EAP分段實施和行為

目錄

簡介

本文說明如何理解可擴展身份驗證協定(EAP)會話並對其進行故障排除。

背景資訊

本文檔的各個部分專門介紹以下領域的覆蓋範圍：

- 身份驗證、授權和記帳(AAA)伺服器返回可擴展身份驗證協定 — 傳輸層安全(EAP-TLS)會話的伺服器證書時的行為
- 請求方返回EAP-TLS會話的客戶端證書時的行為
- 同時使用Microsoft Windows Native Supplicant客戶端和Cisco AnyConnect Network Access Manager(NAM)時的互操作性
- IP、RADIUS和EAP-TLS中的分段和網路接入裝置執行的重組過程
- RADIUS已框架最大傳輸單元(MTU)屬性

- AAA伺服器執行EAP-TLS資料包分段時的行為

必要條件

需求

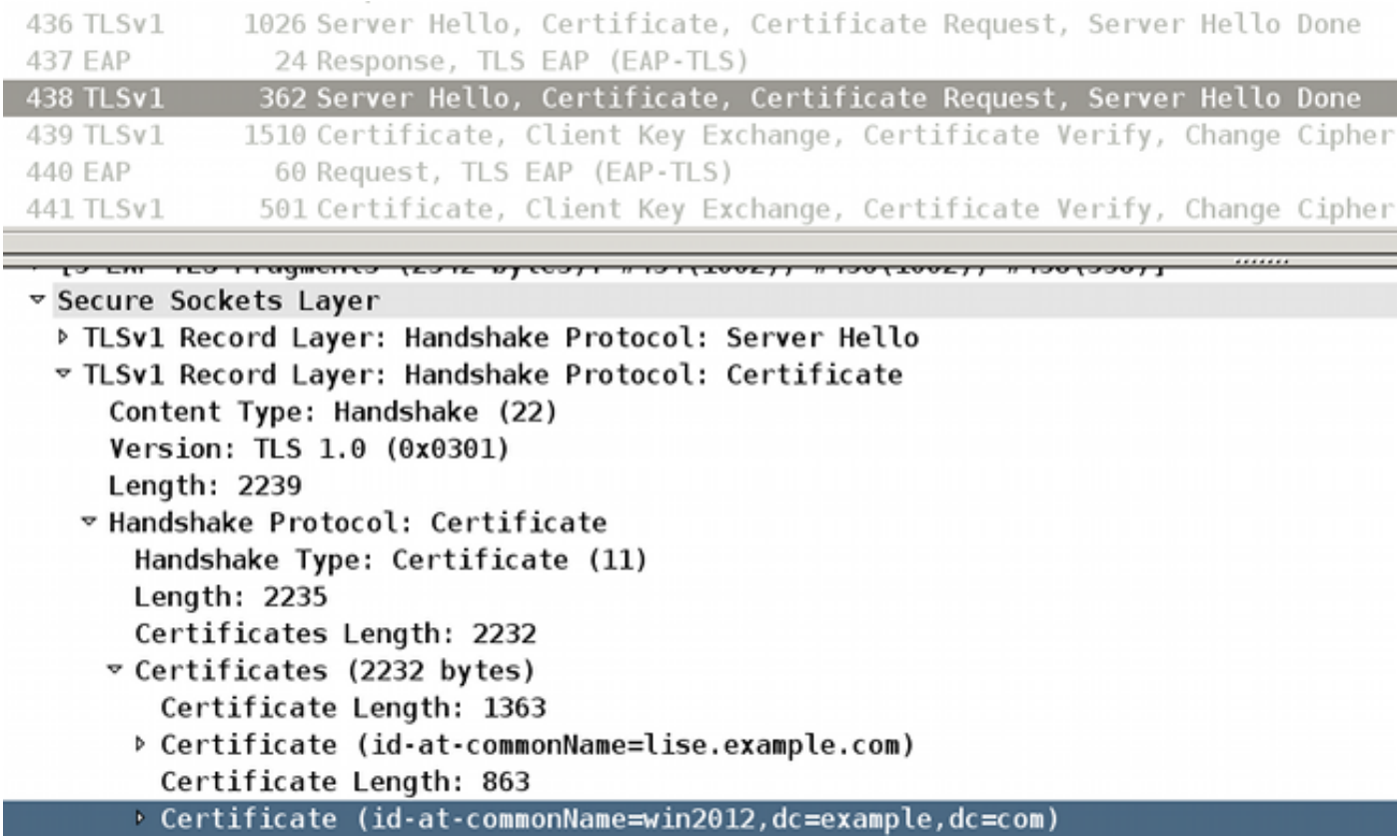
思科建議您瞭解以下主題：

- EAP和EAP-TLS協定
- 思科身份服務引擎(ISE)的配置
- Cisco Catalyst交換機的CLI配置

要理解本文的內容，必須對EAP和EAP-TLS有很好的理解。

伺服器返回的證書鏈

AAA伺服器(訪問控制伺服器(ACS)和ISE)始終返回帶有伺服器Hello和伺服器證書的EAP-TLS資料包的完整鏈：



ISE身份證書(通用名稱(CN)=lise.example.com)與簽署CN=win2012,dc=example，dc=com的證書頒發機構(CA)一起返回。ACS和ISE的行為相同。

