TCPWave IPAM

Managing TCPWave IPAM Using Command Line Interface

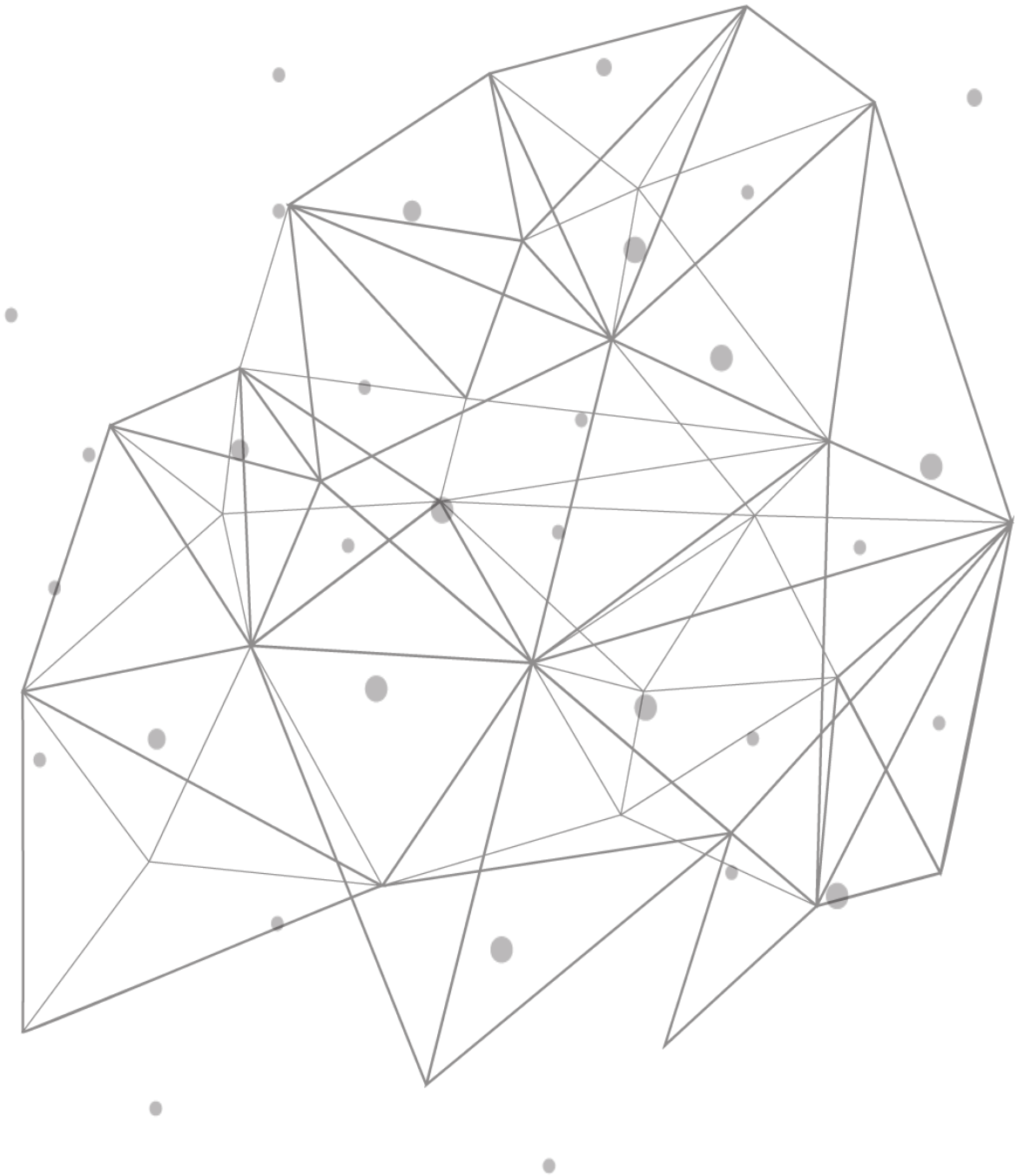


Table of Contents

Introduction

A graphical user interface (GUI) is a common type of interface used for interacting with systems. TCPWave's friendly GUI enables network administrators to perform all DNS, DHCP, and IP Address Management (DDI) activities. However, many administrators prefer using command-line interface (CLI) commands, which are often faster and more efficient than using other types of interfaces. TCPWave provides about 500 CLI commands to manage DDI activities.

TCPWave's CLI Framework

TCPWave's CLI framework is well-regarded and provides the basis to run commands and automate tasks in your DDI infrastructure. Scripts, such as shell scripts, enable you to customize functionality and automate processes you may want to repeat. For example, rather than entering multiple commands to create various branch office VLANs with DHCP associations and DNS entries, an administrator can use a script that takes arguments and runs multiple commands to create all needed DDI entries. Also, CLI commands can be used to quickly perform bulk operations not possible in the GUI.

Leverage The Power Of CLI

As a network administrator or IT decision-maker, you will find that integrating TCPWave's CLI framework in automation helps you improve efficiency and reduce operational costs. Also, TCPWave CLI commands can interoperate with systems such as OpenStack, ServiceNow, Ansible, and Splunk to reduce manual tasks in DNS and DHCP operations.

In each release, TCPWave continues to improve the CLI functionality available by providing new CLI commands.

Managing TCPWave IPAM Using Command Line Interface

In TCPWave IPAM, the term Organization refers to a separate workspace independent of each other that allows you to manage your DDI activities effectively.

> All commands are executed only when you use 'twc.' At any point in time, if you want to know additional functions available with a command, you can execute '**twc <command> help.**'

Here are some sample commands that help you accomplish daily operations in your TCPWave IPAM.

Sample CLI commands

CLI Command	Description	Tasks that you can Achieve
addorg	In the TCPWave IPAM organization is used to maintain users and networks. Each organization can be associated with multiple networks.	You can segregate the DDI activities of one department of your company without interfering with other departments' daily operations. Control the user privileges for each organization so that you don't have to manage too many permissions for a user.
addnetwork	It creates a network in the TCPWave IPAM	It's essential to create a network to define objects like TCPWave Remote Appliance and Microsoft Appliance to add the DNS, DHCP server, and other network devices.
adddomain	It creates a DNS domain in the TCPWave IPAM.	For all networks, the domain is n is part of every network address, including website addresses, email addresses. It helps you associate your networks with the specific domain you chose.
addsubnet	It creates a subnet in the TCPWave the IPAM.	You need a subnet to create network objects for network devices
addobject	This command is used to add an object in the TCPWave IPAM	You can define objects of network devices like TCPWave and Microsoft appliances, routers, and other network devices, also define DNS, DHCP parameters of each object
setdnsservertmpl	It creates or updates a DNS appliance template in the TCPWave IPAM.	Define DNS appliance-specific parameters like type, dynamic update, logging, etc.
setdnsopttmpl	It creates or updates a DNS option template in the TCPWave IPAM.	The option template input file will contain parameters like recursion, zone transfer, DNSSEC, TTL, etc.
setdnsserver	It creates or updates DNS appliance configuration in the TCPWave IPAM.	DNS server input file will have parameters recursion, dns resolver, logging, monitoring, etc.
adddnszonetmpl	It creates a DNS zone template in the TCPWave IPAM.	Define parameters like refresh, retry, expiry, negative cache, forwarders, etc.
adddnszone	It creates a DNS Zone in the TCPWave IPAM.	Create zone based on the template defined and parameters DNSSEC, allow update parameters, etc.
addobject	This command is used to add an	Create objects again target IP address,

CLI Command	Description	Tasks that you can Achieve
	object in the TCPWave IPAM	with parameters like name, allocation type, domain, ttl, mac address, location, etc.
editdnsreversezone	It updates a DNS reverse zone in the TCPWave IPAM	If you want to edit the zone and associate it with a zone template, edit parameters like ttl, DNSSEC, extended attributes, contact, etc
addrr	Creates a DNS resource record in 'object,' 'zone' or 'network' scopes	Enables you to create resource records at the Object level, in the zone, and in scopes, with parameters like type, class, ttl, record specific information for all types of records like CNAME, SRV record, CNAME, MX record, etc.
syncdnsserver	It performs a DNS Server full configuration sync with the TCPWave IPAM.	Synchronize the DNS server configuration with IPAM in case of any changes are made or records are added, modified, or deleted. It will also allow specific types of DNS servers to be synced together.
editglobalopts	Updates a Global Option in the TCPWave IPAM.	You can edit the global parameters defined under Global policy management like enable debugging, DNS and DHCP global options, etc.
search	Performs a search operation in the TCPWave IPAM.	You can perform a search with full text, wildcard search, regex search and match the search with matching phrases.
setdhcppolicytmpl	Creates or updates a DHCP policy template in the TCPWave IPAM.	In the input file of the policy template, you can define parameters like max lease time, bootp, default lease time, one lease per client, etc.
Setdhcptmpl	Creates or updates DHCP option template in the TCPWave IPAM.	The Input file will have parameters like User authentication protocol options, TCP parameters, vendor options, log server, domain server, DHCP extensions, etc.
setdhcpserver	It creates or updates the DHCP server configuration in the TCPWave IPAM.	The DHCP server will be added with the parameters defined in the input file, such as policy template, NTP, DNS resolvers, monitoring, etc.
adddhcpfailoverpeer	It creates a DHCP failover peer in the TCPWave IPAM.	Need to add DHCP server to form failover and then add dhcp failover peer with the parameters like failover port, max

CLI Command	Description	Tasks that you can Achieve
		response delay, MCLT, split, max unacked updates, etc.
editsubnet	It updates a subnet in the TCPWave IPAM.	This allows us to edit the subnet parameters like domain name, type, subnet template, extended attributes, location, contact, etc.
addscope	Add a DHCP scope specified by --start_ip and --end_ip arguments in the TCPWave IPAM.	Create DHCP scope with parameters like ttl, object type, DDNS enabled or not, allow or deny vendor, client, etc.
addadmin	Creates an administrator in the TCPWave IPAM.	The user is identified uniquely by the combination of First Name, Last Name, and Email Id. In addition, the administrator group and role can be added.
addadminpermission	Defines an administrator/administrator group permission in the TCPWave IPAM.	You can give read, write, or permissions to different entities like networks, subnets, zones, objects, etc.
exescheduledjob	It executes a scheduled job in the TCPWave IPAM.	You can immediately execute scheduled jobs to run before the scheduled time
addcontact	Create a contact for a given organization in the TCPWave IPAM.	You may wish to have contacts added for different subnets, zones, or objects based on location or responsibility given.
addlocation	It creates a location for a given organization in the TCPWave IPAM.	You may wish to have contacts added for different subnets, zones, or objects based on location or responsibility given.
editsubnet	It updates a subnet in the TCPWave IPAM.	Edit the parameters like location, contact, type, subnet group, router address, subnet template, dhcp appliance, etc.
addext	It creates an extended attribute in the TCPWave IPAM	Define extended attribute values by type such as ‘string,’ ‘date,’ ‘numeric,’ ‘ip’ or ‘list,’ list values, etc., to different entities as admin, network, subnet, object, zone.
rptdnssoa	It generates a DNS SOA report to a specified file in the TCPWave IPAM.	You can get the report into output files in CSV or PDF format.
rptv4netspaceutil	It generates the IPv4 network space utilization audit report in a specified file in TCPWave IPAM	You can get the report into output files in CSV or PDF format.
uploadpatch	It uploads the specified patch to	You need to mention the location of the

CLI Command	Description	Tasks that you can Achieve
	the TCPWave IPAM.	patch file it is uploaded; the/tmp file will have read access for the twcadm account.
deploypatch	Deploy a patch in the TCPWave IPAM.	The uploaded patch can only be deployed in the TCPWave IPAM. Give the details like file name, package name, major version, minor version, appliance type, and patch level. You can also schedule the patch deployment.

