

Déterminer le certificat correct pour LDAPS

Contenu

[Introduction](#)

[Conditions préalables](#)

[Conditions requises](#)

[Components Used](#)

[Informations générales](#)

[Déterminer s'il peut y avoir un problème avec le ou les certificats.](#)

[Pour déterminer le certificat/la chaîne à utiliser.](#)

Introduction

Ce document décrit comment déterminer le ou les certificats corrects pour le protocole LDAP (Lightweight Directory Access Protocol) sécurisé.

Conditions préalables

Conditions requises

Aucune spécification déterminée n'est requise pour ce document.

Components Used

Ce document n'est pas limité à des versions de matériel et de logiciel spécifiques.

The information in this document was created from the devices in a specific lab environment. All of the devices used in this document started with a cleared (default) configuration. If your network is live, make sure that you understand the potential impact of any command.

Informations générales

Le protocole LDAP sécurisé nécessite que le domaine Unified Computing System (UCS) dispose du certificat ou de la chaîne de certificats appropriés installés en tant que point de confiance.

Si un certificat (ou une chaîne) incorrect est configuré, ou s'il n'en existe aucun, l'authentification échoue.

[Déterminer s'il peut y avoir un problème avec le ou les certificats.](#)

Si vous rencontrez des problèmes avec Secure LDAP, utilisez le débogage LDAP pour vérifier si les certificats sont corrects.

```
[username]
[password]
connect nxos      *(make sure we are on the primary)
debug ldap all
term mon
```

Ensuite, ouvrez une deuxième session et essayez de vous connecter avec vos informations d'identification LDAP sécurisées.

La session avec débogage activé enregistre la tentative de connexion. Sur la session de journalisation, exécutez la commande **undebug** pour arrêter la sortie.

```
undebug all
```

Pour déterminer s'il existe un problème potentiel avec le certificat, consultez la sortie de débogage de ces lignes.

```
2018 Sep 25 10:10:29.144549 ldap: ldap_do_process_tls_resp: (user f-ucsapac-01) - ldap start TLS
sent succesfully;      Calling ldap_install_tls
2018 Sep 25 10:10:29.666311 ldap: ldap_do_process_tls_resp: (user f-ucsapac-01) - TLS START
failed
```

En cas d'échec de TLS, une connexion sécurisée n'a pas pu être établie et l'authentification échoue.

Pour déterminer le certificat/la chaîne à utiliser.

Une fois que vous avez déterminé qu'il n'y a pas eu d'échec de l'établissement de la connexion sécurisée, déterminez quel(s) certificat(s) correct(s) doit(doivent) être.

Utilisez ethanalyzer pour capturer la communication, puis extrayez le certificat (ou la chaîne) du fichier.

Dans votre session de débogage, exécutez la commande suivante :

```
ethanalyzer local interface mgmt capture-filter "host <address of controller/load balancer>"
limit-captured-frames 100 write volatile:ldap.pcap
```

Ensuite, essayez une autre connexion via avec vos informations d'identification.

Une fois que vous ne voyez plus de nouvelle sortie dans la session de débogage, supprimez la capture. Utiliser (**ctrl + c**).

Transférez la capture de paquets à partir de l'interconnexion de fabric (FI) à l'aide de cette commande :

```
copy volatile:ldap.pcap tftp:
```

