EAP分段实施和行为

目录

简介

本文档介绍如何了解可扩展身份验证协议(EAP)会话并对其进行故障排除。

背景信息

本文档的各部分专门介绍以下方面的内容：

- 身份验证、授权和记帐(AAA)服务器返回可扩展身份验证协议 — 传输层安全(EAP-TLS)会话的服务器证书时的行为
- 请求方返回EAP-TLS会话的客户端证书时的行为
- 同时使用Microsoft Windows本地请求方和Cisco AnyConnect网络访问管理器(NAM)时的互操作性
- 网络接入设备执行的IP、RADIUS和EAP-TLS分段和重组流程
- RADIUS Framed-Maximum Transmission Unit(MTU)属性

- AAA服务器执行EAP-TLS数据包分段时的行为

先决条件

要求

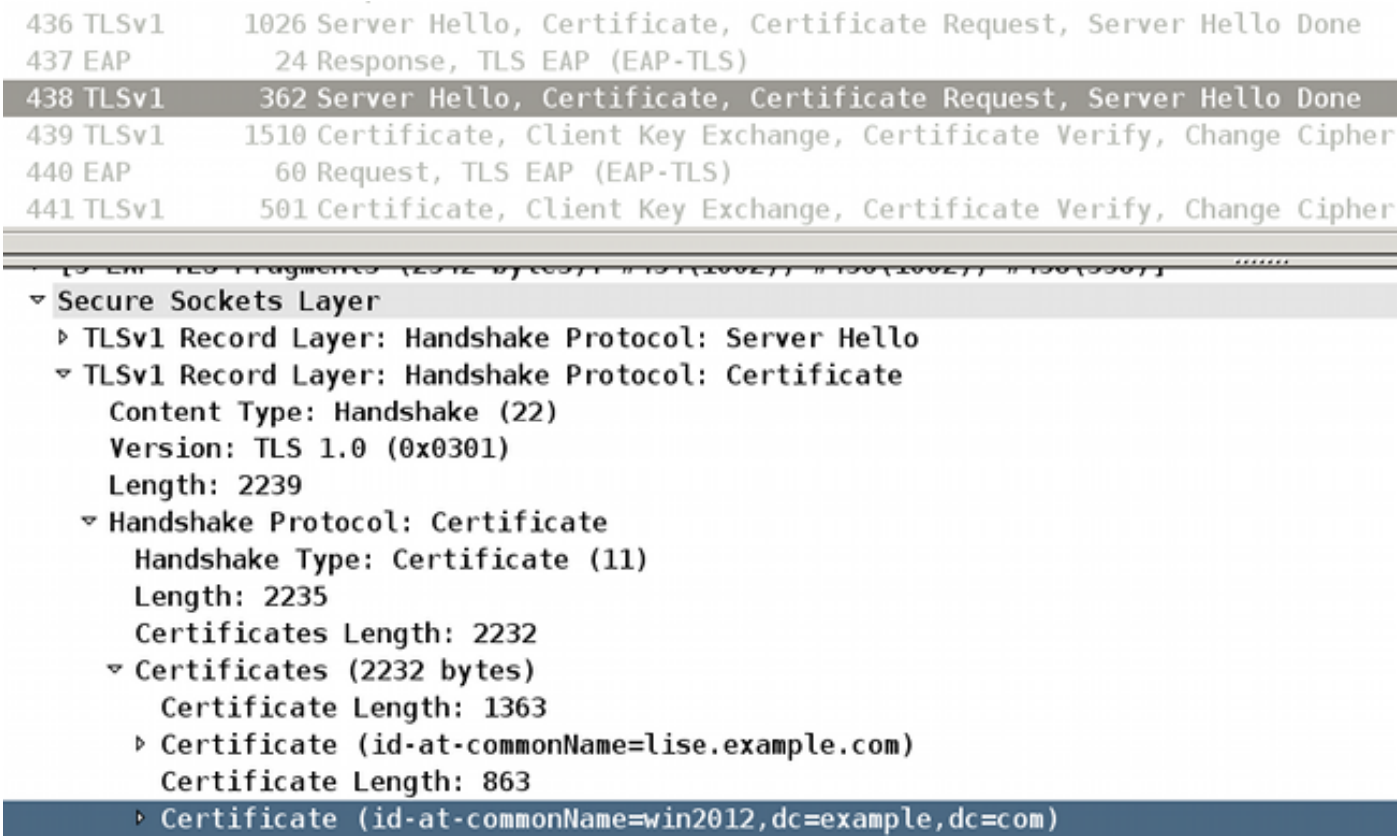
Cisco 建议您了解以下主题：

- EAP和EAP-TLS协议
- 思科身份服务引擎(ISE)的配置
- Cisco Catalyst交换机的CLI配置

要理解本文，必须充分了解EAP和EAP-TLS。

服务器返回的证书链

AAA服务器(访问控制服务器(ACS)和ISE)始终返回包含服务器Hello和服务器证书的EAP-TLS数据包的完整链：



ISE身份证书(公用名称(CN)=lise.example.com)与签署CN=win2012,dc=example，dc=com的证书颁发机构(CA)一起返回。ACS和ISE的行为相同。