Please refer to the steps demonstrated below to understand how to use these sample commands.

Adding Organization

As per your changing business requirements, to segregate the DDI management activities, you have the privilege to create a separate organization using TCPWave IPAM CLI.

Command

addorg

Example Usage

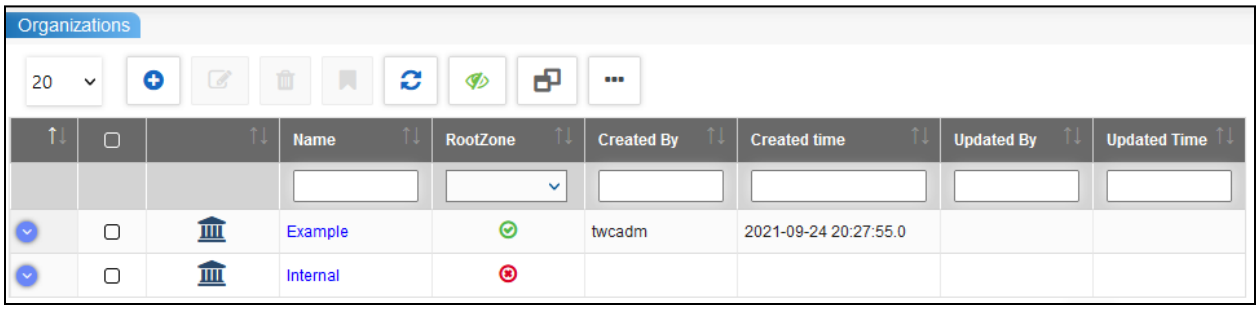
Here, we used an organization named "Example."

```
twc addorg --name="Example" --desc="Example Organization description" --
enable_root_zone=1
```

> If you don't want to enable root zone creation in the organization, set the "—enable_root_zone" value to '0.'

Navigation

The system displays the added Organization in Configuration Management >> Organizations >> Organizations grid.



Adding Network

After creating the organization, you must create a network to define objects, add appliances and domains, etc., in the IPAM using the CLIs. The following command creates a network in the TCPWave IPAM.

Command

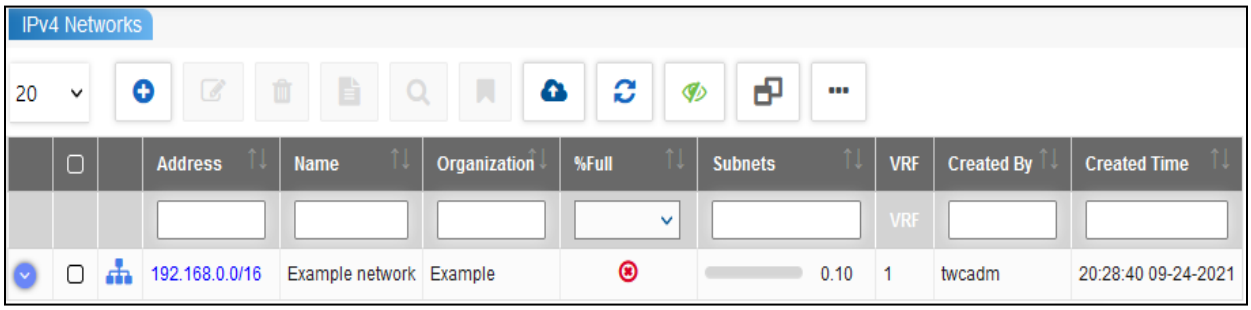
addnetwork

Example Usage

```
twc addnetwork --network=192.168.0.0 --name="Example network" --org=Example --
mask=16 --dnssec=0 --desc="Example lab Network"
```

Navigation

The system displays the added network in Network Management>> IPv4 Address Space >> IPv4 Networks gird.



Adding Domain

You must create a domain as all the networks, subnets, and objects are linked to the domain. The following command 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 in the same domain.

Command

adddomain

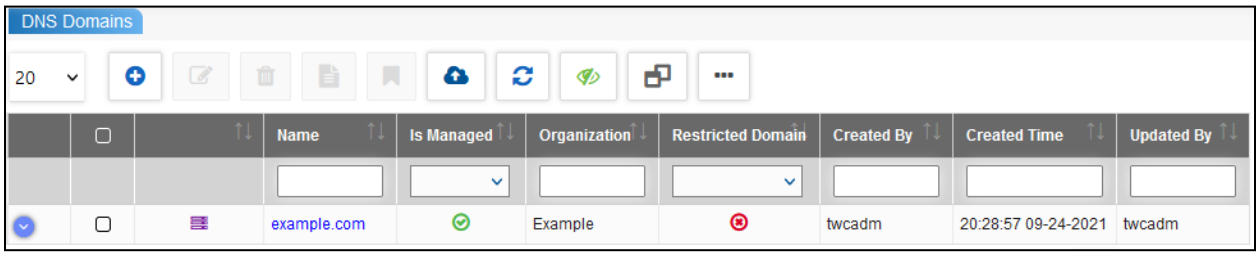
Example Usage

Here the domain is taken as example.com.

```
twc adddomain--name="example.com" --org=Example --desc="Example lab Domain"
```

Navigation

The system displays the added domain in Network Management >> DNS Management >> DNS Domains grid.



Adding and Associating Subnet

Post the subnet creation, and you must associate it with the domain. You have the privilege to specify any of the following subnet-types:

- Non-DHCP
- DHCP-Enabled

- Cloud-Hosted

Command

addsubnet

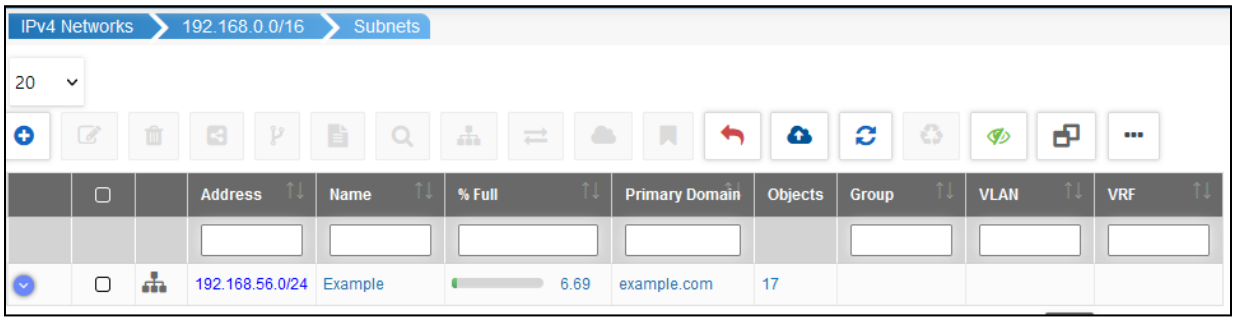
Example Usage

The following example uses the subnet-type as “Non-DHCP”

```
twc addsubnet --subnet=192.168.56.0 --mask=24 --network=192.168.0.0 --type=Non-DHCP --domain=example.com  --option=single --router_addr=192.168.56.1 --org=Example
```

Navigation

The system displays the created subnet in Network Management >> IPv4 Address Space >> IPv4 Networks >> Select Network >> Subnets grid.



Run the “twc addsubnet help” command for more information on other subnet attributes.

Editing Subnet

You can associate contacts, locations, etc. to the existing subnet as shown in the following screenshots:

Command

editsubnet

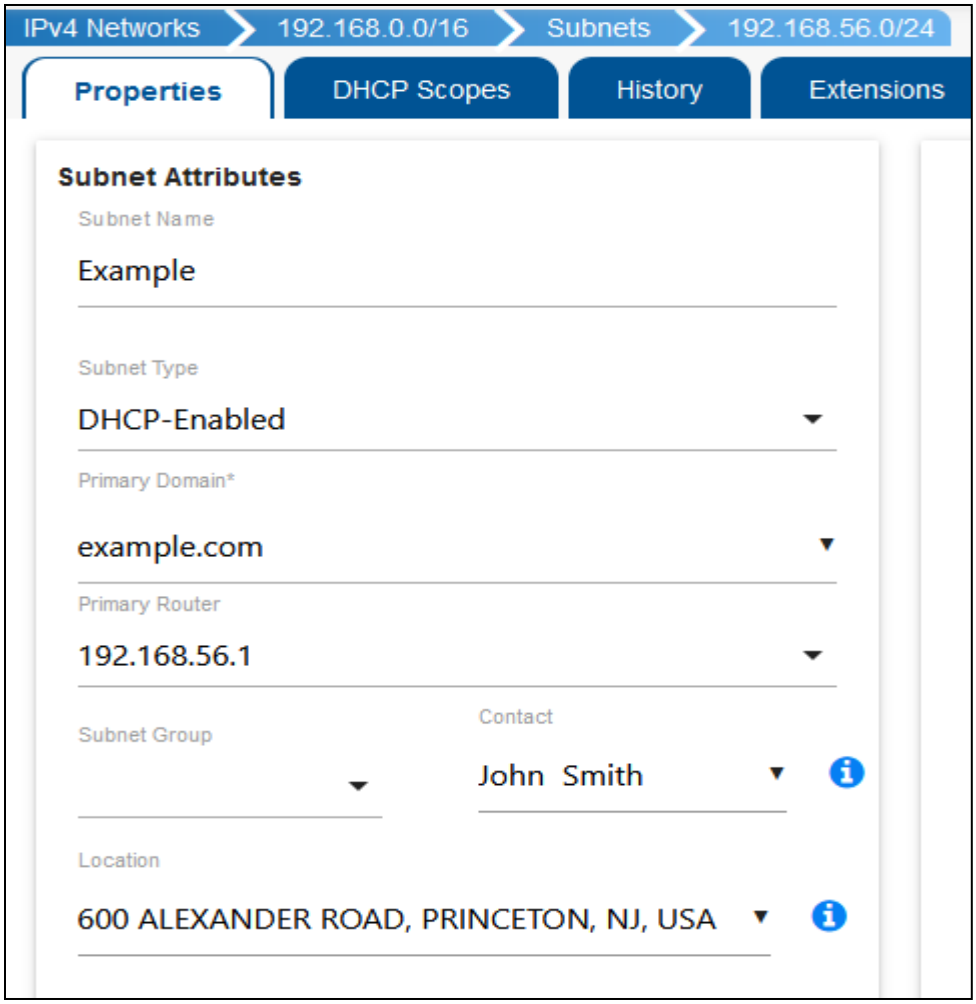
Example Usage

```
twc editsubnet --subnet=192.168.56.0 --contact_fname=John --contact_lname=Smith --contact_email=john.smith@example.com --org=Example
```

```
twc editsubnet --subnet=192.168.56.0 --type=DHCP-Enabled --street1="600 ALEXANDER ROAD" --city="PRINCETON" --state=NJ --country=USA --zip=08540 --org=Example
```

