCRIPTION:

Imports the DHCP Server from a name-value pair file into the TCPWave IPAM.

ARGUMENTS:

--input_file

Path on the target IPAM server to the input file to import the
DHCP servers from [mandatory]

--output_file

Path on the target IPAM server to the output file to write the results
of the import [mandatory]

--error_file

Path on the target IPAM server to the file to write the records that
failed to import [mandatory]

--max_errors

Maximum permissible errors. The import is aborted if the error count
reaches this value [mandatory]

EXAMPLE USAGE:

```
twc importdhcpserver --input_file=/tmp/dhcpserver.txt --output_file=/tmp/output --  
error_file=/tmp/error --max_errors=100
```

FILE FORMAT:

The output file format is as follows:

Each server starts with a [dhcp-server] section

Each configuration section begins with a [<section name>] field followed
by <name>=<value> pairs one per line

SECTIONS & CONFIGURATION PARAMETERS:

[dhcp-server]

IP_ADDRESS IP Address of the DHCP server

ORGANIZATION_NAME Organization Name of the DHCP server

POLICY_TEMPLATE Policy template name for the DHCP server

APPLIANCE_GROUP Name of the Appliance group to be associated
 ENABLE_MONIT '0' to enable monitoring and '1' to disable monitoring
 TIME_ZONE Time zone
 DESCRIPTION Description of the DHCP Server.

[ntp]
 NTP_SERVERS comma separated list of IP addresses of NTP servers

[snmp]
 TRAP_SINK_1 IP address of SNMP trap sink
 TRAP_SINK_2 IP address of SNMP trap sink
 COMMUNITY_STRING Community string for SNMP
 SYSTEM_LOCATION System Location
 SYSTEM_CONTACT System contact
 PROCESS_LIST comma separated list of processes to be monitored. The following is a valid list of processes:
 ntpd, dns, bgpd, zebra, crond, sshd, monit, syslog-ng,dhcpd
 [snmpv3users]
 ENABLE_SNMPv3 Takes 'true' or 'false' to enable or disable SNMPv3 respectively
 SNMPv3_USERS Takes list of JSON objects to define SNMPv3 users. Example
 [{"userName":"Smith","password":"abcd1234","authentication_protocol":"SHA","encryption_protocol":"AES"}]

[macexclusions]

MAC MAC Address to be exclude
 DESCRIPTION Description of the MAC address exclusion

[tacacs]
 ENABLE_TACACS Takes '0' or '1'. '1' indicates TACACS+ configuration should be enabled for this server. '0' indicates TACACS+ configuration should be disabled
 TACACS_PASSKEY TACACS passkey
 TACACS_SERVERS Comma separated list of TACACS servers.

SAMPLE FILE CONTENTS:

```
[dhcp-server]
IP_ADDRESS=10.1.10.86
ORGANIZATION_NAME=TCPWave
POLICY_TEMPLATE=policy
APPLIANCE_GROUP=ApplianceGroup1
ENABLE_MONIT=1
TIME_ZONE=GMT (GMT)
DESCRIPTION=
[ntp]
NTP_SERVERS=192.168.1.1,192.168.1.2,192.168.1.3,192.168.1.4,
```

```
[snmp]
TRAP_SINK_1=1.1.1.1
TRAP_SINK_2=1.1.1.2
COMMUNITY_STRING=sph1nkx5
SYSTEM_LOCATION=
SYSTEM_CONTACT=
PROCESS_LIST=ntpd,dns,sshd,monit,syslog-ng,dhcpcd,
[snmpv3users]
ENABLE_SNMPv3=false
SNMPv3_USERS=
[macexclusions]
MAC=04:a1:51:8d:f6:96
DESCRIPTION=Detected as abusive DHCP client
[macexclusions]
MAC=04:a1:51:8d:f6:97
DESCRIPTION=Detected as abusive DHCP client
```

importipv6dhcpserver**NAME**

importipv6dhcpserver

DESCRIPTION

Imports the IPv6 DHCP Appliances from a name-value pair file into the TCPWave IPAM.

ARGUMENTS

--input_file

Path on the target IPAM server to the input file to import the DHCP servers from [mandatory]

--output_file

Path on the target IPAM server to the output file to write the results of the import [mandatory]

--error_file

Path on the target IPAM server to the file to write the records that failed to import [mandatory]

--max_errors

Maximum permissible errors. The import is aborted if the error count reaches this value [mandatory]

EXAMPLE

```
twc importipv6dhcpserver --input_file=/tmp/dhcpserver.txt --output_file=/tmp/output --  
error_file=/tmp/error --max_errors=100
```

FILE FORMAT

The output file format is as follows:

Each server starts with a [dhcp-server] section

Each configuration section begins with a [<section name>] field followed by <name>=<value> pairs one per line

SECTIONS & CONFIGURATION PARAMETERS

[dhcp-server]

IP_ADDRESS IPv6 Address of the DHCP server
 ORGANIZATION_NAME Organization Name of the DHCP server
 POLICY_TEMPLATE Policy template name for the DHCP server
 USAGE Name of the Appliance group to be associated
 ENABLE_MONIT '0' to enable monitoring and '1' to disable monitoring
 TIME_ZONE Time zone
 DHCPD_INTERFACES Specify DHCPD Interfaces.
 DESCRIPTION Description of the DHCP Server.
 DISCOVERY '0' to enable discovery and '1' to disable discovery

[ntp]

NTP_SERVERS comma separated list of IP addresses of NTP servers

[dns_resolver]

NAME_APPLIANCES IP address of Name Appliances

SEARCH_SUFFIXES Specify search suffixes

[snmpv3users]

ENABLE_SNMPv3 Takes 'true' or 'false' to enable or disable SNMPv3 respectively
 SNMPv3_USERS Takes list of JSON objects to define SNMPv3 users. Example

`[{"userNamed": "Smith", "password": "abcd1234", "authentication_protocol": "SHA", "encryption_protocol": "AES"}]`

[syslogng-options]

TIME_REOPEN It is the time to wait in seconds before a dead connection is reestablished, and the default value is 60.

TIME_REUSE It is the time to wait before closing idle connections. The default value is 60.

FLUSH_LINES It specifies the number of lines flushed to a destination at a time. The default value is 60.

STATS_FREQ It is the time to wait between statistics messages in seconds. The default value is 60.

LOG_FIFO_SIZE It specifies the number of messages that the output queue can store. The default value is 1000.

LOG_MSG_SIZE It specifies the maximum length of a message in bytes.

KEEP_TIMESTAMP It specifies whether Syslog-NG should accept the timestamp received from the sending application or client. The default value is Yes.

[syslogng-source]

SOURCE_NAME Enter the Source Name of the Syslog-NG sources.

INTERNAL_MSG By default, this field is enabled. All messages generated internally by Syslog-NG use the source driver internal().

SYSTEM_MSG Syslog-NG automatically collects the system-specific log messages of the host on several platforms using the system() driver.

[syslogng-filter]

FILTER_NAME Enter the name of the Syslog_NG filter.

CONDITION Simple or Complex.

COMPLEX_CONDITION On selecting, complex condition, you must provide a valid filter string to add to the configuration file.

[syslogng-filter]

FILTER_NAME Enter the name of the Syslog_NG filter.

CONDITION Simple or Complex.

COMPLEX_CONDITION On selecting, complex condition, you must provide a valid filter string to add to the configuration file.

[syslogng-destination]

DESTINATION_NAME Enter the Name of the Destination file.

TYPE_SNG Type of SNG.

LOG_FILE_NAME Name of the log file.

ENABLE_SYNC Specify 'yes' or 'no' to enable sync.

[syslogng-destination]

DESTINATION_NAME Enter the Name of the Destination file.

TYPE_SNG Type of SNG.

LOG_FILE_NAME Name of the log file.

ENABLE_SYNC Specify 'yes' or 'no' to enable sync.

[syslogng-target]

SOURCE Name of the destination file.

FILTER Select a value from the dropdown.

DESTINATION Enter the log file path.

[syslogng-target]

SOURCE Name of the destination file.

FILTER Select a value from the dropdown.

DESTINATION Enter the log file path.

SAMPLE FILE CONTENTS:

[dhcp-server]

IP_ADDRESS=9000::

ORGANIZATION_NAME=EARTH

POLICY_TEMPLATE=DEF-POLICY

USAGE=Primary

ENABLE_MONIT=1

```
TIME_ZONE=GMT (GMT)
DHCPD_INTERFACES=
DESCRIPTION=Testing
DISCOVERY=0
[ntp]
NTP_SERVERS=5000::2,
[dns_resolver]
NAME_APPLIANCES=5000::2
SEARCH_SUFFIXES=
[snmpv3users]
ENABLE_SNMPv3=false
SNMPv3_USERS=[]
[syslogng-options]
TIME_REOPEN=60
TIME_REAP=60
FLUSH_LINES=60
STATS_FREQ=600
LOG_FIFO_SIZE=1000
LOG_MSG_SIZE=65536
KEEP_TIMESTAMP=Yes
[syslogng-source]
SOURCE_NAME=s_sys
INTERNAL_MSG=YES
SYSTEM_MSG=YES
[syslogng-filter]
FILTER_NAME=f_default
CONDITION=complex
COMPLEX_CONDITION=level(info..emerg) and not (facility(mail) or facility(authpriv) or
facility(cron))
[syslogng-filter]
FILTER_NAME=f_cron
CONDITION=complex
COMPLEX_CONDITION=facility(cron)
[syslogng-destination]
DESTINATION_NAME=d_mesg
TYPE_SNG=File
LOG_FILE_NAME=messages
```

```
ENABLE_SYNC=Yes
[syslogng-destination]
DESTINATION_NAME=d_cron
TYPE_SNG=File
LOG_FILE_NAME=cron
ENABLE_SYNC=Yes
[syslogng-target]
SOURCE=s_sys
FILTER=f_default
DESTINATION=d_mesg
[syslogng-target]
SOURCE=s_sys
FILTER=f_cron
DESTINATION=d_cron
```

importdnsacl

NAME:

importdnsacl

DESCRIPTION:

Imports the DNS ACLs from a CSV file into the TCPWave IPAM.

ARGUMENTS:

--input_file

Path on the target IPAM server to the input csv file to import the
ACLs from [mandatory]

--output_file

Path on the target IPAM server to the output file to write the results
of the import [mandatory]

--error_file

Path on the target IPAM server to the file to write the records that
failed to import [mandatory]

--max_errors

Maximum permissible errors. The import is aborted if the error count
reaches this value [mandatory]

EXAMPLE USAGE:

```
twc importdnsacl --input_file=/tmp/acl.csv --output_file=/tmp/output --error_file=/tmp/error --  
max_errors=100
```

IMPORT FILE:

The information to create a DNS ACL should be specified, as a comma separated values as specified, in the order below

"NAME","DESCRIPTION","ACL"

FIELD FORMATS:

ACL is a comma separated list of ACL elements in one of the following formats:

IPAddress/permission (192.168.0.1/Allow)

ACL-name/permission (internal/Deny)

IPAddress/mask/permission (192.168.0.0/24/Allow)

EXAMPLE DATA:

"external","external servers","172.0.0.1/24/Allow,172.0.0.2/Deny,internal/Deny"

importdnsOPTIONtmp

NAME:

importdnsOPTIONtmp

DESCRIPTION:

Imports the DNS Option templates from a name-value pair file into the TCPWave IPAM.

ARGUMENTS:

--input_file

Path on the target IPAM server to the input file to import the DNS option templates from [mandatory]

--output_file

Path on the target IPAM server to the output file to write the results of the import [mandatory]

--error_file

Path on the target IPAM server to the file to write the records that failed to import [mandatory]

--max_errors

Maximum permissible errors. The import is aborted if the error count reaches this value [mandatory]

EXAMPLE USAGE:

tvc importdnsOPTIONtmp --input_file=/tmp/dnsOPTIONtmp.txt --output_file=/tmp/output --error_file=/tmp/error --max_errors=100

IMPORT FILE:

The information in the input file is in a format as described below

Each DNS option template starts with a section [dns-option-template] followed by various DNS parameters in the format <param-name>=<param-value>

one per each line.

TemplateName is the name of the DNS option template and is mandatory

SECTIONS & CONFIGURATION PARAMETERS:

TemplateName Enter name of the DNS option template.
Organization Enter the organization name where template must be created.
Description Enter description for the DNS option template.
Type Takes 'BIND AUTH' or 'BIND CACHE' or 'UNBOUND' or 'NSD' or 'DNS PROXY'
allow-query Defines an address match list of IP address(es) which are allowed to issue queries to the server.
allow-recursion Defines an address match list of IP address(es) which are allowed to issue recursive queries to the server.
allow-transfer Defines an address match list e.g. IP address(es) that are allowed to transfer (copy) the zone information from the server
blackhole Defines an address match list of hosts that the server will NOT respond to, or answer queries for
lame-ttl Defines the number of seconds to cache lame delegations or lame servers, that is, servers which should be authoritative (obtained via a referral or delegation from a parent) but do not respond as authoritative.
max-ncache-ttl Sets the maximum time (in seconds) for which the server will cache negative (NXDOMAIN) answers (positives are defined by max-cache-ttl)
tcp-clients The tcp-clients allows the user to define the maximum number of TCP connections to be supported.
responses-per-second This parameter defines the number of identical responses per second allowed from any given source IP address and lies in the range 0 to 1000.
window Default is no. If set to yes then the rate limiting function will not be performed will log when the rate-limit function would have been invoked.
transfers-in Only used by slave zones. It determines the number of concurrent inbound zone transfers. Default is 10.
transfers-out Only used by master zones. It determines the number of concurrent outbound zone transfers. Default is 10.
transfers-per-ns Only used by slave zones. It determines the number of concurrent inbound zone transfers for any zone. Default is 2.
directory It is a quoted string defining the absolute path for the server e.g. "/var/named". All subsequent relative paths use this base directory.
statistics-file the pathname of the file the server appends statistics to when instructed to do so using rndc stats. If not specified, the default is named.stats in the server's current directory.
dump-file It is a quoted string defining the absolute path where BIND dumps the database (cache) in response to a rndc dumpdb.
pid-file It is a quoted string which allows to define where the pid (Process Identifier) used by BIND is written.
session-keyfile The pathname of the file into which to write a TSIG session key generated by named for use by nsupdate.
rrset-order It defines the order in which multiple records of the same type are returned.
check-srv-cname If check-integrity is set then fail, warn or ignore SRV records that refer to CNAMES. The default is to warn.
check-mx-cname If check-integrity is set then fail, warn or ignore MX records that refer to CNAMES. The default is to warn.
check-mx Check whether the MX record appears to refer to an IP address. The default is to warn. Other possible values are fail and ignore.

check-names The check-names statement will cause any host name for the zone to be checked for compliance with RFC 952 and RFC 1123 and take the defined action.

recursion If recursion is set to 'yes' the server will always provide recursive query behaviour if requested by the client. If set to 'no' the server will only provide iterative query behaviour - normally resulting in a referral.

empty-zones-enable By default empty-zones-enable is set to yes which means that reverse queries for IPv4 and IPv6 addresses covered by RFCs 1918, 4193, 5737 and 6598 but which is not covered by a locally defined zone clause will automatically return an NXDOMAIN response from the local name server.

listen-on-v6 It turns on BIND to listen for IPv6 queries.

version It specifies the string that will be returned to a version.bind query when using the chaos class only.

dnssec-enable It indicates that a secure DNS service is being used which may be one, or more, of TSIG, SIG(0) or DNSSEC.

dnssec-validation It indicates that a resolver (a caching or caching-only name server) will attempt to validate replies from DNSSEC enabled (signed) zones.

minimal-responses If yes the server will only add NS resource records to the Authority section and A or AAAA resource records to the Additional sections of a query response when they are required by the protocol, for instance, delegations and negative responses.

zone-statistics If zone-statistics is 'yes', the server will collect statistical data on all zones.

EXAMPLE DATA:

```
[dns-option-template]
TemplateName=BIND AUTH Template
Organization=TCPWave
Description=
Type=BIND AUTH
allow-query=any/Allow;
allow-recursion=any/Allow;
allow-transfer=none/Allow;
blackhole=23259
lame-ttl=0
max-ncache-ttl=60
tcp-clients=500
responses-per-second=0
window=15
transfers-in=10
transfers-out=10
transfers-per-ns=2
directory=/
statistics-file=/var/named/log/named.stats
dump-file=/var/named/log/named_dump.db
pid-file=/var/run/named/named.pid
session-keyfile=/var/run/named/session.key
rrset-order=cyclic
check-srv-cname=ignore
check-mx-cname=ignore
check-mx=ignore
check-names=master ignore,response ignore
recursion=no
empty-zones-enable=no
listen-on-v6=none
```

```
version=TCPWave DNS Server
dnssec-enable=yes
dnssec-validation=yes
minimal-responses=yes
zone-statistics=yes
```

importipv6dnsserver

NAME

importipv6dnsserver

DESCRIPTION

Imports the IPv6 DNS appliances from a name-value pair file into the TCPWave IPAM.

ARGUMENTS

--input_file

Path on the target IPAM server to the input file to import the
DNS servers [mandatory]

--output_file

Path on the target IPAM appliances to the output file to write the results
of the import [mandatory]

--error_file

Path on the target IPAM appliances to the file to write the records that
failed to import [mandatory]

--max_errors

Maximum permissible errors. The import is aborted if the error count
reaches this value [mandatory]

--appliance_type

Takes 'auth' or 'cache' as value. If the value is specified as 'auth' then the command
imports all the Authoritative DNS appliances from the input file. If the value is specified as
'cache' then the command imports all the Cache DNS appliances from the input file.
[mandatory]

EXAMPLE:

```
twc importipv6dnsserver --appliance_type=auth --input_file=/tmp/dnsserver.txt --  
output_file=/tmp/output --error_file=/tmp/error --max_errors=100
```

```
twc importipv6dnsserver --appliance_type=cache --input_file=/tmp/dnsserver.txt --
output_file=/tmp/output --error_file=/tmp/error --max_errors=100
```

FILE FORMAT:

The output file format is as follows:

Each server starts with a [dns-server] section

Each configuration section begins with a [<section name>] field followed by <name>=<value> pairs one per line

SECTIONS & CONFIGURATION PARAMETERS:

[dns-server]

TYPE Takes 'BIND AUTH' or 'BIND CACHE' or 'UNBOUND' or 'NSD' or
'DNS PROXY'

OPTION_TEMPLATE DNS Option template name

SERVER_TEMPLATE DNS server template

IPV6_ADDRESS IPv6 address of the server

ORGANIZATION_NAME Organization Name of the server

ENABLE_MONIT '0' to enable monitoring and '1' to disable monitoring

INTERNAL_CACHE Applicable for servers of type 'BIND CACHE'. '0' indicates

that the server is rooted at an internal root server. '1'

indicates that the server is rooted at public internet root appliance

NSM_TEMPLATE NSM Template is applicable if the selected appliance type is

ISC BIND Cache appliance, Unbound Cache Appliance, or recursion enabled ISC
BIND Authoritative appliance.

DESCRIPTION DNS appliance description

TIME_ZONE Time zone

[ntp]

IPV6_NTP_SERVERS comma separated list of IP addresses of NTP servers

[snmp]

IPV6_TRAP_SINK_1 IP address of SNMP trap sink
IPV6_TRAP_SINK_2 IP address of SNMP trap sink
COMMUNITY_STRING Community string for SNMP
SYSTEM_LOCATION System Location
SYSTEM_CONTACT System contact
PROCESS_LIST comma separated list of processes to be monitored. The following is a valid list of processes:
ntpd, dns, bgpd, zebra, crond, sshd, monit, syslog-ng, dhcpcd

[snmpv3users]

ENABLE_SNMPv3 Takes 'true' or 'false' to enable or disable SNMPv3 respectively
SNMPv3_USERS Takes list of JSON objects to define SNMPv3 users. Example

`[{"userName":"Smith","password":"abcd1234","authentication_protocol":"SHA","encryption_protocol":"AES"}]`

[tacacs]

ENABLE_TACACS Takes '0' or '1'. '1' indicates TACACS+ configuration should be enabled for this server. '0' indicates TACACS+ configuration should be disabled
TACACS_PASSKEY TACACS passkey
TACACS_SERVERS Comma separated list of TACACS servers.

[syslogng-options]

TIME_REOPEN The time to wait in seconds before a dead connection is reestablished. Takes a value less than or equal to 32767.
TIME_REAP If no new messages are written to a destination within the specified time in seconds, the connection will be closed, and its state will be freed. Takes a value less

than or equal to 32767.

FLUSH_LINES Specifies how many lines are flushed to a destination at a time. Takes a value less than or equal to 32767.

STATS_FREQ Syslog-NG OSE periodically sends a log statistics message. Takes a value less than or equal to 32767.

LOG_FIFO_SIZE The number of messages that the output queue can store. Takes a value less than or equal to 32767.

LOG_MSG_SIZE The maximal length of the log messages is limited by this option It is not recommended to set the option value higher than 10 MiB. Takes a value less than or equal to 32767.

KEEP_TIMESTAMP Specifies whether syslog-NG should accept the timestamp received from the sending application or client.

Takes value 'Yes' or 'No'.

[syslogng-source]

SOURCE_NAME Name of the Source

INTERNAL_MSG Internal syslog-NG message, takes input values as '0' or '1'. default value is '1'.

SYSTEM_MSG System specific log message, takes input value as '0' or '1'.

MSG_TXT_FILE Message from text file, takes the file name as input.

MSG_MULTI_TXT_FILE Message from multiple text files, takes input '0' or '1'. if this flag is '1' need to specify the **FILE_PATH** and **FILE_PATTERN**.

FILE_PATH File patch to the multiple text file.

FILE_PATTERN File Name pattern.

SYSLOG_SERVER Syslog-NG sever, takes the input as '0' or '1'.

IP_ADDRESS IP address of the syslog server.

PORT Port number of the syslog server.

NETWORK_PROTOCOL Network protocol, supports 'UDP' and 'TCP'.

[syslogng-filter]

FILTER_NAME Name of the Filter.

CONDITION Takes the input as 'complex' or 'simple'.

FACILITIES Allow values are one or more comma separated option given below.

auth, authpriv, cron, deamon, kern, lpr, mail,mark, news, syslog, user, uucp, local0, local1,local2, local3, local4, local5, local6, local7.

PRIORITIES Allow values are one or more comma separated option given below.

info, notice, warning, err, crit, alert, emerg.

HOST_NAME Name of the host.

IP_NETWORK IP address with mask length.

MATCH_EXPRESSION Match expression.

PROGRAM Program.

[syslogng-destination]

DESTINATION_NAME Name of the destination.

TYPE_SNG Type of the destination. takes the value between 1 to 5.

'1'= File

'2'= Named pipe

'3'= Local Users

'4'= All logged-in users

'5'= Syslog server.

LOG_FILE_NAME File name to log the message, mandatory when TYPE_SNG is specified as '1'.

NAMED_PIPE_NAME Named pipe name, mandatory when TYPE_SNG is specified as '2'.

LOCAL_USERS Local users, mandatory when TYPE_SNG is specified as '3'.

SYSLOG_SERVER IP address of the syslog server, mandatory when TYPE_SNG is specified as '5'.

PORT Port number of the syslog server, mandatory when TYPE_SNG is specified as '5'.

NETWORK_PROTOCOL Network protocol, supports 'UDP' and 'TCP', mandatory when TYPE_SNG is specified as '5'.

[syslogng-target]

SOURCE Name of the source.
FILTER Name of the filter.
DESTINATION Name of the destination.

SAMPLE FILE CONTENTS:

[dns-server]

TYPE=BIND AUTH
OPTION_TEMPLATE=BIND AUTH Default Template
SERVER_TEMPLATE=BIND AUTH Default Server Template
IPV6_ADDRESS=5000::2
ORGANIZATION_NAME=TCPWave
ENABLE_MONIT=1
DESCRIPTION=IPV6 DNS Appliance
TIME_ZONE=GMT (GMT)
ENABLE_RECURSION=no
INTERNAL_CACHE=0

[ntp]

IPV6_NTP_SERVERS=5000::2,5000::3,

[dns_resolver]

IPV6_NAME_APPLIANCES=2000::3
SEARCH_SUFFIXES=

[snmp]

IPV6_TRAP_SINK_1=1000::1
IPV6_TRAP_SINK_2=1000::2
COMMUNITY_STRING=sph1nkx5
SYSTEM_LOCATION=
SYSTEM_CONTACT=

```
PROCESS_LIST=ntpd,dns,sshd,monit,syslog-ng,dhcpcd,
```

```
[snmpv3users]
```

```
ENABLE_SNMPv3=false  
SNMPv3_USERS=[]
```

```
[syslogng-options]
```

```
TIME_REOPEN=60  
TIME_REAP=60  
FLUSH_LINES=60  
STATS_FREQ=600  
LOG_FIFO_SIZE=1000  
LOG_MSG_SIZE=2048  
KEEP_TIMESTAMP=Yes
```

```
[syslogng-source]
```

```
SOURCE_NAME=s_sys  
INTERNAL_MSG=YES  
SYSTEM_MSG=YES
```

```
[syslogng-filter]
```

```
FILTER_NAME=f_default  
CONDITION=complex  
COMPLEX_CONDITION=level(info..emerg) and not (facility(mail) or facility(authpriv) or facility(cron))
```

```
[syslogng-filter]
```

```
FILTER_NAME=f_cron  
CONDITION=complex  
COMPLEX_CONDITION=facility(cron)
```

[syslogng-destination]

```
DESTINATION_NAME=d_mesg
TYPE_SNG=File
LOG_FILE_NAME=messages
ENABLE_SYNC=Yes
```

[syslogng-destination]

```
DESTINATION_NAME=d_cron
TYPE_SNG=File
LOG_FILE_NAME=cron
ENABLE_SYNC=Yes
```

[syslogng-target]

```
SOURCE=s_sys
FILTER=f_default
DESTINATION=d_mesg
```

[syslogng-target]

```
SOURCE=s_sys
FILTER=f_cron
DESTINATION=d_cron
importdnsserver
```

NAME:

importdnsserver

DESCRIPTION:

Imports the DNS Server from a name-value pair file into the TCPWave IPAM.

ARGUMENTS:

--input_file

Path on the target IPAM server to the input file to import the
DNS servers from [mandatory]

--output_file

Path on the target IPAM server to the output file to write the results
of the import [mandatory]

--error_file

Path on the target IPAM server to the file to write the records that failed to import [mandatory]

--max_errors

Maximum permissible errors. The import is aborted if the error count reaches this value [mandatory]

--appliance_type

Takes 'auth' or 'cache' as value. If the value is specified, as 'auth' then the command imports all the Authoritative DNS Servers from the input file. If the value is specified, as 'cache' then the command imports all the Cache DNS Servers from the input file. [mandatory]

EXAMPLE USAGE:

```
twc importdnsserver --appliance_type=auth --input_file=/tmp/dnsserver.txt --output_file=/tmp/output --error_file=/tmp/error --max_errors=100
```

```
twc importdnsserver --appliance_type=cache --input_file=/tmp/dnsserver.txt --output_file=/tmp/output --error_file=/tmp/error --max_errors=100
```

FILE FORMAT:

The output file format is as follows:

Each server starts with a [dns-server] section

Each configuration section begins with a [<section name>] field followed by <name>=<value> pairs one per line

SECTIONS & CONFIGURATION PARAMETERS:**[dns-server]**

TYPE Takes 'BIND AUTH' or 'BIND CACHE' or 'UNBOUND' or 'NSD' or 'DNS PROXY'

OPTION_TEMPLATE DNS Option template name

SERVER_TEMPLATE DNS server template

IP_ADDRESS IP address of the server

ORGANIZATION_NAME Organization Name of the server

ENABLE_MONIT '0' to enable monitoring and '1' to disable monitoring

INTERNAL_CACHE Applicable for servers of type 'BIND CACHE'. '0' indicates that the server is rooted at an internal root server. '1' indicates that the server is rooted at public internet

root server

DMZ_VISIBLE When a cache server is root to a public internet root server '1' indicates visibility of internal zones, '0' indicates internal zone are not visible. This flag is not applicable for cache servers rooted at an internal root server

DESCRIPTION DNS server description

TIME_ZONE Time zone

[ntp]

NTP_SERVERS comma separated list of IP addresses of NTP servers

[snmp]

TRAP_SINK_1 IP address of SNMP trap sink

TRAP_SINK_2 IP address of SNMP trap sink

COMMUNITY_STRING Community string for SNMP

SYSTEM_LOCATION System Location

SYSTEM_CONTACT System contact

PROCESS_LIST comma separated list of processes to be monitored. The following is a valid list of processes:

ntpd, dns, bgpd, zebra, crond, sshd, monit, syslog-ng, dhcpcd

[snmpv3users]

ENABLE_SNMPV3 Takes 'true' or 'false' to enable or disable SNMPv3 respectively

SNMPV3_USERS Takes list of JSON objects to define SNMPv3 users. Example

`[{"userName":"Smith","password":"abcd1234","authentication_protocol":"SHA","encryption_protocol":"AES"}]`

[tacacs]

ENABLE_TACACS Takes '0' or '1'. '1' indicates TACACS+ configuration should be enabled for this server. '0' indicates

TACACS+ configuration should be disabled

TACACS_PASSKEY TACACS passkey

TACACS_SERVERS Comma separated list of TACACS servers.

SAMPLE FILE CONTENTS:**[dns-server]**

TYPE=BIND AUTH

OPTION_TEMPLATE=BIND AUTH Default Template

SERVER_TEMPLATE=BIND AUTH Default Server Template

IP_ADDRESS=10.1.10.86

ORGANIZATION_NAME=QAOrg

APPLIANCE_GROUP=

ENABLE_MONIT=1

DESCRIPTION=

TIME_ZONE=GMT (GMT)

ENABLE_RECURSION=no

INTERNAL_CACHE=0

[ntp]

NTP_SERVERS=192.168.1.1,192.168.1.2,192.168.1.3,192.168.1.4,

[snmp]

TRAP_SINK_1=1.1.1.1

TRAP_SINK_2=1.1.1.2

COMMUNITY_STRING=sph1nkx5

SYSTEM_LOCATION=

```
SYSTEM_CONTACT=
PROCESS_LIST=ntpd,dns,sshd,monit,syslog-ng,dhcpd,
[snmpv3users]
ENABLE_SNMPv3=false
SNMPv3_USERS=
```

importdnsservertmp1

NAME:

importdnsservertmp1

DESCRIPTION:

Imports the DNS server templates from a CSV file into the TCPWave IPAM.

ARGUMENTS:

--input_file

Path on the target IPAM server to the input csv file to import the
DNS server template from [mandatory]

--output_file

Path on the target IPAM server to the output file to write the results
of the import [mandatory]

--error_file

Path on the target IPAM server to the file to write the records that
failed to import [mandatory]

--max_errors

Maximum permissible errors. The import is aborted if the error count
reaches this value [mandatory]

EXAMPLE USAGE:

```
twc importdnsservertmp1 --input_file=/tmp/dnsservertmp1.csv --output_file=/tmp/output --  
error_file=/tmp/error --max_errors=100
```

IMPORT FILE:

The information to create a DNS server template should be specified, as a
comma separated values as specified, in the order below

"TYPE","NAME","DESCRIPTION","EMAIL","DYN_UPD","ALGORITHM","LOGGER"

FIELD FORMATS:

TYPE takes one of the following values: 'BIND AUTH', 'BIND CACHE',
'UNBOUND', 'NSD'

DYN_UPD takes '0' or '1'. '1' indicates dynamic updates must be enabled.
'0' indicates dynamic updates are disable

ALGORITHM is a comma separated list of algorithm specification as follows:

Algorithm:Bit_size

Algorithm should be one of the valid TSIG algorithms. Bit_size should be between minimum bit size and maximum bit size specified, for that algorithm

Example: "HMAC-SHA1:150,HMAC-SHA256:200"

LOGGER is a comma separated list of logger specification as follows:

LogCategory/LogChannels

LogChannels is a colon separated list of log channels

Example: "client/default_stderr:default_debug:default_syslog"

EXAMPLE DATA:

"BIND AUTH","BIND AUTH Default Server Template","BIND Authoritative Default Server Template","","1","HMAC-SHA512:512","default/named,queries/queries"
"BIND CACHE","BIND CACHE Default Server Template","BIND Cache Default Server Template","","0","","queries/queries,default/named"
"DNS PROXY","DNS PROXY Default Server Template","DNS PROXY Default Server Template","","1","HMAC-SHA512:512","queries/queries,default/named"
"UNBOUND","UNBOUND Default Server Template","UNBOUND Default Server Template","","0","",""
"NSD","NSD Default Server Template","NSD Default Server Template","","0","",""

importdomain

NAME:

importdomain

DESCRIPTION:

Imports the domains from a CSV file into the TCPWave IPAM.

ARGUMENTS:

--input_file

Path on the target IPAM server to the input csv file to import the domains from [mandatory]

--output_file

Path on the target IPAM server to the output file to write the results of the import [mandatory]

--error_file

Path on the target IPAM server to the file to write the records that failed to import [mandatory]

--max_errors

Maximum permissible errors. The import is aborted if the error count reaches this value [mandatory]

EXAMPLE USAGE:

```
twc importdomain --input_file=/tmp/input.txt --output_file=/tmp/output.txt --  
error_file=/tmp/error.txt --max_errors=10
```

IMPORT FILE:

The information to create a domain should be specified, as a comma separated values as specified, in the order below

DOMAIN,ORGANIZATION,DESCRIPTION

EXAMPLE DATA:

"tcpwave.com","TCPWave","TCPwave Domain"

importextension**NAME:**

importextension

DESCRIPTION:

Imports the extended attributes from a CSV file into the TCPWave IPAM.

ARGUMENTS:**--input_file**

Path on the target IPAM server to the input csv file to import the extended attributes from. [mandatory]

--output_file

Path on the target IPAM server to the output file to write the results of the import. [mandatory]

--error_file

Path on the target IPAM server to the file to write the records that failed to import. [mandatory]

--max_errors

Maximum permissible errors. The import is aborted if the error count reaches this value. [mandatory]

EXAMPLE USAGE:

```
twc importextension --input_file=/tmp/input.txt --output_file=/tmp/output.txt --  
error_file=/tmp/error.txt --max_errors=4
```

IMPORT FILE:

The information to create an extended attribute should be specified, as a comma separated values as specified, in the order below.

"NAME","DESCRIPTION","EXTENSION_TYPE","CONSTRAINTS","LOW","HIGH","IS_UNIQUE","FLAG"

EXAMPLE DATA:

"NAME","DESCRIPTION","EXTENSION_TYPE","CONSTRAINTS","LOW","HIGH","IS_UNIQUE","FLAG"
 "nextension","","STRING","","","","","","
 "MPLList","","LIST","LIST_VALUES=domain,network,subnet","","","",""
 "subbaiah","","STRING","","","","","","
 "First_Seen_In_Cloud","DO NOT DELETE THIS ATTRIBUTE OR EDIT THE VALUE OF IT.","STRING","","","","","1"
 "Last_Seen_In_Cloud","DO NOT DELETE THIS ATTRIBUTE OR EDIT THE VALUE OF IT.","STRING","","","","","1"
 "testxtn","","STRING","","","","","","
 "Subnet_ext","test","STRING","mandatory","","","",""
importipamappliance

NAME:

`importipamappliance`

DESCRIPTION:

Imports the IPAM appliances from a CSV file into the TCPWave IPAM.

ARGUMENTS:

`--input_file`

Path on the target IPAM server to the input csv file to import the IPAM appliance from. [mandatory]

`--output_file`

Path on the target IPAM server to the output file to write the results of the import. [mandatory]

`--error_file`

Path on the target IPAM server to the file to write the records that failed to import. [mandatory]

`--max_errors`

Maximum permissible errors. The import is aborted if the error count reaches this value. [mandatory]

EXAMPLE USAGE:

`tvc importipamappliance --input_file=/tmp/input.txt --output_file=/tmp/output.txt --error_file=/tmp/error.txt --max_errors=4`

SECTIONS & CONFIGURATION PARAMETERS:

[ipam-appliance]

NAME Name of the IPAM Appliance.

IP_ADDRESS IP Address of the IPAM Appliance.
 TYPE Type of the IPAM Appliance.
 BANNER_COLOR Color of the banner in the Appliance.
 BANNER_TITLE Title of the banner in the Appliance.
 DESCSCRIPTION Description for the IPAM Appliance.

[ntp]
 NTP_SERVERS comma separated list of IP addresses of NTP servers

[snmp]
 TRAP_SINK_1 IP address of SNMP trap sink
 TRAP_SINK_2 IP address of SNMP trap sink
 COMMUNITY_STRING Community string for SNMP
 SYSTEM_LOCATION System Location
 SYSTEM_CONTACT System contact
 PROCESS_LIST comma separated list of processes to be monitored. The following is a valid list of processes:
 ntpd, dns, bgpd, zebra, crond, sshd, monit, syslog-ng, dhcpcd

[snmpv3users]
 ENABLE_SNMPv3 Takes 'true' or 'false' to enable or disable SNMPv3 respectively
 SNMPv3_USERS Takes list of JSON objects to define SNMPv3 users. Example
 [{"userName":"Smith","password":"abcd1234","authentication_protocol":"SHA","encryption_protocol":"AES"}]

[tacacs]
 ENABLE_TACACS Takes '0' or '1'. '1' indicates TACACS+ configuration should be enabled for this server. '0' indicates TACACS+ configuration should be disabled
 TACACS_PASSKEY TACACS passkey
 TACACS_SERVERS Comma separated list of TACACS servers.

