PacketFence

Microsoft PKI (MSPKI) Quick Installation Guide

for PacketFence version 7.4.0

Microsoft PKI (MSPKI) Quick Installation Guide

by Inverse Inc.

Version 7.4.0 - Jan 2018

Table of Contents

Chapter 1

About this Guide

This guide has been created to give a quick start to configure the Microsoft PKI with PacketFence 5.4+. This guide does not include advanced troubleshooting of EAP-TLS connections. Refer to the relevant documentation of EAP-TLS, RADIUS and OpenSSL for advanced features.

Assumptions

- You have at least one server with PacketFence 5.4 or later.
- The server already has a properly configured switch or access point with 802.1X support.
- The PacketFence RADIUS server is working in your environment.
- You have a Microsoft Windows 2008 R2 Enterprise server installed.
- The PacketFence management IP will be 192.168.1.5.
- The RADIUS shared secret is "useStrongerSecret".
- In this guide you will see a lot of use of <ServerDNSName>, most of the MSPKI services requires in their configuration to use the FQDN of the server and not his IP.

Installation

Step 1: Install Active Directory Certificate Service (ADCS)

Note

This section will cover the configuration for Active Directory Certificate Services (ADCS) on Microsoft Windows 2008 R2 Enterprise. The installation of ADCS is not covered by this guide, refer to the Microsoft documentation about it for more information (http://social.technet.microsoft.com/wiki/contents/articles/9063.network-device-enrollment-service-ndes-in-active-directory-certificate-services-ad-cs.aspx).

For the integration with PacketFence, the following subroles need to be installed in ADCS:

- Certification Authority Web Enrollment
- Network Device Enrollment Service
- Online Responder

Before you start the configuration, a hotfix is necessary due to a Microsoft issue. After restarting the ADCS service, the server cannot enroll new certificates and display the following error message: "The RPC Server is unavailable". The hotfix is available here: https://support.microsoft.com/en-us/kb/2633200

Communication between the MSPKI and PacketFence will be using port 80.

Configuring Network Device Enrollment Service (NDES)

For the deployment of ADCS you will need to configure Network Device Enrollment Service (NDES). This subrole will allow us to exchange certificates with the MSPKI server via Simple Certificate Exchange Protocol (SCEP).

Every configuration change has to be done by an account with administrative privileges.

Challenge Password

Microsoft SCEP (MSCEP) includes by default a challenge password, which is unique and dynamically generated for each device which wants to enroll. In a BYOD deployment, this can be a barrier as a user cannot register a device by himself without the intervention of an administrator. Since we use NDES with PacketFence, our security to obtain a certificate would be the credentials necessary to access the enrollment system.

To disable the challenge password you need to modify the following key in the Windows registry.

Click `Start` and enter `regedit`.

Navigate to `Computer->HKEY_LOCAL_MACHINE->SOFTWARE->Microsoft->Cryptography->MSCEP->EnforcePassword`.

Change the value of `EnforcePassword` to `0` (default is `1`).

Extend URL length for the request

Best practices recommends to extend the URL length to avoid issue with longer request.

To do so, enter the following command in the CLI on the NDES server:

```
%systemroot%\system32\inetsrv\appcmd.exe set config /section:system.webServer/
security/requestFiltering /requestLimits.maxQueryString:”16384” /commit:apphost
```