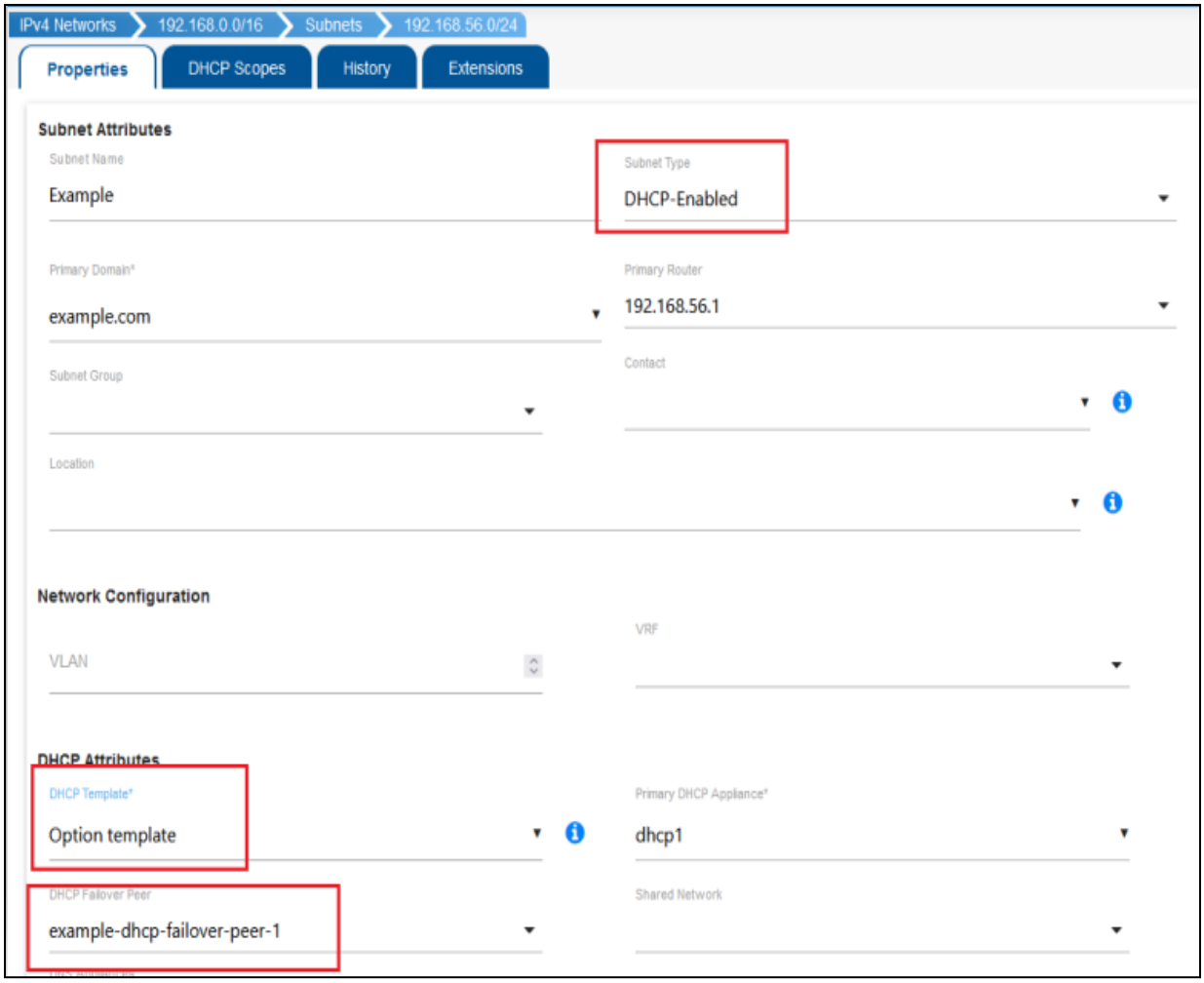
Navigation

The system displays the associated location in Network Management >> IPv4 Networks >> Select a Network >> Subnets >> Select a Subnet >> Properties tab.



You can also associate the DHCP Appliances.

```
twc editsubnet --subnet=192.168.56.0 --domain=example.com --type=DHCP-Enabled --dhcp_tmpl="Option template" --dhcp_appliance=192.168.56.12 --dhcp_failover_peer=example-dhcp-failover-peer-1 --router_addr=192.168.56.1 --org=Example
```



Adding Object

In TCPWave IPAM, the object maps a name and IP Address. When you create an object, the system stores the entry in a relational database where various integrity checks, Host naming policies, A to CNAME conflict checks, duplicate checks, etc., are performed. Upon successful creation of the object, the system automatically updates the zone of the DNS device. Unlike many DDI vendors, TCPWave does not use the insecure UDP-based RFC 2136 to send the updates to the remotes. TCPWave uses a unique encrypted channel that uses TCP over SSL to send a DDNS update to the DNS remotes. The system audits every, and it also has an undo option.

The below command adds objects to the network.

Command

addobject

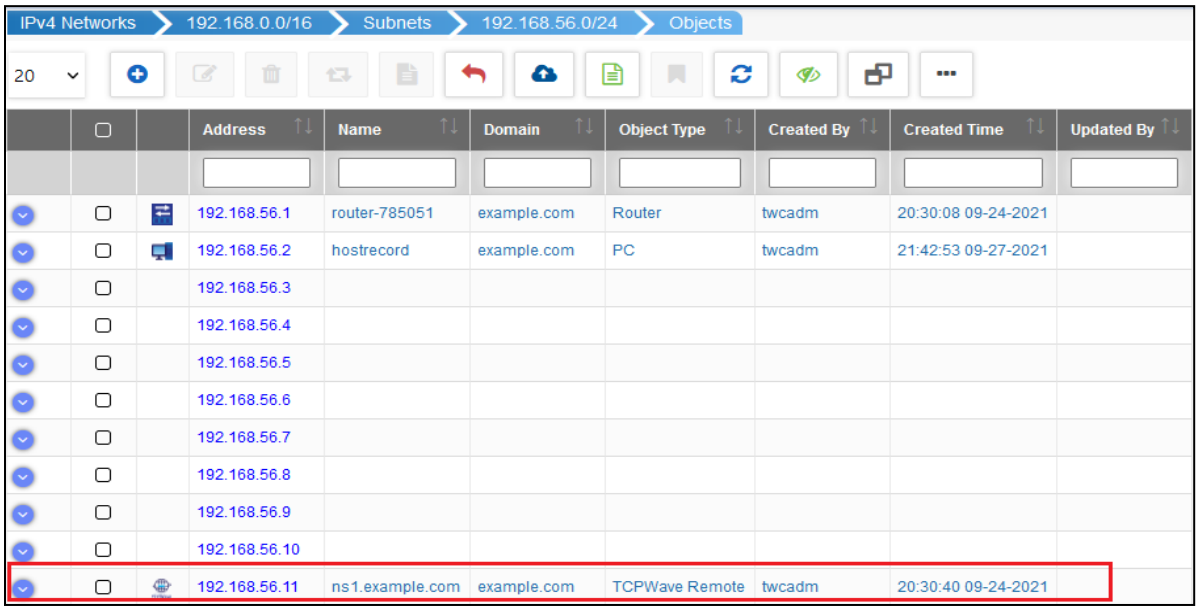
Example Usage

The following example uses TCPWave Remote object as ns1.example.com.

```
 twc addobject --object=192.168.56.11 --name="ns1.example.com" --
class_code="TCPWave Remote" --domain=example.com --alloc_type=static --
desc="Example lab Server" --ttl=300 --ns_a=1 --ns_ptr=1 --ddns_a=1 --ddns_ptr=1 -
-ddns_cname=1 --ddns_mx=1 --org=Example
```

Navigation

The system displays the created object in Network Management >> IPv4 Networks >> Select Network >> IPv4 Subnets >> Select Subnet >> IPv4 Objects grid.



For more information on other object attributes, run the “twc addobject help” command.

Adding Resource Record – Object Level

TCPWave supports various object-level resource records. Here, A Canonical Name (CNAME) resource record is taken as an example.

To create an object-level CNAME resource record, there should be a corresponding A record in the zone. Here in the example, an ‘A’ record ‘hostrecord’ is added to which the corresponding CNAME “cnamerecord” is created.

```
twc addobject --object=192.168.56.2 --name="hostrecord" --class_code=PC --
domain=example.com --alloc_type=Static --ttl=300 --ns_a=1 --ns_ptr=1 --ddns_a=1 -
-ddns_ptr=1 --ddns_cname=1 --ddns_mx=1 --org=Example
```

The following command Creates a DNS resource record in 'object', 'zone' or 'network' scopes

Command

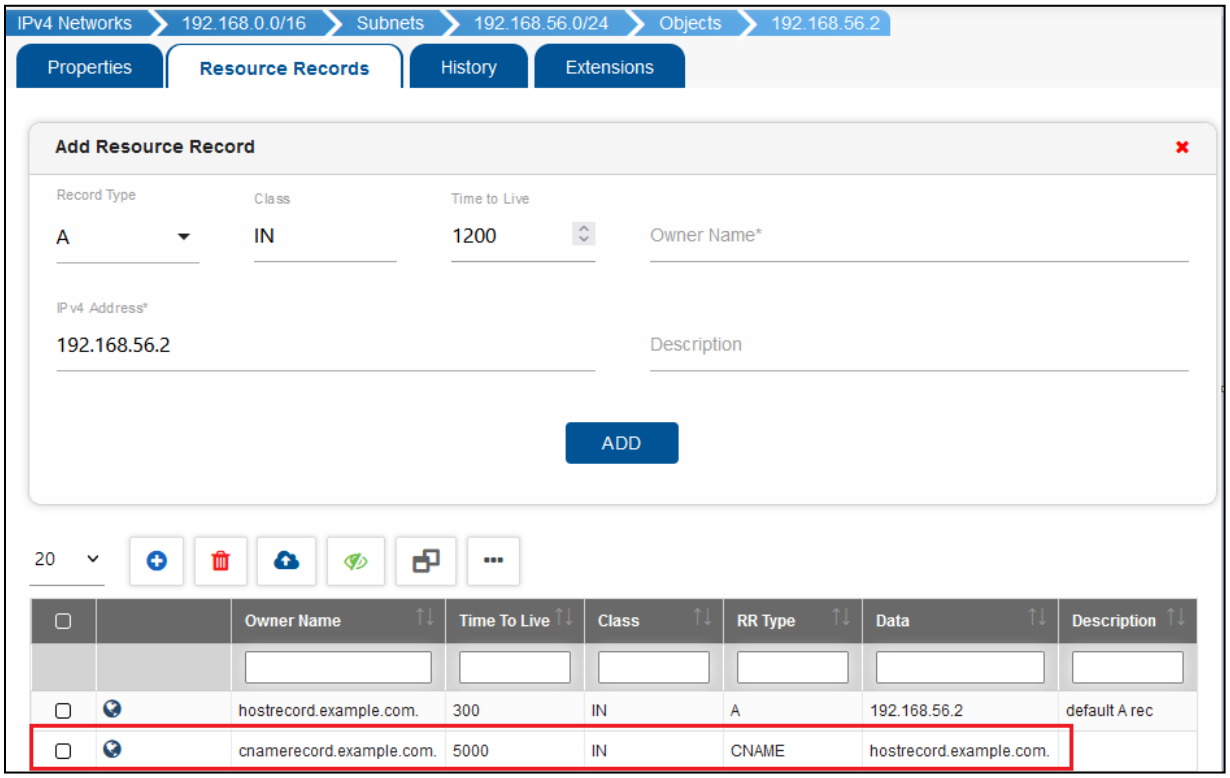
addrr

Example Usage

```
twc addrr --type=CNAME --class=IN --ttl=5000 --owner=cnamerecord --
ipv4=192.168.56.2 --cname=hostrecord --rr_scope=object --org=Example
```

Navigation

The system displays the CNAME record in Network Management >> IPv4 Address Space >> IPv4 Networks >> Select Network >> Subnets >> Select Subnet >> Objects >> Select Object >> Resource Records tab >> Add Resource Record grid.



