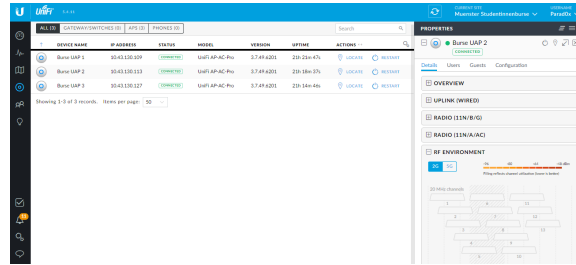


Ubiquiti Unifi-Controller im Freifunk Münsterland - Netzwerk

Inhalt

- Über die Software
- Installation
 - Hardware
- Software
- Nutzung
 - IP-Adressen / Zugriff unifi
 - Hauptserver
 - IP-Adressen / Zugriff unifi-legacy
- Let's-Encrypt-Zertifikat
- Performance-Anpassung



Über die Software

Bei einigen komplexen Umgebungen kann es Sinn ergeben neben den „üblichen“ Routern mit Freifunk-Software (Gluon) auch Geräte mit Original-Firmware der Hersteller zu nutzen. Solche Geräte sollten regelmäßig Updates erhalten. Unter anderem hierfür kann ein zentrales Management genutzt werden. Für einige Geräte der Firma Ubiquiti bietet sich hier die kostenfreie Software "UniFi Controller" an.

Von UniFi-Controller unterstützte Hardware:

- Alle Geräte mit UniFi im Namen
- Picostation ~~Wird nicht mehr unterstützt~~
unter <https://help.ubnt.com/hc/en-us/articles/205146010-UniFi-Convert-PicoStation-M2-HP-to-run-UniFi> kann eine Firmware heruntergeladen werden die man unter AirOS einspielt und die Picostation danach mit einem Unifi Controller konfigurierbar ist (getestet von dippydipp 06/2017)

Diese zentrale Software ist ein Hilfsmittel. Wir Freifunker finden Dezentralität gut. Daher sollen gerne auch eigene Management-Lösungen genutzt werden. Wir unterstützen auch beim Aufbau eigener Systeme. Wer nur wenige Geräte hat oder keine Möglichkeit für ein eigenes Management hat, der kann gerne das zentrale System mitbenutzen.

Am 24.10.2018 wurde ein zweiter Server unifi-legacy installiert, damit EOL-Geräte auf diesen verschoben werden können. Erst danach konnte der Hauptserver auf die aktuelle Version aktualisiert werden.

Installation

Hardware

Unser Server hat nur eine Netzwerkkarte. Diese ist mit dem Freifunk-Netzwerk verbunden.

```
<domain type='kvm'>
  <name>unifi</name>
  <uuid>77b1989e-37d9-4f20-a2e9-bd1152a8208f</uuid>
  <memory unit='KiB'>4194304</memory>
  <currentMemory unit='KiB'>4194304</currentMemory>
  <vcpu placement='static'>2</vcpu>
  <os>
    <type arch='x86_64' machine='pc-i440fx-xenial'>hvm</type>
    <boot dev='hd' />
  </os>
  <features>
    <acpi/>
    <apic/>
  </features>
  <cpu mode='custom' match='exact'>
    <model fallback='allow'>IvyBridge</model>
  </cpu>
  <clock offset='utc'>
    <timer name='rtc' tickpolicy='catchup' />
    <timer name='pit' tickpolicy='delay' />
    <timer name='hpet' present='no' />
  </clock>
</domain>
```

```
<on_poweroff>destroy</on_poweroff>
<on_reboot>restart</on_reboot>
<on_crash>restart</on_crash>
<pm>
  <suspend-to-mem enabled='no' />
  <suspend-to-disk enabled='no' />
</pm>
<devices>
  <emulator>/usr/bin/kvm-spice</emulator>
  <disk type='file' device='disk'>
    <driver name='qemu' type='raw' cache='none' />
    <source file='/home/libvirt/images/ffms/unifi.raw' />
    <target dev='vda' bus='virtio' />
    <address type='pci' domain='0x0000' bus='0x00' slot='0x07' function='0x0' />
  </disk>
  <controller type='usb' index='0' model='ich9-ehci1'>
    <address type='pci' domain='0x0000' bus='0x00' slot='0x06' function='0x7' />
  </controller>
  <controller type='usb' index='0' model='ich9-uhci1'>
    <master startport='0' />
    <address type='pci' domain='0x0000' bus='0x00' slot='0x06' function='0x0' multifunction='on' />
  </controller>
  <controller type='usb' index='0' model='ich9-uhci2'>
    <master startport='2' />
    <address type='pci' domain='0x0000' bus='0x00' slot='0x06' function='0x1' />
  </controller>
  <controller type='usb' index='0' model='ich9-uhci3'>
    <master startport='4' />
    <address type='pci' domain='0x0000' bus='0x00' slot='0x06' function='0x2' />
  </controller>
  <controller type='pci' index='0' model='pci-root' />
  <controller type='virtio-serial' index='0'>
    <address type='pci' domain='0x0000' bus='0x00' slot='0x05' function='0x0' />
  </controller>
  <interface type='network'>
    <mac address='52:54:00:14:bb:9f' />
    <source network='Clientnetz-Dom16' />
    <model type='virtio' />
    <address type='pci' domain='0x0000' bus='0x00' slot='0x03' function='0x0' />
  </interface>
  <serial type='pty'>
    <target port='0' />
  </serial>
  <console type='pty'>
    <target type='serial' port='0' />
  </console>
  <channel type='spicevmc'>
    <target type='virtio' name='com.redhat.spice.0' />
    <address type='virtio-serial' controller='0' bus='0' port='1' />
  </channel>
  <input type='tablet' bus='usb' />
  <input type='mouse' bus='ps2' />
  <input type='keyboard' bus='ps2' />
  <graphics type='spice' autoport='yes'>
    <image compression='off' />
  </graphics>
  <sound model='ich6'>
    <address type='pci' domain='0x0000' bus='0x00' slot='0x04' function='0x0' />
  </sound>
  <video>
    <model type='qxl' ram='65536' vram='65536' vgamem='16384' heads='1' />
    <address type='pci' domain='0x0000' bus='0x00' slot='0x02' function='0x0' />
  </video>
  <redirdev bus='usb' type='spicevmc'>
</redirdev>
  <redirdev bus='usb' type='spicevmc'>
</redirdev>
  <memballoon model='virtio'>
    <address type='pci' domain='0x0000' bus='0x00' slot='0x08' function='0x0' />
  </memballoon>
</devices>
```

```
</domain>
```

