

단편화 문제 해결: Azure에서 c9800 Wireless Controller에 영향을 미침

목차

[소개](#)

[증상](#)

[ISE 서버에서 오류 발생](#)

[자세한 로그 분석:](#)

[무선 컨트롤러 EPC:](#)

[ISE TCP 덤프](#)

[분석을 통한 Azure Side Capture:](#)

[무선 컨트롤러 쪽에서 제안하는 해결 방법:](#)

[해결책:](#)

소개

이 문서에서는 Azure 플랫폼의 알려진 문제로 인해 시퀀스가 잘못된 프래그먼트가 잘못 처리되어 패킷이 손실되는 것에 대해 설명합니다.

증상

영향을 받는 제품: Azure에서 호스팅되는 Catalyst 9800-CL Wireless Controller 또는 Azure에서 호스팅되는 Identity Service Engine.

SSID 설정: 중앙 인증이 있는 802.1x EAP-TLS에 대해 구성되었습니다.

행동: EAP-TLS 기반 SSID를 사용하여 Azure 플랫폼에서 호스팅된 9800-CL을 사용하는 동안 연결 문제가 발생할 수 있습니다. 인증 단계에서 클라이언트에 문제가 발생할 수 있습니다.

ISE 서버에서 오류 발생

EAP-TLS 인증서 교환 중에 서 폴리 컨 트가 ISE와의 통신을 중지 했다는 것을 나타내는 오류 코드 5411.

자세한 로그 분석:

다음은 영향을 받는 컨피그레이션 중 하나에 대한 예입니다. 9800 Wireless Controller에서 SSID는 802.1x에 대해 설정되고 AAA 서버는 EAP-TLS에 대해 구성됩니다. 클라이언트가 인증을 시도할 때 , 특히 인증서 교환 단계 중에 클라이언트는 무선 컨트롤러에서 MTU(Maximum Transmission Unit)

크기를 초과하는 인증서를 보냅니다. 그런 다음 9800 Wireless Controller는 이 큰 패킷을 프래그먼트화하고 프래그먼트를 순차적으로 AAA 서버에 전송합니다. 그러나 이러한 프래그먼트는 물리적 호스트에 올바른 순서로 도착하지 않으므로 패킷이 삭제됩니다.

다음은 클라이언트가 연결하려고 할 때 무선 컨트롤러에서 RA 추적을 보여 줍니다.
L2 인증 상태로 들어가는 클라이언트 및 EAP 프로세스가 시작됩니다.

```
2023/04/12 16:51:27.606414 {wncd_x_R0-0}{1}: [dot1x] [19224]: (정보):
[Client_MAC:capwap_900000004] 요청 상태 입력
2023/04/12 16:51:27.606425 {wncd_x_R0-0}{1}: [dot1x] [19224]: (정보):
[0000.0000.0000:capwap_900000004] EAPOL 패킷 전송 중
2023/04/12 16:51:27.606494 {wncd_x_R0-0}{1}: [dot1x] [19224]: (정보):
[Client_MAC:capwap_900000004] EAPOL 패킷 전송 - 버전: 3,EAPOL 유형: EAP, 페이로드
길이: 1008, EAP 유형 = EAP-TLS
2023/04/12 16:51:27.606496 {wncd_x_R0-0}{1}: [dot1x] [19224]: (정보):
[Client_MAC:capwap_900000004] EAP 패킷 - 요청, ID: 0x25
2023/04/12 16:51:27.606536 {wncd_x_R0-0}{1}: [dot1x] [19224]: (정보):
[Client_MAC:capwap_900000004] 클라이언트로 전송된 EAPOL 패킷
2023/04/12 16:51:27.640768 {wncd_x_R0-0}{1}: [dot1x] [19224]: (정보):
[Client_MAC:capwap_900000004] 수신된 EAPOL 패킷 - 버전: 1,EAPOL 유형: EAP, 페이로드
길이: 6, EAP-Type = EAP-TLS
2023/04/12 16:51:27.640781 {wncd_x_R0-0}{1}: [dot1x] [19224]: (정보):
[Client_MAC:capwap_900000004] EAP 패킷 - 응답, ID: 0x25
```

무선 컨트롤러가 AAA 서버에 액세스 요청을 보내고 패킷 크기가 1500바이트(무선 컨트롤러의 기본 MTU) 미만인 경우, 액세스 챌린지가 복잡성 없이 수신됩니다.

```
2023/04/12 16:51:27.641094 {wncd_x_R0-0}{1}: [radius] [19224]: (정보):
RADIUS: 172.16.26.235:1812 id 0/6, len 552로 액세스 요청 보내기
2023/04/12 16:51:27.644693 {wncd_x_R0-0}{1}: [radius] [19224]: (정보):
RADIUS: Received from id 1812/6 172.16.26.235:0, Access-Challenge, len
1141
```

