了解9800 WLC上的RADIUS MTU和分段

目录

简介

本文档介绍如何配置WLC发送到RADIUS服务器的RADIUS数据包的MTU。

先决条件

要求

Cisco建议您对这些主题有基本的了解：

- 9800无线LAN控制器(WLC)AAA配置
- 身份验证、授权和记帐(AAA)RADIUS概念
- 可扩展身份验证协议EAP
- 最大传输单位 (MTU)

使用的组件

- 思科身份服务工程师(ISE)3.2
- Catalyst 9800无线控制器系列(Catalyst 9800-L)
- 思科IOS® XE 17.15.2
- Windows 11无线客户端

本文档中的信息都是基于特定实验室环境中的设备编写的。本文档中使用的所有设备最初均采用原始（默认）配置。如果您的网络处于活动状态，请确保您了解所有命令的潜在影响。

背景

在本文档中，使用的远程身份验证拨入用户服务(RADIUS)服务器是Cisco ISE。首先，演示了数据包在可扩展身份验证协议(EAP)过程中如何无任何外部干预地流动。接下来是配置选项，用于更改WLC发送到任何RADIUS服务器的访问请求的大小。此选项在IOS-XE版本17.11中添加。

9800 RADIUS MTU

通常，RADIUS数据包的MTU并不重要，因为它们通常很小，并且无论如何都不会到达MTU。但是，当一端必须发送证书（通常为2-5KB）时，设备需要对该证书进行分段以将其置于其MTU下。

当客户端必须向RADIUS服务器发送证书时(如EAP传输层安全(EAP-TLS)),WLC会遇到由于必须随数据包一起发送的RADIUS数据量而需要重新分段数据包的情况。直到17.11之前，网络管理员几乎无法控制此过程，但现在工程师可以选择控制WLC发送的访问请求的大小。

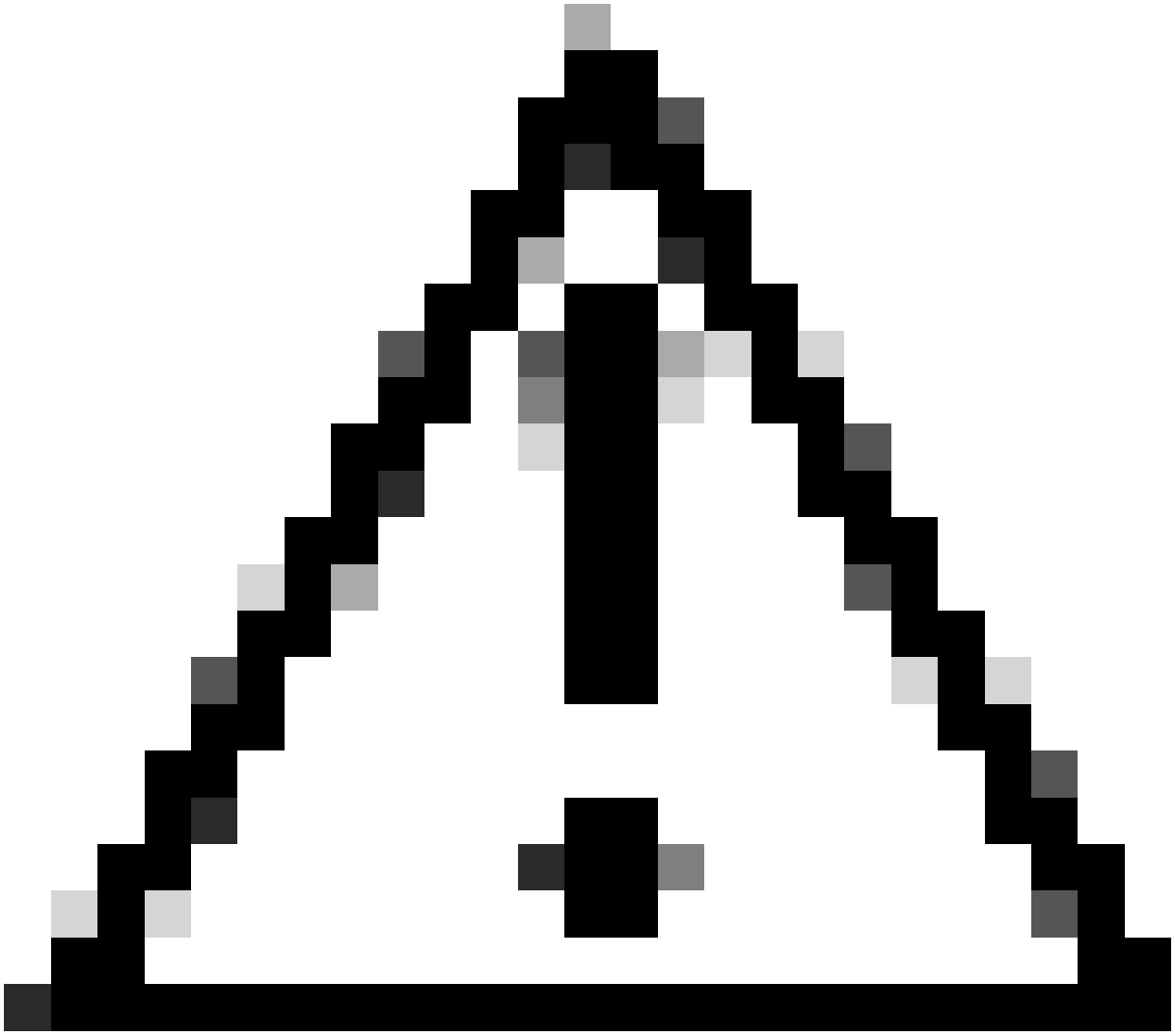
EAP-TLS数据包流

这在一定程度上深入研究了数据包的外观以及无线基础架构如何处理数据包。为了完全理解本文档中介绍的更改，在使用dot1x和更具体的EAP-TLS时，了解无线身份验证过程中的数据包流非常重要。

如果您已经深入了解EAP和RADIUS数据包流在Cisco无线基础设施中的工作方式，则转到行为更改部分，该部分说明在17.11中添加的内容，让网络管理员对RADIUS MTU拥有更多控制权。首先，请查看EAP标识(EAP-ID)。

EAP-ID

EAP-ID由身份验证器（在本例中为WLC）发起。这必须是EAP流程的第一部分。有时，无线客户端会发送EAPOL-Start。这通常意味着客户端从未收到EAP-ID请求或想要重新开始。



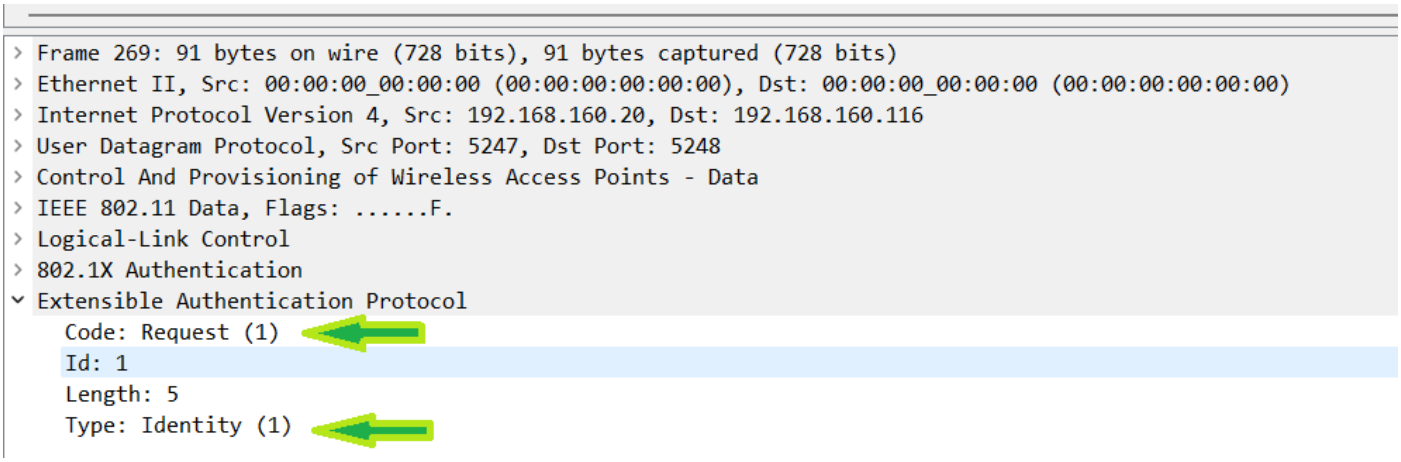
警告：EAP-ID数据包与EAP数据包ID之间有所不同。EAP-ID数据包用于标识请求方，其中EAP数据包ID是一个数字，用于在特定数据包通过网络传输时对其进行跟踪。

EAP-ID请求

首先，无线客户端设备使用正常的关联过程连接到网络。当无线局域网(WLAN)配置为dot1x时，WLC首先需要知道客户端是谁，然后才能从RADIUS服务器请求访问。要查找此信息，WLC会发送客户端和EAP-ID请求。

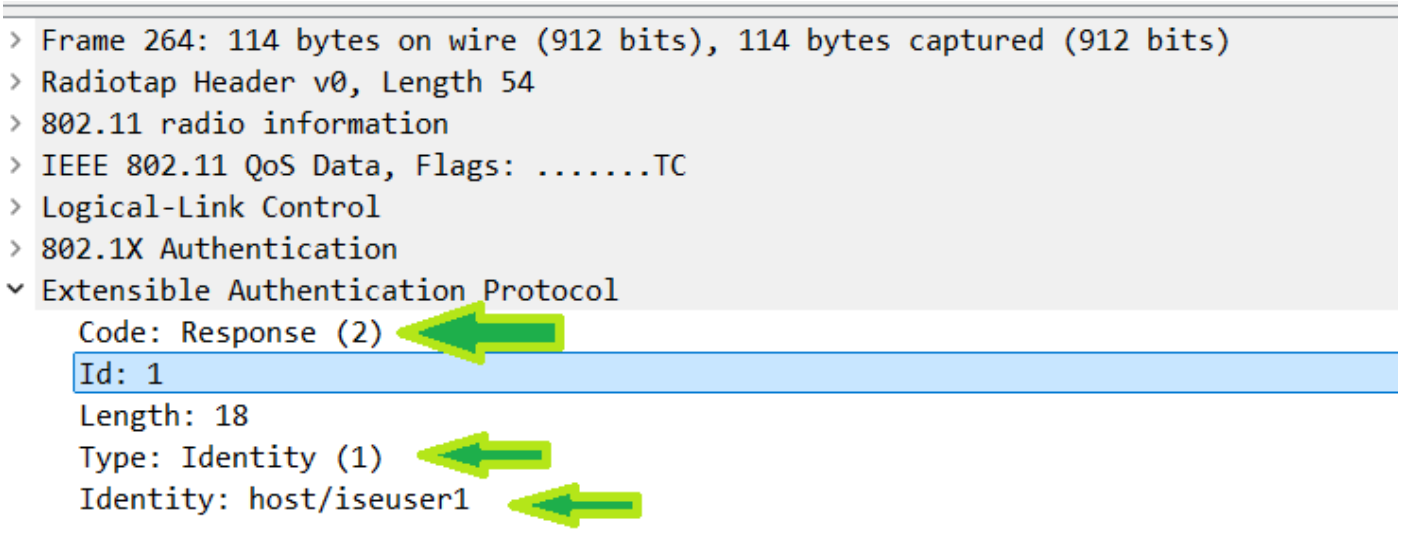
客户端应使用EAP-ID响应进行响应。这为WLC提供了构建访问请求并将其发送到ISE所需的功能。EAP-ID请求是指在正常PEAP身份验证中要求客户端输入其用户名和密码的时间。