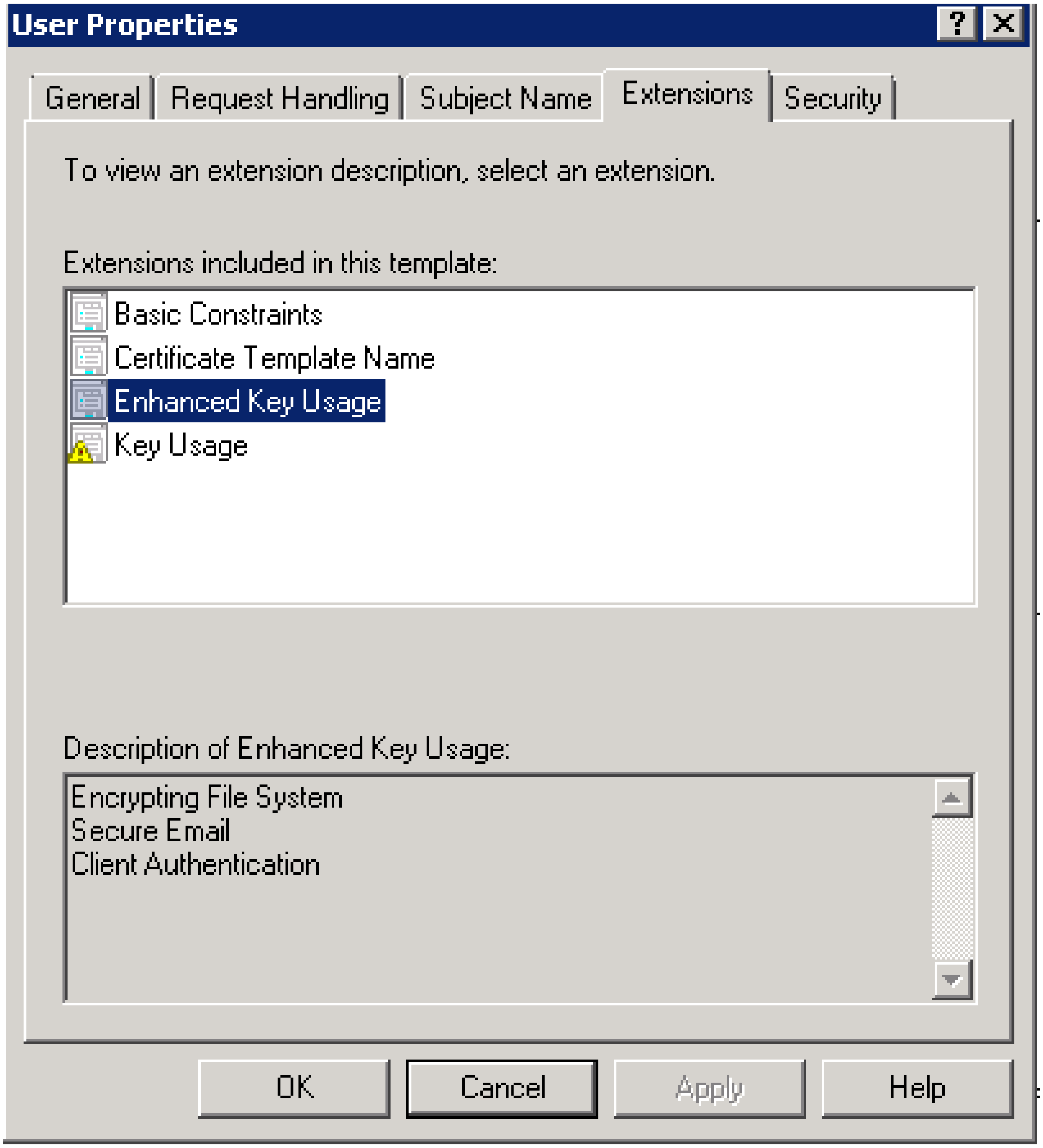
Certificate template

Caution

Remember that the validity of your CA can impact your whole certificate architecture.

The goal is to deliver certificates for `user Authentication`, this means you will need to setup a specific template.

First, the certificate template needs at least the following `Enhanced Key Usage` and `Key Usage`:


User Properties

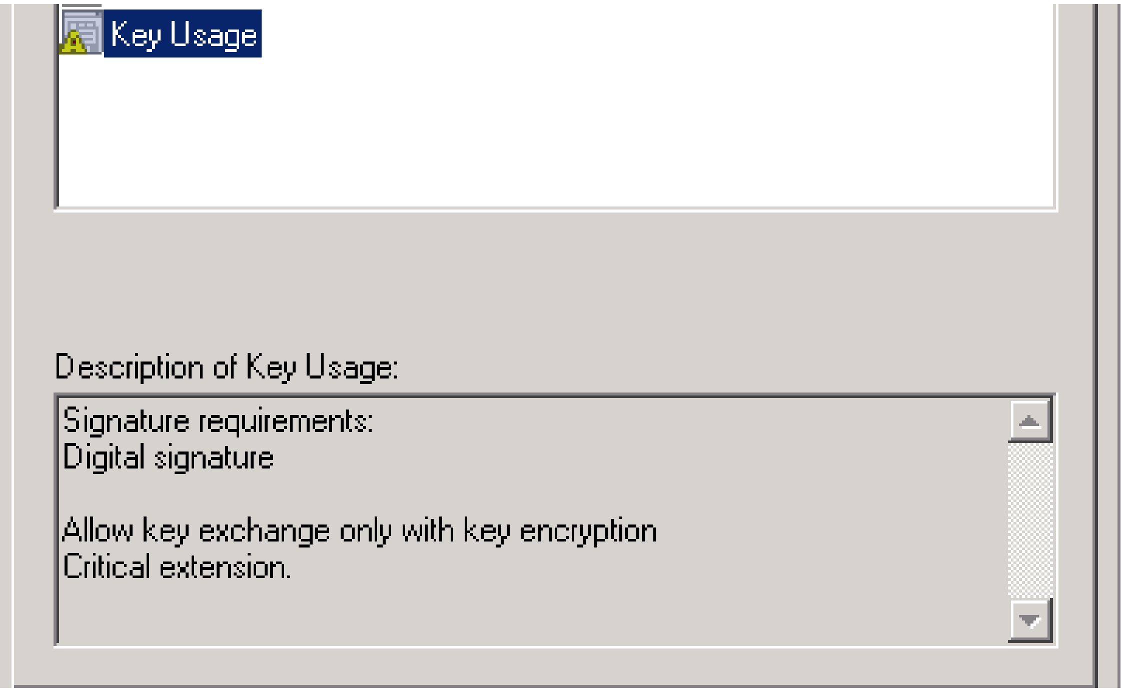
To view an extension description, select an extension.