SAMPLE INPUT FILE CONTENTS:

```
[ipam-appliance]
NAME=COMMON-DEVELOPER-IPAM
IP_ADDRESS=10.1.10.240
TYPE=Production
BANNER_COLOR=red
BANNER_TITLE=This is a Production Server. All actions are audited. Please do not make any unauthorized changes.
DESCSCRIPTION=
[ntp]
NTP_SERVERS=172.253.172.253,172.253.172.254,
[snmp]
TRAP_SINK_1=10.1.10.1
TRAP_SINK_2=10.1.10.2
COMMUNITY_STRING=E61B8541B79BC35E8E5BEFBE908EB050
```

```
SYSTEM_LOCATION=
SYSTEM_CONTACT=
PROCESS_LIST=ntpd,sshd,de,cli,search,tims,timsscheduler,mysql,
[snmpv3users]
ENABLE_SNMPv3=false
SNMPv3_USERS=
```

importipv6object

NAME

importipv6object

DESCRIPTION

The twc importipv6object CLI command is used to import IPv6 objects from a csv file into IPAM. The syntax of this command is shown below. The user invoking this command is expected to have authentication permission and should be authorized to perform an import IPv6 object. The TCPWave IPAM audits this action. Successful completion of this command exits with a status code 0. The TCPWave IPAM audits this action. Successful completion of this command exits with a status code 0.

ARGUMENTS

--input_file

Path on the target IPAM server to the input csv file to import the IPv6 objects from.
[mandatory]

--output_file

Path on the target IPAM server to the output file to write the results of the import.
[mandatory]

--error_file

Path on the target IPAM server to the file to write the records that failed to import.
[mandatory]

--max_errors

Maximum permissible errors. The import is aborted if the error count reaches this value.
[mandatory]

--ignore_duplicates

It takes '0' or '1'. If this argument is specified as '1' duplicate object's name check in the input file is ignored. If it is not specified or specified as '0' duplicate object's name check in the input file is done.

--failed_entries_file

Path on the target IPAM server to the file to write the only failed record entries.

EXAMPLE

```
twc importipv6object --input_file=/tmp/input.txt --output_file=/tmp/output.txt --
error_file=/tmp/error.txt --max_errors=10 --ignore_duplicates=1 --
failed_entries_file=/tmp/failed_entries
```

IMPORT FILE:

The information to create an IPv6 object should be specified as a comma separated values as specified in the order below

"ADDRESS","ORGANIZATION","NAME","DOMAIN","OBJECT_TYPE","ALLOCATION_TYPE","MAC","OPTION_TEMPLATE","TTL","NS_A","NS_PTR","DDNS_A","DDNS_PTR","DDNS_CNAME","DDNS_MX","CONTACT_FIRST_NAME","CONTACT_MIDDLE_NAME","CONTACT_LAST_NAME","CONTACT_EMAIL","DESCRIPTION","MANAGED_BY","MONITORED_BY","CHANGE_CONTROL_TICKET","TERMINAL_SERVER_KVM","END_OF_LIFE","ROOM","FLOOR"

OBJECT_TYPE represents various predefined device types in the network. For example 3G Phone, Access Router etc.

ALLOC_TYPE takes '1','2','3','4' representing Static, Manual, Auto and Dynamic allocation respectively

NS_A, NS_PTR, DDNS_A, DDNS_PTR, DDNS_CNAME, DDNS_MX take '1' or '0'. These flags enable or disable updates/dynamic updates for the corresponding IPv6 object in the name servers.

EXAMPLE DATA:

```
"6002::12","Internal","Access00001Router","tcp.com","Access  
Router","1","","","","1200","1","1","1","1","","","","","","6002::12","","","","","","","",""
```

importipv6objectrr

NAME:

importipv6objectrr

DESCRIPTION:

The twc importipv6objectrr CLI command is used to import IPv6 objects resource record from a csv file into IPAM. The syntax of this command is shown below. The user invoking this command is expected to have authentication permission and should be authorized to perform an import IPv6 Object RR. The TCPWave IPAM audits this action.

Successful completion of this command exits with a status code 0.

The TCPWave IPAM audits this action. Successful completion of this command exits with a status code 0.

ARGUMENTS:

--input_file

Path on the target IPAM server to the input csv file to import the IPv6 objects resource record from [mandatory]

--output_file

Path on the target IPAM server to the output file to write the results of the import [mandatory]

--error_file

Path on the target IPAM server to the file to write the records that failed to import [mandatory]

--max_errors

Maximum permissible errors. The import is aborted if the error count reaches this value

EXAMPLE USAGE:

```
twc importipv6objectrr --input_file=/tmp/input.txt --output_file=/tmp/output.txt --  
error_file=/tmp/error.txt --max_errors=10
```

IMPORT FILE:

The information to create an IPv6 object resource record should be specified, as a comma separated values as specified, in the order below

"IP_ADDRESS","ORGANIZATION_NAME","OWNER","TTL","CLASS","TYPE","DATA"

Type takes one of 'AAAA','CNAME','MX','SRV','TXT' or 'NAPTR'

EXAMPLE DATA:

"5000::2","tcpwave","www.tcpwave.com.","1200","IN","AAAA","5000::2"

importipv6subnet

NAME

importipv6subnet

DESCRIPTION

Imports the IPv6 subnets from a CSV file into the TCPWave IPAM.

ARGUMENTS

--input_file

Path on the target IPAM server to the input csv file to import the IPv6 subnets from [mandatory]

--output_file

Path on the target IPAM server to the output file to write the results of the import [mandatory]

--error_file

Path on the target IPAM server to the file to write the records that failed to import [mandatory]

--max_errors

Maximum permissible errors. The import is aborted if the error count reaches this value [mandatory]

--failed_entries_file

Path on the target IPAM server to the file to write the only failed record entries.

EXAMPLE

```
twc importipv6subnet --input_file=/tmp/ipv6subnet.txt --output_file=/tmp/output --  
error_file=/tmp/error --max_errors=10 --failed_entries_file=/tmp/failed_entries
```

IMPORT FILE

The information to create a IPv6 subnet should be specified as a comma separated values as specified in the order below

"ADDRESS","MASK","BLOCK_ADDRESS","NAME","ORGANIZATION","DOMAIN","SUBNET_GROUP","ROUTER","DHCP_TEMPLATE_NAME","PRIMARY_DHCP_SERVER","STREET_1","STREET_2","CITY","STATE","COUNTRY","ZIP","DESC","CONTACT_F_NAME","CONTACT_M_NAME","CONTACT_L_NAME","CONTACT_EMAIL","VLAN","VRF","DISCOVERY_TEMPLATE"

EXAMPLE DATA

importipv6subnetgroup

NAME:

importipv6subr

DESCRIPTION:

Imports the IPW

MENTS:

_file

extended attributes from. [mandatory]

--output_file

Path on the target IPAM server to the output file to write the results of the import. [mandatory]

--error_file

Path on the target IPAM server to the file to write the records that failed to import. [mandatory]

--max_errors

Maximum permissible errors. The import is aborted if the error count reaches this value. [mandatory]

EXAMPLE USAGE:

```
twc importipv6subnetgroup --input_file=/tmp/input.txt --output_file=/tmp/output.txt --error_file=/tmp/error.txt --max_errors=4
```

IMPORT FILE:

The information to create an IPv6 subnet group should be specified, as a comma separated values as specified, in the order below.

"NAME","ORG_NAME","DESCRIPTION"

EXAMPLE DATA:

```
"NAME","ORG_NAME","DESCRIPTION"  
"v6Sub_gr1","Internal","","  
"v6sg","EARTH","test"
```

importipv6block**NAME**

importipv6block

DESCRIPTION

Imports the IPv6 block from a CSV file into the TCPWave IPAM.

ARGUMENTS

--input_file

Path on the target IPAM server to the input csv file to import the IPv6 block from.
[mandatory]

--output_file

Path on the target IPAM server to the output file to write the results of the import.
[mandatory]

--error_file

Path on the target IPAM server to the file to write the records that failed to import.
[mandatory]

--max_errors

Maximum permissible errors. The import is aborted if the error count reaches this value.
[mandatory]

--failed_entries_file

Path on the target IPAM server to the file to write the only failed record entries.

EXAMPLE

```
twc      importipv6block      --input_file=/tmp/block.txt      --output_file=/tmp/output      --
error_file=/tmp/error --max_errors=50 --failed_entries_file=/tmp/failed_entries
```

IMPORT FILE

The information to create a IPv6 block should be specified as a comma separated values as specified in the order below

"ADDRESS","MASK","POOL_ADDRESS","NAME","ORG_NAME","DNSSEC","NSEC_OPT","ZONE_TEMPLATE","DMZ_VISIBLE","DESCRIPTION","CONTACT_F_NAME","CONTACT_M_NAME","CONTACT_L_NAME","CONTACT_EMAIL","CLOUD_REGION","DISCOVERY_TEMPLATE","VRF"

FIELD FORMATS

DNSSEC accept '0' or '1'

NSEC_OPT accepts 'NSEC' or 'NSEC3'

DMZ_VISIBLE accepts '1' or '0'.

'1' indicates that the reverse zone is visible to the cache server rooted at a public internet root server. '0' indicates that the zone is not visible.

If this argument is not specified the value is defaulted to '0'

EXAMPLE DATA

"8001::","48","TCPWave","","USA","pool1","0","","","","0","TCPWave pool","","","","","","","","","",""	IPv6	address
---	------	---------

importipv6pool

NAME

importipv6pool

DESCRIPTION

Imports the IPv6 pool from a CSV file into the TCPWave IPAM.

ARGUMENTS

--input_file

Path on the target IPAM server to the input csv file to import the IPv6 pool from. [mandatory]

--output_file

Path on the target IPAM server to the output file to write the results of the import. [mandatory]

--error_file

Path on the target IPAM server to the file to write the records that failed to import. [mandatory]

--max_errors

Maximum permissible errors. The import is aborted if the error count reaches this value. [mandatory]

--failed_entries_file

Path on the target IPAM server to the file to write the only failed record entries.

EXAMPLE

```
twc      importipv6pool      --input_file=/tmp/pools.txt      --output_file=/tmp/output      --
error_file=/tmp/error --max_errors=50 --failed_entries_file=/tmp/failed_entries
```

IMPORT FILE

The information to create a IPv6 pool should be specified as a comma separated values as specified in the order below

"IP_ADDR","MASK","ORG_NAME","REGION","NAME","DNSSEC","NSEC_OPT","ZONE_TEMPLATE","D

MZ_VISIBLE","DESCRIPTION","CONTACT_F_NAME","CONTACT_M_NAME","CONTACT_L_NAME","CONTACT_EMAIL","CLOUD_REGION","DISCOVERY_TEMPLATE","VRF"

FIELD FORMATS

DNSSEC accept '0' or '1'

NSEC_OPT accepts 'NSEC' or 'NSEC3'

DMZ_VISIBLE accepts '1' or '0'.

'1' indicates that the reverse zone is visible to the cache server rooted at a public internet root server. '0' indicates that the zone is not visible. If this argument is not specified the value is defaulted to '0'

EXAMPLE DATA

"8001::","48","TCPWave","","USA","pool1","0","","","","0","TCPWave" IPv6 address
pool","","","","","","","","","",""

importlocation

NAME:

importlocation

DESCRIPTION:

Imports the locations from a CSV file into the TCPWave IPAM.

ARGUMENTS:

--input_file

Path on the target IPAM server to the input csv file to import the locations from [mandatory]

--output_file

Path on the target IPAM server to the output file to write the results of the import [mandatory]

--error_file

Path on the target IPAM server to the file to write the records that failed to import [mandatory]

--max_errors

Maximum permissible errors. The import is aborted if the error count

reaches this value [mandatory]

EXAMPLE USAGE:

```
twc importlocation --input_file=/tmp/locations.txt --output_file=/tmp/output --error_file=/tmp/error --max_errors=50
```

IMPORT FILE:

The information to create a location should be specified, as a comma separated values as specified, in the order below

```
"STREET1","STREET2","CITY","STATE","ZIP","COUNTRY","ORG_NAME"
```

EXAMPLE DATA:

```
"600 ALEXANDER ROAD","","PRINCETON","NJ","08540","USA","TCPWave"
```

importlogchannel**NAME:**

importlogchannel

DESCRIPTION:

Imports the DNS Log Channels from a CSV file into the TCPWave IPAM.

ARGUMENTS:**--input_file**

Path on the target IPAM server to the input csv file to import the log channels from [mandatory]

--output_file

Path on the target IPAM server to the output file to write the results of the import [mandatory]

--error_file

Path on the target IPAM server to the file to write the records that failed to import [mandatory]

--max_errors

Maximum permissible errors. The import is aborted if the error count reaches this value [mandatory]

EXAMPLE USAGE:

```
twc importlogchannel --input_file=/tmp/logchannel.csv --output_file=/tmp/output --error_file=/tmp/error --max_errors=10
```

IMPORT FILE:

The information to create a DNS Log Channel should be specified, as a comma separated values as specified, in the order below

```
""NAME","TYPE","FILE_PATH","VERSION","SIZE","FACILITY","SEVERITY","DEBUG_LEVEL","PRIN
```

T_TIME","PRINT_SEVERITY","PRINT_CATEGORY"

FIELD FORMATS:

NAME is the name of the DNS Log Channel

TYPE takes 'FILE', 'SYSLOG', 'STDERR' or 'NULL'

FILE_PATH a valid file path when TYPE is 'FILE'

VERSION is a valid integer indicating the maximum number of log file versions to be retained on disk during log rotation before purging the oldest log file. The default is '6'

SIZE is a valid integer in bytes indicating the maximum size of a log file before a new log file is created during log rotation. The default is '1024000'

FACILITY is the facility name when TYPE is 'SYSLOG'

SEVERITY takes 'dynamic', 'debug', 'info', 'notice', 'warning', 'error' or 'critical'. The default is 'dynamic'

DEBUG_LEVEL is a valid integer indicating the debug level when SEVERITY is specified, as 'debug'. The default is '0'

PRINT_TIME, PRINT_SEVERITY, PRINT_CATEGORY takes '0' or '1'

EXAMPLE DATA:

"queries","FILE","/var/named/log/query.log","6","1024000","","dynamic","","1","1","1"
"named","FILE","/var/named/log/named.log","6","1024000","","dynamic","","1","1","1"

importnetwork

NAME:

importnetwork

DESCRIPTION:

Imports the networks from a CSV file into the TCPWave IPAM.

ARGUMENTS:

--input_file

Path on the target IPAM server to the input csv file to import the network from [mandatory]

--output_file

Path on the target IPAM server to the output file to write the results of the import [mandatory]

--error_file

Path on the target IPAM server to the file to write the records that

failed to import [mandatory]

--max_errors
Maximum permissible errors. The import is aborted if the error count reaches this value [mandatory]

EXAMPLE USAGE:

twc importnetwork --input_file=/tmp/network.txt --output_file=/tmp/output --error_file=/tmp/error --max_errors=50

IMPORT FILE:

The information to create a network should be specified, as a comma separated values as specified, in the order below

"IP_ADDR","MASK","ORG_NAME","NAME","PERCENTAGE_FULL","SNMP_CHECK","LOG_CHECK","DNSSEC","NSEC_OPT","ZONE_TEMPLATE","DMZ_VISIBLE","DESCRIPTION"

Append comma separated list of applicable extended attributes at the end.

Extended attribute column name format should be in the format:

XTN_<Extended attribute name in capital letters>.

Example: If extended attribute name is Ip, column name should be XTN_IP.

Below is the header example with two extended attributes called Obj and Zone

"IP_ADDR","MASK","ORG_NAME","NAME","PERCENTAGE_FULL","SNMP_CHECK","LOG_CHECK","DNSSEC","NSEC_OPT","ZONE_TEMPLATE","DMZ_VISIBLE","DESCRIPTION","XTN_OBJ","XTN_ZONE"

FIELD FORMATS:

SNMP_CHECK, LOG_CHECK, DNSSEC accept '0' or '1'

NSEC_OPT accepts 'NSEC' or 'NSEC3'

DMZ_VISIBLE accepts '1' or '0'.

'1' indicates that the reverse zone is visible to the cache server rooted at a public internet root server. '0' indicates that the zone is not visible.

If this argument is not specified, the value is defaulted to '0'

EXAMPLE DATA:

"10.1.10.0","24","TCPWave","TCPWave network","100","0","0","1","NSEC3","","0","TCPWave Network Add"

"192.168.56.0","24","TCPWave","TCPWave1 network","100","0","0","1","NSEC3","","0","TCPWave Network Add"

importobject**NAME:**

importobject

DESCRIPTION:

The twc importobject CLI command is used to import objects from a csv file

into IPAM. The syntax of this command is shown below. The user invoking this command is expected to have authentication permission and should be authorized to perform an import object. The TCPWave IPAM audits this action. Successful completion of this command exits with a status code 0.

The TCPWave IPAM audits this action. Successful completion of this command exits with a status code 0.

ARGUMENTS:

--input_file

Path on the target IPAM server to the input csv file to import the objects from [mandatory]

--output_file

Path on the target IPAM server to the output file to write the results of the import [mandatory]

--error_file

Path on the target IPAM server to the file to write the records that failed to import [mandatory]

--max_errors

Maximum permissible errors. The import is aborted if the error count reaches this value [mandatory]

EXAMPLE USAGE:

```
twc importobject --input_file=/tmp/input.txt --output_file=/tmp/output.txt --
error_file=/tmp/error.txt --max_errors=10 --ignore_duplicates=1
```

IMPORT FILE:

The information to create an object should be specified, as a comma separated values as specified, in the order below

"IP_ADDRESS","ORGANIZATION_NAME","NAME","DOMAIN_NAME","CLASS_CODE","ALLOC_TYPE","MAC_ADDR","OPTION_TEMPLATE_NAME","TTL","NS_A","NS_PTR","DDNS_A","DDNS_PTR","DDNS_CNAME","DDNS_MX","CONTACT_FIRST_NAME","CONTACT_MIDDLE_NAME","CONTACT_LAST_NAME","CONTACT_EMAIL","DESCRIPTION","MANAGED_BY","MONITORED_BY","CHANGE_CONTROL_TICKET","TERMINAL_SERVER_KVM","END_OF_LIFE","ROOM","FLOOR","RES_EXP_DATE"

Append comma separated list of applicable extended attributes at the end.

Extended attribute column name format should be in the format:

XTN_<Extended attribute name in capital letters>.

Example: If extended attribute name is Ip, column name should be XTN_IP.

Below is the header example with two extended attributes called Ip and domain

"IP_ADDRESS","ORGANIZATION_NAME","NAME","DOMAIN_NAME","CLASS_CODE","ALLOC_TYPE","MAC_ADDR","OPTION_TEMPLATE_NAME","TTL","NS_A","NS_PTR","DDNS_A","DDNS_PTR","DDNS_CNAME","DDNS_MX","CONTACT_FIRST_NAME","CONTACT_MIDDLE_NAME","CONTACT_LAST_NAME","CONTACT_EMAIL","DESCRIPTION","MANAGED_BY","MONITORED_BY","CHANGE

_CONTROL_TICKET","TERMINAL_SERVER_KVM","END_OF_LIFE","ROOM","FLOOR","RES_EXP_DATE","XTN_IP","XTN_DOMAIN"

CLASS_CODE represents various predefined device types in the network.
For example 3G Phone, Access Router etc.

ALLOC_TYPE takes '1','2','3','4' representing Static, Manual, Auto and
Dynamic allocation respectively

NS_A, NS_PTR, DDNS_A, DDNS_PTR, DDNS_CNAME, DDNS_MX take '1' or '0'. These
flags enable or disable updates/dynamic updates for the corresponding
object in the name servers.

EXAMPLE DATA:

Adding a DHCP-Dynamic object (ALLOC_TYPE=4)

10.1.10.18,TCPWave,"3G-Phone-2733663",tcpwave.com,"3G
Phone",4,01:23:45:67:89:ab,"Generic-
template",30,0,0,1,0,1,0,John,Francis,Smith,john.smith@tcpwave.com,"TCPwave Internal 3G
phone","","","","","","","","","","","","","","

Adding a Static object (ALLOC_TYPE=1)

10.1.10.1,TCPWave,"Router-27",tcpwave.com,"Access
Router",1,01:23:45:67:89:ab,"",30,0,0,1,0,1,0,John,Francis,Smith,john.smith@tcpwave.com,"TCP
wave Internal Router","","","","","","","","","","","","

importobjectrr

NAME:

importobjectrr

DESCRIPTION:

The twc importobjectrr CLI command is used to import objects resource record
from a csv file into IPAM. The syntax of this command is shown below. The
user invoking this command is expected to have authentication permission and
should be authorized to perform an import Object RR. The TCPWave IPAM audits
this action.

Successful completion of this command exits with a status code 0.

The TCPWave IPAM audits this action. Successful completion of this command
exits with a status code 0.

ARGUMENTS:

--input_file

Path on the target IPAM server to the input csv file to import the
objects resource record from [mandatory]

--output_file

Path on the target IPAM server to the output file to write the results
of the import [mandatory]

--error_file

Path on the target IPAM server to the file to write the records that failed to import [mandatory]

--max_errors

Maximum permissible errors. The import is aborted if the error count reaches this value

EXAMPLE USAGE:

```
twc importobjectrr --input_file=/tmp/input.txt --output_file=/tmp/output.txt --error_file=/tmp/error.txt --max_errors=10
```

IMPORT FILE:

The information to create an object resource record should be specified, as a comma separated values as specified, in the order below

IP_ADDRESS, ORGANIZATION_NAME, OWNER, TTL, CLASS, TYPE, DATA

Type takes one of 'A','CNAME','MX','SRV','NS','TXT', 'NAPTR' or 'TLSA'

EXAMPLE DATA:

```
"148.242.128.1","tcpwave","suc_4733.redes.tcpwave.com.","60","IN","A","148.242.128.1"
```

importobjecttype**NAME:**

importobjecttype

DESCRIPTION:

Imports the object types from a CSV file into the TCPWave IPAM.

ARGUMENTS:**--input_file**

Path on the target IPAM server to the input csv file to import the object types from. [mandatory]

--output_file

Path on the target IPAM server to the output file to write the results of the import. [mandatory]

--error_file

Path on the target IPAM server to the file to write the records that failed to import. [mandatory]

--max_errors

Maximum permissible errors. The import is aborted if the error count reaches this value. [mandatory]

EXAMPLE USAGE:

```
twc importobjecttype --input_file=/tmp/input.txt --output_file=/tmp/output.txt --  
error_file=/tmp/error.txt --max_errors=4
```

IMPORT FILE:

The information to create an object type should be specified, as a comma separated values as specified, in the order below.

```
"CODE","DESCRIPTION","LOGO_PATH","USER_DEFINED","PREFIX","SUFFIX","INITIAL_SEQUEN  
CE_NUMBER","PREFIX_ZEROS"
```

EXAMPLE DATA:

```
"CODE","DESCRIPTION","LOGO_PATH","USER_DEFINED","PREFIX","SUFFIX","INITIAL_SEQUEN  
CE_NUMBER","PREFIX_ZEROS"  
"3G Phone","Smart Phone","","N","3G","Phone","1","Yes"  
"Access Router","A Multiservice Router","","N","Access","Router","1","Yes"  
"Audio MCU","Audio MCU bridges for IP and ISDN-based  
videoconferencing.","N","Audio","MCU","1","Yes"  
"Bridge","A Generic Bridge","","N","Bri","dge","1","Yes"
```

importorg**NAME:**

importorg

DESCRIPTION:

Imports the organization provides the functionality to import the data of different components that is present in an organization into the TCPWave IPAM.

ARGUMENTS:**--input_file**

Path on the target IPAM server to the input csv file to import the organizations from. [mandatory]

--output_file

Path on the target IPAM server to the output file to write the results of the import. [mandatory]

--error_file

Path on the target IPAM server to the file to write the records that failed to import. [mandatory]

--max_errors

Maximum permissible errors. The import is aborted if the error count reaches this value. [mandatory]

EXAMPLE USAGE:

```
twc importorg --input_file=/tmp/orgazization.zip --output_file=/tmp/output.txt --
```

```
error_file=/tmp/error.txt --max_errors=4
```

IMPORT FILE:

It takes a zip file as input file that contains the CSV files of the following components listed in the below.

Note:- The names of the CSV files must match with the component file name listed below.

Component	File Name
Organization	organization.csv
Administrator	Group admingroup.csv
Administrator	user.csv
Location	location.csv
Contact	contact.csv
Domain	domain.csv
Appliance Group	appliance_group.csv
VRF	vrf.csv
Extension	tea_extension.csv
IPv4 Network	network.csv
IPv4 Subnet Group	subnetgroup.csv
IPv4 Subnet	subnet.csv
IPv4 Object	object.csv
DNS Option Template	dnsoptiontemplate.csv
DNS Authoritative Appliance	dns_auth_server.csv
DNS Cache Appliance	dns_cache_server.csv
DHCP Appliance	dhcpserver.csv
DHCP Faiover Peer	dhcp_failover_peer.csv
Shared Network	shared_network.csv
IPv4 Subnet DHCP Association	subnetdhcp.csv
Zone Template	zonetemplate.csv
Reverse Zone Template	reversezonetemplate.csv
Object Resource Record	object_rr.csv
Zone	zone.csv
Zone Resource Record	zone_rr.csv
IPv6 Network	v6network.csv
IPv6 Subnet Group	v6_subnet_group.csv
IPv6 Subnet	v6subnet.csv
IPv6 Object	v6object.csv
Address Block	addressblock.csv
Administrator Permission	adminpermission.csv
Asset	asset.csv
IPv DHCP Scope	scope.csv
Reverse Zone Resource Record	rev_zone_rr.csv
IPv6 Reverse Zone Resource Record	v6object_rr.csv

importreversezonetmpl

NAME:

importreversezonetmpl

DESCRIPTION:

Imports the DNS reverse zones templates from a csv file in the TCPWave IPAM.

ARGUMENTS:

--input_file

Path on the target IPAM server to the input csv file to import the zones from [mandatory]

--output_file

Path on the target IPAM server to the output file to write the results of the import [mandatory]

--error_file

Path on the target IPAM server to the file to write the records that failed to import [mandatory]

--max_errors

Maximum permissible errors. The import is aborted if the error count reaches this value [mandatory]

EXAMPLE USAGE:

```
twc importreversezonetmpl --input_file=/tmp/reverse_zone_template.csv --  
output_file=/tmp/output --error_file=/tmp/error --max_errors=10
```

IMPORT FILE:

The information to create a network reverse zone template should be specified, as a comma separated values as specified, in the order below

"IP_ADDR","ZONE_TEMPLATE","ORGANIZATION","MASK_LENGTH"

FIELD FORMATS:**EXAMPLE DATA:**

"192.193.219.0","Zone_Template_1","TCPWave","24"

importrevzonerr

NAME:

importrevzonerr

DESCRIPTION:

Imports the reverse zone resource records from a csv file in the TCPWave IPAM.

ARGUMENTS:

--input_file

Path on the target IPAM server to the input csv file to import the reverse zone resource records from [mandatory]

--output_file

Path on the target IPAM server to the output file to write the results

of the import [mandatory]

--error_file

Path on the target IPAM server to the file to write the records that failed to import [mandatory]

--max_errors

Maximum permissible errors. The import is aborted if the error count reaches this value [mandatory]

EXAMPLE USAGE:

```
twc importrevzonerr --input_file=/tmp/reverse_zone_rr.csv --output_file=/tmp/output --error_file=/tmp/error --max_errors=100
```

IMPORT FILE:

The information to create a reverse zone resource record should be specified, as a comma separated values as specified, in the order below

"NETWORK_IP","ORGANIZATION_NAME","OWNER","TTL","CLASS","TYPE","DATA"

EXAMPLE DATA:

```
"10.1.10.0","TCPWave","www.tcpwave.com.", "1200", "IN", "PTR", "10.1.10.5"
```

importreversezone

NAME:

importreversezone

DESCRIPTION:

Imports the DNS reverse zones from a CSV file into the TCPWave IPAM.

ARGUMENTS:

--input_file

Path of the input CSV file on the target IPAM server to import the zones [mandatory]

--output_file

Path of the output file on the target IPAM server to write the results of the import [mandatory]

--error_file

Path the error file on the target IPAM server to write the records that are failed to import [mandatory]

--max_errors

Maximum permissible errors. The import is aborted if the error count reaches this value [mandatory]

EXAMPLE USAGE:

```
twc importreversezone --input_file=/tmp/revzone.csv --output_file=/tmp/output --
error_file=/tmp/error --max_errors=10
```

IMPORT FILE:

The information to create a DNS reverse zone should be specified, as a comma separated values as specified, in the order below

```
"NAME","ORG_NAME","TMPL_NAME","DNSSEC","NSEC_OPT","MONIT","DESCRIPTION","DMZ_V
ISIBLE","CONTACT_F_NAME","CONTACT_M_NAME","CONTACT_L_NAME","CONTACT_EMAIL","A
DDRESS","MASK_LENGTH"
```

Append comma separated list of applicable extended attributes at the end.

Extended attribute column name format should be in the format:

XTN_<Extended attribute name in capital letters>.

Example: If extended attribute name is Ip, column name should be XTN_IP.

Below is the header example

```
"NAME","ORG_NAME","TMPL_NAME","DNSSEC","NSEC_OPT","MONIT","DESCRIPTION","DMZ_V
ISIBLE","CONTACT_F_NAME","CONTACT_M_NAME","CONTACT_L_NAME","CONTACT_EMAIL","A
DDRESS","MASK_LENGTH","XTN_IP"
```

FIELD FORMATS:

DNSSEC takes '0' or '1'. 1 indicates that DNSSEC must be enabled for the zone. 0 indicates that DNSSEC must be disabled.

NSEC_OPT takes 'NSEC' or 'NSEC3' as values.

MONIT takes '0' or '1'. 1 indicates that the monitoring must be enabled for this zone. 0 indicates that the monitoring must be disabled for this zone.

EXAMPLE DATA:

```
"10.1.10.in-addr.arpa","TCPWave","TestZoneTemplate","0","NSEC","0","test reverse
zone","1","John","","Smith","ohn.smith@tcpwave.com","10.1.10.0","24"
```

importipv6reversezone**NAME**

importipv6reversezone

DESCRIPTION

Imports the DNS IPv6 reverse zones from a CSV file into the TCPWave IPAM.

ARGUMENTS

--input_file

Path of the input CSV file on the target IPAM server to import the zones [mandatory]

--output_file

Path of the output file on the target IPAM server to write the results of the import [mandatory]

--error_file

Path the error file on the target IPAM server to write the records that are failed to import [mandatory]

--max_errors

Maximum permissible errors. The import is aborted if the error count reaches this value [mandatory]

--failed_entries_file

Path on the target IPAM server to the file to write the only failed record entries.

EXAMPLE

```
twc importipv6reversezone --input_file=/tmp/revzone.csv --output_file=/tmp/output --  
error_file=/tmp/error --max_errors=10 --failed_entries_file=/tmp/failed_entries
```

IMPORT FILE

The information to create a DNS reverse zone should be specified as a comma separated values as specified in the order below

"NAME","ORG_NAME","TMPL_NAME","DNSSEC","NSEC_OPT","MONIT","DESCRIPTION","DMZ_VISIB
LE","CONTACT_F_NAME","CONTACT_M_NAME","CONTACT_L_NAME","CONTACT_EMAIL","ADDRES
S","MASK_LENGTH"

Append comma separated list of applicable extended attributes at the end.

Extended attribute column name format should be in the format:

XTN_<Extended attribute name in capital letters>.

Example: If extended attribute name is IP, column name should be XTN_IP.

Below is the header example

```
"NAME","ORG_NAME","TMPL_NAME","DNSSEC","NSEC_OPT","MONIT","DESCRIPTION","DMZ_VISIB  
LE","CONTACT_F_NAME","CONTACT_M_NAME","CONTACT_L_NAME","CONTACT_EMAIL","ADDRES  
S","MASK_LENGTH"
```

FIELD FORMATS

DNSSEC takes '0' or '1'. 1 indicates that DNSSEC must be enabled for the zone. 0 indicates that DNSSEC must be disabled.

NSEC_OPT takes 'NSEC' or 'NSEC3' as values.

MONIT takes '0' or '1'. 1 indicates that the monitoring must be enabled for this zone. 0 indicates that the monitoring must be disabled for this zone.

EXAMPLE DATA

```
"0.0.0.0.0.5.ip6.arpa.","Internal","","0","NSEC3","1","","0","","","","","","5000::","24"
```

importscope

NAME:

importscope

DESCRIPTION:

Imports the DHCP scopes from a CSV file into the TCPWave IPAM.

ARGUMENTS:

--input_file

Path on the target IPAM server to the input csv file to import the DHCP scopes from [mandatory]

--output_file

Path on the target IPAM server to the output file to write the results of the import [mandatory]

--error_file
Path on the target IPAM server to the file to write the records that failed to import [mandatory]

--max_errors
Maximum permissible errors. The import is aborted if the error count reaches this value [mandatory]

EXAMPLE USAGE:

```
twc importscope --input_file=/tmp/scope.csv --output_file=/tmp/output --error_file=/tmp/error --max_errors=10
```

IMPORT FILE:

The information to create a DHCP scope should be specified, as a comma separated values as specified, in the order below

```
"START_IP","END_IP","OBJECT_TYPE","OPTION_TEMPLATE","DHCP_SERVER","NS_A","NS_PTR","DDNS_A","DDNS_PTR","DDNS_CNAME","DDNS_MX","ALLOW_CLIENT","ALLOW_VENDOR","ALLOW_USER","DENY_CLIENT","DENY_VENDOR","DENY_USER","TTL","DESCRIPTION","ORGANIZATION"
```

FIELD FORMATS:

START_IP, END_IP IPv4 address format

NS_A, NS_PTR, DDNS_A, DDNS_PTR, DDNS_CNAME, DDNS_MX take '0' or '1'

ALLOW_CLIENT, ALLOW_VENDOR, ALLOW_USER, DENY_CLIENT, DENY_VENDOR, DENY_USER

These flags determine allow or deny DHCP classes for client, vendor and user classes

EXAMPLE DATA:

```
"192.168.56.4","192.168.56.10","3G Phone","opt1","QA-ADNS-PDHCP","1","1","1","1","1","","","","","","1200","","TCPWave"
"192.168.56.14","192.168.56.20","Audio MCU","opt1","QA-ADNS-PDHCP","1","1","1","1","1","","","","","","1200","","TCPWave"
```

importsharednetwork

NAME:

importsharednetwork

DESCRIPTION:

Imports the shared networks from a CSV file into the TCPWave IPAM.

ARGUMENTS:

--input_file
Path on the target IPAM server to the input csv file to import the

shared networks from. [mandatory]

--output_file
Path on the target IPAM server to the output file to write the results of the import. [mandatory]

--error_file
Path on the target IPAM server to the file to write the records that failed to import. [mandatory]

--max_errors
Maximum permissible errors. The import is aborted if the error count reaches this value. [mandatory]

EXAMPLE USAGE:

```
twc importsharednetwork --input_file=/tmp/input.txt --output_file=/tmp/output.txt --error_file=/tmp/error.txt --max_errors=4
```

IMPORT FILE:

The information to create a shared network should be specified, as a comma separated values as specified, in the order below.

```
"NAME","ORGANIZATION_NAME","DESCRIPTION","APPLIANCE_NAME","APPLIANCE_IP"
```

EXAMPLE DATA:

```
"NAME","ORGANIZATION_NAME","DESCRIPTION","APPLIANCE_NAME","APPLIANCE_IP"  
"TCPWave shared Network","TCPWave","","dns-server02-sl0984","172.186.214.222"
```

importsubnet**NAME:**

importsubnet

DESCRIPTION:

Imports the subnets from a CSV file into the TCPWave IPAM.

ARGUMENTS:

--input_file
Path on the target IPAM server to the input csv file to import the subnets from [mandatory]

--output_file
Path on the target IPAM server to the output file to write the results of the import [mandatory]

--error_file
Path on the target IPAM server to the file to write the records that failed to import [mandatory]

--max_errors

Maximum permissible errors. The import is aborted if the error count reaches this value [mandatory]

EXAMPLE USAGE:

```
twc importsubnet --input_file=/tmp/subnet.txt --output_file=/tmp/output --error_file=/tmp/error
--max_errors=10
```

IMPORT FILE:

The information to create a subnet should be specified, as a comma separated values as specified, in the order below

"ADDRESS","MASK","NETWORK_ADDRESS","NAME","DOMAIN","SUBNET_GROUP","ROUTER_A
DDRESS","TEMPLATE_NAME","PRIMARY_DHCP_SERVER","STREET_1","STREET_2","CITY","STAT
E","COUNTRY","ZIP","DESCRIPTION","VLAN_ID","VRF","SHARED_NETWORK","ORGANIZATION"

Append comma separated list of applicable extended attributes at the end.

