Réalisations et comportement de fragmentation d'EAP

Contenu

Introduction

Ce document décrit comment comprendre et dépanner des sessions de Protocole EAP (Extensible Authentication Protocol). Ces questions sont discutées :

- Comportement des serveurs d'Authentification, autorisation et comptabilité (AAA) quand ils renvoient le certificat de serveur pour la session de Layer Security de Protocol-transport d'authentification extensible (EAP-TLS)
- Comportement des suppliants quand ils renvoient le certificat client pour la session d'EAP-TLS
- Interopérabilité quand le suppliant indigène de Microsoft Windows et le gestionnaire d'accès au réseau de Cisco AnyConnect (NAM) sont utilisés
- Fragmentation dans le processus IP, de RAYON, et d'EAP-TLS et de remontage exécuté par

des périphériques d'accès au réseau
- L'attribut d'unité de transmission d'Encadrer-maximum de RAYON (MTU)
- Le comportement des serveurs d'AAA quand ils exécutent la fragmentation des paquets d'EAP-TLS

Conditions préalables

Conditions requises

Cisco vous recommande de prendre connaissance des rubriques suivantes :

- Protocoles d'EAP et d'EAP-TLS
- Configuration du Logiciel Cisco Identity Services Engine (ISE)
- Configuration CLI des commutateurs Cisco Catalyst

Il est nécessaire d'avoir une bonne compréhension de l'EAP et de l'EAP-TLS afin de comprendre cet article.

Chaîne de certificat retournée par le serveur

Le serveur d'AAA (serveur de contrôle d'accès (ACS) et ISE) renvoie toujours la pleine chaîne pour le paquet d'EAP-TLS avec le serveur bonjour et le certificat de serveur :

Le certificat d'identité ISE (nom commun (NC) =lise.example.com) est renvoyé avec l'Autorité de certification (CA) qui a signé le CN=win2012,dc=example,dc=com. Le comportement est identique pour ACS et ISE.

Chaîne de certificat retournée par le suppliant

Suppliant d'indigène de Microsoft Windows

Le suppliant d'indigène de Microsoft Windows 7 configuré afin d'utiliser l'EAP-TLS, avec ou sans « la sélection simple de certificat », n'envoie pas la pleine chaîne du certificat client. Ce comportement se produit même lorsque le certificat client est signé par un CA différent (chaîne différente) que le certificat de serveur.

Cet exemple est lié au serveur bonjour et au certificat présenté dans le tir d'écran précédent. Pour ce scénario, le certificat ISE est signé par le CA avec l'utilisation d'un nom du sujet, CN=win2012,dc=example,dc=com. Mais le certificat utilisateur installé dans la mémoire de Microsoft est signé par un CA différent, CN=CA, C=PL, S=Cisco CA, L=Cisco CA, O=Cisco CA.

En conséquence, le suppliant de Microsoft Windows répond avec le certificat client seulement. Le CA qui le signe (CN=CA, S=PL, S=Cisco CA, L=Cisco CA, O=Cisco CA) n'est pas relié.

En raison de ce comportement, les serveurs d'AAA pourraient rencontrer des problèmes quand ils valident des certificats client. L'exemple associe au professionnel SP1 de Microsoft Windows 7.

Solution

Une pleine chaîne de certificat devrait être installée sur le stock de certificat d'ACS et d'ISE (tous les certificats client de signature CA et de sous-titre CA).

Des problèmes avec la validation de certificat peuvent être facilement détectés sur ACS ou ISE. Les informations sur le certificat non approuvé sont présentées et des états ISE :

```
12514 EAP-TLS failed SSL/TLS handshake because of an unknown CA in the client
certificates chain
```

Les problèmes avec la validation de certificat sur le suppliant ne sont pas facilement décelables. Habituellement le serveur d'AAA répond que le « point final a abandonné la session d'EAP » :

AnyConnect NAM

L'AnyConnect NAM n'a pas cette limite. Dans le même scénario, il relie la chaîne complète du certificat client (le CA correct est relié) :

Suppliant indigène de Microsoft Windows avec AnyConnect NAM

Quand les deux services sont, AnyConnect NAM a la priorité. Même lorsque le service NAM ne fonctionne pas, il s'accroche toujours sur Microsoft Windows l'API et en avant les paquets d'EAP, qui peuvent poser des problèmes pour le suppliant d'indigène de Microsoft Windows. Voici un exemple d'une telle panne.