Adding DNS Appliance Template

Before creating DNS Appliance, you must create DNS Appliance Template. You can configure options such as enable recursion, NTP servers, monitoring, logging, etc. that are the default options for all DNS Appliances.

Command

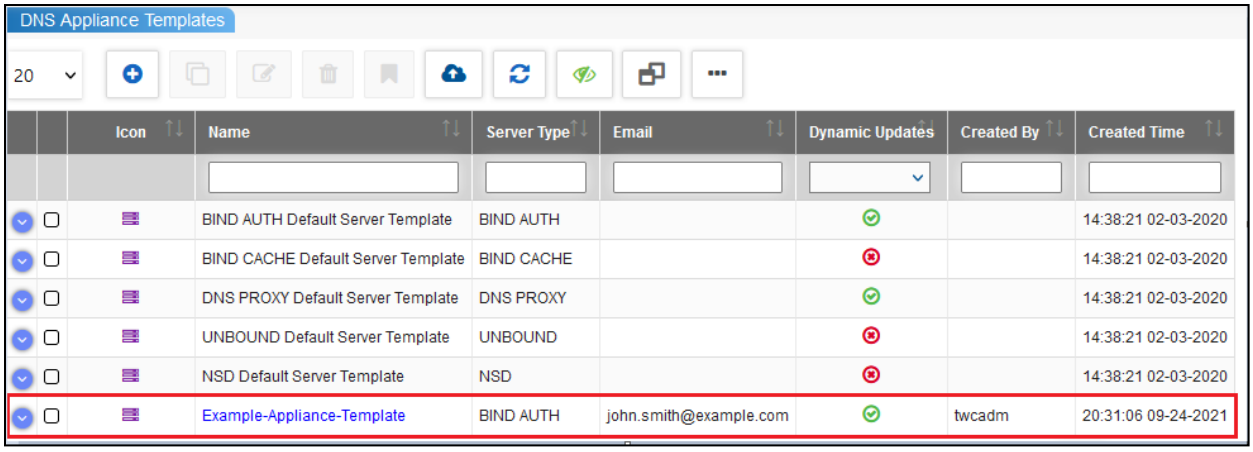
setdnsservertmpl

Example Usage

```
twc setdnsservertmpl --name="Example-Appliance-Template" --
desc="Example AUTH Server Template" --email=john.smith@example.com --
type="BIND AUTH" --dyn_upd=1 --algo="HMAC-SHA1:150,HMAC-SHA256:200" --
logger="client/default_stderr:default_debug:default_syslog" --action=add
```

Navigation

The system displays the added DNS Appliance Template in Network Management >> DNS Management >> DNS Appliance Templates grid.



Adding DNS Option Template

Before creating the DNS Option Template using CLI, you must define the input file named 'dnsoptiontemplate.txt' file in the /tmp folder with the following information.

```
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
Responses Per Second(responses-per-second)=0
Window(window)=15
Dump File(dump-file)=/var/named/log/named_dump.db
Lame TTL(lame-ttl)=0
Max Negative Cache TTL(max-ncache-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
Custom Parameters(custom-params)=
```

> The content of the input file varies based on the parameter '—type' that is used. The different types of DNS Appliances are = BIND AUTH, BIND CACHE, UNBOUND, NSD, DNS PROXY. For more information, run 'twc setdnsopttmpl help.'

A DNS Option template contains a set of parameters that decide the behavior of the DNS Appliance. While creating a DNS Appliance, a DNS Option template is associated.

Command

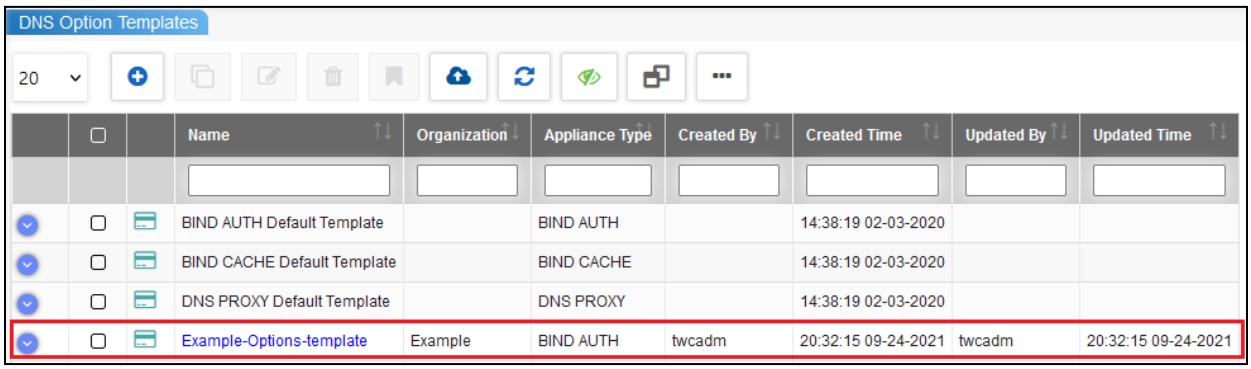
setdnsopttmpl

Example Usage

```
twc setdnsopttmpl --name="Example-Options-template" --type="BIND AUTH" --
input_file=/tmp/dnsoptiontemplate.txt --action=add --org=Example --
desc="Example AUTH Template"
```

Navigation

The system displays the added DNS Option Template in Network Management >> DNS Management >> DNS Option Templates >> DNS Option Templates grid.



Adding DNS Appliance

Before adding an Authoritative DNS Appliance from CLI, you must define an input file named 'dnsserver3.txt' in the /tmp folder using vi editor with the following information.

```
[dns-server]

TYPE=BIND AUTH
OPTION_TEMPLATE=Example-Options-template
APPLIANCE_TEMPLATE=Example-Appliance-Template
IP_ADDRESS=192.168.56.11
ORGANIZATION_NAME=Example
FIREWALL_TEMPLATE=
ENABLE_MONIT=1
DESCRIPTION=
TIME_ZONE=GMT (GMT)
STEALTH_APPLIANCE=0
ENABLE_RECURSION=no
INTERNAL_CACHE=0
APPLIANCE_GROUP=
NSM_TEMPLATE=

[ntp]

NTP_SERVERS=10.1.10.242
UPSTREAM=
DOWNSTREAM=

[dns-resolver]

NAME_APPLIANCES=8.8.8.8
SEARCH_SUFFIIXES=

[snmp]
```

```
TRAP_SINK_1=1.1.1.1
TRAP_SINK_2=2.2.2.2
COMMUNITY_STRING=sph1nkx5
SYSTEM_LOCATION=
SYSTEM_CONTACT=
PROCESS_LIST=ntpd,dns,sshd,monit,syslog-ng,dhcpd
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-filter]

FILTER_NAME=f_dhcp
CONDITION=complex
COMPLEX_CONDITION=facility(local7)

[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-destination]

DESTINATION_NAME=d_dhcp
TYPE_SNG=1
LOG_FILE_NAME=dhcpd.log
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

[syslogng-target]

SOURCE=s_sys
FILTER=f_dhcp
DESTINATION=d_dhcp

[view]

NAMES=

[banner]

The default value of Banner configuration for DNS and DHCP appliances.
```

The following command Creates or updates DNS appliance configuration in the TCPWave IPAM.

Command

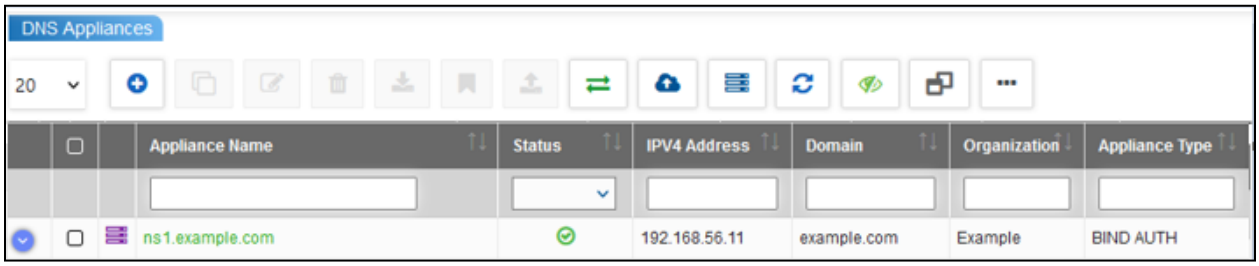
setdnsserver

Example Usage

```
twc setdnsserver --input_file=/tmp/dnsserver3.txt --action=add
```

Navigation

The system displays the added DNS Appliance in Network Management >> DNS Management >> DNS Appliances >> TCPWave DNS Appliances grid.



The content of the input file varies based on the parameter '—type' that is used. The different types of DNS Appliances are = BIND AUTH, BIND CACHE, UNBOUND, NSD, DNS PROXY. For more information on the appliance attributes, run '*twc setdnsserver help*.'

Sync

You can perform a DNS Server full configuration sync using TCPWave CLI.