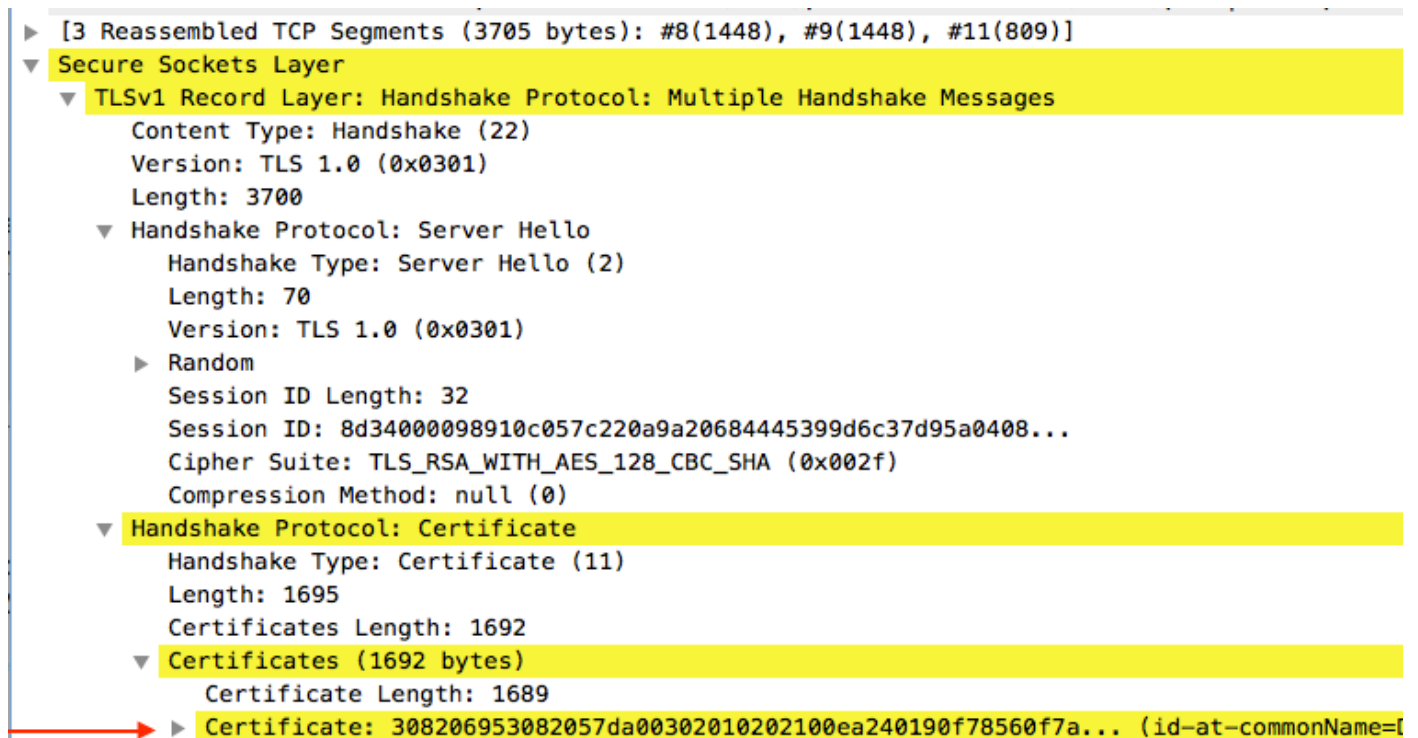
Une fois que vous avez le fichier ldap.pcap, ouvrez le fichier dans Wireshark et recherchez un paquet qui commence à initialiser la connexion TLS.

Vous pouvez voir un message similaire dans la section **Info** du paquet, comme illustré dans l'image :

Server Hello, Certificate, Certificate Request, Server Hello Done			
7	0.498834	SSLv2	190 Client Hello
8	0.753397	TCP	1514 [TCP segment of a reassembled PDU]
9	0.755902	TCP	1514 [TCP segment of a reassembled PDU]
10	0.755940	TCP	66 56328 - 3268 [ACK] Seq=156 Ack=2943 Win=11776 Len=0 TSval=1166916677 TSecr=112994803
11	1.005008	TLSv1	875 Server Hello, Certificate, Certificate Request, Server Hello Done
12	1.007214	TLSv1	73 Alert (Level: Fatal, Description: Unknown CA)

Sélectionnez ce paquet et développez-le :

```
Secure Sockets Layer
-->TLSv? Record Layer: Handshake Protocol: Multiple Handshake Messages
---->Handshake Protocol: Certificate
----->Certificates (xxxx bytes)
```