Extended attribute column name format should be in the format:

XTN_<Extended attribute name in capital letters>.

Example: If extended attribute name is Ip, column name should be XTN_IP.

Below is the header example with two extended attributes called Obj and Zone

"ADDRESS","MASK","NETWORK_ADDRESS","NAME","DOMAIN","SUBNET_GROUP","ROUTER_A
DDRESS","TEMPLATE_NAME","PRIMARY_DHCP_SERVER","STREET_1","STREET_2","CITY","STAT
E","COUNTRY","ZIP","DESCRIPTION","VLAN_ID","VRF","SHARED_NETWORK","ORGANIZATION"
, "XTN_OBJ", "XTN_ZONE"

EXAMPLE DATA:

"10.1.10.0","24","10.1.0.0","import-sn-test-1","tcpwave.com,dev.tcpwave.com","IT-SG","10.1.10.1","","","","600 ALEXANDER ROAD","","PRINCETON","NJ","USA","08540","sn-grp-test","","

importsubnettemplate**NAME**

importsubnettemplate

DESCRIPTION

Imports the Subnet templates from a name-value pair text file into the

TCPWave IPAM.

ARGUMENTS**--input_file**

Path on the target IPAM server to the input text file to import the
subnet templates from [mandatory]

--output_file

Path on the target IPAM server to the output file to write the results

of the import [mandatory]

--error_file
Path on the target IPAM server to the file to write the records that failed to import [mandatory]

--max_errors
Maximum permissible errors. The import is aborted if the error count reaches this value [mandatory]

--failed_entries_file
Path on the target IPAM server to the file to write the only failed record entries.

EXAMPLE

```
twc importsubnettemplate --input_file=/tmp/subnettemplate.txt --output_file=/tmp/output --error_file=/tmp/error --max_errors=10 --failed_entries_file=/tmp/failed_entries
```

FILE FORMAT

The output file format is as follows:

Each template starts with a [subnet-template] section

Each configuration section begins with a [<section name>] field followed by <name>=<value> pairs one per line

SECTIONS & CONFIGURATION PARAMETERS:

[subnet-template]

NAME Name of the subnet template.

ORGANIZATION_NAME Organization Name of the subnet template.

PRIMARY_DOMAIN Primary domain for the subnet created using this template

ROUTER_OPTION Select the first or last object as a router or give the router object offset.

DESCRIPTION Description of the subnet template.

LOCATION_NAME Location of the subnet template.

CONTACT_NAME Contact for the subnet template.

SUBNET_TYPE Non-DHCP, DHCP-Enabled or Cloud Hosted

(If you select the subnet type as DHCP-Enabled, then the system displays the DHCP Attributes fields.)

If you select the subnet type as Cloud Hosted, then the system displays the Cloud Attributes fields.)

SUBNET_GROUP Subnet group for the subnet template.

VLAN_ID ID of VLAN.

VRF Select VRF.

ENABLE_DISCOVERY Takes 'true' or 'false' to enable or disable discovery respectively.

ENABLE_RECLAIM Takes 'true' or 'false' to enable or disable reclaim respectively.

DISCOVERY_TEMPLATE Name of discovery template.

DHCP_OPTION_TEMPLATE Name of DHCP Option Template.

DHCP_APPLIANCE Name of DHCP Appliance.

DHCP_APPLIANCE_ADDRESS IP of the DHCP Appliance.

DHCP_FAILOVER_PEER Specify the DHCP failover peer.

SHARED_NETWORK Specify the shared networks.

DNS_APPLIANCES IP of DNS Appliances.

CLOUD_PROVIDER Specify cloud provider.

SECONDARY_DOMAINS Specify the secondary domains.

[address-allocations]

START_OFFSET Enter the start offset.

END_OFFSET Enter the end offset.

CLASS_CODE Select object type.

ALLOCATION_TYPE Specify if static, dynamic or reserved.

DOMAIN Name of the domain.

CLIENT_CLASS_ALLOW Specify the allowed client classes.

CLIENT_CLASS_DENY Specify the denied client classes.

USER_CLASS_ALLOW Specify the allowed user classes.

USER_CLASS_DENY Specify the user client classes.

VENDOR_CLASS_ALLOW Specify the allowed vendor classes.

VENDOR_CLASS_DENY Specify the denied vendor classes.

TTL Specify the time to live.

RES_EXPIRY_DATE Specify the expiry date for reserved objects.

EXAMPLE DATA:

[subnet-template]

NAME=v4-template

ORGANIZATION_NAME=Internal

PRIMARY_DOMAIN=tcpwave.com

ROUTER_OPTION=first

DESCRIPTION=

LOCATION_NAME=test hyd telangana India

CONTACT_NAME=ramya bali ramya.bali@tcpwave.com

SUBNET_TYPE=Non-DHCP

SUBNET_GROUP=subnet-test-group

VLAN_ID=100

VRF=test-vrf

ENABLE_DISCOVERY=yes

ENABLE_RECLAIM=yes

DISCOVERY_TEMPLATE=Default Discovery Template

DHCP_OPTION_TEMPLATE=

DHCP_APPLIANCE=

DHCP_APPLIANCE_ADDRESS=

DHCP_FAILOVER_PEER=null

SHARED_NETWORK=null

DNS_APPLIANCES=

CLOUD_PROVIDER=

SECONDARY_DOMAINS=null

[address-allocations]

START_OFFSET=10

```
END_OFFSET=1000
CLASS_CODE=3G Phone
ALLOCATION_TYPE=Static
DOMAIN=tcpwave.com
CLIENT_CLASS_ALLOW=
CLIENT_CLASS_DENY=
USER_CLASS_ALLOW=
USER_CLASS_DENY=
VENDOR_CLASS_ALLOW=
VENDOR_CLASS_DENY=
TTL=1200
RES_EXPIRY_DATE=
```

importsubnetdhcp**NAME:**

importsubnetdhcp

DESCRIPTION:

Imports the subnet to DHCP server associations from a CSV file into the TCPWave IPAM.

ARGUMENTS:**--input_file**

Path on the target IPAM server to the input csv file to import the associations from [mandatory]

--output_file

Path on the target IPAM server to the output file to write the results of the import [mandatory]

--error_file

Path on the target IPAM server to the file to write the records that failed to import [mandatory]

--max_errors

Maximum permissible errors. The import is aborted if the error count reaches this value [mandatory]

EXAMPLE USAGE:***importsubnetgroup*****NAME:**

importsubnetgroup

DESCRIPTION:

Imports the subnet groups from a CSV file into the TCPWave IPAM.

ARGUMENTS:

--input_file

Path on the target IPAM server to the input csv file to import the subnet groups from [mandatory]

--output_file

Path on the target IPAM server to the output file to write the results of the import [mandatory]

--error_file

Path on the target IPAM server to the file to write the records that failed to import [mandatory]

--max_errors

Maximum permissible errors. The import is aborted if the error count reaches this value [mandatory]

EXAMPLE USAGE:

```
twc importsubnetgroup --input_file=/tmp/input.txt --output_file=/tmp/output.txt --error_file=/tmp/error.txt --max_errors=10
```

IMPORT FILE:

The information to create a subnet group should be specified, as a comma separated values as specified, in the order below

"NAME","ORG_NAME","DESCRIPTION"

EXAMPLE DATA:

"IT_SG","TCPWave","IT Subnet Group"

importvrf

NAME:

importvrf

DESCRIPTION:

Imports the VRFs from a CSV file into the TCPWave IPAM.

ARGUMENTS:

--input_file

Path on the target IPAM server to the input csv file to import the VRFs from. [mandatory]

--output_file

Path on the target IPAM server to the output file to write the results

of the import. [mandatory]

--error_file
Path on the target IPAM server to the file to write the records that failed to import. [mandatory]

--max_errors
Maximum permissible errors. The import is aborted if the error count reaches this value. [mandatory]

EXAMPLE USAGE:

```
twc importvrf --input_file=/tmp/input.txt --output_file=/tmp/output.txt --  
error_file=/tmp/error.txt --max_errors=4
```

IMPORT FILE:

The information to create a VRF should be specified, as a comma separated values as specified, in the order below.

"NAME","ORG_NAME","DESCRIPTION"

EXAMPLE DATA:

```
"NAME","ORG_NAME","DESCRIPTION"  
"VRF1","TCPWave","Virtual routing and forwarding"
```

importzone**NAME:**

importzone

DESCRIPTION:

Imports the DNS zones from a CSV file into the TCPWave IPAM.

ARGUMENTS:

--input_file
Path on the target IPAM server to the input csv file to import the zones from [mandatory]

--output_file
Path on the target IPAM server to the output file to write the results of the import [mandatory]

--error_file
Path on the target IPAM server to the file to write the records that failed to import [mandatory]

--max_errors
Maximum permissible errors. The import is aborted if the error count reaches this value [mandatory]

EXAMPLE USAGE:

```
twc importzone --input_file=/tmp/zone.csv --output_file=/tmp/output --error_file=/tmp/error --max_errors=10
```

IMPORT FILE:

The information to create a DNS zone should be specified, as a comma separated values as specified, in the order below

```
"NAME","ORG_NAME","TMPL_NAME","DNSSEC","NSEC_OPT","AD_UPDATES","DC_IP","AD_SEC",  
"MONIT","DESCRIPTION","DMZ_VISIBLE","ACL"
```

Append comma separated list of applicable extended attributes at the end.

Extended attribute column name format should be in the format:

XTN_<Extended attribute name in capital letters>.

Example: If extended attribute name is Ip, column name should be XTN_IP.

Below is the header example with two extended attributes called Ip and domain

```
"NAME","ORG_NAME","TMPL_NAME","DNSSEC","NSEC_OPT","AD_UPDATES","DC_IP","AD_SEC",  
"MONIT","DESCRIPTION","DMZ_VISIBLE","ACL","XTN_IP","XTN_DOMAIN"
```

FIELD FORMATS:

DNSSEC takes '0' or '1'. 1 indicates that DNSSEC should be enabled for the zone. 0 indicates that DNSSEC is not enabled

NSEC_OPT takes 'NSEC' or 'NSEC3' as values

AD_UPDATES take '0' or '1'. '1' indicates Active Directory updates are enabled for this zone. '0' indicates Active Directory updates are disabled for this zone.

DC_IP is a comma separated values of IPs of domain controllers applicable for this zone.

AD_SEC takes '0' or '1'. '1' indicates Active Directory secure updates are enabled for this zone. '0' indicates Active Directory secure updates are disabled for this zone.

MONIT takes '0' or '1'. '1' indicates monitoring is enabled for this zone.
'0' indicates monitoring is disabled for this zone.

ACL is a comma separated names of ACL applicable for this zone.

EXAMPLE DATA:

```
"tcpwave.com","tcpwave","TestZoneTemplate","0","NSEC","1","192.168.1.10","1","1","test zone  
for tcpwave.com","0",""
```

importzonerr

NAME:

importzonerr

DESCRIPTION:

Imports the zone resource records from a csv file into the TCPWave IPAM.

ARGUMENTS:

--input_file

Path on the target IPAM server to the input csv file to import the resource records from [mandatory]

--output_file

Path on the target IPAM server to the output file to write the results of the import [mandatory]

--error_file

Path on the target IPAM server to the file to write the records that failed to import [mandatory]

--max_errors

Maximum permissible errors. The import is aborted if the error count reaches this value [mandatory]

EXAMPLE USAGE:

```
twc importzonerr --input_file=/tmp/arr.csv --output_file=/tmp/output --error_file=/tmp/error --max_errors=10
```

IMPORT FILE:

The information to create a zone resource records should be specified, as a comma separated values as specified, in the order below

"ZONE_NAME","ORGANIZATION_NAME","OWNER","TTL","CLASS","TYPE","DATA","EXTERNAL","DESCRIPTION"

Type takes one of 'A','AAAA','CNAME','MX','SRV','NS','TXT', 'NAPTR', 'DNAME', 'HINFO', 'CAA', 'LOC' or 'TLSA'

EXAMPLE DATA:

"tcpwave.com","TCPWave","www.tcpwave.com.", "300", "IN", "CNAME", "dev.tcpwave.com.", "0", "CNAME Record"

importzonetemplate**NAME:**

importzonetemplate

DESCRIPTION:

Imports the DNS zone templates from a CSV file into the TCPWave IPAM.

ARGUMENTS:

--input_file

Path on the target IPAM server to the input csv file to import the

zone templates from [mandatory]

--output_file

Path on the target IPAM server to the output file to write the results of the import [mandatory]

--error_file

Path on the target IPAM server to the file to write the records that failed to import [mandatory]

--max_errors

Maximum permissible errors. The import is aborted if the error count reaches this value [mandatory]

EXAMPLE USAGE:

```
twc importzonetemplate --input_file=/tmp/zonetemplate.csv --output_file=/tmp/output --error_file=/tmp/error --max_errors=10
```

IMPORT FILE:

The information to create a zone template should be specified, as a comma separated values as specified, in the order below

"NAME","ORG_NAME","DEFAULT_TTL","SOA_EMAIL","SOA_REFRESH","SOA_RETRY","SOA_EXPIRY","SOA_NEGCACHE","SOA_SERIAL","ALLOW_NOTIFY","ALLOW_QUERY","ALLOW_XFR","ALLOW_UPDATE","FORWARD","FWD_IPV4","FWD_IPV6","MASTERS","SLAVES","DESCRIPTION"

FIELD FORMATS:

DEFAULT_TTL, SOA_REFRESH, SOA_RETRY, SOA_EXPIRY, SOA_NEGCACHE accept time format as follows:

Time formats are specified, as integer/time_unit. time_unit can be one of the following values: S,MIN,H,D,W,MON,Y representing seconds, minutes, hours, days, weeks, months, years

Example: 84600/S or 30/D

SOA_SERIAL accept 'DATE' or 'NODATE'

ALLOW_NOTIFY, ALLOW_QUERY, ALLOW_XFR, ALLOW_UPDATE accept one of the following ACL formats

IPAddress/permission (192.168.0.1/Allow)

ACL-name/permission (internal/Deny)

IPAddress/mask/permission (192.168.0.0/24/Allow)

MASTERS, SLAVES are comma separated lists of IP addresses of authoritative servers acting as slaves for the reverse zone

FORWARD takes 'first' or 'only' as values

FWD_IPV4 semicolon separated list of ipv4 addresses

FWD_IPV6 semicolon separated list of ipv6 addresses

EXAMPLE DATA:

```
"TestZoneTemplate","tcpwave","52000/S","admin@tcpwave.com","","","","","","","","any/Allow,192.16  
8.1.4/Deny","any/Allow,192.168.1.3/Deny","any/Allow,192.168.1.2/Deny","any/Allow,192.168.1.1/  
Deny","","","","","192.168.1.102,192.168.1.107","","Test zone template"
```

initdb

NAME:

initdb -

DESCRIPTION:

Initializes the database from a predefined database snapshot

ARGUMENTS:

--f

Force initialize flag to initialize the database. [optional]

EXAMPLE:

```
twc initdb --flag=1
```

importlicense

NAME

importlicense

DESCRIPTION

Imports the license from a CSV file into the TCPWave IPAM.

ARGUMENTS

--input_file

Path on the target IPAM server to the input csv file to import the license from. [mandatory]

--output_file

Path on the target IPAM server to the output file to write the results of the import.

[mandatory]

--error_file

Path on the target IPAM server to the file to write the records that failed to import.

[mandatory]

--max_errors

Maximum permissible errors. The import is aborted if the error count reaches this value.

[mandatory]

--failed_entries_file

Path on the target IPAM server to the file to write the only failed record entries.

EXAMPLE

```
twc importlicense --input_file=/tmp/license.txt --output_file=/tmp/output --
error_file=/tmp/error --max_errors=10 --failed_entries_file=/tmp/failed_entries
```

IMPORT FILE

The information to import a license should be specified as a comma separated values as specified in the order below.

"APPLIANCE_NAME","APPLIANCE_IP","APPLIANCE_TYPE","LICENSE_KEY"

EXAMPLE DATA

```
"TCPWave00002Remote","192.168.0.3","DNS","00CA8D17101FD417F02EE71CED0B25A1BCAD8573
C4774F0EADEA1D8A694F1E5C783BD7D0C6AEE413960CDB05665A9B5DB5ACD4E45F3FF6F1"
```

importdnsforwarderstmpl

NAME:

importdnsforwarderstmpl

DESCRIPTION:

Imports the DNS forwarder template with forwarder zones from a CSV file into the TCPWave IPAM.

ARGUMENTS:

--input_file

Path on the target IPAM server to the input csv file to import the DNS forwarder template from [mandatory]

--output_file

Path on the target IPAM server to the output file to write the results of the import [mandatory]

--error_file

Path on the target IPAM server to the file to write the records that failed to import [mandatory]

--max_errors

Maximum permissible errors. The import is aborted if the error count reaches this value [mandatory]

--failed_entries_file

Path on the target IPAM server to the file to write the only failed record entries.

EXAMPLE

```
twc importdnsforwarderstmpl --input_file=/tmp/forwarders.txt --output_file=/tmp/output --
error_file=/tmp/error --max_errors=50 --failed_entries_file=/tmp/failed_entries
```

IMPORT FILE:

The information to create a DNS forwarder template should be specified values as specified in the order below

EXAMPLE DATA:

```
[dns-forwarders]
NAME=TestForwarder
DESCRIPTION=Description for the forwarders
[forwarder]
ZONENAME=test1.com
FORWARD=only
FORWARDERSIPV4=10.1.10.12;10.1.10.13
FORWARDERSIPV6=
ISPRESENT=true
[forwarder]
ZONENAME=zone.com
FORWARD=only
FORWARDERSIPV4=10.1.10.15;10.1.10.16
FORWARDERSIPV6=
ISPRESENT=true
[dns-forwarders]
NAME=TestTwo
DESCRIPTION=Description for the forwarders
[forwarder]
ZONENAME=test2.com
FORWARD=only
FORWARDERSIPV4=10.1.10.18;10.1.10.19
FORWARDERSIPV6=
ISPRESENT=true
[forwarder]
ZONENAME=zone3.com
FORWARD=only
FORWARDERSIPV4=10.1.10.20;10.1.10.21
FORWARDERSIPV6=
ISPRESENT=true
```

Lists

listactivelease

NAME:

listactivelease

DESCRIPTION:

Lists the active leases for a given DHCP server or subnet defined in the TCPWave IPAM.

The list includes the following information.

"IP Address", "Host Name", "Hardware Ethernet", "Status", "Start Date" and "End Date".

ARGUMENTS:

--dhcp_appliance

Address of the DHCP server. This argument is not mandatory if --subnet argument is specified.

--subnet

Address of the subnet. This argument is not mandatory if --dhcp_appliance argument is specified.

--org

Name of the organization to which specified, subnet belongs. This argument is mandatory if the user is 'FADM'.

--d

Delimiter character separating the columns. If this argument is not specified, then comma will be used as a delimiter.

EXAMPLE USAGE:

```
twc listactivelease --dhcp_appliance=10.1.10.180
```

```
twc listactivelease --subnet=10.1.10.0 --org=TCPWave --d=,
```

listavailablesubnet

NAME

listavailablesubnet

DESCRIPTION

Lists all the available within a given network in the TCPWave IPAM.

ARGUMENTS

--ip

IP address of the network for which subnets are to be listed. [mandatory]

--network_mask

Mask of the given network. [mandatory]

--subnet_mask

Mask of the subnet. [mandatory]

--org

Organization name associated with network IP. [mandatory]

EXAMPLE

```
twc listavailablesubnet --ip=10.0.10.0 --network_mask=16 --subnet_mask=24 --org=Internal
```

listzonerr

DESCRIPTION:

Lists the resource records of a managed zone from the given organization in the TCPWave IPAM
ARGUMENTS:

--name

Name of the zone. [mandatory]

--org

Name of the organization. [mandatory]

--record_type

Type of resource record.

EXAMPLE:

```
twc listzonerr --name=tcpwave.com --org=TCPWave
```

```
twc listzonerr --name=tcpwave.com --org=TCPWave --record_type=A
```

listadminrole

NAME:

listadminrole

DESCRIPTION:

Lists all the administrator roles from the TCPWave IPAM.

ARGUMENTS:

--d

Delimiter character separating the columns. If this argument is not specified, then comma will be used as a delimiter.

EXAMPLE:

```
twc listadminrole --d=,
```

listadmin

NAME:

listadmin

DESCRIPTION:

Lists the administrators defined in the TCPWave IPAM. The list includes the following information.

"FirstName", "MiddleName", "LastName", "Email", "Phone", "Organization",
"AdminPrivileges", "LastLogin", "LastChanged", "PasswordExpired",
"AccountLocked", "FailedAttempts" and "Role"

ARGUMENTS:

--d

Delimiter character separating the columns. If this argument is not specified then comma will be used as a delimiter.

--org

Organization name for which administrators are to be listed.

EXAMPLE:

twc listadmin --d=,

twc listadmin --org=TcpWave --d=,

listadmingroup

NAME:

listadmingroup

DESCRIPTION:

Lists the administrator groups defined in the TCPWave IPAM. The list includes the following information.

"Name", "Organization" and "Description".

ARGUMENTS:

--d

Delimiter character separating the columns. If this argument is not specified then comma will be used as default delimiter.

EXAMPLE:

twc listadmingroup --d=,

listadminpermission

NAME

listadminpermission

DESCRIPTION

Lists the permissions which defined in the TCPWave IPAM. The list includes the following information. "Permission Level", "Admin/Admin Group", "Function", "Value", "Select All", "Organization" and "Privilege"

ARGUMENTS

--d

Delimiter character separating the columns. If this argument is not specified then comma will be used as a delimiter.

EXAMPLE

twc listadminpermission --d=,

listappliancegroup

NAME:

listappliancegroup

DESCRIPTION:

Lists the appliance groups defined in the TCPWave IPAM.
The list includes the following information.
"Name", "Organization" and "Description"

ARGUMENTS:

--org

Organization name for which appliance groups needs to be listed. If this argument is omitted all the appliance groups in the TCPWave IPAM will be listed.

--d

Delimiter character separating the columns. If this argument is not specified, then comma will be used as default delimiter.

EXAMPLE USAGE:

twc listappliancegroup --org=TCPWave --d=,

listpatch

NAME:

listpatch

DESCRIPTION:

Lists the patches defined in the TCPWave IPAM.

ARGUMENTS:

--d

Delimiter character separating the columns. If this argument is not specified, then comma will be used as a delimiter.

EXAMPLE USAGE:

twc listpatch --d=,

listalerts

NAME:

listalerts

DESCRIPTION:

Lists the alerts defined in the TCPWave IPAM.

The list includes the following information.

"Last Check", "State Duration", "IP Address", "Host", "Service", "Status" and
"Level"

ARGUMENTS:

--level

Severity level of alerts to be listed. Takes '0', '1', '2', '3' for
OK, WARNING, CRITICAL, UNKNOWN levels respectively. If this argument is
not specified, then alerts for all severity levels will be listed.

--d

Delimiter character separating the columns. If this argument is not
specified, then comma will be used as default delimiter.

EXAMPLE USAGE:

twc listalerts --level=3 --d=,

twc listalerts

listawsimages

NAME:

listawsimages

DESCRIPTION:

Lists the Amazon machine images defined in the TCPWave IPAM.

The list includes the following information.

"Image ID", "Name", "Cloud Provider", "Organization" and "Description".

ARGUMENTS:

--d

Delimiter character separating the columns. If this argument is not
specified, then comma will be used as a delimiter.

EXAMPLE USAGE:

twc listawsimages --d=,

listawsinstance

NAME:

listawsinstance

DESCRIPTION:

Lists the AWS instance templates in the TCPWave IPAM.

ARGUMENTS:

--d

Delimiter character separating the columns. If this argument is not

specified, then comma will be used as a delimiter.

EXAMPLE USAGE:

```
twc listasinstance --d=,
```

listasnumber**NAME**

listasnumber

DESCRIPTION

Lists the Autonomous System Number defined in the TCPWave IPAM.

ARGUMENTS

--org

Organization name for which the Autonomous System Number are being listed.

--d

Delimiter character separating the columns. If this argument is not specified then comma will be used as default delimiter.

EXAMPLE

```
twc listasnumber --d=,
```

```
twc listasnumber --d=, --org=TCPWave
```

listcloudprovider**NAME:**

listcloudprovider

DESCRIPTION:

Lists the cloud providers defined in the TCPWave IPAM.

ARGUMENTS:

--org

Organization name under which the cloud provider is being created.
[mandatory]

--d

Delimiter character separating the columns. If this argument is not specified, then comma will be used as a delimiter.

EXAMPLE USAGE:

```
twc listcloudprovider --org=Internal --d=,
```

listcloudprovidertypes

NAME:

listcloudprovidertypes

DESCRIPTION:

Lists the cloud provider types in the TCPWave IPAM.

ARGUMENTS:

--d

Delimiter character separating the columns. If this argument is not specified, then comma will be used as a delimiter.

EXAMPLE USAGE:

```
twc listcloudprovidertypes --d=,
```

listcontact

NAME

listcontact

DESCRIPTION

Lists the contacts defined in TCPWave IPAM. By specifying --org parameter, contacts for a given organization can be listed if user has access. Omitting --org parameter will list the contacts from the default organization where user belong to.

ARGUMENTS

--org

Organization name for which the contacts are being listed.

--d

Delimiter character separating the columns. If this argument is not specified, comma will be used as default delimiter.

EXAMPLE

```
twc listcontact --org=Tcpwave --d=,
```

listcustomfolder

NAME:

listcustomfolder**DESCRIPTION:**

Lists the custom folders defined in the TCPWave IPAM.

The list includes the following information.

"Name", "Created By", "Created Time", "Updated By" and "Updated Time".

ARGUMENTS:

--d

Delimiter character separating the columns. If this argument is not specified, then comma will be used as default delimiter.

EXAMPLE USAGE:

```
twc listcustomfolder --d=,
```

listdhcpclass

NAME:

listdhcpclass

DESCRIPTION:

Lists the DHCP classes defined in the TCPWave IPAM.

ARGUMENTS:

--type

Type of the DHCP class. Takes 'user', 'vendor' or 'client' [mandatory]

--d

Delimiter character separating the columns. If this argument is not specified, then comma will be used as a delimiter.

EXAMPLE USAGE:

```
twc listdhcpclass --type=vendor --d=,
```

```
twc listdhcpclass --type=user
```

```
twc listdhcpclass --type=client
```

listhcpfailoverpeer

NAME:

listhcpfailoverpeer

DESCRIPTION:

Lists the DHCP failover peers defined in the TCPWave IPAM.

The list includes the following information.

"Name", "Organization", "Primary Appliance Name", "Primary Appliance Address", "Failover Appliance Name" and "Failover Appliance Address".

ARGUMENTS:

--d

Delimiter character separating the columns. If this argument is not specified, then comma will be used as a default delimiter.

EXAMPLE USAGE:

twc listdhcpfailoverpeer --d=,

listdhcpfingerprint

NAME:

listdhcpfingerprint

DESCRIPTION:

Lists the DHCP fingerprint data defined in the TCPWave IPAM.

ARGUMENTS:

--d

Delimiter character separating the columns. If this argument is not specified, then comma will be used as a delimiter.

EXAMPLE USAGE:

twc listdhcpfingerprint --d=,

listdhcpoption

NAME:

listdhcpoption

DESCRIPTION:

Lists the user defined DHCP options defined in the TCPWave IPAM.

ARGUMENTS:

--d

Delimiter character separating the columns. If this argument is not specified, then comma will be used as a delimiter.

EXAMPLE USAGE:

twc listdhcpoption -d=,

listdhcpserver

NAME:

listdhcpserver

DESCRIPTION:

Lists the DHCP servers defined in the TCPWave IPAM.

ARGUMENTS:

--d

Delimiter character separating the columns. If this argument is not

specified, then comma will be used as a delimiter.

--org

Organization name for which the DHCP appliances are being listed.

EXAMPLE USAGE:

twc listdhcpserver --d=,

listdhcpsharednetwork

NAME:

listdhcpsharednetwork - Lists the DHCP shared networks defined in the TCPWave IPAM.

DESCRIPTION:

Lists the DHCP shared networks defined in the TCPWave IPAM.

ARGUMENTS:

--org

Name of the organization from which the DHCP shared networks to be listed.

--d

Delimiter character separating the columns. If this argument is not specified then comma will be used as a delimiter.

EXAMPLE:

twc listdhcpsharednetwork --org=TCPwave --d=,

twc listdhcpsharednetwork --d=,

listdhcptmpl

NAME:

listdhcptmpl

DESCRIPTION:

Lists the DHCP option templates and policy templates defined in the TCPWave IPAM.

ARGUMENTS:

--type

Takes 'policy' or 'option'. Specifies the type of DHCP templates to be listed [mandatory]

--d

Delimiter character separating the columns. If this argument is not

specified, then comma will be used as a delimiter.

EXAMPLE USAGE:

twc listdhcptmpl --type=policy --d=,

twc listdhcptmpl --type=option --d=,

listdiscovertask

NAME:

listdiscovertask

DESCRIPTION:

Lists the discovered tasks defined in the TCPWave IPAM.

The list includes the following information.

"Command Id","Subnet","Devices Discovered","Status","Discovered By",
"Start Time" and "End Time".

ARGUMENTS:

--d

Delimiter character separating the columns. If this argument is not specified, then comma will be used as default delimiter.

EXAMPLE USAGE:

twc listdiscovertask --d=,

listdiscoverytmpl

NAME:

listdiscoverytmpl

DESCRIPTION:

Lists the discovery template defined in the TCPWave IPAM.

The list includes the following information.

"Name","Organization","Discovery Method","SNMP Request Retries",
"SNMP Response Timeout","Reverse DNS Lookup Timeout",
"Add Non-conflicting Object","Conflicting Object Accept Preferences".

ARGUMENTS:

--d

Delimiter character separating the columns. If this argument is not specified, then comma will be used as default delimiter.

EXAMPLE USAGE:

twc listdiscoverytmpl --d=,

listdnsacl

NAME:

listdnsacl**DESCRIPTION:**

Lists the DNS ACLs defined in the TCPWave IPAM.

The fields are displayed in the order of "NAME","ACL","DESCRIPTION".

ACL field is a comma separated list of ACL elements in one of the following formats:

IPAddress/permission (192.168.0.1/Allow)
ACL-name/permission (internal/Deny)
IPAddress/mask/permission (192.168.0.0/24/Allow)

ARGUMENTS:

--d

Delimiter character separating the columns. If this argument is not specified, then comma will be used as a delimiter.

EXAMPLE USAGE:

twc listdnsacl --d=,

SAMPLE OUTPUT:

"NAME","ACL","DESCRIPTION"
"none","","matches no hosts"
"any","","matches all hosts"
"localhost","","matches all the IP address(es) of the server on which DNS server is running"
"localnets","","matches all the IP address(es) and subnet masks of the server on which DNS server is running"
"internal","192.168.0.1/24/Allow,192.168.0.2/Allow","internal servers"
"external","internal/Deny,192.168.0.1/24/Allow,192.168.0.1/Allow","external servers"

listdnsforwarders**NAME:**

listdnsforwarders

DESCRIPTION:

Lists DNS forwarders which are used to resolve DNS zones that are not managed by the TCPWave. Forwarders exist on an internal 'BIND CACHE' or 'UNBOUND' DNS server in the TCPWave IPAM.

ARGUMENTS:

--appliance_ip

IP Address of the DNS internal cache server [mandatory]

--d

Delimiter character separating the columns. If this argument is not specified, then comma will be used as a delimiter.

EXAMPLE USAGE:

```
twc listdnsforwarders --appliance_ip=10.1.10.29
```

```
twc listdnsforwarders --appliance_ip=10.1.10.29 --d=,
```

listdnsreversezone**NAME:****listdnsreversezone**

Lists the DNS reverse zones defined in the TCPWave IPAM.

ARGUMENTS:**--d**

Delimiter character separating the columns. If this argument is not specified, then comma will be used as a delimiter.

EXAMPLE USAGE:

```
twc listdnsreversezone --d=,
```

twc listdnsreversezone***listdnsopttmpl*****NAME:****listdnsopttmpl****DESCRIPTION:**

Lists the DNS option templates defined in the TCPWave IPAM.

ARGUMENTS:**--d**

Delimiter character separating the columns. If this argument is not specified, then comma will be used as a delimiter.

EXAMPLE USAGE:

```
twc listdnsopttmpl --d=,
```

listipv6dnsserver**NAME****listipv6dnsserver****DESCRIPTION**

Lists the IPv6 DNS appliances defined in the TCPWave IPAM.

ARGUMENTS**--org**

Name of the organization from which the DNS appliances to be listed.

--d

Delimiter character separating the columns. If this argument is not specified then comma will be used as a delimiter.

EXAMPLE

```
twc listipv6dnsserver --org=TCPWave --d=,  
twc listipv6dnsserver --d=,
```

listdnsserver

NAME:

listdnsserver

DESCRIPTION:

Lists the DNS servers defined in the TCPWave IPAM.

ARGUMENTS:

--d

Delimiter character separating the columns. If this argument is not specified, then comma will be used as a delimiter.

EXAMPLE USAGE:

```
twc listdnsserver --d=,
```

listdnsservertmpl

NAME:

listdnsservertmpl

DESCRIPTION:

Lists the DNS servers' templates defined in the TCPWave IPAM.

ARGUMENTS:

--d

Delimiter character separating the columns. If this argument is not specified, then comma will be used as default delimiter.

EXAMPLE USAGE:

```
twc listdnsservertmpl --d=,
```

listdnsview

NAME:

listdnsview

DESCRIPTION:

Lists the DNS views defined in the TCPWave IPAM.

ARGUMENTS:**--org**

Organization name to list the DNS views from. This argument is applicable only if the user is FADM.

--d

Delimiter character separating the columns. If this argument is not specified, then comma will be used as a delimiter.

EXAMPLE USAGE:

```
twc listdnsview --d=,
```

```
twc listdnsview --org=TCPWave --d=,
```

listdnszone**NAME:**

listdnszone - Lists the DNS zones defined in the TCPWave IPAM.

DESCRIPTION:

Lists the DNS zones defined in the TCPWave IPAM.

ARGUMENTS:**--appliance_name**

Name of the appliance for which zones are to be listed. If this argument is not specified all the zones in TCPWave IPAM will be listed.

--rev_zone

It takes '1' or '0'. If it is specified as '1' only reverse zones will be listed.

--d

Delimiter character separating the columns. If this argument is not specified then comma will be used as a delimiter.

EXAMPLE:

```
twc listdnszone --appliance_name=twcvm001 --rev_zone=0 --d=,
```

```
twc listdnszone --rev_zone=1 --d=,
```

```
twc listdnszone
```

listdnszonetmpl**NAME:**

listdnszonetmpl

DESCRIPTION:

Lists the DNS zone templates defined in the TCPWave IPAM.

The list includes the following information.

"Name", "Organization" and "Description".

ARGUMENTS:

--org

Name of the organization from which DNS zone templates must be listed.

--d

Delimiter character separating the columns. If this argument is not specified, then comma will be used as a delimiter.

EXAMPLE USAGE:

```
twc listdnszonetmpl --d=,
```

```
twc listdnszonetmpl --org=TCPWave --d=,
```

listdomain

NAME:

listdomain

DESCRIPTION:

Lists the DNS domains defined in the TCPWave IPAM.

The list includes the following information.

"Domain", "Organization" and "Description"

ARGUMENTS:

--d

Delimiter character separating the columns. If this argument is not specified, then comma will be used as default delimiter.

EXAMPLE USAGE:

```
twc listdomain --d=,
```

listext

NAME:

listext

DESCRIPTION:

Lists the extended attributes defined in the TCPWave IPAM.

ARGUMENTS:

--d

Delimiter character separating the columns. If this argument is not specified, then comma will be used as a delimiter.

--entity

Entity for which the extension attributes to be listed. It takes one of 'admin', 'network', 'subnet', 'object' or 'zone'.

EXAMPLE USAGE:

```
twc listext --d=,  
twc listext --entity=admin --d=,  
twc listext --entity=network --d=,  
twc listext --entity=subnet --d=,  
twc listext --entity=object --d=,  
twc listext --entity=zone --d=,
```

listextvalue**NAME****listextvalue****DESCRIPTION**

Lists the extended attribute values of specified entity defined in the TCPWave IPAM.
Applicable entities are 'admin', 'network', 'subnet', 'object' and 'zone'

ARGUMENTS**--name**

Name of the extended attribute. [mandatory]

--entity

Entity of the extended attribute. [mandatory]

--org

Name of the organization. [mandatory]

--d

Delimiter character separating the columns. If this argument is not specified then comma will be used as a delimiter.