請求方返回的證書鏈

Microsoft Windows本機請求方

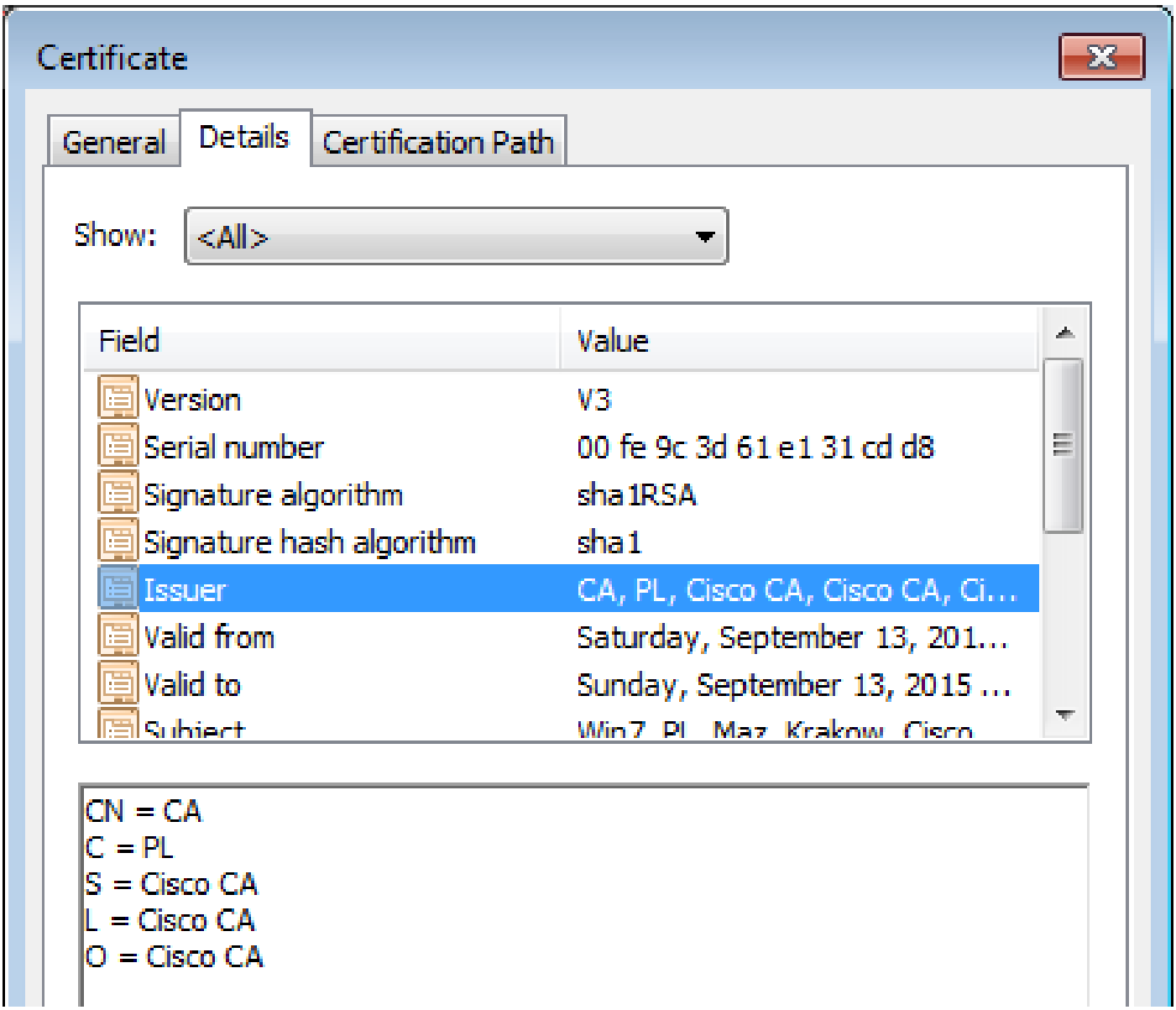
為使用EAP-TLS而配置的Microsoft Windows 7本機請求方（無論是否選擇「簡單證書選擇」）不會傳送客戶端證書的完整鏈結。

即使客戶端證書是由與伺服器證書不同的CA（不同鏈）簽名時，也會發生此行為。

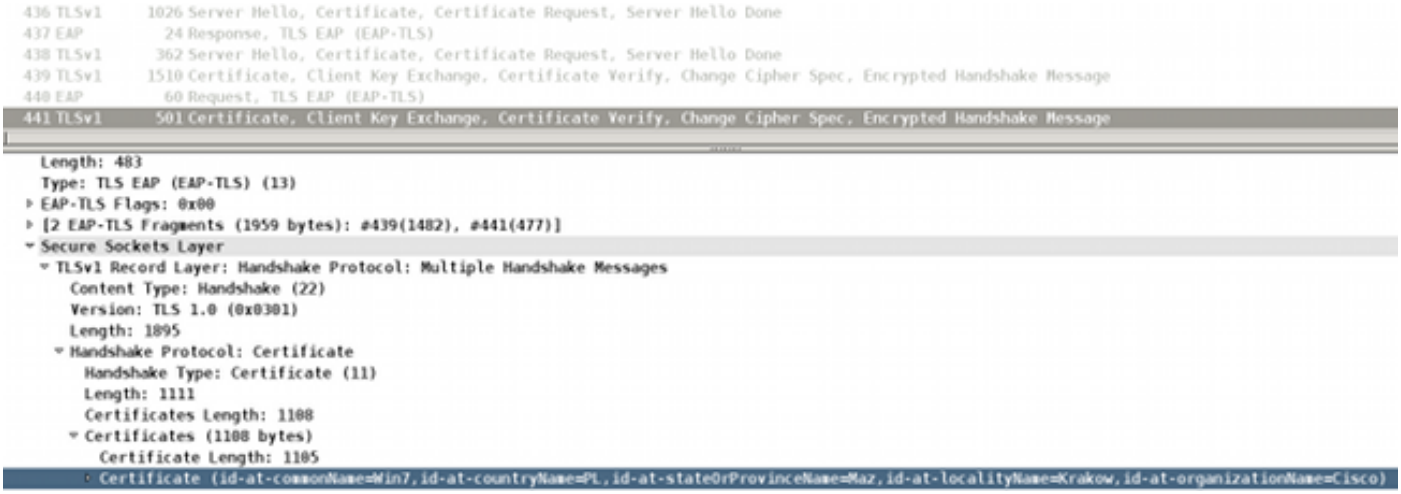
此示例與上一螢幕截圖中所示的Server Hello和Certificate相關。