请求方返回的证书链

Microsoft Windows本地请求方

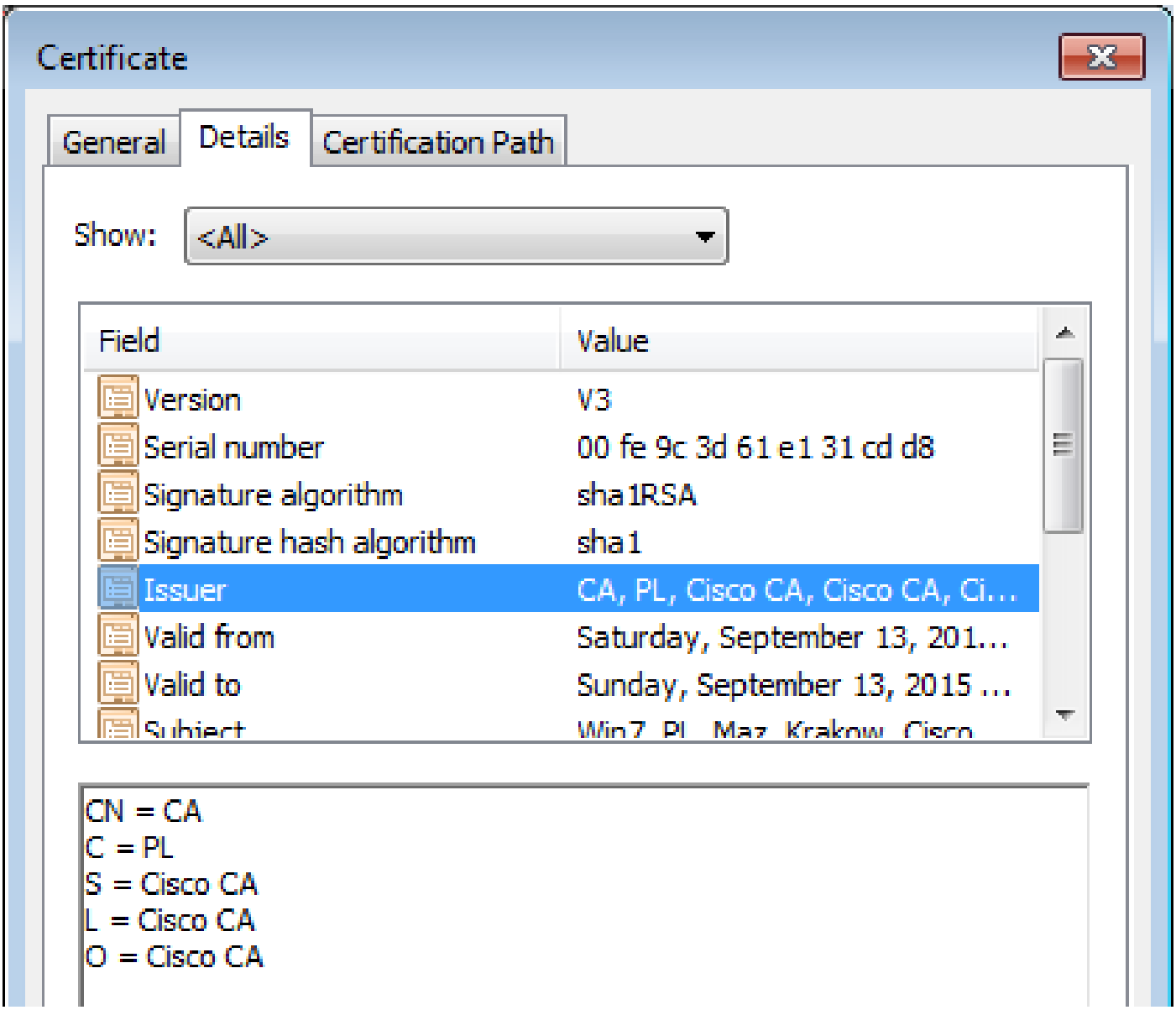
为使用EAP-TLS而配置的Microsoft Windows 7本地请求方（无论是否选择“简单证书选择”）不会发送客户端证书的完整链。

即使客户端证书是由与服务器证书不同的CA（不同链）签名时也会出现此行为。

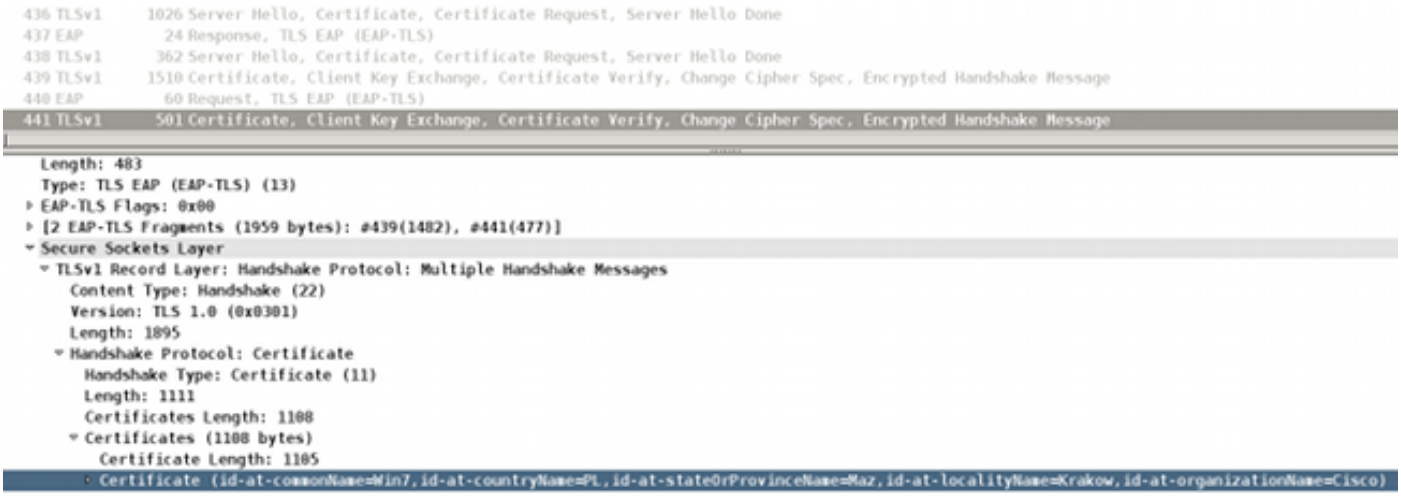
此示例与上一屏幕截图中所示的服务器Hello和证书有关。

对于此场景，ISE证书由CA使用使用者名称签名，CN=win2012,dc=example，dc=com。

但是，安装在Microsoft应用商店中的用户证书由不同的CA签名，CN=CA，C=PL，S=Cisco CA，L=Cisco CA，O=Cisco CA。



因此，Microsoft Windows请求方仅使用客户端证书进行响应。未附加签署它的CA(CN=CA，S=PL，S=Cisco CA，L=Cisco CA，O=Cisco CA)。