EXAMPLE:

```
twc listextvalue --name="CHG_TKT" --entity=network --org=TCPWave --d=,
```

listdumps**NAME:****listdumps****DESCRIPTION:**

List all available database snapshots for recovery

ARGUMENTS:**--d**

Delimiter character separating the columns. If this argument is not specified, then comma will be used as a delimiter

EXAMPLE USAGE:

twc listdumps --d=,

listfunctions

NAME

listfunctions

DESCRIPTION

Lists all the functions which defined in the TCPWave IPAM. The list includes the following information.

"Name","Granular Support","Description"

ARGUMENTS

--d

Delimiter character separating the columns. If this argument is not specified then comma will be used as a delimiter.

EXAMPLE

twc listfunctions --d=,

listipv6subnettmp**NAME**

listipv6subnettmp

DESCRIPTION

Lists the IPv6 subnet templates created in the TCPWave IPAM.

EXAMPLE

twc listipv6subnettmp --d=,

listfirewalltmp**NAME:**

listfirewalltmp

DESCRIPTION:

Lists the firewall templates defined in the TCPWave IPAM.

ARGUMENTS:

--org

Organization name to list the firewall templates from. This argument is applicable only if the user is FADM.

--d

Delimiter character separating the columns. If this argument is not specified, then comma will be used as a delimiter.

EXAMPLE USAGE:

twc listfirewalltmp --d=,

twc listfirewalltmp --org=TCPWave --d=,

listglobalopts**NAME:**

listglobalopts

DESCRIPTION:

Lists the Global Options defined in the TCPWave IPAM.

ARGUMENTS:

--d

Delimiter character separating the columns. If this argument is not specified, then comma will be used as a delimiter.

EXAMPLE USAGE:

twc listglobalopts --d=,

listhaipam

NAME:

listhaipam

DESCRIPTION:

Lists the IPAM available in high availability cluster.

The list includes the following information.

"IPAM", "Address", "Master IPAM", "Remote Managed" and "Remote Connected".

ARGUMENTS:

--d

Delimiter character separating the columns. If this argument is not specified, then comma will be used as default delimiter.

EXAMPLE USAGE:

twc listhaipam

listharemove

NAME:

listharemove

DESCRIPTION:

Lists the remote appliances available in high availability cluster.

The list includes the following information.

"Remote", "Address", "Managed By" and "Preferred Order".

ARGUMENTS:

--d

Delimiter character separating the columns. If this argument is not specified, then comma will be used as default delimiter.

EXAMPLE USAGE:

twc listharemove

listipv6dhcpserver

NAME:

listipv6dhcpserver

DESCRIPTION:

Lists the IPv6 DHCP servers defined in the TCPWave IPAM.

ARGUMENTS:

--d

Delimiter character separating the columns. If this argument is not specified, then comma will be used as a delimiter.

EXAMPLE USAGE:

twc listipv6dhcpserver --d=,

listipv6dhcptmpl**NAME:**

listipv6dhcptmpl

DESCRIPTION:

Lists the IPv6 DHCP option templates defined in the TCPWave IPAM.

ARGUMENTS:

--d

Delimiter character separating the columns. If this argument is not specified, then comma will be used as a delimiter.

EXAMPLE USAGE:

twc listipv6dhcptmpl --d=,

listipv6network**NAME:**

listipv6network

DESCRIPTION:

Lists the IPv6 networks defined in the TCPWave IPAM.

ARGUMENTS:

--d

Delimiter character separating the columns. If this argument is not specified, then comma will be used as a delimiter.

EXAMPLE USAGE:

twc listipv6network --d=,

listipv6object**NAME**

listipv6object

DESCRIPTION

Lists the objects in a given IPv6 subnet defined in the TCPWave IPAM.

ARGUMENTS

--subnet

IPv6 address of the target subnet to list the objects from. [mandatory]

--org

Organization name of the associated IPv6 subnet. [mandatory]

--d

Delimiter character separating the columns. If this argument is not specified then comma will be used as default delimiter.

EXAMPLE

```
twc listipv6object --subnet=2000::a000:0:0:0:0 --org=TCPWave --d=,
```

listipv6scope**NAME:**

listipv6scope

DESCRIPTION:

Lists the IPv6 DHCP scopes defined in the TCPWave IPAM.

ARGUMENTS:

--d

Delimiter character separating the columns. If this argument is not specified, then comma will be used as a delimiter.

EXAMPLE USAGE:

twc listipv6scope --d=,

listipv6subnet**NAME**

listipv6subnet

DESCRIPTION

Lists the IPv6 subnets from a block defined in the TCPWave IPAM.

ARGUMENTS

--block_ip

IPv6 address of the target block from which subnets are to be listed. [mandatory]

--org

Organization name of the associated IPv6 block. [mandatory]

--d

Delimiter character separating the columns. If this argument is not specified then comma will be used as default delimiter.

EXAMPLE

```
twc listipv6subnet --block_ip=2000:: --org=TCPWave --d=,
```

listipv6subnetgroup

NAME

listipv6subnetgroup

DESCRIPTION

Lists the subnet groups defined in the TCPWave IPAM.

ARGUMENTS

--org

Organization name for which subnet groups needs to be listed. If this argument is omitted all the subnet groups in the TCPWave IPAM will be listed.

--d

Delimiter character separating the columns. If this argument is not specified then comma will be used as a delimiter

EXAMPLE

```
twc listipv6subnetgroup --org=tcpwave --d=,
```

listipv6block

NAME

listipv6block

DESCRIPTION

Lists the IPv6 address blocks from an IPv6 address pool defined in the TCPWave IPAM.

ARGUMENTS

--pool_ip

Address of the target IPv6 address pool from which block has to be listed. [mandatory]

--org

Name of the organization to which IPv6 pool exist. [mandatory]

--d

Delimiter character separating the columns. If this argument is not specified then comma will be used as a delimiter.

EXAMPLE

```
twc listipv6block --pool_ip=2001:db8:: --org=TCPWave --d=,
```

listipv6pool

NAME

listipv6pool

DESCRIPTION

Lists the IPv6 address pool defined in the TCPWave IPAM.

ARGUMENTS

--org

Name of the organization from which IPv6 address pools has to be listed. [mandatory]

--d

Delimiter character separating the columns. If this argument is not specified then comma will be used as a delimiter.

EXAMPLE

```
twc listipv6pool --org=TCPWave --d=,
```

listipv6dnsreversezone**NAME**

```
listipv6dnsreversezone
```

DESCRIPTION

Lists the IPv6 DNS reverse zones defined in the TCPWave IPAM.

ARGUMENTS

```
--org
```

Organization name from which the reverse zone to be listed.

```
--d
```

Delimiter character separating the columns. If this argument is not specified then comma will be used as a delimiter.

EXAMPLE

```
twc listipv6dnsreversezone --org=TCPWave --d=,
```

```
twc listipv6dnsreversezone --d=,
```

listlocation**NAME:**

```
listlocation
```

DESCRIPTION:

Lists the locations defined in the TCPWave IPAM.

ARGUMENTS:

```
--org
```

Organization name for which the locations are being listed. This argument is applicable only for Functional administration. Omitting this argument will list all the locations in the TCPWave IPAM.

```
--d
```

Delimiter character separating the columns. If this argument is not specified, then comma will be used as a delimiter.

EXAMPLE USAGE:

```
twc listlocation --org=TCPWave --d=,
```

```
twc listlocation --d=,
```

listlogchannel**NAME:**

listlogchannel

DESCRIPTION:

Lists the DNS log channels defined in the TCPWave IPAM.

ARGUMENTS:

--d

Delimiter character separating the columns. If this argument is not specified, then comma will be used as a delimiter.

EXAMPLE USAGE:

twc listlogchannel --d=,

listmirroredzone

NAME:

listmirroredzone

DESCRIPTION:

Lists the DNS managed mirrored zones defined in the TCPWave IPAM.

The list includes the following information.

"Name", "Organization", "Zone", "Created By", "Created Time", "Updated By", "Updated Time" and "Description".

ARGUMENTS:

--d

Delimiter character separating the columns. If this argument is not specified, then comma will be used as default delimiter.

EXAMPLE USAGE:

twc listmirroredzone --d=,

listmicrosoftdhcpserver

NAME:

listmicrosoftdhcpserver

DESCRIPTION:

Lists the Microsoft DHCP appliances defined in the TCPWave IPAM.

ARGUMENTS:

--d

Delimiter character separating the columns. If this argument is not specified, then comma will be used as a delimiter.

EXAMPLE USAGE:

twc listmicrosoftdhcpserver --d=,

listmicrosoftdnsserver

NAME:

listmicrosoftdnsserver

DESCRIPTION:

Lists the Microsoft DNS appliances defined in the TCPWave IPAM.

ARGUMENTS:

--d

Delimiter character separating the columns. If this argument is not specified, then comma will be used as a delimiter.

EXAMPLE USAGE:

twc listmicrosoftdnsserver --d=,

listnetwork**NAME**

listnetwork

DESCRIPTION

Lists the networks defined in the TCPWave IPAM.

ARGUMENTS

--org

Organization name for which the networks are being listed.

--d

Delimiter character separating the columns. If this argument is not specified, then comma will be used as a delimiter.

EXAMPLE

twc listnetwork --org=TCPWave --d=,

listnsmtmpl

NAME***listnsmtmpl*****DESCRIPTION**

Lists the NSM templates defined in the TCPWave IPAM. The list includes the following information.
"NSM Template name", "Organization" and "Description"

ARGUMENTS

--d

Delimiter character separating the columns. If this argument is not specified then comma will be used as default delimiter.

EXAMPLE

twc listnsmtmpl --d=,

listmicrosoftadsites**NAME**

listmicrosoftadsites

DESCRIPTION

Lists Microsoft AD sites from the TCPWave IPAM.

ARGUMENTS**--ip**

IP address of the appliance. [mandatory]

--org

Name of the organization. [mandatory]

EXAMPLE

twc listmicrosoftadsites --ip=10.0.0.10 --org=TCPWave

listobject**NAME:**

listobject

DESCRIPTION:

Lists the objects in a given subnet defined in the TCPWave IPAM.

ARGUMENTS:**--subnet**

Subnet address to list the objects from. [mandatory]

--org

Name of the organization to which specified, subnet belongs. This argument is mandatory if the user is 'FADM'.

--class_code

Specifying this argument filters the list by the given class code. No filter is applied if the argument is not specified.

--alloc_type

Takes static, dynamic, manual, auto or free. Specifying this argument filters the list by the given allocation type. 'free' lists the blocks of unallocated IP address in the format StartIP, BlockAddress, EndIP. If this argument is not specified, all the objects for allocation type static, dynamic, manual, auto will be displayed.

--expand

Takes 0 | 1. Applicable only when alloc_type is free. '1' lists all the unallocated IP addresses. '0' lists the blocks of unallocated IP address in the format StartIP, BlockAddress, EndIP. Default is 0.

--d

Delimiter character separating the columns. If this argument is not specified, then comma will be used as a delimiter.

EXAMPLE USAGE:

```
twc listobject --d=, --subnet=9.0.0.0 --class_code=Router --org=TCPWave  
twc listobject --d=, --subnet=9.0.0.0 --alloc_type=static --org=TCPWave  
twc listobject --d=, --subnet=9.0.0.0 --alloc_type=free --org=TCPWave  
twc listobject --d=, --subnet=9.0.0.0 --alloc_type=free --expand=1 --org=TCPWave
```

listobjecttype

NAME:

listobjecttype

DESCRIPTION:

Lists the object types defined in the TCPWave IPAM.

The list includes the following information.

"Object Type", "Prefix", "Suffix", "ISN" and "Prefix ISN with Zeros"

ARGUMENTS:

--d

Delimiter character separating the columns. If this argument is not specified, then comma will be used as default delimiter.

EXAMPLE USAGE:

```
twc listobjecttype --d=,
```

listorg

NAME:

listorg

DESCRIPTION:

Lists the organizations defined in the TCPWave IPAM.

The list includes the following information.

"Name", "Root Zone" and "Description"

ARGUMENTS:

--d

Delimiter character separating the columns. If this argument is not specified, then comma will be used as default delimiter.

EXAMPLE USAGE:

```
twc listorg --d=,
```

listroundrobin

NAME

listroundrobin**DESCRIPTION**

Lists the DNS round-robs for a given domain defined in the TCPWave IPAM.

ARGUMENTS

--domain

Domain Name for which round robins are to be listed [mandatory]

--org

Name of the organization in which specified domain exists. [mandatory]

--d

Delimiter character separating the columns. If this argument is not specified then comma will be used as a delimiter.

EXAMPLE

twc listroundrobin --domain=tcpwave.com --org=TCPWave --d=,

listrpztmp

NAME:

listrpztmp

DESCRIPTION:

Lists the DNS Response policy zone(RPZ) templates defined in the TCPWave IPAM.

ARGUMENTS:

--d

Delimiter character separating the columns. If this argument is not specified, then comma will be used as a delimiter.

EXAMPLE USAGE:

twc listrpztmp --d=,

listrr

NAME:

listrr

DESCRIPTION:

Lists the resource records in the TCPWave IPAM.

ARGUMENTS:

--scope

Resource records type. Takes 'zone','object' or 'network' [mandatory]

--ip

IP address of the network or object for which you want resource records.

If the scope is network or object then it is mandatory.

--zone_name

Zone name for which the resource records are to be listed. If the scope is zone then it is mandatory.

--org

Organization name in which specified, scope exists. This argument is mandatory if the user is FADM.

--is_proxy

DNS Proxy root zone flag. It takes '0' or '1'. If it is specified, as '1' resource records will be listed from proxy root zone. If it is specified, as '0' resource records will be listed from root zone. This argument is applicable when --scope=zone and --zone_name=.(dot).

--d

Delimiter character separating the columns. If this argument is not specified, then comma will be used as a delimiter.

EXAMPLE USAGE:

```
twc listrr --scope=zone --zone_name=tcpwave.com --org=TCPWave --d=,
```

```
twc listrr --scope=zone --zone_name=. --org=TCPWave --is_proxy=0 --d=,
```

```
twc listrr --scope=network --ip=1.10.10.0 --org=TCPWave --d=,
```

```
twc listrr --scope=object --ip=192.168.2.9 --org=TCPWave --d=,
```

listrootaccessmgmt

NAME:

listrootaccessmgmt

DESCRIPTION:

Lists the Vault Types supported by the TCPWave IPAM.

ARGUMENTS:

- NA

EXAMPLE USAGE:

```
twc listrootaccessmgmt
```

listscheduledjob

NAME:

listscheduledjob

DESCRIPTION:

Lists the scheduled jobs defined in the TCPWave IPAM.

ARGUMENTS:

--d

Delimiter character separating the columns. If this argument is not specified, then comma will be used as a delimiter.

EXAMPLE USAGE:

twc listscheduledjob --d=,

listscope

NAME:

listscope

DESCRIPTION:

Lists the DHCP scopes defined in the TCPWave IPAM.

ARGUMENTS:

--d

Delimiter character separating the columns. If this argument is not specified, then comma will be used as a delimiter.

EXAMPLE USAGE:

twc listscope --d=,

listserversforzone

NAME:

listserversforzone

DESCRIPTION:

Lists the DNS servers for a given zone or zone template defined in the TCPWave IPAM.

ARGUMENTS:

--type

Takes 'zone' or 'template'. Indicates whether the dns servers displayed are for a zone or a zone template [mandatory]

--name

Name of the zone or zone template for which the list of dns servers are to be displayed [mandatory]

--org

organization name for which this operation is being performed. This argument is mandatory if user is FADM.

--d

Delimiter character separating the columns. If this argument is not specified, then comma will be used as a delimiter.

EXAMPLE USAGE:

twc listserversforzone --type=zone --name=tcpwave.com --org=TCPWave --d=,

twc listserversforzone --type=template --name="default-zone-template" --org=TCPWave --d=,

listsubnet

NAME:

listsubnet

DESCRIPTION

Lists the subnets defined in the TCPWave IPAM.

ARGUMENTS

--network

IP address of the network for which subnets are to be listed. [mandatory]

--org

Name of the organization to which specified network belongs. [mandatory]

--d

Delimiter character separating the columns. If this argument is not specified then comma will be used as a delimiter.

EXAMPLE

```
twc listsubnet --network=50.0.0.0 --org=TCPWave --d=,
```

listsubnetgroup**NAME:**

listsubnetgroup

DESCRIPTION:

Lists the subnet groups defined in the TCPWave IPAM.

ARGUMENTS:**--org**

Organization name for which subnet groups needs to be listed. If this argument is omitted all the subnet groups in the TCPWave IPAM will be listed.

--d

Delimiter character separating the columns. If this argument is not specified, then comma will be used as a delimiter

EXAMPLE USAGE:

twc listsubnetgroup --org=tcpwave --d=,

listsubnetsforserver**NAME:**

listsubnetsforserver

DESCRIPTION:

Lists the subnets associated with a given DHCP Server defined in the TCPWave IPAM.

ARGUMENTS:**--appliance_ip**IP Address of the DHCP server for which subnets are to be listed
[mandatory]**--d**

Delimiter character separating the columns. If this argument is not specified, then comma will be used as a delimiter.

EXAMPLE USAGE:

twc listsubnetsforserver --appliance_ip=10.1.10.10 --d=,

listipv4subnettemplate**NAME:**

listipv4subnettemplate

DESCRIPTION:

Lists the IPv4 Subnet templates created in the TCPWave IPAM.

ARGUMENTS:

EXAMPLE USAGE:

```
twc listipv4subnettemplate --d=,
```

listrestricteddomain

listrestricteddomain - Lists the domain name data from the TCPWave IPAM.

DESCRIPTION:

Lists all the restricted domains in the specified organization, If the domain name is specified, else it will lists only specified domain data.

ARGUMENTS:

--domain

Name of the domain.

--org

Name of the organization. [mandatory]

EXAMPLE:

```
twc listrestricteddomain --domain=tcp.com --org=TcpWave
```

```
twc listrestricteddomain --org=TcpWave
```

listpolicycompliance**NAME**

listpolicycompliance

DESCRIPTION

Displays the DNS, DHCP and IPAM policy compliance check information from the TCPWave IPAM.

ARGUMENTS

--ip

IP address of an appliance.

--org

Organization name to which the appliance or appliance group belongs. It is mandatory to display the policy compliance check information on a particular 'appliances' or 'appliance group'. It is not mandatory for IPAM appliance.

--type

Type of the appliance. It takes 'BIND AUTH' or 'BIND CACHE' or 'UNBOUND' or 'NSD' or 'DNS PROXY' 'DHCP' or 'IPAM' as a input. If want to display the policy compliance on all the DNS appliances type should be 'DNS'. [mandatory]

--level

Policy compliance runs based on the level. It accepts 'appliance' or 'appliance_group' as a input. Specify type as 'appliance' if want to display compliance check on particular appliance or all the

appliance. Type should be specified as appliance_group if want to display compliance check on all the appliances that belongs to a particular appliance group. [mandatory]

--all

It accepts '1' or '0' value. '1' indicates all the 'appliances' or 'appliance groups' are eligible to list policy compliance.'0' indicates particular 'appliance' or 'appliance group' is eligible to list policy compliance.

--appliance_group

Appliance group name to list policy compliance on particular group. If want to list the appliance group policy compliance on all the DNS appliances type should be 'DNS'.

EXAMPLE

```
twc listpolicycompliance --type="DNS" --level="appliance" --all=1 --org=TCPWave
```

```
twc listpolicycompliance --type="BIND AUTH" --level="appliance" --all=0 --ip=192.168.10.15 --org=TCPWave
```

```
twc listpolicycompliance --type="DHCP" --level="appliance" --all=0 --ip=192.168.12.10 --org=TCPWave
```

```
twc listpolicycompliance --type="DHCP" --level="appliance" --all=1 --org=TCPWave
```

```
twc listpolicycompliance --type="IPAM" --level="appliance" --all=1
```

```
twc listpolicycompliance --type="IPAM" --level="appliance" --all=0 --ip=192.168.25.14
```

```
twc listpolicycompliance --type="DHCP" --level="appliance_group" --all=1 --org=TCPWave
```

```
twc listpolicycompliance --type="DNS" --level="appliance_group" --all=0 --appliance_group=TCPWave-group --org=TCPWave
```

```
twc listpolicycompliance --type="IPAM" --level="appliance_group" --all=1
```

listnsmtmpl

NAME

listnsmtmpl

DESCRIPTION

Lists the NSM templates defined in the TCPWave IPAM. The list includes the following information. "NSM Template name", "Organization" and "Description"

ARGUMENTS

--d

Delimiter character separating the columns. If this argument is not specified then comma will be used as default delimiter.

EXAMPLE

```
twc listnsmtmpl --d=,
```

listvrf**NAME**

listvrf

DESCRIPTION

Lists the VRFs defined in the TCPWave IPAM.

ARGUMENTS`--org`

Name of the organization from which the VRFs to be listed.

`--d`

Delimiter character separating the columns. If this argument is not specified then comma will be used as a delimiter.

EXAMPLE

```
twc listvrf --org=TCPWave --d=,
```

```
twc listvrf --d=,
```

haipampreference**NAME:**

haipampreference

DESCRIPTION:

Set the IPAM appliance preference list for the remote in the high availability cluster.

ARGUMENT:`--preferred_ipam`

IP address of the preferred IPAM. It accepts the comma separated IPAM IP addresses, if this argument not specified mentioned remotes IPAM preference will be removed.

`--remote_ip`

IP address of the remotes for which IPAM appliance preference to be changed.

It accepts the comma separated IP addresses. [Mandatory]

EXAMPLE USAGE:

```
twc          haipampreference--preferred_ipam=192.168.0.235-remote_ip=192.168.0.236,  
192.168.0.239,192.168.0.238
```

Other Commands***applydraction*****NAME:**

applydraction

DESCRIPTION:

Apply the specified, disaster recovery action in the TCPWave IPAM.

ARGUMENTS:

--action
Action to be applied. Takes one of the following:
'Failover', 'Suspend File Sync'
or 'Resume File Sync'.

--flag
Force apply flag to configure disaster recovery.

EXAMPLE USAGE:

```
twc applydraction --action=Failover --flag=1  
twc applydraction --action="Suspend File Sync" --flag=0  
twc applydraction --action="Resume File Sync" --flag=0
```

applysnapshot**NAME:**

applysnapshot

DESCRIPTION:

Loads a database snapshot into a recovery database in the TCPWave IPAM.

ARGUMENTS:

--dump_dir [mandatory]
Dump directory from which snapshot changes are to be applied.

EXAMPLE USAGE:

```
twc applysnapshot --dump_dir=/tmp/dumps/Dump_1408948935
```

autologin**NAME:**

autologin

DESCRIPTION:

Enables/disables auto login access to the TCPWave IPAM through CLI.
Auto login enables users to login to the TCPWave IPAM through CLI with the pre-configured password.
After enabling auto login, IPAM user must be associated with a LINUX user (Ex: twc autologinmanager --map_user=linux-user --to_users=nadm1,padm3,uadm123)
and set the password of the IPAM user using autologinmanager command (Ex: twc autologinmanager --user=ipamuser).

ARGUMENTS:

--enable

Auto login enable/disable flag. It takes either '1' or '0'.
If it is specified, as '1', auto login will be enabled from CLI.
If it is specified, as '0', auto login will be disabled from CLI.

--status

Auto login status flag. It takes only '1'.
If it is specified, auto login status will be shown.

EXAMPLE USAGE:

twc autologin --enable=1

twc autologin --enable=0

twc autologin --status=1

autologinmanager

NAME:

autologinmanager

DESCRIPTION:

Performs auto login configuration operations in the TCPWave IPAM CLI.
Auto login enables users to login to the TCPWave IPAM through CLI with the
pre-configured password.

After enabling auto login, IPAM user must be associated with a
LINUX user (Ex: twc autologinmanager --map_user=linux-user
--to_users=nadm1,padm3,uadm123)
and set the password of the IPAM user using autologinmanager
command (Ex: twc autologinmanager --user=ipamuser).

ARGUMENTS:

--map_user

Name of the LINUX user. This LINUX user will be associated to the IPAM
users specified, using to_users argument.

--to_users

Comma separated list of IPAM users to be associated with the specified,
LINUX user.

--user

Login name of the user for which password must be set after enabling auto
login.

--show_users

Name of the LINUX user for which IPAM user mappings must be shown.

--remove_user

Name of the LINUX user for which IPAM user mappings have to be removed.

EXAMPLE USAGE:

```
twc autologinmanager --map_user=linux-user --to_users=nadm1,padm3,uadm123
```

```
twc autologinmanager --user=ipamuser
```

```
twc autologinmanager --show_users=linux-user
```

```
twc autologinmanager --remove_user=linux-user
```

addnsmtempl

NAME

```
addnsmtempl
```

DESCRIPTION

Create a NSM template for a given organization in the TCPWave IPAM.

ARGUMENTS

```
--org
```

Organization name for which the NSM template is being created. [mandatory]

```
--tmpl_name
```

Name of the NSM template. [mandatory]

```
--desc
```

Description for the NSM template.

```
--network_interface
```

Network interface for NSM template. [mandatory]

```
--anomaly_detection
```

It takes 'true' or 'false'. If it is true, Anomaly detection will enable on the NSM template. If it is 'false', Anomaly detection will disable on the nsm template.

```
--ml_model
```

It accepts the numeric value from '0' to '7'.

```
--intrusion_detection
```

It takes 'true' or 'false'.If it is true, Intrusion detection will enable on the NSM template. If it is 'false', Intrusion detection will disable on the NSM template.

```
--intrusion_prevention
```

It takes 'true' or 'false'.If it is true, Intrusion prevention will enable on the NSM template.If it is 'false', Intrusion prevention will disable on the NSM template.

```
--ips_rules
```

It takes multiple rules by separating with '|' symbol.

```
--rule
```

It takes address or port, name, and value for the rule variable by separating with comma. It can accept multiple values by separating with pipe symbol. Example:
address,HOME,10.1.10.1|port,HOME_NET,123

EXAMPLE:

```
twc addnsmtmpl --org=TCPWave --tmpl_name=FirstTemplate --network_interface=eth0 --anomaly_detection=true --ml_model=1 --intrusion_detection=true --intrusion_prevention=true --ips_rules="alert dns any any -> any any(msg:TCPWAVE DNS TITAN This is a alert test for Example; dns_query; content:www.example.com; depth:14; fast_pattern; endswith; nocase; classtype:pup-activity; sid:9999991; rev:33;)" --rule="address,HOME,10.1.10.1|port,NET,123" --desc="TCPWave NSM Template"
```

```
twc addnsmtmpl --org=TCPWave --tmpl_name=FirstTemplate --network_interface=eth0 --anomaly_detection=false --ml_model=0 --intrusion_detection=false --intrusion_prevention=false --desc="TCPWave NSM Template"
```

backupbinlog

NAME

backupbinlog

DESCRIPTION:

Backs up database incremental changes. Binary logs are copied under binlog directory of the latest dump.

ARGUMENTS:

EXAMPLE USAGE:

```
twc backupbinlog
```

changepassword

NAME:

changepassword

DESCRIPTION:

Changes the password of administrator(s) in the TCPWave IPAM.
If --login_name option is not specified; the current user password will be changed.

ARGUMENTS:

--login_name

Comma separated list of login name of the administrators whose password must change.

EXAMPLE USAGE:

```
twc changepassword
```

```
twc changepassword --login_name=john
```

```
twc changepassword --login_name=john,smith,peter
```

changereference**NAME:**

changereference

DESCRIPTION:

Updates the references of resources records which point to a specific IP address with new IP address in the TCPWave IPAM. It updates only A and PTR resource records. After update, it does sync operation for all the zone whose resource records have been updated.

ARGUMENTS:

--from_ip

Existing reference of the resource records which is to be changed.
[mandatory]

--to_ip

New reference of the resource records. [mandatory]

--org

Name of the organization to which the resource records belong. This argument is mandatory if user is 'FADM'.

EXAMPLE USAGE:

```
twc changereference --from_ip=192.193.102.175 --to_ip=192.193.111.222 --org=TCPWave
```

changerootpassword**NAME:**

changerootpassword

DESCRIPTION:

Root Password should have at least 8 characters, 1 upper case alphabet, 1 lower case alphabet, 1 numeric character and 1 special symbol.

ARGUMENTS:

--ip

IP Address of the DHCP/DNS server. [mandatory]

--remote_type

Type of the remote server. It takes either 'DHCP' or 'DNS'. [mandatory]

EXAMPLE USAGE:

```
twc changerootpassword --ip=10.1.10.180 --remote_type=DHCP
```

```
twc changerootpassword --ip=10.1.10.188 --remote_type=DNS
```

clearaudit**NAME:**

```
clearaudit
```

DESCRIPTION:

Clears the audits older than the specified, number of days from the TCPWave IPAM.

ARGUMENTS:

```
--days
```

Number of days audit to be preserved. [mandatory]

EXAMPLE USAGE:

```
twc clearaudit --days=90
```

clonednszone**NAME:**

```
clonednszone
```

DESCRIPTION:

Clone an existing DNS Zone to create a new DNS zone in the TCPWave IPAM.

ARGUMENTS:

```
--existing_zone
```

Name of the existing DNS Zone to be cloned. [mandatory]

```
--new_zone
```

Name of the DNS Zone to be created. [mandatory]

```
--org
```

Organization name associated with the existing zone. If this argument is not specified, organization associated with the user will be used.

EXAMPLE USAGE:

```
twc clonednszone --existing_zone="tcpwave.com" --new_zone="tcpwave1.com" --org=TCPWave
```

```
twc clonednszone --existing_zone="tcpwave.com" --new_zone="tcpwave1.com"
```

configureha**NAME:**

```
configureha
```

DESCRIPTION:

Configures the high availability cluster in the TCPWave IPAM.

ARGUMENTS:

- NA

EXAMPLE USAGE:

twc configureha

configrootaccessmgmt

NAME:

configrootaccessmgmt

DESCRIPTION:

Configures primary and secondary TCPWave IPAM appliances and active remote appliances with selected root access management preferences.

Supported vault types are Native, CyberArk and Hashicorp.

EXAMPLE:

```
twc configrootaccessmgmt --vault_type=Native
```

```
twc configrootaccessmgmt --vault_type=CyberArk
```

```
twc configrootaccessmgmt --vault_type=Hashicorp --vault_server_host=1.1.1.1 --  
vault_server_port=1122 --use_ssl=false
```

```
twc configrootaccessmgmt --vault_type=Hashicorp --vault_server_host=1.1.1.1 --  
vault_server_port=1122 --use_ssl=true --vault_server_cert_path="/tmp/rootCA.crt"
```

ARGUMENTS:

EXAMPLE USAGE:

```
twc configrootaccessmgmt --vault_type=Native
```

```
twc configrootaccessmgmt --vault_type=CyberArk
```

```
twc configrootaccessmgmt --vault_type=Hashicorp --vault_server_host=1.1.1.1 --  
vault_server_port=1122 --use_ssl=false
```

```
twc configrootaccessmgmt --vault_type=Hashicorp --vault_server_host=1.1.1.1 --  
vault_server_port=1122 --use_ssl=true --vault_server_cert_path="/tmp/rootCA.crt"
```

baselinepolicycompliance

NAME

baselinepolicycompliance

DESCRIPTION

Set Baseline policy compliance to DNS, DHCP and IPAM appliances in the TCPWave IPAM.

ARGUMENTS

--ip

IP address of an appliance. [mandatory]

--type

Type of the appliance. It takes 'BIND AUTH' or 'BIND CACHE' or 'UNBOUND' or 'NSD' or 'DNS'

PROXY' 'DHCP' or 'IPAM' as a input. [mandatory]

--all

It accepts '1' or '0' value. '1' indicates all the 'appliances' or 'appliance groups' are eligible to list policy compliance.'0' indicates particular 'appliance' or 'appliance group' is eligible to list policy compliance.

--policy_name

Policy name of an appliance. To set the baseline for particular appliance.

--threshold_name

Threshold name of an appliance. To set the baseline for particular appliance.

EXAMPLE:

```
twc baselinepolicycompliance --ip=192.168.15.56 --type="IPAM" --all=1
twc baselinepolicycompliance --ip=10.1.15.25 --type="IPAM" --all=0 --
policy_name=TCPWave-Application-Check-Policy --threshold_name="System Users With Restricted
Shells"
twc baselinepolicycompliance --ip=192.165.25.12 --type="BIND AUTH" --all=0 --
policy_name=TCPWave-Application-Check-Policy --threshold_name="Zone Count"
twc baselinepolicycompliance --ip=172.56.10.25 --type="DHCP" --all=0 --
policy_name=TCPWave-Application-Check-Policy --threshold_name="Zone Count"
```

changeschedjobtype**NAME:****changeschedjobtype**

changeschedjobtype - Change the type of the scheduled job in the TCPWave IPAM.

DESCRIPTION:

Change the type of the scheduled job in the TCPWave IPAM.

ARGUMENTS:**--job_id**

Id of the scheduled job. [mandatory]

--job_type

Type of the scheduled job. It takes 'admin' or 'normal' as input. [mandatory]

EXAMPLE:

twc changeschedjobtype --job_id=RemoteMonitStatsOperation --job_type=admin

enablefadmaccess**NAME**

enablefadmaccess

DESCRIPTION

It enables and disables the FADM user access in the TCPWave IPAM. UADM users can only enable or disable FADM access. The system will check the user type before giving access please make sure that this command only runs by UADM user.

ARGUMENTS**--option**

Option to enable the FADM access. It takes as 'Yes' or 'No' [mandatory]

EXAMPLE

twc enablefadmaccess --option=Yes

fetchnamedlog**NAME****fetchnamedlog****DESCRIPTION**

Fetch the DNS server named logs from the TCPWave IPAM. DNS server should be 'BIND AUTH','BIND CACHE' or 'DNS PROXY'.

ARGUMENTS**--ip**

IP Address of the DNS server. [mandatory]

--start_date

Start date for logs generation. Date format is mm/dd/yyyy. [mandatory]

--end_date

End date for logs generation. Date format is mm/dd/yyyy. [mandatory]

--output_file

Full path to the output file to which named logs are to be written. If the file path is not specified, the output is written to the standard output.

--count

Number of lines to be fetched from the named log. If this argument is not specified, 100 lines will be fetched.

EXAMPLE:

```
twc fetchnamedlog --ip=10.1.10.190 --output_file=/tmp/namedlog.txt --count=200 --  
start_date=08/12/2021 --end_date=08/15/2021
```

login**NAME****login****DESCRIPTION**

Login into TCPWave IPAM to execute commands on CLI. The command shall prompt to enter the password. The password typed will not be echoed on to the screen.

ARGUMENTS

--u

User name. [mandatory]

--auto

Auto login flag. It takes only '1'. If it is specified, user will not be asked to enter the password. To enable the auto login for a user use the 'autologin' CLI.

EXAMPLE

twc login --u=jsmith

twc login --u=jsmith --auto=1

logout

NAME

logout

DESCRIPTION

Logout from the TCPWave IPAM to execute commands on CLI.

EXAMPLE

twc logout

mergesubnet**NAME:**

mergesubnet

DESCRIPTION:

Merge the subnets in the TCPWave IPAM.

ARGUMENTS:**--first_subnet**IP address of the first subnet in the range of subnets to be merged
[mandatory]**--last_subnet**IP address of the last subnet in the range of subnets to be merged
[mandatory]**--mask**

Desired mask length of the subnet resulting from the merge [mandatory]

--network

Start Address of the associated network [mandatory]

--name

Name of the subnet resulting from the merge.

--subnet_group

Name of the associated subnet group.

--domain

Domain name to be associated with the subnet. [mandatory]

--router_addr

IP Address of the router associated with the subnet resulting from the merge. [mandatory]

--desc

Description text for the subnet.

--street1

Street1 part of the location information.

--street2

Street2 part of the location information.

--city

City part of the location information.

--state

State part of the location information.

--zip

Zip code part of the location information.

--dhcp_tmpl

Template name specifying the dhcp options for the subnet.

--dhcp_appliance

Primary DHCP server for the subnet.

--org

Name of the organization to which specified, network belongs. This argument is mandatory if user is 'FADM'.

EXAMPLE USAGE:

```
tvc mergesubnet --first_subnet=15.1.10.0 --last_subnet=15.1.10.128 --name="subnet-test" --mask=24 --network=15.1.10.0 --domain=123.com --dhcp_tmpl="DHCP_Option_Template" --dhcp_appliance="nusalx-any-sl0983-189" --router_addr=15.1.10.1 --org=TCPWave
```

```
tvc mergesubnet --first_subnet=13.0.160.0 --last_subnet=13.0.168.0 --name="subnet-test" --mask=20 --network=13.0.0.0 --domain=tcpwave.com --dhcp_tmpl="Generic Template" --dhcp_appliance="nusalx-trv10-sl0984" --router_addr=13.0.160.1 --org=TCPWave
```

```
tvc mergesubnet --first_subnet=11.12.0.0 --last_subnet=11.12.0.192 --name="subnet-test" --mask=24 --network=11.12.0.0 --domain=tcpwave123.com --router_addr=11.12.0.1 --subnet_group=SG-1 --street1="600 ALEXANDER ROAD" --city="PRINCETON" --state=NJ --country=USA --zip=08540 --org=TCPWave
```

moveobject

NAME:

moveobject

DESCRIPTION:

Moves object in TCPWave IPAM.

