Technical Bulletin CS-20-09

Migrating CMS Data Between CMS Versions

This technical bulletin explains how to migrate your CMS database to a new version of CMS or a new build of CMS to avoid losing any device or system configuration.

System Affected

CMS Versions 2.4 and higher

Description

CMS has built in functions that allow you to backup and restore your database between versions. This mechanism helps to simplify the process of migration and helps to ensure that the important pieces of your configuration are retained. Keep in mind that each CMS has a few unique values; i.e. the account key and certificate. These values are all included in your backup file and will be restored into the CMS upon running the import.sh script that we'll discuss shortly.

The New CMS

The obvious first step is to build your new CMS server. You'll need to have a platform on which to install the backup file you take from your old CMS, so this seems like a good place to start. The server specifications should mirror those of your old server unless you have reason to add more RAM or increase the number CPU's available to the server. If you have questions about whether that is necessary, please reach out to Spectralink Technical Support to discuss. Most installations that require such changes are ones that are experiences CMS UI connectivity issues due to system load or other design considerations.

You can obtain the new OVF file for the CMS from the Spectralink Support Website, https://support.spectralink.com/cms, and use that to start your new server build. You won't be able to use the patch that may be available if you're standing up a new server as the patches are only relevant for existing server platforms that are being updated.

Note

You cannot patch between major OS versions with CMS. Between CMS version 2.5 to 2.6 the OS was changed from Ubuntu 14 to Ubuntu 16. As a result, you cannot simply patch a 2.5 version of CMS to get to 2.6. You must stand up a new version of CMS.

Note

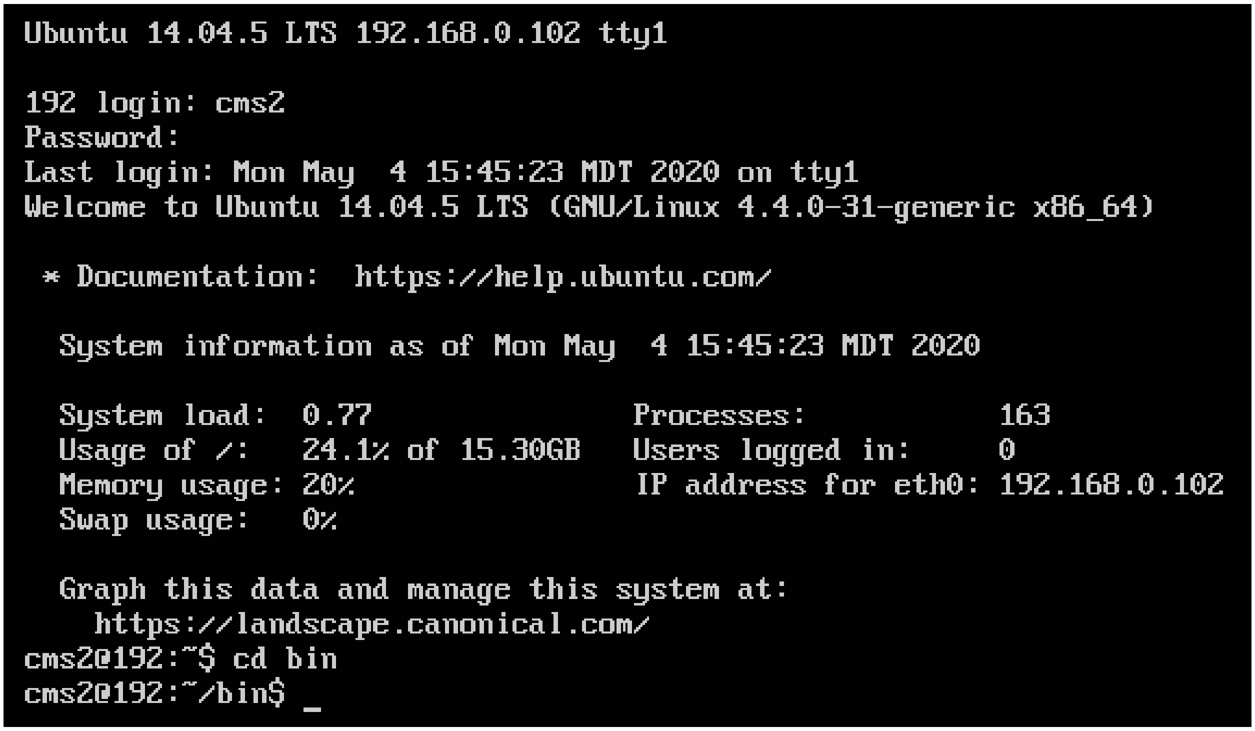
If you are upgrading from the current GA version of 2.5.0 CMS and you have 8400 handsets; it is critical that you first upgrade your CMS to version 2.5.2.1223 via the available patch. This will ensure your configuration files export properly.

Warning

It's important to remember that CMS version 2.6 and later do not support Spectralink PIVOT handsets.

We're not going to cover the steps for actually installing the OVF here since that's covered quite well in the CMS Administration Guide, which you can also get from the same site as the OVF. Instead, we'll jump straight into the details of how this all works.