但是，此讨论围绕EAP-TLS，因此此处的EAP-ID响应将仅具有用户ID。在演示中，用户ID为iseuser1。在此数据包中，您可以看到WLC发送到无线客户端的EAP-ID请求，询问他们是谁。由于这是一个无线客户端，因此WLC将请求封装在CAPWAP中，并将其发送到无线接入点(AP)，以便通过无线发送。在EAP数据中，代码1表示它是一个请求，类型1表示它是用于身份的。



EAP-ID响应

接下来，预计无线客户端将使用EAP-ID响应进行响应。在EAP数据中，代码已更改为2，表示这是一个响应，但类型仍为1，仍然显示为身份标识。在这里，您甚至可以看到客户端使用的用户名。还要检查这些数据包的是EAP数据包的ID号。对于EAP-ID交换，它始终为1，但此编号随后在ISE参与后更改为其他值。



您可以看到两个数据包都相当小，因此MTU在此没有承载，因为它远远低于网络中使用的1500字节。

Access-Request和Access-Challenge

与客户端的通信是EAP，WLC和ISE之间的通信是RADIUS。 对于RADIUS通信，使用access-request和access-challenge数据包。WLC从请求方接收EAP数据包，并使用RADIUS访问请求将其转发到ISE。在工作网络中，ISE会以访问质询做出响应。

访问请求

现在，WLC知道客户端的身份，它需要询问RADIUS服务器是否允许该客户端访问网络。为此，WLC通过发送access-request数据包来请求该客户端的访问权限。WLC将随同EAP数据一起发送

其他数据。这些数据统称为属性值对、AVP或AV对，具体取决于说话者。

本文档不会深入介绍AVP，因为这已超出本讨论范围。在这里，您只需看到用户名（EAP数据）已包含并发送到RADIUS服务器，在本例中为ISE。此外，您可以看到EAP-ID编号1也发送到ISE。请记住，当您查看空中EAP数据包ID时，也有1。此处需要注意的最后一点是，由于WLC已添加所有这些AVP，客户端发送的114字节数据包现在将转换为488字节数据包，然后才发送到ISE。

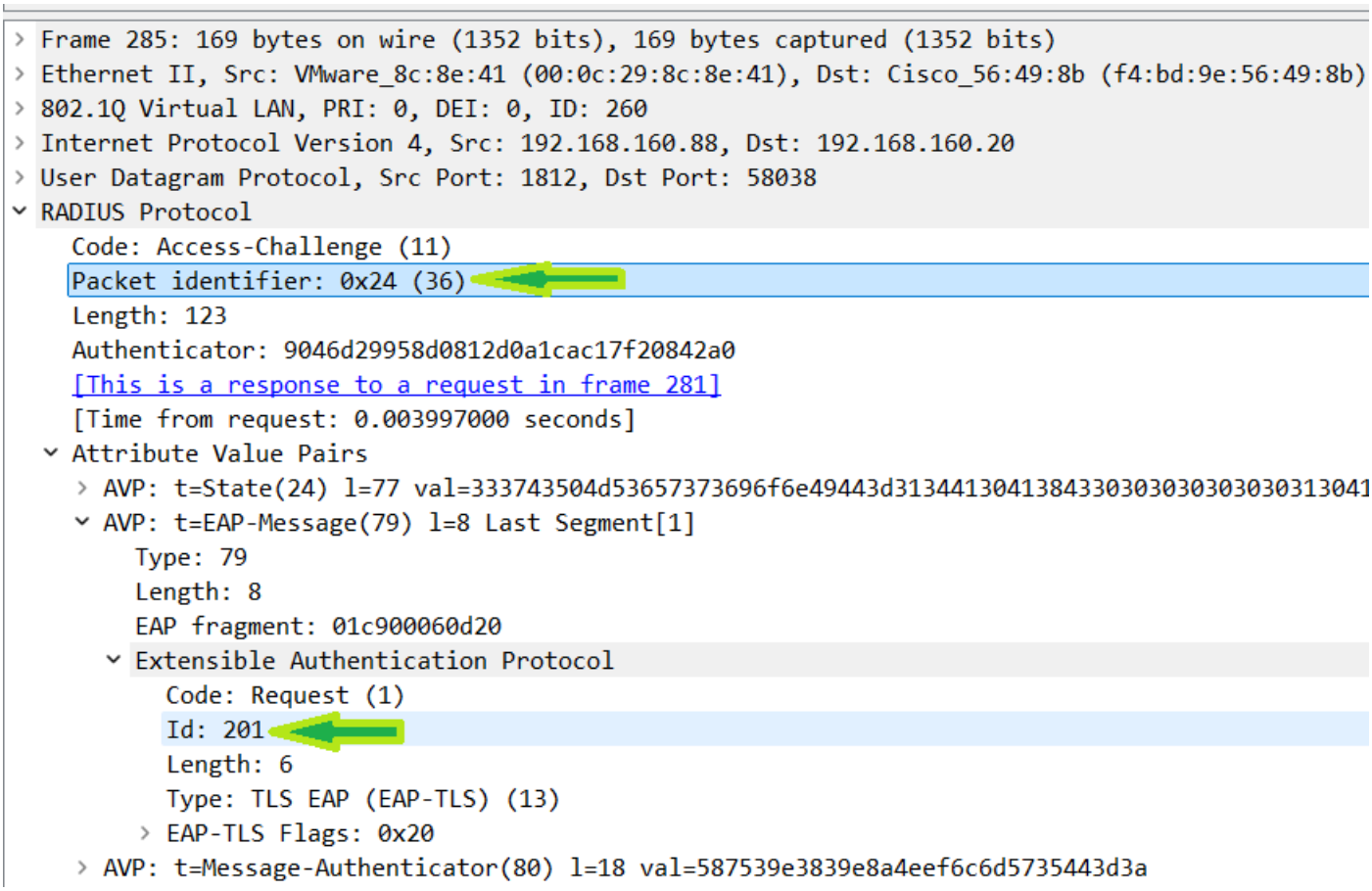
```
> Frame 281: 506 bytes on wire (4048 bits), 506 bytes captured (4048 bits)
> Ethernet II, Src: 00:00:00_00:00:00 (00:00:00:00:00:00), Dst: 00:00:00_00:00:00 (00:00:00:00:00:00)
> Internet Protocol Version 4, Src: 192.168.160.20, Dst: 192.168.160.88
> User Datagram Protocol, Src Port: 58038, Dst Port: 1812
v RADIUS Protocol
    Code: Access-Request (1)
    Packet identifier: 0x24 (36)
    Length: 464
    Authenticator: 48f74e792b11604d9188e4d947629485
    [The response to this request is in frame 285]
  v Attribute Value Pairs
    v AVP: t=User-Name(1) l=15 val=host/iseuser1
        Type: 1
        Length: 15
        User-Name: host/iseuser1
    > AVP: t=Service-Type(6) l=6 val=Framed(2)
    > AVP: t=Vendor-Specific(26) l=27 vnd=ciscoSystems(9)
    > AVP: t=Framed-MTU(12) l=6 val=576
    v AVP: t=EAP-Message(79) l=20 Last Segment[1]
        Type: 79
        Length: 20
        EAP fragment: 0201001201686f73742f6973657573657231
      v Extensible Authentication Protocol
          Code: Response (2)
          Id: 1
          Length: 18
          Type: Identity (1)
          Identity: host/iseuser1
    > AVP: t=Message-Authenticator(80) l=18 val=262b63190f7340d9b9db2f888ea1cb79
    > AVP: t=EAP-Key-Name(102) l=2 val=
```

访问质询

假设ISE收到访问请求并决定响应，此响应预期会作为ISE的访问质询。回顾访问请求，您会看到RADIUS数据包ID为36,AVP才会启动。

当WLC收到访问质询时，RADIUS ID必须与访问请求的数据包ID匹配。RADIUS数据包ID用于ISE和WLC之间的RADIUS通信。您还可以在此数据包中看到ISE已设置新的EAP ID 201，用于跟踪ISE和客户端之间的通信。此时，WLC只是用于ISE和客户端之间通信的通道。

请务必在此处记录所有这些数据包ID，以便您了解通信流以及如何通过网络跟踪这些数据包。在生产环境中，通常同时进行多个身份验证。使用calling-station-id命令将数据包与客户端的MAC地址进行匹配。然后，您可以使用RADIUS数据包ID和EAP数据包ID跟踪此特定客户端的数据包流。到目前为止，双方都没有发送任何证书，因此仍然不必担心MTU。



