Ubiquiti UniFi

Device CLI

Yes, the UniFi switches actually have a CLI... who knew! No you don't have to wait three minutes for the Unifi controller to update... □□

You can make configuration changes from the cli, but they will not be persistent across reboots or resyncs with the controller.

The commands below are from a USW-48-PoE (Standard). Other models may vary.

```
cli
# or
telnet localhost
```

```
show running-config
```

```
show mac address-table
show mac address-table interfaces GigabitEthernet 1
show mac address-table vlan 2
```

```
# wouldn't it be nice if Ubiquiti would actually share this information from the
controller!?!?!
show lldp neighbor
```

```
show interfaces GigabitEthernet 1 status
show interfaces GigabitEthernet 1-8 status
show interfaces GigabitEthernet 1-26 status
show interfaces GigabitEthernet 1-52 status

# example output
```

```
USW-48-G2# show interfaces GigabitEthernet 1-8
  <cr>
  protected  Configure an interface to be a protected port
  status     Port status information
JRN1A-SW2-USW-48-G2# show interfaces GigabitEthernet 1-52 status
Port  Name                 Status      Vlan  Duplex  Speed    Type
gi1   Port 1               notconnect  11    auto    auto     Copper
gi2   Port 2               connected   11    a-full  a-100M   Copper
gi3   Port 3               connected   11    a-full  a-1000M  Copper
gi4   Port 4               notconnect  11    auto    auto     Copper
gi5   Port 5               connected   11    a-full  a-1000M  Copper
gi6   Port 6               notconnect  11    auto    auto     Copper
gi7   Port 7               notconnect  11    auto    auto     Copper
gi8   Port 8               notconnect  11    auto    auto     Copper
```

```
show vlan
show vlan 2 interfaces GigabitEthernet 1-52 membership
```

```
# show 802.1x configuration for a port or multiple ports
show authentication interfaces GigabitEthernet 1
show authentication interfaces GigabitEthernet 1-24
show authentication interfaces GigabitEthernet 1-48
```

```
# show 802.1x authentication sessions
show authentication sessions

# example output with MAC Addresses redacted
USW-48-G2# show authentication  sessions
Total Session Number: 13

Interface  MAC Address       Type    Status       Session ID
---------- ----------------- ------- ------------ ----------------
gi3        XX:YY:ZZ:11:22:33 mac     Guest        0000000100015C5C
gi5        XX:YY:ZZ:11:22:33 mac     Guest        0000000200015C5C
gi26       XX:YY:ZZ:11:22:33 mac     Guest        0000000300015D10
gi40       XX:YY:ZZ:11:22:33 mac     Guest        0000000500015F0E
```

```
gi39       XX:YY:ZZ:11:22:33 mac     Guest        0000000700016256
gi34       XX:YY:ZZ:11:22:33 mac     Guest        0000000A00016DAA
gi18       XX:YY:ZZ:11:22:33 mac     Guest        0000000D00017930
gi2        XX:YY:ZZ:11:22:33 dot1x   Authorized   0000000E000180B0
gi43       XX:YY:ZZ:11:22:33 dot1x   Authorized   0000000F000188EE
gi20       XX:YY:ZZ:11:22:33 dot1x   Authorized   00000012003531F1
gi24       XX:YY:ZZ:11:22:33 dot1x   Authorized   00000013004F754C
gi28       XX:YY:ZZ:11:22:33 dot1x   Authorized   00000016006F8AB7
gi31       XX:YY:ZZ:11:22:33 dot1x   Authorized   0000001700731DA1
```

```
# configure
(config)# interface GigabitEthernet 34
(config-if)# shutdown
(config-if)# no shutdown
```

Random Stuff

List users in the UniFi Controller database

```
# show list of users in the unifi mongodb database
mongo --port 27117 ace --eval "db.admin.find().forEach(printjson);"
```

Change password for a UniFi Controller user

```
# change <UserName> to an actual user on the unifi controller
# the command will reset that user password to 'password'
mongo --port 27117 ace --eval 'db.admin.update( { "name" : "<UserName>" }, { $set : {
"x_shadow" :
"$6$GgQYRQnUs4wYkRd$7g6mig.les9salut9CZjUrG/UqqF6R/2RiCaCQEpEzz/7UtAtzeeQsVDnacAW1el2KH/jvUuJ4
Eh08xy.KGl0/" } } )'
```

[Decrypt a UniFi Controller backup]

Force dhcp to renew ip address

1. Get the PID of the udhcpc process
2. Send that process the USR1 signal which tells udhcpc to renew its IP address □□

```
# find the PID of the udhcpc process
ps | grep udhcpc
```

```
# output of the above command
 4052 admin     3480 S    /sbin/udhcpc -f -i eth0 -V ubnt -A 10 -s /etc/udhcpc/udhcpc -p
/var/run/udhcpc.eth0.pid
 6648 admin     3504 R    sh /usr/etc/syswrapper.sh ssh-trace-cmd -c ps | grep udhcpc -n 4 -i
 6650 admin     3480 R    grep udhcpc
```

```
# instruct udhcpc to renew its IP address by sending it's process the USR1 signal
kill -USR1 4052
```

Finding the site a device is connected to

```
# if you're using the Unifi OS Server, you need to enter the docker container first
uosserver shell

# Connect using the traditional mongo shell on UniFi's default port
mongo --port 27117

# find the site_id of a device
db.device.find({mac: "xx:xx:xx:xx:xx:xx"}, {site_id: 1, name: 2})

# find the description of the site
db.site.find({_id: ObjectId("YOUR_SITE_ID_HERE")}, {name: 1, desc: 1})
```

Manually removing items from mongo

```
# if you're using the Unifi OS Server, you need to enter the docker container first
uosserver shell

# Connect using the traditional mongo shell on UniFi's default port
mongo --port 27117
```

```
// ============================================================
// Once inside the shell, switch to the primary UniFi database
use ace



// ============================
// Force forget a ghost device

// For modern UniFi setups (MongoDB 5.0+)
db.device.deleteMany({"mac":"xx:xx:xx:xx:xx:xx"})
db.user.deleteMany({"mac":"xx:xx:xx:xx:xx:xx"})



// ======================
// Delete an Admin Profile

db.admin.deleteMany({"email":"admin@example.com"})



exit

// =============
// apply changes

// On Linux / Ubuntu
sudo systemctl restart unifi

// On Cloud Key / UDM
init 6 or restart via the OS Settings Console
```

:end

Revision #10
Created 2023-03-03 22:25:53 UTC by bluecrow76
Updated 2026-10-12 19:44:39 UTC by bluecrow76