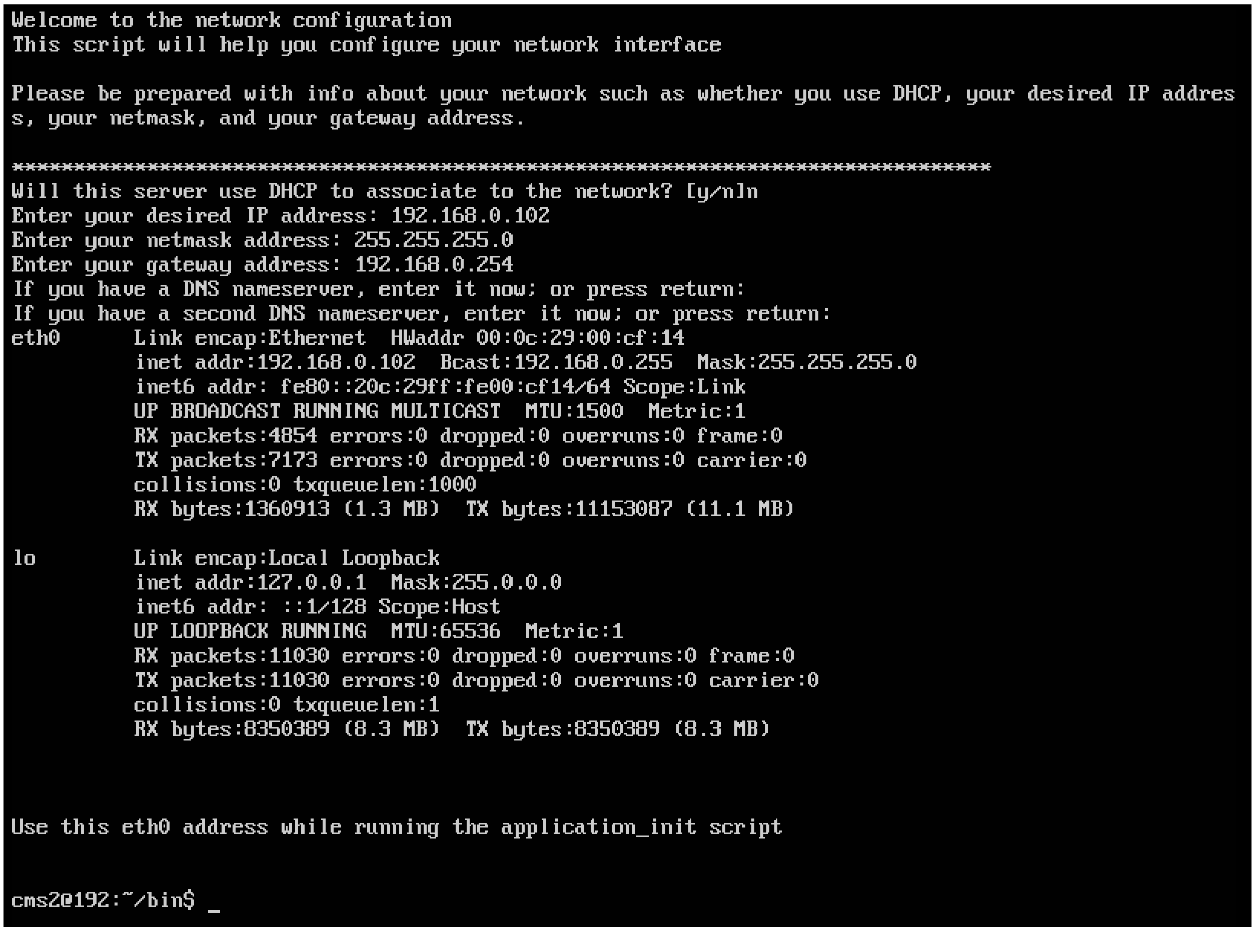
Let's assume you've got your OVF installed, the new CMS is booted up and you're sitting at the login prompt. Go ahead and log in using the default login credentials, cms2/cms2. Then we'll navigate to the bin folder so we can start the initial configuration process.



```
Ubuntu 14.04.5 LTS 192.168.0.102 tty1

192 login: cms2
Password:
Last login: Mon May  4 15:45:23 MDT 2020 on tty1
Welcome to Ubuntu 14.04.5 LTS (GNU/Linux 4.4.0-31-generic x86_64)

 * Documentation:  https://help.ubuntu.com/

  System information as of Mon May  4 15:45:23 MDT 2020

  System load:  0.77              Processes:           163
  Usage of /:   24.1% of 15.30GB  Users logged in:     0
  Memory usage: 20%               IP address for eth0: 192.168.0.102
  Swap usage:   0%

  Graph this data and manage this system at:
    https://landscape.canonical.com/
cms2@192:~$ cd bin
cms2@192:~/bin$ _
```

Now we can run the initialization scripts to setup the new CMS. But first, let's go over what's going to happen. When you perform a CMS migration, you still have to build the new CMS as though it were going to be a brand new CMS with its own account key, certificate and so on. This is because we will need to be able to access the web UI of this new CMS in order to upload the backup from the old CMS. Once we load the backup file and run the import script it is going to overwrite the database, which includes the account key and certificate; with the data from the old CMS server. Then we will just need to change the server's IP Address, potentially, to match the old CMS and verify the data set. But let's jump into this and we'll go through those steps shortly.

Let's start by running the network initiatlization script, network_init.py, to get the IP Address for this new server setup. Issue the command, sudo python network_init.py and press enter.



```
Welcome to the network configuration
This script will help you configure your network interface

Please be prepared with info about your network such as whether you use DHCP, your desired IP addres
s, your netmask, and your gateway address.

********************************************************************************
Will this server use DHCP to associate to the network? [y/n]n
Enter your desired IP address: 192.168.0.102
Enter your netmask address: 255.255.255.0
Enter your gateway address: 192.168.0.254
If you have a DNS nameserver, enter it now; or press return:
If you have a second DNS nameserver, enter it now; or press return:
eth0      Link encap:Ethernet  HWaddr 00:0c:29:00:cf:14
          inet addr:192.168.0.102  Bcast:192.168.0.255  Mask:255.255.255.0
          inet6 addr: fe80::20c:29ff:fe00:cf14/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:4854 errors:0 dropped:0 overruns:0 frame:0
          TX packets:7173 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:1360913 (1.3 MB)  TX bytes:11153087 (11.1 MB)

lo        Link encap:Local Loopback
          inet addr:127.0.0.1  Mask:255.0.0.0
          inet6 addr: ::1/128 Scope:Host
          UP LOOPBACK RUNNING  MTU:65536  Metric:1
          RX packets:11030 errors:0 dropped:0 overruns:0 frame:0
          TX packets:11030 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1
          RX bytes:8350389 (8.3 MB)  TX bytes:8350389 (8.3 MB)



Use this eth0 address while running the application_init script


cms2@192:~/bin$ _
```