Move the object from source IP address to the specified, destination IP address, destination IP address should be empty.

ARGUMENTS:

--current_addr

Current IP Address of the object. [mandatory]

--destination_addr

Destination IP Address of the object. [mandatory]

--option

Takes input as '1' or '2'.

'1' = Move to another object with different IP Address.

'2' = Move as an alias to another object.

--org

Name of the organization. [mandatory]

```
--current_name
    Current name of the object.

--current_domain
    Current domain name of the object.

--destination_name
    Destination name of the object.

--destination_domain
    Destination domain name of the object.
```

EXAMPLE USAGE:

```
twc moveobject --current_addr=10.1.1.5 --destination_addr=10.1.1.9 --option=1 --org=TCPWave

twc moveobject --current_name=TestObj --current_domain=tcp.com --
destination_name=SecondObj --destination_domain=test.com --option=2 --org=TCPWave
```

patchdeploymentinfo**NAME:**

patchdeploymentinfo

DESCRIPTION:

Information of patch deployment in the TCPWave IPAM.

ARGUMENTS:

```
--package_name
    Name of the package. [mandatory]

--major_version
    Major version of the TIMS. [mandatory]

--minor_version
    Minor version of the TIMS. [mandatory]

--patch_name
    Name of the patch. [mandatory]

--appliance_type
    Type of the appliance. [mandatory]

--patch_level
    Level of the patch. [mandatory]
```

EXAMPLE USAGE:

```
twc patchdeploymentinfo --package_name=TCPWaveIPAM --major_version=11 --
minor_version=27 --patch_name="Test patch" --appliance_type=IPAM --patch_level=2
```

purgeipamenti

NAME

`purgeipamenti`

DESCRIPTION

Purge the entity from the recycle bin in the TCPWave IPAM.

ARGUMENTS

`--entity_type`

Type of the entity, specifies entity type 'object', 'zone', 'scope', 'revzone' and subnet.

This field is not needed when flag --all mention as '1'.

`--entity_name`

Name of the entity, specifies IP address when entity_type is object, subnet and scope.

When entity_type is subnet specify IP address with mask length.

This field is not needed when flag --all mention as '1'.

`--operation`

Name of the operation, takes add, delete or edit.

This field is not needed when flag --all mention as '1'.

`--org`

Name of the organization. This argument is mandatory if the --all is not mentioned.

`--all`

Flag to purge all the entities from the recycle bin. This argument takes '1' or '0'. If this argument is specified as '1' all entities will be purged. Default value of this argument is '0'.

EXAMPLE

```
twc purgeipamenti --entity_name=10.0.3.16/28 --entity_type=subnet --operation=delete --org=TCPWave
```

```
twc purgeipamenti --all=1
```

promoteipamtomaster**NAME:**

`promoteipamtomaster`

DESCRIPTION:

Promotes an IPAM to master in the high availability cluster.

ARGUMENTS:

`--ip`

IP address of the IPAM. [mandatory]

EXAMPLE USAGE:

```
twc promoteipamtomaster --ip=172.16.0.172
```

rebuildsearch**NAME:**

rebuildsearch

DESCRIPTION:

Rebuilds the search indexes and restarts the replicator service in the TCPWave IPAM.

ARGUMENTS:

- NA

EXAMPLE USAGE:

```
twc rebuildsearch
```

redistributeremotes**NAME:**

redistributeremotes

DESCRIPTION:

Redistributes the remote appliances in the high availability cluster.

ARGUMENTS:

--ip

IP address of the remotes. It accepts the comma separated IP addresses If this argument is not specified it will redistribute all remotes.

EXAMPLE USAGE:

```
twc redistributeremotes --ip=10.0.0.201,10.0.0.202
```

resetdr**NAME:**

resetdr

DESCRIPTION:

Reset the disaster recovery configuration in the TCPWave IPAM.

ARGUMENTS:

- flag
Force flag to reset disaster recovery.

EXAMPLE USAGE:

```
twc resetdr --flag=1
```

resethaconfig**NAME:**

resethaconfig

DESCRIPTION:

Resets the high availability cluster configuration in the TCPWave IPAM.

ARGUMENTS:**EXAMPLE USAGE:**

twc resethaconfig

restoredb

NAME:

restoredb

DESCRIPTION:

Restores database from a given database snapshot.

ARGUMENTS:

--dump_dir [mandatory]

Dump directory from which snapshot changes are to be applied.

--apply_binlog

Takes '0' or '1'. '1' indicates that the incremental changes after the snapshot was taken need to be applied. '0' indicates that the incremental changes need not be applied. If this argument is not specified, the value defaults to '0'.

EXAMPLE USAGE:

twc restoredb --dump_dir=/tmp/dumps/Dump_1408948935 --apply_binlog=0

twc restoredb --dump_dir=/tmp/dumps/Dump_1408948935 --apply_binlog=1

rndcflush

NAME:

rndcflush

DESCRIPTION:

Performs rndc flush operation on a BIND CACHE DNS server in the TCPWave IPAM.

If only IP address option is specified, then 'rndc flush' command will be executed in the BIND CACHE DNS server.

If only IP address and name options are specified, 'rndc flushname <name>' command will be executed in the BIND CACHE DNS server.

If tree is specified, as '1' along with IP address and name options 'rndc flushtree <name>' command will be executed in the BIND CACHE DNS server.

ARGUMENTS:

--ip

IP address of the BIND CACHE DNS server. [mandatory]

--name

Name that must be flushed. It takes fully qualified domain name.

--tree

Flag to flush specified, name as well as all records below that name.
It takes either '1' or '0'. If It is specified, as '1' specified, name
will be flushed as well as all records below that name.

EXAMPLE USAGE:

```
twc rndcflush --ip=10.1.10.180
```

```
twc rndcflush --ip=10.1.10.180 --name=www.tcpwave.com.
```

```
twc rndcflush --ip=10.1.10.180 --name=tcpwave.com. --tree=1
```

rollbackpatch**NAME:**

rollbackpatch

DESCRIPTION:

Roll back a patch from the TCPWave IPAM.

Deployed patch can only be roll back from the TCPWave IPAM.

ARGUMENTS:**--file_name**

Name of the zip file. [mandatory]

--package_name

Name of the package. [mandatory]

--major_version

Major version of the TIMS. [mandatory]

--minor_version

Minor version of the TIMS. [mandatory]

--patch_name

Name of the patch. [mandatory]

--appliance_type

Type of the appliance. [mandatory]

--patch_level

Level of the patch. [mandatory]

--schd_time

Specifies the schedule time, time format should be "yyyy-MM-dd HH:mm:SS".

EXAMPLE USAGE:

```
twc rollbackpatch --file_name=Patch_IPAM_11.27_2_6301.zip --package_name=TCPWaveIPAM -
```

```
-major_version=11 --minor_version=27 --patch_name="Test patch" --appliance_type=IPAM --
patch_level=2
```

```
twc rollbackpatch --file_name=Patch_IPAM_11.27_2_6301.zip --package_name=TCPWaveIPAM -
-major_version=11 --minor_version=27 --patch_name="Test patch" --appliance_type=IPAM --
patch_level=2 --schd_time="2019-05-08 03:12:00"
```

chkipexistence

DESCRIPTION:

Checks whether IP address is used in the given organization in the TCPWave IPAM.

ARGUMENTS:

--ip

Specify IP address which to check is used in the organization. [mandatory]

--org

Name of the organization. [mandatory]

EXAMPLE:

```
twc chkipexistence --ip=10.0.0.12 --org=TcpWave
```

chkobjnameexistence

DESCRIPTON:

Lists all the objects with the specified name along with CNAME records of the objects in the given organization from the TCPWave IPAM

ARGUMENTS:

--name

Name of the object. [mandatory]

--org

Name of the organization. [mandatory]

EXAMPLE:

```
twc chkobjnameexistence --name=3G00001Phone --org=TcpWave
```

rptadminaudit

NAME:

rptadminaudit

DESCRIPTION:

Generates an admin audit report to a specified, file in the TCPWave IPAM.

ARGUMENTS:**--name**

Name or Login ID of the admin. This argument is not required if --role argument is specified.,

--role

Role of the admin. This argument is not required if --name argument is specified,. It takes 'FADM','NAMD','PADM','RADM','SADM' or 'UADM'.

--from_date

Start date for report generation.Date format is mm/dd/yyyy [mandatory]

--to_date

End date for report generation.Date format is mm/dd/yyyy [mandatory]

--action

filter for admin actions. Takes 'add','alias','config','converge', 'create','deactivate','delete','deploy','edit','execute','export', 'generate','get','import','list','login','merge','modify', 'PackageDeploy','PackageRollback','PatchDeploy','PatchRollback', 'rollback','Schedule Create','schedule-deploy','schedule-rollback', 'sched_delete','sched_modify','Schedule Report Email','split','update', 'upload','validate','view',' or 'Zone Force Sync'.

--output_file

Full path to the output file to which audit report is to be written. [mandatory]

--status

It takes 'active' or 'deleted'. It is mandatory if --name argument is specified.,

--format

Format of the output file. It takes 'csv' or 'pdf'. [mandatory]

EXAMPLE USAGE:

```
twc rptadminaudit --name=Jhon --status=active --from_date=07/01/2015 --to_date=09/01/2015 --
output_file=/tmp/output.csv --format=csv
```

```
twc rptadminaudit --role=SADM --from_date=07/01/2015 --to_date=09/01/2015 --
output_file=/tmp/output.pdf --action=login --format=pdf
```

rptdhcpactivelease**NAME:**
rptdhcpactivelease
DESCRIPTION:

Generates a DHCP server active lease report to a specified, file the TCPWave

IPAM.

ARGUMENTS:

--dhcp_appliance

Name or Address of the DHCP server. [mandatory]

--output_file

Full path to the output file to which audit report is to be written.
[mandatory]

--format

Format of the output file. It takes 'csv' or 'pdf'. [mandatory]

EXAMPLE USAGE:

```
twc rptdhcpactivelease --dhcp_appliance=192.168.1.235 --output_file=/tmp/output.csv --format=csv
```

```
twc rptdhcpactivelease --dhcp_appliance=192.168.1.235 --output_file=/tmp/output.pdf --format=pdf
```

rptdhcpdnssvrevents**NAME:**

rptdhcpdnssvrevents

DESCRIPTION:

Generates a DHCP/DNS server event report to a specified, file in the TCPWave IPAM.

ARGUMENTS:

--appliance

Address or Name of the server. This argument is not required if
--last_100_event argument is specified.,

--last_100_event

Last 100 event flag. This takes only '1'.

--from_date

Start date for report generation. Date format is mm/dd/yyyy. This argument is not required if --last_100_event argument is specified.,

--to_date

End date for report generation.Date format is mm/dd/yyyy. This argument is not required if --last_100_event argument is specified.,

--output_file

full path to the output file to which audit report is to be written.
[mandatory]

--format

Format of the output file. It takes 'csv' or 'pdf'. [mandatory]

EXAMPLE USAGE:

```
twc rptdhcpdnssvrevents --appliance=tcpwave-dhcp-server-31 --from_date=07/01/2015 --to_date=09/01/2015 --output_file=/tmp/output.csv --format=csv
```

```
twc rptdhcpdnssvrevents --appliance=tcpwave-dhcp-server-31 --from_date=07/01/2015 --to_date=09/01/2015 --output_file=/tmp/output.pdf --format=pdf
```

```
twc rptdhcpdnssvrevents --last_100_event=1 --output_file=/tmp/output.csv --format=csv
```

```
twc rptdhcpdnssvrevents --last_100_event=1 --output_file=/tmp/output.pdf --format=pdf  
rptdhcpopttmplaudit
```

NAME:

`rptdhcpopttmplaudit`

DESCRIPTION:

Generates a DHCP option template audit report to a specified, file in the TCPWave IPAM.

ARGUMENTS:

`--opt_tmpl`

Name of the DHCP option template. [mandatory]

`--from_date`

Start date for report generation.Date format is mm/dd/yyyy [mandatory]

`--to_date`

End date for report generation.Date format is mm/dd/yyyy [mandatory]

`--output_file`

Full path to the output file to which audit report is to be written.
[mandatory]

`--status`

It takes 'active' or 'deleted'. [mandatory]

`--format`

Format of the output file. It takes 'csv' or 'pdf'. [mandatory]

EXAMPLE USAGE:

```
twc rptdhcpopttmplaudit --opt_tmpl=option-template --status=active --from_date=06/01/2015 --to_date=08/01/2015 --output_file=/tmp/output.csv --format=csv
```

```
twc rptdhcpopttmplaudit --opt_tmpl=option-template --status=deleted --from_date=06/01/2015 --to_date=08/01/2015 --output_file=/tmp/output.pdf --format=pdf
```

rptdhcpolicytmplaudit**NAME:**

rptdhcpolicytmplaudit

DESCRIPTION:

Generates a DHCP policy template audit report to a specified, file in the TCPWave IPAM.

ARGUMENTS:

--dhcp_policy_tmpl

Name of the DHCP policy template [mandatory]

--from_date

Start date for report generation.Date format is mm/dd/yyyy [mandatory]

--to_date

End date for report generation.Date format is mm/dd/yyyy [mandatory]

--output_file

Full path to the output file to which audit report is to be written
[mandatory]

--status

It takes 'active' or 'deleted' [mandatory]

--format

Format of the output file. It takes 'csv' or 'pdf'. [mandatory]

EXAMPLE USAGE:

```
twc rptdhcpolicytmplaudit --dhcp_policy_tmpl=Policy-template --status=active --  
from_date=06/01/2015 --to_date=08/01/2015 --output_file=/tmp/output.csv --format=csv
```

```
twc rptdhcpolicytmplaudit --dhcp_policy_tmpl=Policy-template --status=active --  
from_date=06/01/2015 --to_date=08/01/2015 --output_file=/tmp/output.pdf --format=pdf
```

rptdhcpsvraudit**NAME:**

rptdhcpsvraudit

DESCRIPTION:

Generates a DHCP server audit report to a specified, file in the TCPWave IPAM.

ARGUMENTS:

--dhcp_appliance

Name or Address of the DHCP server [mandatory]

--from_date
Start date for report generation.Date format is mm/dd/yyyy [mandatory]

--to_date
End date for report generation.Date format is mm/dd/yyyy [mandatory]

--status
It takes 'active' or 'deleted'. [mandatory]

--output_file
Full path to the output file to which audit report is to be written [mandatory]

--format
Format of the output file. It takes 'csv' or 'pdf'. [mandatory]

EXAMPLE USAGE:

twc rptdhcpsvraudit --dhcp_appliance=dhcp-server-1 --from_date=07/01/2015 --to_date=09/01/2015 --output_file=/tmp/output.pdf --format=pdf --status=active

twc rptdhcpsvraudit --dhcp_appliance=dhcp-server-1 --from_date=07/01/2015 --to_date=09/01/2015 --output_file=/tmp/output.csv --format=csv --status=active

rptdhcpsvrmanagedsnet**NAME:**

rptdhcpsvrmanagedsnet

DESCRIPTION:

Generates a DHCP server managed subnet audit report to a specified, file in the TCPWave IPAM.

ARGUMENTS:

--dhcp_appliance
Address or Name of the server. [mandatory]

--status
Takes 'active' or 'deleted'. [mandatory]

--output_file
Full path to the output file to which audit report is to be written. [mandatory]

--format
Format of the output file. It takes 'csv' or 'pdf'. [mandatory]

EXAMPLE USAGE:

twc rptdhcpsvrmanagedsnet --dhcp_appliance=10.1.10.31 --status=active --output_file=/tmp/output.csv --format=csv

twc rptdhcpsvrmanagedsnet --dhcp_appliance=10.1.10.31 --status=active --

```
output_file=/tmp/output.pdf --format=pdf
```

rptdhcpopt**NAME:****rptdhcpopt****DESCRIPTION:**

Generates a DHCP option report in a pdf file in the TCPWave IPAM.

ARGUMENTS:**--dhcp_opt**

Name of the DHCP option. [mandatory]

--param_value

Parameter value of the specified, DHCP option.

--output_file

Full path to the output file to which audit report is to be written.
[mandatory]

EXAMPLE USAGE:

```
twc rptdhcpopt --dhcp_opt="Domain Name" --param_value=tcpwave.com --  
output_file=/tmp/output.pdf
```

rptdhcptmplassociation**NAME:**

rptdhcptmplassociation - Generates a DHCP option template association report.in specified files in the TCPWave IPAM.

DESCRIPTION:

Generates a DHCP option template association report in specified files in.the TCPWave IPAM.

ARGUMENTS:**--opt_tmpl**

Name of the DHCP option template. [mandatory]

--org

Name of the organization. [mandatory]

--subnet_output_file

Full path to the output file to which DHCP option template subnet.association report is to be written. [mandatory]

--scope_output_file

Full path to the output file to which DHCP option template scope.association report is to be written.
[mandatory]

--object_output_file

Full path to the output file to which DHCP option template object.association report is to be written.
[mandatory]

--format

Format of the output files. It takes 'csv' or 'pdf'. [mandatory]

EXAMPLE:

```
twc rptdhcptmplassociation --opt_tmpl=Generic --org=TcpWave -subnet_output_file=/tmp/output1.csv  
--scope_output_file=/tmp/output2.csv --object_output_file=/tmp/output3.csv --format=csv
```

```
twc rptdhcptmplassociation --opt_tmpl=Generic --org=TcpWave --subnet_output_file=/tmp/output1.pdf  
--scope_output_file=/tmp/output2.pdf --object_output_file=/tmp/output3.pdf --format=pdf
```

rptdhcptoscopecount

NAME:

rptdhcptoscopecount

DESCRIPTION:

Generates a DHCP Server to scope count report to a specified, file in the TCPWave IPAM.

ARGUMENTS:

--output_file

Full path to the output file to which audit report is to be written.
[mandatory]

--format

Format of the output file. It takes 'csv' or 'pdf'. [mandatory]

EXAMPLE USAGE:

```
twc rptdhcptoscopecount --output_file=/tmp/output.csv --format=csv
```

```
twc rptdhcptoscopecount --output_file=/tmp/output.pdf --format=pdf
```

rptdnsopptmplaudit

NAME:

rptdnsopptmplaudit

DESCRIPTION:

Generates a DNS option template audit report to a specified, file in the TCPWave IPAM.

ARGUMENTS:

--opt_tmpl

Name of the DNS option template. [mandatory]

```
--from_date
    Start date for report generation.Date format is mm/dd/yyyy [mandatory]

--to_date
    End date for report generation.Date format is mm/dd/yyyy [mandatory]

--output_file
    Full path to the output file to which audit report is to be written.
    [mandatory]

--status
    It takes 'active' or 'deleted'. [mandatory]

--format
    Format of the output file. It takes 'csv' or 'pdf'. [mandatory]
```

EXAMPLE USAGE:

```
twc rptdnssopttplaudit --opt_tmpl ="TCPWave DNS Auth Option Template(BIND AUTH)" --
status=active --from_date=07/01/2015 --to_date=09/01/2015 --output_file=/tmp/output.csv --
format=csv
```

```
twc rptdnssopttplaudit --opt_tmpl ="TCPWave DNS Auth Option Template(BIND AUTH)" --
status=active --from_date=07/01/2015 --to_date=09/01/2015 --output_file=/tmp/output.pdf --
format=pdf
```

rptdnssoa**NAME:**

rptdnssoa

DESCRIPTION:

Generates a DNS SOA report to a specified, file in the TCPWave IPAM.

ARGUMENTS:

```
--dns_auth_server
    Name or Address of the DNS Authoritative Server. [mandatory]

--output_file
    Full path to the output file to which audit report is to be written.
    [mandatory]

--format
    Format of the output file. It takes 'csv' or 'pdf'. [mandatory]
```

EXAMPLE USAGE:

```
twc rptdnssoa --dns_auth_server=DNS-Auth-Server-1 --output_file=/tmp/output.pdf --
format=pdf
```

```
twc rptdnssoa --dns_auth_server=DNS-Auth-Server-1 --output_file=/tmp/output.csv --
format=csv
```

rptdnssvraudit**NAME:**

rptdnssvraudit

DESCRIPTION:

Generates a DNS server audit report to a specified, file in the TCPWave IPAM.

ARGUMENTS:

--dns_server

Name or Address of the DNS server. [mandatory]

--from_date

Start date for report generation.Date format is mm/dd/yyyy [mandatory]

--to_date

End date for report generation.Date format is mm/dd/yyyy [mandatory]

--output_file

Full path to the output file to which audit report is to be written.
[mandatory]

--status

It takes 'active' or 'deleted'. [mandatory]

--format

Format of the output file. It takes 'csv' or 'pdf'. [mandatory]

EXAMPLE USAGE:

```
twc rptdnssvraudit --dns_server=10.1.10.102 --status=active --from_date=07/01/2015 --  
to_date=09/01/2015 --output_file=/tmp/output.csv --format=csv
```

```
twc rptdnssvraudit --dns_server=10.1.10.102 --status=active --from_date=07/01/2015 --  
to_date=09/01/2015 --output_file=/tmp/output.pdf --format=pdf
```

rptdnssvrtmplaudit**NAME:**

rptdnssvrtmplaudit

DESCRIPTION:

Generates a DNS server template audit report to a specified, file in the
TCPWave IPAM.

ARGUMENTS:

--dns_server_tmpl

Name of the DNS server template. [mandatory]

--from_date

Start date for report generation.Date format is mm/dd/yyyy [mandatory]

--to_date
End date for report generation.Date format is mm/dd/yyyy [mandatory]

--output_file
Full path to the output file to which audit report is to be written.
[mandatory]

--status
It takes 'active' or 'deleted'. [mandatory]

--format
Format of the output file. It takes 'csv' or 'pdf'. [mandatory]

EXAMPLE USAGE:

```
twc rptdnssvrtmplaudit --dns_server_tmpl=DNS-Server-Template --status=active --  
from_date=06/01/2015 --to_date=08/01/2015 --output_file=/tmp/output.csv --format=csv
```

```
twc rptdnssvrtmplaudit --dns_server_tmpl=DNS-Server-Template --status=active --  
from_date=06/01/2015 --to_date=08/01/2015 --output_file=/tmp/output.pdf --format=pdf  
rptglobalallocbyobjtype
```

NAME:

rptglobalallocbyobjtype

DESCRIPTION:

Generates a global allocation report by object type to a specified, file in
the TCPWave IPAM.

ARGUMENTS:

--object_type
Type of the object. [mandatory]

--from_date
Start date for report generation. Date format is mm/dd/yyyy. [mandatory]

--to_date
End date for report generation.Date format is mm/dd/yyyy. [mandatory]

--output_file
Full path to the output file to which audit report is to be written.
[mandatory]

--format
Format of the output file. It takes 'csv' or 'pdf'. [mandatory]

EXAMPLE USAGE:

```
twc rptgloballocbyobjtype --object_type="3G Phone" --from_date=07/01/2015 --  
to_date=09/01/2015 --output_file=/tmp/output.csv --format=csv
```

```
twc rptgloballocbyobjtype --object_type="PC" --from_date=07/01/2015 --to_date=09/01/2015 --  
output_file=/tmp/output.pdf --format=pdf
```

rptipameventaudit**NAME:**

rptipameventaudit

DESCRIPTION:

Generates an IPAM events audit report to a specified, file in the TCPWave IPAM.

ARGUMENTS:

```
--from_date  
      Start date for report generation.Date format is mm/dd/yyyy [mandatory]  
  
--to_date  
      End date for report generation.Date format is mm/dd/yyyy [mandatory]  
  
--output_file  
      Full path to the output file to which audit report is to be written.  
      [mandatory]  
  
--format  
      Format of the output files. It takes 'csv' or 'pdf'. [mandatory]
```

EXAMPLE USAGE:

```
twc rptipameventaudit --from_date=11/20/2015 --to_date=12/01/2015 --  
output_file=/tmp/output.csv --format=csv
```

```
twc rptipameventaudit --from_date=11/20/2015 --to_date=12/01/2015 --  
output_file=/tmp/output.pdf --format=pdf
```

rptloginactivity**NAME:**

rptloginactivity

DESCRIPTION:

Generates a login activity audit report to a specified, file in the TCPWave IPAM.

ARGUMENTS:

```
--from_date
    Start date for report generation.Date format is mm/dd/yyyy [mandatory]

--to_date
    End date for report generation.Date format is mm/dd/yyyy [mandatory]

--output_file
    Full path to the output file to which audit report is to be written.
    [mandatory]
```

EXAMPLE USAGE:

```
twc rptloginactivity --from_date=12/01/2015 --to_date=01/01/2016 --output_file=/tmp/output.csv  
rptmonitoringalerts
```

NAME:

rptmonitoringalerts

DESCRIPTION:

Generates a monitoring alert report to a specified, file in the TCPWave IPAM.

ARGUMENTS:

```
--host
    Name or Address of the host. [mandatory]

--output_file
    Full path to the output file to which audit report is to be written.
    [mandatory]

--daily
    It takes only 1. If this argument is specified, daily monitoring report
    will be generated.

--weekly
    It takes only 1. If this argument is specified, weekly monitoring report
    will be generated.

--monthly
    It takes only 1.If this argument is specified, monthly monitoring report
    will be generated.

--format
    Format of the output file. It takes 'csv' or 'pdf'. [mandatory]
```

EXAMPLE USAGE:

```
twc rptmonitoringalerts --host=dhcp-server-31.tcpwave.com --daily=1 --
output_file=/tmp/output.csv --format=csv
```

```
twc rptmonitoringalerts --host=10.1.10.31 --weekly=1 --output_file=/tmp/output.pdf --
```

format=pdf

```
twc rptmonitoringalerts --host=dhcp-server-31.tcpwave.com --monthly=1 --
output_file=/tmp/output.csv --format=csv
```

```
twc rptmonitoringalerts --host=10.1.10.31 --monthly=1 --output_file=/tmp/output.pdf --
format=pdf
```

rptnetaudit

NAME:

rptnetaudit

DESCRIPTION:

Generates a network audit report to a specified, file in the TCPWave IPAM.

ARGUMENTS:

--ip

Address of the network along with mask separated by '/'. This argument is not required if --all argument is specified.,

--all

It takes only '1'.

--org

Name of the organization in which specified, network(s) exist. This argument is mandatory if user is FADM.

--from_date

Start date for report generation.Date format is mm/dd/yyyy [mandatory]

--to_date

End date for report generation.Date format is mm/dd/yyyy [mandatory]

--output_file

Full path to the output file to which audit report is to be written.
[mandatory]

--format

Format of the output file. It takes 'csv' or 'pdf'. [mandatory]

--status

It takes 'active' or 'deleted'. [mandatory]

EXAMPLE USAGE:

```
twc rptnetaudit --ip=192.168.1.0/24 --status=active --org=TCPWave --from_date=07/01/2015 --
to_date=09/01/2015 --output_file=/tmp/output.csv --format=csv
```

```
twc rptnetaudit --ip=192.168.1.0/24 --status=active --org=TCPWave --from_date=07/01/2015 --
to_date=09/01/2015 --output_file=/tmp/output.pdf --format=pdf
```

```
twc rptnetaudit --all=1 --status=active --org=TCPWave --from_date=07/01/2015 --
to_date=09/01/2015 --output_file=/tmp/output.csv --format=csv
```

```
twc rptnetaudit --all=1 --status=active --org=TCPWave --from_date=07/01/2015 --
to_date=09/01/2015 --output_file=/tmp/output.pdf --format=pdf
```

rptobjaudit**NAME:**

rptobjaudit

DESCRIPTION:

Generates an object audit report to a specified, file in the TCPWave IPAM.

ARGUMENTS:

--ip

Address of the object. This argument is not required if --name argument is specified.,

--name

Name of the object. This argument is not required if --ip argument is specified.,

--org

Name of the organization in which specified, object exist. This argument is mandatory if user is FADM.

--from_date

Start date for report generation.Date format is mm/dd/yyyy [mandatory]

--to_date

End date for report generation.Date format is mm/dd/yyyy [mandatory]

--output_file

Full path to the output file to which audit report is to be written.
[mandatory]

--format

Format of the output file. It takes 'csv' or 'pdf'. [mandatory]

EXAMPLE USAGE:

```
twc rptobjaudit --ip=192.168.1.9 --org=TCPWave --from_date=07/01/2015 --to_date=09/01/2015 --
output_file=/tmp/output.csv --format=csv
```

```
twc rptobjaudit --name=tcpwave-DHCPServer-31 --org=TCPWave --from_date=07/01/2015 --
to_date=09/01/2015 --output_file=/tmp/output.pdf --format=pdf
```

rptrraudit**NAME:**

rptrraudit

DESCRIPTION:

Generates a resource record audit report to a specified, file in the TCPWave IPAM.

ARGUMENTS:

--org
Name of the organization. [mandatory]

--domain
Name of the domain. [mandatory]

--rr_short_name
Resource record short name. [mandatory]

--from_date
Start date for report generation.Date format is mm/dd/yyyy [mandatory]

--to_date
End date for report generation.Date format is mm/dd/yyyy [mandatory]

--output_file
Full path to the output file to which audit report is to be written.
[mandatory]

--format
Format of the output file. It takes 'csv' or 'pdf'. [mandatory]

EXAMPLE USAGE:

```
twc rptraudit --org=Internal --domain=tcpwave.com --rr_short_name=www --  
from_date=07/01/2015 --to_date=09/01/2015 --output_file=/tmp/output.pdf --format=pdf
```

```
twc rptraudit --org=Internal --domain=tcpwave.com --rr_short_name=www --  
from_date=07/01/2015 --to_date=09/01/2015 --output_file=/tmp/output.csv --format=csv  
rptschedeventsaudit
```

NAME:

rptschedeventsaudit

DESCRIPTION:

Generates a scheduled events audit report to a specified, file in the TCPWave IPAM.

ARGUMENTS:

--admin
Name or Login Id of the admin. [mandatory]

--from_date
Start date for report generation.Date format is mm/dd/yyyy [mandatory]

--to_date
End date for report generation.Date format is mm/dd/yyyy [mandatory]

--output_file
Full path to the output file to which audit report is to be written.
[mandatory]

--format
Format of the output file. It takes 'csv' or 'pdf'. [mandatory]

--status
It takes 'active' or 'deleted'. [mandatory]

--admin_job
It takes '1' or '0'. '1' indicates report should contain admin jobs.
'0' indicates report should not contain admin jobs.

--ipam
It takes '1' or '0'. '1' indicates report should contain IPAM jobs.
'0' indicates report should not contain IPAM jobs.

--patch
It takes '1' or '0'. '1' indicates report should contain patch jobs.
'0' indicates report should not contain patch jobs.

EXAMPLE USAGE:

twc rptschedeventsaudit --admin="John Smith" --status=active --from_date=07/01/2015 --to_date=09/01/2015 --output_file=/tmp/output.csv --format=csv --admin_job=1

twc rptschedeventsaudit --admin="John Smith" --status=active --from_date=07/01/2015 --to_date=09/01/2015 --output_file=/tmp/output.pdf --format=pdf --admin_job=1

rptsnetaudit**NAME:**

rptsnetaudit

DESCRIPTION:

Generates a subnet audit report to a specified, file in the TCPWave IPAM.

ARGUMENTS:

--ip
Address of the subnet along with mask separated by '/'. This argument is not required if --all argument is specified.,

--all
It takes only '1'.

--org
Name of the organization in which specified, network(s) exist. This

argument is mandatory if user is FADM.

--from_date

Start date for report generation.Date format is mm/dd/yyyy [mandatory]

--to_date

End date for report generation.Date format is mm/dd/yyyy [mandatory]

--output_file

Full path to the output file to which audit report is to be written.
[mandatory]

--format

Format of the output file. It takes 'csv' or 'pdf'. [mandatory]

EXAMPLE USAGE:

```
twc rptsnetaudit --ip=192.168.1.0/24 --org=TCPWave --from_date=06/01/2015 --  
to_date=08/01/2015 --output_file=/tmp/output.pdf --format=pdf
```

```
twc rptsnetaudit --all=1 --org=TCPWave --from_date=06/01/2015 --to_date=08/01/2015 --  
output_file=/tmp/output.pdf --format=pdf
```

rptsubnetauditbygroup

NAME:

rptsubnetauditbygroup

DESCRIPTION:

Generates a subnet audit report by group to a specified, file in the TCPWave IPAM.

ARGUMENTS:

--subnet_group

Name of the subnet group. This argument is not required if --all is specified, as '1'.

--all

It takes '1' or '0'. If it is specified, as '1' it will generate subnet reports for all groups.

--from_date

Start date for report generation.Date format is mm/dd/yyyy [mandatory]

--to_date

End date for report generation.Date format is mm/dd/yyyy [mandatory]

--output_file

Full path to the output file to which audit report is to be written.

[mandatory]

--format

Format of the output files. It takes 'csv' or 'pdf'. [mandatory]

EXAMPLE USAGE:

```
twc rptsubnetauditbygroup --subnet_group=tcpwave-subnet-group --from_date=11/01/2015 --to_date=12/01/2015 --output_file=/tmp/output.csv --format=csv
```

```
twc rptsubnetauditbygroup --all=1 --from_date=11/01/2015 --to_date=12/01/2015 --output_file=/tmp/output.pdf --format=pdf
```

rptsubnetlistbygroup

NAME:

rptsubnetlistbygroup

DESCRIPTION:

Generates a subnet list report of a subnet group to a specified, file in the TCPWave IPAM.

ARGUMENTS:

--subnet_group

Name of the subnet group. [mandatory]

--output_file

Full path to the output file to which audit report is to be written. [mandatory]

--format

Format of the output files. It takes 'csv' or 'pdf'. [mandatory]

EXAMPLE USAGE:

```
twc rptsubnetlistbygroup --subnet_group=tcpwave-subnet-group --output_file=/tmp/output.csv --format=csv
```

```
twc rptsubnetlistbygroup --subnet_group=tcpwave-subnet-group --output_file=/tmp/output.pdf --format=pdf
```

rptsrvconfigaudit

NAME:

rptsrvconfigaudit

DESCRIPTION:

Generates a server configuration audit report to a specified, file in the TCPWave IPAM.

ARGUMENTS:**--appliance**

Name or Address of the server. [mandatory]

--from_date

Start date for report generation.Date format is mm/dd/yyyy [mandatory]

--to_date

End date for report generation.Date format is mm/dd/yyyy [mandatory]

--output_fileFull path to the output file to which audit report is to be written.
[mandatory]**--format**

Format of the output file. It takes 'csv' or 'pdf'. [mandatory]

Atleast one configuration type should be specified, from the below mentioned configuration types.

--banner

It takes '1' or '0'. '1' indicates report should contain banner configuration type. '0' indicates report should not contain banner configuration type.

--ntp

It takes '1' or '0'. '1' indicates report should contain NTP configuration type. '0' indicates report should not contain NTP configuration type.

--snmp

It takes '1' or '0'. '1' indicates report should contain SNMP configuration type. '0' indicates report should not contain SNMP configuration type.

--tacacs

It takes '1' or '0'. '1' indicates report should contain TACACS configuration type. '0' indicates report should not contain TACACS configuration type.

--bgp_ospf

It takes '1' or '0'. '1' indicates report should contain BGP/OSPF configuration type. '0' indicates report should not contain BGP/OSPF configuration type.

--zebra

It takes '1' or '0'. '1' indicates report should contain zebra configuration type. '0' indicates report should not contain zebra configuration type.

--virtual_ip

It takes '1' or '0'. '1' indicates report should contain virtual IP configuration type. '0' indicates report should not contain virtual IP configuration type.

--nic

It takes '1' or '0'. '1' indicates report should contain NIC configuration type. '0' indicates report should not contain NIC configuration type.

--syslog

It takes '1' or '0'. '1' indicates report should contain SYSLOG configuration type. '0' indicates report should not contain SYSLOG configuration type.

EXAMPLE USAGE:

```
twc rptsrvconfigaudit --appliance=10.1.10.180 --from_date=07/01/2015 --to_date=09/01/2015 --output_file=/tmp/output.pdf --format=pdf --banner=1 --nic=1 --syslog=1
```

rpttopalertproducers**NAME:**

rpttopalertproducers

DESCRIPTION:

Generates a top alerts producers report to a specified, file in the TCPWave IPAM.

ARGUMENTS:**--output_file**

Full path to the output file to which report is to be written. [mandatory]

--format

Format of the output files. It takes 'csv' or 'pdf'. [mandatory]

EXAMPLE USAGE:

```
twc rpttopalertproducers --output_file=/tmp/output.csv --format=csv
```

```
twc rpttopalertproducers --output_file=/tmp/output.pdf --format=pdf
```

rptv4netspaceutil**NAME:**

rptv4netspaceutil

DESCRIPTION:

Generates IPv4 network space utilization audit report in a specified, file in TCPWave IPAM.