Command

syncdnsserver

Example

```
twc syncdnsserver --ip=192.168.56.11 --appliance_type="BIND AUTH" --org=Example
```

The following CLI screenshot displays the confirmation message:

```
[root@ipam ~]# twc syncdnsserver --ip=192.168.56.11 --appliance_type="BIND AUTH" --org=Example
DNS full configuration sync has been completed successfully on DNS Appliance: ns1.example.com
```

The following screenshot displays the updated database upon the successful DNS Appliance synchronization.

```
$ORIGIN .
$TTL 1200       ; 20 minutes
example.com             IN SOA  ns1.example.com.example.com. jsmith.example.com. (
                                2021092727 ; serial
                                21600      ; refresh (6 hours)
                                3600       ; retry (1 hour)
                                604800     ; expire (1 week)
                                86400      ; minimum (1 day)
                                )
                        NS      ns1.example.com.example.com.
$ORIGIN example.com.
$TTL 5000       ; 1 hour 23 minutes 20 seconds
cname                   CNAME   testrecord
cname1                  CNAME   www
cnamerecord             CNAME   hostrecord
$TTL 300        ; 5 minutes
ns1.example.com         A       192.168.56.11
hostrecord              A       192.168.56.2
$TTL 5000       ; 1 hour 23 minutes 20 seconds
mail                    A       192.168.56.3
sipserver               A       192.168.56.3
testcname               CNAME   www2
$TTL 300        ; 5 minutes
testrecord              A       192.168.56.21
$TTL 5000       ; 1 hour 23 minutes 20 seconds
text                    TXT     "spf1 a:mail.tcpwave.com -all"
$TTL 300        ; 5 minutes
webserver               A       192.168.56.200
                        A       192.168.56.201
webserver1              A       192.168.56.202
webserver10             A       192.168.56.211
webserver2              A       192.168.56.203
webserver3              A       192.168.56.204
webserver4              A       192.168.56.205
webserver5              A       192.168.56.206
webserver6              A       192.168.56.207
webserver7              A       192.168.56.208
webserver8              A       192.168.56.209
webserver9              A       192.168.56.210
$TTL 5000       ; 1 hour 23 minutes 20 seconds
www                     A       192.168.56.3
                        MX      10 mail.
$ORIGIN www.example.com.
_sip._tcp               SRV     1 10 7001 sipserver.example.com.
$ORIGIN example.com.
www2                    A       192.168.56.21
```

Adding DNS Zone Template

Before creating a zone, you must create a zone template and associate it with the zone. The following command creates a DNS zone template in the TCPWave IPAM.

Command

adddnszonetmpl

Example Usage

```
twc adddnszonetmpl --name="Example-zone-template" --desc="Example zone template" --soa_email=jsmith@example.com --soa_refresh=21600/S --soa_retry=3600/S --soa_expiry=604800/S --soa_negcache=86400/S --allow_update=localhost,localnets --masters=192.168.56.11 --org=Example --empty_forwarders=1
```

Navigation

The system displays the added DNS Zone Template in Network Management >> DNS Management >> DNS Templates >> DNS Zone Templates >> DNS Zone Templates grid.

		Template Name	Organization	Created By	Created Time	Updated By	Updated Time	Description
☐		Example-zone-template	Example	twcadm	20:38:33 09-24-2021	twcadm	20:38:33 09-24-2021	Example zone template

Adding Zone

You can create a DNS Zone in the TCPWave IPAM using the following command.

Command

adddnszone

Example Usage

```
twc adddnszone --name="example.com" --org=Example --zone_tmpl="Example-zone-template" --dnssec=1 --nsec_opt=NSEC --ad_upd=0 --monit=0 --dmz_visible=0 --is_restricted=0 --desc="Example lab Zone"
```

Navigation

The system displays the added Zone in Network Management >> DNS Management >> DNS Zones >> Managed DNS Zones.

		Name	Organization	Zone Template	Default TTL	Restricted Zone	DNSSEC	Serial number	AD
☐		example.com	Example	Example-zone-template	1200 Sec	⊗	✓	2021092703	⊗

Run the 'twc adddnszone help' command for more information on zone attributes.

Adding Resource Record – Zone Level

TCPWave supports various zone-level resource records. A few of them are A, CNAME, SRV, MX, TXT, etc.

Command

addrr

Example Usages

A Record

```
[root@ipam ~]# twc addrr --type=A --class=IN --ttl=5000 --owner=www.example.com. --ipv4=192.168.56.3 --rr_scope=zone --zone_name=example.com --org=Example
```

CNAME

```
[root@ipam ~]# twc addrr --type=CNAME --class=IN --ttl=5000 --owner=cname1 --cname=www --zone_name=example.com --rr_scope=zone --org=Example
```

MX

```
[root@ipam ~]# twc addrr --type=MX --class=IN --ttl=5000 --owner=www --prefnum=10 --mail
```

SRV

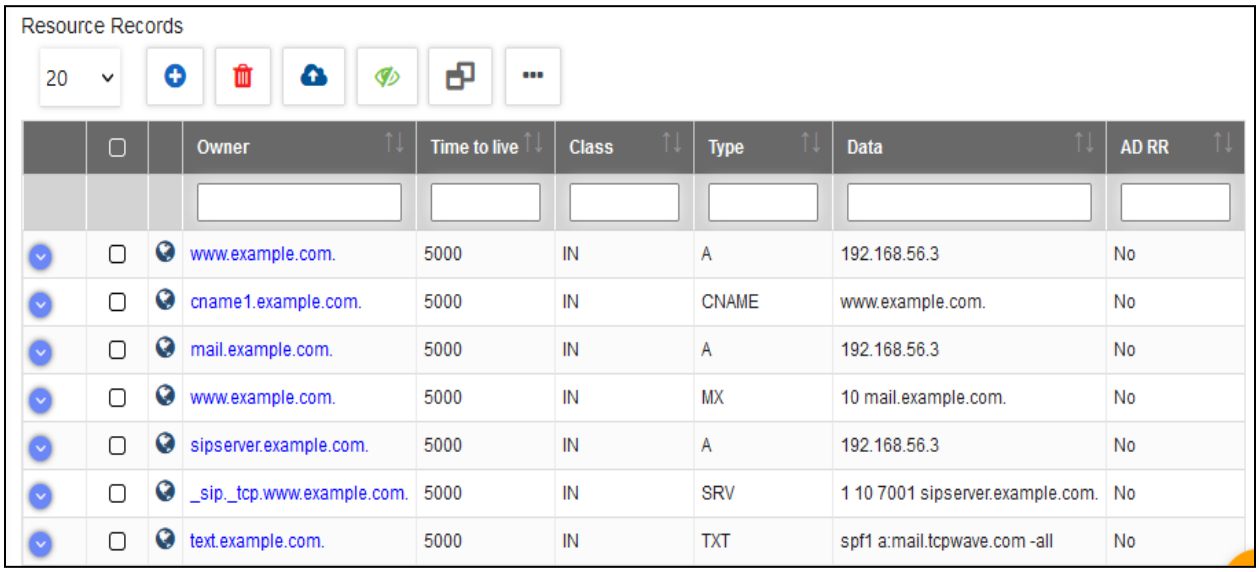
```
[root@ipam ~]# twc addrr --type=SRV --class=IN --ttl=5000 --owner=www --service=sip --protocol=tcp --priority=1 --weight=10 --port=7001 --target=sipserver.example.com. --rr_scope=zone --zone_name=example.com --org=Example
```

TXT

```
[root@ipam ~]# twc addrr --type=TXT --class=IN --ttl=5000 --owner=text --txt="spf1 a:mail.tcpwave.com -all" --zone_name=example.com --rr_scope=zone --org=Example
```

Navigation

The system displays the added resource records in Network Management >> DNS Management >> DNS Zones >> Managed DNS Zones >> Select Zone >> Resource Records tab >> Resource Records grid.



Owner	Time to live	Class	Type	Data	AD RR
www.example.com.	5000	IN	A	192.168.56.3	No
cname1.example.com.	5000	IN	CNAME	www.example.com.	No
mail.example.com.	5000	IN	A	192.168.56.3	No
www.example.com.	5000	IN	MX	10 mail.example.com.	No
sipserver.example.com.	5000	IN	A	192.168.56.3	No
_sip._tcp.www.example.com.	5000	IN	SRV	1 10 7001 sipserver.example.com.	No
text.example.com.	5000	IN	TXT	spf1 a:mail.tcpwave.com -all	No

Associating Zone Template to Reverse Zone

The corresponding reverse zones are automatically created to associate the zone template to the reverse zones created when creating the networks. The reverse zone inherits the configuration defined under the template.

Command

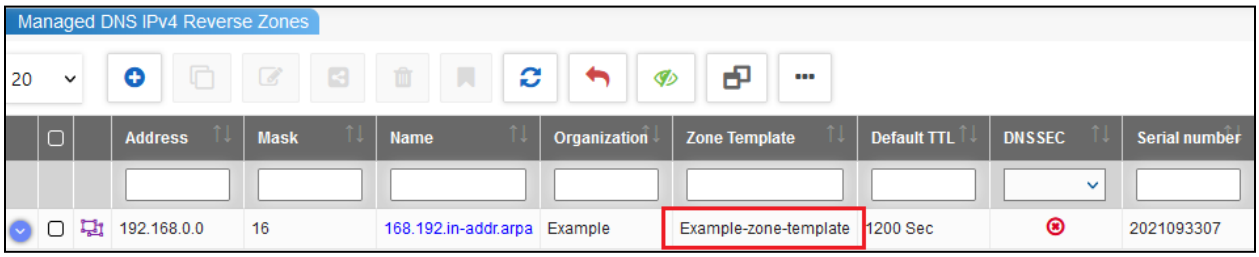
editdnsreversezone

Example

```
twc editdnsreversezone --ip=192.168.0.0 --mask=16 --org=Example --desc="Example lab Reverse Zone" --zone_tmpl="Example-zone-template" --monit=0
```

Navigation

The associated Zone Template is displayed in the Zone Template column of the Managed DNS IPv4 Reverse Zones grid.



Editing Global Options

You can edit the existing global options as per the changing business requirements.

Command

editglobalopts

Example

Here “Enable Debug Logging” is an example of a global option.

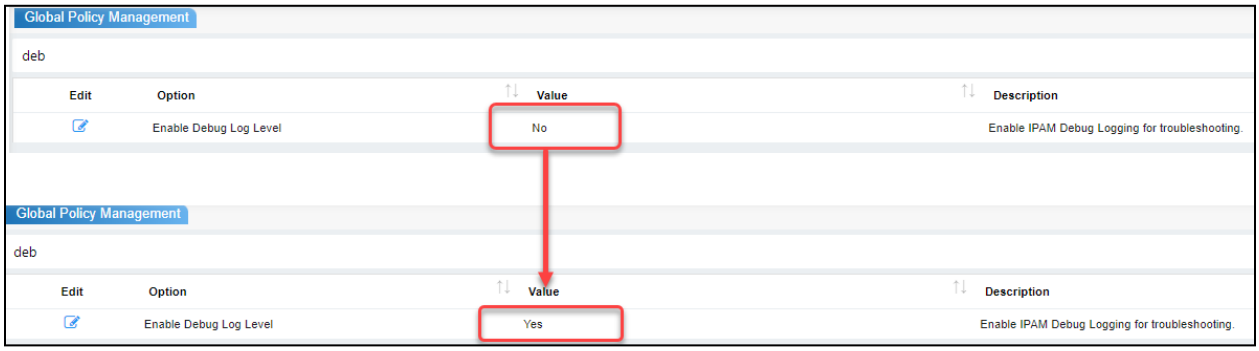
```
[root@ipam ~]# twc editglobalopts --option=ENABLE_DEBUG_LOGGING --value=Yes
```

The following screenshot displays the edited output of the mentioned global option.

```
[root@ipam ~]# twc editglobalopts --option=ENABLE_DEBUG_LOGGING --value=Yes
Global Option successfully updated
```

Navigation

The system displays the updated global option value in Administration >> Configuration Management >> Global Policy Management.



Search

You can perform search operations using various parameters such as search type. Here the search type is taken as “Text.”

