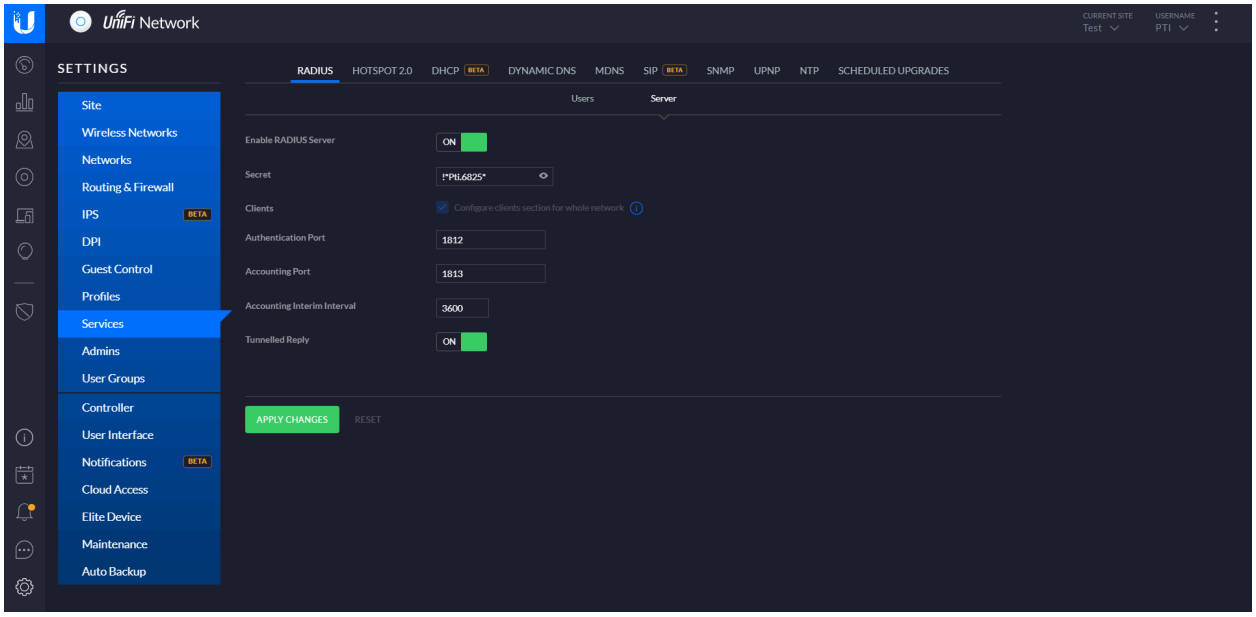
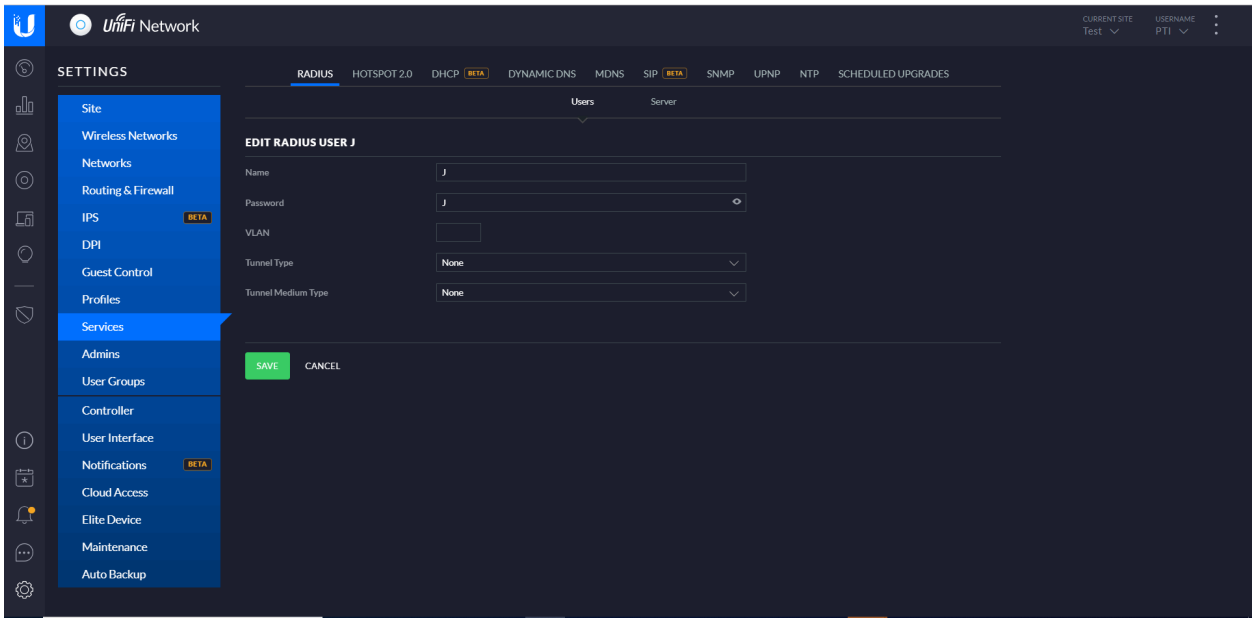
Instructions for OpenVPN Creation and Configuration (RADIUS Auth)

1. Before even using ssh to set the settings; Utilize the Unifi Controller tool to configure everything on your network except for the VPN. (You can add devices as you please, just ensure you've added all the networks you are going to add)
2. Open your unifi controller and login, then go to "settings > services > radius >server". Server should be a sub-menu towards the top middle of the screen.
3. Give your RADIUS server the settings shown in the image below (You can use whatever you want for your secret, but in this example we are going to be using '!*Pti.6825*'):



