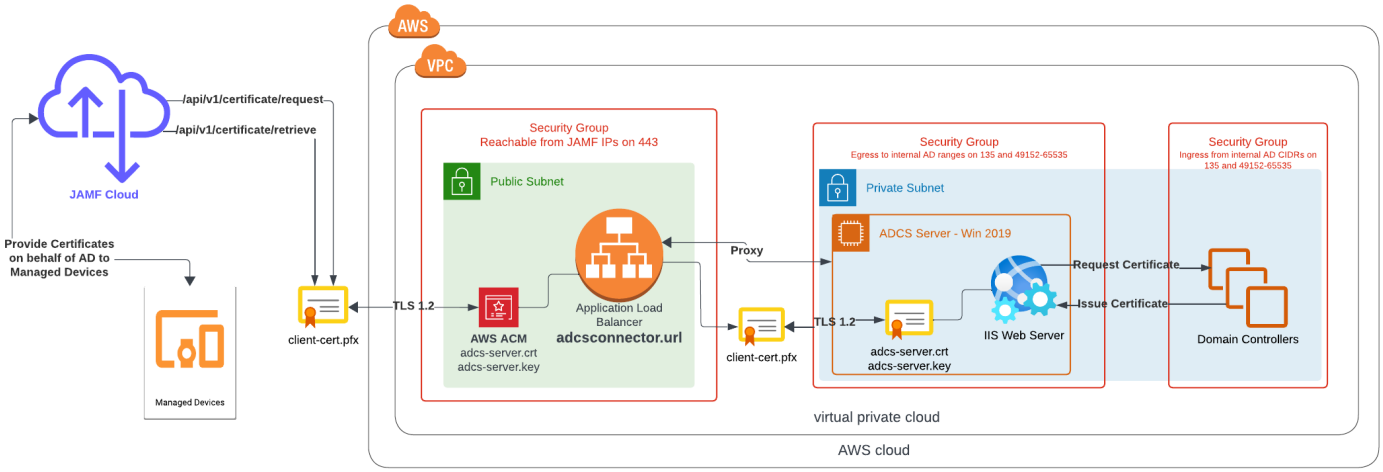
Jamf ADCS Connector using AWS Application Load Balancer

This guide aims to assist in the necessary steps required if you wish to deploy the Jamf AD CS connector behind an application load balancer in AWS.

The Jamf internal documents show only a high level architecture diagram for this, and the diagram makes it look like the client-cert.pfx is what is required to terminate TLS server side, which is incorrect.

Additionally, an AWS application load balancer has to be configured a specific way, so knowing how to get the certificate in place and make sure health checks work for the target group is important. Application load balancers fail open, so even if all targets are unhealthy and failing checks the requests would still route, but with the configurations below, the nice green "Healthy" shows in the AWS console.

Architecture Overview



This entire guide is simply to export the server certificate from the IIS server created by the ADCS Connector installation tool and import it into AWS ACM for use on the load balancer. *NOTE: This guide does not include how to configure or deploy an AWS load balancer. That is out of scope.*

Requirements:

- Access to the AD CS connector host (IIS server)
- Knowledge of the installation location on the IIS server
- Access to AWS to create ACM certificates

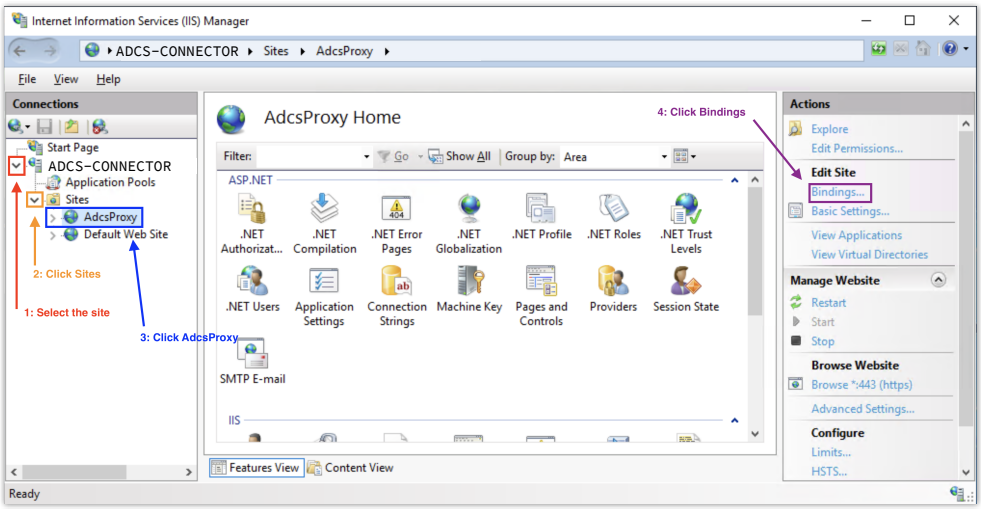
Uploading to AWS

Because of how the connector works with TLS auth, in a reverse proxy / ALB setup we need to take the same server cert that is used by IIS and upload it to ACM for our ALB.

