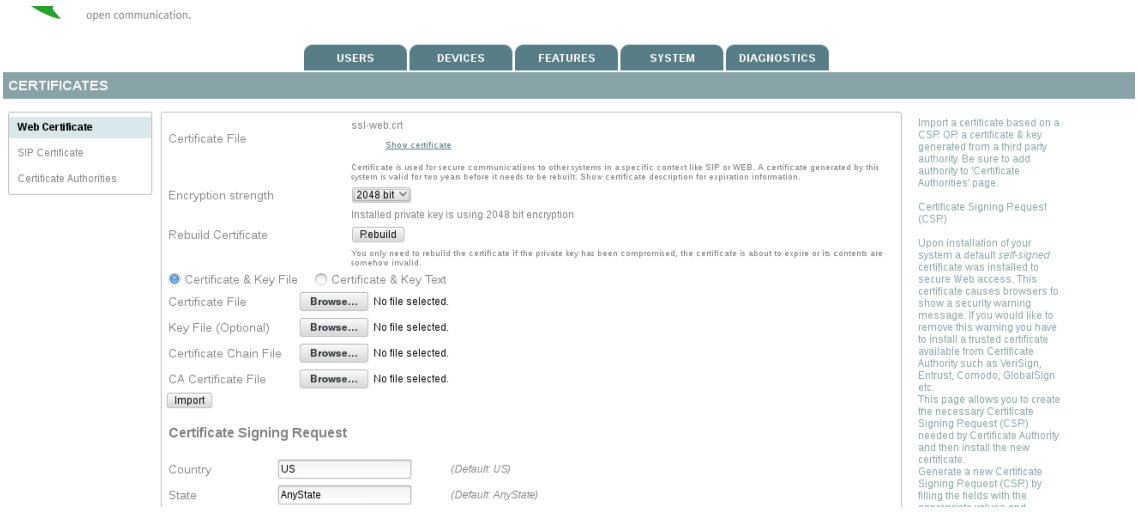
Certificates

System Certificates

Sipxcom works with two types of certificates:

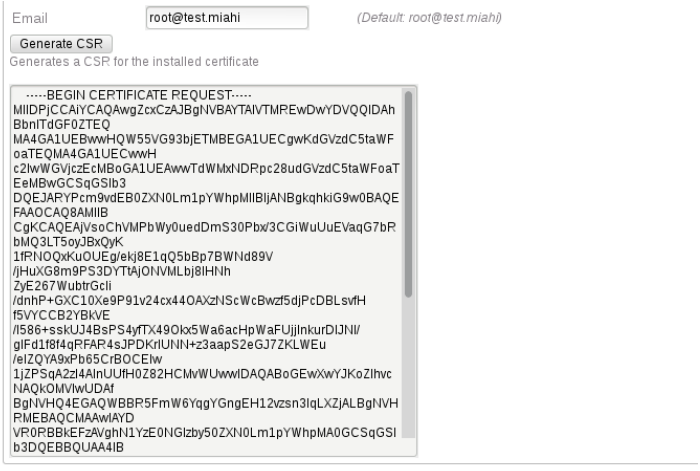
1.Web certificate needed for system GUI and Unite Web -User portal

2. Sip certificate needed for TLS communication between Proxy and Phones



1. To upload a (trusted) web certificate you need to generate a Certificate Signing Request that will be sent to a Certificate Authority. You can do this either by using internal CSR generation mechanism from Sipxcom or by using a different machine. If you are using a different machine you need to make sure that you will have your "private key" stored in a safe place and that you are using complete FQDN or you can ask for a wild-card certificate that should match your domain (go to step C.).

A.generate CSR:



B Sent your csr file to a trusted authority to sign it.

C. If you receive any intermediary certificate from your CA you need to uploaded it here, if not you can skip this step:

D. If you used a different machine to issue your csr then you MUST import your private key (skip if you used sipxcom internal mechanism):

E. Import your signed certificate:

After you reconnect to system GUI you should see in your browser information about your newly signed(or not signed) certificate.

You can check the time of creation and content of ssl-web.crt and ssl-web.key from this path /etc/http/conf.d/ssl/ to make sure it matches your newly uploaded certificate and key (if needed).