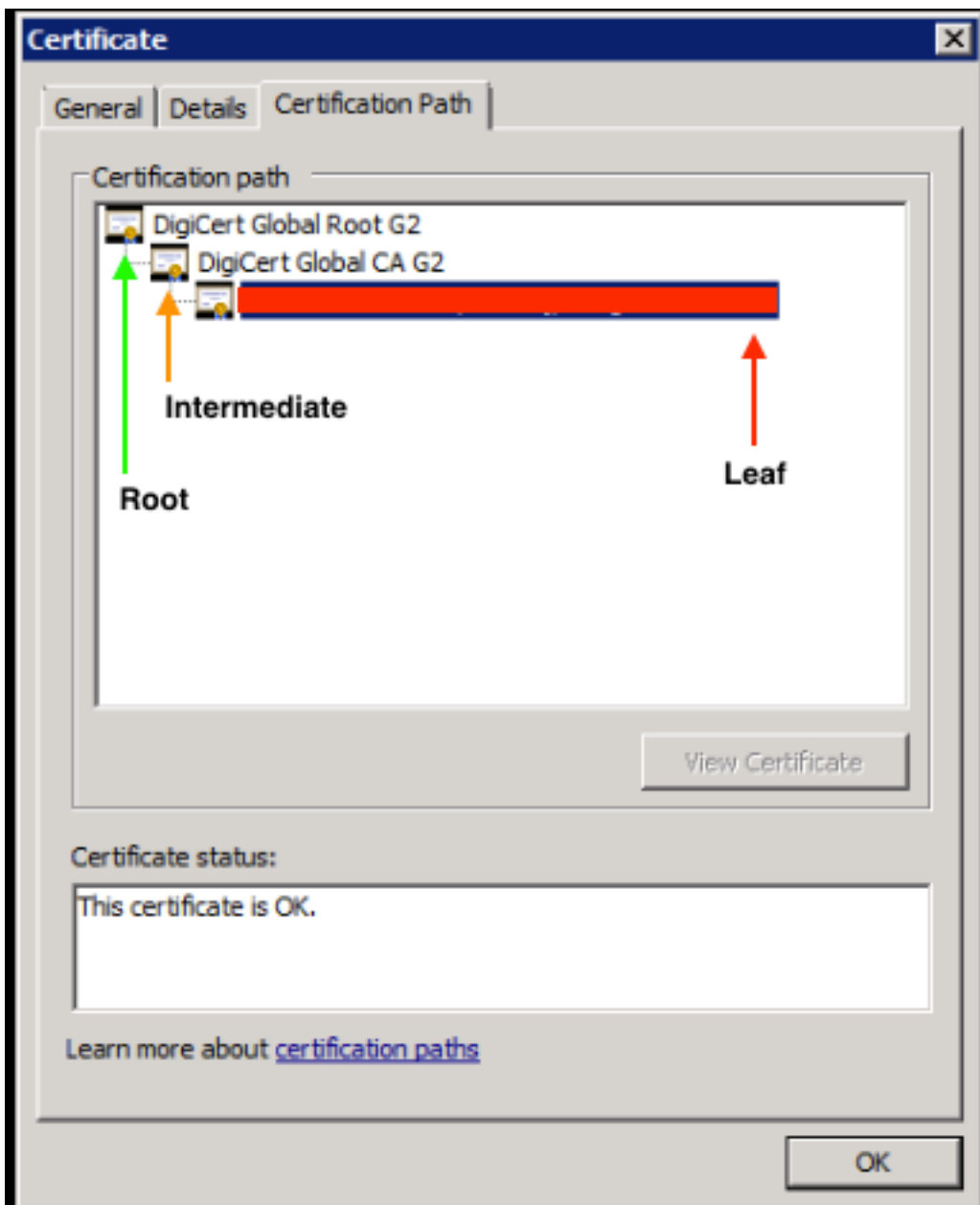
Sélectionnez la ligne intitulée **Certificat**.

Cliquez avec le bouton droit sur cette ligne et sélectionnez **Exporter les octets de paquets** et enregistrez le fichier en tant que fichier **.der**.

Ouvrez le certificat dans Windows et accédez à l'onglet **Chemin d'accès du certificat**.