由于此行为，AAA服务器在验证客户端证书时可能会遇到问题。本示例与Microsoft Windows 7 SP1 Professional有关。

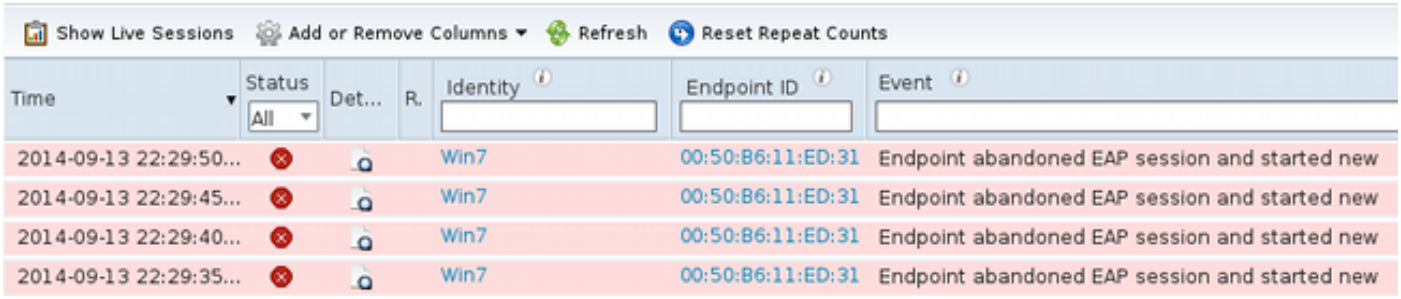
解决方案

完整的证书链将安装在ACS和ISE的证书库上（所有CA和子CA签名客户端证书）。

在ACS或ISE上可以轻松检测到证书验证问题。将显示有关不受信任证书的信息和ISE报告：

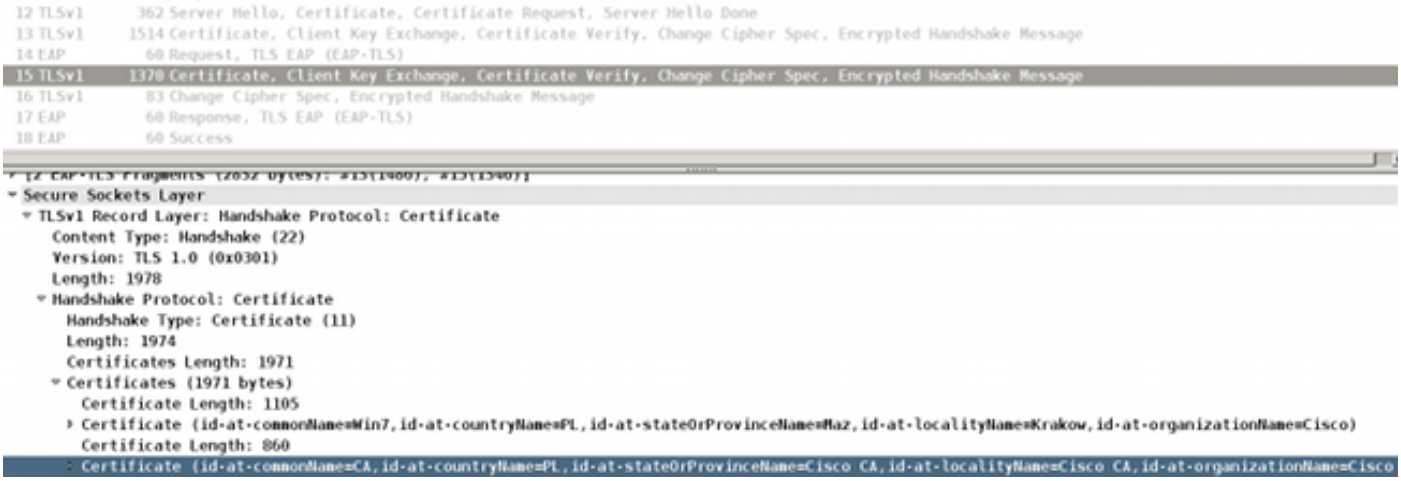
```
12514 EAP-TLS failed SSL/TLS handshake because of an unknown CA in the client
certificates chain
```