ARGUMENTS:

--net_mask

Network mask. It is not required if --all argument is specified.,

--all

Takes only '1'. It is not required if --net_mask is specified.,

--output_file

Full path to the output file to which audit report is to be written.
[mandatory]

--format

Format of the output file. It takes 'csv' or 'pdf'. [mandatory]

EXAMPLE USAGE:

twc rptv4netspaceutil --net_mask=16 --output_file=/tmp/output.csv --format=csv

twc rptv4netspaceutil --net_mask=16 --output_file=/tmp/output.pdf --format=pdf

twc rptv4netspaceutil --all=1 --output_file=/tmp/output.csv --format=csv

twc rptv4netspaceutil --all=1 --output_file=/tmp/output.pdf --format=pdf

rptv4netspaceutil

NAME:

rptv4netspaceutil

DESCRIPTION:

Generates IPv4 subnet space utilization audit report in a specified, file in TCPWave IPAM

ARGUMENTS:

--subnet_mask

Subnet mask. It is not required if --all argument is specified.,

--all

Takes only '1'. It is not required if --subnet_mask is specified.,

--output_file

Full path to the output file to which audit report is to be written.
[mandatory]

--format

Format of the output file. It takes 'csv' or 'pdf'. [mandatory]

EXAMPLE USAGE:

twc rptv4netspaceutil --subnet_mask=16 --output_file=/tmp/output.csv --format=csv

```
twc rptv4snetspaceutil --subnet_mask=16 --output_file=/tmp/output.pdf --format=pdf
```

```
twc rptv4snetspaceutil --all=1 --output_file=/tmp/output.csv --format=csv
```

```
twc rptv4snetspaceutil --all=1 --output_file=/tmp/output.pdf --format=pdf
```

rptzoneaudit

NAME:

rptzoneaudit

DESCRIPTION:

Generates a zone audit report to a specified, file in the TCPWave IPAM.

ARGUMENTS:

--zone

Name of the zone. [mandatory]

--org

Name of the organization in which specified, zone exist. This argument is mandatory if the user is FADM.

--from_date

Start date for report generation.Date format is mm/dd/yyyy [mandatory]

--to_date

End date for report generation.Date format is mm/dd/yyyy [mandatory]

--output_file

Full path to the output file to which audit report is to be written.
[mandatory]

--format

Format of the output file. It takes 'csv' or 'pdf'. [mandatory]

--status

It takes 'active' or 'delete'. [mandatory]

EXAMPLE USAGE:

```
twc rptzoneaudit --zone=tcpwave.com --org=TCPWave --status=active --from_date=05/01/2018 --to_date=08/01/2018 --output_file=/tmp/output.csv --format=csv
```

```
twc rptzoneaudit --zone=tcpwave.com --org=TCPWave --status=active --from_date=05/01/2018 --to_date=08/01/2018 --output_file=/tmp/output.pdf --format=pdf
```

rptzonetmplaudit

NAME:

rptzonetmplaudit

DESCRIPTION:

Generates a zone template audit report to a specified, file in the TCPWave IPAM.

ARGUMENTS:

- zone_tmpl
Name of the template. [mandatory]
- from_date
Start date for report generation.Date format is mm/dd/yyyy [mandatory]
- to_date
End date for report generation.Date format is mm/dd/yyyy [mandatory]
- output_file
Full path to the output file to which audit report is to be written.
[mandatory]
- format
Format of the output file. It takes 'csv' or 'pdf'. [mandatory]
- status
It takes 'active' or 'deleted'. [mandatory]

EXAMPLE USAGE:

```
twc rptzonetmplaudit --zone_tmpl="Zone template" --status=active --from_date=08/01/2015 --to_date=09/01/2015 --output_file=/tmp/output.csv --format=csv
```

```
twc rptzonetmplaudit --zone_tmpl="Zone template" --status=active --from_date=08/01/2015 --to_date=09/01/2015 --output_file=/tmp/output.pdf --format=pdf
```

redistributeremotes**NAME:**

redistributeremotes

DESCRIPTION:

Redistributes the remote appliances in the high availability cluster.

ARUGUMENTS:

- ip
IP address of the remotes. It accepts the comma separated IP addresses. If this argument is not specified it will redistribute all remotes.

EXAMPLE USAGE:

```
twc redistributeremotes --ip=10.0.0.201,10.0.0.202
```

runpolicycompliance

NAME

runpolicycompliance

DESCRIPTION

Run the policy compliance on DNS,DHCP and IPAM appliances in the TCPWave IPAM.

ARGUMENTS

--ip

IP address of an appliance.

--org

Organization name to which the appliance or appliance group belongs. It is mandatory to run policy compliance on a particular 'appliances' or 'appliance group'. It is not mandatory for IPAM policy compliance check on appliance level.

--type

Type of the appliance. It takes 'BIND AUTH' or 'BIND CACHE' or 'UNBOUND' or 'NSD' or 'DNS PROXY' 'DHCP' or 'IPAM' as a input. If want to run the policy compliance on all the DNS appliances type should be 'DNS'.[mandatory]

--level

Policy compliance runs based on the level. It accepts 'appliance' or 'appliance_group' as a input. Specify type as 'appliance' if want to run compliance check on particular appliance or all the appliance. Type should be specified as appliance_group if want to run compliance check on all the appliances that belongs to a particular appliance group. [mandatory]

--all

It accepts '1' or '0' value. '1' indicates all the 'appliances' or 'appliance groups' are eligible to run policy compliance.'0' indicates particular 'appliance' or 'appliance group' is eligible to run policy compliance.

--appliance_group

Appliance group name to run policy compliance on particular group. If want to run the appliance group policy compliance on all the DNS appliances type should be 'DNS'.

EXAMPLE

```
twc runpolicycompliance --type="DNS" --level="appliance" --all=1 --org=TCPWave
```

```
twc runpolicycompliance --type="BIND AUTH" --level="appliance" --all=0 --ip=192.168.10.15 --org=TCPWave
```

```
twc runpolicycompliance --type=DHCP --level="appliance" --all=1 --org=TCPWave
```

```
twc runpolicycompliance --type=DHCP --level="appliance" --all=0 --ip=10.1.10.14
```

```
twc runpolicycompliance --type=IPAM --level="appliance" --all=1
```

```
tvc runpolicycompliance --type=IPAM --level="appliance" --all=0 --ip=10.1.10.12
tvc runpolicycompliance --type="DNS" --level="appliance group" --all=1 --org=TCPWave
tvc runpolicycompliance --type="DNS" --level="appliance group" --all=0 --
appliance_group=Appliance-Group
tvc runpolicycompliance --type=DHCP --level="appliance group" --all=1 --org=TCPWave
tvc runpolicycompliance --type=DHCP --level="appliance group" --all=0 --
appliance_group=Appliance-Group
tvc runpolicycompliance --type=IPAM --level="appliance group" --all=1
tvc runpolicycompliance --type=IPAM --level="appliance group" --all=0 --
appliance_group=Appliance-Group
```

search

NAME

search

DESCRIPTION

Performs a search operation in the TCPWave IPAM.

Text, Wildcard, Regex and Match search types are allowed.

1. Text search allows users to do fulltext search where all the elements containing the given text will be queried and displayed to the user.

2. Wildcard search works with * or ?. The word abb matches the wildcard query a* or *b. The word ab matches the wildcard query a? or ?b. * represents any number of characters. ? represent one character. It is not recommended to use only * in the query.

3. Regex search allows users to search the data using regular expression term queries. The "term queries" means that search engine will apply the regexp to the terms produced by the tokenizer for that field, and not to the original text of the field. The performance of a regexp query heavily depends on the regular expression chosen. Matching everything like .* is very slow. If possible, you should try to use a long prefix before your regular expression starts.

4. Match search allows users to search the data using match_phrase queries. The match_phrase query analyzes the text and creates a phrase query from the analyzed text that allows users to fetch the results for the search term which has special characters in it.

Please refer TCPWave IPAM Admin guide to know allowed regular expression patterns.

Field Name represents the specific name of the column to search for the data. It is mandatory if the search type is Wildcard or Regex. It's an optional field in case of Text search type.

ARGUMENTS

--search_type

Indicates the search type. It takes either 'Text', 'Wildcard', 'Regex' or 'Match' [mandatory]

--field_name

Indicates the field name for which the search is performed for the given search term.

It takes one of the following field names. 'Name','Resource Record Data','Resource Record Owner','Email Address', 'First Name','Phone Number','Last Name','Middle Name','Location', 'Network Address','End Of Life','Monitored By','Object Address', 'Resource Record Type','IPv4 Remote Appliance','IPv6 Remote Appliance', 'IPv4 Subnet','DHCP Lease Information','DHCP Lease Time','Managed By', 'Model Type','Description','Object Type','VLAN ID', 'Compressed V6 Network/Subnet','Expanded V6 Network/Subnet', or 'IPv6 Compressed Address'.

--search_term

Search term for which results are to be displayed. [mandatory]

--entity_type

Indicates the entity type for which the search is performed for the given search term. It takes 'network','object','subnet','subnet_group','zone','domain','admin','admingroup','contact','dns_server_template', 'organization','resource_record','template','location','acl','server', 'log_channel','scope','audit_history','v6_network','v6_subnet' or 'v6_object'.

--show_count

Takes '1' or '0'. '1' displays the counts for various entity types. '0' displays the actual results for the given entity type.

EXAMPLE

```
twc search --search_type="Text" --field_name="Name" --search_term="tcpwave" --
entity_type="Object"

twc search --search_type="Text" --search_term="Thomas" --entity_type=admin

twc search --search_type="Text" --field_name="Resource Record Owner" --search_term="www"

twc search --search_type="Text" --search_term="tcpwave" --show_count=1

twc search --search_type=Wildcard --field_name="Object Address" --search_term=192.*.27

twc search --search_type=Regex --field_name="Name" --search_term="ns.."

twc search --search_type=Regex --field_name="Name" --search_term="ns*" --
entity_type="Object"

twc search --search_type="Match" --field_name="Resource Record Owner" --
search_term="*.company.com."
```

setbgpconfig

NAME:

setbgpconfig

DESCRIPTION:

Updates BGP configuration of a DNS server in the TCPWave IPAM.

ARGUMENTS:

--ip

IP Address of the DNS server. DNS Server type should be 'BIND CACHE' or 'UNBOUND'. [mandatory]

--input_file

Full path to the input file from which the BGP configuration is to be read. [mandatory]

EXAMPLE USAGE:

```
twc setbgpconfig --ip=10.1.10.10 --input_file=/tmp/bgpconfig.txt
```

FILE FORMAT:

The input file format is as follows:

Each configuration section begins with a [<section name>] field followed by <name>=<value> pairs one per line

SECTIONS & CONFIGURATION PARAMETERS:

[Basic_Configuration]

ASN : Takes number [1-65535]

Router_ID : Takes a valid IP address

Debug_BGP_Events : Takes true or false

Debug_BGP_Updates : Takes true or false

Debug_BGP_Filters : Takes true or false

[BGP_Timer]

Keep_Alive : Takes number [0-65535]

Hold_down : Takes number [0-65535]

[Networks]

Network_List : Takes comma separated list of IP address with mask.

[Prefix_List]

Name : Should be a valid name

Sequence : Takes number

IP : Takes an IP address with mask

Prefix_length : Takes a number [Not mandatory]

permission : Takes Permit or Deny

[Route_Map]

Name : Should be a valid name

Permit : Takes number

Prefix_List : Takes prefix name specified, in the [Prefix_List] section

Set_community : Value can be character or number [Not mandatory]

Set_Local_Preference : Value can be character or number [Not mandatory]

[Neighbor_Group]

Name : Should be a valid name

Remote ASN : Takes number

Route_Map_In : Takes route name specified, in the [Route_Map] section [Not mandatory]

Route_Map_Out : Takes route name specified, in the [Route_Map] section [Not mandatory]

Prefix_List_In : Takes prefix name specified, in the [Prefix_List] section [Not mandatory]

Prefix_List_Out : Takes prefix name specified, in the [Prefix_List] section[Not mandatory]

[Neighbor]

Peer : Takes a valid IP address

Peer_Group : Takes neighbor group name specified, in the [Neighbor_Group] section

Description : Description for the neighbor [mandatory]

SAMPLE FILE CONTENTS:

[Basic_Configuration]

ASN=64881

Router_ID=192.168.1.80

Debug_BGP_Events=true

Debug_BGP_Updates=true

Debug_BGP_Filters=true

[BGP_Timer]

Keep_Alive=4

Hold_down=16

[Networks]

Network_List=192.193.215.64/30,192.193.215.68/30,192.193.215.72/30,192.168.1.80/32

```
[Prefix_List]
Name=DNS
Sequence=5
IP=192.193.215.64/30
Prefix_length=
permission=permit
[Prefix_List]
Name=DEFAULT
Sequence=5
IP=0.0.0.0/0
Prefix_length=
permission=permit
[Neighbor_Group]
Name=EBGP-PEERS
Remote ASN=64881
Route_Map_In=
Route_Map_Out=
Prefix_List_In=DEFAULT
Prefix_List_out=DNS
[Neighbor]
Peer=10.1.10.253
Peer_Group=EBGP-PEERS
Description=GSS-PEER-IP1-DESCRIPTION
```

setchangeticket**NAME:**

setchangeticket

DESCRIPTION:

Updates the change ticket associated with the current session in the TCPWave IPAM.

ARGUMENTS:**--ticket**

Ticket to be associated with the current session. [mandatory]

EXAMPLE USAGE:

```
twc setchangeticket --ticket=SN012345678
```

setdhcpclass**NAME:**

setdhcpclass

DESCRIPTION:

Creates or updates a DHCP class in the TCPWave IPAM.

ARGUMENTS:**--name**

Name of the DHCP class [mandatory]	
--type	Type of the DHCP class. Takes 'user','vendor' or 'client' [mandatory]
--action	Takes 'add' or 'edit'. add creates a new DHCP class from the input file contents. edit updates the contents of an existing DHCP class [mandatory]
--input_file	Full path to the input file from which the definition of the class (match expression, options, etc.) are to be read [mandatory]
--desc	Description of the DHCP class.
EXAMPLE USAGE:	
twc setdhcpclass --name=client-class --type=client --action=add --input_file=/tmp/client.txt --desc="Client class definition for client"	
twc setdhcpclass --name=sales --type=user --action=edit --input_file=/tmp/sales.txt --desc="User class definition for sales"	
SAMPLE INPUT FILE CONTENTS:	
option user-class = engineering;	
setdhcppolicytmpl	
NAME:	
setdhcppolicytmpl	
DESCRIPTION:	
Creates or updates a DHCP policy template in the TCPWave IPAM.	
ARGUMENTS:	
--name	Name of the DHCP policy template to be retrieved from TCPWave IPAM [mandatory]
--desc	description of the DHCP policy template.
--action	Takes 'add' or 'edit'. add creates a new DHCP policy template from the input file contents. edit updates the contents of an existing DHCP template [mandatory]
--new_temp_name	New template name if the name of an existing DHCP policy template needs

to be updated.

--input_file

Full path to the input file from which the template contents are to be read [mandatory]

--org

Organization name associated with the policy template.

This argument is for users in FADM role to select appropriate organization to which the operation must be applied.

If this parameter is not specified, for user in FADM role, the operation is by default applied to the organization that the user is associated with.

For users not in FADM role, the operation is by default applied to the organization that the user is associated with.

EXAMPLE USAGE:

```
twc setdhcpolicytmpl --name="base-policy-template" --input_file=/tmp/dhcppolicytemp.txt --action=add --org=TCPWave
```

```
twc setdhcpolicytmpl --name="base-policy-template" --input_file=/tmp/dhcppolicytemp.txt --action=edit --org=TCPWave
```

SAMPLE INPUT FILE CONTENTS:

```
# Global Parameters
Authoritative(authoritative)=yes
DB Time Format(db-time-format )=
Local Port(local-port )=
Local Address(local-address )=
Log Facility(log-facility)=local6
# DHCP Server-Client Communications Parameters
Always Broadcast(always-broadcast)=
Always reply RFC1048(always-reply-rfc1048)=
Min Secs(min-secs)=
Remote Port(remote-port )=
Stash Agent Options(stash-agent-options)=
# Client Handling Parameters
Adaptive Lease Time Percentage(adaptive-lease-time-threshold)=
Boot Unknown Clients(boot-unknown-clients)=
Default Lease time(default-lease-time)=
Get Lease Host Names(get-lease-hostnames)=
Infinite is reserved(infinite-is-reserved)=
Max Lease Time(max-lease-time)=
Min Lease Time(min-lease-time)=
Next Server(next-server)=
One Lease Per Client(one-lease-per-client)=
Ping Check(ping-check)=
Ping Timeout(ping-timeout)=
Use Lease Addr For Default Route(use-lease-addr-for-default-route)=
Server Identifier(server-identifier)=
```

Server Name(server-name)=
Site Option Space(site-option-space)=
Vendor Option Space(vendor-option-space)=

setdhcpserver

NAME:

setdhcpserver

DESCRIPTION:

Creates or updates DHCP server configuration in the TCPWave IPAM.

ARGUMENTS:

--action

Takes 'add' or 'edit' for create or update respectively.

--input_file

Full path to the input file from which the server configuration is to be read.

EXAMPLE USAGE:

```
twc setdhcpserver --input_file=/tmp/dhcpserver.txt --action=add
```

```
twc setdhcpserver --input_file=/tmp/dhcpserver.txt --action=edit
```

FILE FORMAT:

The input file format is as follows:

Each configuration section begins with a [<section name>] field followed by <name>=<value> pairs one per line.

SECTIONS & CONFIGURATION PARAMETERS:

[dhcp-server]

IP_ADDRESS IP address of the server

ORGANIZATION_NAME Organization Name where DHCP server must be created

POLICY_TEMPLATE Policy Template to be associated with the DHCP server

APPLIANCE_GROUP Comma separated name of the appliance groups to be associated

ENABLE_MONIT '0' to enable monitoring and '1' to disable monitoring

TIME_ZONE Time zone

[ntp]

NTP_SERVERS Comma separated list of IP addresses of NTP servers

[dns-resolver]

NAME_APPLIANCES comma separated list of IP addresses of DNS servers. Maximum allowed DNS server is four.

SEARCH_SUFFIXES comma separated list of domain names. Maximum allowed search suffix is six.

[snmp]

TRAP_SINK_1 IP address of SNMP trap sink
TRAP_SINK_2 IP address of SNMP trap sink
COMMUNITY_STRING Community string for SNMP
SYSTEM_LOCATION System Location
SYSTEM_CONTACT System contact
PROCESS_LIST Comma separated list of processes to be monitored. The following is a valid list of processes:
 ntpd, dns, bgpd, zebra, crond, sshd, monit, syslog-ng,dhcpd
ENABLE_SNMPV3 Takes 'true' or 'false'. 'true' indicates that SNMPv3 is enable. 'false' indicates that SNMPv3 is disable.

[snmpv3]

USER_NAME User name of SNMPv3
PASSWORD Password of the specified, user
AUTHENTICATION_PROTOCOL Authentication protocol
ENCRYPTION_PROTOCOL Encryption protocol

[tacacs]

ENABLE_TACACS Takes '0' or '1'. '1' indicates TACACS+ configuration should be enabled for this server. '0' indicates TACACS+ configuration should be disabled
TACACS_PASSKEY TACACS passkey
TACACS_SERVERS Comma separated list of TACACS servers.

[syslogng-global-options]

TIME_REOPEN The time to wait in seconds before a dead connection is reestablished. Takes a value less than or equal to 32767.
TIME_REAP If no new messages are written to a destination within the specified, time in seconds, the connection will be closed, and its state will be freed. Takes a value less than or equal to 32767.
FLUSH_LINES Specifies how many lines are flushed to a destination at a time. Takes a value less than or equal to 32767.
STATS_FREQ Syslog-*ng* OSE periodically sends a log statistics message. Takes a value less than or equal to 32767.
LOG_FIFO_SIZE The number of messages that the output queue can store. Takes a value less than or equal to 32767.
LOG_MSG_SIZE The maximal length of the log messages is limited by this option It is not recommended to set the option value higher than 10 MiB. Takes a value less than or equal to 32767.
KEEP_TIMESTAMP Specifies whether syslog-*ng* should accept the timestamp received from the sending application or client. Takes value 'Yes' or 'No'.

[syslogng-source]

SOURCE_NAME Name of the Source

INTERNAL_MSG Internal syslog-NG message, takes input values as '0' or '1'. default value is '1'.

SYSTEM_MSG System specific log message, takes input value as '0' or '1'.

MSG_TXT_FILE Message from text file, takes the file name as input.

MSG_MULTI_TXT_FILE Message from multiple text files, takes input '0' or '1'.

if this flag is '1' need to specify the FILE_PATH and

FILE_PATTERN.

FILE_PATH File patch to the multiple text file.

FILE_PATTERN File Name pattern.

SYSLOG_SERVER Syslog-NG sever, takes the input as '0' or '1'.

IP_ADDRESS IP address of the syslog server.

PORT Port number of the syslog server.

NETWORK_PROTOCOL Network protocol, supports 'UDP' and 'TCP'.

[syslogng-filter]

FILTER_NAME Name of the Filter.

CONDITION Takes the input as 'complex' or 'simple'.

FACILITIES Allow values are one or more comma separated option given below.

auth, authpriv, cron, deamon, kern, lpr, mail,mark, news, syslog, user, uucp, local0, local1,local2, local3, local4, local5, local6, local7.

PRIORITIES Allow values are one or more comma separated option given below.

info, notice, warning, err, crit, alert, emerg.

HOST_NAME Name of the host.

IP_NETWORK IP address with mask length.

MATCH_EXPRESSION Match expression.

PROGRAM Program.

[syslogng-destination]

DESTINATION_NAME Name of the destination.

TYPE_SNG Type of the destination. takes the value between 1 to 5.

'1'= File

'2'= Named pipe

'3'= Local Users

'4'= All logged-in users

'5'= Syslog server.

LOG_FILE_NAME File name to log the message, mandatory when TYPE_SNG is specified, as '1'.

NAMED_PIPE_NAME Named pipe name, mandatory when TYPE_SNG is specified, as '2'.

LOCAL_USERS Local users, mandatory when TYPE_SNG is specified, as '3'.

SYSLOG_SERVER IP address of the syslog server, mandatory when TYPE_SNG is specified, as '5'.

PORT Port number of the syslog server, mandatory when TYPE_SNG is specified, as '5'.

NETWORK_PROTOCOL Network protocol, supports 'UDP' and 'TCP', mandatory when TYPE_SNG is specified, as '5'.

[syslogng-target]

SOURCE Name of the source.
FILTER Name of the filter.
DESTINATION Name of the destination.

[macexclusions]

MAC MAC address of the device to be excluded
DESCRIPTION Description for the MAC address exclusion

SAMPLE INPUT FILE CONTENTS:**[dhcp-server]**

IP_ADDRESS=10.1.10.180
ORGANIZATION_NAME=TCPWave
POLICY_TEMPLATE=Clone-Policy-1
APPLIANCE_GROUP=Appliance-Group1,Appliance-Group2
ENABLE_MONIT=1
TIME_ZONE=GMT (GMT)
[ntp]

NTP_SERVERS=10.1.10.10,10.1.10.11,10.1.10.12,10.1.10.13,

[dns-resolver]

NAME_APPLIANCES=8.8.8.8
SEARCH_SUFFIXES=tcpwave.com

[snmp]

TRAP_SINK_1=10.1.10.15
TRAP_SINK_2=10.1.10.18
COMMUNITY_STRING=public
SYSTEM_LOCATION=
SYSTEM_CONTACT=
PROCESS_LIST=dns,
ENABLE_SNMPV3=true

[snmpv3]

USER_NAME=admin
PASSWORD=abc123
AUTHENTICATION_PROTOCOL=SHA
ENCRYPTION_PROTOCOL=AES

[syslogng-global-options]

TIME_REOPEN=60
TIME_REAP=60
FLUSH_LINES=60
STATS_FREQ=600

```
LOG_FIFO_SIZE=1000
LOG_MSG_SIZE=2048
KEEP_TIMESTAMP=Yes
```

```
[syslogng-source]
```

```
SOURCE_NAME=s_sys
INTERNAL_MSG=1
SYSTEM_MSG=1
```

```
[syslogng-filter]
```

```
FILTER_NAME=f_kernal
CONDITION=complex
COMPLEX_CONDITION=facility(kern)
```

```
[syslogng-filter]
```

```
FILTER_NAME=f_default
CONDITION=complex
COMPLEX_CONDITION=level(info..emerg) and not (facility(mail) or facility(authpriv) or
facility(cron))
```

```
[syslogng-filter]
```

```
FILTER_NAME=f_auth
CONDITION=complex
COMPLEX_CONDITION=facility(authpriv)
```

```
[syslogng-filter]
```

```
FILTER_NAME=f_mail
CONDITION=complex
COMPLEX_CONDITION=facility(mail)
```

```
[syslogng-filter]
```

```
FILTER_NAME=f_emergency
CONDITION=complex
COMPLEX_CONDITION=level(emerg)
```

```
[syslogng-filter]
```

```
FILTER_NAME=f_news
CONDITION=complex
COMPLEX_CONDITION=facility(uucp) or (facility(news) and level(crit..emerg))
```

```
[syslogng-filter]
```

```
FILTER_NAME=f_boot
CONDITION=complex
```

```
COMPLEX_CONDITION=facility(local7)
```

```
[syslogng-filter]
```

```
FILTER_NAME=f_cron  
CONDITION=complex  
COMPLEX_CONDITION=facility(cron)
```

```
[syslogng-filter]
```

```
FILTER_NAME=filter  
CONDITION=simple  
FACILITIES=auth,authpriv,cron  
PRIORITIES=info,notice  
HOST_NAME=local  
IP_NETWORK=192.166.0.2/24  
MATCH_EXPRESSION=exp  
PROGRAM=prog
```

```
[syslogng-destination]
```

```
DESTINATION_NAME=d_cons  
LOG_FILE_NAME=console  
TYPE_SNG=1  
ENABLE_SYNC=Yes
```

```
[syslogng-destination]
```

```
DESTINATION_NAME=d_mesg  
LOG_FILE_NAME=messages  
TYPE_SNG=1  
ENABLE_SYNC=Yes
```

```
[syslogng-destination]
```

```
DESTINATION_NAME=d_auth  
LOG_FILE_NAME=secure  
TYPE_SNG=1  
ENABLE_SYNC=Yes
```

```
[syslogng-destination]
```

```
DESTINATION_NAME=d_mail  
LOG_FILE_NAME=maillog  
TYPE_SNG=1  
ENABLE_SYNC=Yes  
[syslogng-destination]
```

```
DESTINATION_NAME=d_spooler  
LOG_FILE_NAME=spooler  
TYPE_SNG=1
```

```
ENABLE_SYNC=Yes
```

```
[syslogng-destination]
```

```
DESTINATION_NAME=d_boot  
LOG_FILE_NAME=boot.log  
TYPE_SNG=1  
ENABLE_SYNC=Yes
```

```
[syslogng-destination]
```

```
DESTINATION_NAME=d_cron  
LOG_FILE_NAME=cron  
TYPE_SNG=1  
ENABLE_SYNC=Yes
```

```
[syslogng-destination]
```

```
DESTINATION_NAME=d_kern  
LOG_FILE_NAME=kern  
TYPE_SNG=1  
ENABLE_SYNC=Yes
```

```
[syslogng-destination]
```

```
DESTINATION_NAME=d_mlal  
TYPE_SNG=4
```

```
[syslogng-destination]
```

```
DESTINATION_NAME=ttt  
TYPE_SNG=4
```

```
[syslogng-destination]
```

```
DESTINATION_NAME=port  
TYPE_SNG=5  
SYSLOG_SERVER=192.166.0.2  
NETWORK_PROTOCOL=UDP  
PORT=514
```

```
[syslogng-target]
```

```
SOURCE=s_sys  
FILTER=f_kernal  
DESTINATION=d_kern
```

```
[syslogng-target]
```

```
SOURCE=s_sys  
FILTER=f_default
```

```
DESTINATION=d_mesg

[syslogng-target]

SOURCE=s_sys
FILTER=f_auth
DESTINATION=d_auth

[syslogng-target]

SOURCE=s_sys
FILTER=f_mail
DESTINATION=d_mail

[syslogng-target]

SOURCE=s_sys
FILTER=f_emergency
DESTINATION=d_mlal

[syslogng-target]

SOURCE=s_sys
FILTER=f_news
DESTINATION=d_spol

[syslogng-target]

SOURCE=s_sys
FILTER=f_boot
DESTINATION=d_boot

[syslogng-target]

SOURCE=s_sys
FILTER=f_cron
DESTINATION=d_cron
[macexclusions]

MAC=8E-2C-E7-88-53-7A
DESCRIPTION=

[macexclusions]

MAC=E0:8F:8D:59:CF:60
DESCRIPTION=
setdhcptmpl
NAME:
setdhcptmpl
```

DESCRIPTION:

Creates or updates DHCP option template in the TCPWave IPAM.

ARGUMENTS:

--name

Name of the DHCP option template to be retrieved from TCPWave IPAM
[mandatory]

--desc

Description of the DHCP option template.

--action

Takes 'add' or 'edit'. add creates a new DHCP template from the input file contents. edit updates the contents of an existing DHCP option template [mandatory]

--new_temp_name

New template name if the name of an existing DHCP option template needs to be updated.

--input_file

Full path to the input file from which the template contents are to be read [mandatory]

--org

Organization name associated with the option template.
This argument is for users in FADM role to select appropriate organization to which the operation must be applied.
If this parameter is not specified, for user in FADM role, the operation is by default applied to the organization that the user is associated with.
For users not in FADM role, the operation is by default applied to the organization that the user is associated with.

EXAMPLE USAGE:

```
twc setdhcptmpl --name="voip devices template" --input_file=/tmp/dhcpoPTIONtmpl.txt --action=add --org=TCPWave
```

```
twc setdhcptmpl --name="voip devices template" --input_file=/tmp/dhcpoPTIONtmpl.txt --action=edit --org=TCPWave
```

SAMPLE INPUT FILE CONTENTS:

```
# User Authentication Protocol Options
User Authentication Servers (98)=
# TCP Parameters
Default TCP TTL (37)=125
Keepalive Time (38)=
Keepalive Data (39)=
# SLP Protocol Options
```

Service Location Protocol Directory Agent (78)=
SLP Service Scope (79)=
RFC 3397 Options
Domain Search (119)=
RFC 1497 Vendor Extensions
Subnet Mask (1)=Same as in subnet profile
Time Offset (2)=
Router (3)=Same as in subnet profile
Time Server (4)=
Name Server (5)=
Domain Name Server (6)=
Log Server (7)=
Quotes Server (8)=
LPR Server (9)=
Impress Server (10)=
RLP Server (11)=
Hostname (12)=
Boot File Size (13)=
Merit Dump File (14)=
Domain Name (15)=
Swap Server (16)=
Root Path (17)=
Extension File (18)=
Novell Options
Netware/IP Domain (62)=
Netware/IP Options-nwip.nsq-broadcast (63)=
Netware/IP Options-nwip.preferred-dss (63)=
Netware/IP Options-nwip.nearest-nwip-server (63)=
Netware/IP Options-nwip.autoretries (63)=
Netware/IP Options-nwip.autoretry-secs (63)=
Netware/IP Options-nwip.nwip-1-1 (63)=
Netware/IP Options-nwip.primary-dss (63)=
NDS Servers (85)=
NDS Tree Name (86)=
NDS Context (87)=
Miscellaneous
MTU Subnet (27)=
Address Request (50)=
DHCP Message Type (53)=
Parameter List (55)=
DHCP Max Msg Size (57)=
Home Agent Addresses (68)=
User Class (77)=
Netinfo Address (112)=
Netinfo Tag (113)=
Default URL (114)=
Vendor Identified Vendor-Specific Information (125)=
Client FQDN (81)=
Link Layer Parameters per interface
Trailers (34)=
ARP Timeout (35)=

Ethernet (36)=
IP Layer Params Per Host
Forward On/Off (19)=
Source Routing (20)=
Policy Filter (21)=
Max Datagram Size (22)=
Default IP TTL (23)=
MTU Timeout (24)=
MTU Plateau (25)=
IP Layer Parameters per interface
MTU Interface (26)=
Broadcast Address (28)=
Mask Discovery (29)=
Mask Supplier (30)=
Router Discovery (31)=
Router Request (32)=
Static Route (33)=
DHCP Extensions
Address Time (51)=
Overload (52)=
Vendor Class Id (60)=
Client Id (61)=
Server Name (66)=
Bootfile Name (67)=
Application and Service Parameters
NIS Domain (40)=
NIS Servers (41)=
NTP Servers (42)=
Vendor Specific (43)=
NETBIOS Name Server (44)=
NETBIOS Dist Server (45)=
NETBIOS Node Type (46)=
NETBIOS Scope (47)=
X Window Font (48)=
X Window Manager (49)=
NIS+ Domain Name (64)=
NIS+ Server Address (65)=
SMTP Server (69)=
POP3 Server (70)=
NNTP Server (71)=
WWW Server (72)=
Finger Server (73)=
IRC Server (74)=
StreetTalk Server (75)=
StreetTalk Directory Assistance (STDA) Server (76)=
BCMCS Controller Domain Name (88)=
BCMCS Controller IPv4 address option (89)=
voip-options (CUSTOM OPTIONS)
ip-map (130)=
options (OPTION SPACES)
string (1)=

setdnsacl**NAME:**

setdnsacl

DESCRIPTION:

Creates or updates a DNS ACL in the TCPWave IPAM.

ARGUMENTS:**--name**

Name of the DNS ACL. The name should not exceed 20 characters. Hyphens (-) are allowed. [mandatory]

--new_name

New name of the DNS ACL if the name must be updated when --action argument is specified, as 'edit'.

--desc

Description for the DNS ACL.

--acl

Comma separated list of ACL elements in one of the following formats:
IPAddress/permission (192.168.0.1/Allow)
ACL-name/permission (internal/Deny)
IPAddress/mask/permission (192.168.0.0/24/Allow)
[mandatory]

--action

Takes 'add' or 'edit'. 'add' creates a new DNS ACL. 'edit' updates an existing ACL specified, by --name argument [mandatory]

EXAMPLE USAGE:

```
twc setdnsacl --name=internal --desc="internal servers" --
acl="173.0.2.0/24/Allow,173.0.0.3/Deny" --action=add
```

```
twc setdnsacl --name=external --desc="external servers" --
acl="172.0.0.1/24/Allow,172.0.0.2/Deny,internal/Deny" --action=add
```

```
twc setdnsacl --name=external --desc="external servers" --
acl="172.0.0.1/24/Allow,172.0.0.2/Allow,internal/Allow" --action=edit
```

setdnsdebuglevel**NAME:**

setdnsdebuglevel

DESCRIPTION:

Updates the debug level of a DNS server in the TCPWave IPAM.
DNS server should be 'BIND AUTH', 'BIND CACHE', 'UNBOUND' or 'NSD'.

ARGUMENTS:

--ip
IP address of the DNS server [mandatory]

--level
Debug level value. It takes value from 0 to 99 [mandatory]

EXAMPLE USAGE:

twc setdnsdebuglevel --ip=10.1.10.240 --level=1

setdnsopptmpl

NAME

setdnsopptmpl

DESCRIPTION

Creates or updates a DNS option template in the TCPWave IPAM.

ARGUMENTS

--name
Name of the DNS option template to be added or edited in TCPWave IPAM
[mandatory]

--new_temp_name
New template name if the name of an existing DNS option template needs
to be updated.

--org
Name of the organization. [mandatory]

--type
DNS server type. Takes one of the following values: 'BIND AUTH',
'BIND CACHE', 'UNBOUND', 'NSD', 'DNS PROXY'.

--desc
Description of the DNS option template to be added or edited in TCPWave
IPAM.

--input_file
Full path to the input file from which the template contents are to be

read. For input file format please refer to the output generated by twc getdnsopttmpl command [mandatory]

--action

Takes 'add' or 'edit'. add creates a new DNS option template from the input file contents. edit updates the contents of an existing DNS option template [mandatory]

--org

Organization name associated with the option template.

This argument is for users in FADM role to select appropriate organization to which the operation has to be applied.

If this parameter is not specified for user in FADM role, the operation is by default applied to the organization that the user is associated with.

For users not in FADM role, the operation is by default applied to the organization that the user is associated with.