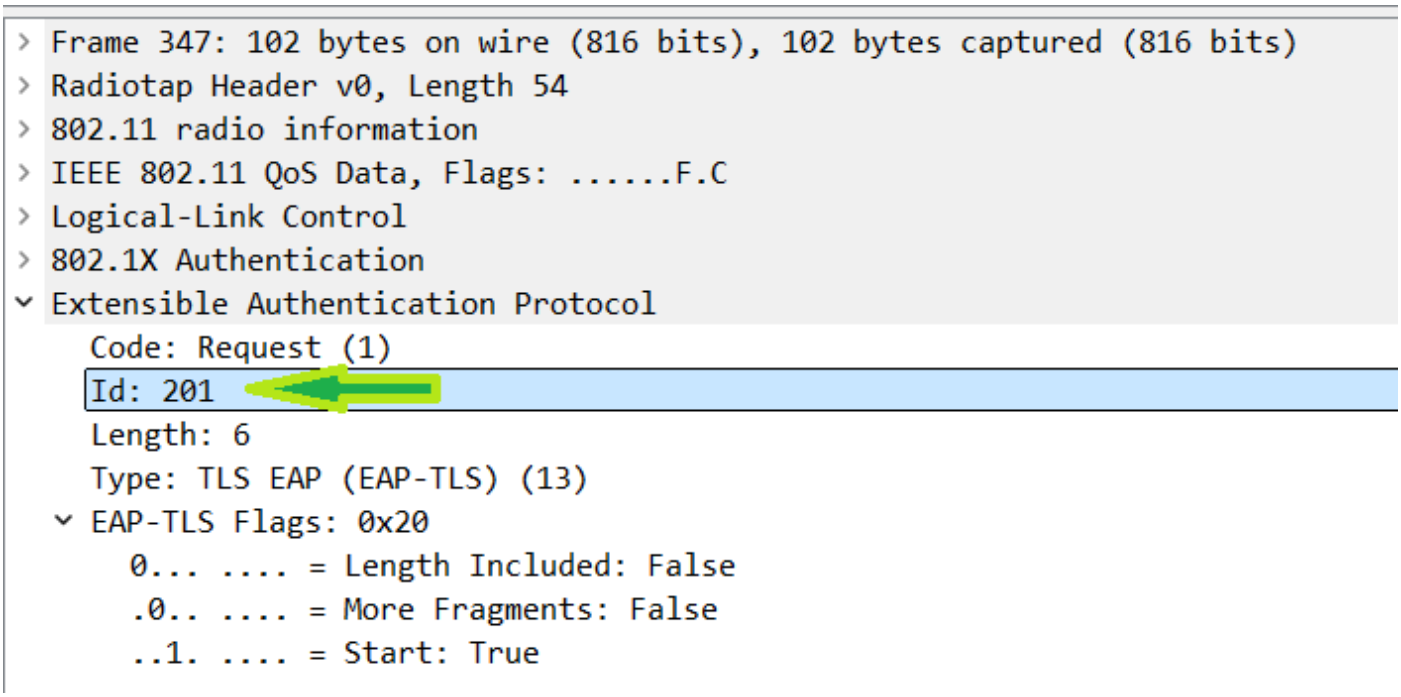
EAP请求和EAP响应

提醒一下，客户端说EAP而不是RADIUS。也就是说，当WLC收到访问质询时，它必须删除RADIUS数据并提取EAP请求，以便可以将其发送到客户端。

EAP请求

这必须与WLC收到访问质询时的情况完全相同。但是，所有RADIUS内容都已被删除，只有EAP部分被发送到客户端。

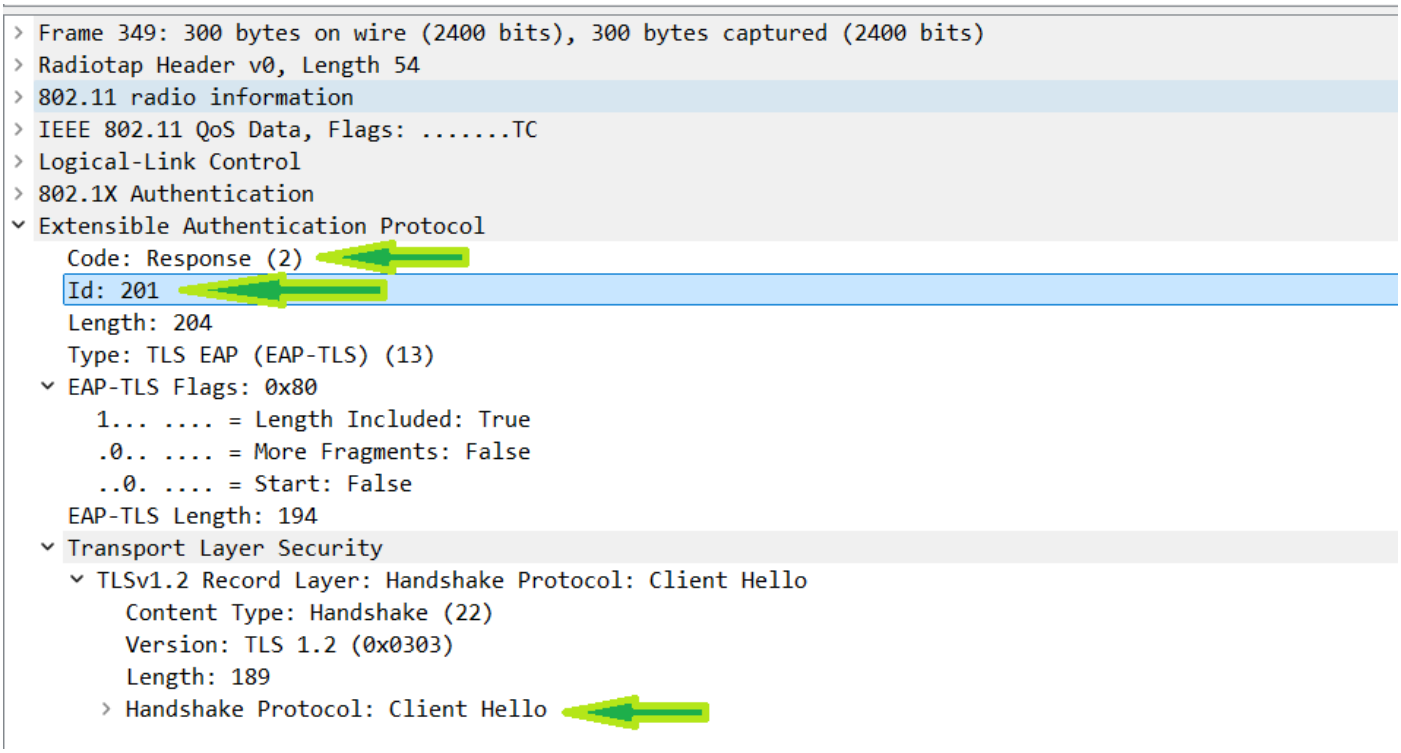
您仍然可以在此处看到EAP数据包ID 201，就像访问质询中一样，因为与WLC从ISE接收的数据相同。此处的流程与EAP-ID相同，只是现在它不是来自WLC并且用于建立EAP方法。此数据包仍然很小，因为它只是用于建立EAP-TLS会话的开始。



EAP响应

当客户端收到EAP-Request时，它必须使用EAP-Response进行响应。 在EAP-Response中，客户端开始建立TLS会话。这看起来与任何其它使用TLS的情况相同。它以“client hello”消息开头。本文档不会深入研究客户端Hello中的内容，因为它与此主题无关。您需要注意的是，正在设置TLS会话。

您可以在此处查看数据包中的数据，就像查看任何其他TLS设置时一样。与EAP-ID响应一样，此数据包会到达WLC并转换为访问请求。ISE使用在访问质询中打包的EAP请求进行响应。从现在起，这将继续是流程。



TLS证书

在这里您将看到数据包大小增加。根据一个或多个中间证书颁发机构(CA)的存在，证书可能相当大。 如果是自签名证书，则显然比设备证书链接到2个中间CA和一个根CA的证书小。无论哪种方式，您通常都会看到证书的所有者开始在此处对其自己的数据包进行分段。

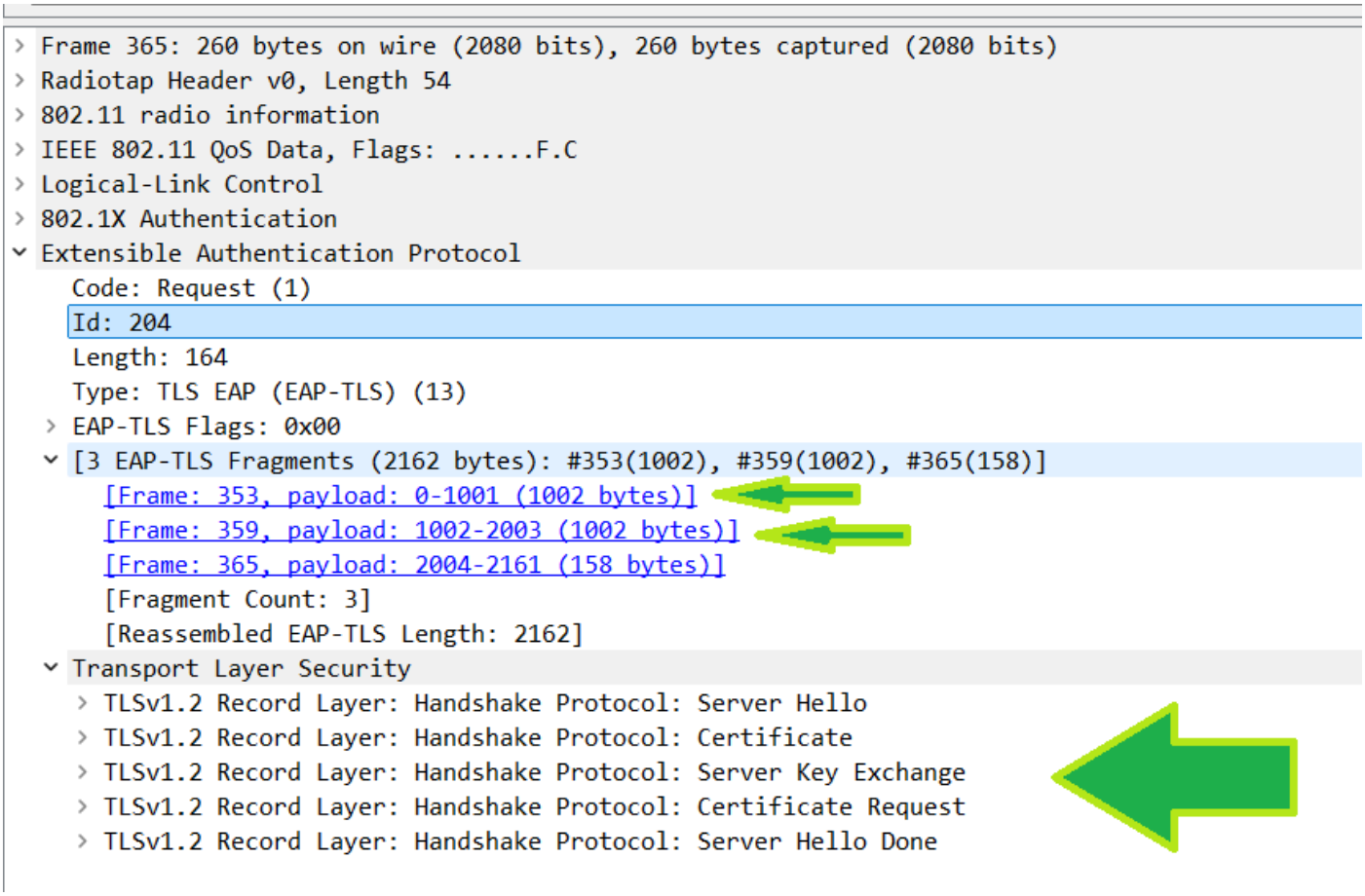
ISE证书

现在ISE已收到TLS客户端hello，它以另一个EAP请求进行响应。在此新EAP请求中，ISE会同时发送“服务器问候”消息、其证书、“服务器密钥交换”、“证书请求”和“服务器问候完成”消息。如果在一个数据包中发送所有此类数据包，则该数据包将通过MTU到达网络。因此，ISE对数据包本身进行分段，使其在MTU下获取。使用ISE时，它会分段数据包的数据部分，使其不大于1002字节，希望避免双重分段。