Command

search

Example Usage

```
twc search --search_type="Text" --field_name="Name" --search_term="webserver" --entity_type="Object"
```

The following screenshot displays the search results for the term “webserver”.

```
[root@ipam tmp]# twc search --search_type="Text" --field_name="Name" --search_term="webserver" --entity_type="Object"
"Address","Name","Domain","ObjectType","AllocationType","MAC",
"192.168.56.200","webserver","example.com","Web Server","Static","",
"192.168.56.201","webserver1","example.com","Web Server","Static","",
"192.168.56.202","webserver2","example.com","Web Server","Static","",
"192.168.56.203","webserver3","example.com","Web Server","Static","",
"192.168.56.204","webserver4","example.com","Web Server","Static","",
"192.168.56.205","webserver5","example.com","Web Server","Static","",
"192.168.56.206","webserver6","example.com","Web Server","Static","",
"192.168.56.207","webserver7","example.com","Web Server","Static","",
"192.168.56.208","webserver8","example.com","Web Server","Static","",
"192.168.56.209","webserver9","example.com","Web Server","Static","",
"192.168.56.210","webserver10","example.com","Web Server","Static","",
"192.168.56.211","webserver10","example.com","Web Server","Static","",
```

Adding DHCP Policy Template

The DHCP Policy Templates are used while creating the DHCP appliances. This granular function provides the appropriate access to the policy templates to be used to create or edit the corresponding DHCP appliances. Before creating DHCP Policy Template, you must define an input file in the /tmp folder with the following information:

```
# Global Parameters

Log Facility(log-facility)=
DHCP Cache Threshold(dhcp-cache-threshold )=50

# DHCP Server-Client Communications Parameters

Always Broadcast(always-broadcast)=false
Always reply RFC1048(always-reply-rfc1048)=true
Min Secs(min-secs)=200
Stash Agent Options(stash-agent-options)=false

# DDNS Update Options

DDNS Update Style(ddns-update-style)=interim
Update Conflict Detection(update-conflict-detection)=false
Update Optimization(update-optimization)=true

# Client Handling Parameters

Adaptive Lease Time Percentage(adaptive-lease-time-threshold)=
Boot Unknown Clients(boot-unknown-clients)=
Default Lease time(default-lease-time)=3600
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)=true
Ping Timeout(ping-timeout)=
Ping Timeout In Milliseconds(ping-timeout-ms)=50
Do not use Fsync(dont-use-fsync)=
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)=

Using VI editor, created the file in /tmp folder

```
[root@ipam tmp]# ls | grep dhcp
dhcppolicytemplate.txt
[root@ipam tmp]#
```

The following command creates or updates a DHCP policy template in the TCPWave IPAM.

Command

setdhcppolicytmpl

Example Usage

```
twc          setdhcppolicytmpl          --name="example-policy-template"          --
input_file=/tmp/dhcppolicytemplate.txt --action=add --org=Example
```

Navigation

The system displays the added DHCP Policy Template in Network Management >> DHCP Management >> DHCP Templates >> DHCP Policy Templates >> DHCP Policy Templates grid.

		Name	Organization	Created By	Created Time	Updated By	Updated Time	Description
☐		example-policy-template	Example	twcadm	00:20:29 09-28-2021			

Adding DHCP Option Template

This granular function provides the appropriate access to the create/update/delete operations of an IPv4 DHCP Option Template. These templates are used to define the IPv4 DHCP scopes.

Create DHCP template input file in /tmp folder

General Option
Template Name=example dhcp option template
Organization=Example
# User Authentication Protocol Options
User Authentication Servers (98)=

TCP Parameters

Default TCP TTL (37)=30
Keepalive Time (38)=24
Keepalive Data (39)=true

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)=Same as the primary domain in subnet profile
Swap Server (16)=
Root Path (17)=
Extension File (18)=

# Novell Options

NDS Servers (85)=
NDS Tree Name (86)=
NDS Context (87)=
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)=

# Miscellaneous

User Class (77)=
Agent/Circut Id (82)=
Agent/Remote Id (82)=
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)=

# 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

Next Server=1.0.0.6
IP Address Lease Time (51)=
Overload (52)=
Vendor Class Id (60)=
Client Id (61)=
Server Name (66)=
Bootfile Name (67)=

Auto DDNS Update Options

DDNS Updates=true
Update Optimization=false
DDNS Domain Name=Same as the primary domain in subnet profile|primary-10.1.10.86|secondary-10.1.10.87
DDNS Reverse Domain Server Details=Same as the DDNS Domain Name options
Client Updates=true
Application and Service Parameters

StreetTalk Server (75)=
StreetTalk Directory Assistance (STDA) Server (76)=
BCMCS Controller Domain Name (88)=
BCMCS Controller IPv4 address option (89)=
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)=

The DHCP option template file has been created in the /tmp folder

```
[root@ipam tmp]# ls | grep dhcpoption
dhcpoptiontemplate.txt
[root@ipam tmp]#
```

The following command Creates or updates a DHCP policy template in the TCPWave IPAM.

Command

setdhcptmpl

Example

```
twc setdhcptmpl --input_file=/tmp/dhcpoptiontemplate.txt --action=add
```

Navigation

The system displays the added DHCP Option Template in Network Management >> DHCP Management >> DHCP Templates >> DHCP Option Templates grid.

		Name	Organization	Created By	Created Time	Updated By	Updated Time	Description
☐		example dhcp option template	Example	twcadm	00:31:27 09-28-2021			

Ensure that you update the correct org name and option template name configured in the IPAM.

Adding DHCP Appliances

Prior to adding the DHCP Appliances, you must define the input files in /tmp folder with the following information:

```
[dhcp-server]
IP_ADDRESS=192.168.56.12
NEW_IP_ADDRESS=
ORGANIZATION_NAME=Example
POLICY_TEMPLATE=example-policy-template
APPLIANCE_GROUP=
ENABLE_MONIT=1
TIME_ZONE=GMT (GMT)
FIREWALL_TEMPLATE=
ENABLE_FINGERPRINT=0
LISTENING_INTERFACES=
DESCRIPTION=
[ntp]
NTP_SERVERS=10.1.10.242
UPSTREAM=
DOWNSTREAM=
[dns-resolver]
NAME_APPLIANCES=8.8.8.8
SEARCH_SUFFIIXES=
[snmp]
TRAP_SINK_1=1.1.1.1
TRAP_SINK_2=2.2.2.2
COMMUNITY_STRING=sph1nkx5
SYSTEM_LOCATION=
SYSTEM_CONTACT=
PROCESS_LIST=
ENABLE_SNMPV3=false
FIREWALL_SNMP_ACL=
[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-filter]

FILTER_NAME=f_dhcp
CONDITION=complex
COMPLEX_CONDITION=facility(local7)

[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-destination]
DESTINATION_NAME=d_dhcp
TYPE_SNG=1
LOG_FILE_NAME=dhcpd.log
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
[syslogng-target]
SOURCE=s_sys
FILTER=f_dhcp
DESTINATION=d_dhcp
[banner]
Example  lab DHCP appliances.
```

The file has been created with Vi editor.

```
[root@ipam tmp]# ls | grep dhcpserver
dhcpserveradd.txt
[root@ipam tmp]#
```

The following command Creates or updates DHCP server configuration in the TCPWave IPAM.

Command

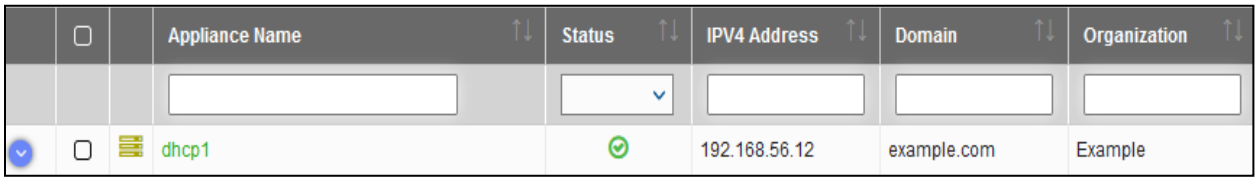
setdhcpserver

Example

```
twc setdhcpserver --input_file=/tmp/dhcpserveradd.txt --action=add
```

Navigation

The system displays the added DHCP Appliance in Network Management >> DHCP Management >> DHCP Appliances >> TCPWave DHCP Appliances grid.

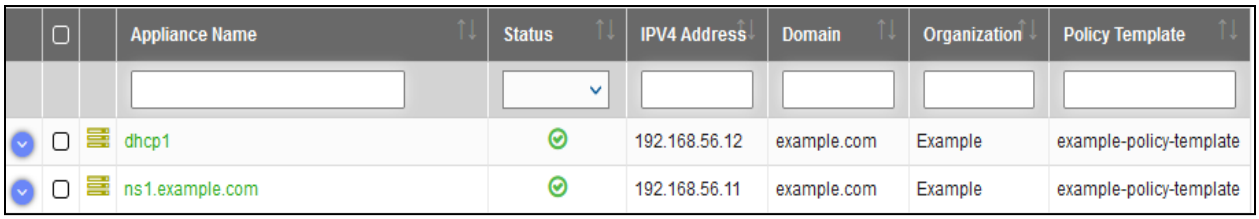


Appliance Name	Status	IPV4 Address	Domain	Organization
dhcp1		192.168.56.12	example.com	Example

Ensure that the DHCP server objects are added to the network before adding to the server.

Adding DHCP Failover Peer

After creating DHCP Appliances, you must create a new input file with the IP of the new appliance. Once this is done, you can see the second DHCP appliance is also added. We have used the existing remote appliance ns1.example.com as DHCP Failover Peer.



Appliance Name	Status	IPV4 Address	Domain	Organization	Policy Template
dhcp1		192.168.56.12	example.com	Example	example-policy-template
ns1.example.com		192.168.56.11	example.com	Example	example-policy-template

The following command creates a DHCP failover peer in the TCPWave IPAM

Command

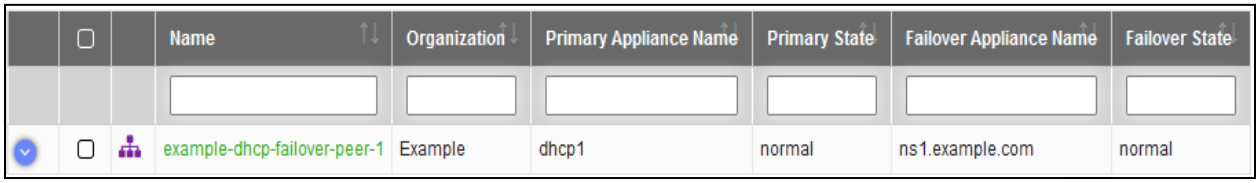
adddhcpfailoverpeer

Example Usage