Troubleshooting:

If something goes wrong don't panic you can always access your system GUI from HTTP by using the below method:

a.service iptables stop

b.connect to http://Your_IP:12000/sipxconfig/app

c. Re-try adding the certificate or simply "Rebuild" sipxcom self signed certificate

Still having problems? then take a look at /var/log/sipxpbx/sipxconfig.log and /var/log/httpd also a service httpd restart may be needed.

Usually a key mismatch is the common problem, you can see this is in /var/log/httpd/ssl_error.log - depending on your certificate provider you will be able to re-key the certificate with the current ssl-web.key or you will need to submit a new CSR.

Very important: keep your initial ssl-web.key safe (copy to another machine). This is the key used when CSR was issued and if you hit Rebuild it will be overwritten.

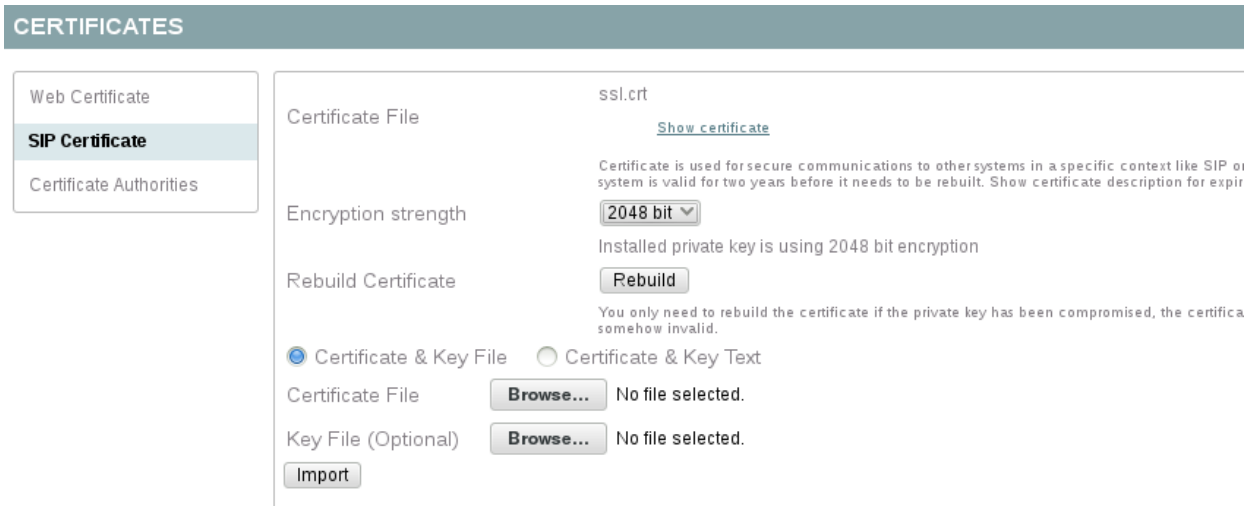
2. Sip Certificate and how to upload CA to a polycom phone.

As mentioned earlier SIP Certificate is needed if you are planning to use TLS signaling between Phones and proxy.

Self signed sip certificate issued by sipxcom can be used simply by setting transport to TLS from Devices–> Line ---> Registration page.