双重分段是什么意思？第一个分段发生在ISE上，因为它要发送的数据太大，无法容纳在网络的MTU中。不过，网络中也可能存在其他位置，即使MTU相同，由于网络的设置方式，设备可能需要对数据包进行分段，以便添加其报头并保持在MTU下。即使选中了do not fragment位，也可能出现这种情况。

VPN隧道或任何隧道就是很好的例子。要将数据放入VPN隧道，VPN路由器必须将其报头添加到流量中。如果此RADIUS流量在MTU处或靠近MTU分段，则当该VPN时，将无法将数据保留在MTU下并添加额外的报头。对于CAPWAP隧道也是如此，稍后您可以看到。

因此，为了避免这些数据包进入其他设备可以再次对其进行分段的情况，ISE会在大多数网络中可以避免这种情况的位置对数据包进行分段。这意味着ISE在多次等待空的EAP响应的EAP请求中发送此数据。EAP ID随每个分片的发送而增加。从WLC的角度来看，这是对每个分片的访问质询和访问请求交换，RADIUS数据包ID会随着每个分片的发送而增加。



客户端证书

一旦ISE发送所有分段并由客户端重组，数据包流将转移到客户端以发送某些内容。在TLS中，客户端此时会发送自己的证书，以便完成身份验证。事情在这里变得更加复杂。与ISE一样，客户端将同时发送多个TLS部分，其中一个是其证书。

与ISE不同，大多数客户端发送其EAP数据仅低于MTU。在本演示中，802.1x数据为1492。问题在于AP需要添加CAPWAP报头才能将其发送到WLC。

如何做到这一点？AP必须对数据包进行分段，以便添加报头并将其发送到WLC。 AP无法在不分段的情况下获取发送到WLC的数据包。也就是说，这里数据包是双分片的，首先来自客户端，然后再次来自AP。但是，此分段通常不是问题，因为对CAPWAP的预期是如此。

无线数据包：

```
> Frame 367: 1588 bytes (12704 bits), 1588 bytes captured (12704 bits)
> Radiotap Header v0, Length 54
> 802.11 radio information
> IEEE 802.11 QoS Data, Flags: .......TC
> Logical-Link Control
> 802.1X Authentication
v Extensible Authentication Protocol
    Code: Response (2)
    Id: 204
    Length: 1492
    Type: TLS EAP (EAP-TLS) (13)
  v EAP-TLS Flags: 0xc0
      1... .... = Length Included: True
      .1.. .... = More Fragments: True
      ..0. .... = Start: False
    EAP-TLS Length: 4692
```

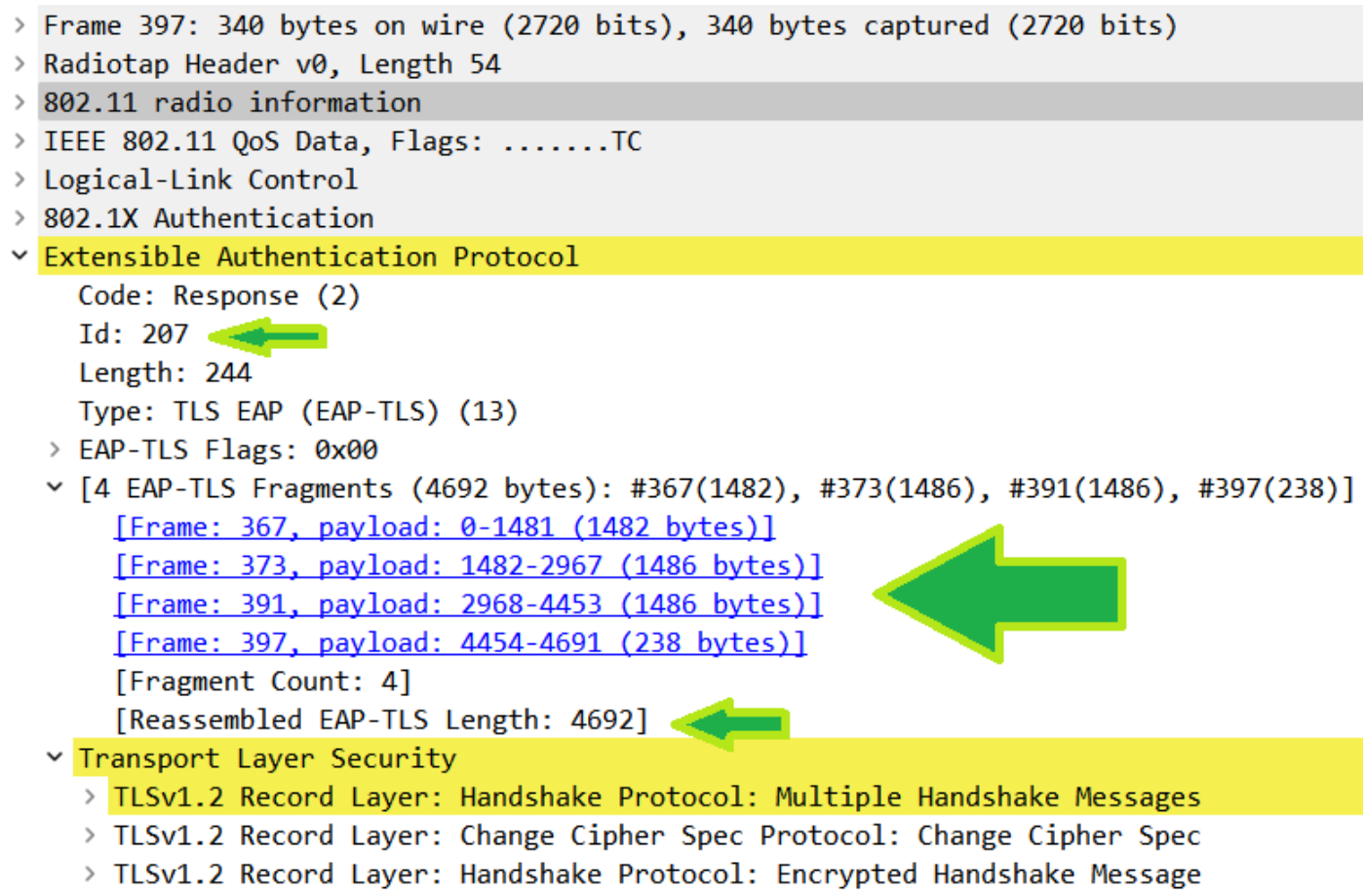
线路上的数据包分段：

```
> Frame 56: 1482 bytes (11856 bits), 1482 bytes captured (11856 bits) on interface /tmp
> Ethernet II, Src: Cisco_b5:e6:00 (0c:75:bd:b5:e6:00), Dst: Cisco_56:49:8b (f4:bd:9e:56:49:8b)
> Internet Protocol Version 4, Src: 192.168.160.116, Dst: 192.168.160.20
> User Datagram Protocol, Src Port: 5248, Dst Port: 5247
> Control And Provisioning of Wireless Access Points - Data
  [Reassembled in: 57]
v Data (1424 bytes)
    Data: 018800000c75bdb3022038689362ec7e0c75bdb3022f00010000aaaa03000000888e0100…
    [Length: 1424]
```

数据包在线路上重组：

```
Wireshark · Packet 57 · FromTheWire2.pcap
> Frame 57: 156 bytes (1248 bits), 156 bytes captured (1248 bits) on interface /tmp/epc_ws/wif_to_ts_pipe, id 0
> Ethernet II, Src: Cisco_b5:e6:00 (0c:75:bd:b5:e6:00), Dst: Cisco_56:49:8b (f4:bd:9e:56:49:8b)
> Internet Protocol Version 4, Src: 192.168.160.116, Dst: 192.168.160.20
> User Datagram Protocol, Src Port: 5248, Dst Port: 5247
> Control And Provisioning of Wireless Access Points - Data
> [2 Message fragments (1530 bytes): #56(1424), #57(106)]
> IEEE 802.11 QoS Data, Flags: .......T
> Logical-Link Control
> 802.1X Authentication
v Extensible Authentication Protocol
    Code: Response (2)
    Id: 204
    Length: 1492
    Type: TLS EAP (EAP-TLS) (13)
  > EAP-TLS Flags: 0xc0
    EAP-TLS Length: 4692
```