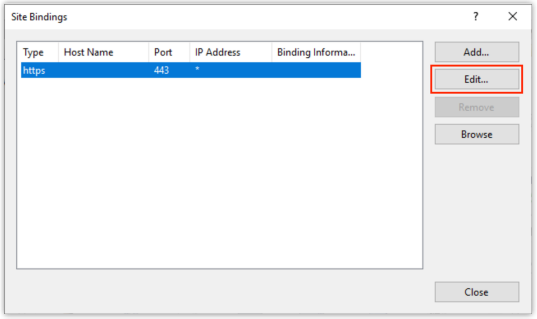
This requires exporting the server cert from the IIS host and creating an ACM cert with the values.

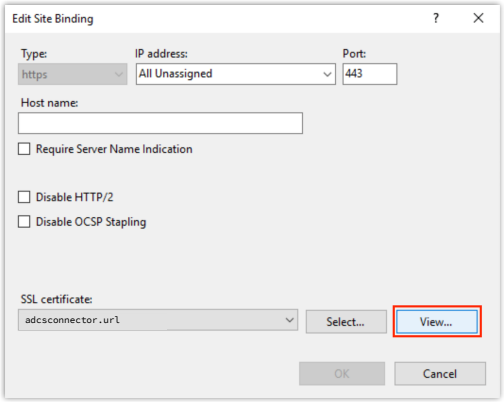
Identifying the Server Certificate

1. Log into the AD CS connector Windows host
2. Open IIS Manager. You can use **Windows + R** and type **inetmgr**, or search for IIS
3. In the left pane, click to expand the site (it is probably the only one)
4. Expand **Sites**
5. Select **AdcsProxy**
6. In the right pane, under **Actions**, click on **Bindings**



7. Select the HTTPS binding, and click Edit. Next to the SSL Certificate at the bottom of the window, click View. This will show you the details of the certificate, usually identified by the hostname you chose as the flag to **-fqdn** when you ran the **.\deploy.ps1** command.

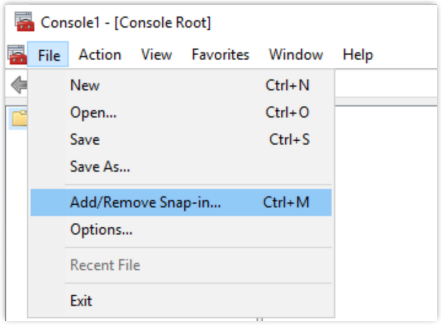




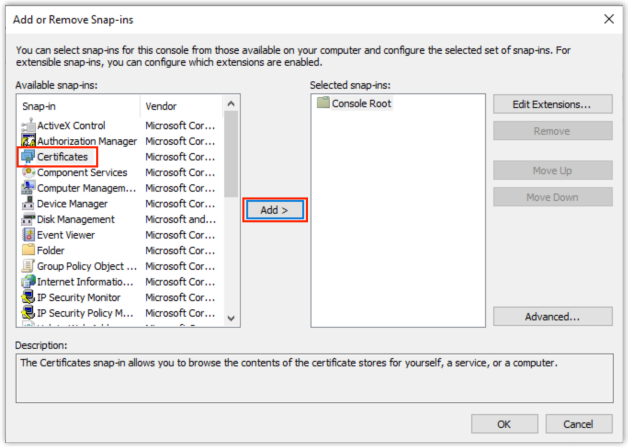
8. Now we need to export the server certificate.