Extensions included in this template:

- Basic Constraints
- Certificate Template Name
- Enhanced Key Usage
- Key Usage

Description of Enhanced Key Usage:

Encrypting File System
Secure Email
Client Authentication



Description of Key Usage:

Signature requirements:
Digital signature

Allow key exchange only with key encryption
Critical extension.

The next step is to duplicate a template where those `Key Usage`, and `Enhanced Key Usage` are already configured. We advise to duplicate the template `User` and change the necessary settings.

To duplicate the template, you need to navigate through `Server Manager->Roles->Active Directory Certificates Services->Certificate templates`. Now right click the template `User` and select `Duplicate this template`.

Once duplicated, right click your new template, go to `Properties`. Navigate to the tab `Subject Name`. Make sure to select `Supplied in the request` over `Built from information in Active Directory`, otherwise the requested CN will be overwritten by NDES.

To allow NDES to use this template you need to navigate to `Server Manager->Roles->Active Directory Certificates Services`, expand `<ServerDNSName>`, right click `Certificate template` and choose `New template to issue`, in the list select your newly created template.

Now that you choose the template to deliver you need to configure it in the registry.

To access the registry editor, press `Start` and type `regedit`.

While in the registry navigate to `Computer->HKEY_LOCAL_MACHINE->SOFTWARE->Microsoft->Cryptography->MSCEP`.

You should have a list of three keys entries:

- EncryptionTemplate,
- GeneralPurposeTemplate,
- SignatureTemplate.

The default value should be `IPSECIntermediateOffline`. Replace each value with your newly created template name.

At this point, you need to reboot the NDES server to apply changes to the registry.

IIS configuration

The use of SCEP with PacketFence also require a change in the IIS configuration.

Navigate to `Server Manager->Web(IIS)`, expand `Default web site` then select `CertSrv->mscep`. Select `Authentication`, and double click `Anonymous Authentication`. Make sure that `Application pool identity` is selected.

Online Certificate Status Protocol (OCSP)

For the configuration of OCSP, the following changes are necessary.

First we need to allow the use of the template `OCSPResponseSigning` by the server, to do so navigate to `Server Manager->Roles->Active Directory Certificates Services`, expand `<ServerDNSName>`, right click `Certificate template` and choose `New template to issue`, in the list select `OCSPResponseSigning`.

After the installation of OCSP we need to create a Revocation Configuration.

To create the Revocation Configuration navigate to `Server Manager->Roles->Active Directory Certificate Services` and expand `OnlineResponder: <ServerDNSName>`. Right click Revocation Configuration, select `Add Revocation Configuration`, click `Next`, choose a name for your configuration and click `Next`.

Choose `Select a certificate for an existing enterprise CA`, click `Next`. Click `Browse` and find your enterprise CA in the list, select it, click `OK` and then `Next`. Choose `Automatically select a signing certificate`, make sure `Auto-Enroll for an OCSP signing certificate` is selected, then choose the default template of OCSP which is `OCSPResponseSigning` in the dropdown list next to `Certificate Template:`. You need to add providers only if you wish to use a CRL in addition to OCSP.

Once created, right click the revocation configuration and select `Edit properties`, go to the `Signing` tab, then select `Enable NONCE extension support` then click `OK`.

Make sure that your OCSP server appears in the CA settings. Right click your CA, choose `Properties`. Navigate to the tab `Extension`, in the dropdown list `Select extension` choose `Authority Information Access (AIA)`. Make sure that you have the following in the list of locations: `http://<ServerDNSName>/OCSP`.

If you do not have it, add it via the button `Add...`. In this menu type the `http://` then insert `<ServerDNSName>` and type `/OCSP`, validate by clicking `OK`. Also verify that `Include in the online certificate status protocol(OCSP) extension` is selected.