所有客户端分段都在空中重组：



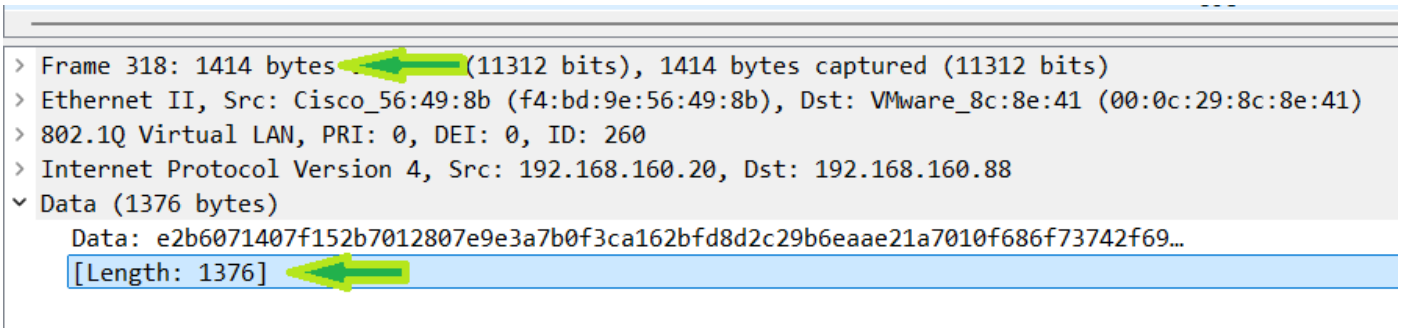
WLC上的客户端证书

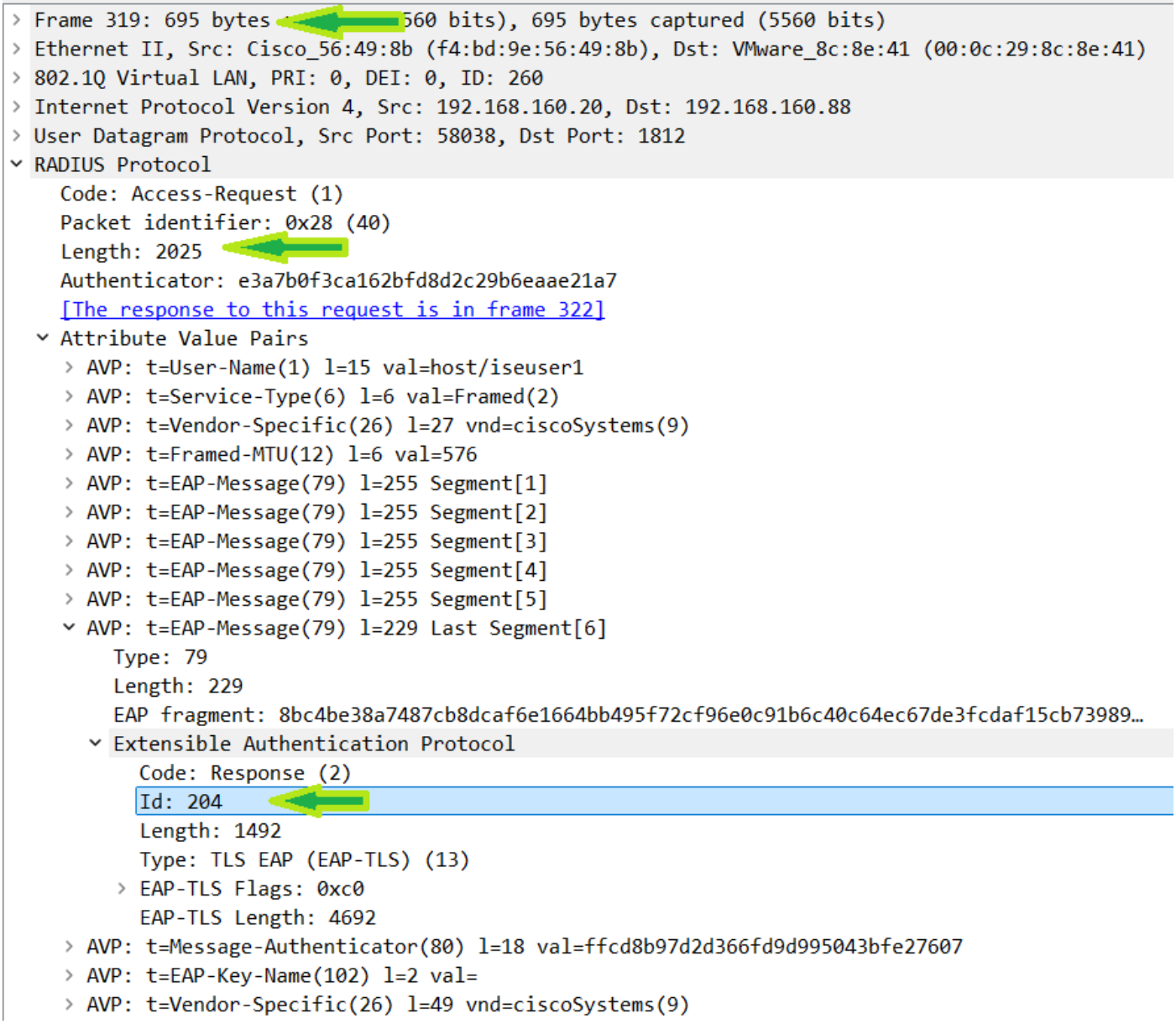
WLC从客户端接收两个CAPWAP分段，并重组它们以接收完整的1492字节数据包，从而恢复数据包 — 但恢复时间不长。此恢复时间很短，因为，如果您回顾WLC如何发送访问请求，您必须记住，它必须将大约400字节的RADIUS AVP添加到数据包，然后才能将数据发送到ISE。

为了简单计算，假设WLC添加408个字节，使数据包总大小达到1900。这远远超过1500 MTU，那么WLC将执行什么操作？再次对数据包进行分段。

此时，WLC将默认将数据包分段为1396。此处的想法与ISE相同。希望使数据包足够小，这样如果数据包必须经过另一个隧道，则无需重新分段即可添加报头。但是，WLC并不像ISE那样谨慎，因此1396在这里足够好。

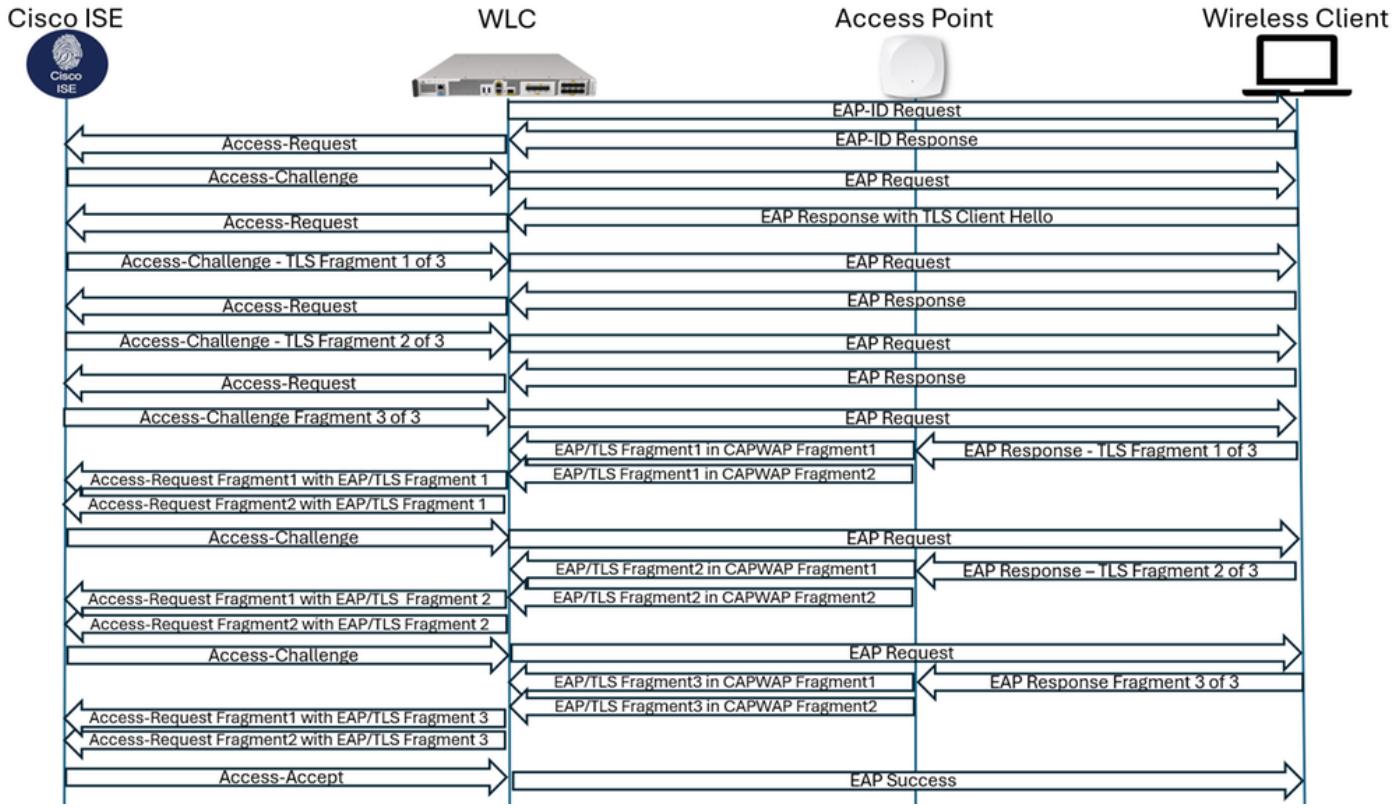
离开WLC的分段数据包：





数据包流TL;DR

当ISE发送其证书时，它会将大小为1002字节的TLS数据包分段。没问题了。当客户端发送其证书时，它们通常会分段接近MTU。由于AP必须将CAPWAP报头添加到数据包，因此也必须对此数据包进行分段。WLc收到分段后，必须重组数据包，但随后必须添加RADIUS AVP，以便再次对数据包进行分段。 数据包流如下所示：