```
twc adddhcpfailoverpeer --name=example-dhcp-failover-peer-1 --org=Example --primary_appliance=192.168.56.12 --failover_appliance=192.168.56.11 --appliance_type=dhcp --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"
```

Navigation

The system displays the added DHCP Failover Peer in the Network Management >> DHCP Management >> DHCP Settings >> DHCP Failover Peers grid.



Name	Organization	Primary Appliance Name	Primary State	Failover Appliance Name	Failover State
example-dhcp-failover-peer-1	Example	dhcp1	normal	ns1.example.com	normal

Adding Scope

The following command helps you to create DHCP scope

Command

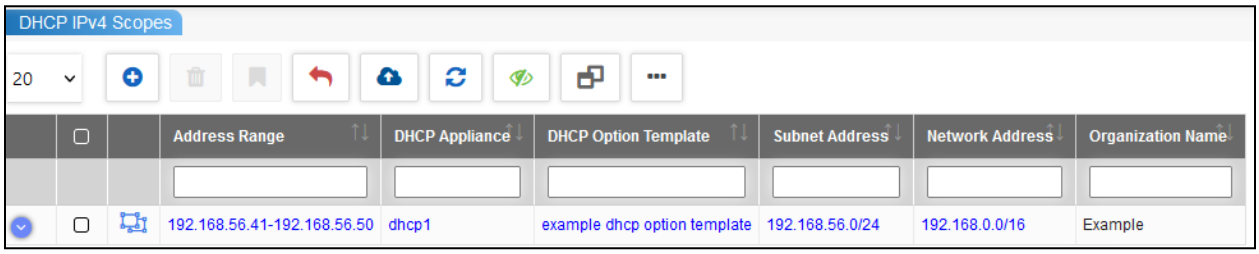
addscope

Example Usage

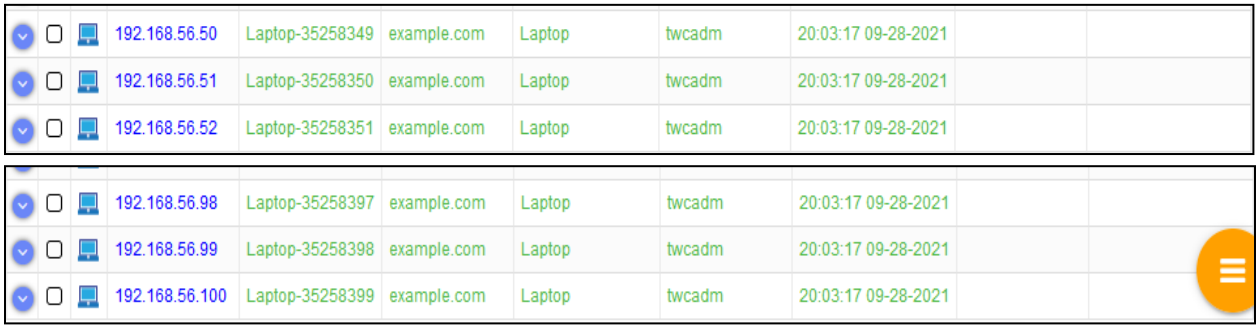
```
twc addscope --start_ip=192.168.56.50 --end_ip=192.168.56.100 --obj_type="Laptop"
--ttl=1200 --ns_a=1 --ns_ptr=1 --ddns_a=1 --ddns_ptr=1 --ddns_cname=1 --ddns_mx=1
--org=Example
```

Navigation

The system displays the added scope in Network Management >> DHCP Management >> DHCP Scopes >> DHCP IPv4 Scopes.



When navigating to the subnet section, the system highlights the added scopes in green.



Adding Admins

TCPWave supports various admin roles. For more information on roles, please refer Annexure.

Example Usages

SADM

```
twc addadmin --first_name=John --middle_name=Fitzgerald --last_name=Kennedy --
email=john.kennedy@example.com --phone=920-310-5555 --login_name=jkennedy --
org=Example --role=SADM --groups="Default SADM Admin Group"
```

PADM

```
twc addadmin --first_name=Don --last_name=Smith --email=don.smith@example.com --
phone=920-310-4455 --login_name=dsmith --org=Example --role=PADM --
groups="Default PADM Admin Group"
```

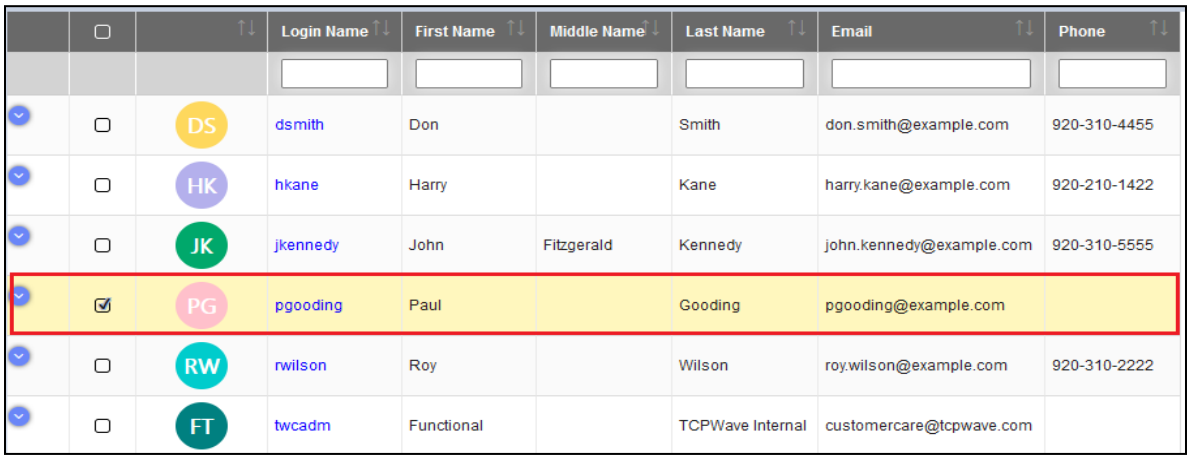
RADM

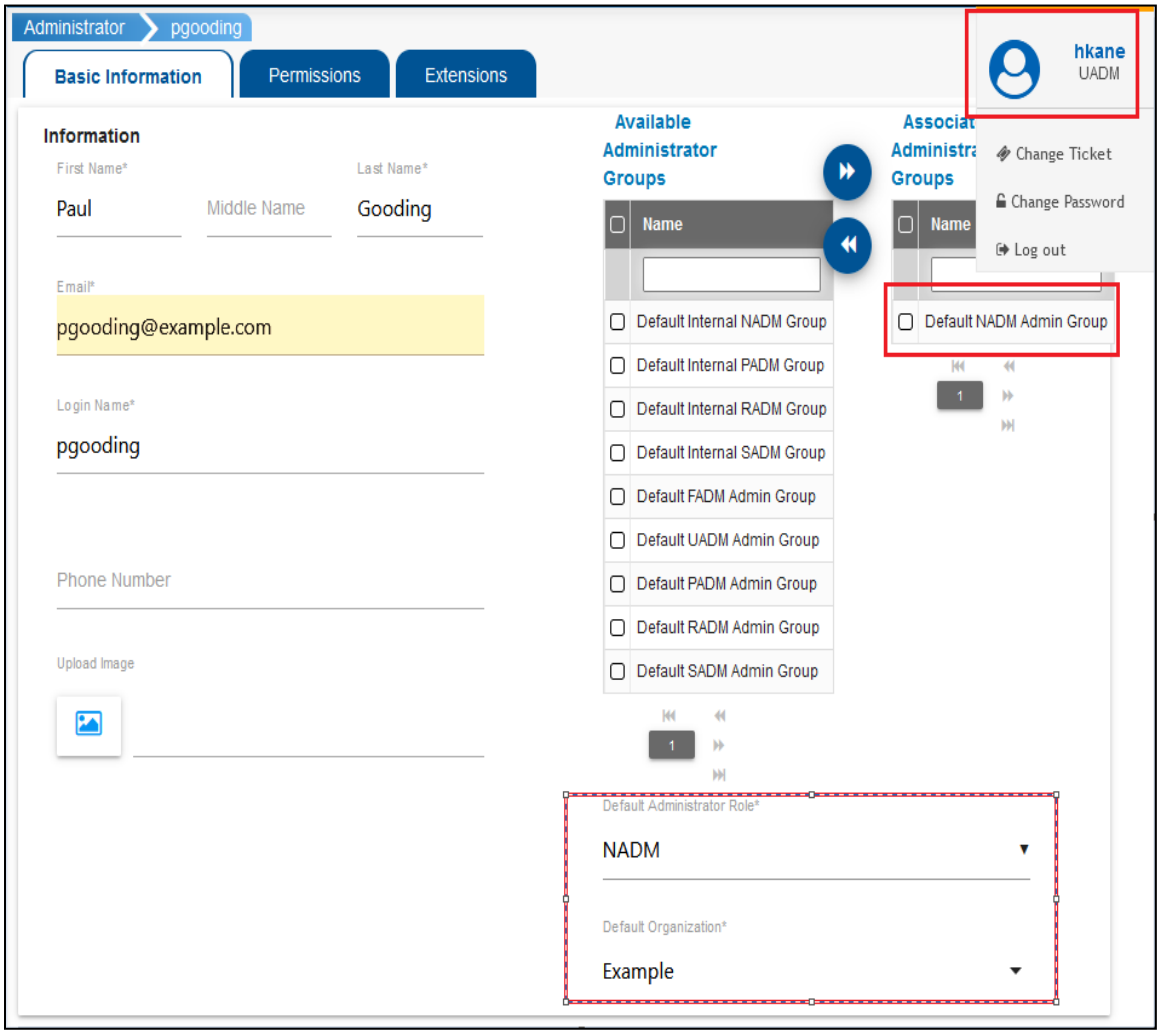
```
twc addadmin --first_name=Roy --last_name=Wilson --email=roy.wilson@example.com -
-phone=920-310-2222 --login_name=rwilson --org=Example --role=RADM --
groups="Default RADM Admin Group"
```

UADM

```
twc addadmin --first_name=Harry --last_name=Kane --email=harry.kane@example.com -
-phone=920-210-1422 --login_name=hkane --org=Example --role=UADM -groups="Default
UADM Admin Group"
```

The following screenshots illustrate that the user has logged in as a UADM and created a NADM





The below screenshot explains the write permissions given to a NADM to IPv4 Subnet – 10.1.8.0/24

```
twc addadminpermission --level=Admin --admin=pgooding --org=Example --role=NADM --function="IPv4 Subnets" --privilege=Write --select_all=1 --list_entity=10.1.8.0/24
```

Adding Contact

Command

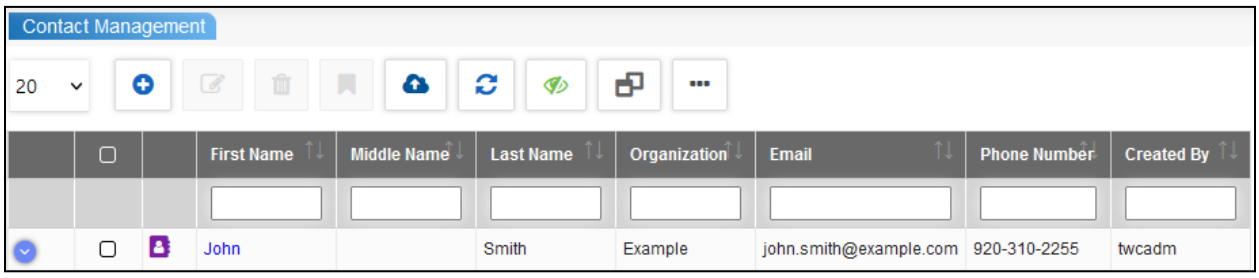
addcontact

Example Usage