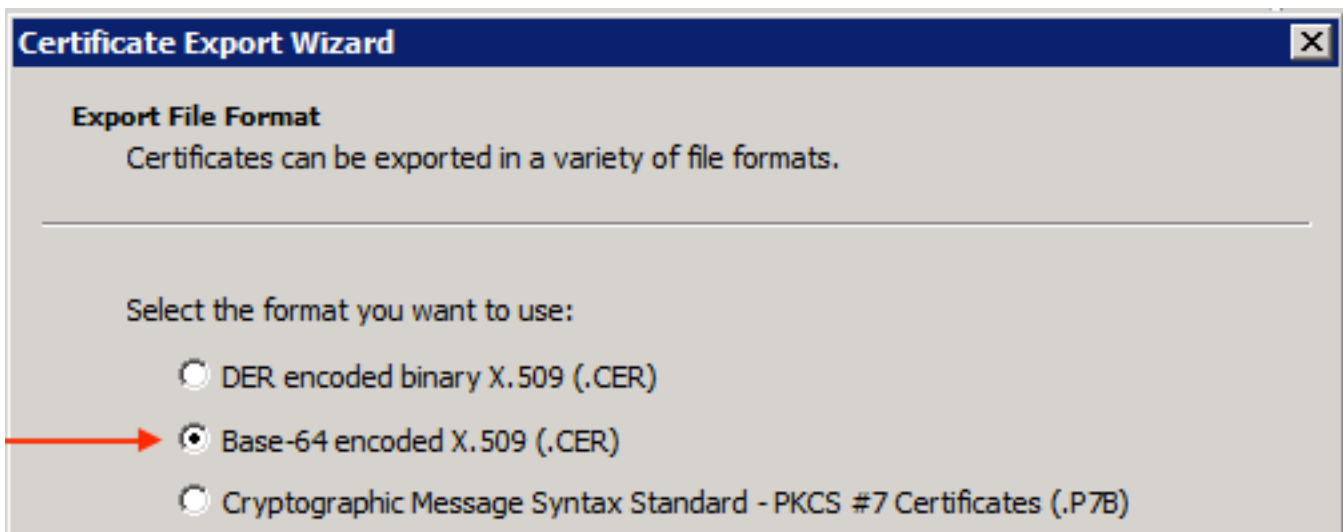
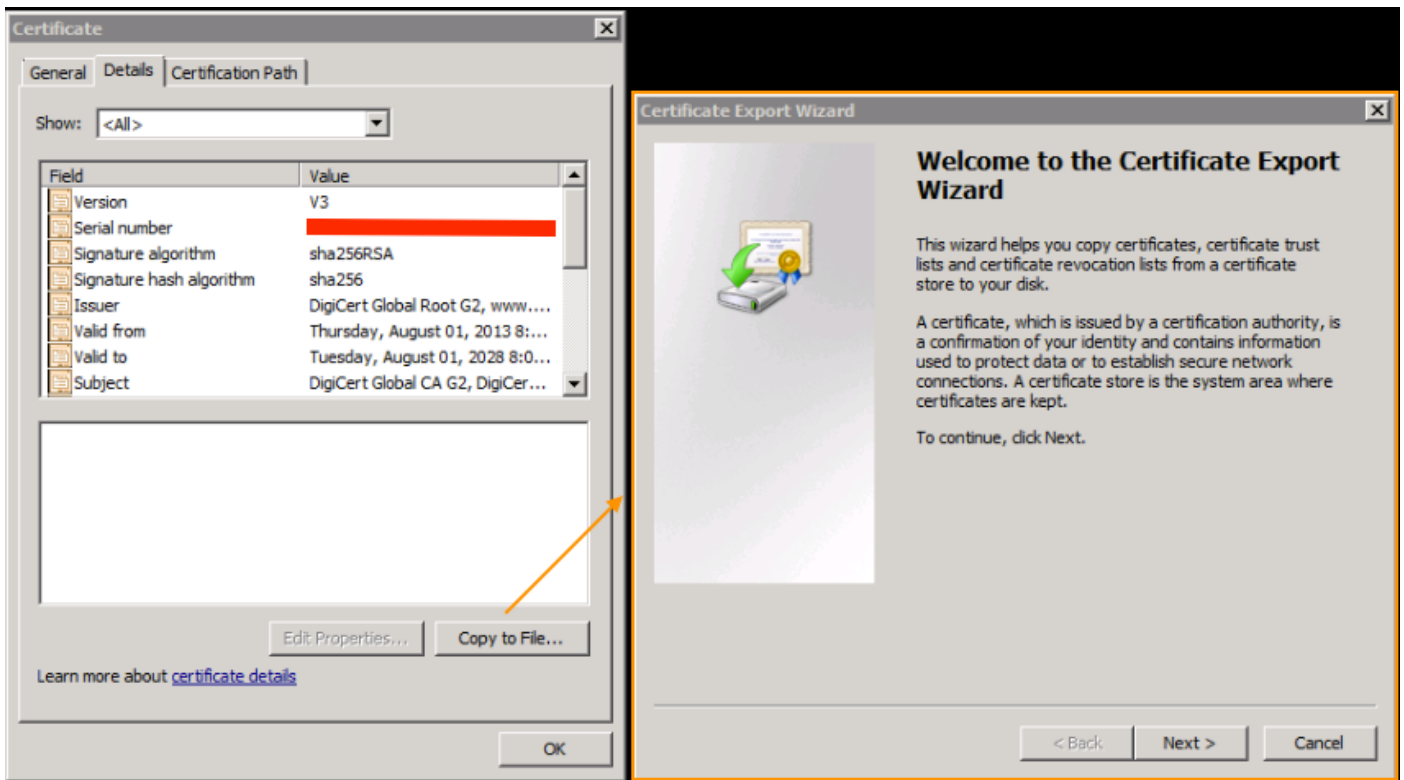
Ceci vous montre le chemin complet du certificat **racine** au **terminal** (hôte final). Effectuez les actions suivantes pour tous les noeuds répertoriés, à l'exception de la **feuille**.

```
Select the node
-->Select 'View Certificate'
---->Select the 'Details' tab
```



Sélectionnez l'option **Copier dans un fichier** et suivez l'**Assistant Exportation de certificat** (assurez-vous d'utiliser le format codé Base-64).

Cela génère un fichier **.cer** pour chacun des noeuds de la liste lorsque vous les terminez.



Ouvrez ces fichiers dans Bloc-notes, Bloc-notes++, Sublime, etc. pour afficher le certificat haché.

Pour générer la chaîne (s'il en existe une), ouvrez un nouveau document et collez-le dans le certificat haché du dernier noeud.

Développez la liste en collant chaque certificat haché et en finissant par l'**autorité de certification racine**.

Collez l'**autorité de certification racine** (s'il n'y a pas de chaîne) ou la chaîne entière que vous avez générée dans le point de confiance.