RADIUS MTU行为更改

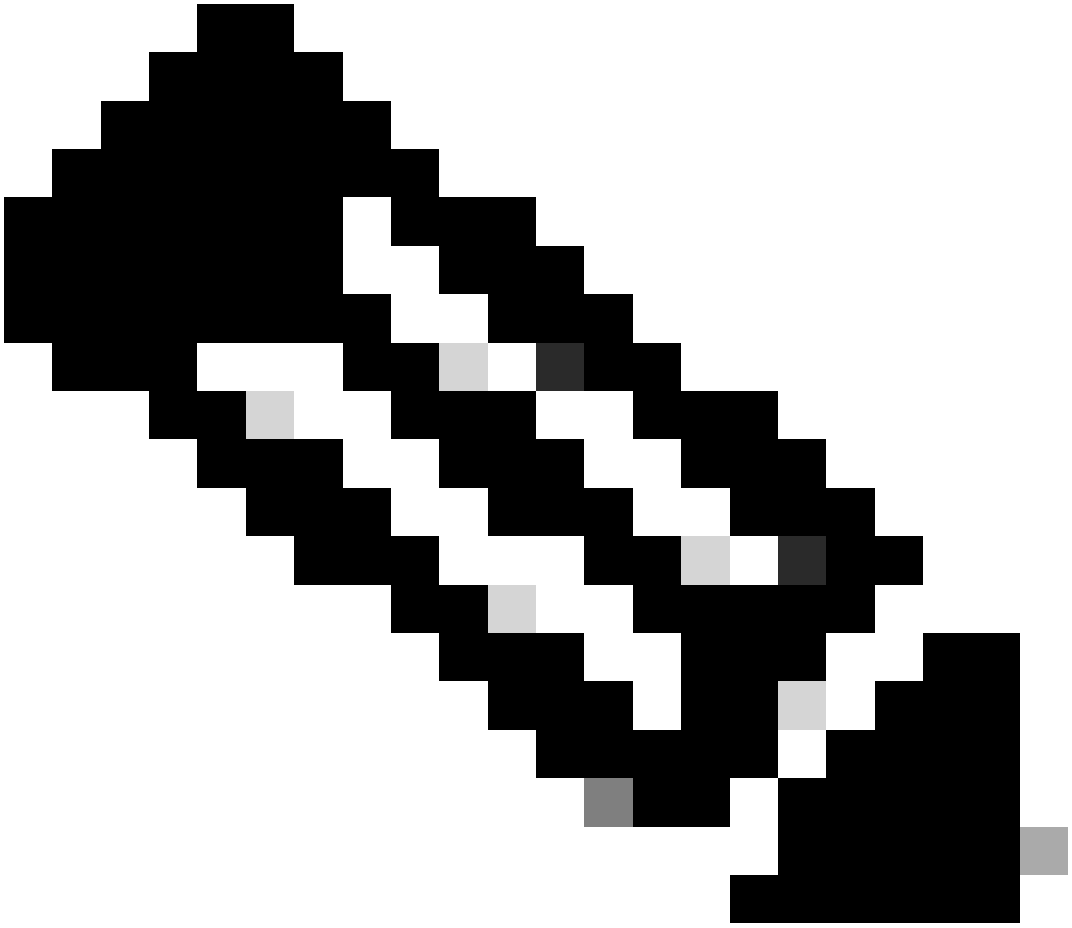
当您查看任何无线客户端数据流量的数据包流时，您会发现无线基础设施只在少数地方对其有影响。第一个位置是流量离开AP时，第二个位置是流量离开WLC时。但TCP流量除外，无线基础设施可以调整客户端MSS。但是，EAP并不属于TCP，事实上，它是自己的协议。

当您查看EAP和RADIUS流量时，还可以看到当原始数据包大小太接近MTU时，网络实际上影响AP和WLC上的流量大小。通过正确了解WLC在此交换中的作用，您可以看到实际上只有一个地方它影响RADIUS数据包的大小。这是当EAP响应进入且您将其更改为RADIUS访问请求时。

更改内容

如果EAP响应通过MTU，则添加RADIUS AVP后，您必须对其进行分段。由于不管数据包的内容如何，您都必须对其进行分段，因此您至少可以决定分段大小如何。这就是17.11中引入的行为更改的来源。

在Cisco bug id CSCwc81849中跟踪的功能请求中，您希望添加对RADIUS巨型数据包的支持。这样做的方法是RADIUS数据包不再在1396自动分段。现在，如果添加ip radius source-interface <interface X>命令，则会在该接口的MTU处发送RADIUS访问请求。

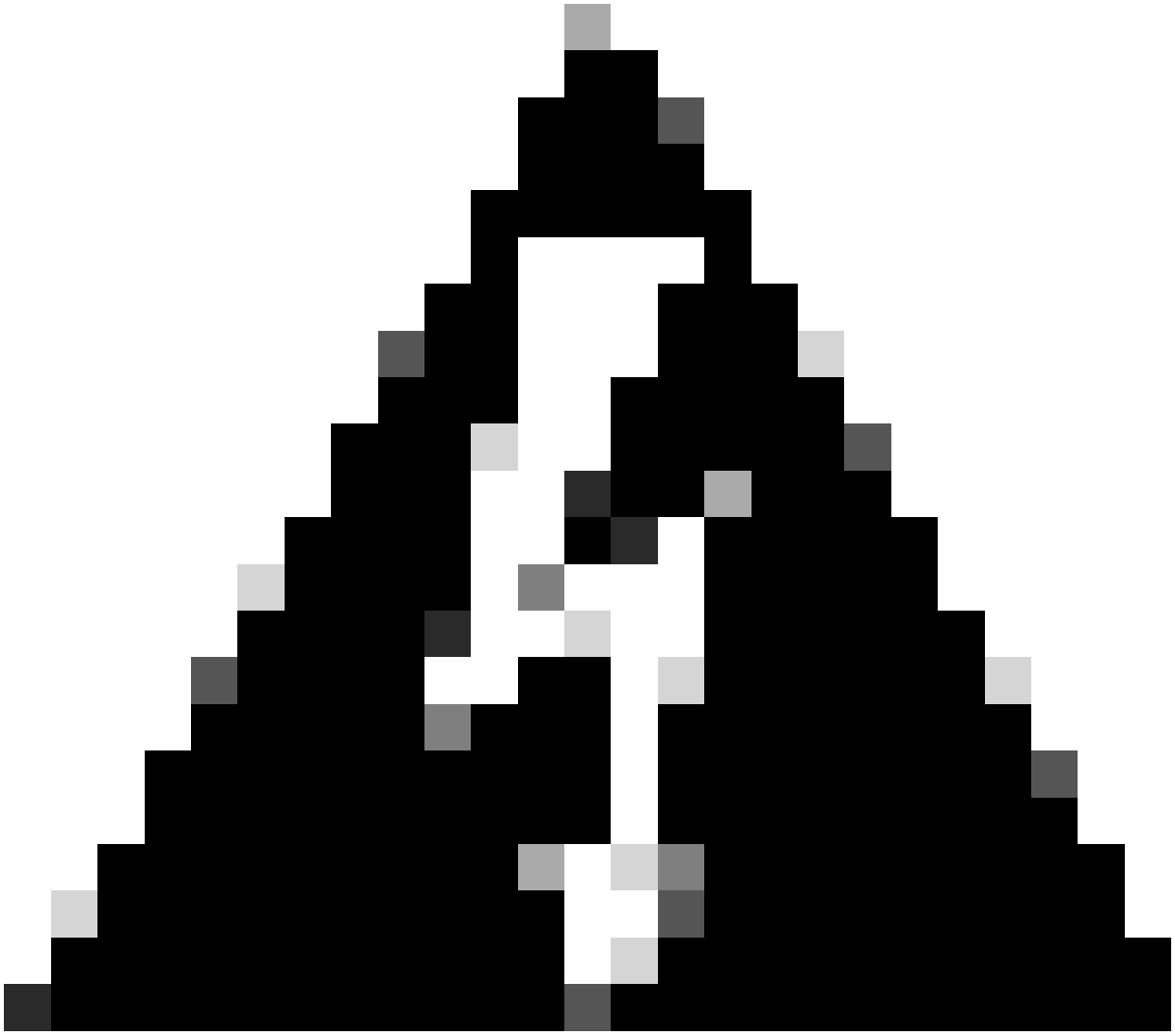


注意：如果您使用的是Cisco Catalyst Center，则在调配AAA配置时，它会自动将源接口添加到服务器组。这会将默认行为更改为在该命令中使用的接口的MTU大小进行分段。

如何使用此更改

由于所有接口的默认MTU为1500，因此这是要分段的新的MTU。用于所有RADIUS流量的默认接口是无线管理接口(WMI)。 当您查看服务器组的配置时，如果没有指定源接口，则WLC使用WMI在1396发送RADIUS流量。但是，如果您进入服务器组配置，并告知源接口是WMI，WLC现在会发送仍然使用WMI的1500处的RADIUS流量。

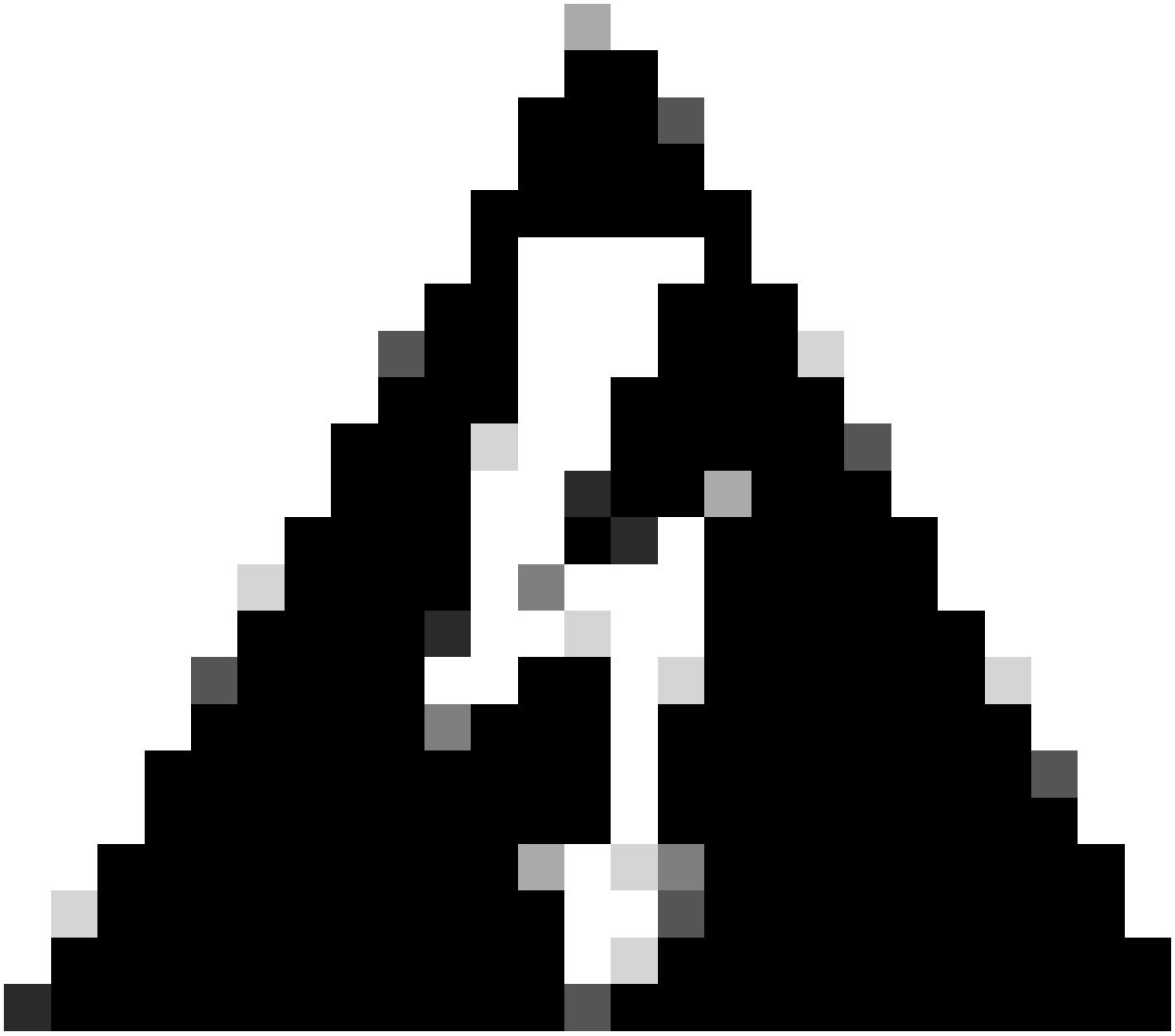
现在，假设网络中有之前讨论过的VPN设备。您不希望流量被双重分段，因此可以将接口的MTU更改为较小的值，以便在不同的位置对数据包进行分段。您可以将MTU更改为类似于1200的设置，这样数据包将以1200字节标记分片，而不是1500字节。