클라이언트가 인증을 위해 인증서를 보낼 수도 있습니다. 패킷 크기가 MTU를 초과할 경우 더 이상 전송되기 전에 프래그먼트화됩니다.

```
2023/04/12 16:51:27.758366 {wncd_x_R0-0}{1}: [radius] [19224]: (정보):
RADIUS: 172.16.26.235:1812 id 0/8, len 2048로 액세스 요청 보내기
2023/04/12 16:51:37.761885 {wncd_x_R0-0}{1}: [radius] [19224]: (정보):
RADIUS: 5초 시간 초과 시작됨
2023/04/12 16:51:42.762096 {wncd_x_R0-0}{1}: [radius] [19224]: (정보):
RADIUS: ID 0/8의 재전송 대상(172.16.26.235:1812,1813)
2023/04/12 16:51:32.759255 {wncd_x_R0-0}{1}: [radius] [19224]: (정보):
RADIUS: ID 0/8의 재전송 대상(172.16.26.235:1812,1813)
2023/04/12 16:51:32.760328 {wncd_x_R0-0}{1}: [radius] [19224]: (정보):
RADIUS: 5초 시간 초과 시작됨
```

2023/04/12 16:51:37.760552 {wncd_x_R0-0}{1}: [radius] [19224]: (정보):
RADIUS: ID 0/8의 재전송 대상(172.16.26.235:1812,1813)
2023/04/12 16:51:42.762096 {wncd_x_R0-0}{1}: [radius] [19224]: (정보):
RADIUS: ID 0/8의 재전송 대상(172.16.26.235:1812,1813)

패킷 크기가 2048이며, 이는 기본 MTU를 능가합니다. 따라서 AAA 서버에서 응답이 없습니다. 무선 컨트롤러는 최대 재시도 횟수에 도달할 때까지 액세스 요청을 지속적으로 재전송합니다. 응답이 없기 때문에 무선 컨트롤러는 궁극적으로 EAPOL 프로세스를 재설정합니다.

2023/04/12 16:51:45.762890 {wncd_x_R0-0}{1}: [dot1x] [19224]: (정보):
[Client_MAC:capwap_900000004] 클라이언트에서 EAPOL_START 게시
2023/04/12 16:51:45.762956 {wncd_x_R0-0}{1}: [dot1x] [19224]: (정보):
[Client_MAC:capwap_900000004] 초기화 상태 시작
2023/04/12 16:51:45.762965 {wncd_x_R0-0}{1}: [dot1x] [19224]: (정보):
[Client_MAC:capwap_900000004] 클라이언트에서 !AUTH_ABORT 게시
2023/04/12 16:51:45.762969 {wncd_x_R0-0}{1}: [dot1x] [19224]: (정보):
[Client_MAC:capwap_900000004] 재시작 상태 시작

이 프로세스는 루프로 진행되며 클라이언트는 인증 단계에서만 중단됩니다.

무선 컨트롤러에서 캡처된 내장형 패킷 캡처는 여러 액세스 요청 및 챌린지 교환이 1500바이트 미만의 MTU로 이루어진 후 무선 컨트롤러가 클라이언트의 인증서가 포함된 1500바이트를 초과하는 액세스 요청을 전송함을 보여줍니다. 이 큰 패킷은 프래그먼트화를 거칩니다. 그러나 이 특정 액세스 요청에 대한 응답은 없습니다. 무선 컨트롤러는 최대 재시도 횟수에 도달할 때까지 이 요청을 계속 재전송하며, 그 후 EAP-TLS 세션이 재시작됩니다. 이 이벤트 순서는 계속 반복되며, 클라이언트가 인증을 시도할 때 EAP-TLS 루프가 발생함을 나타냅니다. 아래에 제공된 무선 컨트롤러와 ISE에서 동시 패킷 캡처를 참조하여 더 명확하게 파악하십시오.