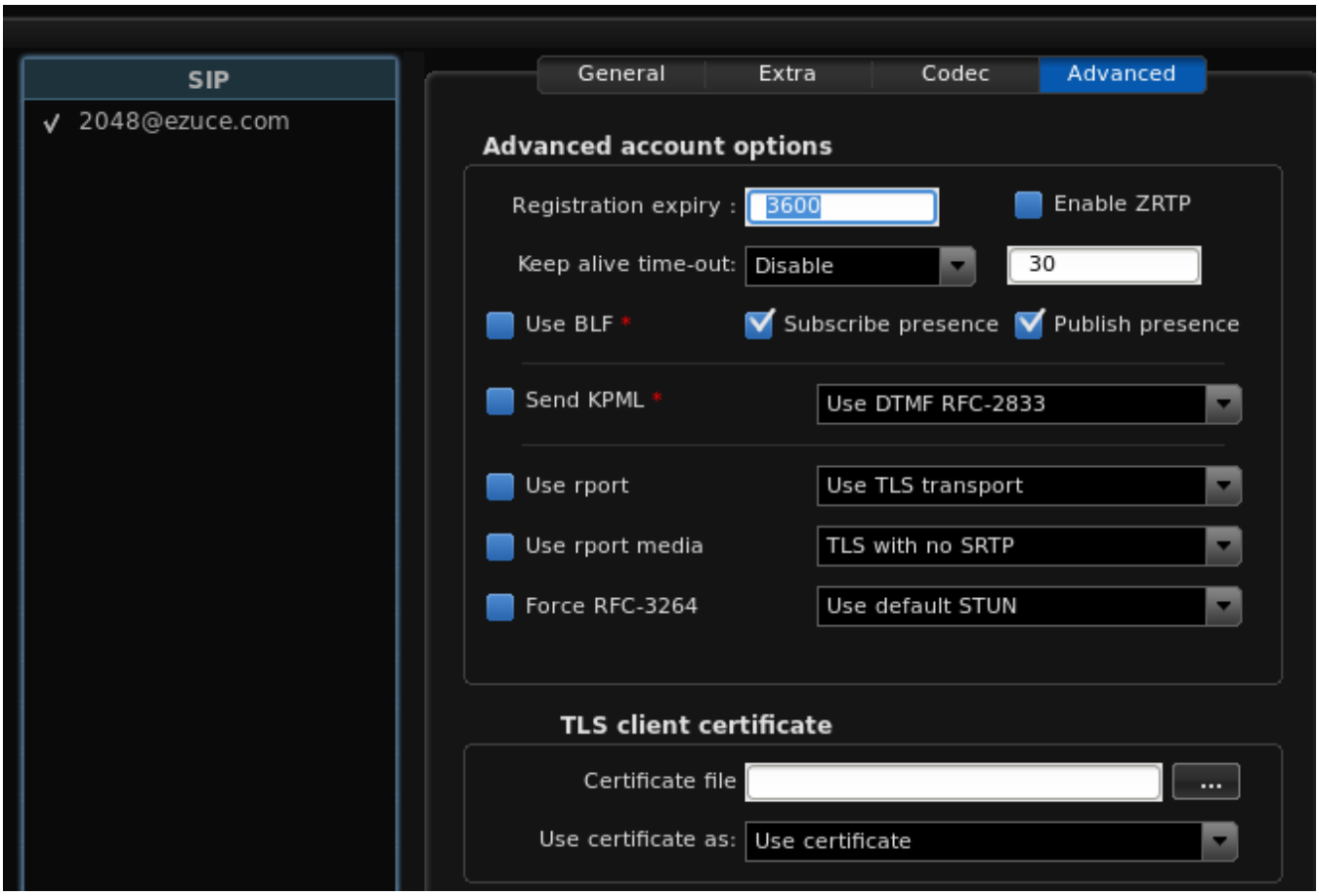
Since polycom phones are auto-provisioned they will acquire Certifcate Autorithy corresponding to sipxcom file from the beginning.

If you are planning to use a signed sip certificate or a self signed certificate (as always don't forget to save your private key) you will need to upload the both under System–> Certificate --> Sip Certificate page



In order for clients (in this case phones) to trust a certificate they need to have a valid CA file for the issues of the sip certificate they will receive. Since the new certificate is not created by sipxcom you will need to upload CA file to phones.

if you are using a softphone like zoiper you can do this by simply reaching to the menu that will let you upload CA.

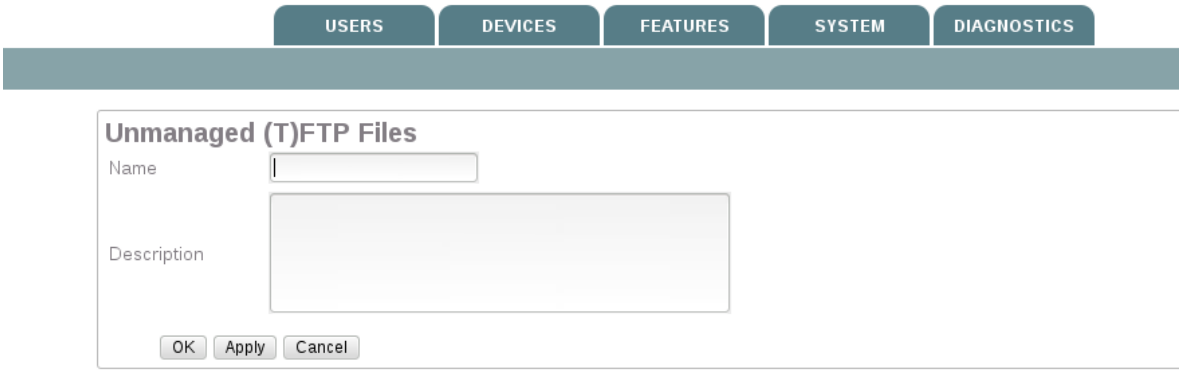


If you are using a provisioned phone like polycom with firmware => 4.x you can create a custom config file with the following content