警告：更改WMI的MTU会影响所有进出WLC管理IP地址的流量。

即使您不想更改WMI的MTU，指定源接口的要点是将其从WMI更改为另一个接口，并将该接口用于RADIUS流量，以及更改该接口上的MTU。由于此配置是在服务器组级别完成的，您可以非常具体地了解您希望此更改受哪些RADIUS流量影响。

此配置不与AAA服务器或WLAN关联。可以有多个服务器组，其中含有相同的服务器，并且您选择时，只需在其中一个服务器上指定源接口。此服务器组将添加到方法列表，然后添加到WLAN。例如，如果只有一个WLAN您希望进行此更改，即使您只有一个AAA服务器，也可以创建一个新的服务器组，使用指向要使用的MTU的接口的ip radius source-interface命令，将AAA服务器添加到此新组，使用此新组创建一个新方法列表，然后将该方法列表添加到希望进行此更改的特定WLAN。



警告：在对现用网络进行ANY更改时，始终建议您在维护时段进行更改。

证据在数据包捕获中

一般认为在网络中，如果您未捕获到它，则无法证明它。下面是几个配置示例，这些更改已准备就绪，可向您展示如何工作。

这是WLAN配置。在测试期间，仅更改方法列表中使用的服务器组。

```
9800#show run wlan
wlan TLS-Test 2 TLS-Test
 radio policy dot11 24ghz
 radio policy dot11 5ghz
 no security ft adaptive
 security dot1x authentication-list TLS-AuthC
 no shutdown
!
```

使用默认MTU添加Source-Interface命令

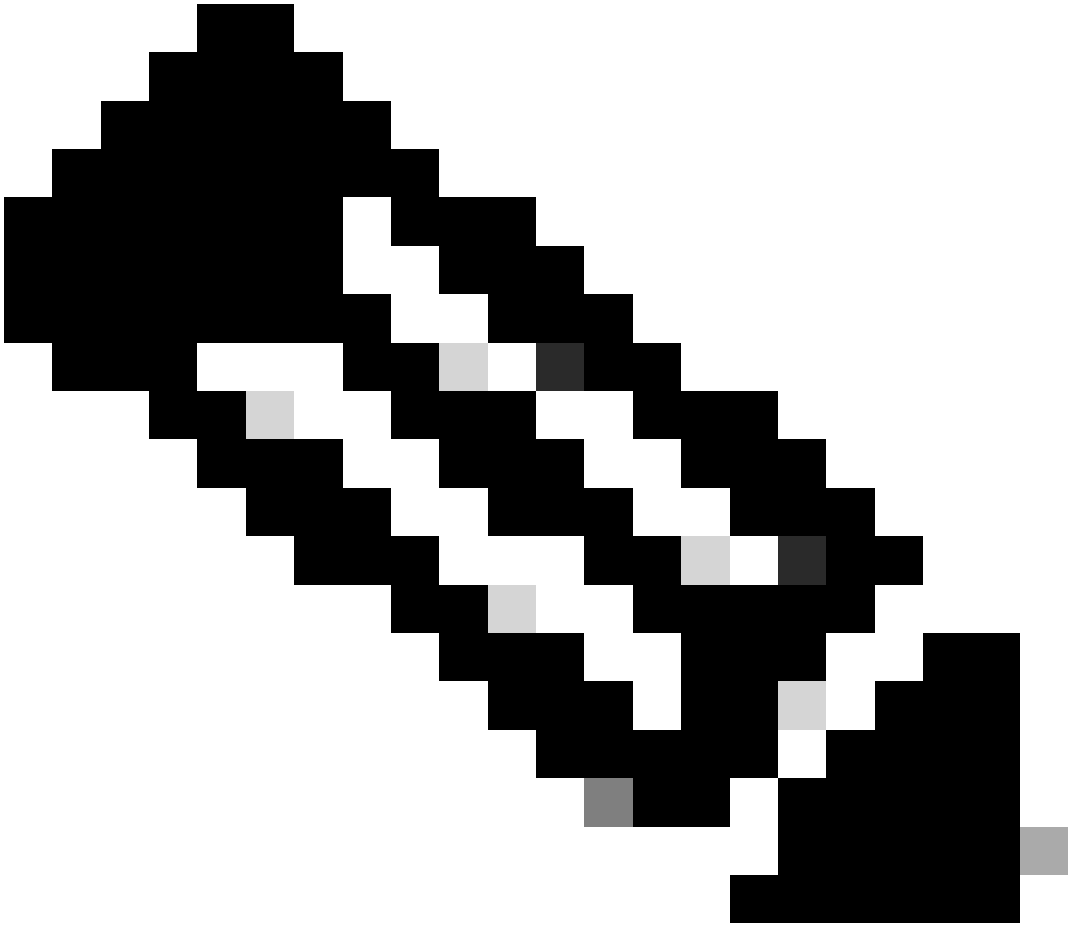
在这里，它只是指向ISE服务器的普通服务器组。添加的源接口命令指向未设置MTU的WMI。配置如下所示。

```
9800#show run aaa
!
aaa authentication dot1x TLS-AuthC group NoMTU
!
!
radius server ISE
 address ipv4 192.168.160.10 auth-port 1812 acct-port 1813
 key 6 _`gINMNXObF[^bPBVNaYibbBMhNMFAbKUAAB
!
aaa group server radius NoMTU
 server name ISE
 ip radius source-interface Vlan260
 deadtime 5
!
9800#show run inter vlan 260
!
interface Vlan260
 ip address 192.168.160.20 255.255.255.0
 no ip proxy-arp
end
```

如您所见，NoMTU服务器组已添加到与WLAN关联的身份验证方法列表。ip radius source-interface VLAN260命令用于此服务器组，而VLAN 260未指定MTU，表示它使用MTU 1500。为了确认，MTU为1500，您可以使用show run all命令并在输出中查找接口。

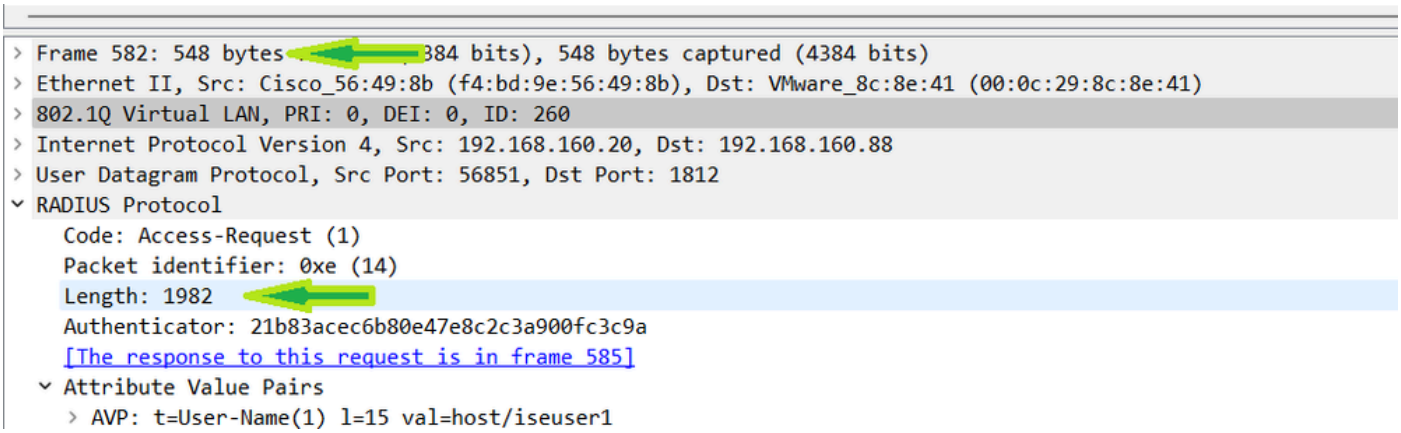
```
interface Vlan260
 ip address 192.168.160.20 255.255.255.0
 no ip clear-dont-fragment
 ip redirects
 ip unreachables
 no ip proxy-arp
 ip mtu 1500
```

现在，查看WLC添加RADIUS数据后客户端证书必须发送到ISE的数据包：



注意：此处，线路上的字节为1518。这包括以太网负载外部的报头，例如VLAN报头和第2层报头。这些不计入MTU。

```
> Frame 581: 1518 bytes on wire (12144 bits), 1518 bytes captured (12144 bits)
> Ethernet II, Src: Cisco_56:49:8b (f4:bd:9e:56:49:8b), Dst: VMware_8c:8e:41 (00:0c:29:8c:8e:41)
> 802.1Q Virtual LAN, PRI: 0, DEI: 0, ID: 260
> Internet Protocol Version 4, Src: 192.168.160.20, Dst: 192.168.160.88
v Data (1480 bytes)
    Data: de13071407c63226010e07be21b83acec6b80e47e8c2c3a900fc3c9a010f686f73742f69…
    [Length: 1480]
```