Vous activez le suivi sur Microsoft Windows avec cette commande :

```
C:\netsh ras set tracing * enable
```

L'exposition de suivis (**c:\windows\trace\svchost _RASTLS.LOG**) :

```
[2916] 09-14 21:29:11:254: >> Received Request (Code: 1) packet: Id: 55, Length:
6, Type: 13, TLS blob length: 0. Flags: S
[2916] 09-14 21:29:11:254: << Sending Response (Code: 2) packet: Id: 55, Length:
105, Type: 13, TLS blob length: 95. Flags: L
[1804] 09-14 21:29:11:301: >> Received Request (Code: 1) packet: Id: 56, Length:
1012, Type: 13, TLS blob length: 2342. Flags: LM
[1804] 09-14 21:29:11:301: << Sending Response (Code: 2) packet: Id: 56, Length:
6, Type: 13, TLS blob length: 0. Flags:
[1804] 09-14 21:29:11:348: >> Received Request (Code: 1) packet: Id: 57, Length:
1008, Type: 13, TLS blob length: 0. Flags: M
[1804] 09-14 21:29:11:348: << Sending Response (Code: 2) packet: Id: 57, Length:
6, Type: 13, TLS blob length: 0. Flags:
[1804] 09-14 21:29:11:363: >> Received Request (Code: 1) packet: Id: 58, Length:
344, Type: 13, TLS blob length: 0. Flags:
[1804] 09-14 21:29:11:363: << Sending Response (Code: 2) packet: Id: 58, Length:
1492, Type: 13, TLS blob length: 1819. Flags: LM
[3084] 09-14 21:31:11:203: >> Received Request (Code: 1) packet: Id: 122, Length:
6, Type: 13, TLS blob length: 0. Flags: S
[3084] 09-14 21:31:11:218: << Sending Response (Code: 2) packet: Id: 122, Length:
```

```
105, Type: 13, TLS blob length: 95. Flags: L
[3420] 09-14 21:31:11:249: >> Received Request (Code: 1) packet: Id: 123, Length:
1012, Type: 13, TLS blob length: 2342. Flags: LM
[3420] 09-14 21:31:11:249: << Sending Response (Code: 2) packet: Id: 123, Length:
6, Type: 13, TLS blob length: 0. Flags:
[3420] 09-14 21:31:11:281: >> Received Request (Code: 1) packet: Id: 124, Length:
1008, Type: 13, TLS blob length: 0. Flags: M
[3420] 09-14 21:31:11:281: << Sending Response (Code: 2) packet: Id: 124, Length:
6, Type: 13, TLS blob length: 0. Flags:
[3420] 09-14 21:31:11:281: >> Received Request (Code: 1) packet: Id: 125, Length:
344, Type: 13, TLS blob length: 0. Flags:
[3420] 09-14 21:31:11:296: << Sending Response (Code: 2) packet: Id: 125, Length:
1492, Type: 13, TLS blob length: 1819. Flags: LM
```

Le dernier paquet est un certificat client (fragment 1 d'EAP-TLS avec taille 1492 d'EAP) envoyé par le suppliant d'indigène de Microsoft Windows. Malheureusement, Wireshark n'affiche pas ce paquet :

Et ce paquet n'est pas vraiment envoyé (dernier était le troisième fragment du certificat de serveur de transport d'EAP-TLS). Il a été consommé par le module d'AnyConnect NAM ce des crochets sur Microsoft Windows API.

C'est pourquoi on ne lui informe pas utiliser AnyConnect avec le suppliant d'indigène de Microsoft Windows. Quand vous utilisez tous les services d'AnyConnect, on ne lui informe utiliser NAM également (quand les services de 802.1x sont nécessaires), pas le suppliant d'indigène de Microsoft Windows.