무선 컨트롤러 EPC:

radius.code == 1				
o.	Time	Protocol	Length	Info
109	12:21:27.510959	RADIUS	594	Access-Request id=3
110	12:21:27.510959	RADIUS	594	Access-Request id=3, Duplicate Request
117	12:21:27.554963	RADIUS	594	Access-Request id=4
118	12:21:27.554963	RADIUS	594	Access-Request id=4, Duplicate Request
125	12:21:27.599959	RADIUS	594	Access-Request id=5
126	12:21:27.599959	RADIUS	594	Access-Request id=5, Duplicate Request
135	12:21:27.640958	RADIUS	594	Access-Request id=6
136	12:21:27.640958	RADIUS	594	Access-Request id=6, Duplicate Request
143	12:21:27.676951	RADIUS	594	Access-Request id=7
144	12:21:27.676951	RADIUS	594	Access-Request id=7, Duplicate Request
154	12:21:27.758948	RADIUS	714	Access-Request id=8
796	12:21:32.759955	RADIUS	714	Access-Request id=8, Duplicate Request
1130	12:21:37.761954	RADIUS	714	Access-Request id=8, Duplicate Request
1868	12:21:42.762945	RADIUS	714	Access-Request id=8, Duplicate Request
2132	12:21:45.796955	RADIUS	538	Access-Request id=9
2133	12:21:45.796955	RADIUS	538	Access-Request id=9, Duplicate Request
2144	12:21:45.854951	RADIUS	760	Access-Request id=10
2145	12:21:45.854951	RADIUS	760	Access-Request id=10, Duplicate Request
2168	12:21:45.914945	RADIUS	594	Access-Request id=11
2169	12:21:45.914945	RADIUS	594	Access-Request id=11, Duplicate Request
2176	12:21:45.959941	RADIUS	594	Access-Request id=12

WLC의 패킷 캡처

무선 컨트롤러가 특정 액세스 요청 ID = 8에 대한 여러 개의 중복 요청을 전송하는 것을 확인합니다



참고: EPC에서 다른 ID에 대한 단일 중복 요청이 있음을 알 수 있습니다. 이렇게 하면 다음과 같은 질문이 표시됩니다. 그렇게 중복이 예상되는가? 이 중복이 예상되는지에 대한 대답은 그렇다, 이다. 그 이유는 'Monitor Control Plane' 옵션이 선택된 무선 컨트롤러의 GUI에서 캡처를 가져왔기 때문입니다. 따라서 RADIUS 패킷의 여러 인스턴스가 CPU로 전송되기 때문에 관찰하는 것이 일반적입니다. 이 경우 소스 및 대상 MAC 주소가 00:00:00으로 설정된 상태에서 액세스 요청을 확인해야 합니다.

No.	Time	Protocol	Length	Info
109	12:21:27.510959	RADIUS	594	Access-Request id=3
110	12:21:27.510959	RADIUS	594	Access-Request id=3, Duplicate Request
117	12:21:27.554963	RADIUS	594	Access-Request id=4
118	12:21:27.554963	RADIUS	594	Access-Request id=4, Duplicate Request

> Frame 109: 594 bytes on wire (4752 bits), 594 bytes captured (4752 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)

> Destination: 00:00:00_00:00:00 (00:00:00:00:00:00)

> Source: 00:00:00_00:00:00 (00:00:00:00:00:00)

Type: IPv4 (0x0800)

WLC의 CPU에 Radius Access-Request Punted

지정된 소스 및 대상 MAC 주소의 액세스 요청만 실제로 무선 컨트롤러에서 전송해야 합니다.

No.	Time	Protocol	Length	Info
109	12:21:27.510959	RADIUS	594	Access-Request id=3
110	12:21:27.510959	RADIUS	594	Access-Request id=3, Duplicate Request
117	12:21:27.554963	RADIUS	594	Access-Request id=4
118	12:21:27.554963	RADIUS	594	Access-Request id=4, Duplicate Request

> Frame 110: 594 bytes on wire (4752 bits), 594 bytes captured (4752 bits)

▼ Ethernet II, Src: Microsoft_ [REDACTED], Dst: [REDACTED]

- > Destination: 12:34:56:78:9a:bc (12:34:56:78:9a:bc)
- > Source: Microsoft_95:42:9e (00:22:48:95:42:9e)
- Type: IPv4 (0x0800)

AAA 서버로 전송된 RADIUS 액세스 요청

ID = 8로 식별되는 문제의 액세스 요청. 이 요청은 여러 번 전송되었으며 AAA 서버에서 응답이 표시되지 않았습니다. 추가 조사 결과, Access-request ID=8의 경우 MTU를 초과하는 크기로 인해 UDP 프래그먼트화가 발생하는 것으로 관찰되었습니다(아래 그림 참조).