By default OCSP has a two days delay to refresh it’s CRL information. Which means if you revoke a certificate on MSPKI, it will take two days before PacketFence detects the certificate is revoked. If this delay is too long for your needs, you can change it on the NDES server. To do so, navigate to `Server Manager->Roles->Active Directory Certificate Service` and right click `Enterprise PKI`, in the menu select `Options...`. The delay can be changed by modifying the value of `Set CRL status to Expiring when expiring in:` to your convenience.

RADIUS Certificate generation

Using the Microsoft PKI involves that all your certificates will be delivered by the root CA of the MSPKI.

As for RADIUS authentication you will need to generate a certificate for PacketFence.

To generate the RADIUS certificate, the template `WebServer` will be used.

The next step is to create the request (CSR), a private key from the PacketFence server and submit the CSR to the NDES server. Connect to PacketFence via SSH and type the following in the CLI to generate the CSR and sign it with the private key:

```
openssl req -new -newkey rsa:2048 -nodes -keyout server.key -out server.csr
```

You will be prompted for some information, here is an example of a valid configuration.

- CN=packetfence.local
- C=CA
- ST=QC
- Locality=Montreal
- Organization=Inverse
- Organization Unit=IT

No fields are mandatory other than the CN.

Once you have your CSR you will submit it to the NDES server.

To submit the request you need to copy the content of the request (CSR) on the MSPKI enrollment website. The URL to input the request will be: `http://<ServerDNSName>/CertSrv/`.

When reaching the website, click `Request a certificate`, select `advanced certificate request`. Paste the content of your CSR file and select the template `Web Server`. Click `Submit`. On this page select `Base 64 encoded` and click `Download certificate`.

This will give you the certificate (public key) for PacketFence.

Now download the CA file by reaching the following URL in your browser: `http://<ServerDNSName>/CertSrv/`.

Click `Download a CA certificate, certificate chain or CRL`, select your CA certificate in the list, select `Base 64` as the encoding method and finally click `Download CA certificate`.

Copy those files to PacketFence.

Step 2: Configuring PacketFence

Certificate storage on PacketFence

It is recommended to create a separate directory to separate EAP-TLS certificates from server certificates:

```
# mkdir /usr/local/pf/conf/ssl/tls_certs/
```

RADIUS EAP-TLS authentication requires three files, the CA certificate, the server certificate and the server private key.

Copy those files in your newly created folder:

- Private Key of the RADIUS server (obtained while generating the CSR)
- Certificate for RADIUS (obtained from the submitted CSR)
- CA Certificate (downloaded from the NDES website)

Ensure that the files are readable by the user `pf`:

```
# chown pf:pf /usr/local/pf/conf/ssl/tls_certs/*
```

RADIUS EAP-TLS and MSPKI

In order to use the certificates generated by the MSPKI, edit the radius EAP configuration file.

Edit `/usr/local/pf/conf/radiusd/eap.conf` and replace the following lines with references to your new certificates in the `tls` configuration block:

```
private_key_file = [% install_dir %]/conf/ssl/server.key
certificate_file = [% install_dir %]/conf/ssl/server.pem
```

E.g.

```
private_key_file = [% install_dir %]/conf/ssl/tls_certs/server.key
certificate_file = [% install_dir %]/conf/ssl/tls_certs/server.pem
ca_file = [% install_dir %]/conf/ssl/tls_certs/MyCA.pem
```

Certificate revocation checks have to be configured in the `OCSP` sub-block of `tls`.

For example:

```
ocsp {
    enable = yes
    override_cert_url = yes
    url = "http://<MSPKI ServerDNSName or IP>/ocsp"
}
```

Restart radiusd to regenerate the new configuration files and enable EAP-TLS using your CA signed certificates:

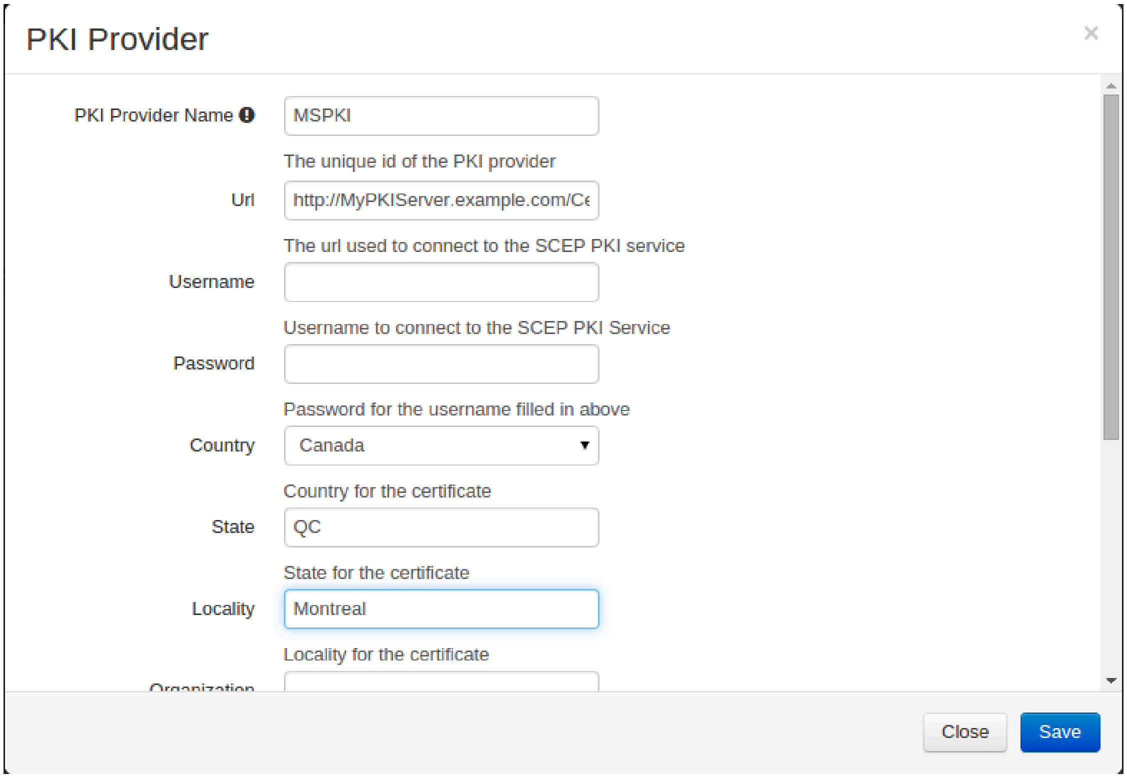
```
# /usr/local/pf/bin/pfcmd service radiusd restart
```

PacketFence PKI provider configuration

Using the PKI requires configuring the PKI providers section in the PacketFence GUI under `Configuration->Users->PKI Providers`. The provider configuration defines how PacketFence connects to the MSPKI and what information will be sent.

Add a new PKI provider and select SCEP.

Fill out the form for a PKI provider according to your Certificate of Authority configuration.

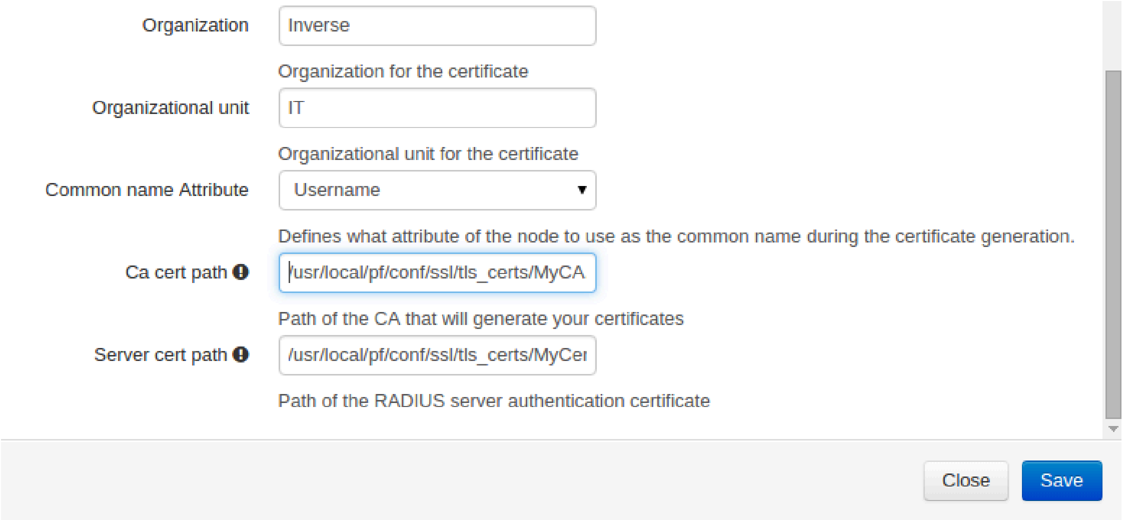


PKI Provider

PKI Provider Name	MSPKI The unique id of the PKI provider
Url	http://MyPKIServer.example.com/Ce The url used to connect to the SCEP PKI service
Username	 Username to connect to the SCEP PKI Service
Password	 Password for the username filled in above
Country	Canada Country for the certificate
State	QC State for the certificate
Locality	Montreal Locality for the certificate
Organization	

For the URL it will be `http://<ServerDNSName>/CertSrv/mscep/`.