Exporting the Server Certificate

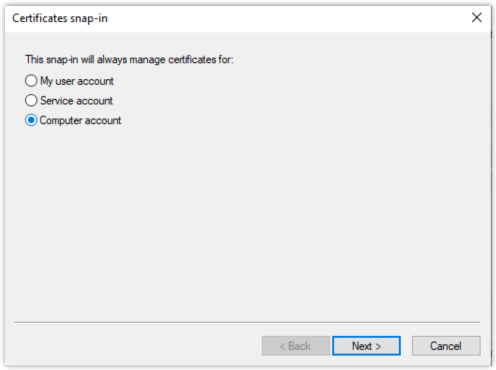
1. Open **mmc** (**Windows + R** and type **mmc**)
2. Go to **File → Add/Remove Snap-In**

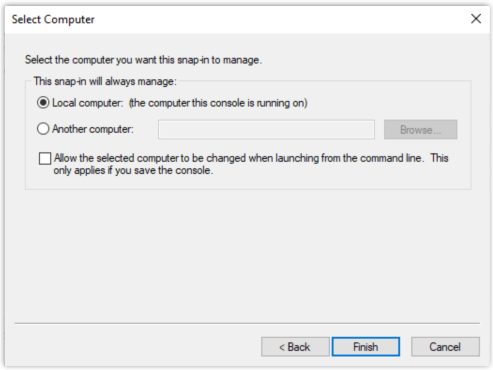


3. Select **Certificates** and click **Add**

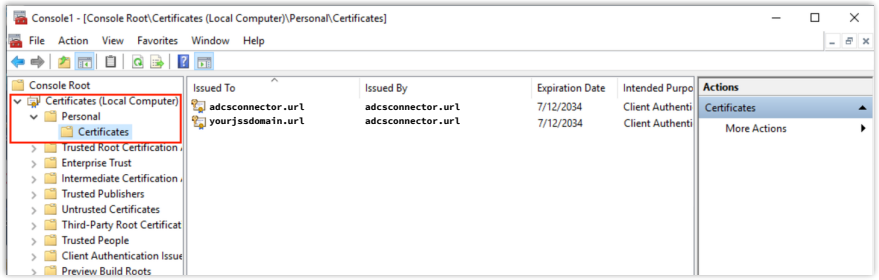


4. Choose **Computer Account**, click **Next**, then select **Local Computer** and click **Finish** and then **OK**.

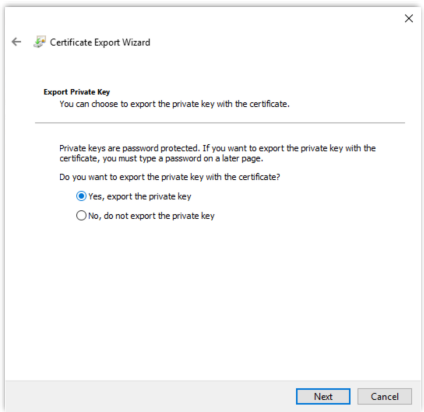




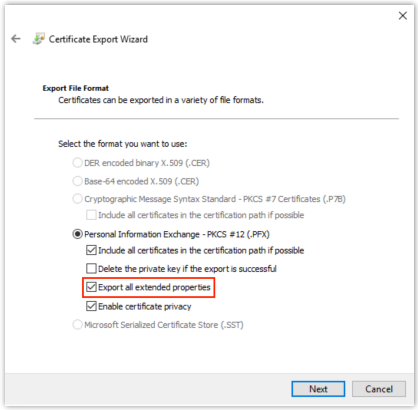
5. Expand **Certificates (Local Computer)** in the left pane
6. Navigate to **Personal → Certificates**
7. Locate the certificate in the left pane that was identified in the steps above (usually the fqdn of the ADCS connector host)



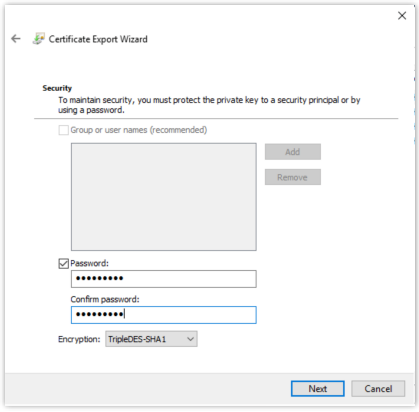
8. Right click on the certificate, select **All Tasks → Export**
9. In the **Certificate Export Wizard**, select **Yes, export the private key** and click **Next**



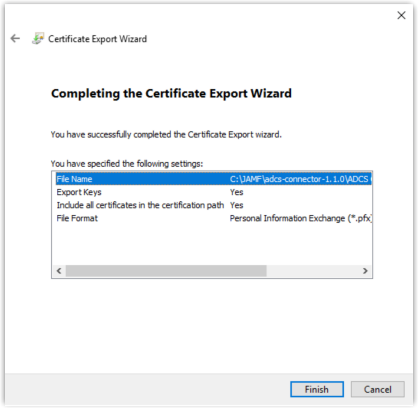
10. Select **Export all extended properties** and click **Next**



11. Select **Password** and enter a password to use to create the export. You will need it later, so make sure its something you can remember.