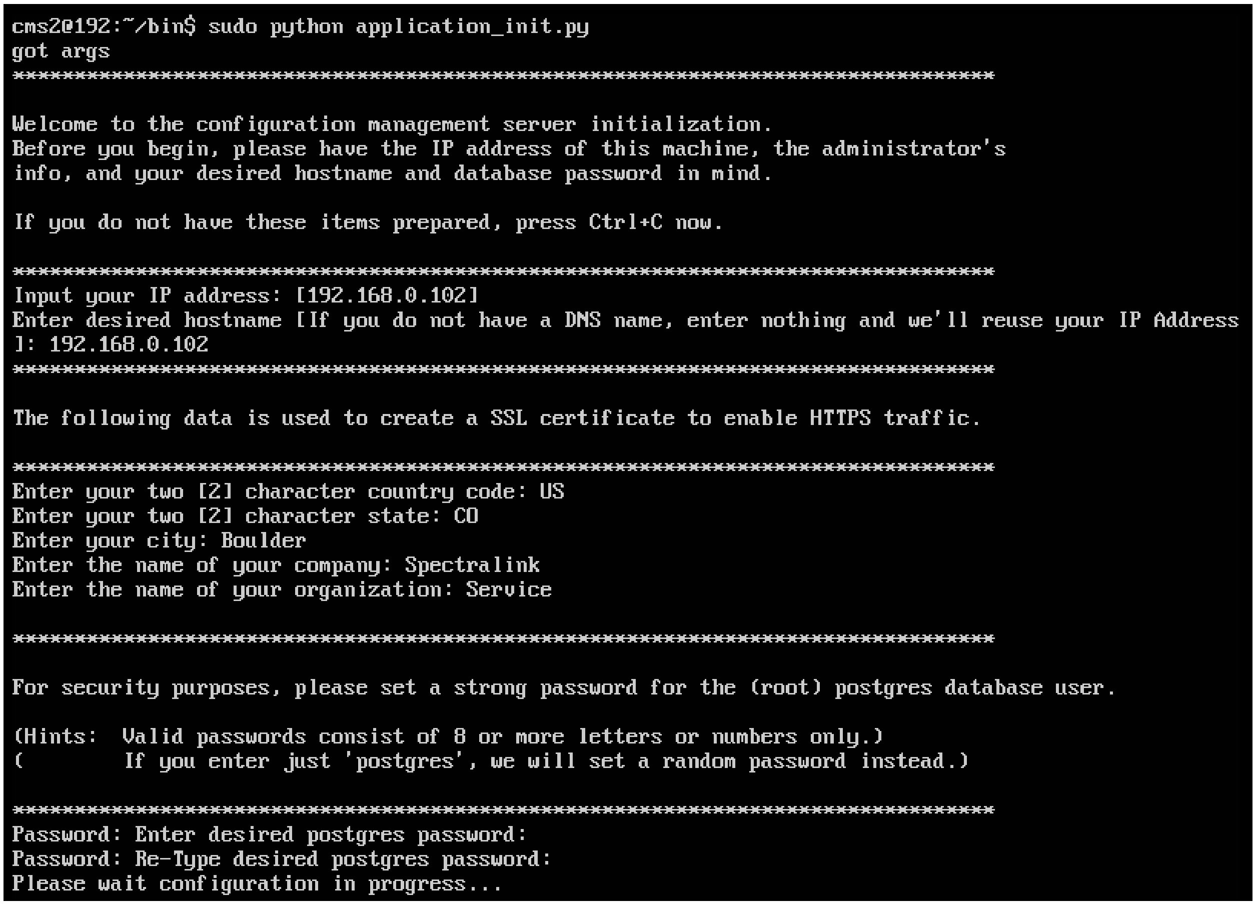
Walk through the prompts and enter the information as appropriate. If your old CMS is using a static IP Address, which it really should be, then you'll want to continue using a static IP address.

Note

You have a couple of options at this point. If you have already pulled the backup from your old CMS, then you could just power it down now and reuse the IP address on the new CMS now. Or, you can assign a different IP address to this new CMS now to allow access to it and the old CMS and we'll change the IP Address of the new CMS to match the old CMS at the end of the process.

We're going to assume that you will be keeping your old CMS active for now so let's assign a new IP Address to this CMS server for now. We'll go over how to change it to match the old CMS at the end of this migration process. So, walk through the prompts and enter your new IP Address, netmask and gateway. Don't bother to enter DNS server information right now, even if it is present on the other CMS. Once you've entered all the information, the script will complete and configure the Ethernet interface of the CMS server and we can move onto the next step.

Now we need to initialize the application so we can gain access to the web UI of the CMS. Enter the command, sudo python application_init.py and press enter.



```
cms2@192:~/bin$ sudo python application_init.py
got args
********************************************************************************

Welcome to the configuration management server initialization.
Before you begin, please have the IP address of this machine, the administrator's
info, and your desired hostname and database password in mind.

If you do not have these items prepared, press Ctrl+C now.

********************************************************************************
Input your IP address: [192.168.0.102]
Enter desired hostname [If you do not have a DNS name, enter nothing and we'll reuse your IP Address
]: 192.168.0.102
********************************************************************************

The following data is used to create a SSL certificate to enable HTTPS traffic.

********************************************************************************
Enter your two [2] character country code: US
Enter your two [2] character state: CO
Enter your city: Boulder
Enter the name of your company: Spectralink
Enter the name of your organization: Service

********************************************************************************

For security purposes, please set a strong password for the (root) postgres database user.

(Hints:  Valid passwords consist of 8 or more letters or numbers only.)
(        If you enter just 'postgres', we will set a random password instead.)

********************************************************************************
Password: Enter desired postgres password:
Password: Re-Type desired postgres password:
Please wait configuration in progress...
```

You'll enter the IP Address you chose in the first script and then go ahead and just enter the IP Address again. Even if your other CMS uses a hostname, we'll not need it yet. So just enter the IP Address again for now. Complete each of the prompts to generate the new certificate, again, we’re not really going to use this certificate so it doesn’t much matter what you put in these fields but they must be valid values or the script will fail. For the Postgres password, try to enter the same password you used on the old CMS. However, if you do not remember the password or your old password does not work, don’t worry too much and just choose a new password.

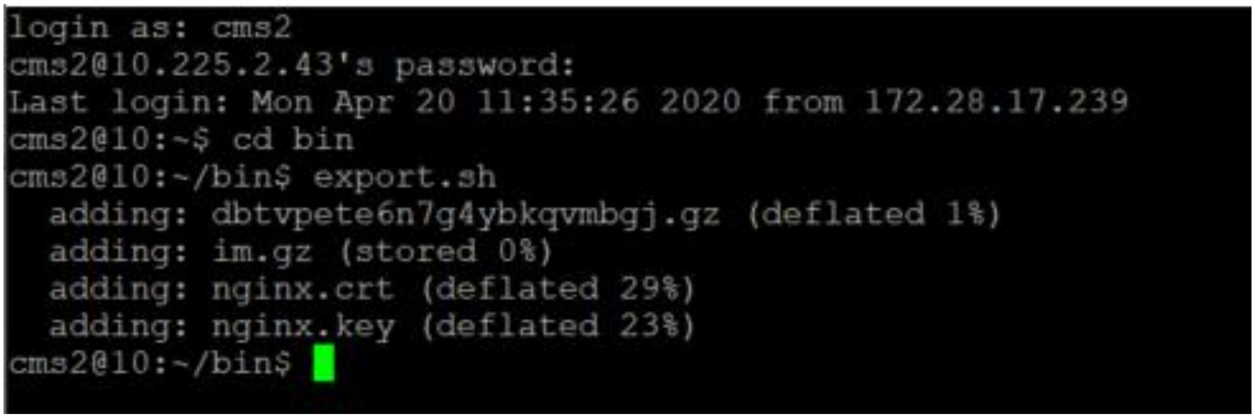
Allow the script to complete at which point the CMS will reboot on its own and come back up to the login prompt. Now we can move to the web UI for the next few steps.