4. Click on the Users tab and add a user. Don't worry about a VLAN, VPN tunnel type, or VPN tunnel medium type as those settings do not matter. You only need a username and a password such as this:



5. You are now done with the controller tool and need to open up putty.exe
6. SSH to your firewall's address, in our case it is 192.168.1.1 . If you are having trouble remembering your username and password they can be found in your Unifi Controller settings under "settings > Sites" towards the bottom of the page.
7. Enter the Following bolded commands:

sudo bash

curl –O http://ftp.us.debian.org/debian/pool/main/e/easy-rsa/easy-rsa_2.2.2-1_all.deb

This downloads the package needed to make RSA certificates

sudo dpkg -i easy-rsa_2.2.2-1_all.deb

This command installs the program package

cd /usr/share/easy-rsa

#This navigates to the correct directory in the firewall

. vars

#Run the vars program which we will use to make our certificates

./clean-all

./build-ca

Give a common name such as "OpenVPN CA"

./build-key-server server

#In this example we set the common name to "server"

#Type yes when asked to sign your certificate and commit the configuration

./build-dh

#Go get lunch this step takes a LONG time.

#once it is done there will be no more dots generated

mkdir /config/auth/keys/

cp keys/* /config/auth/keys/

#This copies the files you just generated into the correct directory

#After copying the keys it's time to configure the firewall a bit

configure

set interfaces openvpn vtun0 mode server

set interfaces openvpn vtun0 server subnet 10.1.1.0/24

set interfaces openvpn vtun0 tls ca-cert-file /config/auth/keys/ca.crt

set interfaces openvpn vtun0 tls cert-file /config/auth/keys/server.crt

set interfaces openvpn vtun0 tls key-file /config/auth/keys/server.key

set interfaces openvpn vtun0 tls dh-file /config/auth/keys/dh2048.pem

set interfaces openvpn vtun0 encryption aes128

set interfaces openvpn vtun0 openvpn-option "--keepalive 8 30"

set interfaces openvpn vtun0 openvpn-option "--comp-lzo"

set interfaces openvpn vtun0 openvpn-option "--duplicate-cn"

set interfaces openvpn vtun0 openvpn-option "--user nobody --group nogroup"

set interfaces openvpn vtun0 openvpn-option "--plugin /usr/lib/openvpn/openvpn-auth-pam.so openvpn"

set interfaces openvpn vtun0 openvpn-option "--client-cert-not-required --username-as-common-name"

set interfaces openvpn vtun0 openvpn-option "--verb 1"

set interfaces openvpn vtun0 openvpn-option "--proto udp6"

set interfaces openvpn vtun0 openvpn-option "--port 1194"

set interfaces openvpn vtun0 openvpn-option "--push redirect-gateway def1"

set interfaces openvpn vtun0 openvpn-option "--push dhcp-option DNS 8.8.8.8"

set interfaces openvpn vtun0 openvpn-option "--push dhcp-option DNS 8.8.4.4

set firewall name WAN_LOCAL rule 20 action accept

set firewall name WAN_LOCAL rule 20 description "Allow OpenVPN clients in”

set firewall name WAN_LOCAL rule 20 destination port 1194

set firewall name WAN_LOCAL rule 20 log disable

set firewall name WAN_LOCAL rule 20 protocol udp

set service nat rule 5010 description "Masquerade for WAN"

set service nat rule 5010 outbound-interface eth0

set service nat rule 5010 type masquerade

#Next you save your work or suffer the consequences!!

commit

save

exit

#Next you need to generate a .ovpn file for things to work clientside

-When generating the .ovpn file please use the “OVPN RADIUS Template.ovpn” file found in teams. This Template follows this how-to exactly so you can use it.

-If you don’t know how to open and copy your certificate block, use the following commands:

cd /config/auth/keys/

vi ca.crt

#At this time you should drag and click to copy, and use your keys to navigate ONLY, if you do not, then look up vim commands for assistance. Once you are done with this certificate use the following command:

:wq

8. Now you need to make sure your firewall can contact the radius server. Follow the bolded commands:
   cd /etc/
   vi pam_radius_auth.conf
9. Now you create the ‘pam_radius_auth.conf’ script. The script should have the following bolded info:
   <yourfirewall’s IP> <your PSK>
10. Save the script and enter the following command to prepare your next script:

cd /pam.d

vi openvpn

11. Now enter the bolded info:
    auth sufficient pam_radius_auth.so debug
    account sufficient pam_permit.so
    session sufficient pam_permit.so

12. Now save your work and test your VPN connection
13. Make your changes persistent within the firewall by entering the following commands:
    mkdir /config/scripts/openvpnconfiguration/
    cp /etc/pam_radius_auth.conf /config/scripts/openvpnconfiguration
    cp /etc/pam.d/openvpn/ /config/scripts/openvpnconfiguration
    vi /config/scripts/postprovision.sh
14. Now enter the following script in bold:

```
#!/bin/vbash

readonly logFile="/var/log/postprovision.log"

cp /config/scripts/openvpnconfiguration/pam_radius_auth.conf /etc

cp /config/scripts/openvpnconfiguration/openvpn /etc/pam.d/openvpn

source /opt/vyatta/etc/functions/script-template

configure > ${logFile}

delete system task-scheduler task postprovision >> ${logFile}

commit >> ${logFile}

save >> ${logFile}

exit
```

15. Now make that script an executable:

```
chmod +x /config/scripts/postprovision.sh
```