Software

Die Installation ist per APT recht einfach. Siehe Schritt für Schritt unter <https://help.ubnt.com/hc/en-us/articles/220066768-UniFi-How-to-Install-Update-via-APT-on-Debian-or-Ubuntu>

Die Installation wurde umgestellt auf das Script von Glenn: <https://community.ui.com/questions/UniFi-Installation-Scripts-or-UniFi-Easy-Update-Script-or-UniFi-Lets-Encrypt-or-UniFi-Easy-Encrypt-/ccbc7530-dd61-40a7-82ec-22b17f027776>

Nutzung

Im Freifunk-Münsterland-Netz wird ein Unifi-Controller als Service zur Verfügung gestellt.

Es gibt verschiedene sogenannte Sites (Standorte bzw. Gruppierung von Geräten). Jedem dieser Standorte können Benutzer hinzugefügt werden. Für deine eigene Site sprich gerne [Sebastian Danek](#) oder einen der anderen Administratoren an oder komme zu einem unserer Treffen.

IP-Adressen / Zugriff unifi Hauptserver

- <https://unifi.ffmsl.de:8443> (Sowohl V6 direkt, als auch V4 über Proxy)
- <https://10.43.128.15:8443> (nur im Freifunk nutzbar, wirft Zertifikatwarnung)
- [https://\[2a03:2260:115:4300:5054:ff:fe14:bb9f\]:8443](https://[2a03:2260:115:4300:5054:ff:fe14:bb9f]:8443) (ungültiges Zertifikat)
- <https://unifi.freifunk-muensterland.de:8443> (alle FF-Münsterland TLDs funktionieren)

IP-Adressen / Zugriff unifi-legacy

- <https://unifi-legacy.ffmsl.de:8443> (Sowohl V6 direkt, als auch V4 über Proxy)
- <https://10.43.128.15:8443> (nur im Freifunk nutzbar, wirft Zertifikatwarnung)
- [https://\[2a03:2260:115:4300:5054:ff:fefe:680f\]:8443](https://[2a03:2260:115:4300:5054:ff:fefe:680f]:8443) (ungültiges Zertifikat)
- <https://unifi-legacy.freifunk-muensterland.de:8443> (alle FF-Münsterland TLDs funktionieren)

Let's-Encrypt-Zertifikat

Neben dem normalen acme.sh-Certbot braucht man noch ein Skript, was den Import in den Controller übernimmt.

```
wget https://source.sosdg.org/brielle/lets-encrypt-scripts/raw/master/gen-unifi-cert.sh -O /root/bin/gen-unifi-cert.sh
```

Dieses Skript prüft automatisch, ob eine Erneuerung erforderlich ist und stoppt dann ggfs. automatisch den Controller. Der Aufruf erfolgt so:

```
/root/bin/gen-unifi-cert.sh -e info@freifunk-muensterland.de -d unifi.freifunk-muenster.de -d unifi.ffmsl.de -d unifi.freifunk-muensterland.de -d unifi.freifunk-muensterland.org -d unifi.freifunk-muensterland.net
```

Das ACME- und das Unifi-Cert-Skript benötigen einen Cronjobeintrag:

```
#1 0 * * 1 "/root/.acme.sh"/acme.sh --cron --home "/root/.acme.sh" > /dev/null
16 0 * * 1 /root/bin/gen-unifi-cert.sh -e info@freifunk-muensterland.de -d unifi.freifunk-muenster.de -d unifi.ffmsl.de -d unifi.freifunk-muensterland.de -d unifi.freifunk-muensterland.org -d unifi.freifunk-muensterland.net
```

Performance-Anpassung

Am 11.07.2018 haben wir einige Anpassungen an der Datei system.properties (Siehe <https://help.ubnt.com/hc/en-us/articles/205202580>) vorgenommen.

Als Grundlage haben wir die Tipps von der Seite <https://help.ubnt.com/hc/en-us/articles/115005159588-UniFi-How-to-Tune-the-Controller-for-High-Number-of-UniFi-Devices> genommen.

/var/lib/unifi/system.properties

```
unifi.xmx=3072
unifi.xms=3072
unifi.G1GC.enabled=true
inform.num_thread=500
inform.max_keep_alive_requests=100
```