You do not need any Username/Password combination for this configuration.



Organization: Inverse

Organization for the certificate

Organizational unit: IT

Organizational unit for the certificate

Common name Attribute: Username

Defines what attribute of the node to use as the common name during the certificate generation.

Ca cert path: /usr/local/pf/conf/ssl/tls_certs/MyCA

Path of the CA that will generate your certificates

Server cert path: /usr/local/pf/conf/ssl/tls_certs/MyCer

Path of the RADIUS server authentication certificate

The "Server cert path" and "CA cert path" both need to be absolute (e.g. `/usr/local/pf/conf/ssl/tls_certs/MyCA.pem` is an absolute path).

The "Common name attribute" field defines how the certificate will be generated and what type of "ownership" will associate the certificate to the connection. If you select *MAC address*, a certificate will be generated using the MAC address as the identifier. If you select *Username*, a certificate will be generated using his login name on the authentication backend.

Provisioners configuration

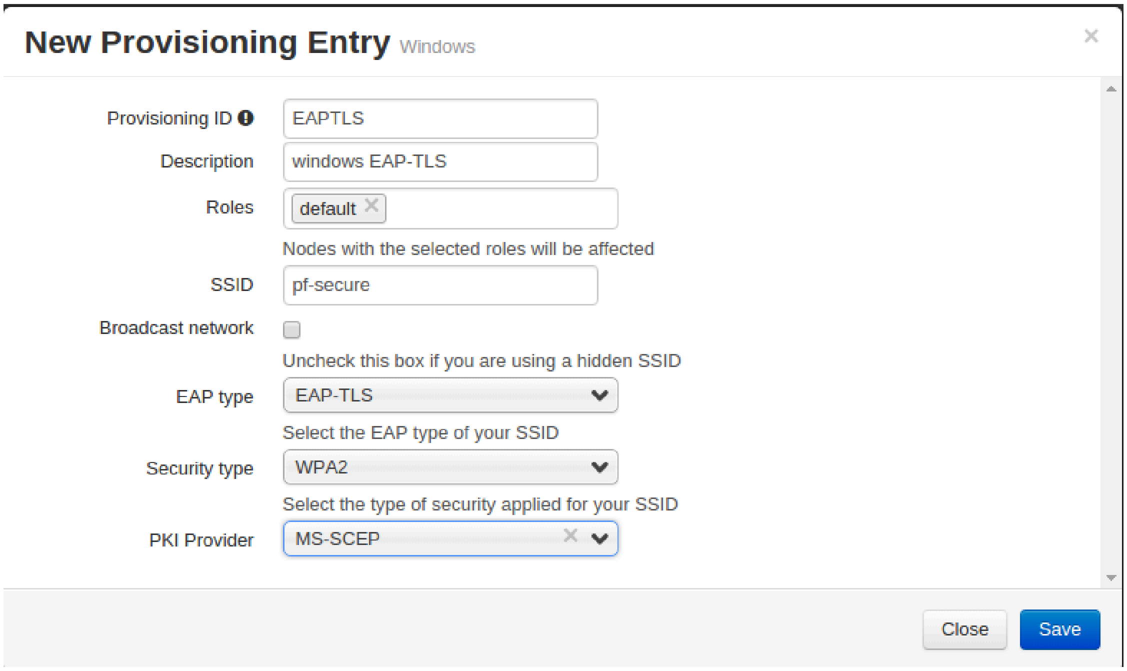
Provisioners allow devices to automatically configure themselves to connect to the proper SSID (if applicable), use the proper authentication method (e.g. EAP-TLS) and trust the CA certificate and any certificate signed by it.

Provisioners are configured in the PacketFence administration GUI under `Configuration->Users->Provisioners`.

Add a new provisioner for each of the classes of devices to be supported amongst Android, Apple Devices and Windows. Fill out the form, choosing a different Provisioning Id per provisioner.

- Roles: The "Roles" field defines which devices will be affected by the provisioning item. If empty, all devices for this class will be affected.
- SSID: The "SSID" field defines which SSID will be configured on the device using the authentication profile.
- EAP-Type: The EAP type defines the authentication method supported and should be set to EAP-TLS to integrate with the PacketFence PKI.
- Security type: The security type should be set to WPA2 to integrate with the PacketFence PKI.
- PKI Provider: This should match the provider you configured earlier in the PKI provider section.