很难检测到请求方上的证书验证问题。通常，AAA服务器会响应“终端已放弃EAP会话”：



AnyConnect NAM

AnyConnect NAM没有此限制。在同一个场景中，它会附加完整的客户端证书链（已附加正确的CA）：



Microsoft Windows原生Supplicant客户端和AnyConnect NAM

当两种服务都启用时，AnyConnect NAM优先。

即使NAM服务未运行，它仍然挂接在Microsoft Windows API上并转发EAP数据包，这会导致Microsoft Windows本机请求方出现问题。

下面是此类故障的一个示例。

使用以下命令可在Microsoft Windows上启用跟踪：

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

跟踪(c:\windows\trace\svchost_RASTLS.LOG)显示：

<#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
```

最后一个数据包是Microsoft Windows本机请求方发送的客户端证书（EAP-TLS片段1,EAP大小为1492）。遗憾的是，Wireshark不显示该数据包：

	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

而且该数据包并未真正发送；最后一个数据包是携带服务器证书的EAP-TLS的第三个分段。

它已被挂接在Microsoft Windows API上的AnyConnect NAM模块占用。

因此，不建议将AnyConnect与Microsoft Windows本地请求方配合使用。

当您使用任何AnyConnect服务时，建议同时使用NAM（当需要802.1x服务时），而不是Microsoft Windows本地请求方。

分段

分段可能发生在多个层上：

- IP
- RADIUS属性值对(AVP)
- EAP-TLS

Cisco IOS®交换机非常智能。他们可以理解EAP和EAP-TLS格式。

虽然交换机无法解密TLS隧道，但是当封装在LAN上的可扩展身份验证协议(EAPoL)或RADIUS中时，它负责对EAP数据包进行分段、组装和重组。