You will be prompted to enter a user account for the CMS. Be sure to use the same credentials you use on the old CMS. Once you’ve created the login user account and can access the CMS, navigate to the About Us page.

Obtaining Your Backup

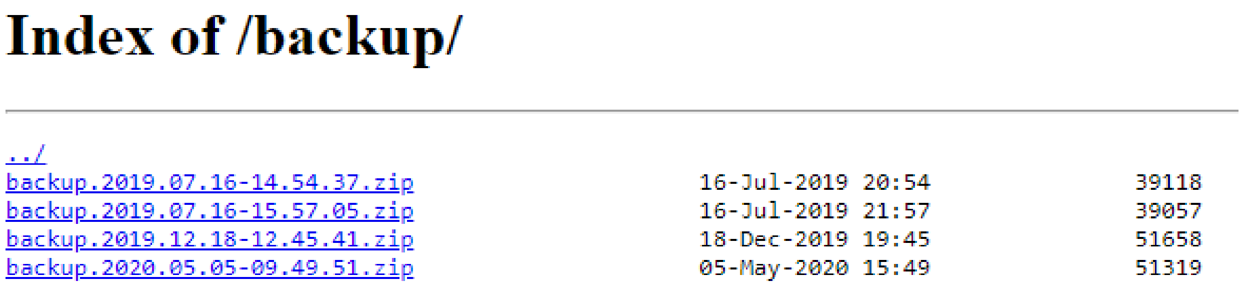
Let’s pause for a moment and grab a backup from the old CMS server. This step is really simple but obviously pretty critical. Keep in mind that at the point you take your backup you should stop making any changes in the old CMS to avoid having configurations that do not carry over to the new CMS.

Log into the command line or console on your old CMS server. Then navigate to the bin folder to run the export.sh script.



```
login as: cms2
cms2@10.225.2.43's password:
Last login: Mon Apr 20 11:35:26 2020 from 172.28.17.239
cms2@10:~$ cd bin
cms2@10:~/bin$ export.sh
  adding: dbtvpete6n7g4ybkqvmbgj.gz (deflated 1%)
  adding: im.gz (stored 0%)
  adding: nginx.crt (deflated 29%)
  adding: nginx.key (deflated 23%)
cms2@10:~/bin$
```

Running the export script will package up your database which will include your account key information, certificate and all of the configuration for your phones. This file then needs to be downloaded from your CMS server. You will want to navigate to http://*CMSIPAddress*/backup and be sure to note that we used HTTP and not HTTPS there. The CMS direct you to the right place if you use the wrong protocol. What you’ll see is a list of all the backup files created.



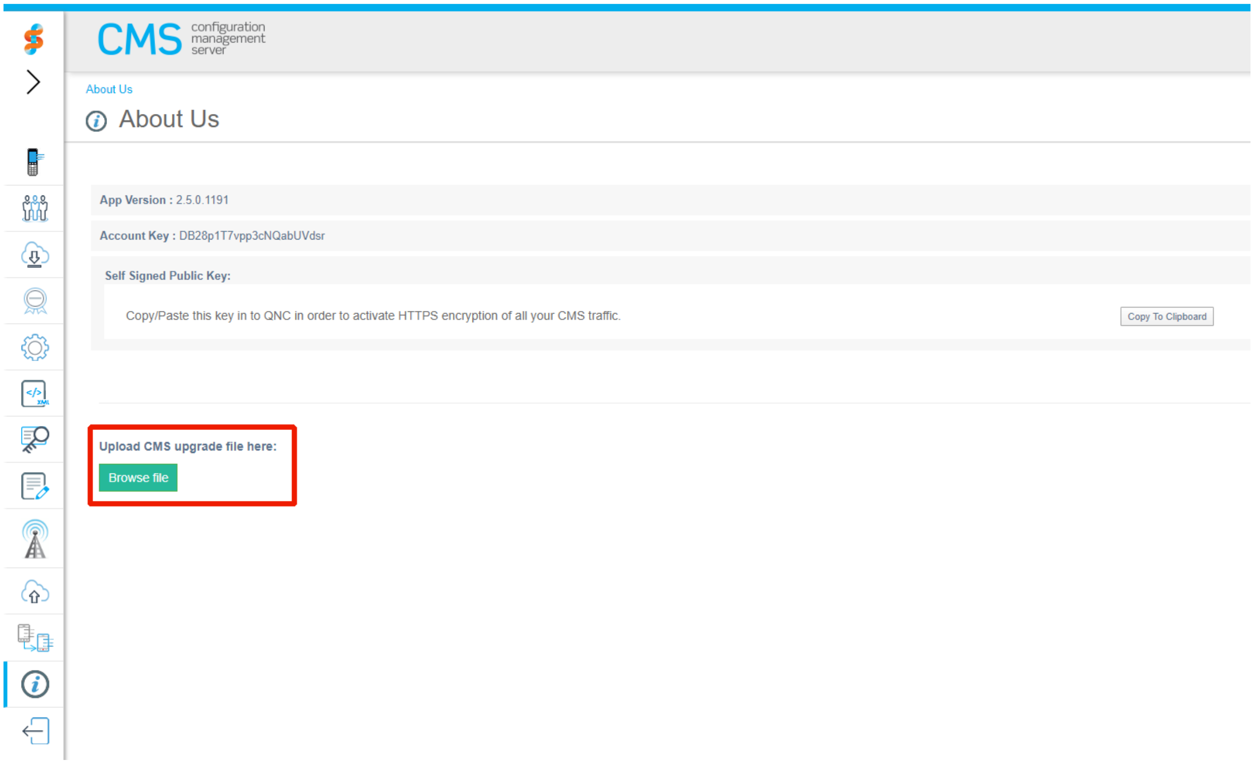
Index of /backup/

```
../
backup.2019.07.16-14.54.37.zip                    16-Jul-2019 20:54              39118
backup.2019.07.16-15.57.05.zip                    16-Jul-2019 21:57              39057
backup.2019.12.18-12.45.41.zip                    18-Dec-2019 19:45              51658
backup.2020.05.05-09.49.51.zip                    05-May-2020 15:49              51319
```

Look at the dates in the middle column and click on the file with the most recent date. This will download it to your PC. Be sure you know where it gets stored, usually the Downloads folder on Windows, as we'll need it for the restore process on the new CMS.

Time to Restore

To restore the configuration on the new CMS we will largely do the reverse steps. Go back to the web UI on your new CMS that should still be sitting on the About Us page. It may have timed out and you'll need to log in again. Click on the "Browse file" button and locate the backup file you just downloaded from your old CMS.