Fragmentation

La fragmentation pourrait se produire sur de plusieurs couches :

- IP
- Paires de valeur d'attribut RADIUS (AVP)
- EAP-TLS

Les Commutateurs de Cisco IOS® sont très intelligents. Ils peuvent comprendre des formats d'EAP et d'EAP-TLS. Bien que le commutateur ne puisse pas déchiffrer le TLS perce un tunnel, il est responsable de la fragmentation, et l'assemblage et le remontage des paquets d'EAP quand encapsulation dans Extensible Authentication Protocol au-dessus de RÉSEAU LOCAL (EAPoL) ou de RAYON.

Le protocole d'EAP ne prend en charge pas la fragmentation. Voici un extrait de RFC 3748 (EAP) :

La « fragmentation n'est pas prise en charge dans l'EAP lui-même ; cependant, les différentes méthodes d'EAP peuvent prendre en charge ceci. »

L'EAP-TLS est un tel exemple. Voici un extrait de RFC 5216 (EAP-TLS), la section 2.1.5 (fragmentation) :

« Quand un pair d'EAP-TLS reçoit un paquet d'Eap-demande avec le positionnement de bit M, il DOIT répondre avec une Eap-réponse avec EAP-Type=EAP-TLS et aucune données. Ceci sert de fragment ACK. **Le serveur d'EAP DOIT attendre jusqu'à ce qu'il reçoive l'Eap-réponse avant d'envoyer un autre fragment**. »

La dernière phrase décrit une caractéristique très importante des serveurs d'AAA. Ils doivent attendre l'ACK avant qu'ils puissent envoyer un autre fragment d'EAP. Une règle semblable est utilisée pour le suppliant :

« Le pair d'EAP DOIT attendre jusqu'à ce qu'il reçoive l'Eap-demande avant d'envoyer un autre fragment. »

Fragmentation dans la couche IP

La fragmentation peut se produire seulement entre le périphérique d'accès au réseau (NAD) et le serveur d'AAA (IP/UDP/RADIUS utilisé comme transport). Cette situation se produit quand des essais NAD (commutateur de Cisco IOS) pour envoyer la demande RADIUS qui contient la charge utile d'EAP, qui est un plus grand puis MTU de l'interface :

La plupart des versions de Cisco IOS ne sont pas assez intelligentes et n'essayent pas d'assembler des paquets d'EAP reçus par l'intermédiaire d'EAPoL et de les combiner dans un paquet RADIUS qui peut s'adapter dans le MTU de l'interface physique vers le serveur d'AAA.

Les serveurs d'AAA sont plus intelligents (comme présenté dans les sections suivantes).

Fragmentation dans le RAYON

Ce n'est pas vraiment aucun genre de fragmentation. Selon RFC 2865, un attribut RADIUS simple peut avoir jusqu'à 253 octets de données. En raison du ce, la charge utile d'EAP est toujours transmise dans de plusieurs attributs RADIUS d'Eap-message :

Ces attributs d'Eap-message sont rassemblés et interprétés par Wireshark (l'attribut de « dernier segment » indique la charge utile du paquet entier d'EAP). L'en-tête de longueur dans le paquet d'EAP est to1,012 égal, et quatre RAYONS AVPs sont exigés pour le transporter.

Fragmentation dans l'EAP-TLS

Du même tir d'écran, vous pouvez voir cela :

- La longueur de paquet d'EAP est 1,012
- La longueur d'EAP-TLS est 2,342

Ceci suggère que ce soit le premier fragment d'EAP-TLS et le suppliant devrait attendre plus, qui peuvent être confirmés si vous examinez les indicateurs d'EAP-TLS :

Ce genre de fragmentation se produit le plus souvent dans :

- Access-défi de RAYON envoyé par le serveur d'AAA, qui porte l'Eap-demande avec le certificat de serveur de Secure Sockets Layer (SSL) avec la chaîne entière.
- L'Access-demande de RAYON envoient par le NAD, qui porte l'Eap-réponse avec le certificat client SSL avec la chaîne entière.

Confirmation de fragment d'EAP-TLS

Comme expliqué plus tôt, chaque fragment d'EAP-TLS doit être reconnu avant que des fragments ultérieurs soient envoyés.

