

Unifi

- [Migrating to new Controller](#)
- [Setting up the UniFi Network Controller using Docker](#)
- [Unifi Console Commands](#)
- [Unifi Controller as a Windows Service](#)

Migrating to new Controller

1. Take a backup of the old server
2. Restore the backup onto the new server
3. on the old server set the override inform host option to point at the new controller
4. verify the devices have shown up in the new controller
5. shutdown the old controller

Setting up the UniFi Network Controller using Docker

<https://pimylifeup.com/unifi-docker/>

<https://community.ui.com/questions/How-To-Unifi-Contoller-Docker-home-network/fc6a9441-94b0-48b0-801a-6f38e2d4bb21>

Once you have Docker installed, our next step is to prepare your system to run the UniFi Network Controller container.

The main thing we need to do here is create a directory to store the UniFi containers Compose file and all of its data.

You can create this directory by running the following command within the terminal.

```
sudo mkdir -p /opt/stacks/unifi
```

[Copy](#)

By utilizing the “`-p`” option, this command will create any missing directories in the given path.

5. After creating a directory to store our Compose file, we can change to the directory [by using the `cd` command](#).

```
cd /opt/stacks/unifi
```

[Copy](#)

Writing a Compose file for the UniFi Docker Container

6. Our next step is writing the Compose file for the UniFi docker container. This Compose file will instruct Docker what image to download and the environment variables to pass through to it.

You can begin writing this file by typing in the following command.

```
sudo nano compose.yaml
```

[Copy](#)

7. Within this file, you will want to type in the following lines.

You must replace “<TIMEZONE>” with a valid TZ Identifier. You can find a list of [time zone identifiers from Wikipedia](#).

For example, we would use “Australia/Hobart” for our time zone.

```
services:
  unifi:
    user: unifi
    image: ghcr.io/jacobalberty/unifi-docker
    container_name: unifi-controller
    restart: unless-stopped
    ports:
      - "8080:8080"
      - "8443:8443"
      - "3478:3478/udp"
      - "10001:10001/udp"
    environment:
      TZ: "<TIMEZONE>"
    volumes:
      - ./data:/unifi
```

[Copy](#)

8. Once you have written the lines above, save and quit by pressing `CTRL + X`, followed by `Y`, and then the `ENTER` key.

Starting the UniFi Docker Container

9. With the Compose file written, all you need to do to bring the UniFi network controller Docker container online is to use the following command.

We use the “`-d`” option to detach from the current terminal session once the container has started.

```
docker compose up -d
```

[Copy](#)

Accessing the UniFi Network Controller Web Interface

10. Now that you have the Docker container for the UniFi network controller running, you will want to access its web interface. You will also need to know the IP address later in this guide.

If you don't know the local IP address of your machine, you can get it by running the following command.

```
hostname -I
```

[Copy](#)

Ensure you set up your device with a static IP address before proceeding further. You do not want your UniFi controller to be assigned a dynamic IP address.

11. Once you know the IP address of your device, go to the following address in your favourite web browser.

Ensure that you replace “<IPADDRESS>” with your IP.

```
https://<IPADDRESS>:8443
```

Initial Set Up of the UniFi Network Controller

12. When you first access your new Docker-based UniFi Network controller, you must complete some initial set up steps.

If you are setting up this controller as a brand new one, give it a server name, select your country, and accept the terms and conditions (**1.**).

After filling out your servers details, click the “[Next](#)” button (**3.**) to continue.

However, if you are restoring this from an older UniFi backup, click the “[Restore Server from a Backup](#)” link (**2.**)

[Name your Docker UniFi Network Controller or Restore from Backup](#)

13. If you are setting this controller up from scratch, you will now be asked to sign in to your Ubiquiti account (**1.**).

After filling out an email and password you can click the “[Create a UI Account](#)” button to create a new account, or you can click the “[Sign In](#)” button (**3.**) if you already have an account.

If you prefer not to use a Ubiquiti account on your controller, click the “[Advanced Setup](#)” link and follow the prompts (**2.**).

Sign in to UI Account

14. At this point, you should now have access to the UniFi network controller web interface.

However, before you can start adopting your Ubiquiti network devices, there is an option you must enable.

UniFi Network Controller Running in Docker Container

Enabling the Inform Host Option

15. For your Ubiquiti devices to detect your Docker-based UniFi network controller, we must enable the “[Inform Host](#)” option.