在該場景中，ISE證書由CA使用主題名稱CN=win2012,dc=example，dc=com進行簽名。

但是，安裝在Microsoft應用商店中的使用者證書是由不同的CA簽名的，CN=CA，C=PL，S=Cisco CA，L=Cisco CA，O=Cisco CA。



因此，Microsoft Windows請求方僅使用客戶端證書進行響應。簽署它的CA(CN=CA，S=PL，S=Cisco CA，L=Cisco CA，O=Cisco CA)未連線。



由於此行為，AAA伺服器在驗證客戶端證書時可能會遇到問題。本示例與Microsoft Windows 7 SP1 Professional有關。

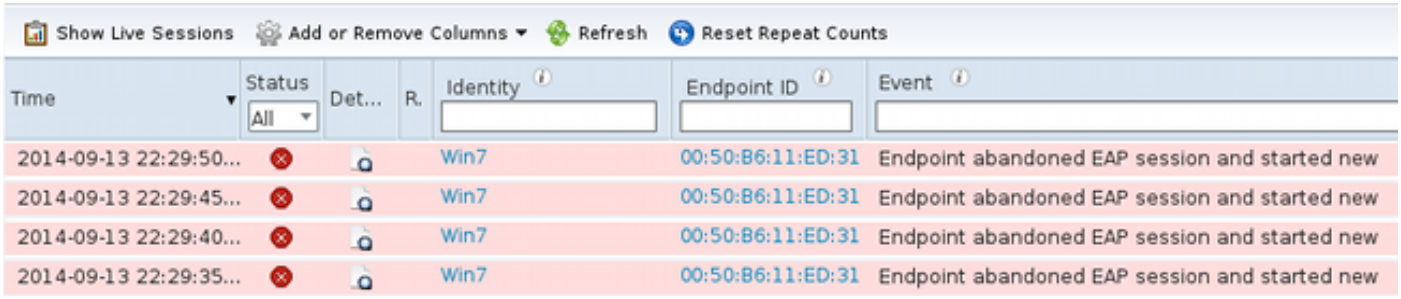
解決方案

完整的證書鏈將安裝在ACS和ISE的證書儲存上（所有CA和子CA簽名客戶端證書）。

在ACS或ISE上可以輕鬆檢測到證書驗證問題。將顯示有關不受信任證書的資訊和ISE報告：

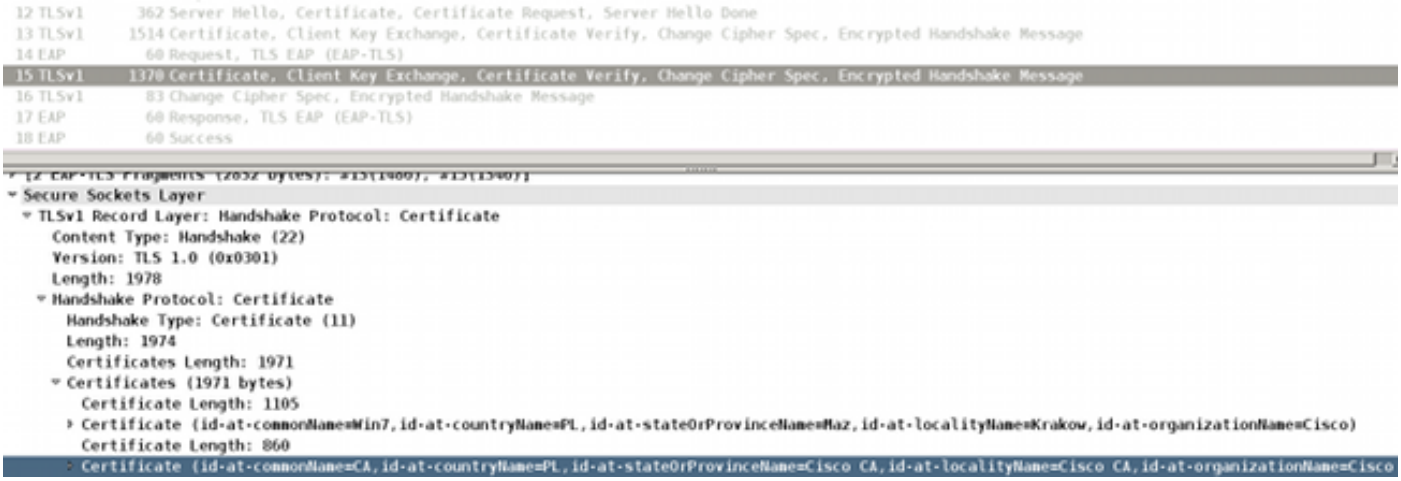
```
12514 EAP-TLS failed SSL/TLS handshake because of an unknown CA in the client
certificates chain
```

很難檢測到請求方上的證書驗證問題。通常AAA伺服器會響應「終端已放棄EAP會話」：



AnyConnect NAM

AnyConnect NAM沒有此限制。在相同的情況中，它會附加完整的客戶端證書鏈（已附加正確的CA）：



Microsoft Windows Native Supplicant客戶端與AnyConnect NAM

當兩種服務都啟動時，AnyConnect NAM優先。

即使NAM服務未運行，它仍然掛接在Microsoft Windows API上並轉發EAP資料包，這可能會導致Microsoft Windows本機請求方出現問題。