```
twc addcontact --first_name=John --last_name=Smith --org=Example --email=john.smith@example.com --phone=920-310-2255
```

Navigation

The system displays the added contact in the Administration >> Contact Management grid



Adding Location

You can create a location for a given organization in the TCPWave IPAM.

Command

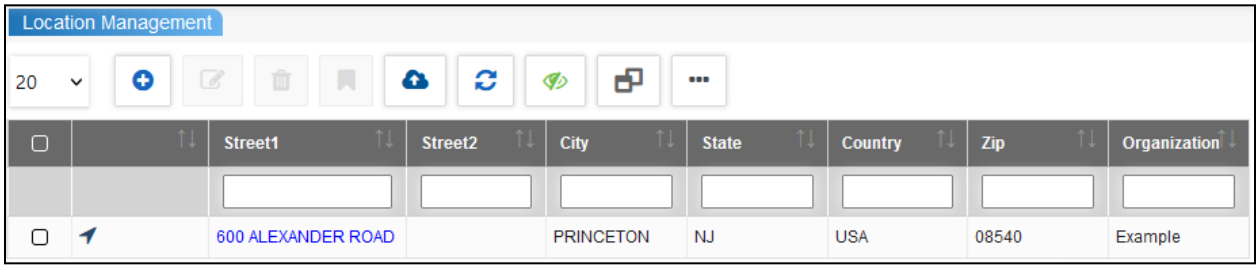
addlocation

Example Usage

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

Navigation

The system displays the added location in the Administration >> Location Management grid.



Adding Extension

You can add user-defined attributes for an admin, network, subnet, object, zone, revzone, rr. The following example displays the “Location” as an attribute in the network zone levels.

Command

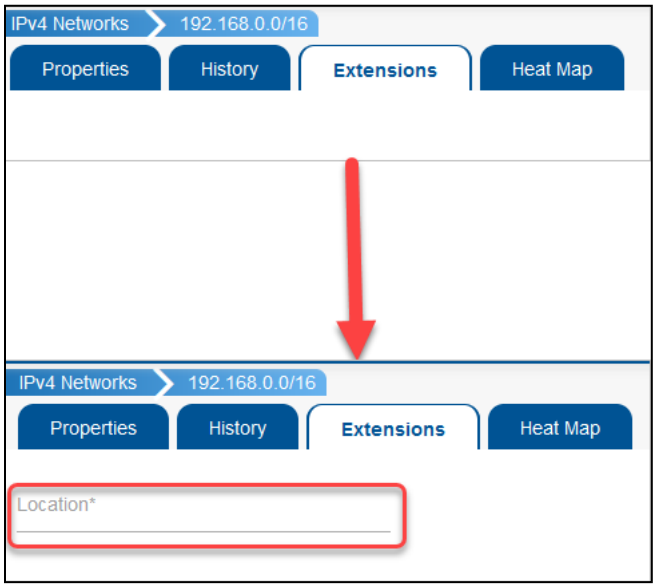
addext

Example Usage 1

```
twc addext --name=Location --type=string --entities=network/1,zone/1 --mandatory=1 --desc="location"
```

Navigation

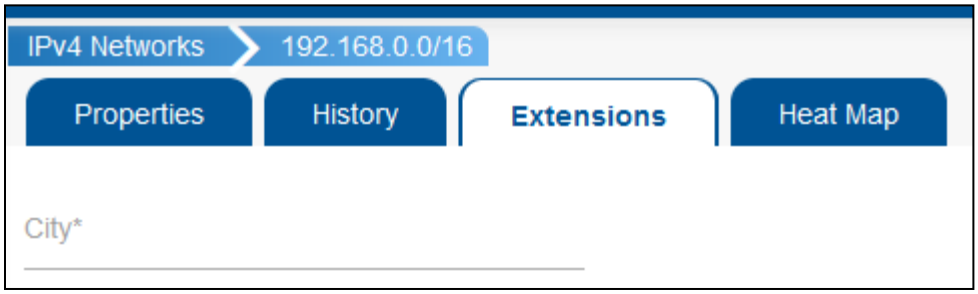
At the network level, the system displays the added extension in Network Management >> IPv4 Address Space >> IPv4 Networks >> Select a network >> Extensions tab.



Example Usage 2

The following example displays an updated extension value “City” at a network level.

```
twc editext --name=Location --new_name=City --mandatory=1
```



Adding Scheduled Job

Apart from the existing Scheduled Jobs, you can also create a new Scheduled Jobs.

Command

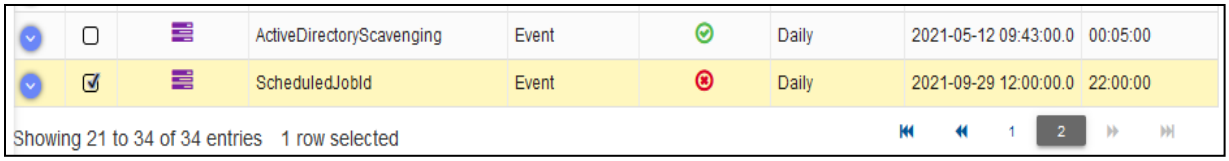
addscheduledjob

Example Usage

```
twc addscheduledjob --job_id=ScheduledJobId --job_type=event-handler --event_handler=RemoteCheckoutOperation --repeat_type=daily --start_date="2021-09-29 12:00:00" --end_date="2021-09-30 00:00:00" --exe_at=22:00:00 --desc="The IPAM Checkout operation"
```

Navigation

The system displays the added scheduledjob in Administration >> Scheduler Management >> Scheduled Tasks.



	☐		ActiveDirectoryScavenging	Event		Daily	2021-05-12 09:43:00.0	00:05:00
	☑		ScheduledJobId	Event		Daily	2021-09-29 12:00:00.0	22:00:00

Showing 21 to 34 of 34 entries 1 row selected

Executing Scheduled Jobs

Apart from executing the pre-defined scheduled jobs, you can also execute custom-defined scheduled jobs.

Command

exescheduledjob

Example Usage

```
twc exescheduledjob --job_id=DailyReports
```

The CLI output is shown below:

```
[root@ipam ~]# twc exescheduledjob --job_id=DailyReports
Scheduled job has been executed successfully
```

Reports

TCPWave provides an enterprise-grade DDI reporting framework built into the core DDI product. It gives about 120 reports, and you can generate various reports in different formats. Example: DNS SOA, Network Utilization Report.

Command

rptdnssoa

Example Usage

```
twc rptdnssoa --dns_auth_server=ns1.example.com --output_file=/tmp/output.csv --format=csv
```

The following screenshot illustrates the output of the reports in csv format:

```
"Server Name","Server IP","Server Type","Zone Name","Name Server","Email","Serial Number","Refresh Time","Retry Time","Expire Time","Negative Cache"
"ns1.example.com","192.168.56.11","BIND AUTH","example.com","ns1.example.com.example.com.","jsmith.example.com.","2021093534","21600","3600","604800","86400"
"ns1.example.com","192.168.56.11","BIND AUTH","168.192.in-addr.arpa","ns1.example.com.example.com.","jsmith.example.com.","2021093207","21600","3600","604800","86400"
```

Network Utilization Report

```
twc rptv4netspaceutil --net_mask=16 --output_file=/tmp/output-network.csv --format=csv
```

```
"NetworkAddress","percentagefull","StaticObjects","DynamicObjects","ManualObjects","AutoObjects"
"192.168.0.0/16","0.11","18","51","0","0"
/tmp/output-network.csv (END)
```

Uploading Patch

You can upload the specified patch to the TCPWave IPAM.

Command

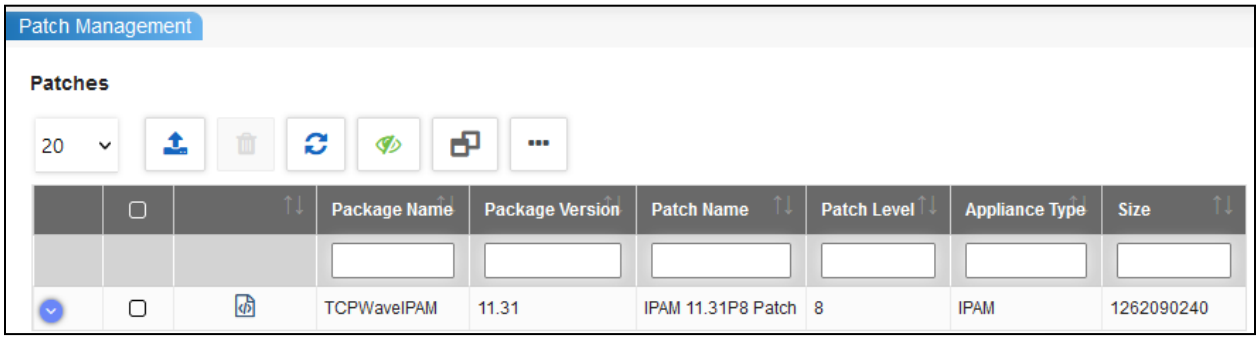
uploadpatch

Example Usage

```
twc uploadpatch --patch_file=/tmp/Patch_IPAM_11.31_8_FFFF.zip
```

Navigation

The system displays the uploaded patch in the Administration >> Patch Management grid.



Deploying Patch

You can deploy the uploaded patch.

Command

deploypatch

Example Usage

```
twc deploypatch --file_name=Patch_IPAM_11.31_8_FFFF.zip --
package_name=TCPWaveIPAM --major_version=11 --minor_version=31 --
patch_name="IPAM_11.31.8" --appliance_type=IPAM --patch_level=8
```

Output

```
2021-09-28 21:59:33.940 - The image file :  tims/IPAM/tmp/Patch_IPAM_11.31_8_FFFF/image.tar and install method: copy
2021-09-28 21:59:33.941 - Untar using command line : TAR  tims/IPAM/tmp/Patch_IPAM_11.31_8_FFFF/image.tar to  tims/IPAM/p
2021-09-28 21:59:33.941 - Untar using command line := tar xf  tims/IPAM/tmp/Patch_IPAM_11.31_8_FFFF/image.tar -C  tims/IP
2021-09-28 21:59:35.965 - Copying twcUpgrade_11_31_8_FFFF.tar.gz to  tims/IPAM/patches/twcUpgrade_11_31_8_FFFF.tar.gz
2021-09-28 21:59:37.038 - System patched successfully.
```

Conclusion

TCPWave's CLI helps the administrators manage the DDI activities flexibly, efficiently, and transparently.

TCPWave's DDI solution helps our customers manage and modernize their enterprise-grade solutions by ensuring they have the most innovative technology with minimal risks.
For a quick demo, contact the TCPWave Sales Team

Annexure

- **SADM:** Super Admin, Access to all the system's functionality, except user administration.
- **FADM:** Special Admin with Functional Account Privileges.
- **UADM:** User Admin, Access to User administration only.
- **NADM:** Normal Admin, Access to Object and Scope only.
- **PADM:** Power Admin, Access to Zone/Domain/Network/Subnet/Scope/Template/Object.
- **RADM:** Read-only Admin.
- **CADM:** Custom Admin.