CMS configuration management server
About Us
About Us

App Version : 2.5.0.1191

Account Key : DB28p1T7vpp3cNQabUVdsr

Self Signed Public Key:

Copy/Paste this key in to QNC in order to activate HTTPS encryption of all your CMS traffic.

Copy To Clipboard
Upload CMS upgrade file here:
Browse file

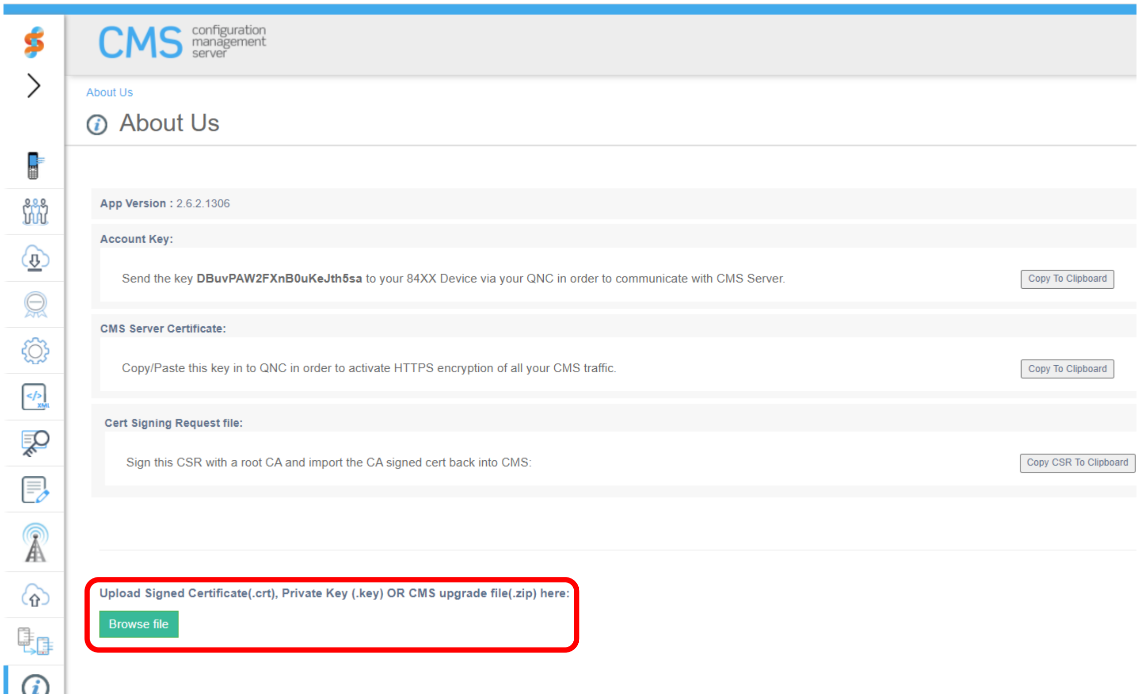
Wait for it to finish uploading to the CMS and when it reports that it is complete you can then close the web UI window and return to the command line or console.

1. From the CMS console navigate to: cd bin
2. Enter the command: import.sh

a. Wait for CMS to finish the import at which point you will then have the database of devices and configurations, certificate and account key of the CMS appliance you generated the backup from.

b. Reboot the CMS Server. From the CMS console, enter the command sudo init 6. This will reboot CMS.

Now you will need to apply the CMS R2.6.2.1306 Update. Go back to the web UI on your new CMS that should still be sitting on the About Us page. It may have timed out and you'll need to log in again. Click on the "Browse file" button and locate the artifacts.zip file.



About Us

App Version : 2.6.2.1306

Account Key:

Send the key **DBuvPAW2FXnB0uKeJth5sa** to your 84XX Device via your QNC in order to communicate with CMS Server.

CMS Server Certificate:

Copy/Paste this key in to QNC in order to activate HTTPS encryption of all your CMS traffic.

Cert Signing Request file:

Sign this CSR with a root CA and import the CA signed cert back into CMS:

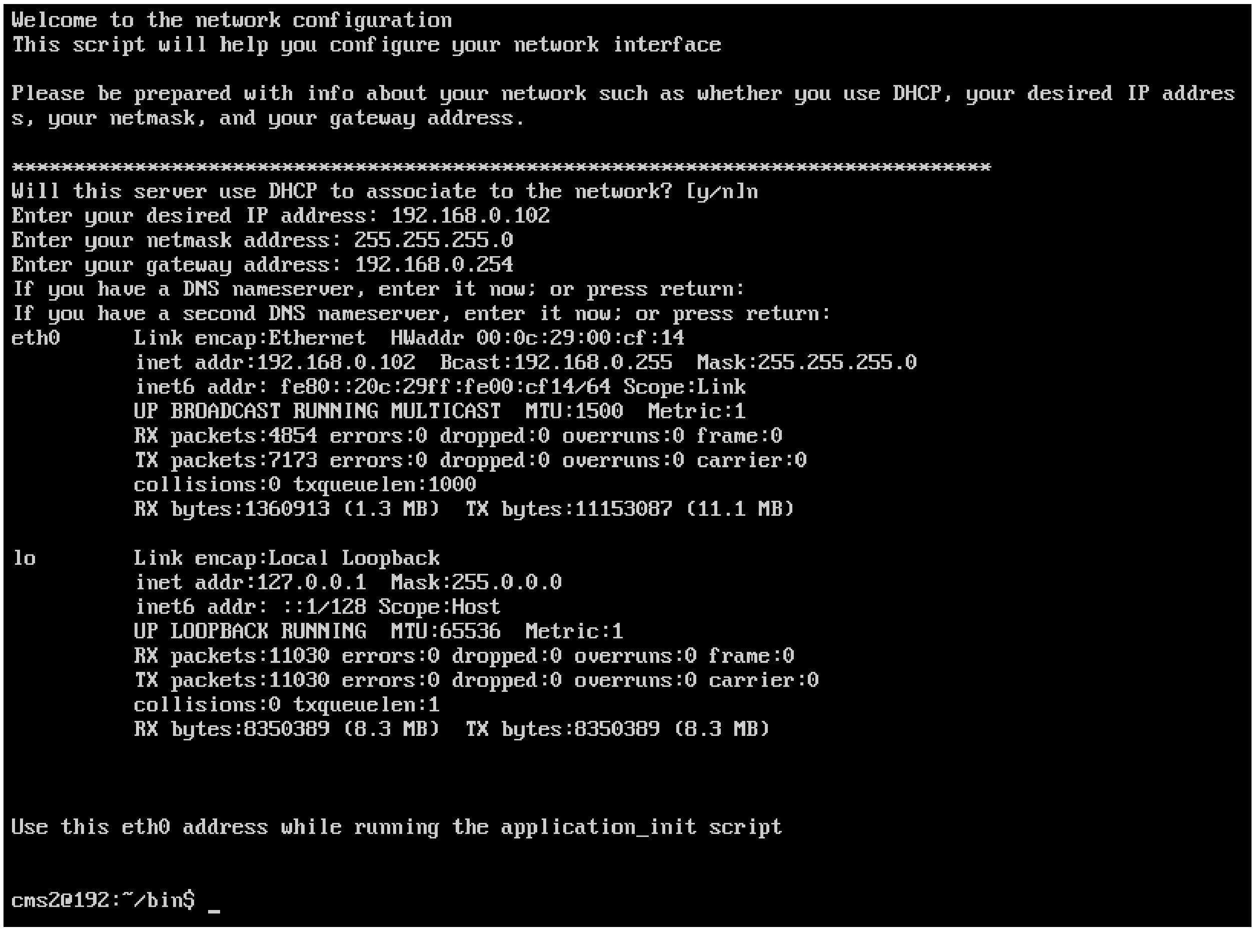
Upload Signed Certificate(.crt), Private Key (.key) OR CMS upgrade file(.zip) here:

3. Wait for it to finish uploading to the CMS and when it reports that it is complete you can then close the web UI window and return to the command line or console for the last steps.
4. From the CMS console navigate to: cd bin
5. Enter the command: upgrade.sh

Cleaning up the Migration

You should now be able to open up your web UI and see your device list is present in the new CMS. However, you are still using the wrong IP Address so your phones aren't going to be able to talk to this CMS yet. So let's wrap this up by getting the IP Address reconfigured. Be sure to shut down your old CMS before you do anything else or you will have IP Address conflicts.

You should still be in the command line or console of the CMS from the last step. Go ahead and run the network initialization script again, sudo python network_init.py and press enter.



```
Welcome to the network configuration
This script will help you configure your network interface

Please be prepared with info about your network such as whether you use DHCP, your desired IP addres
s, your netmask, and your gateway address.

********************************************************************************
Will this server use DHCP to associate to the network? [y/n]n
Enter your desired IP address: 192.168.0.102
Enter your netmask address: 255.255.255.0
Enter your gateway address: 192.168.0.254
If you have a DNS nameserver, enter it now; or press return:
If you have a second DNS nameserver, enter it now; or press return:
eth0      Link encap:Ethernet  HWaddr 00:0c:29:00:cf:14
          inet addr:192.168.0.102  Bcast:192.168.0.255  Mask:255.255.255.0
          inet6 addr: fe80::20c:29ff:fe00:cf14/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:4854 errors:0 dropped:0 overruns:0 frame:0
          TX packets:7173 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:1360913 (1.3 MB)  TX bytes:11153087 (11.1 MB)

lo        Link encap:Local Loopback
          inet addr:127.0.0.1  Mask:255.0.0.0
          inet6 addr: ::1/128 Scope:Host
          UP LOOPBACK RUNNING  MTU:65536  Metric:1
          RX packets:11030 errors:0 dropped:0 overruns:0 frame:0
          TX packets:11030 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1
          RX bytes:8350389 (8.3 MB)  TX bytes:8350389 (8.3 MB)



Use this eth0 address while running the application_init script


cms2@192:~/bin$ _
```

Caution

This time, if you used a hostname for your CMS, make absolutely sure that you enter the DNS server addresses. However, if you aren't using a hostname with your CMS server you can continue to leave the DNS server fields blank.

Walk through the script steps again and when it is complete you should close any browser windows and reopen them to connect to the web UI. Confirm that you're able to connect to the IP Address or hostname of your CMS now that it's using the correct address.

Note

There's one last step for you to complete if you've just migrated to a 2.6.x CMS from a prior version. You likely noticed that when you imported your database that the version number changed to the old version of your previous CMS. To remedy this, upload the artifacts.zip file for the 2.6.1 version to your CMS via the About Us page and then run the upgrade.sh script. This will only update the software version back to the correct version.

If you're satisfied with that status of things, you should go ahead and take a snapshot of the new CMS VM and then decommission the old CMS VM. However, it's entirely up to you to decide whether you want to decommission to the old CMS now or wait for a while. Otherwise, you should be all set to start using your new CMS.

Renewing the CMS Certificate

This section will discuss the procedure for renewing your CMS certificate while maintaining communication with your Spectralink handsets to the maximum level possible. Depending on your current deployment, it is possible that during this process that you might have a period of time where your Spectralink handsets are out of communication with CMS. During this process we will be sure to point out where you may encounter potential losses of communication or if there could be an impact to end-users, which will be extremely limited.

These procedures are only relevant for pre-2.6.2 CMS versions. After CMS release 2.6.2 there will be new methods implemented for handling the renewal of CMS certificates. The processes will be similar to these when 2.6.2 becomes available.

Generating the New Certificate

The first step is to get a new self-signed certificate generated for the CMS so that we can get that pushed to the handsets. To accomplish this step, we've made a new script available that will create the new certificate and also generate a new custom XML file for you that we'll use in a later step to get the certificate onto the handsets. There are two scripts that are related to this process and they can be obtained from the Spectralink Support Site.

The first script, renewCertAndGenCFG.py, will be used to generate the new certificate and custom XML file. The second script, replaceDefaultCertAndRestart.py, will handle installing the new certificate into the CMS and restarting the relevant services. Once you obtain the scripts you will want to upload them to your CMS and place them in the bin folder, /home/cms2/bin/. If you need assistance with loading these scripts to your CMS you can contact Spectralink Technical Support, but you will also want to ensure you've followed the technical bulletin "CS-17-09 Enabling SSH on CMS 2.x Server"[1] and then you can use a program such as PSFTP or FileZilla to upload the files. Just remember to enable SSH first or you will be unable to connect using the cms2/cms2 user on port 22. Once you've placed the files into the bin folder you will need to change the permissions on the files using the following commands:

```
sudo chmod 755 /home/cms2/bin/renewCertAndGenCFG.py
sudo chmod 755 /home/cms2/bin/replaceDefaultCertAndRestart.py
```

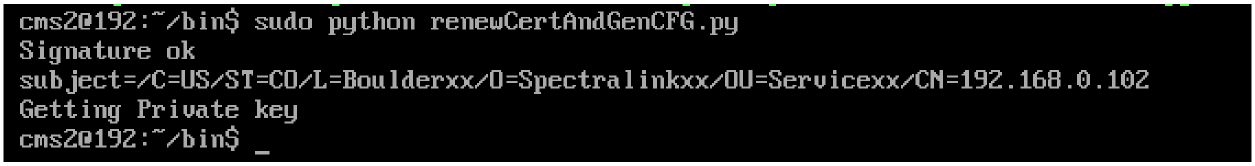
[1] https://support.spectralink.com/sites/default/files/resource_files/CS-17-09%20Enabling%20SSH%20on%20CMS%202.x%20Server.pdf