以下是此類故障的示例。

使用以下命令在Microsoft Windows上啟用跟蹤：

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

跟蹤(c:\windows\trace\svchost_RASTLS.LOG)顯示：

<#root>

```
[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

最後一個資料包是Microsoft Windows本機請求方傳送的客戶端證書（EAP-TLS片段1,EAP大小為1492）。很遺憾，Wireshark沒有顯示該資料包：

	Protocol	Length	Info
8	EAP	48	Response, Identity
9	EAP	60	Request, TLS EAP (EAP-TLS)
10	SSL	123	Client Hello
11	TLSv1	1030	Server Hello, Certificate, Certificate Request, Server Hello Done
12	EAP	24	Response, TLS EAP (EAP-TLS)
13	TLSv1	1026	Server Hello, Certificate, Certificate Request, Server Hello Done
14	EAP	24	Response, TLS EAP (EAP-TLS)
15	TLSv1	362	Server Hello, Certificate, Certificate Request, Server Hello Done
20	TLSv1	362	Ignored Unknown Record
28	TLSv1	362	Ignored Unknown Record

而且該資料包並未真正傳送；最後一個是EAP-TLS攜帶伺服器證書的第三個片段。

它已被掛接在Microsoft Windows API上的AnyConnect NAM模組佔用。

這就是不建議將AnyConnect與Microsoft Windows本機請求方一起使用的原因。

當您使用任何AnyConnect服務時，建議同時使用NAM（當需要802.1x服務時），而不是Microsoft Windows本機請求方。

分段

分段可能發生在多個層上：