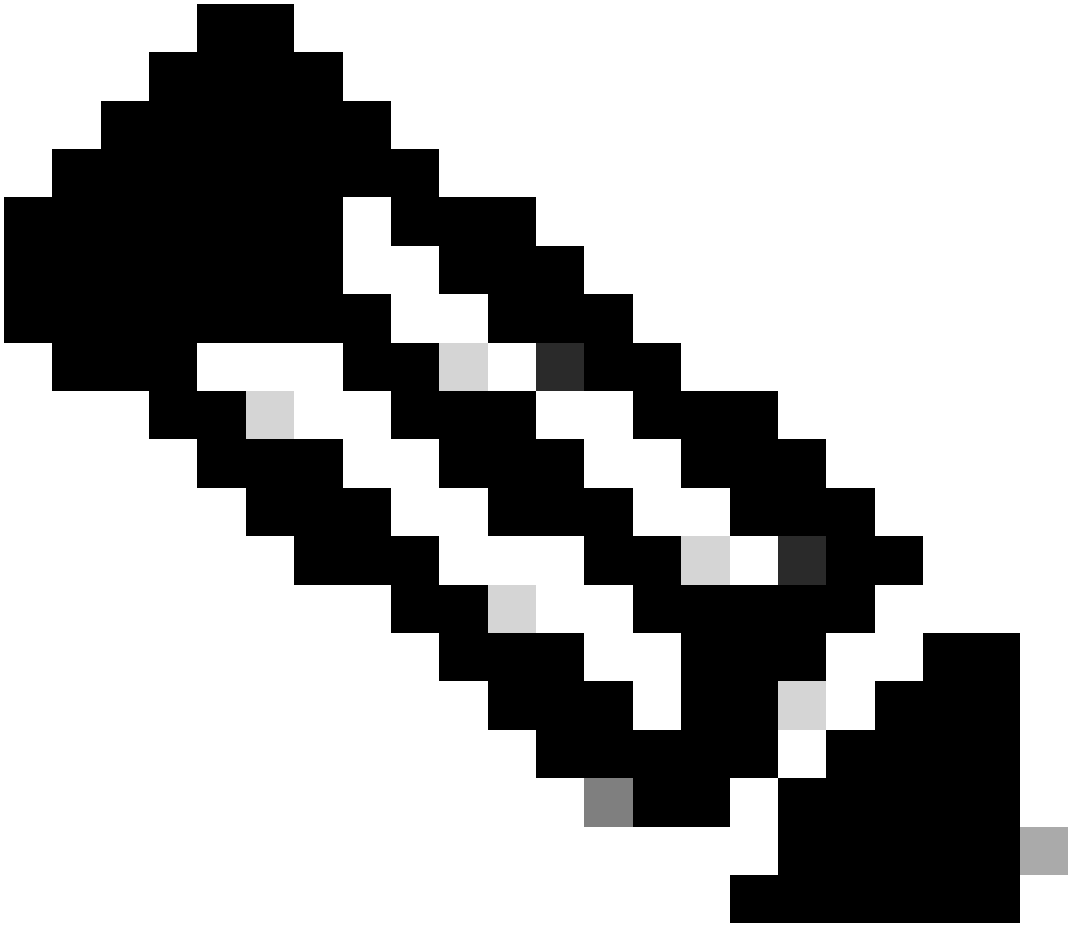
在这里，您可以看到数据部分在1480被分段。您可以在WMI上的1500 MTU下获取该片段。下一个数据包小于550字节，但您可以看到RADIUS数据的总大小为1982。也就是说，使用新的MTU进行分段现在可以正常工作。

使用MTU为1200的非WMI接口

现在，假设您想在较小的MTU处进行分段，但不想让此更改影响任何其他流量。没有问题，配置保持不变，只有源接口配置将指向仅为此目的创建的SVI。更改方法列表以指向此新服务器组，并且此服务器组使用的源接口不是我的WMI，并且MTU设置为1200。配置如下所示：

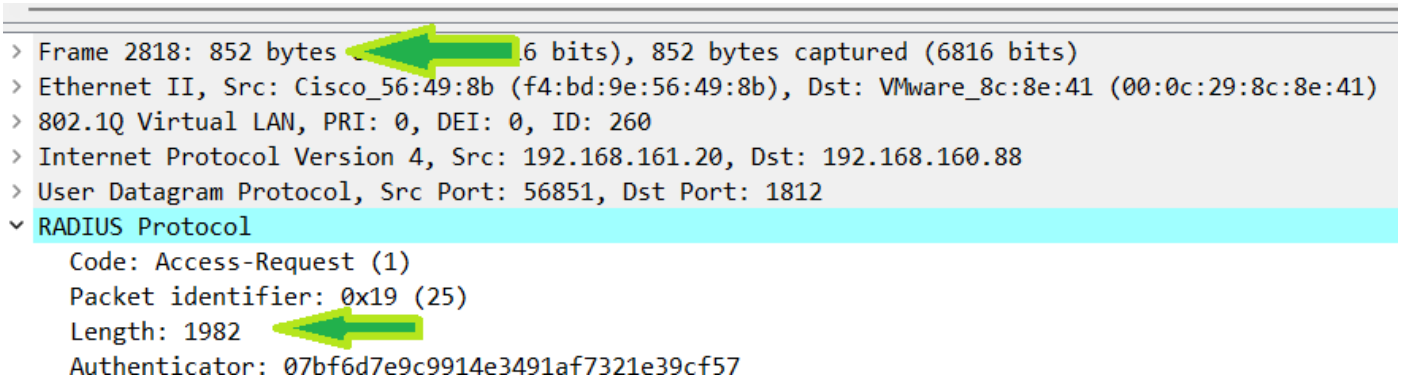
```
9800#show run aaa
!
aaa authentication dot1x TLS-AuthC group MTU1200
!
!
radius server ISE
 address ipv4 192.168.160.10 auth-port 1812 acct-port 1813
 key 6 _`gINMNXObFibbBMhNMFAbKUAAB
!
aaa group server radius MTU1200
 server name ISE
 ip radius source-interface Vlan261
 deadtime 5
!
9800#show run inter vlan 261
!
interface Vlan261
 ip address 192.168.161.20 255.255.255.0
 no ip proxy-arp
 ip mtu 1200
end
```

接下来，查看数据包在此较低的MTU下的外观。



注意：降低MTU和更改分段点不是新行为的一部分。这始终是事实。如果在1396进行分段的默认行为不适合MTU，则始终会在不同的点进行分片。本部分内容只是为了帮助解释可用的选项。

```
> Frame 2817: 1214 bytes c      2 bits), 1214 bytes captured (9712 bits)
> Ethernet II, Src: Cisco_56:49:8b (f4:bd:9e:56:49:8b), Dst: VMware_8c:8e:41 (00:0c:29:8c:8e:41)
> 802.1Q Virtual LAN, PRI: 0, DEI: 0, ID: 260
> Internet Protocol Version 4, Src: 192.168.161.20, Dst: 192.168.160.88
v Data (1176 bytes)
    Data: de13071407c6b995011907be07bf6d7e9c9914e3491af7321e39cf57010f686f73742f69…
    [Length: 1176]
```



这里，RADIUS数据仍然是1982个字节，但这次数据被分片为1176，而不是默认的1376，如果没有使用源接口，它应该已经分片。请记住，当您将MTU设置为1500并使用source-interface命令时，您将在1480进行分段。使用此处的配置，您可以控制到较低MTU的流量，而不会干扰WLC上的其他流量。

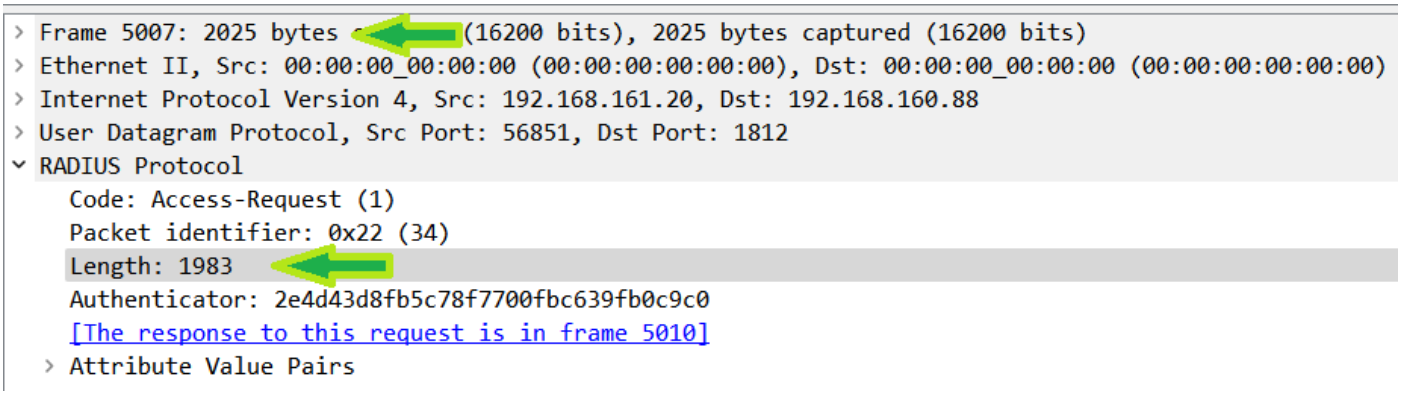
对巨型帧使用9000的MTU

由于该功能用于发送巨型帧，因此如果不测试该功能，同时仍使用VLAN 261的非WMI接口，将是一种耻辱。但是，现在IP MTU设置为9000。快速注意，为了能够在SVI上设置IP MTU，您必须将MTU设置为高于IP MTU的设置。您可以在此配置中看到以下内容：

```
9800(config-if)#do sho run inter vl 261
!
interface Vlan261
 mtu 9100
 ip address 192.168.161.20 255.255.255.0
 no ip proxy-arp
 ip mtu 9000
end
```

在这里，您可以看到捕获的数据包从未分段。它作为一个完整的数据包发送，RADIUS数据大小为1983。请记住，要使此功能发挥作用，网络其余部分需要配置为允许这样大小的数据包通过。

此处需要注意的另一点是，客户端MTU未更改，因此客户端仍然在1492对EAP数据包进行分段。区别在于WLC可以添加发送数据包到ISE所需的所有RADIUS数据，而无需对客户端数据进行分段。



结论

使用EAP-TLS时，客户端需要将其证书发送到AAA服务器。这些证书通常大于MTU，因此客户端必须对其进行分段。客户端对数据进行分段时非常接近MTU。由于AP必须添加CAPWAP报头，因此客户端发送的内容必须分段。WLC接收这两个数据包，将它们放在一起，但随后必须对其重新分段以添加RADIUS数据。此时，网络管理员可以控制客户端发送的EAP数据包的WLC分片方式。

如果将ip radius source-interface <interface you want to use>命令添加到AAA服务器组，则WLC将使用您放置的任何接口，而非（或包括）WMI。使用此命令还告知WLC在该接口的MTU而非默认的1396处进行分段。这样，您就可以更好地控制数据包在网络中的传输方式。

使用Cisco Catalyst Center时，source interface命令会添加到服务器组，从而更改默认行为。

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）。