Now we can move on to actually generating the certificate and getting it onto the handsets. You'll need to be logged into your CMS console or via SSH, which if you've enabled it as described above may be the easiest way to proceed. Navigate to the bin folder and we'll run the first script.

```
cd bin
```

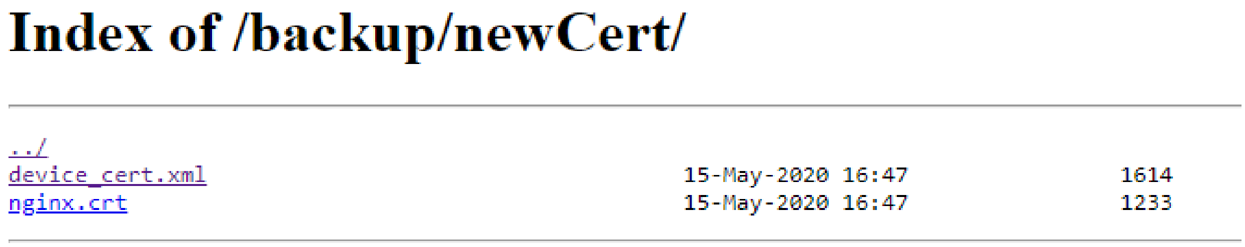
```
sudo python renewCertAndGenCFG.py
```

This will create the new certificate and the custom XML file. You should see output the looks something similar to the following:



```
cms2@192:~/bin$ sudo python renewCertAndGenCFG.py
Signature ok
subject=/C=US/ST=CO/L=Boulderxx/O=Spectralinkxx/OU=Servicexx/CN=192.168.0.102
Getting Private key
cms2@192:~/bin$ _
```

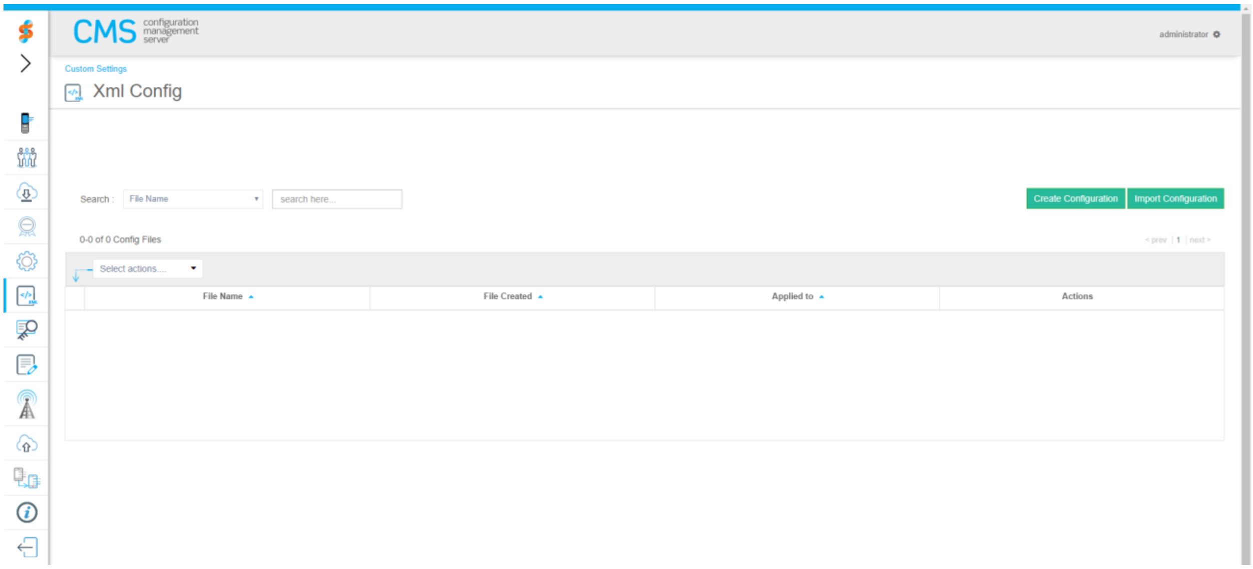
The details that display will be relevant to your specific setup based on what you entered when initial deploying your CMS and running the application_init.py script. You will now be able to navigate to the web server and download the new certificate and custom XML file that were just created. Open a browser and go to http://*cmsIPaddress*/backup/newCert where you will see the following:



Index of /backup/newCert/

```
../
device_cert.xml                                  15-May-2020 16:47                1614
nginx.crt                                        15-May-2020 16:47                1233
```

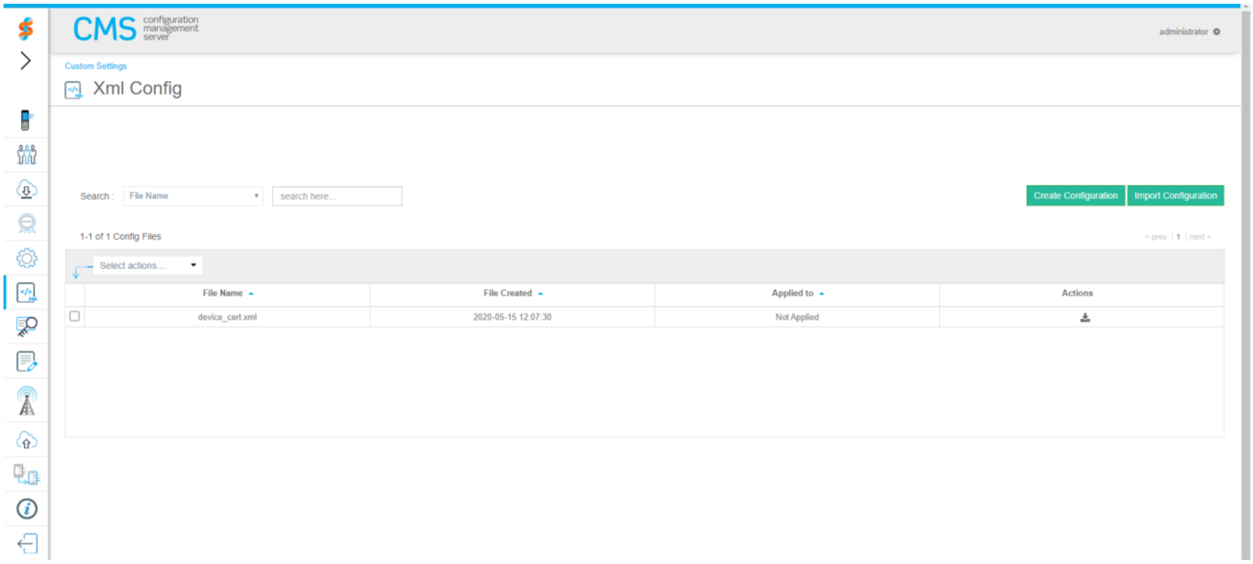
You'll want to right click on the device_cert.xml and choose "Save Link As" and put the file in a location you can remember and easily reach. Now you will need to open up the CMS web UI, log in and navigate to the Custom XML page.