This option lets us tell the IP address the controller should broadcast to available devices. Currently, it will be using the Dockers internal IP rather than the host IP.

To set this option, you must first switch to the settings page by clicking the cog in the sidebar.

Open Settings page

16. Once you are on the settings screen, change to the “[System](#)” menu (**1.**).

After changing to the system settings page, swap to the “[Advanced](#)” tab (**2.**).

Change to Advanced Settings

17. On this page, you should find an option labeled “[Inform Host](#)”.

Click the checkbox to allow us to begin overriding the host IP.

Toggle on Inform Host

18. Before you can override the inform host value, you will get a warning that misconfiguring this option can cause you to lose access to your devices. You may have to physically reset your Ubiquiti devices if something goes wrong.

You need to ensure that the IP address of your host is correct and reachable. If you are happy to proceed, click the “[Confirm](#)” button.

Confirm you want to Inform Host

19. A text box will now appear below the “`Inform Host`” option. Type in the local IP address (**1.**) of the machine on which you are running the UniFi network controller using Docker.

Once you are happy that you have typed in the correct IP address, click the “`Apply Changes`” button (**2.**).

Type in `HOST IP For UniFi Docker Container Machine`

20. Once you have applied your settings, you must restart the UniFi Docker container.

Back in the terminal, you can bring the UniFi container down by using the following command

```
docker compose down
```

[Copy](#)

21. Start UniFi back up on your device by running the following command within the terminal.

```
docker compose up -d
```

[Copy](#)

Updating your UniFi Docker Container

One massive advantage of using the UniFi network controller through a Docker container is that updating it is straightforward.

The following steps will walk you through the effortless way of updating to the latest version. You can see what versions are available by [visiting the GitHub page for this project](#).

1. To update the UniFi container, we must change to the directory we created earlier in this guide.

We need to use the Compose file we wrote to pull the latest version of the image.

```
cd /opt/stacks/unifi
```

[Copy](#)

2. After swapping to the directory where we wrote the Compose file, you can use the command below to download the latest version of the UniFi container.

```
docker compose pull
```

[Copy](#)

3. Finally, if a new version is found, you can get Docker to move over to it using the command below in the terminal.

Docker will detect the new image and restart the container using it.

```
docker compose up -d
```

Unifi Console Commands

Command	Example	Function
info	<code>info</code>	Displays device information
set-default	<code>set-default</code>	Factory reset device
set-inform	<code>set-inform http://192.168.1.1:8080/inform</code>	Set URL of the controller for adoption. More info
upgrade	<code>upgrade https://<firmware-url>.bin</code>	Upgrade firmware – More info
fwupdate	<code>fwupdate --url https://<firmware- url>.bin</code>	Update firmware
reboot	<code>reboot</code>	Reboot the device
poweroff	<code>poweroff</code>	Shutdown device
uptime	<code>uptime</code>	Shows device uptime

Network related SSH Commands

The following Unifi SSH Commands can really help you with finding network-related issues with your Unifi Device.

Command	Example	Function
ifconfig	<code>ifconfig</code>	Show network interface information
ip address add	<code>ip address add 192.168.1.143/24 dev br0</code>	Set static IP Address
ip route	<code>ip route</code>	Display current gateway
ip router add	<code>ip route add default via 192.168.1.1</code>	Set default gateway
	<code>echo "nameserver 192.168.1.1" > /etc/resolv.conf</code>	Set DNS Server
ping	<code>ping 1.1.1.1</code>	Check network connection to device
arp	<code>arp -a</code>	Show arp table
ip neigh	<code>ip neigh</code>	Show IPv6 neighbors

Unifi OS SSH Commands

When you connect to your UDM Pro (or another controller that is running Unifi OS), then you will have a couple of other options:

Command	Example	Function
ubnt-systool help	ubnt-systool help	Show all commands
ubnt-systool cputemp	ubnt-systool cputemp	Show CPU Temp
ubnt-systool cpuload	ubnt-systool cpuload	Show CPU load
ubnt-systool portstatus	ubnt-systool portstatus	Show port status
ubnt-systool hostname	ubnt-systool hostname <newname>	Set new hostname
ubnt-systool reboot	ubnt-systool reboot	Reboot device
ubnt-systool poweroff	ubnt-systool poweroff	Shutdown device
ubnt-systool reset2defaults	ubnt-systool reset2defaults	Factory reset device
ubnt-device-info summary	ubnt-device-info summary	Show system information
ubnt-tools ubnt-discover	ubnt-tools ubnt-discover	Show Unifi devices in the network
cat /mnt/data/udapi-config/dnsmasq.lease	cat /mnt/data/udapi-config/dnsmasq.lease	Show DHCP Leases
cat /mnt/data/udapi-config/unifi	cat /mnt/data/udapi-config/unifi	Show configuration
/etc/init.d/S95unifios restart	/etc/init.d/S95unifios restart	Restart Unifi OS Web interface