EXAMPLE

```
twc setdnsopttmpl --name="bind-auth-template" --type="BIND AUTH" --  
input_file=/tmp/dnsopttemp.txt --action=add --org=TCPWave --desc="BIND AUTH Template"
```

```
twc setdnsopttmpl --name="bind-auth-template" --new_temp_name="BIND-AUTH-TEMP" --  
type="BIND AUTH" --input_file=/tmp/dnsopttemp.txt --action=edit --org=TCPWave
```

SAMPLE INPUT FILE CONTENTS FOR BIND AUTH TYPE:

```
Directory(directory)=/  
Allow Query(allow-query)=any/Allow;  
Allow Recursion(allow-recursion)=any/Allow;  
Allow Transfer(allow-transfer)=none/Allow;  
Blackhole(blackhole)=none/Allow;  
DNSSEC Enable(dnssec-enable)=yes  
DNSSEC Validation(dnssec-validation)=yes  
Listen On v6(listen-on-v6)=none  
Check SRV CNAME(check-srv-cname)=ignore  
Check MX CNAME(check-mx-cname)=ignore  
Check MX(check-mx)=ignore
```

Check Names(check-names)=master ignore,response ignore
Dump File(dump-file)=/var/named/log/named_dump.db
Lame TTL(lame-ttl)=0
Max Negative Cache TTL(max-nocache-ttl)=60
Minimal Responses(minimal-responses)=yes
PID File(pid-file)=/var/run/named/named.pid
Recursion(recursion)=no
Session Key File(session-keyfile)=/var/run/named/session.key
Statistics File(statistics-file)=/var/named/log/named.stats
TCP Clients(tcp-clients)=500
Version(version)=TCPWave DNS Server
Zone Statistics(zone-statistics)=yes
Empty Zones Enable(empty-zones-enable)=no
Responses Per Second(responses-per-second)=0
Referrals Per Second(referrals-per-second)=0
NODATA Per Second(nodata-per-second)=0
NXDOMAINs Per Second(nxdomains-per-second)=0
Errors Per Second(errors-per-second)=0
All Per Second(all-per-second)=0
Window(window)=15
QPS Scale(qps-scale)=
IPv4 Prefix Length(ipv4-prefix-length)=24
IPv6 Prefix Length(ipv6-prefix-length)=56
Slip(slip)=0
Log Only(log-only)=yes
Exempt Clients(exempt-clients)=none;
Max Table Size(max-table-size)=500
Min Table Size(min-table-size)=500
RRSet Order(rrset-order)=cyclic
Sort List(sortlist)={192.168.2.23;{212.1.2.0/24;};{1.1.1.1;{24.234.4.56;12.34.67.0/24;}};}
Transfers In(transfers-in)=10

Transfers Out(transfers-out)=10
Transfers Per NS(transfers-per-ns)=2
Recursive Clients(recursive-clients)=20000
Forward(forward)=first
Forwarders(forwarders)=
Custom Parameters(custom-params)=
DNS Cookies(cookie-enabled)=yes

SAMPLE INPUT FILE CONTENTS FOR BIND CACHE TYPE:

Directory(directory)=/
Allow Query(allow-query)=any/Allow;
Allow Recursion(allow-recursion)=any/Allow;
Recursive Clients(recursive-clients)=20000
Allow Transfer(allow-transfer)=none/Allow;
Blackhole(blackhole)=none/Allow;
DNSSEC Enable(dnssec-enable)=yes
DNSSEC Validation(dnssec-validation)=yes
Listen On v6(listen-on-v6)=none
Check SRV CNAME(check-srv-cname)=ignore
Check MX CNAME(check-mx-cname)=ignore
Check MX(check-mx)=ignore
Check Names(check-names)=master ignore,response ignore
Dump File(dump-file)=/var/named/log/named_dump.db
Lame TTL(lame-ttl)=600
Max Negative Cache TTL(max-nocache-ttl)=10800
Minimal Responses(minimal-responses)=yes
PID File(pid-file)=/var/run/named/named.pid
Recursion(recursion)=yes
Session Key File(session-keyfile)=/var/run/named/session.key
Statistics File(statistics-file)=/var/named/log/named.stats
TCP Clients(tcp-clients)=150
Version(version)=TCPWave DNS Server

Zone Statistics(zone-statistics)=yes
Empty Zones Enable(empty-zones-enable)=no
Forward(forward)=first
Forwarders(forwarders)=
Responses Per Second(responses-per-second)=0
Referrals Per Second(referrals-per-second)=0
NODATA Per Second(nodata-per-second)=0
NXDOMAINs Per Second(nxdomains-per-second)=0
Errors Per Second(errors-per-second)=0
All Per Second(all-per-second)=0
Window(window)=15
QPS Scale(qps-scale)=
IPv4 Prefix Length(ipv4-prefix-length)=24
IPv6 Prefix Length(ipv6-prefix-length)=56
Slip(slip)=0
Log Only(log-only)=no
Exempt Clients(exempt-clients)=none;
Max Table Size(max-table-size)=500
Min Table Size(min-table-size)=500
Custom Parameters(custom-params)=
DNS Cookies(cookie-enabled)=no

SAMPLE INPUT FILE CONTENTS FOR UNBOUND TYPE:

Extended Statistics(extended-statistics)=yes
Interface(interface)=0.0.0.0
Outgoing Number of TCP(outgoing-num-tcp)=50
Incoming Number of TCP(incoming-num-tcp)=50
SO_RCVBUF(so-rcvbuf)=4m
EDNS Buffer Size(edns-buffer-size)=4096
Access Control(access-control)=0.0.0.0/0 allow
Message Buffer Size(msg-buffer-size)=65552
Message Cache Size(msg-cache-size)=8m

Number Of Queries Per Thread(num-queries-per-thread)=1024

Do IP4(do-ip4)=yes

Do IP6(do-ip6)=no

Do UDP(do-udp)=yes

Do TCP(do-tcp)=yes

Do Daemonize(do-daemonize)=yes

CHROOT(chroot)=/opt/tcpwave/etc/unbound

Username(username)=twcadmin

Directory(directory)=/opt/tcpwave/etc/unbound

Use Syslog(use-syslog)=no

Log File(logfile)=/var/log/twcdns.log

PID File(pidfile)=/opt/tcpwave/etc/unbound/unbound.pid

Root Hints(root-hints)=/opt/tcpwave/etc/unbound/db.cache

Hide Version(hide-version)=yes

Harden Glue(harden-glue)=yes

Log Time ASCII(log-time-ascii)=yes

Private Address(private-address)=1.0.0.0/8

Local Zone(local-zone)=10.in-addr.arpa nodefault

Cache Max Negative TTL(cache-max-negative-ttl)=3600

Module Config(module-config)=validator iterator

Module Config(module-config)=first

Module Config(module-config)=

Custom Parameters(custom-params)=

DNS Cookies(cookie-enabled)=no

SAMPLE INPUT FILE CONTENTS FOR NSD TYPE:

Server Count(server-count)=1

Hide Version(hide-version)=no

Version(version)=NSD

Identity(identity)=unidentified server

NSID(nsid)=aabbccdd

TCP Count(tcp-count)=100

TCP Query Count(tcp-query-count)=0
TCP Timeout(tcp-timeout)=120
IPv4 EDNS Size(ipv4-edns-size)=4096
Transfer Reload Timeout(xfrd-reload-timeout)=1
Ascii Log Time(log-time-ascii)=yes
Round Robin(round-robin)=no
Zone Files Check(zonefiles-check)=yes
Zone Files Write Seconds(zonefiles-write)=3600
RRL Size(rrl-size)=1000000
RRL Rate Limit(rrl-ratelimit)=200
RRL Slip(rrl-slip)=2
RRL IPv4 Prefix Length(rrl-ipv4-prefix-length)=24
RRL IPv6 Prefix Length(rrl-ipv6-prefix-length)=64
RRL Whitelist Rate limit(rrl-whitelist-ratelimit)=2000
Custom Parameters(custom-params)=
DNS Cookies(cookie-enabled)=no

SAMPLE INPUT FILE CONTENTS FOR DNS PROXY TYPE:

Directory(directory)=/
Allow Query(allow-query)=any/Allow;
Allow Recursion(allow-recursion)=any/Allow;
Allow Transfer(allow-transfer)=none/Allow;
Blackhole(blackhole)=none/Allow;
Listen On(listen-on)=127.0.0.1
Check SRV CNAME(check-srv-cname)=ignore
Check MX CNAME(check-mx-cname)=ignore
Check MX(check-mx)=ignore
Check Names(check-names)=master ignore,response ignore
Responses Per Second(responses-per-second)=0
Window(window)=15
Dump File(dump-file)=/var/named/log/named_dump.db
Lame TTL(lame-ttl)=600

Max Negative Cache TTL(max-nocache-ttl)=10800
Minimal Responses(minimal-responses)=yes
PID File(pid-file)=/var/run/named/named.pid
Recursion(recursion)=no
Session Key File(session-keyfile)=/var/run/named/session.key
Statistics File(statistics-file)=/var/named/log/named.stats
TCP Clients(tcp-clients)=150
Version(version)=TCPWave DNS Server
Zone Statistics(zone-statistics)=yes
Empty Zones Enable(empty-zones-enable)=no
Custom Parameters(custom-params)=
DNS Cookies(cookie-enabled)=no

setdhcptotmpl**NAME**

setdhcptotmpl

DESCRIPTION

Creates or updates DHCP option template in the TCPWave IPAM.

ARGUMENTS

--name

Name of the DHCP option template to be retrieved from TCPWave IPAM
mandatory when input file is not specified.

--desc

Description of the DHCP option template.

--action

Takes 'add' or 'edit'. add creates a new DHCP template from the input
file contents. edit updates the contents of an existing DHCP option
template. [mandatory]

--new_temp_name

New template name if the name of an existing DHCP option template needs
to be updated.

--input_file

Full path to the input file from which the template contents are to be
read, mandatory if the action is add. This field is not mandatory if user
wants to update the template name only.

--org

Organization name associated with the option template.

EXAMPLE:

```
twc setdhcptotmpl --input_file=/tmp/dhcpooptiontmpl.txt --action=add
twc setdhcptotmpl --input_file=/tmp/dhcpooptiontmpl.txt --action=edit
twc setdhcptotmpl --name="voip devices template" --new_temp_name="dhcp-opt-temp" --
action=edit --org=TCPWave
```

SAMPLE INPUT FILE CONTENTS

```
# User Authentication Protocol Options  
User Authentication Servers (98)=  
  
# TCP Parameters  
Default TCP TTL (37)=125  
  
Keepalive Time (38)=  
  
Keepalive Data (39)=  
  
# SLP Protocol Options  
Service Location Protocol Directory Agent (78)=  
  
SLP Service Scope (79)=  
  
# RFC 3397 Options  
Domain Search (119)=  
  
# RFC 1497 Vendor Extensions  
Subnet Mask (1)=Same as in subnet profile  
  
Time Offset (2)=  
  
Router (3)=Same as in subnet profile  
  
Time Server (4)=  
  
Name Server (5)=  
  
Domain Name Server (6)=  
  
Log Server (7)=  
  
Quotes Server (8)=  
  
LPR Server (9)=  
  
Impress Server (10)=  
  
RLP Server (11)=  
  
Hostname (12)=  
  
Boot File Size (13)=  
  
Merit Dump File (14)=  
  
Domain Name (15)=  
  
Swap Server (16)=  
  
Root Path (17)=
```

Extension File (18)=

Novell Options

Netware/IP Domain (62)=

Netware/IP Options-nwip.nsq-broadcast (63)=

Netware/IP Options-nwip.preferred-dss (63)=

Netware/IP Options-nwip.nearest-nwip-server (63)=

Netware/IP Options-nwip.autoretries (63)=

Netware/IP Options-nwip.autoretry-secs (63)=

Netware/IP Options-nwip.nwip-1-1 (63)=

Netware/IP Options-nwip.primary-dss (63)=

NDS Servers (85)=

NDS Tree Name (86)=

NDS Context (87)=

Miscellaneous

Netinfo Address (112)=

Netinfo Tag (113)=

Default URL (114)=

Vendor Identified Vendor-Specific Information (125)=

MTU Subnet (27)=

Parameter List (55)=

DHCP Max Msg Size (57)=

DHCP Renewal time (58)=

DHCP Rebinding time (59)=

Home Agent Addresses (68)=

User Class (77)=

Agent/Circuit Id (82)=

Agent/Remote Id (82)=

Link Layer Parameters per interface

Trailers (34)=

ARP Timeout (35)=

Ethernet (36)=

IP Layer Params Per Host
Forward On/Off (19)=
Source Routing (20)=
Policy Filter (21)=
Max Datagram Size (22)=
Default IP TTL (23)=
MTU Timeout (24)=
MTU Plateau (25)=
IP Layer Parameters per interface
MTU Interface (26)=
Broadcast Address (28)=
Mask Discovery (29)=
Mask Supplier (30)=
Router Discovery (31)=
Router Request (32)=
Static Route (33)=
DHCP Extensions
Address Time (51)=
Overload (52)=
Vendor Class Id (60)=
Client Id (61)=
Server Name (66)=
Bootfile Name (67)=
Application and Service Parameters
NIS Domain (40)=
NIS Servers (41)=
NTP Servers (42)=
Vendor Specific (43)=
NETBIOS Name Server (44)=
NETBIOS Dist Server (45)=

NETBIOS Node Type (46)=
NETBIOS Scope (47)=
X Window Font (48)=
X Window Manager (49)=
NIS+ Domain Name (64)=
NIS+ Server Address (65)=
SMTP Server (69)=
POP3 Server (70)=
NNTP Server (71)=
WWW Server (72)=
Finger Server (73)=
IRC Server (74)=
StreetTalk Server (75)=
StreetTalk Directory Assistance (STDA) Server (76)=
BCMCS Controller Domain Name (88)=
BCMCS Controller IPv4 address option (89)=
voip-options (CUSTOM OPTIONS)
ip-map (130)=
options (OPTION SPACES)
string (1)=

setipv6remotemonitor

NAME

setipv6remotemonitor

DESCRIPTION

Updates the monitoring status of an IPv6 DNS or DHCP appliance from the TCPWave IPAM.

ARGUMENTS

--ip

The IP address of the IPv6 DNS or DHCP appliance [mandatory].

--enable_monitor

Enable or disable the remote monitoring status of an IPv6 DNS or DHCP appliance. It takes 'Yes' or 'No' as input [mandatory].

--type

Type of the appliance. It takes 'DHCP' or 'DNS' as input [mandatory].

EXAMPLE

```
twc setipv6remotemonitor --ip=2001:1::2 --enable_monitor=Yes --type=DNS
```

```
twc setipv6remotemonitor --ip=2001:1::2 --enable_monitor=No --type=DHCP
```

setdnsserver

NAME

setdnsserver

DESCRIPTION

Creates or updates DNS appliance configuration in the TCPWave IPAM.

ARGUMENTS

--action

Takes 'add' or 'edit' for create or update, respectively.

--input_file

Full path to the input file from which the appliance configuration is to be read.

EXAMPLE

```
twc setdnsserver --input_file=/tmp/dnsserver.txt --action=add
```

```
twc setdnsserver --input_file=/tmp/dnsserver.txt --action=edit
```

FILE FORMAT

The input file format is as follows:

Each configuration section begins with a [<section name>] field followed

by <name>=<value> pairs one per line.

SECTIONS & CONFIGURATION PARAMETERS:

[dns-server]

TYPE Takes 'BIND AUTH' or 'BIND CACHE' or 'UNBOUND' or 'NSD' or 'DNS PROXY'

OPTION_TEMPLATE DNS Option template name

APPLIANCE_TEMPLATE DNS appliance template

IP_ADDRESS IP address of the appliance

ORGANIZATION_NAME Organization Name of the DNS appliance

APPLIANCE_GROUP Comma separated name of the appliance groups to be associated

ENABLE_MONIT '0' to enable monitoring and '1' to disable monitoring

INTERNAL_CACHE Applicable for appliances of type 'BIND CACHE' and 'BIND AUTH'. '0' indicates that the appliance is rooted at an internal root server. '1' indicates that the appliance is rooted at public internet root appliance.

DMZ_VISIBLE When a cache server is root to a public internet root server '1' indicates visibility of internal zones, '0' indicates internal zones are not visible. This flag is not applicable for cache appliances rooted at an internal root server

FIREWALL_TEMPLATE Firewall template name

DESCRIPTION DNS appliance description

TIME_ZONE Time zone

STEALTH_APPLIANCE Applicable for appliances of type 'BIND AUTH'. Accepts 1 or 0 only. '1' indicates that the server can act as a stealth server. This option cannot be enabled when **ENABLE_RECURSION** option is set to 'yes'.

ENABLE_RECURSION Applicable for appliances of type 'BIND AUTH'. 'yes' indicates that the appliance will act as a recursive appliance. This option cannot be enabled when **STEALTH_APPLIANCE** option is set to '1'.

RPZ_TEMPLATE Response policy zone(RPZ) template name. Applicable when **INTERNAL_CACHE** is '0' and **TYPE** is 'BIND CACHE'.

[ntp]

NTP_SERVERS Comma separated list of IP addresses of NTP servers

UPSTREAM To authenticate with the NTP Server, user need to enable Upstream Authentication and fill the following details in the given format.

<IP>-<Key>-<SHA1>,<IP>-<Key>-<SHA1>

Ex: 192.168.0.10-1-zxcvqwer,192.168.0.11-2-asdflkjhg

DOWNSTREAM Comma separated NTP Keys and the sha1, sha1 will auto generate.

if not specified. Keys of downstream should not be same in the upstream authentication key.

<key>-<sha1>,<key>-<sha1>,<key>

EX: 2-262f8ff934271eea15f68b5c7481935e5f00fb9b, 3-595c0bcd44c76232315a9bd6b5cd0de1cd78d40a,5

[dns-resolver]

NAME_APPLIANCES comma separated list of IP addresses of DNS servers. Maximum allowed DNS server is four.

SEARCH_SUFFIXES comma separated list of domain names. Maximum allowed search suffix is six.

[snmp]

TRAP_SINK_1 IP address of SNMP trap sink

TRAP_SINK_2 IP address of SNMP trap sink

COMMUNITY_STRING Community string for SNMP

SYSTEM_LOCATION System Location

SYSTEM_CONTACT System contact

PROCESS_LIST Comma separated list of processes to be monitored. The following is a valid list of processes:

ntpd, dns, bgpd, zebra, crond, sshd, monit, syslog-ng,dhcpd

ENABLE_SNMPV3 Takes 'true' or 'false'. 'true' indicates that SNMPv3 is enabled 'false' indicates that SNMPv3 is disabled.

FIREWALL_SNMP_ACL Name of the SNMP ACL.

[snmpv3]

USER_NAME User name of SNMPv3

AUTHENTICATION_PASSWORD Authentication password of the specified user

APPROVE_PASSWORD Approve password of the specified user

AUTHENTICATION_PROTOCOL Authentication protocol

ENCRYPTION_PROTOCOL Encryption protocol

[ldap-ssh]

ENABLE_LDAP_SSH Takes '0' or '1'. '1' indicates that enable LDAP Authentication on appliance. '0' indicates that disable LDAP Authentication on appliance.

[tacacs]

ENABLE_TACACS Takes '0' or '1'. '1' indicates TACACS+ configuration should be enabled for this server. '0' indicates TACACS+ configuration should be disabled

TACACS_PASSKEY TACACS passkey

TACACS_SERVERS Comma separated list of TACACS servers.

[syslogng-global-options]

TIME_REOPEN The time to wait in seconds before a dead connection is reestablished. Takes a value less than or equal to 32767.

TIME_REAP If no new messages are written to a destination within the specified time in seconds, the connection will be closed, and its state will be freed. Takes a value less than or equal to 32767.

FLUSH_LINES Specifies how many lines are flushed to a destination at a time. Takes a value less than or equal to 32767.

STATS_FREQ Syslog-NG OSE periodically sends a log statistics message. Takes a value less than or equal to 32767.

LOG_FIFO_SIZE The number of messages that the output queue can store. Takes a value less than or equal to 32767.

LOG_MSG_SIZE The maximal length of the log messages is limited by this option It is not recommended to set the option value higher than 10 MiB. Takes a value less than or equal to 32767.

KEEP_TIMESTAMP Specifies whether syslog-NG should accept the timestamp received from the sending application or client. Takes value 'Yes' or 'No'.

[syslogng-source]

SOURCE_NAME Name of the Source

INTERNAL_MSG Internal syslog-NG message, takes input values as '0' or '1'. Default value is '1'.

SYSTEM_MSG System specific log message, takes input value as '0' or '1'.

MSG_TXT_FILE Message from text file, takes the file name as input.

MSG_MULTI_TXT_FILE Message from multiple text files, takes input '0' or '1'. If this flag is '1' need to specify the **FILE_PATH** and **FILE_PATTERN**.

FILE_PATH File patch to the multiple text file.

FILE_PATTERN File Name pattern.

SYSLOG_SERVER Syslog-NG sever, takes the input as '0' or '1'.

IP_ADDRESS IP address of the syslog server.

PORT Port number of the syslog server.

NETWORK_PROTOCOL Network protocol, supports 'UDP' and 'TCP'.

[syslogng-filter]

FILTER_NAME Name of the Filter.

CONDITION Takes the input as 'complex' or 'simple'.

FACILITIES Allow values are one or more comma separated option given below.

auth, authpriv, cron, deamon, kern, lpr, mail,mark, news,
syslog, user, uucp, local0, local1,local2, local3,
local4, local5, local6, local7.

PRIORITIES Allow values are one or more comma separated option given below.

info, notice, warning, err, crit, alert, emerg.

HOST_NAME Name of the host.
IP_NETWORK IP address with mask length.
MATCH_EXPRESSION Match expression.
PROGRAM Program.

[syslogng-destination]

DESTINATION_NAME Name of the destination.
TYPE_SNG Type of the destination. Takes the value between 1 to 5.
'1'= File
'2'= Named pipe
'3'= Local Users
'4'= All logged-in users
'5'= Syslog server.
LOG_FILE_NAME File name to log the message, mandatory when TYPE_SNG is specified as '1'.
NAMED_PIPE_NAME Named pipe name, mandatory when TYPE_SNG is specified as '2'.
LOCAL_USERS Local users, mandatory when TYPE_SNG is specified as '3'.
SYSLOG_SERVER IP address of the syslog server, mandatory when TYPE_SNG is specified as '5'.
PORT Port number of the syslog server, mandatory when TYPE_SNG is specified as '5'.
NETWORK_PROTOCOL Network protocol, supports 'UDP' and 'TCP', mandatory when TYPE_SNG is specified as '5'.

[syslogng-target]

SOURCE Name of the source.
FILTER Name of the filter.
DESTINATION Name of the destination.

[view]

NAMES Comma separated list of DNS view names. Sequence of views are ordered from left to right in ascending order.

[banner]

Banner title of the appliance.

SAMPLE FILE CONTENTS:

[dns-server]

TYPE=BIND AUTH

OPTION_TEMPLATE=testdns

APPLIANCE_TEMPLATE=ISC BIND Authoritative Appliance Template

IP_ADDRESS=10.1.10.201

ORGANIZATION_NAME=TCPWave

APPLIANCE_GROUP=Appliance-Group1,Appliance-Group2

ENABLE_MONIT=1

FIREWALL_TEMPLATE=Default_Firewall

DESCRIPTION=Root

TIME_ZONE=America/New_York (Eastern Time)

[ntp]

NTP_SERVERS=17.253.68.253,17.253.16.243,17.253.80.243,17.253.6.243,

[dns-resolver]

NAME_APPLIANCES=8.8.8.8,8.8.4.4

SEARCH_SUFFIXES=tcpwave.com,demo.tcpwave.com

[snmp]

TRAP_SINK_1=194.41.67.51

TRAP_SINK_2=194.41.65.177

COMMUNITY_STRING=sph1nkx5

SYSTEM_LOCATION=Datacenter for systematic trading infrastructure

SYSTEM_CONTACT=GNCC +1 877 462 2284

PROCESS_LIST=ntpd,dns,bgpd,zebra,crond,

ENABLE_SNMPV3=true

FIREWALL_SNMP_ACL=TestAcl

[snmpv3]

```
USER_NAME=admin
AUTHENTICATION_PASSWORD=zxcv1234
APPROVE_PASSWORD=abc1234567
AUTHENTICATION_PROTOCOL=SHA
ENCRYPTION_PROTOCOL=AES
[ldap-ssh]
ENABLE_LDAP_SSH=1
[tacacs]
ENABLE_TACACS=1
TACACS_PASSKEY=abc123
TACACS_SERVERS=10.1.10.173,10.1.10.172,1.2.3.4,2.3.4.5,
[syslogng-global-options]
TIME_REOPEN=60
TIME_REAP=60
FLUSH_LINES=60
STATS_FREQ=600
LOG_FIFO_SIZE=1000
LOG_MSG_SIZE=2048
KEEP_TIMESTAMP=Yes
```

```
[syslogng-source]
```

```
SOURCE_NAME=s_sys
INTERNAL_MSG=1
SYSTEM_MSG=1
MSG_TXT_FILE=/var/tmp/mft.txt
MSG_MULTI_TXT_FILE=1
FILE_PATH=mfts.txt
FILE_PATTERN=/var/tmp
SYSLOG_SERVER=1
IP_ADDRESS=192.168.0.2
PORT=53
NETWORK_PROTOCOL=UDP
[syslogng-filter]
```

```
FILTER_NAME=f_default
CONDITION=complex
COMPLEX_CONDITION=level(info..emerg) and not (facility(mail) or facility(authpriv) or facility(cron))
```

```
[syslogng-destination]
DESTINATION_NAME=d_mesg
LOG_FILE_NAME=messages
TYPE_SNG=1
ENABLE_SYNC=Yes
[nic]
ADAPTER_NAME=eth0
SPEED=100
DUPLEX=half
AUTO_NEGOTIATION=off
WAKE_ON_LAN=enable
[view]
NAMES=view1,view2
[banner]
```

The default value of Banner configuration for DNS and DHCP appliances.

setipv6dnsserver

NAME

setipv6dnsserver

DESCRIPTION

Creates or updates IPv6 DNS appliance configuration in the TCPWave IPAM.

ARGUMENTS

--action

Takes 'add' or 'edit' for create or update respectively. [mandatory]

--input_file

Full path to the input file from which the appliance configuration is to be read. [mandatory]

EXAMPLE

```
twc setipv6dnsserver --input_file=/tmp/ipv6dnsserver.txt --action=add
```

```
twc setipv6dnsserver --input_file=/tmp/ipv6dnsserver.txt --action=edit
```

FILE FORMAT

The input file format is as follows:

Each configuration section begins with a [<section name>] field followed by <name>=<value> pairs one per line.

SECTIONS & CONFIGURATION PARAMETERS:

[dns-server]

TYPE Takes 'BIND AUTH' or 'BIND CACHE' or 'UNBOUND' or 'NSD' or 'DNS PROXY'

OPTION_TEMPLATE DNS Option template name

APPLIANCE_TEMPLATE DNS appliance template name

IPV6_ADDRESS IPv6 address of the appliance

ORGANIZATION_NAME Organization Name of the DNS appliance

APPLIANCE_GROUP Comma separated names of the appliance groups to be associated

ENABLE_MONIT '0' to enable monitoring and '1' to disable monitoring

INTERNAL_CACHE Applicable for appliances of type 'BIND CACHE' and at an internal root server.'1' indicates that the appliance is rooted at public internet root appliance.

DMZ_VISIBLE When a cache server is root to a public internet root server '1' indicates visibility of internal zones, '0' indicates internal zones are not visible. This flag is not applicable for cache appliances rooted at an internal root server

FIREWALL_TEMPLATE Firewall template name

DESCRIPTION DNS appliance description

TIME_ZONE Time zone

STEALTH_APPLIANCE Applicable for appliances of type 'BIND AUTH'. Accepts 1 or 0 only.'1' indicates that the server can act as a stealth server.This option can not be enabled when

ENABLE_RECUSION option is set to 'yes'.ENABLE_RECUSION Applicable for appliances of type 'BIND AUTH'.

'yes' indicates that the appliance will act as a recursive appliance.

This option Can not be enabled when STEALTH_APPLIANCE option is set to '1'.

RPZ_TEMPLATE Response policy zone(RPZ) template name. Applicable when INTERNAL_CACHE is '0' and TYPE is 'BIND CACHE', 'BIND AUTH + CACHE', UNBOUND.

NSM_TEMPLATE Network Security Monitoring (NSM) template name. Applicable for 'BIND CACHE', 'BIND AUTH + CACHE", UNBOUND appliances.

[ntp]

IPV6_NTP_SERVERS Comma separated list of IPv6 addresses of NTP servers

UPSTREAM To authenticate with the NTP Server, user needs to enable Upstream Authentication and fill the following details in the given format.

<IP>-<Key>-<SHA1>,<IP>-<Key>-<SHA1>

Ex: 5000::2-1-zxcvqwer,3000::-2-asdflkjhg

DOWNSTREAM Comma separated NTP Keys and the sha1, sha1 will auto generate if not specified.

Keys of downstream should not be same in the upstream authentication key.

<key>-<sha1>,<key>-<sha1>,<key>

EX: 2-262f8ff934271eea15f68b5c7481935e5f00fb3b,

3-595c0bcd44c76232315a9bd6b5cd0de1cd78d40a,5

[dns-resolver]

IPV6_NAME_APPLIANCES Comma separated list of IPv6 addresses of DNS servers.

Maximum allowed DNS servers are 4.

SEARCH_SUFFIXES comma separated list of domain names. Maximum allowed search suffixes are 6.

[snmp]

IPV6_TRAP_SINK_1 IPv6 address of SNMP trap sink

IPV6_TRAP_SINK_2 IPv6 address of SNMP trap sink

COMMUNITY_STRING Community string for SNMP

SYSTEM_LOCATION System Location

SYSTEM_CONTACT System contact

PROCESS_LIST Comma separated list of processes to be monitored. The following is a valid list of processes:

ntpd, dns, bgpd, zebra, crond, sshd, monit, syslog-ng,dhcpd

ENABLE_SNMPV3 Takes 'true' or 'false'. 'true' indicates that SNMPv3 is enabled. 'false' indicates that SNMPv3 is disabled.

FIREWALL_SNMP_ACL Name of the SNMP ACL.

[snmpv3]

USER_NAME User name of SNMPv3

AUTHENTICATION_PASSWORD Authentication password of the specified user

APPROVE_PASSWORD Approve password of the specified user

AUTHENTICATION_PROTOCOL Authentication protocol

ENCRYPTION_PROTOCOL Encryption protocol

[ldap-ssh]

ENABLE_LDAP_SSH Takes '0' or '1'. '1' indicates that LDAP Authentication on appliance is enabled.'0' indicates that LDAP Authentication on appliance is disabled.

[tacacs]

ENABLE_TACACS Takes '0' or '1'. '1' indicates TACACS+ configuration is enabled for this server. '0' indicates TACACS+ configuration is disabled

TACACS_PASSKEY TACACS passkey

IPV6_TACACS_SERVERS Comma separated list of TACACS servers.

[syslogng-global-options]

TIME_REOPEN The time to wait in seconds before a dead connection is reestablished. Takes a value less than or equal to 32767.

TIME_REAP If no new messages are written to a destination within the specified time in seconds, the connection will be closed, and its state will be freed. Takes a value less than or equal to 32767.

FLUSH_LINES Specifies how many lines are flushed to a destination at a time. Takes a value less than or equal to 32767.

STATS_FREQ Syslog-NG OSE periodically sends a log statistics message. Takes a value less than or equal to 32767.

LOG_FIFO_SIZE The number of messages that the output queue can store. Takes a value less than or equal to 32767.

LOG_MSG_SIZE The maximal length of the log messages is limited by this option. It is not recommended to set the option value higher than 10 MiB. Takes a value less than or equal to 32767.

KEEP_TIMESTAMP Specifies whether syslog-NG should accept the timestamp received from the sending application or client. Takes value 'Yes' or 'No'.

[syslogng-source]

SOURCE_NAME Name of the Source

INTERNAL_MSG Internal syslog-NG message, takes input values as '0' or '1'.
Default value is '1'.

SYSTEM_MSG System specific log message, takes input value as '0' or
'1'.

MSG_TXT_FILE Message from text file, takes the file name as input.

MSG_MULTI_TXT_FILE Message from multiple text files, takes input '0' or '1'.
If this flag is '1' need to specify the FILE_PATH and
FILE_PATTERN.

FILE_PATH File path to the multiple text files.

FILE_PATTERN File Name pattern.

SYSLOG_SERVER Syslog-NG sever, takes the input as '0' or '1'.

IP_ADDRESS IP address of the syslog server.

PORT Port number of the syslog server.

NETWORK_PROTOCOL Network protocol, supports 'UDP' and 'TCP'.

[syslogng-filter]

FILTER_NAME Name of the Filter.
CONDITION Takes the input as 'complex' or 'simple'.
FACILITIES Allowed values are one or more comma separated options given below.
 auth, authpriv, cron, deamon, kern, lpr, mail,mark, news,
 syslog, user, uucp, local0, local1,local2, local3,
 local4, local5, local6, local7.
PRIORITIES Allowed values are one or more comma separated options given below.
 info, notice, warning, err, crit, alert, emerg.
HOST_NAME Name of the host.
IP_NETWORK IP address with mask length.
MATCH_EXPRESSION Match expression.
PROGRAM Program.

[syslogng-destination]

DESTINATION_NAME Name of the destination.
TYPE_SNG Type of the destination. Takes the value between 1 to 5.
 '1'= File
 '2'= Named pipe
 '3'= Local Users
 '4'= All logged-in users
 '5'= Syslog server.
LOG_FILE_NAME File name to log the message, mandatory when TYPE_SNG is specified as '1'.
NAMED_PIPE_NAME Named pipe name, mandatory when TYPE_SNG is specified as '2'.
LOCAL_USERS Local users, mandatory when TYPE_SNG is specified as '3'.
SYSLOG_SERVER IP address of the syslog server, mandatory when TYPE_SNG is specified as '5'.
PORT Port number of the syslog server, mandatory when TYPE_SNG is specified as '5'.
NETWORK_PROTOCOL Network protocol, supports 'UDP' and 'TCP', mandatory when TYPE_SNG is specified as '5'.

[syslogng-target]

SOURCE Name of the source.
FILTER Name of the filter.
DESTINATION Name of the destination.

[view]

NAMES Comma separated list of DNS views names. Sequence of views are ordered from left to right in ascending order.

[banner]

Banner title of the appliance.

SAMPLE FILE CONTENTS:

[dns-server]

```
TYPE=UNBOUND
OPTION_TEMPLATE=UNBOUND Default Template
APPLIANCE_TEMPLATE=UNBOUND Default Server Template
IPV6_ADDRESS=9000::116
ORGANIZATION_NAME=Internal
FIREWALL_TEMPLATE=
ENABLE_MONIT=1
DESCRIPTION=IPv6 DNS
TIME_ZONE=GMT (GMT)
INTERNAL_CACHE=1
APPLIANCE_GROUP=
NSM_TEMPLATE=
```

[ntp]

```
IPV6_NTP_SERVERS=9000::11
IPV6_UPSTREAM=
DOWNSTREAM=
```

[dns-resolver]

```
IPV6_NAME_APPLIANCES=2345::34
SEARCH_SUFFIXES=
```

[snmp]

```
IPV6_TRAP_SINK_1=
IPV6_TRAP_SINK_2=
COMMUNITY_STRING=
SYSTEM_LOCATION=
SYSTEM_CONTACT=
PROCESS_LIST=
ENABLE_SNMPV3=false
FIREWALL_SNMP_ACL=
```

[ldap-ssh]

```
ENABLE_LDAP_SSH=0
```

[syslogng-global-options]

```
TIME_REOPEN=60
TIME_REAP=60
FLUSH_LINES=60
STATS_FREQ=600
```

```
LOG_FIFO_SIZE=1000
LOG_MSG_SIZE=65536
KEEP_TIMESTAMP=Yes
```

```
[syslogng-source]
```

```
SOURCE_NAME=s_sys
INTERNAL_MSG=1
SYSTEM_MSG=1
```

```
[syslogng-filter]
```

```
FILTER_NAME=f_default
CONDITION=complex
COMPLEX_CONDITION=level(info..emerg) and not (facility(mail) or facility(authpriv) or facility(cron))
```

```
[syslogng-filter]
```

```
FILTER_NAME=f_cron
CONDITION=complex
COMPLEX_CONDITION=facility(cron)
```

```
[syslogng-destination]
```

```
DESTINATION_NAME=d_mesg
TYPE_SNG=1
LOG_FILE_NAME=messages
ENABLE_SYNC=Yes
```

```
[syslogng-destination]
```

```
DESTINATION_NAME=d_cron
TYPE_SNG=1
LOG_FILE_NAME=cron
ENABLE_SYNC=Yes
```

```
[syslogng-target]
```

```
SOURCE=s_sys
FILTER=f_default
DESTINATION=d_mesg
```

```
[syslogng-target]
```

```
SOURCE=s_sys
FILTER=f_cron
DESTINATION=d_cron
```

[view]

NAMES=

[banner]

The default value of Banner configuration for DNS and DHCP appliances.

setdnsservertmpl

NAME

setdnsservertmpl

DESCRIPTION

Creates or updates a DNS appliance template in the TCPWave IPAM.

ARGUMENTS

--name

Name of the DNS appliance template to be added or edited in TCPWave IPAM [mandatory]

--desc

Description of the DNS appliance template to be added or edited in TCPWave IPAM.

--email

Email associated with the corresponding appliance.

--type

DNS appliance type. Takes one of the following values: 'BIND AUTH', 'BIND CACHE', 'UNBOUND', 'NSD', 'DNS PROXY' [mandatory]

--action

Takes 'add' or 'edit'. 'add' creates a new DNS appliance template. 'edit' updates the contents of an existing DNS server template [mandatory]

--dyn_upd

Takes '0' or '1'. '1' indicates dynamic updates has to be enabled. '0' indicates dynamic updates are disabled.