- IP
- RADIUS屬性值對(AVP)
- EAP-TLS

Cisco IOS®交換器非常智慧。他們可以理解EAP和EAP-TLS格式。

雖然交換器無法解密TLS通道，但若封裝在LAN可擴充驗證通訊協定(EAPoL)或RADIUS中，則交換器負責分段、組裝和重組EAP封包。

EAP協定不支援分段。以下是RFC 3748(EAP)的摘錄：

"EAP本身不支援分段；但是，單個EAP方法可能支援此功能。"

EAP-TLS就是這樣的例子。以下是RFC 5216(EAP-TLS)第2.1.5節（分段）的摘錄：

「當EAP-TLS對等體收到設定了M位的EAP-Request資料包時，它必須使用EAP-Type=EAP-TLS且無資料的EAP-Response進行響應。

這用作分段ACK。EAP伺服器必須等待，直到收到EAP-Response後再傳送另一個片段。」

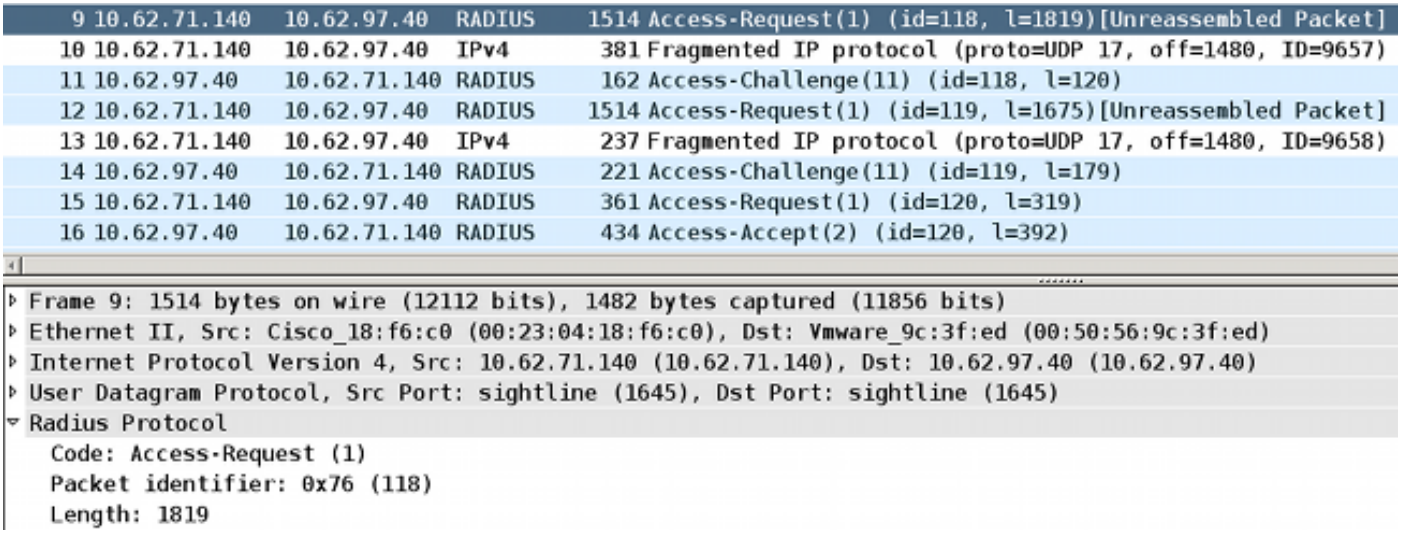
最後一句描述了AAA伺服器的一個非常重要的功能。他們必須等待ACK，然後才能傳送另一個EAP片段。請求方使用類似的規則：

"EAP對等體必須等待，直到它收到EAP-Request後再傳送另一個片段。"

IP層中的分段

分段只能在網路接入裝置(NAD)和AAA伺服器（用作傳輸的IP/UDP/RADIUS）之間進行。

當NAD（Cisco IOS交換器）嘗試傳送包含EAP負載（大於介面的MTU）的RADIUS要求時，會發生這種情況：



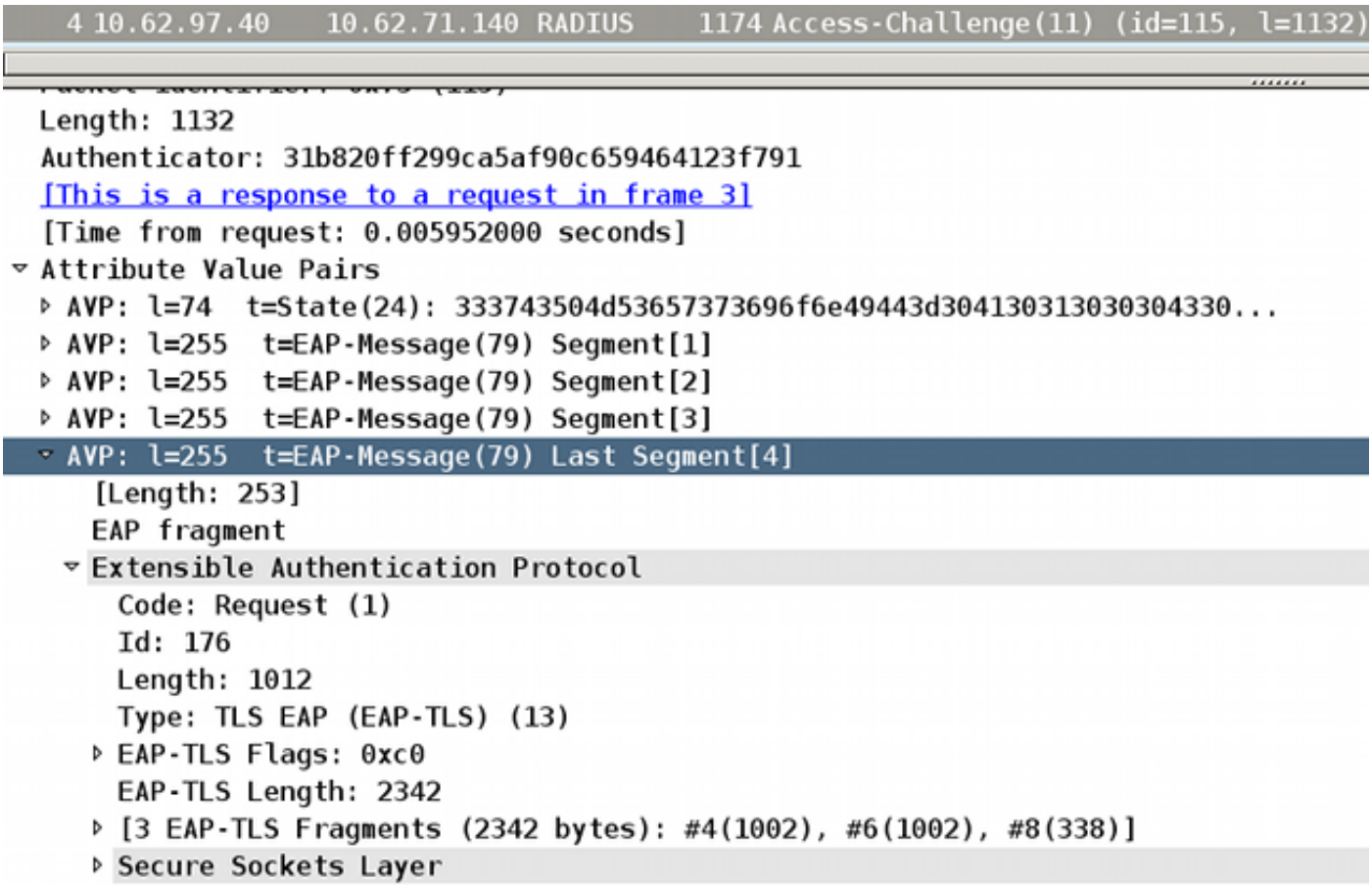
大多數Cisco IOS版本不夠智慧，不會嘗試組合通過EAPoL接收的EAP資料包，並將它們合併到可以容納通向AAA伺服器的物理介面的MTU的RADIUS資料包中。

AAA伺服器更加智慧（如下一節所示）。

RADIUS中的分段

這實際上不是任何形式的分裂。根據RFC 2865，單個RADIUS屬性最多可以有253位元組的資料。

因此，EAP負載始終以多個EAP消息RADIUS屬性傳輸：



這些EAP-Message屬性由Wireshark重新組合併解釋（「最後段」屬性顯示整個EAP資料包的負載）。

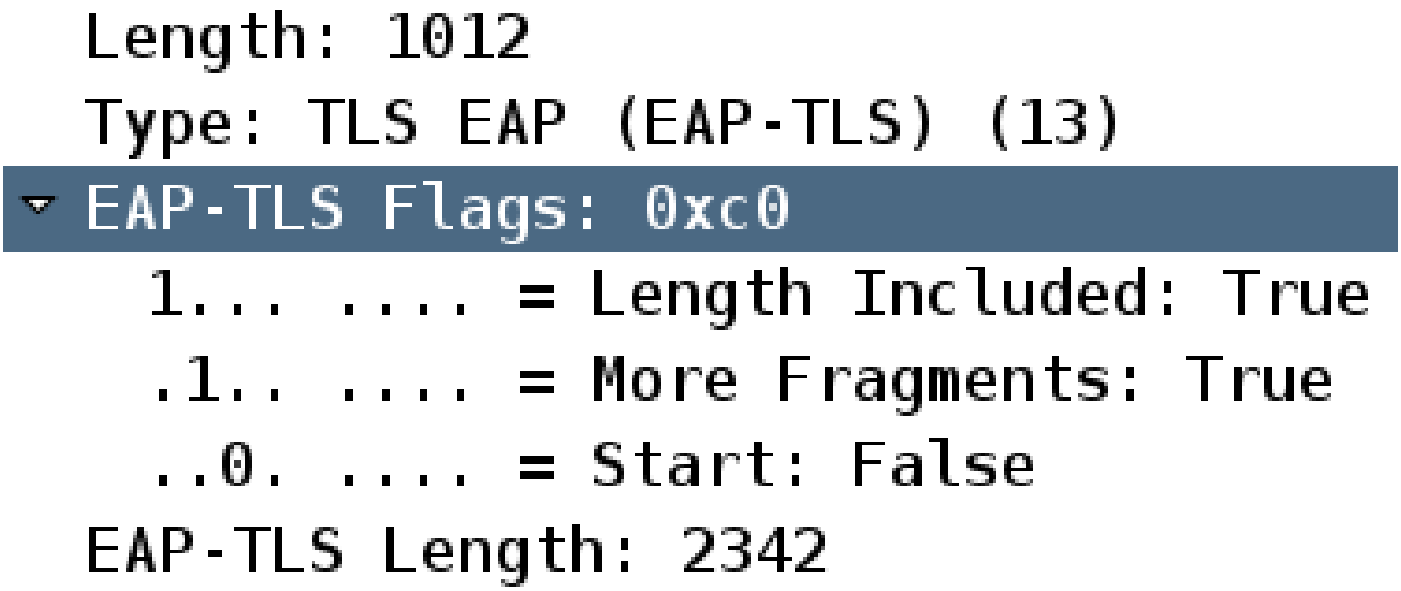