Voici un exemple (captures de paquet pour EAPoL entre le suppliant et le NAD) :

Les trames d'EAPoL et le serveur d'AAA renvoient le certificat de serveur :

- Ce certificat est introduit un fragment d'EAP-TLS (paquet 8).
- Le suppliant reconnaît ce fragment (paquet 9).
- Le deuxième fragment d'EAP-TLS est expédié par NAD (paquet 10).
- Le suppliant reconnaît ce fragment (paquet 11).
- Le troisième fragment d'EAP-TLS est expédié par NAD (paquet 12).
- Le suppliant n'a pas besoin de reconnaître ceci ; en revanche, il se poursuit par le certificat client qui commence au paquet 13.

Voici les détails du paquet 12 :

Vous pouvez voir que Wireshark a rassemblé les paquets 8, 10, et 12. La taille de l'EAP fragmente is1,002, 1,002, et 338, qui apporte la taille totale du message d'EAP-TLS à 2342 (la longueur de message totale d'EAP-TLS est annoncée dans chaque fragment). Ceci peut être confirmé si vous examinez des paquets RADIUS (entre le serveur NAD et d'AAA) :

Les paquets RADIUS 4, 6, et 8 portent ces trois fragments d'EAP-TLS. Les deux premiers fragments sont reconnus. Wireshark peut présenter les informations sur l'EAP-TLS fragmente (taille : 1,002 + 1,002 + 338 = 2,342).

Ces scénario et exemple étaient faciles. Le commutateur de Cisco IOS n'a pas eu besoin de changer la taille de fragment d'EAP-TLS.

Fragments d'EAP-TLS rassemblés avec la taille différente

Considérez ce qui se produit quand le MTU NAD vers le serveur d'AAA est de 9,000 octets (trame jumbo) et le serveur d'AAA est également connecté à l'utilisation de l'interface qui prend en charge des Trames étendues. La plupart des suppliants typiques sont connectées à l'utilisation d'un lien 1Gbit avec un MTU de 1,500.

Dans un tel scénario, le commutateur de Cisco IOS exécute l'assemblage et le remontage « asymétriques » d'EAP-TLS et change des tailles de fragments d'EAP-TLS. Voici un exemple pour un grand message d'EAP envoyé par le serveur d'AAA (certificat de serveur SSL) :

1. Le serveur d'AAA doit envoyer un message d'EAP-TLS avec un certificat de serveur SSL. La taille totale de ce paquet d'EAP est 3,000. Après qu'il soit encapsulé dans le RAYON Access-Challenge/UDP/IP, il est encore moins que le MTU d'Interface serveur d'AAA. Un paquet IP simple est envoyé avec des attributs d'Eap-message de 12 RAYONS. Il n'y a aucune fragmentation IP ni d'EAP-TLS.

2. Le commutateur de Cisco IOS reçoit un tel paquet, le désencapsule, et décide que l'EAP doit être envoyé par l'intermédiaire d'EAPoL au suppliant. Puisqu'EAPoL ne prend en charge pas la fragmentation, le commutateur doit exécuter la fragmentation d'EAP-TLS.

3. Le commutateur de Cisco IOS prépare le premier fragment d'EAP-TLS qui peut s'insérer dans le MTU de l'interface vers le suppliant (1,500).

4. Ce fragment est confirmé par le suppliant.

5. Un autre fragment d'EAP-TLS est envoyé après que l'accusé de réception soit reçu.

6. Ce fragment est confirmé par le suppliant.

7. Le dernier fragment d'EAP-TLS est envoyé par le commutateur.

Ce scénario indique cela :

- Dans certaines circonstances, le NAD doit créer des fragments d'EAP-TLS.
- Le NAD est responsable de l'envoi/reconnaissant ces fragments.

La même situation peut se produire pour un suppliant connecté par l'intermédiaire d'un lien qui prend en charge des Trames étendues tandis que le serveur d'AAA a un plus petit MTU (alors le commutateur de Cisco IOS crée des fragments d'EAP-TLS quand il envoie le paquet d'EAP vers le serveur d'AAA).