147	12:21:27.683955	TLSv1.2	104	Server Hello, Certificate, Server Key Exchange, Certificate Request, Server Hello Done
148	12:21:27.683955	EAP	104	Request, TLS EAP (EAP-TLS)
149	12:21:27.756949	CAPWAP-Data	1450	CAPWAP-Data (Fragment ID: 50383, Fragment Offset: 0)
150	12:21:27.756949	EAP	188	Response, TLS EAP (EAP-TLS)
151	12:21:27.756949	EAP	1580	Response, TLS EAP (EAP-TLS)
152	12:21:27.758948	IPv4	1410	Fragmented IP protocol (proto=UDP 17, off=0, ID=b156) [Reassembled in #154]
153	12:21:27.758948	IPv4	1410	Fragmented IP protocol (proto=UDP 17, off=0, ID=b156) [Reassembled in #154]
154	12:21:27.758948	RADIUS	714	Access-Request id=8
155	12:21:27.758948	IPv4	714	Fragmented IP protocol (proto=UDP 17, off=1376, ID=b156)
156	12:21:28.084987	TLSv1.2	1070	Application Data

WLC 패킷 캡처에서 프래그먼트화가 발생

>	Frame 152: 1410 bytes on wire (11280 bits), 1410 bytes captured (11280 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)
>	Destination: 00:00:00_00:00:00 (00:00:00:00:00:00)
>	Source: 00:00:00_00:00:00 (00:00:00:00:00:00)
	Type: IPv4 (0x0800)
▼	Internet Protocol Version 4, Src: 10.100.9.15, Dst: 172.16.26.235
	0100 = Version: 4
	 0101 = Header Length: 20 bytes (5)
>	Differentiated Services Field: 0x00 (DSCP: CS0, ECN: Not-ECT)
	Total Length: 1396
	Identification: 0xb156 (45398)
>	001. = Flags: 0x1, More fragments
	...0 0000 0000 0000 = Fragment Offset: 0
	Time to Live: 64
	Protocol: UDP (17)
	Header Checksum: 0xc9b4 [validation disabled]
	[Header checksum status: Unverified]
	Source Address: 10.100.9.15
	Destination Address: 172.16.26.235
	[Reassembled IPv4 in frame: 154]
>	Data (1376 bytes)

조각화된 패킷 - I

```

> Frame 153: 1410 bytes on wire (11280 bits), 1410 bytes captured (11280 bits)
v Ethernet II, Src: Microsoft_ [REDACTED], Dst: [REDACTED]
  > Destination: 12:34:56:78:9a:bc (12:34:56:78:9a:bc)
  > Source: Microsoft_95:42:9e (00:22:48:95:42:9e)
  Type: IPv4 (0x0800)
v Internet Protocol Version 4, Src: 10.100.9.15, Dst: 172.16.26.235
  0100 .... = Version: 4
  .... 0101 = Header Length: 20 bytes (5)
  > Differentiated Services Field: 0x00 (DSCP: CS0, ECN: Not-ECT)
  Total Length: 1396
  Identification: 0xb156 (45398)
  > 001. .... = Flags: 0x1, More fragments
  ...0 0000 0000 0000 = Fragment Offset: 0
  Time to Live: 64
  Protocol: UDP (17)
  Header Checksum: 0xc9b4 [validation disabled]
  [Header checksum status: Unverified]
  Source Address: 10.100.9.15
  Destination Address: 172.16.26.235
  [Reassembled IPv4 in frame: 154]

```

조각화된 패킷 - II

152	12:21:27.758948	IPv4	1410	Fragmented IP protocol (proto=UDP 17, off=0, ID=b156) [Reassembled in #154]
153	12:21:27.758948	IPv4	1410	Fragmented IP protocol (proto=UDP 17, off=0, ID=b156) [Reassembled in #154]
154	12:21:27.758948	RADIUS	714	Access-Request id=8
155	12:21:27.758948	IPv4	714	Fragmented IP protocol (proto=UDP 17, off=1376, ID=b156)

```

> Frame 154: 714 bytes on wire (5712 bits), 714 bytes captured (5712 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)
v Internet Protocol Version 4, Src: 10.100.9.15, Dst: 172.16.26.235
  0100 .... = Version: 4
  .... 0101 = Header Length: 20 bytes (5)
  > Differentiated Services Field: 0x00 (DSCP: CS0, ECN: Not-ECT)
  Total Length: 700
  Identification: 0xb156 (45398)
  > 000. .... = Flags: 0x0
  ...0 0000 1010 1100 = Fragment Offset: 1376
  Time to Live: 64
  Protocol: UDP (17)
  Header Checksum: 0xebc0 [validation disabled]
  [Header checksum status: Unverified]
  Source Address: 10.100.9.15
  Destination Address: 172.16.26.235
v [3 IPv4 Fragments (2056 bytes): #152(1376), #153(1376), #154(680)]
  [Frame: 152, payload: 0-1375 (1376 bytes)]
  > [Frame: 153, payload: 0-1375 (1376 bytes)]
  [Frame: 154, payload: 1376-2055 (680 bytes)]
  [Fragment count: 3]
  [Reassembled IPv4 length: 2056]

```

리어셈블된 패킷

교차 검증을 위해 ISE 로그를 검토했으며 무선 컨트롤러에서 프래그먼트화된 액세스 요청이 ISE에 의해 전혀 수신되지 않음을 발견했습니다.