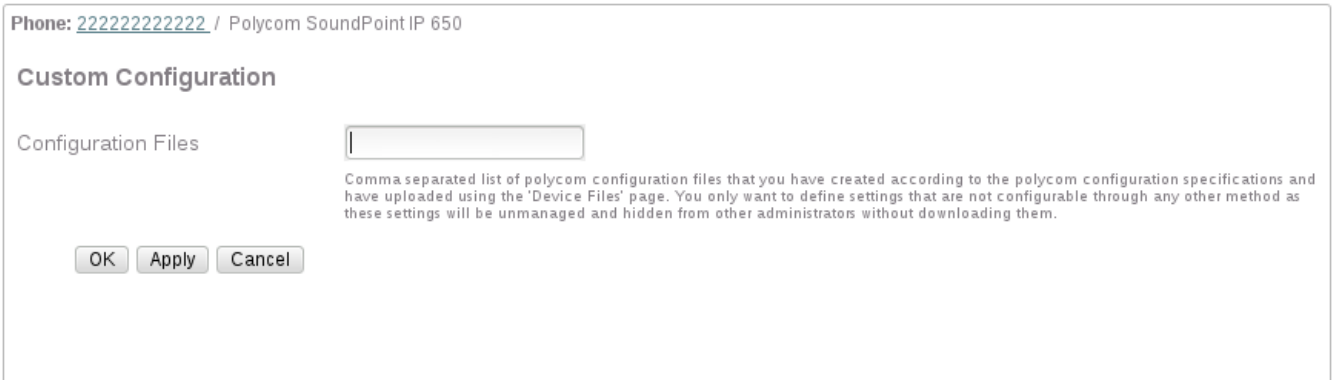
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<polycomConfig xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:noNamespaceSchemaLocation="polycomConfig.xsd">
<sec
sec.TLS.customCaCert.2="—BEGIN CERTIFICATE---

YOUR CERTIFICATE AUTHORITY FILE
-----END CERTIFICATE-----"
sec.TLS.profile.2.deviceCert="Application2"
sec.TLS.profileSelection.SIP="ApplicationProfile2"
sec.TLS.profile.2.caCert.application2="0"
sec.TLS.profile.2.caCert.application3="0"
sec.TLS.profile.2.caCert.application4="0"
sec.TLS.profile.2.caCert.application5="0"
sec.TLS.profile.2.caCert.application6="0"
sec.TLS.profile.2.caCert.defaultList="0"
sec.TLS.profile.2.caCert.platform1="0"
sec.TLS.profile.2.caCert.platform2="0"
/>
</polycomConfig>

Then go to Devices–> Devices Files–> Unmanaged TFTP and upload this custom file



Last go to Devices --> MAC–> Custom Configuration–> Select File name(exactly as the name of the cfg file uploaded at previous step) ---> Send profiles



Chrome and Self Signed Certificates

Chrome likes to complain about self signed certificates. To resolve this either, upload a valid web certificate or:

On a Windows PC

On a Mac

1. In the address bar, click the red lock with the X. This will bring up an information screen. In the 'Connection' tab, click the link that says "Certificate Information".
2. Click and drag the image of the certificate to your desktop.
3. Double-click on the certificate on your desktop will bring up the Keychain Access utility. Enter your password to enter the application.
4. Click on the lock icon in the program to unlock it.
5. Find the certificate in the System keychain. Double click on it.
6. Expand 'Trust'
7. Find 'When using this certificate' entry select "Always Trust" from the drop down selector.

Close Keychain Access and restart Chrome, and the self-signed certificate should be recognized now by the browser. (it still won't be green in the address bar, but the browser won't keep prompting you)

On a Fedora Workstation

Incomplete

This page needs your help to finish.