Encadrer-MTU d'attribut RADIUS

Pour le RAYON, il y a un attribut d'Encadrer-MTU défini dans RFC 2865 :

« Cet attribut indique le Maximum Transmission Unit à configurer pour l'utilisateur, quand il n'est pas négocié par quelques autres moyens (tels que le PPP). Il PEUT être utilisé dans des paquets d'acceptation d'accès. **Il PEUT être utilisé dans un paquet de demande d'accès comme signe par le NAS au serveur qu'il préférerait cette valeur, mais le serveur n'est pas requis d'honorer le signe.** »

ISE n'honore pas le signe. La valeur du l'Encadrer-MTU envoyé par NAD dans l'Access-demande n'a aucune incidence sur la fragmentation exécutée par ISE.

Les plusieurs commutateurs de Cisco IOS modernes ne permettent pas des modifications au MTU de l'interface Ethernet excepté des configurations de Trames étendues activées globalement sur le commutateur. La configuration des Trames étendues affecte la valeur de l'attribut d'Encadrer-MTU introduit l'Access-demande de RAYON. Par exemple, vous avez placé :

```
Switch(config)#system mtu jumbo 9000
```

Ceci force le commutateur pour envoyer l'Encadrer-MTU = 9000 dans toutes les Access-demandes de RAYON. Les mêmes pour le system mtu sans Trames étendues :

```
Switch(config)#system mtu 1600
```

Ceci force le commutateur pour envoyer l'Encadrer-MTU = 1600 dans toutes les Access-demandes de RAYON.

Notez que les commutateurs de Cisco IOS modernes ne te permettent pas pour diminuer la valeur de system mtu en-dessous de 1,500.

Comportement de serveurs et de suppliant d'AAA quand vous envoyez des fragments d'EAP

ISE

ISE essais toujours pour envoyer les fragments d'EAP-TLS (habituellement serveur bonjour avec le certificat) qui sont de 1,002 octets de long (bien que le dernier fragment est habituellement plus petit). Il n'honore pas l'Encadrer-MTU de RAYON. Il n'est pas possible de le modifier pour envoyer de plus grands fragments d'EAP-TLS.

Policy server de réseau Microsoft (NPS)

Il est possible de configurer la taille des fragments d'EAP-TLS si vous configurez l'attribut d'Encadrer-MTU localement sur NPS.

L'événement bien que le configurer la taille de charge utile d'EAP sur l'article de Microsoft NPS mentionne que la valeur par défaut d'un MTU encadré pour le serveur de RAYON NPS est 1,500, le laboratoire du centre d'assistance technique Cisco (TAC) a prouvé qu'il envoie 2,000 avec les valeurs par défaut (confirmées sur un centre d'hébergement de Microsoft Windows 2012).

On le teste que plaçant l'Encadrer-**MTU localement** selon le guide précédemment mentionné est respecté par NPS, et il fragmente les messages d'EAP dans des fragments d'une taille réglée dans l'Encadrer-MTU. Mais l'attribut d'Encadrer-MTU reçu dans l'Access-demande n'est pas utilisé (les mêmes que sur ISE/ACS).

L'établissement de cette valeur est un contournement valide afin de réparer des questions dans la topologie comme ceci :

Suppliant [MTU 1500] ---- ---- [MTU 9000]Switch [MTU 9000] ----- ----- [MTU 9000]NPS

Actuellement les Commutateurs ne te permettent pas pour placer le MTU par port ; pour 6880 Commutateurs, cette caractéristique est ajoutée avec l'ID de bogue Cisco CSCuo26327 - EAP-TLS de 802.1x ne fonctionnant pas sur des ports de hôte FEX.

AnyConnect

AnyConnect envoie les fragments d'EAP-TLS (habituellement certificat client) qui sont de 1,486 octets de long. Pour cette taille de valeur, la trame Ethernet est de 1,500 octets. Le dernier fragment est habituellement plus petit.

Suppliant d'indigène de Microsoft Windows

Microsoft Windows envoie les fragments d'EAP-TLS (habituellement certificat client) qui sont de 1,486 ou 1,482 octets de long. Pour cette taille de valeur, la trame Ethernet est de 1,500 octets. Le dernier fragment est habituellement plus petit.

Informations connexes

- Configurer l'authentification basée sur port de 802.1x d'IEEE
- Support et documentation techniques - Cisco Systems