12. Choose a location to save the exported **pfx** file, and click **Next**
13. Review the information and click **Finish**



Converting the pfx for use with AWS ACM

Now that we have the server cert, we need to convert it from **pfx** to **pem** so that we can use it with ACM.

1. Export the key. It will ask you for a password—this was the password we set in step 11 above, and we will remove it in the next step:

```
openssl pkcs12 -in adcs-server.pfx -nocerts -out adcs-server.key
```

2. Remove the password from the key file (you will be prompted):

```
openssl rsa -in adcs-server.key -out adcs-server-nopass.key
mv adcs-server-nopass.key adcs-server.key
```

3. Export the cert:

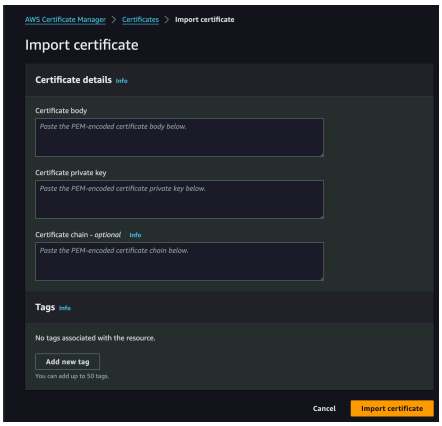
```
openssl pkcs12 -in adcs-server.key -clcerts -nokeys -out adcs-server.crt
```

4. Make sure you also have the **adcs-proxy-ca.cer** file from earlier steps. If you do not have it, you can get it from the Jamf AD CS host located in the location where you ran the **.\deploy.ps1** installer, e.g. **C:\JAMF\adcs-connector-1.1.0\ADCS Connector\adcs-proxy-ca.cer**

5. Convert the chain **cer** to **pem**:

```
openssl x509 -inform der -in adcs-proxy-ca.cer -out adcs-proxy-ca.pem
```

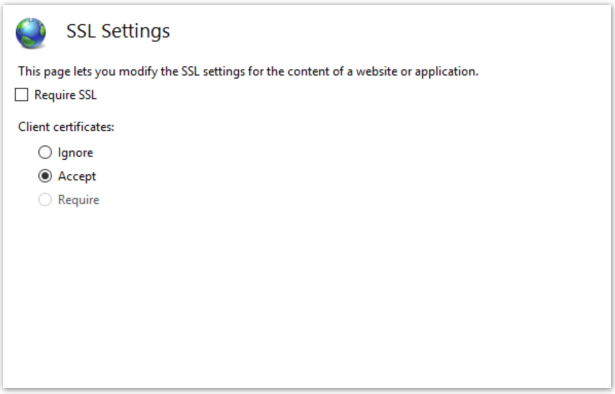
6. Now that you have a key, cert, and chain, you can use upload them to ACM, and associate the uploaded ACM cert with your load balancer!



Load Balancer Health Checks

The load balancer does not have the client cert, so in order for health checks to pass, we need to change some IIS settings:

1. Open the **IIS Manager**
2. Select the site
3. Double Click on **SSL Settings**
4. Uncheck the box for **Require SSL**. Since we only allow traffic to the host on 443, non SSL requests will fail anyway, except for the load balancer health checks. Make sure to restart the site afterwards.



Troubleshooting

If you run into any issues, there are a few things you can check to make sure the load balancer is configured correctly.

ELB Security Policy

Jamf's AD CS connector does not support TLS 1.3. If your ELB Security Policy is set to **ELBSecurityPolicy-TLS13-1-2-2021-06**, make sure to change it to **ELBSecurityPolicy-TLS-1-2-Ext-2018-06**

Security Group Ports

Make sure that the load balancer's security group is accessible from the JAMF IP address from this page marked as **Outbound Traffic from Jamf Cloud** on port 443.

Only your AD CS host needs to be about to communicate outbound to your AD Certificate Server on 135 and 49152-65535.

AWS ACM says "Key does not match certificate"

Make sure to follow the steps above carefully for exporting the server cert from the ADCS host, and splitting it into a key and cert, and grabbing the chain from the files created by the installer. If you accidentally flip the cert chain and cert body you will get an error.

Additional Resources

- Installing the Jamf AD CS Connector
- Importing certificates into AWS Certificate Manager