--algo

Comma separated list of algorithm specification as follows:

Algorithm:Bit_size

Algorithm should be one of the valid TSIG algorithms. Bit_size should be between minimum bit size and maximum bit size specified for that

algorithm

Example: "HMAC-SHA1:150,HMAC-SHA256:200"

--logger

Comma separated list of logger specification as follows:

LogCategory/LogChannels

LogChannels is a colon separated list of log channels

Example: "client/default_stderr:default_debug:default_syslog"

--new_name

Field to specify a new name to an existing template.

--dnstap_logs

This is applicable for BIND AUTH and BIND CACHE type appliances. It takes input as '0' or '1'. '1' indicates DNSTAP logs are enabled. '0' indicates DNSTAP logs are disabled.

EXAMPLE

```
twc setdnsservertmpl --name="AUTH-Server-Template" --desc="AUTH Server Template" --email=john.smith@tcpwave.com --type="BIND AUTH" --dyn_upd=1 --algo="HMAC-SHA1:150,HMAC-SHA256:200" --logger="client/default_stderr:default_debug:default_syslog" --action=add
```

```
twc setdnsservertmpl --name="CACHE-Server-Template" --desc="CACHE Server Template" --email=john.smith@tcpwave.com --type="BIND" --logger="client/default_stderr:default_debug:default_syslog" --action=add
```

```
twc setdnsservertmpl --name="UNBOUND-Server-Template" --desc="UNBOUND Server Template" --email=john.smith@tcpwave.com --type="UNBOUND" --action=add
```

```
twc setdnsservertmpl --name="NSD-Server-Template" --desc="NSD Server Template" --email=john.smith@tcpwave.com --type="NSD" --action=add
```

```
twc setdnsservertmpl --name="AUTH-Server-Template" --desc="Base Server Template" --email=john.smith@tcpwave.com --type="BIND AUTH" --dyn_upd=1 --algo="HMAC-SHA1:150,HMAC-SHA256:200" --logger="client/default_stderr:default_debug:default_syslog" --action=edit --new_name="TCPWave-DNS-Server-Template"
```

setfirewalltmpl**NAME:**

setfirewalltmpl

DESCRIPTION:

Creates or updates a firewall template in the TCPWave IPAM. It accepts the input file format as generated by twc getfirewalltmpl. Rules will be applied in the same order as defined in the input file.

ARGUMENTS:

--input_file

Full path to the input file from which the firewall configuration is to be read. [mandatory]

--action

Takes 'add' or 'edit' for create or update respectively. [mandatory]

EXAMPLE USAGE:

```
twc setfirewalltmp1 --input_file=/tmp/firewalltmp1.txt --action=add
```

```
twc setfirewalltmp1 --input_file=/tmp/firewalltmp1.txt --action=edit
```

FILE FORMAT:

The input file format is as follows:

Each configuration section begins with a [**<section name>**] field followed by **<name>=<value>** pairs one per line

SECTIONS & CONFIGURATION PARAMETERS:**[Firewall-Template]**

Name	Name of the firewall template.
New_Name	New name of the firewall template. It is applicable only for edit operation.
Organization	Organization name.
Description	Description for the firewall template.
[Rule]	
Name	Name of the rule.
Action	Action type. It takes one of 'ACCEPT','DROP','REJECT','QUEUE','RETURN' or 'LOG'.
Chain	Chain type. It takes one of 'INPUT','FORWARD' or 'OUTPUT'.
Protocol	Protocol type. It takes one of 'all','tcp','udp','udplite','icmp','esh','ah' or 'sctp'.
Source	Source address. It takes an IPv4 address or IPv4 network address.
Invert_Source	It takes 'true' or 'false'.
Source_Port	Source Port number. It takes an integer value.
Invert_Source_Port	It takes 'true' or 'false'.
Destination	Destination address. It takes an IPv4 address or IPv4 network address.
Invert_Destination	It takes 'true' or 'false'.
Destination_Port	Destination Port number. It takes an integer value.
Invert_Destination_Port	It takes 'true' or 'false'.
Incoming_Interface	Incoming network interface name.
Outgoing_Interface	Outgoing network interface name.
Ethernet_Address	Source Ethernet Address
Fragment	It takes 'true' or 'false'.
DNS	It takes 'true' or 'false'.
DNS_Query	It takes 'true' or 'false'. It is applicable only when DNS parameter value is 'true'.
DNS_Response	It takes 'true' or 'false'. It is applicable only when DNS parameter value is 'true'.
DNS_Query_Type	DNS query type. It takes one of 'ANY','A','NS','CNAME','SOA','PTR','MX','TXT','AAAA','SRV' or 'A6'.
EDNS0	It takes 'true' or 'false'. It is applicable only when DNS parameter value is 'true'.
EDNS0_Buffer_Size	It takes an integer value. It is applicable only when EDNS0 parameter value is 'true'.
Match_String	String-matching filter.
Extension	Extension rule.

SAMPLE FILE CONTENTS:

```
[Firewall-Template]
Name = TCPWave-Firewall-Tmpl
New_Name = TCPWave-Default-Firewall
Organization = TCPWave
Description = TCPWave Default Firewall Template
[Rule]
Name = Default
Action = ACCEPT
Chain = INPUT
Protocol = udp
Source = 192.168.1.4
Invert_Source = false
Source_Port = 1122
Invert_Source_Port = true
Destination =
Invert_Destination = false
Destination_Port =
Invert_Destination_Port = false
Incoming_Interface =
Outgoing_Interface =
Ethernet_Address =
Fragment = false
DNS = true
DNS_Query = true
DNS_Response = false
DNS_Query_Type = A
EDNS0 = true
EDNS0_Buffer_Size = 4321
Match_String =
Extension =
[Rule]
Name =
Action = ACCEPT
Chain = INPUT
Protocol = udp
Source = 10.1.10.0/24
Invert_Source = true
Source_Port =
Invert_Source_Port = false
Destination =
Invert_Destination = false
Destination_Port =
Invert_Destination_Port = false
Incoming_Interface =
Outgoing_Interface =
Ethernet_Address =
Fragment = false
DNS = true
DNS_Query = false
DNS_Response = false
```

```
DNS_Query_Type = ANY
EDNS0 = false
EDNS0_Buffer_Size =
Match_String =
Extension =
```

setipv6dhcpserver

NAME:

```
setipv6dhcpserver
```

DESCRIPTION:

Creates or updates an IPv6 DHCP server configuration in the TCPWave IPAM.

ARGUMENTS:

```
--action
```

Takes 'add' or 'edit' for create or update respectively. [mandatory]

```
--input_file
```

Full path to the input file from which the server configuration is to be read. [mandatory]

EXAMPLE USAGE:

```
twc setipv6dhcpserver --input_file=/tmp/dhcpserver.txt --action=add
```

```
twc setipv6dhcpserver --input_file=/tmp/dhcpserver.txt --action=edit
```

SAMPLE INPUT FILE CONTENTS:

```
[dhcp-server]
IPv6_ADDRESS=2001:db8::4
ORGANIZATION_NAME=TCPWave
POLICY_TEMPLATE=Clone-Policy-1
IPv4_DHCP_APPLIANCE=10.1.10.181
ENABLE_MONIT=1
TIME_ZONE=GMT (GMT)
[ntp]
NTP_SERVERS=10.1.10.10,10.1.10.11,10.1.10.12,10.1.10.13,
[snmp]
TRAP_SINK_1=10.1.10.15
TRAP_SINK_2=10.1.10.18
COMMUNITY_STRING=public
SYSTEM_LOCATION=
SYSTEM_CONTACT=
PROCESS_LIST=dns,
[syslog]
DESTINATION_TYPE=F
DESTINATION=/var/log/messages
ACTIVE=1
SELECTORS=*.info;mail.none;authpriv.none;cron.none
```

```
[syslog]
DESTINATION_TYPE=F
DESTINATION=/var/log/secure
ACTIVE=1
SELECTORS=authpriv.*

[syslog]
DESTINATION_TYPE=F
SYNC_EACH_MESSAGE=1
DESTINATION=/var/log/maillog
ACTIVE=1
SELECTORS=mail.*

[syslog]
DESTINATION_TYPE=F
DESTINATION=/var/log/cron
ACTIVE=1
SELECTORS=cron.*

[syslog]
DESTINATION_TYPE=AU
DESTINATION=
ACTIVE=1
SELECTORS=*.emerg

[syslog]
DESTINATION_TYPE=F
DESTINATION=/var/log/spooler
ACTIVE=1
SELECTORS=uucp,news.crit

[syslog]
DESTINATION_TYPE=F
DESTINATION=/var/log/boot.log
ACTIVE=1
SELECTORS=local7.*
```

```
[macexclusions]
MAC=8E-2C-E7-88-53-7A
DESCRIPTION=
[macexclusions]
MAC=E0:8F:8D:59:CF:60
DESCRIPTION=
```

setipv6dhcptmpl

NAME:

setipv6dhcptmpl

DESCRIPTION:

Creates or updates an IPv6 DHCP option template in the TCPWave IPAM.

ARGUMENTS:

--name

Name of the IPv6 DHCP option template to be retrieved from TCPWave IPAM
[mandatory]

```
--desc
Description of the IPv6 DHCP option template.

--action
Takes 'add' or 'edit'. add creates a new IPv6 DHCP template from the
input file contents. edit updates the contents of an existing DHCP
option template [mandatory]

--new_temp_name
New template name if the name of an existing IPv6 DHCP option template
needs to be updated.

--input_file
Full path to the input file from which the template contents are to be
read [mandatory]

--org
Organization name associated with the option template. This argument is
for users in FADM role to select appropriate organization to which the
operation must be applied. For users not in FADM role, the operation
is by default applied to the organization that the user is associated
with.
```

EXAMPLE USAGE:

```
twc setipv6dhcptmpl --name="voip devices template" --input_file=/tmp/input_dhcptemp.txt --
action=add --org=TCPWave
```

```
twc setipv6dhcptmpl --name="voip devices template" --new_temp_name="voip-devices-
template" --input_file=/tmp/input_dhcptemp.txt --action=edit --org=TCPWave
```

SAMPLE INPUT FILE CONTENTS:

```
CLIENTID = SN000011
SERVERID = SVRNO00055
SIP SERVER D = abc.com
NIS DOMAIN NAME = tcpwave.com
IA NA =
IA TA =
IAADDR =
ORO =
PREFERENCE =
UNICAST =
RAPID COMMIT =
VENDOR OPTS =
INTERFACE ID =
RECONF ACCEPT =
SIP SERVER A =
DNS SERVERS =
DOMAIN LIST =
NIS SERVERS =
NISP SERVERS =
```

NISP DOMAIN NAME =
SNTP SERVERS =
INFORMATION REFRESH TIME =
BCMCS SERVER D =
BCMCS SERVER A =
SUBSCRIBER ID =
CLIENT FQDN = tcpwave1.com

setremotenicsyncoption

NAME:

setremotenicsyncoption

DESCRIPTION:

Sync the provided appliances with the given option.

option- '0' = None

'1' = Sync remote with IPAM

'2' = Sync IPAM with remote

ARGUMENTS:

--option

Option takes the input 0, 1 or 2. '0' indicates None. '1' sync remote with IPAM and '2' sync IPAM with remote. [mandatory]

--appliances

It takes 'All' or comma separated appliances. [mandatory]

EXAMPLE USAGE:

tvc setremotenicsyncoption --option=0 --appliances=10.1.10.230

tvc setremotenicsyncoption --option=1 --appliances=10.1.10.245,192.168.0.123,1.0.0.10

tvc setremotenicsyncoption --option=2 --appliances=All

setremotedebug

NAME

setremotedebug

DESCRIPTION

Updates the remote debugging of a DNS or DHCP appliance from the TCPWave IPAM.

ARGUMENTS

--ip

IP address of the DNS or DHCP appliance. [mandatory]

--enable_debug

Enable or disable the remote debugging on DNS or DHCP appliance. It takes 'yes' or 'no' as an input. [mandatory]

--type

Type of the appliance. It takes 'DHCP' or 'DNS' as input. [mandatory]

EXAMPLE

```
twc setremotedebug --ip=10.1.10.24 --enable_debug=yes --type=DNS
```

```
twc setremotedebug --ip=10.1.10.25 --enable_debug=no --type=DHCP
```

setipv6remotedebug

NAME

setipv6remotedebug

DESCRIPTION

Updates the remote debugging Status of an IPv6 DNS or DHCP appliance from the TCPWave IPAM.

ARGUMENTS

--ip

The IP address of the IPv6 DNS or DHCP appliance. [mandatory]

--enable_debug

Enable or disable the remote debugging status on IPv6 DNS or DHCP appliance. It takes 'Yes' or 'No' as input. [mandatory]

--type

Type of the appliance. It takes 'DHCP' or 'DNS' as input. [mandatory]

EXAMPLE

```
twc setipv6remotedebug --ip=2001:1::2 --enable_debug=Yes --type=DNS
```

```
twc setipv6remotedebug --ip=2001:1::2 --enable_debug=No --type=DHCP
```

setipv6remotectrllog

NAME

setipv6remotectrllog

DESCRIPTION

Updates the remote central logging status of an IPv6 DNS or DHCP appliance from the TCPWave IPAM.

ARGUMENTS

--ip

The IP address of the IPv6 DNS or DHCP appliance [mandatory].

--enable_ctrl_log

Enable or disable the remote central logging status on IPv6 DNS or DHCP

appliance. It takes 'Yes' or 'No' as input [mandatory].

--type

Type of the appliance. It takes 'DHCP' or 'DNS' as input [mandatory].

EXAMPLE

```
twc setipv6remotecntrllog --ip=2001:1::2 --enable_central_log=Yes --type=DNS
```

```
twc setipv6remotecntrllog --ip=2001:1::2 --enable_central_log=No --type=DHCP
```

setsubnetmpl

NAME

setsubnetmpl

DESCRIPTION

Creates or updates a subnet template in the TCPWave IPAM.

ARGUMENTS

--action

Takes 'add' or 'edit' to create or update respectively.

--name

Name of the subnet template.

--new_name

New name of the subnet template.

--org

Name of the organization.

--input_file

Subnet template key value pair configuration absolute file path.

EXAMPLE

```
twc setsubnetmpl --name=subnetmpl --org=TCPWave --input_file=/tmp/subnetmpl.txt --action=add
```

```
twc setsubnetmpl --name=subnetmpl --org=TCPWave --new_name=template --input_file=/tmp/subnetmpl.txt --action=edit
```

FILE FORMAT

The input file format is as follows:

Each configuration section begins with a [<section name>] field followed by <name>=<value> pairs one per line.

SECTIONS & CONFIGURATION PARAMETERS

[subnet-template]

PRIMARY_DOMAIN Name of the domain to be associated with subnet.
SUBNET_TYPE Takes Non-DHCP, DHCP-Enabled, Cloud Hosted subnet type.
PRIMARY_ROUTER Takes first, last or offset as primary router.
DHCP_TEMPLATE Name of the DHCP template.
PRIMARY_DHCP_APPLIANCE Primary DHCP appliance.
CLOUD_PROVIDER Name of the cloud provider.
OFFSET_FOR_PRIMARY_ROUTER Integer value of an offset.
CONTACT_FIRST_NAME First name of the contact.
CONTACT_MIDDLE_NAME Middle name of the contact.
CONTACT_LAST_NAME Last name of the contact.
CONTACT_EMAIL_ID Email address of the contact.
STREET1 Street1 of the location.
STREET2 Street2 of the location.
CITY Name of the city for location.
STATE Name of the state for location.
COUNTRY Name of the country for location.
ZIP Zip code of the location.
SECONDARY_DOMAINS Name of the secondary domains to be associated with subnet.

[address-range]

DOMAIN Name of the domain to be associated with objects within subnet.
ALLOCATION_TYPE Takes Static, Dynamic, Reserves allocation type.
OBJECT_TYPE Takes the object type like 3G Phone, Access Router, Audio MCU etc..
TIME_TO_LIVE Takes the integer in seconds.
START_OFFSET Takes integer value.
END_OFFSET Takes integer value.

SAMPLE FILE CONTENTS:

[subnet-template]

PRIMARY_DOMAIN=tcp.com

SUBNET_TYPE=Non-DHCP

PRIMARY_ROUTER=first

DHCP_TEMPLATE =
PRIMARY_DHCP_APPLIANCE=
CLOUD_PROVIDER=
OFFSET_FOR_PRIMARY_ROUTER=
CONTACT_FIRST_NAME=
CONTACT_MIDDLE_NAME=
CONTACT_LAST_NAME=
CONTACT_GMAIL_ID=
STREET1=
STREET2=
CITY=
STATE=
COUNTRY=
ZIP=
DESCRIPTION=
SECONDARY_DOMAINS=tcpwave.com,tcpzone.com
[address-range]
DOMAIN=tcp.com
ALLOCATION_TYPE=Static
OBJECT_TYPE=3G Phone
TIME_TO_LIVE=1200
START_OFFSET=11
END_OFFSET=20
[address-range]
DOMAIN=test.com
ALLOCATION_TYPE=Dynamic
OBJECT_TYPE=AWS Instance
TIME_TO_LIVE=1200
START_OFFSET=21
END_OFFSET=30
[address-range]

```
DOMAIN=tcp.com
ALLOCATION_TYPE=Reserved
OBJECT_TYPE=PC
TIME_TO_LIVE=1200
START_OFFSET=1
END_OFFSET=10
EXPIRY_DATE=2015-03-31
```

[address-range]

```
DOMAIN=tcp.com
ALLOCATION_TYPE=Static
```

```
OBJECT_TYPE=Laptop
TIME_TO_LIVE=1200
START_OFFSET=40
END_OFFSET=50
```

setipv6subnetmpl

NAME

setipv6subnetmpl

DESCRIPTION

Creates or updates an IPv6 subnet template in the TCPWave IPAM.

ARGUMENTS

--action

Takes 'add' or 'edit' to create or update respectively [mandatory].

--name

Name of the IPv6 subnet template [mandatory].

--new_name

The new name of the IPv6 subnet template.

--org

Name of the organization [mandatory].

--input_file

Subnet template key-value pair configuration absolute file path [mandatory].

EXAMPLE

```
twc setipv6subnetmpl --name=subnetmpl --org=TCPWave --input_file=/tmp/subnettemplate.txt --action=add
```

```
twc setipv6subnetmpl --name=subnetmpl --org=TCPWave --new_name=template --input_file=/tmp/subnettemplate.txt --action=edit
```

FILE FORMAT

The input file format is as follows:

Each configuration section begins with a [<section name>] field followed by <name>=<value> pairs one per line.

SECTIONS & CONFIGURATION PARAMETERS:

[subnet-template]

PRIMARY_DOMAIN Name of the domain to be associated with IPv6 subnet.

SUBNET_TYPE Takes Non-DHCP, DHCP-Enabled, Cloud Hosted IPv6 subnet type.

PRIMARY_ROUTER Takes first, last or offset as the primary router.

DHCP_TEMPLATE Name of the DHCP template.

IPV6_PRIMARY_DHCP_APPLIANCE Name of the Primary DHCP appliance.

OFFSET_FOR_PRIMARY_ROUTER Integer value of an offset.

[address-range]

DOMAIN Name of the domain to be associated with objects within IPv6 subnet.

ALLOCATION_TYPE Takes Static, Dynamic, Reserves allocation types.

OBJECT_TYPE Takes the object type like 3G Phone, Access Router, Audio MCU etc.

TIME_TO_LIVE Takes the integer in seconds.

IPV6_START_OFFSET Takes an integer value.

IPV6_END_OFFSET Takes integer value.

SAMPLE FILE CONTENTS:

[subnet-template]

PRIMARY_DOMAIN=tcpwave.com

PRIMARY_ROUTER=first

SUBNET_TYPE=Non-DHCP

DHCP_TEMPLATE=

PRIMARY_DHCP_APPLIANCE=

OFFSET_FOR_PRIMARY_ROUTER=

DESCRIPTION=

```
[address-range]
DOMAIN=tcpwave.com
OBJECT_TYPE=3G Phone
TIME_TO_LIVE=1200
START_OFFSET=11
END_OFFSET=20
ALLOCATION_TYPE=Static
```

[address-range]

```
DOMAIN=tcpwave.com
OBJECT_TYPE=AWS Instance
TIME_TO_LIVE=1200
START_OFFSET=21
END_OFFSET=30
ALLOCATION_TYPE=Static
```

setupdr**NAME:**

setupdr

DESCRIPTION:

Setup disaster recovery configuration in the TCPWave IPAM.

ARGUMENTS:

```
--master_ip
    IP address of the primary IPAM. [mandatory]
--master_port
    Port number of the primary IPAM. [mandatory]

--slave_ip
    IP address of the secondary IPAM. [mandatory]
--slave_port
    Port number of the secondary IPAM. [mandatory]

--flag
    Force setup flag to configure disaster recovery.
```

EXAMPLE USAGE:

```
twc setupdr --master_ip=10.1.10.60 --master_port=7443 --slave_ip=10.1.10.61 --slave_port=7443 -
-flag=1
```

showconnected**NAME:**

showconnected

DESCRIPTION:

Displays all the connections that are listening on port 61617 in the TCPWave IPAM.
TCP port 61617 is used to communicate with TCPWave remotes over SSL.
The list includes the following information.
"Proto", "Recv-Q", "Send-Q", "Local Address", "Foreign Address", and "State"

ARGUMENTS:

- NA

EXAMPLE USAGE:

twc showconnected

SAMPLE OUTPUT:

Proto	Recv-Q	Send-Q	Local Address	Foreign Address	State
tcp	0	0	10.1.10.240:61617	0.0.0.0:*	LISTEN
tcp	0	0	10.1.10.240:61617	10.1.10.185:36482	ESTABLISHED
tcp	0	0	10.1.10.240:61617	172.171.98.71:56754	ESTABLISHED
tcp	0	0	10.1.10.240:61617	10.1.10.182:58350	ESTABLISHED
tcp	0	0	10.1.10.240:61617	10.1.10.181:42192	ESTABLISHED
tcp	0	0	10.1.10.240:61617	10.1.10.126:54702	ESTABLISHED
tcp	0	0	10.1.10.240:61617	10.1.10.184:58834	ESTABLISHED
tcp	0	0	10.1.10.240:61617	10.1.10.181:42194	ESTABLISHED
tcp	0	0	10.1.10.240:61617	172.193.1.50:60510	ESTABLISHED
tcp	0	0	10.1.10.240:61617	10.1.10.181:42128	ESTABLISHED
tcp	0	0	10.1.10.240:61617	172.171.98.71:56760	ESTABLISHED
tcp	0	0	10.1.10.240:61617	10.1.10.185:50570	ESTABLISHED
tcp	0	0	10.1.10.240:61617	172.193.1.50:60466	ESTABLISHED
tcp	0	0	10.1.10.240:61617	10.1.10.180:53922	ESTABLISHED
tcp	0	0	10.1.10.240:61617	10.1.10.184:45236	ESTABLISHED
tcp	0	0	10.1.10.240:51424	10.1.10.240:61617	ESTABLISHED

showdefaultroute

NAME:

showdefaultroute

DESCRIPTION:

Displays the default gateway settings for TCPWave IPAM.
This command displays the following information.
"Interface of the default gateway", "IPv4 address of the default gateway"
and "IPv6 address of the default gateway" (If configured).

ARGUMENTS:

EXAMPLE USAGE:

twc showdefaultroute

SAMPLE OUTPUT:

Gateway for the default route: eth0
IPv4 default Gateway: 10.1.10.1

showdevices

NAME:

showdevices

DESCRIPTION:

Shows a list of the discovered objects on a subnet in the TCPWave IPAM.
The list includes the following information.
"IP Address","Device Name","Device Name In IPAM","MAC","MAC in IPAM",
"Latency","Ports","Vendor","OS","Contact","Location" and "Description"

ARGUMENTS:

--discovery_id

Command Id of the discovered subnet. Use the following command to see all command IDs of the discovered subnets: 'twc listdiscovertask --d=,'.
[mandatory]

--d

Delimiter character separating the columns. If this argument is not specified, then comma will be used as default delimiter.

EXAMPLE USAGE:

twc showdevices --discovery_id=1391 --d=,

showdrconfig

NAME:

showdrconfig

DESCRIPTION:

Display the disaster recovery configuration in the TCPWave IPAM.

ARGUMENTS:

- NA

EXAMPLE USAGE:

twc showdrconfig

showjobexehistory

NAME:

showjobexehistory

DESCRIPTION:

Display the scheduled job execution history in the TCPWave IPAM.

ARGUMENTS:

--job_id

Id of the scheduled job. [mandatory]

--count

Number of lines to be displayed. If this argument is not specified, 100 lines will be displayed.

--d

Delimiter character separating the columns. If this argument is not specified, then comma will be used as a delimiter.

EXAMPLE USAGE:

twc showjobexehistory --job_id=ScheduledJobId --count=50 --d=
showlicense

NAME:

showlicense

DESCRIPTION:

Shows the license information of the TCPWave IPAM.

ARGUMENTS:

EXAMPLE USAGE:

twc showlicense
showsecuritylog

NAME:

showsecuritylog

DESCRIPTION:

Displays the security logs in CEF format from the TCPWave IPAM.

ARGUMENTS:

--output_file

Full path to the output file to which security logs are to be written.
If the file path is not specified, the output is written to the standard output.

EXAMPLE USAGE:

twc showsecuritylog --output_file=/tmp/securitylog.txt
splitsubnet

NAME:

splitsubnet

DESCRIPTION:

Splits a given subnet in the TCPWave IPAM.

ARGUMENTS:

--subnet
IP address of the subnet to be split [mandatory]

--mask
Mask length of the resultant subnets [mandatory]

--router_opt
Takes 'first' or 'last'
'first' indicates that the router IP Address will be the first address of the subnet address range. 'last' indicates that the router IP Address will be the last address of the subnet address range.

--org
Name of the organization to which subnet belongs. This argument is mandatory if user is 'FADM'

EXAMPLE USAGE:

```
twc splitsubnet --subnet=80.0.0.0 --mask=26 --router_opt=first --org=TCPWave  
syncdhcpserver
```

NAME:

syncdhcpserver

DESCRIPTION:

Performs a DHCP Server configuration sync with the TCPWave IPAM.

ARGUMENTS:

--dhcp_appliance
IP Address of the server. Should be a valid object in TCPWave IPAM [mandatory]

--org
Organization name associated with the DHCP server. [mandatory]

--all
Flag to sync all DHCP server. Takes '0' and '1'. If it is specified, as '1' all DHCP server are synchronized.

EXAMPLE USAGE:

```
twc syncdhcpserver --dhcp_appliance=10.1.10.180 --org=TCPWave
```

```
twc syncdhcpserver --org=TCPWave --all=1
```

syncipv6dhcpserver

NAME

syncipv6dhcpserver

DESCRIPTION

Performs an IPv6 DHCP appliance configuration sync with the TCPWave IPAM.

ARGUMENTS

--ip

IPv6 Address of the DHCP appliance. Should be a valid object in TCPWave IPAM [mandatory]

--org

Organization name associated with the DHCP appliance. [mandatory]

--all

Flag to sync all DHCP appliance. Takes '0' and '1'. If it is specified as '1' all DHCP appliances are synchronized.

EXAMPLE

```
twc syncipv6dhcpserver --ip=5000::2 --org=TCPWave
```

```
twc syncipv6dhcpserver --org=TCPWave --all=1
```

syncdnsserver

NAME:

syncdnsserver

DESCRIPTION:

Performs a DNS Server full configuration sync with the TCPWave IPAM.

ARGUMENTS:

--appliance_ip

IP Address of the server. Should be a valid object in TCPWave IPAM. This argument is mandatory if --all argument is not specified, or specified, as '0'.

--appliance_type

DNS server type. Takes one of the following values: 'BIND AUTH', 'BIND CACHE', 'UNBOUND' or 'DNS PROXY' [mandatory]

--org

Organization name associated with the DNS server [mandatory]

--all

Flag to sync all DNS server of specified, type. Takes '0' and '1'. If it is specified, as '1' all DNS server of specified, type are synchronized.

EXAMPLE USAGE:

```
twc syncdnsserver --appliance_ip=9.0.0.1 --appliance_type="BIND AUTH" --org=TCPWave
```

```
twc syncdnsserver --appliance_type="BIND AUTH" --org=TCPWave --all=1
```

syncipv6dnsserver

NAME

syncipv6dnsserver

DESCRIPTION

Performs a IPv6 DNS appliance full configuration sync with the TCPWave IPAM.

ARGUMENTS

--ip

IP Address of the appliance. Should be a valid object in TCPWave IPAM. This argument is mandatory if --all argument is not specified or specified as '0'.

--appliance_type

DNS appliance type. Takes one of the following values: 'BIND AUTH', 'BIND CACHE', 'UNBOUND' or 'DNS PROXY' [mandatory]

--org

Organization name associated with the DNS appliance [mandatory]

--all

Flag to sync all DNS appliance of specified type. Takes '0' and '1'. If it is specified as '1' all DNS appliance of specified type are synchronized.

EXAMPLE

```
twc syncipv6dnsserver --ip=5000::2 --appliance_type="BIND AUTH" --org=TCPWave
```

```
twc syncipv6dnsserver --appliance_type="BIND AUTH" --org=TCPWave --all=1
```

syncmicrosoftdhcpserver

NAME:

syncmicrosoftdhcpserver

DESCRIPTION:

Performs a configuration sync on Microsoft DHCP appliance on the TCPWave IPAM.

ARGUMENTS:

--addr

IP address of the appliance. [mandatory]

--org

Name of the organization. [mandatory]

EXAMPLE USAGE:

```
twc syncmicrosoftdhcpserver --addr=10.0.0.10 --org=TCPWave
```

syncmicrosoftdnsserver

NAME:

syncmicrosoftdnsserver

DESCRIPTION:

Performs a configuration sync on Microsoft DHCP appliance on the TCPWave IPAM.

ARGUMENTS:

--addr

IP address of the appliance. [mandatory]

--org

Name of the organization. [mandatory]

EXAMPLE USAGE:

```
twc syncmicrosoftdnsserver --addr=10.0.0.10 --org=TCPWave
```

syncobject

NAME:

syncobject

DESCRIPTION:

Synchronizes an object with master and slave DNS servers in the TCPWave IPAM.

ARGUMENTS:

--object

IP Address of the object to be synchronized. [mandatory]

EXAMPLE USAGE:

```
twc syncobject --object=172.16.2.4 --org=TCPWave
```

synczone**NAME**

synczone

DESCRIPTION

Performs a DNS zone force sync on the TCPWave managed DNS appliances.

ARGUMENTS

--zone_name

Name of the DNS Zone [mandatory]

--org

Organization name associated with the DNS zone [mandatory]

--is_proxy

DNS Proxy root zone flag. It takes '0' or '1'. If it is specified as '1' zone will be considered as proxy root zone. If it is specified as '0' zone will be considered as root zone. Default value is '0'.

EXAMPLE

```
twc synczone --zone_name="tcpwave.com" --org=TCPWave
```

```
twc synczone --zone_name="." --org=TCPWave --is_proxy=0
```

```
twc synczone --zone_name="." --org=TCPWave --is_proxy=1
```

```
twc synczone --zone_name="0-10.1.in-addr.arpa" --org=TCPWave
```

syncactivelease**NAME:**

syncactivelease

DESCRIPTION:

Sync active leases of the DHCP appliance or Subnet.

ARGUMENTS:

--ip

IP address of the appliance. [mandatory]

--dhcp_type

Type of the dhcp appliance. It takes the input as 'DHCP' or 'MSDHCP'. [mandatory]

--org

Name of the organization. [mandatory]

--type

Element type, it takes the input as 'appliance' or 'subnet'. [mandatory]

--subnet

IP address of the subnet, mandatory when the type is specified as subnet.

EXAMPLE USAGE:

```
twc syncactivelease --ip=192.168.0.241 --dhcp_type=dhcp --org=TcpWave --type=appliance
```

```
twc syncactivelease --ip=192.168.0.241 --subnet=192.168.0.0 --dhcp_type=dhcp --org=TcpWave --type=subnet
```

undolist**NAME:**

undolist

DESCRIPTION:

Lists all the objects from the recycle bin that can be restored in the TCPWave IPAM.

ARGUMENTS:

--d

Delimiter character separating the columns. If this argument is not specified, then comma will be used as a delimiter.

EXAMPLE USAGE:

```
twc undolist --d=,
```

updatelicense**NAME:**

updatelicense

DESCRIPTION:

Updates the license with the license key provided by TCPWave Technical Support Team.

ARGUMENTS:

--license_key

License key need to update the IPAM. [mandatory]

EXAMPLE USAGE:

```
twc updatelicense --  
license_key=8B84A0F3DEC6B73AFD49414256CDAAB49D56E573FFB629D4572FF200E6C59361
```

undoipamenti

NAME:

undoipamenti

DESCRIPTION:

Undo an Add, Edit and Delete operation from the recycle bin
in the TCPWave IPAM.

ARGUMENTS:

--entity_type

Type of the entity. Defines the context in which the undo operation has
to be applied. Takes 'object', 'zone', 'scope','revzone' or subnet.
[mandatory]

--entity_name

Name of the entity. Value should be an IP address when entity_type is
object or scope.
When entity_type is subnet, value should be an IP address with mask
length.
Value should be a domain name when entity_type is zone or rezone.
[mandatory]

--operation

Name of the operation. Takes add, delete or edit. [mandatory]

--org

Name of the organization entity belongs. This argument is mandatory if
the user is FADM. For users not in FADM role, the operation is applied
to the organization that the user is associated with.

EXAMPLE USAGE:

```
twc undoipamenti --entity_name=10.0.3.16/28 --entity_type=subnet --operation=delete --  
org=TCPWave
```

```
twc undoipamenti --entity_name=10.0.0.15 --entity_type=object --operation=add --  
org=TCPWave
```

```
twc undoipamenti --entity_name=32-28.32.1.0.10.in-addr.arpa --entity_type=revzone --  
operation=add --org=TCPWave
```

```
twc undoipamenti --entity_name=[10.0.0.4-10.0.0.8] --entity_type=scope --operation=add --  
org=TCPWave
```

```
twc undoipamentity --entity_name=test.com --entity_type=zone --operation=add --  
org=TCPWave
```

uploadpatch**NAME:**

```
uploadpatch
```

DESCRIPTION:

Upload the specified, patch to the TCPWave IPAM.

ARGUMENTS:

```
--patch_file  
File name of the patch. [mandatory]
```

EXAMPLE USAGE:

```
twc uploadpatch --patch_file=/tmp/Patch_IPAM_11.27_2_6301.zip
```

zoneaxfrtotims**NAME:**

```
zoneaxfrtotims
```

DESCRIPTION:

Convert the DNS resource records of a zone hosted by specified, name server into TCPWave DNS hosted resource record in the TCPWave IPAM. Rules of conversion are as follow :

1. All NS resource records of the hosted zone will not be converted into TCPWave DNS hosted resource record.
2. If 'A' resource record with same IP address appears more than once in the hosted DNS zone, first one will be converted into a TCPWave object having object type 'Others' and others will be converted into object resource record in the TCPWave IPAM.
2. Remaining all resource records will be converted into zone type resource records in the TCPWave IPAM.

ARGUMENTS:

```
--name_server  
IP address or domain name of the name server from which specified, zone  
transfer must be done. [mandatory]
```



```
--zone  
DNS zone which is to be transfer from specified, name server. [mandatory]
```



```
--org  
Organization name to which specified, zone has been created in the TCPwave  
IPAM. [mandatory]
```

EXAMPLE USAGE:

```
twc zoneaxfrtotims --name_server=10.1.10.180 --zone=somezone.com --org=TCPWave  
twc zoneaxfrtotims --name_server=ns.nameserver.com --zone=somezone.com --org=TCPWave
```

pauseschedjob

NAME:

Pauseschedjob

Pause the execution of a scheduled job in the TCPWave IPAM.

DESCRIPTION:

Pause the execution of a scheduled job in the TCPWave IPAM.

ARGUMENTS:

--job_id

Id of the scheduled job. [mandatory]

EXAMPLE:

```
twc pauseschedjob --job_id=RemoteMonitStatsOperation
```

resumeschedjob

NAME:

resumeschedjob

resumeschedjob - Resume the execution of a scheduled job in the TCPWave IPAM.

DESCRIPTION:

Resume the execution of a scheduled job in the TCPWave IPAM.

ARGUMENTS:

--job_id

Id of the scheduled job. [mandatory]

EXAMPLE:

```
twc resumeschedjob --job_id=RemoteMonitStatsOperation
```

setipamappliance

NAME:

setipamappliance - Creates or updates IPAM appliance configuration in the TCPWave IPAM.

DESCRIPTION:

Creates or updates IPAM appliance configuration in the TCPWave IPAM.

ARGUMENTS:

--action

Takes 'add' or 'edit' for create or update, respectively.

--input_file

Full path to the input file from which the appliance configuration is to be read.

EXAMPLE:

```
twc setipamappliance --input_file=/tmp/ipamappliance.txt --action=add
```

```
twc setipamappliance --input_file=/tmp/ipamappliance.txt --action=edit
```