EAP协议不支持分段。以下是RFC 3748(EAP)的摘录：

"EAP本身不支持分段；但是，单个EAP方法可能支持此操作。"

EAP-TLS就是这样的例子。以下是RFC 5216(EAP-TLS)第2.1.5节（分段）的摘录：

“当EAP-TLS对等体收到设置了M位的EAP-Request数据包时，它必须以EAP-Type=EAP-TLS且无数据的EAP-Response进行响应。

这用作分段ACK。EAP服务器必须等到收到EAP-Response后再发送另一个片段。"

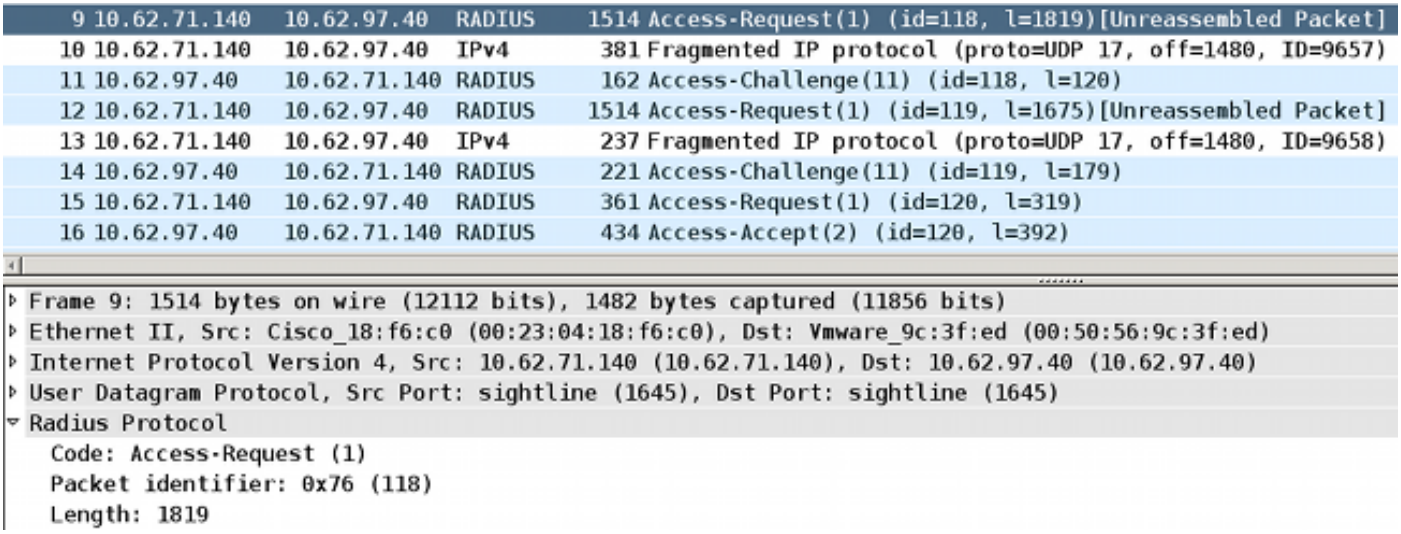
最后一句描述了AAA服务器的一个非常重要的功能。他们必须等待ACK，然后才能发送另一个EAP分段。请求方也使用类似的规则：

"EAP对等体必须等到收到EAP请求后再发送另一个分段。"

IP层分段

分段只能在网络接入设备(NAD)和AAA服务器（用作传输的IP/UDP/RADIUS）之间发生。

当NAD（Cisco IOS交换机）尝试发送包含EAP负载的RADIUS请求时，会出现这种情况，该负载大于接口的MTU:



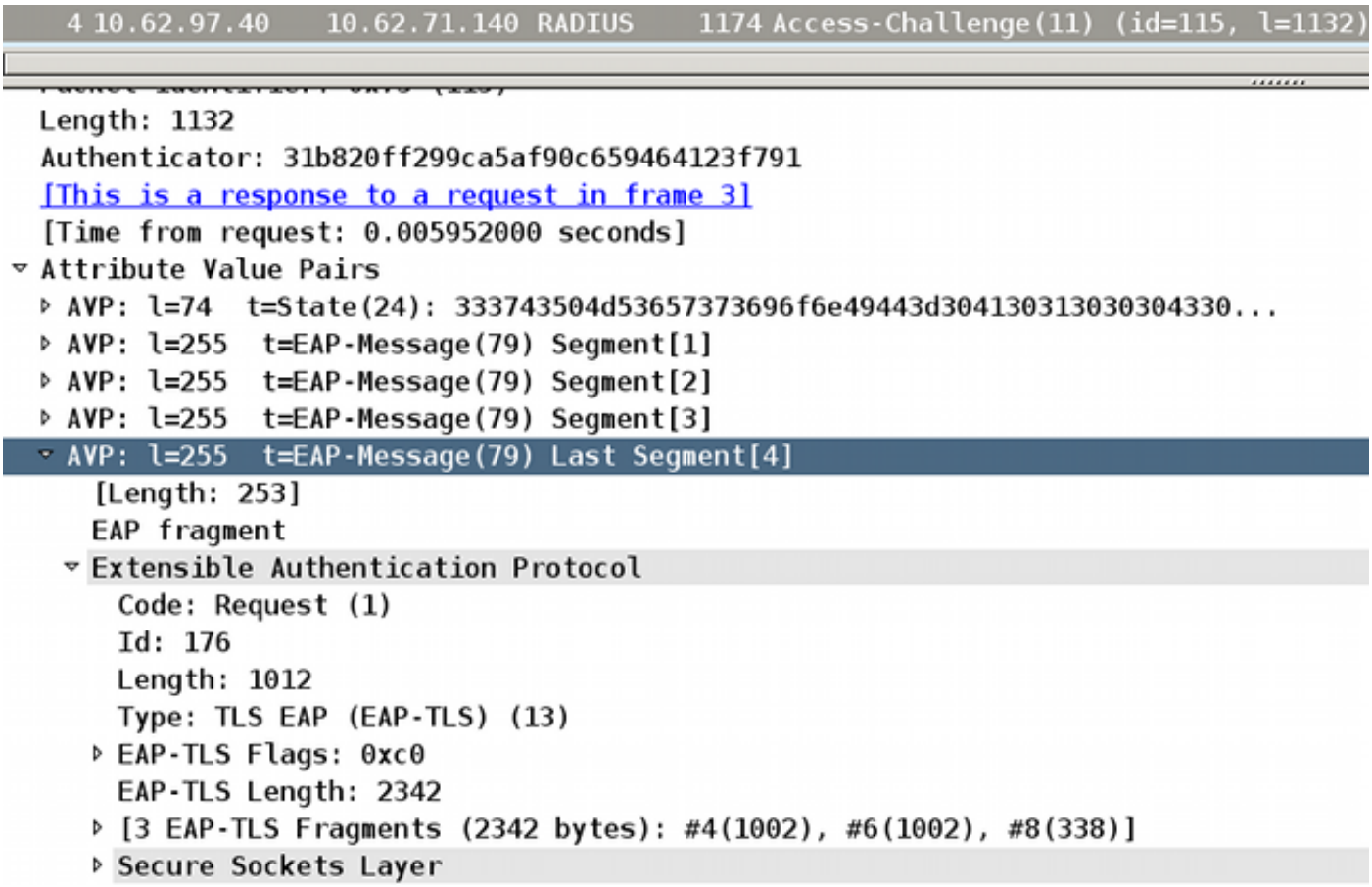
大多数Cisco IOS版本不够智能，不会尝试组合通过EAPoL接收的EAP数据包，并将其组合到可以容纳到AAA服务器的物理接口的MTU中的RADIUS数据包中。

AAA服务器更加智能（如下一节所示）。

RADIUS中的分段

这实际上不是任何形式的分裂。根据RFC 2865，单个RADIUS属性最多可以有253字节的数据。因

此，EAP负载始终以多个EAP消息RADIUS属性传输：



这些EAP-Message属性由Wireshark重新组合和解释（“最后一个数据段”属性显示整个EAP数据包的负载）。

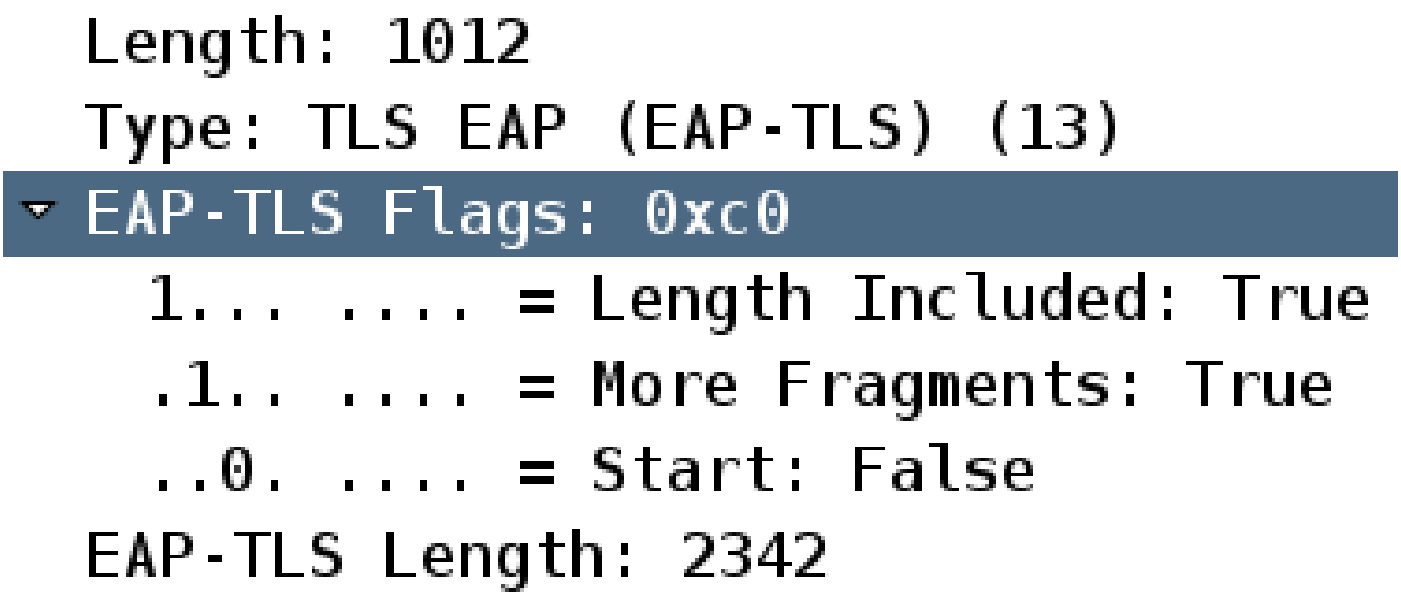
EAP数据包中的Length报头等于1,012，传输它需要四个RADIUS AVP。

EAP-TLS中的分段

从同一个屏幕截图中，您可以看到：

- EAP数据包长度为1,012