EAP資料包中的Length報頭等於1,012，傳輸它需要四個RADIUS AVP。

EAP-TLS中的分段

從同一個螢幕截圖中，您可以看到：

- EAP資料包長度為1,012
- EAP-TLS長度為2,342

這表示它是第一個EAP-TLS片段，請求方期望更多，如果您檢查EAP-TLS標誌，可以確認這一點：



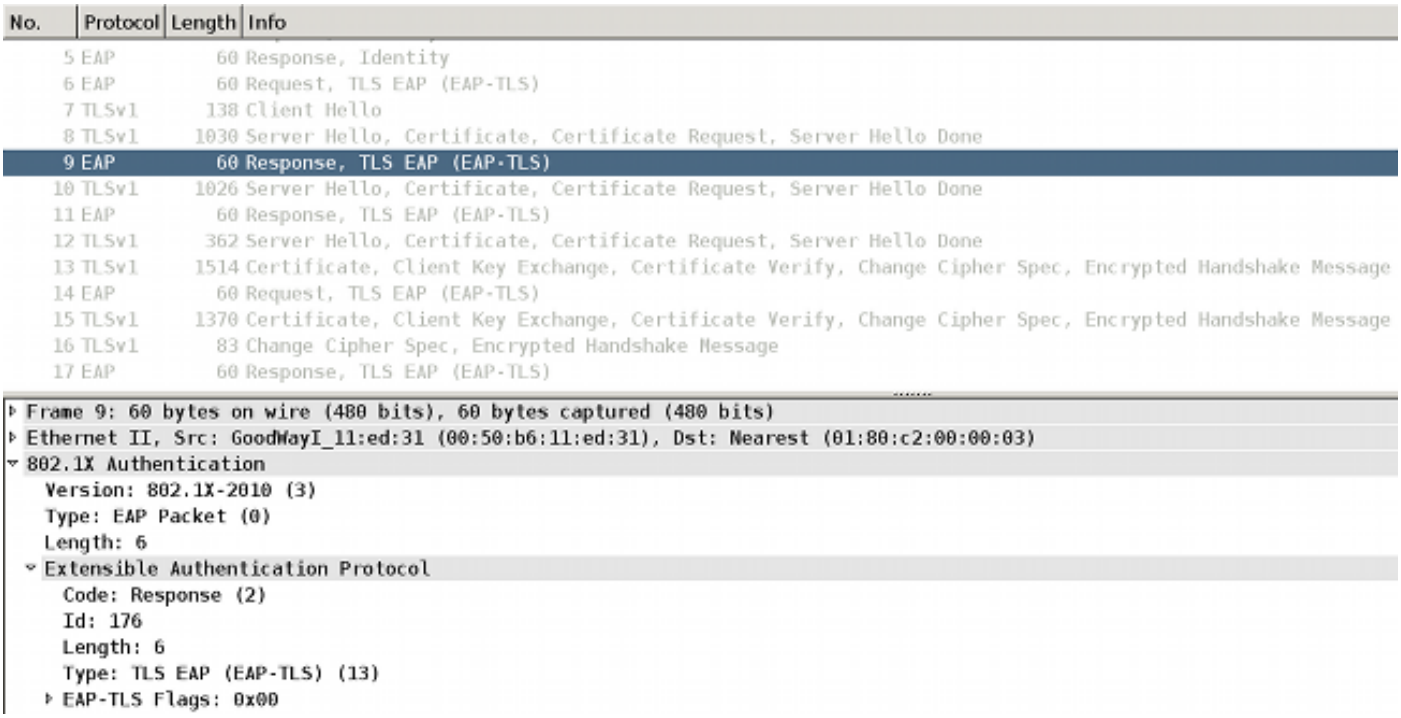
此類分段最常見於：

- AAA伺服器傳送的RADIUS Access-Challenge，該伺服器攜帶帶有整個鏈的安全套接字層(SSL)伺服器證書的EAP請求。
- NAD傳送的RADIUS Access-Request ，它攜帶帶有整個鏈的SSL客戶端證書的EAP-Response。

EAP-TLS片段確認

如前所述，必須在傳送後續片段之前確認每個EAP-TLS片段。

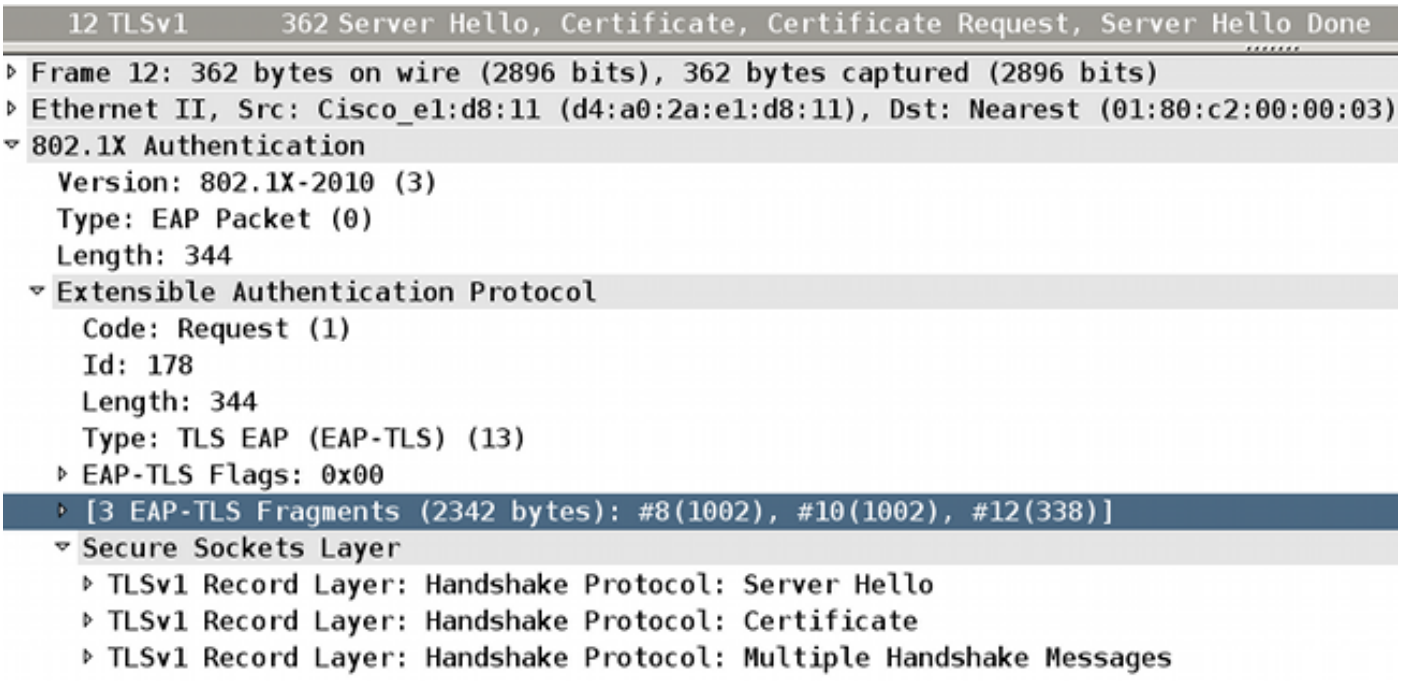
以下是範例（請求方和NAD之間EAPoL的封包擷取）：



EAPoL幀和AAA伺服器返回伺服器證書：

- 該證書在EAP-TLS片段（資料包8）中傳送。
- 請求方確認該分段（封包9）。
- 第二個EAP-TLS片段由NAD（資料包10）轉發。
- 請求方確認該分段（封包11）。
- 第三個EAP-TLS片段由NAD（資料包12）轉發。
- 請求方不需要確認此情況；相反，它會使用從資料包13開始的客戶端證書繼續操作。

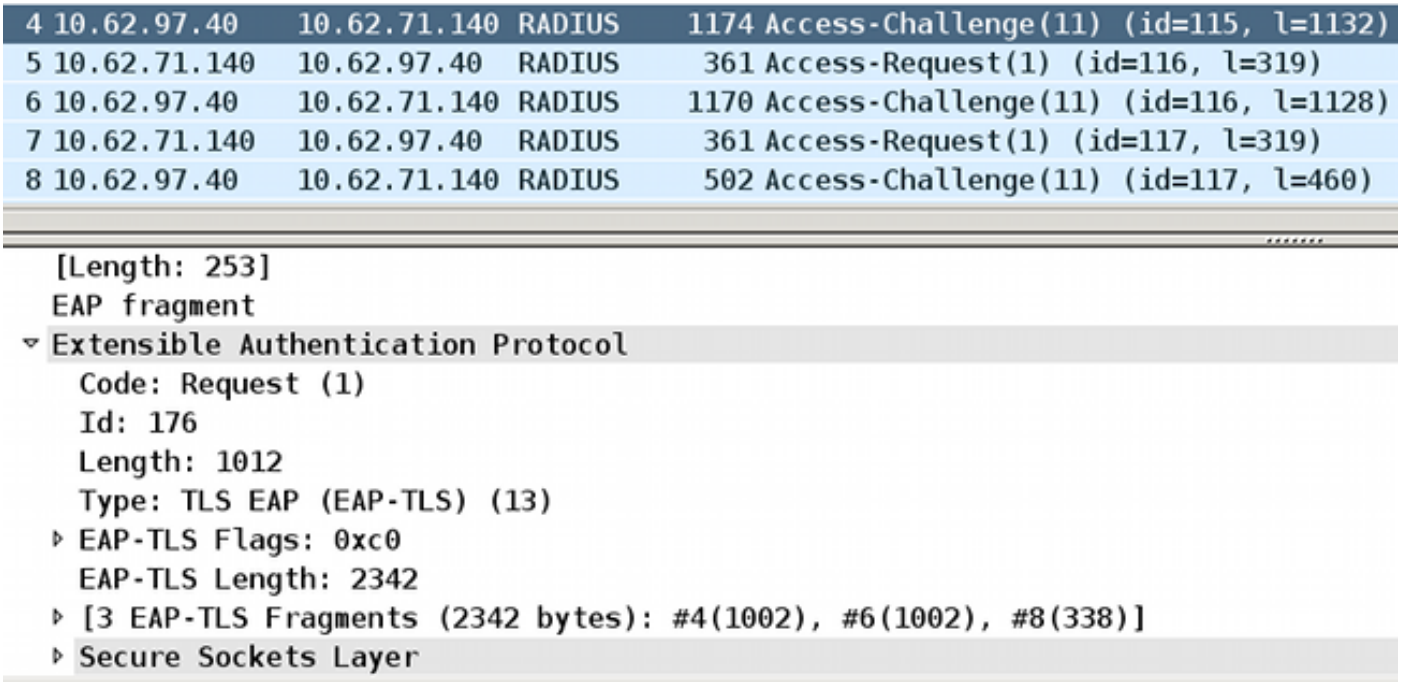
以下是封包12的詳細資訊：



您可以看到Wireshark重組了資料包8、10和12。

EAP片段大小為1,002、1,002和338，使得EAP-TLS消息的總大小為2342;

在每個片段中通告總EAP-TLS消息長度。如果檢查RADIUS封包（在NAD和AAA伺服器之間），可以確認這點：



RADIUS封包4、6和8承載這三個EAP-TLS片段。前兩個片段已確認。

Wireshark能夠顯示有關EAP-TLS片段的資訊（大小：1,002 + 1,002 + 338 = 2,342）。

這個場景和例子很簡單。Cisco IOS交換機不需要更改EAP-TLS片段大小。

使用不同大小重新組裝EAP-TLS片段