The following is an example on how to configure an EAP-TLS connection for Windows/Android/Mac OS X/iOS



New Provisioning Entry Windows

Provisioning ID: EAPTLS

Description: windows EAP-TLS

Roles: default ✕

Nodes with the selected roles will be affected

SSID: pf-secure

Broadcast network: □

Uncheck this box if you are using a hidden SSID

EAP type: EAP-TLS

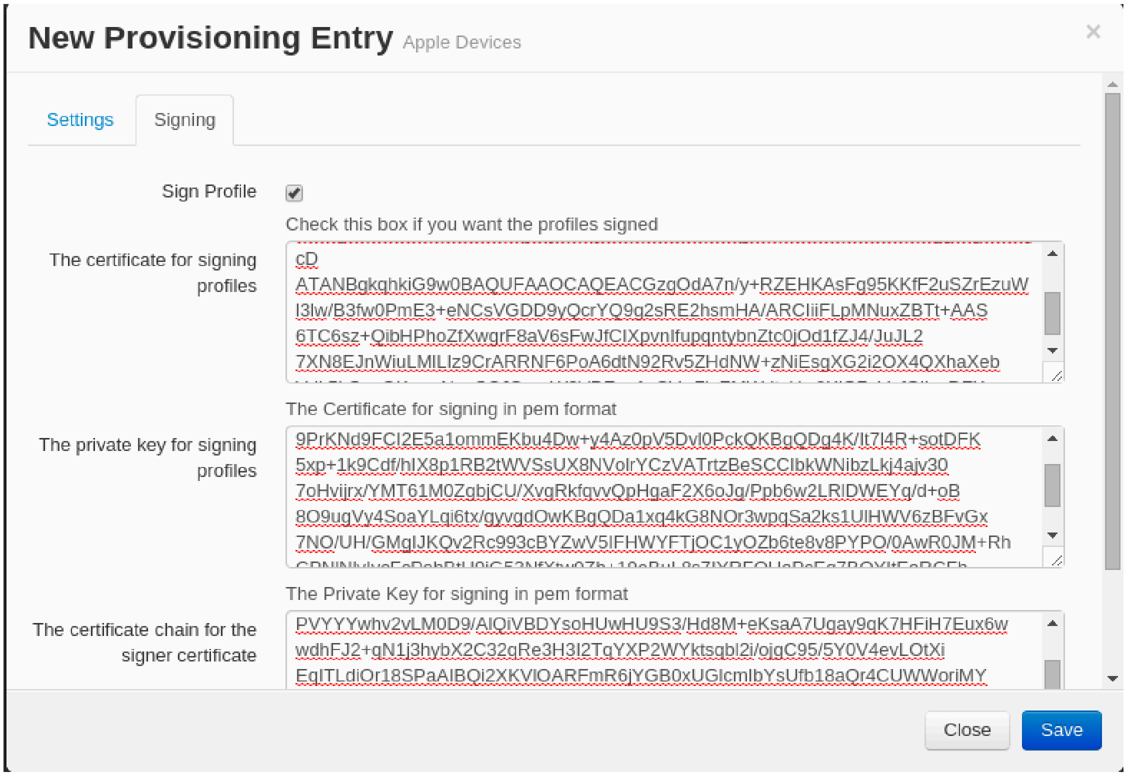
Select the EAP type of your SSID

Security type: WPA2

Select the type of security applied for your SSID

PKI Provider: MS-SCEP

Mac OS X/iOS require the provisioning profile to be signed if you want to remove the `untrusted` warning when installing the profile. You need to sign it with a Certification Authority already trusted by the device such as e.g. VeriSign. Configuring this has to be done in the *Signing* tab in the "Apple devices".



New Provisioning Entry Apple Devices

Sign Profile: ☑

Check this box if you want the profiles signed

The certificate for signing profiles:

cD
ATANBgkqhkiG9w0BAQUFAAOCAQEACGzgOdA7n/y+RZEHKAsFg95KKfF2uSZrEzuW
l3lw/B3fw0PmE3+eNCsVGDD9yQcrYQ9g2sRE2hsmHA/ARCIiiFLpMNuxZBTt+AAS
6TC6sz+QibHPhoZfXwgrF8aV6sFwJfCIXpvnIfupqntybnZtc0jOd1fZJ4/JuJL2
7XN8EJnWiuLMILIz9CrARRNF6PoA6dtN92Rv5ZHdNW+zNiEsgXG2i2OX4QXhaXeb

The Certificate for signing in pem format

The private key for signing profiles:

9PrKNd9FCI2E5a1ommEKbu4Dw+y4Az0pV5Dvl0PckQKBgQDg4K/It7I4R+sotDFK
5xp+1k9Cdf/hIX8p1RB2tWVSsUX8NVolrYCzVATrtzBeSCCIbkWNibzLkj4ajv30
7oHvijrx/YMT61M0ZgbjCU/XvgRkfqvvQpHgaF2X6oJg/Ppb6w2LRIDWEYq/d+oB
8O9ugVy4SoaYLqi6tx/gyvgdOwKBgQDa1xq4kG8NOr3wpqSa2ks1UlHWV6zBFvGx
7NO/UH/GMgIJKQv2Rc993cBYZwV5IFHWYFTjOC1yOZb6te8v8PYPO/0AwR0JM+Rh

The Private Key for signing in pem format

The certificate chain for the signer certificate:

PVYYYwhv2vLM0D9/AIQiVBDYsoHUwHU9S3/Hd8M+eKsaA7Ugay9qK7HFiH7Eux6w
wdhFJ2+qN1j3hybX2C32qRe3H3I2TqYXP2WYktsqbI2i/ojgC95/5Y0V4evLOtXi
EqITLdiOr18SPaAIBQi2XKVlOARFmR6jYGB0xUGIcmIbYsUfb18aQr4CUWWoriMY

Fill out the fields with the contents of the Base64 encoded certificates. To extract this information from a pem formatted certificate, copy the file content.

Certificate file example:

```
----- BEGIN CERTIFICATE -----
1234567890asdfghjkl
zxcvbnmqwertyuiop78
----- END CERTIFICATE -----
```

Copy everything from the BEGIN to END lines. Repeat this operation for the certificate key and intermediate certificate.

```
----- BEGIN PRIVATE KEY -----
1234567890asdfghjkl
zxcvbnmqwertyuiop78
----- END PRIVATE KEY -----
```

Connection Profiles Configuration

Provisioners have to be enabled on the Connection Profiles configuration in the PacketFence GUI.

Under `Configuration->Main->Connection Profiles`, select each of the provisioners created above which should be active for the profile. If no connection profile is defined, configure the "default" profile to use the provisioners created.

Passthroughs required for Android

Android devices require passthroughs to be created to allow them to fetch the configuration application from the Google Play Store.

Important

Passthroughs will vary depending on the location where your Google account was created. You will need to add some extra passthroughs for the store of your country. In the section debug there is a how-to determine which address you need to add.

Add the following to the "Fencing" section of the Configuration tab in the PacketFence GUI.

```
passthrough=enabled
passthroughs=*.ggpht.com,*.googleusercontent.com,android.clients.google.com,
  *.googleapis.com,*.android.clients.google.com,*.gvt1.com
```

Chapter 4

Appendix

Debugging MSPKI integration with PacketFence

This is a way to do the procedure of enrollment manually, mainly for debugging purposes.

First you need to generate a request and its private key via the openssl command. Type the following command in PacketFence CLI

```
# openssl req new -newkey rsa:2048 -nodes -keyout local.key -out local.csr -subj
 '/C=CA/ST=QC/L=Montreal/O=Inverse/OU=IT/CN=www.test.example.com'
```

This will create 2 files in your current directory, local.csr and local.key.

Now you need to obtain the CA and some specific certificates from the MSPKI.

```
# sscep getca -u http://<ServerDNSName>/CertSrv/mscep/ -c path/MyCA
```

Now you need to use the "CEP encryption" certificate and the "Enrollment agent". Both were obtained when doing the getca. You should have at least three certificates with the same name and a different number at the end. e.g. MyCA-0, MyCA-1, etc.

To display the content of a certificate use the following command:

```
# openssl x509 -in path/MyCA-0 -text
# openssl x509 -in path/MyCA-1 -text
```

In the output search for `X509v3 extensions:`. When using the scep enroll command you will need the "CEP Encryption" certificate as an argument for -e and the "Enrollment agent" certificate as an argument for -c. -d is use for the debug output. -l is the local file where your certificate will be save.

```
# sscep enroll -c path/MyCA-0 -e path/MyCA-1 -k path/local.key -r path/local.csr
 -l path/MyCert.crt -S sha1 -u http://<ServerDNSName>/CertSrv/mscep/ -d
```

To verify your certificate against the OCSP you can use the following openssl command:

```
# openssl OCSP -issuer path/CA-Certificate -cert path/Certificate-to-verify -text
 -url http://<ServerDNSName>/OCSP
```

Debugging Android passthroughs

If you need to add domains to passthroughs, we advise you to capture the traffic coming from the device which cannot access the Google Play Store. To do this you can use tcpdump for instance, collect the IP address of the device then run the following in PacketFence CLI:

```
tcpdump -i $REGISTRATION_INTERFACE -n dst port 53 and src host @IP_Device
```

This will output any DNS requests from the device to PacketFence. You will need to find **google** related domain and add them to your passthroughs list.