- EAP-TLS长度为2,342

这表明，它是第一个EAP-TLS分段，请求方期望更多，如果检查EAP-TLS标志，则可以确认这一点：



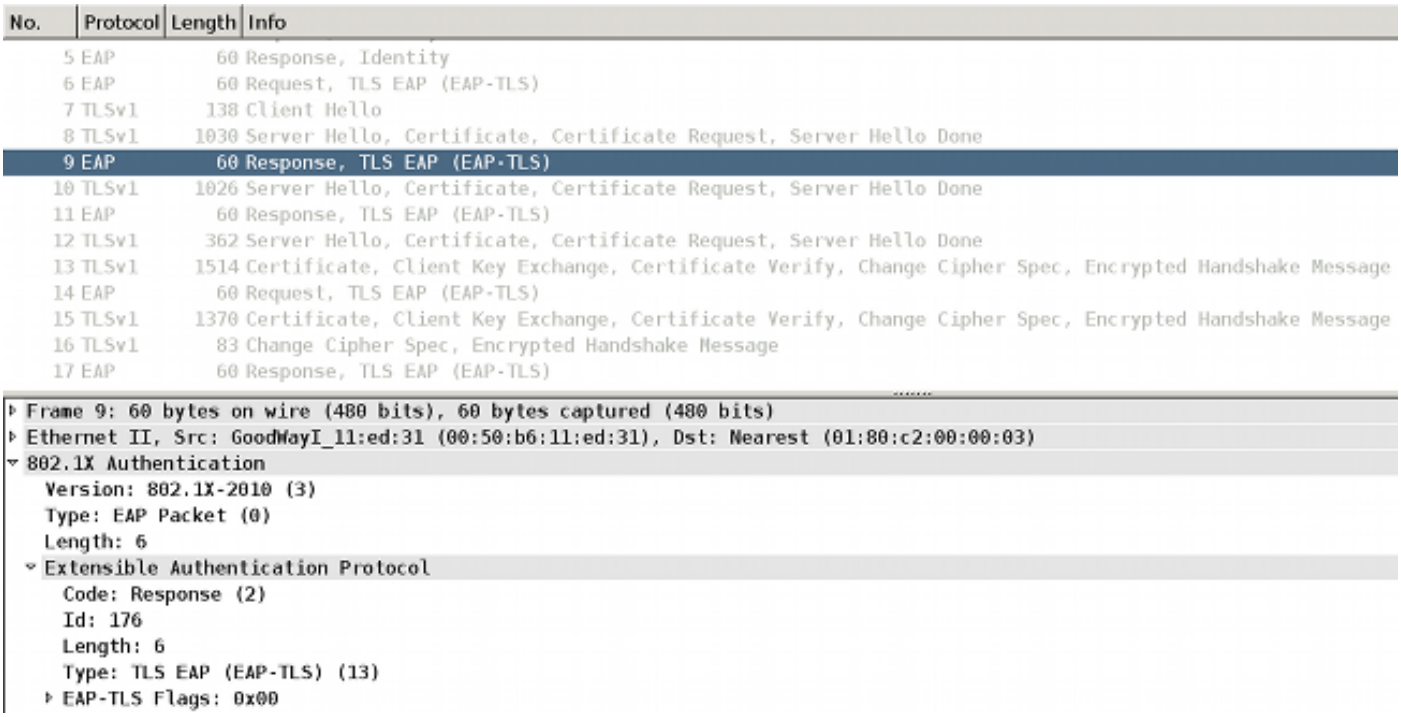
此类分段最常发生于：

- 由AAA服务器发送的RADIUS访问质询，该质询带有安全套接字层(SSL)服务器证书和整个链的EAP请求。
- 由NAD发送的RADIUS访问请求，该请求携带带有整个链的SSL客户端证书的EAP响应。

EAP-TLS分段确认

如前所述，在发送后续分段之前，必须确认每个EAP-TLS分段。

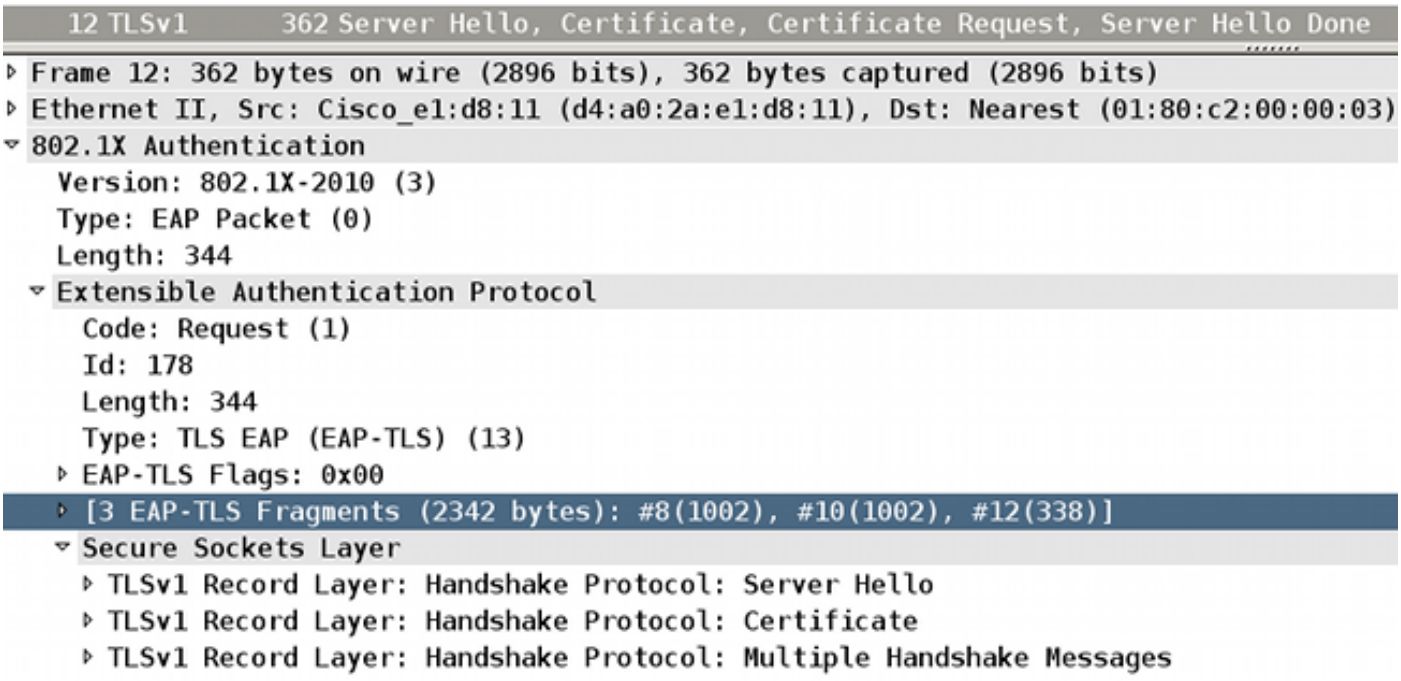
以下是一个示例（请求方和NAD之间EAPoL的数据包捕获）：



EAPoL帧和AAA服务器返回服务器证书：

- 该证书在EAP-TLS分片（数据包8）中发送。
- 请求方确认该分段（数据包9）。
- 第二个EAP-TLS分段由NAD（数据包10）转发。
- 请求方确认该分段（数据包11）。
- 第三个EAP-TLS分段由NAD（数据包12）转发。
- 请求方不需要确认这一点；相反，它继续使用从数据包13开始的客户端证书。

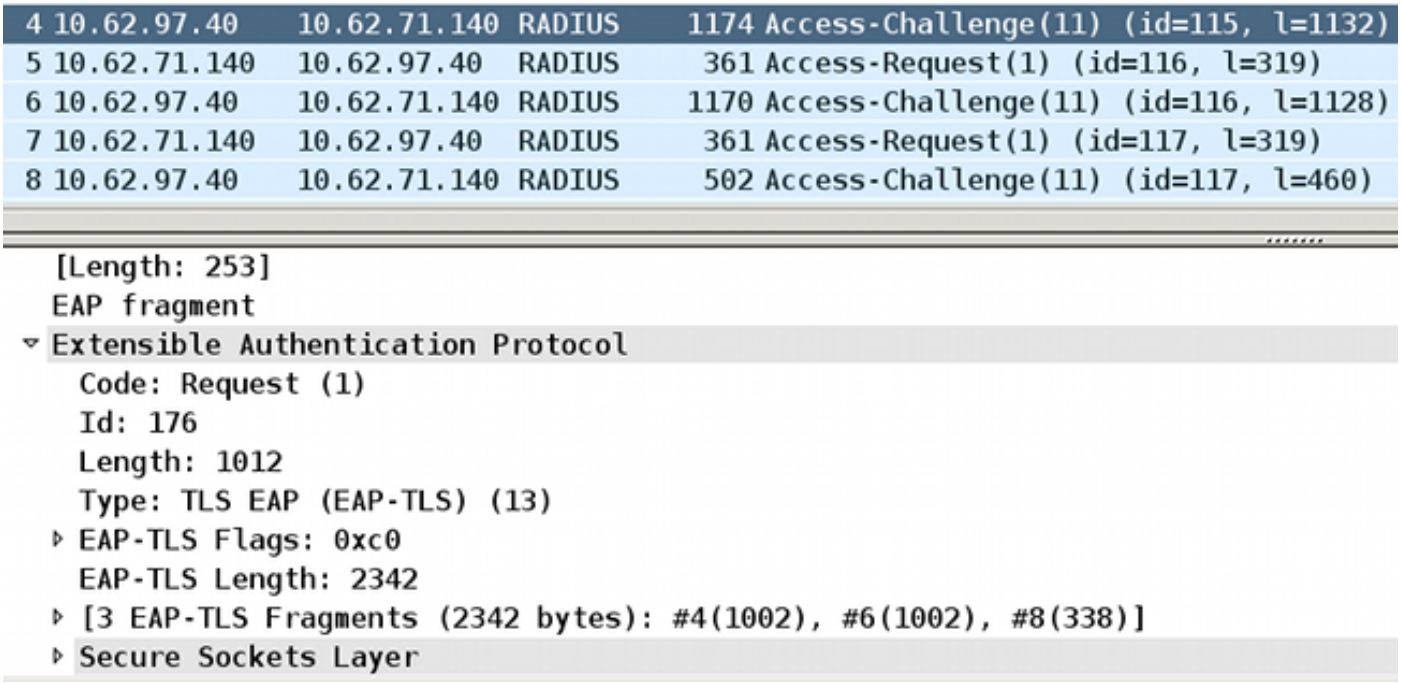
以下是数据包12的详细信息：



您可以看到Wireshark重组了数据包8、10和12。

EAP分段的大小为1,002、1,002和338，这使EAP-TLS消息的总大小为2342;

EAP-TLS消息总长度在每个分段中通告。如果检查RADIUS数据包（在NAD和AAA服务器之间），可以确认这一点：



RADIUS数据包4、6和8传输这三个EAP-TLS分段。确认前两个分段。

Wireshark可以显示有关EAP-TLS分片的信息（大小：1,002 + 1,002 + 338 = 2,342）。

这个场景和例子很简单。Cisco IOS交换机不需要更改EAP-TLS分段大小。

使用不同大小重组EAP-TLS分段

考虑当指向AAA服务器的NAD MTU为9,000字节（巨型帧）并且AAA服务器也通过支持巨型帧的接口连接时会发生什么情况。

大多数典型的Supplicant客户端都使用1Gbit链路连接，MTU为1,500。