0-0 of 0 Config Files

File Name	File Created	Applied to	Actions

Here you will want to press the Import Configuration button on the right side of the screen and then click the Browse XML Files button and navigate to where you saved the device_cert.xml file and select it. This will then import the file into the CMS immediately. You will now see the file listed in the Custom XML file view in the CMS as shown here.



1-1 of 1 Config Files

	File Name	File Created	Applied to	Actions
□	device_cert.xml	2020-05-15 12:07:30	Not Applied	

Now you need to associate this file with the handsets. You can choose to associate it at the Enterprise level, Group Level or even at a Device level for individual phones.

Caution

Once you associate the Custom XML file to phones, they will pick up the change on their next heartbeat. This configuration change will cause the phones to reboot as soon as they pick it up, so you may want to consider scheduling the timing of when you associate the Custom XML file to avoid impacting users.

The next step is the most important, and may well take the most amount of time. It's critical that the handsets pick up the new certificate but in order to do so they must be powered on so that they can heartbeat with the CMS. We don't want to put the new certificate into operation on the CMS until we're confident that the handsets have the new certificate. Just be conscious of the fact that you will be unable to manage the handsets from CMS during this transition period as we assume that the current CMS certificate will be replaced with the new certificate. The handsets will not be otherwise impacted which will allow users to continue using the handsets during this time.

Tip

The 84-Series handset has two platform slots that can be used for the CMS certificate. The script described in this process places the new CMS certificate into the second slot in the handset. If the current CMS certificate is in the first slot then it is possible for the handsets to remain in communication with CMS throughout this procedure.

It will be somewhat difficult to know for certain whether all the phones have picked up the new certificate beyond the handsets no longer being able to heartbeat with CMS. But this will look very similar to a handset that's just powered off. So it will be important to inform users to turn on handsets so they can pick up the change regardless of whether they are used regularly. Make a plan and follow up with users regularly to ensure handsets are getting powered on. Once you feel confident, or at least comfortable, that you've achieved success with updating your handsets you can move onto the last step.

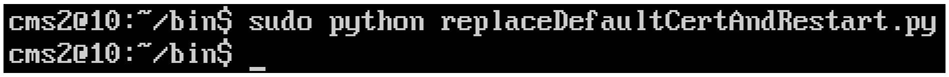
Applying the Certificate to CMS

For the last step all you have to do is apply the certificate to the CMS server. The best part here is that everything you need has already been loaded into CMS and created with the first script. Running the script will return the handsets to full communication with CMS by adding the certificate to the services that require it and then restarting those services. From your CMS console or via SSH, navigate to the bin folder again and run the replaceDefaultCertAndRestart.py script.

```
cd bin
```

```
sudo python replaceDefaultCertAndRestart.py
```

Your output will look like the following:



```
cms2010:~/bin$ sudo python replaceDefaultCertAndRestart.py
cms2010:~/bin$ _
```

If the you are returned to the prompt with no messages, you will need to restart CMS. Enter sudo init 6.

At this point, you should see your handsets immediately resume communicating with the CMS without further interaction. If you look at the Device List in the CMS Web UI you will start to see handsets display as "Active" once more.

Clean up and Final Considerations

If you have new handsets or discover handsets after you've completed this process that did not pick up the new certificate then you will likely need to use a QNC tool to restore them to operation. For handsets that were missed, you may want to consider just factory defaulting them and using QNC. Once they reconnect with CMS they will pick up all their configuration once more. Just remember to update the certificate being loaded via your QNC tool. You can obtain the certificate contents at any time again using the "Copy to Clipboard" button the About Us page in your CMS. Or by going back to the http://*cmsIPaddress*/backup/newCert where you can find it labeled as nginx.crt.

If you encounter problems or have questions, then contact Spectralink Technical Support or your Spectralink Authorized Reseller for assistance.

Document Status Sheet

Document Control Number: CS-20-09

Document Title: Migrating CMS Data Between CMS Versions

Revision History: I01 – Released *May 5, 2020*
I02 – Released *May 15, 2020*
I03 – Released

Date: *May 5, 2020*

Status: ☐Draft ☒Issued ☐Closed

Distribution Status: ☐Author Only ☐Internal ☐Partner ☒Public

Copyright Notice

Notice

Spectralink Corporation has prepared this document for use by Spectralink personnel and customers. The drawings and specifications contained herein are the property of Spectralink and shall be neither reproduced in whole or in part without the prior written approval of Spectralink, nor be implied to grant any license to make, use, or sell equipment manufactured in accordance herewith.

Spectralink reserves the right to make changes in specifications and other information contained in this document without prior notice, and the reader should in all cases consult Spectralink to determine whether any such changes have been made.

NO REPRESENTATION OR OTHER AFFIRMATION OF FACT CONTAINED IN THIS DOCUMENT INCLUDING BUT NOT LIMITED TO STATEMENTS REGARDING CAPACITY, RESPONSE-TIME PERFORMANCE, SUITABILITY FOR USE, OR PERFORMANCE OF PRODUCTS DESCRIBED HEREIN SHALL BE DEEMED TO BE A WARRANTY BY SPECTRALINK FOR ANY PURPOSE, OR GIVE RISE TO ANY LIABILITY OF SPECTRALINK WHATSOEVER.

Warranty

The *Product Warranty and Software License and Warranty* and other support documents are available at http://support.spectralink.com.

Contact Information

US Location
+1 800-775-5330

Spectralink Corporation
2560 55th Street
Boulder, CO 80301
USA

info@spectralink.com

Denmark Location
+45 7560 2850

Spectralink Europe ApS
Bygholm Soepark 21 E Stuen
8700 Horsens
Denmark

infoemea@spectralink.com

UK Location
+44 (0) 20 3284 1536

Spectralink Europe UK
329 Bracknell, Doncastle Road
Bracknell, Berkshire, RG12 8PE
United Kingdom

infoemea@spectralink.com