試想一下，當AAA伺服器的NAD MTU為9,000位元組（巨型幀），並且AAA伺服器還連線到支援巨型幀的介面時，會發生什麼情況。

大多數典型請求方都使用1Gbit鏈路連線，MTU為1,500。

在這種情況下，Cisco IOS交換機執行EAP-TLS「assymetric」彙編和重組，並更改EAP-TLS片段大小。

以下是由AAA伺服器（SSL伺服器憑證）傳送的大型EAP訊息的範例：

1. AAA伺服器必須傳送帶有SSL伺服器證書的EAP-TLS消息。該EAP資料包的總大小為3,000。封裝在RADIUS Access-Challenge/UDP/IP中後，其仍然小於AAA伺服器介面MTU。傳送帶有12個RADIUS EAP-Message屬性的單個IP資料包。沒有IP或EAP-TLS分段。

2. Cisco IOS交換機收到此類資料包，將其解除封裝，並決定需要通過EAPoL將EAP傳送到請求方。由於EAPoL不支援分段，因此交換機必須執行EAP-TLS分段。

3. Cisco IOS交換機準備第一個可以容納到通向請求方(1,500)的介面MTU中的EAP-TLS片段。

4. 此片段由請求方確認。

5. 收到確認消息後，將傳送另一個EAP-TLS片段。

6. 此片段由請求方確認。

7. 最後一個EAP-TLS片段由交換機傳送。

此案例顯示：

- 在某些情況下，NAD必須建立EAP-TLS片段。
- NAD負責傳送/確認這些片段。

對於通過支援巨型幀的鏈路連線的請求方，當AAA伺服器的MTU較小時（然後Cisco IOS交換機在向AAA伺服器傳送EAP資料包時建立EAP-TLS片段），也會出現相同的情況。

RADIUS屬性已框架化 — MTU

對於RADIUS，在RFC 2865中定義有一個已框架的MTU屬性：

「此屬性指示當通過其他方式（如PPP）未協商使用者配置的最大傳輸單元。它可用於訪問接受資料包。

NAS可在訪問請求資料包中將其用作向伺服器發出它希望使用該值的提示，但並不要求伺服器執行此提示。」

ISE不執行提示。NAD在Access-Request中傳送的Framed-MTU值對ISE執行的分段沒有任何影響。

除了交換機上全域性啟用的巨型幀設定外，多台現代Cisco IOS交換機不允許更改乙太網介面的MTU。巨型訊框的組態會影響RADIUS Access-Request中傳送的Framed-MTU屬性的值。例如，您設定：