在这种情况下，Cisco IOS交换机执行EAP-TLS“assymetric”组合和重组，并更改EAP-TLS分段大小。

以下是AAA服务器（SSL服务器证书）发送的大型EAP消息的示例：

1. AAA服务器必须发送带有SSL服务器证书的EAP-TLS消息。该EAP数据包的总大小为3,000。将其封装在RADIUS Access-Challenge/UDP/IP中后，它仍然小于AAA服务器接口MTU。发送一个具有12个RADIUS EAP-Message属性的IP数据包。没有IP或EAP-TLS分段。

2. Cisco IOS交换机收到此类数据包，将其解封，然后确定需要通过EAPoL将EAP发送给请求方。由于EAPoL不支持分段，因此交换机必须执行EAP-TLS分段。

3. Cisco IOS交换机准备第一个可以适合到通向请求方(1,500)的接口的MTU的EAP-TLS分段。

4. 此片段由请求方确认。

5. 收到确认消息后，将发送另一个EAP-TLS分段。

6. 此片段由请求方确认。

7. 最后一个EAP-TLS分段由交换机发送。

此场景显示：

- 在某些情况下，NAD必须创建EAP-TLS分段。
- NAD负责发送/确认这些分段。

对于通过支持巨型帧的链路连接的请求方，当AAA服务器具有更小的MTU时，会出现相同的情况（然后，当Cisco IOS交换机向AAA服务器发送EAP数据包时，会创建EAP-TLS分段）。

RADIUS属性Framed-MTU

对于RADIUS，RFC 2865中定义了Framed-MTU属性：

“此属性指示为用户配置的最大传输单元，当它不是通过其他方式（例如PPP）协商时。它可用于Access-Accept数据包。

NAS可能会在访问请求数据包中使用它作为它首选该值的提示，但并不要求服务器执行提示。”

ISE不执行提示。NAD在访问请求中发送的Framed-MTU值对ISE执行的分段没有任何影响。

除交换机上全局启用的巨帧设置外，多台现代Cisco IOS交换机不允许更改以太网接口的MTU。巨型帧的配置会影响RADIUS Access-Request中发送的Framed-MTU属性的值。例如，您可以设置：