Unifi Log files

There are a lot of log files that you can access to help you debug any Unifi related problem:

Command	Function
<code>cat /var/log/messages</code>	Output the error log
<code>tail -f /var/log/messages</code>	Monitor log file
<code>cat /mnt/data/unifi-os/unifi-core/config/settings.yaml</code>	Server settings
<code>cat /mnt/data/unifi-os/unifi-core/logs/discovery.log</code>	Discovery log
<code>cat /mnt/data/unifi-os/unifi-core/logs/system.log</code>	System log
<code>cat /mnt/data/unifi-os/unifi/logs/server.log</code>	Server log
<code>cat /mnt/data/unifi-os/unifi-core/logs/errors.log</code>	Http errors

Unifi SSH

UniFi Controller as a Windows Service

How to Set Up the UniFi Network Controller as a Windows Service

1. Close any instances of the UniFi software on the computer. If the UniFi Network Controller was just installed, make sure to open the controller software manually at least once, or let it run at the end of the wizard. Once you see the message UniFi Controller (a.b.c) started, the controller may be closed.

2. Launch the Command Prompt as an Administrator. On Windows 10 this would entail right-clicking and selecting "Run as administrator". Please see Windows documentation to identify the appropriate method for your specific Windows version.

3. Change directory to the location of UniFi in the computer using the following command (exactly as it is here, no need to substitute anything):

```
cd "%UserProfile%\Ubiquiti UniFi\"
```

4. Once in the root of the UniFi folder, issue the following command to install the UniFi Network Controller service:

```
java -jar lib\ace.jar installsvc
```

5. Once you're at a new command prompt line, after it says "Complete Installation...", issue the following:

```
java -jar lib\ace.jar startsvc
```

6. Close the command prompt window either by entering the exit command, or clicking the X.

7. To access the UniFi Network Controller now open a browser and go to <https://localhost:8443>. Alternatively, the desired interface IP, or FQDN that is mapped to that host (in place of "localhost") may be used.

NOTE: We recommend using Oracle Java JRE 8, which can be downloaded in Oracle's website. For the UniFi Cloud Access portal (<https://network.unifi.ui.com/>) to work properly, only use x64 Java.

How to Upgrade a UniFi Network Controller that is Running as a Windows Service

1. Open the UniFi Network Controller and make a backup.

2. Launch the Command Prompt as an Administrator. On Windows 10 this would entail right-clicking and selecting "Run as administrator". Please see Windows documentation to identify the appropriate method for your specific Windows version.

3. Change directory to the location of UniFi in the computer using the following command (exactly as it is here, no need to substitute anything):

```
cd "%UserProfile%\Ubiquiti UniFi\"
```

4. Once in the root of the UniFi folder, issue the following to stop and uninstall the UniFi Network Controller service:

```
java -jar lib\ace.jar stopsvc
```

```
java -jar lib\ace.jar uninstallsvc
```

5. Once a new command prompt line is presented, that means uninstalling the service was successful. Update the UniFi Network Controller application's version with any of the regular methods. Go to Settings > Maintenance in the Controller to upgrade to the latest public released version, or alternatively download the UniFi Network Controller installation package for Windows

from the Downloads page or the Releases section of the Community.

6. Once the software update is successful, close the UniFi Network Controller application before going back to the Command Prompt.

7. In the Command Prompt issue the following to install the UniFi Network Controller service once again:

```
java -jar lib\ace.jar installsvc
```

8. Once you're at a new command prompt line, after seeing the "Complete Installation..." message, issue the following:

```
java -jar lib\ace.jar startsvc
```

9. Close the Command Prompt window either by entering the exit command, or clicking the X.

10. To access the UniFi Network Controller now open a browser and go to <https://localhost:8443>. Alternatively, the desired interface IP, or FQDN that is mapped to that host (in place of "localhost") may be used.

Unifi