```
<#root>

Switch(config)#

system mtu jumbo 9000
```

這會強制交換器在所有RADIUS存取要求中傳送Framed-MTU = 9000。沒有巨型訊框的系統MTU也一樣：

```
<#root>

Switch(config)#

system mtu 1600
```

這會強制交換器在所有RADIUS存取要求中傳送Framed-MTU = 1600。

請注意，現代Cisco IOS交換機不允許您將系統MTU值降低到1,500以下。

傳送EAP片段時的AAA伺服器和請求方行為

ISE

ISE始終嘗試傳送1,002位元組的EAP-TLS片段（通常為Server Hello with Certificate）（儘管最後一個片段通常更小）。

它不遵守RADIUS Framed-MTU。無法將其重新配置為傳送更大的EAP-TLS片段。

Microsoft網路策略伺服器(NPS)

如果在NPS上本地配置Framed-MTU屬性，則可以配置EAP-TLS片段的大小。

事件：儘管在Microsoft NPS上配置EAP負載大小文章提到NPS RADIUS伺服器的已框架化MTU的預設值為1,500，但Cisco技術支援中心(TAC)實驗室已顯示，它使用預設設定傳送2,000個（在Microsoft Windows 2012資料中心上確認）。

經測試，NPS遵守根據前述指南在本地設定Framed-MTU，並將EAP消息分段為在Framed-MTU中設定大小的片段。但是未使用Access-Request中接收的Framed-MTU屬性（與ISE/ACS上相同）。

設定此值是有效的解決方法，可以修復拓撲中出現的問題，如下所示：

Supplicant客戶端[MTU 1500] ---- ---- [MTU 9000]Switch[MTU 9000] ----- ----- [MTU 9000]NPS

目前交換器不允許您設定每個連線埠的MTU；若是6880交換器，此功能已加入思科錯誤ID CSCuo26327 - 802.1x EAP-TLS無法在FEX主機連線埠上使用。

AnyConnect

AnyConnect傳送長度為1,486位元組的EAP-TLS片段（通常是客戶端證書）。對於此值大小，乙太網幀為1,500位元組。最後一個片段通常更小。

Microsoft Windows本機請求方

Microsoft Windows傳送1,486或1,482位元組的EAP-TLS片段（通常是客戶端證書）。對於此值大小，乙太網幀為1,500位元組。最後一個片段通常更小。

相關資訊

- 配置IEEE 802.1x基於埠的身份驗證
- 技術支援與文件 - Cisco Systems

關於此翻譯

思科已使用電腦和人工技術翻譯本文件，讓全世界的使用者能夠以自己的語言理解支援內容。請注意，即使是最佳機器翻譯，也不如專業譯者翻譯的內容準確。Cisco Systems, Inc. 對這些翻譯的準確度概不負責，並建議一律查看原始英文文件（提供連結）。