```
<#root>

Switch(config)#

system mtu jumbo 9000
```

这会强制交换机在所有RADIUS访问请求中发送Framed-MTU = 9000。无巨型帧的系统MTU也一样：

```
<#root>

Switch(config)#

system mtu 1600
```

这会强制交换机在所有RADIUS Access-Requests中发送Framed-MTU = 1600。

请注意，现代Cisco IOS交换机不允许您将系统MTU值降低到1,500以下。

发送EAP分段时的AAA服务器和请求方行为

ISE

ISE始终尝试发送1,002字节长的EAP-TLS分段（通常为带证书的服务器呼叫）（尽管最后一个分段通常较小）。

它不遵守RADIUS Framed-MTU。无法将其重新配置为发送更大的EAP-TLS分段。

Microsoft网络策略服务器(NPS)

如果在NPS上本地配置Framed-MTU属性，则可以配置EAP-TLS分段的大小。

事件：虽然在Microsoft NPS上配置EAP负载大小文章提到NPS RADIUS服务器的框架MTU的默认值为1,500，但思科技术支持中心(TAC)实验室已显示，它使用默认设置发送2,000（在Microsoft Windows 2012数据中心上确认）。

经测试，NPS尊重根据前面提到的指南在本地设置Framed-MTU，并将EAP消息分段为Framed-MTU中设置的大小的片段。但是，在访问请求中收到的Framed-MTU属性未使用（与ISE/ACS上相同）。

设置此值是一种有效的解决方法，可以解决拓扑中的以下问题：

请求方[MTU 1500] ---- ---- [MTU 9000]交换机[MTU 9000] ----- ----- [MTU 9000]NPS

目前，交换机不允许您设置每个端口的MTU；对于6880交换机，此功能已添加思科漏洞ID CSCuo26327 - 802.1x EAP-TLS在FEX主机端口上不起作用。

AnyConnect

AnyConnect发送长度为1,486字节的EAP-TLS分段（通常是客户端证书）。对于此值大小，以太网帧为1,500字节。最后一个片段通常更小。

Microsoft Windows本地请求方

Microsoft Windows发送长度为1,486或1,482字节的EAP-TLS分段（通常是客户端证书）。对于此值大小，以太网帧为1,500字节。最后一个片段通常更小。

相关信息

- 配置基于 IEEE 802.1x 端口的身份验证
- 技术支持和文档 - Cisco Systems

关于此翻译
思科采用人工翻译与机器翻译相结合的方式将此文档翻译成不同语言，希望全球的用户都能通过各自的语言得到支持性的内容。

请注意：即使是最好的机器翻译，其准确度也不及专业翻译人员的水平。

Cisco Systems, Inc. 对于翻译的准确性不承担任何责任，并建议您总是参考英文原始文档（已提供链接）。