ISE TCP 덤프

radius.code == 1

No.	Time	Protocol	Length	Info
1	12:21:27.387158	RADIUS	538	Access-Request id=0
3	12:21:27.428304	RADIUS	760	Access-Request id=1
5	12:21:27.492019	RADIUS	594	Access-Request id=2
7	12:21:27.527949	RADIUS	594	Access-Request id=3
9	12:21:27.572272	RADIUS	594	Access-Request id=4
11	12:21:27.617147	RADIUS	594	Access-Request id=5
13	12:21:27.657917	RADIUS	594	Access-Request id=6
15	12:21:27.694381	RADIUS	594	Access-Request id=7
17	12:21:45.814195	RADIUS	538	Access-Request id=9
19	12:21:45.871163	RADIUS	760	Access-Request id=10
21	12:21:45.932076	RADIUS	594	Access-Request id=11
23	12:21:45.977012	RADIUS	594	Access-Request id=12
25	12:21:46.018562	RADIUS	594	Access-Request id=13

ISE 엔드에서의 캡처

분석을 통한 Azure Side Capture:

Azure 팀이 Azure 내 물리적 호스트에서 캡처를 수행했습니다. Azure 호스트 내의 vSwitch에서 캡처된 데이터는 UDP 패킷이 순서를 벗어나서 도착하고 있음을 나타냅니다. 이러한 UDP 조각의 순서가 올바르지 않으므로 Azure에서 해당 조각을 삭제합니다. 아래는 액세스 요청 ID = 255에 대해 Azure End 및 Wireless Controller에서 동시에 캡처한 것이며, 여기서 패킷 오류가 명확하게 드러납니다.

무선 컨트롤러의 EPC(Encapsulated Packet Capture)는 조각화된 패킷이 무선 컨트롤러에서 나가는 순서를 표시합니다.

The image shows a Wireshark packet capture window titled "0619_AURAUENRAWLC01_source.pcap". The filter bar shows "ip.addr == 10.100.9.15 && ip.addr == 172.16.26.235 && (ip.id == 33004 or ip.id == 35253)". The packet list shows several fragmented IP packets (protocol 17, offset 0) and RADIUS access requests. A red arrow points to the "Access-Request id=255" packet (No. 5631). Below the packet list, a red text box states: "VM Capture will see packet length of 1410 is sent first, then 696". The packet details pane for frame 5721 shows the arrival time as "Jun 19, 2023 12:09:10.029997000 AUS Eastern Standard Time".

WLC의 조각화된 패킷 시퀀스

물리적 호스트에서 패킷이 올바른 순서로 도착하지 않습니다

No.	Absolute Time	Source	Destination	Protocol	Identification	Length	Sequence Number	Info
276	12:09:13.810263	10.100.9.15	172.16.26.235	IPv4	0x80ec (33004)	696		Fragmented IP protocol (proto=UDP 17, off=1376, ID=80ec)
277	12:09:13.810264	10.100.9.15	172.16.26.235	RADIUS	0x80ec (33004)	1410		Access-Request id=255[BoundErrorUnresembled Packet]
278	12:09:13.810306	10.100.9.15	172.16.26.235	RADIUS	0x81ec (33260), 0x80ec (33004)	1460		Access-Request id=255, Duplicate Request[BoundErrorUnresembled Packet]
384	12:09:18.810390	10.100.9.15	172.16.26.235	IPv4	0x8905 (35253)	696		Fragmented IP protocol (proto=UDP 17, off=1376, ID=8905)
385	12:09:18.810391	10.100.9.15	172.16.26.235	RADIUS	0x8905 (35253)	1410		Access-Request id=255, Duplicate Request[BoundErrorUnresembled Packet]
386	12:09:18.810449	10.100.9.15	172.16.26.235	RADIUS	0x8ab5 (35509), 0x8905 (35253)	1460		Access-Request id=255, Duplicate Request[BoundErrorUnresembled Packet]

Physical host will always see 696 packet first and then 1410 packet length. Packet with length 696 will not leave the physical host

Packet comments

- > Frame 276: 696 bytes on wire (5568 bits), 256 bytes captured (2048 bits) on interface vNIC:Synthetic:05b0f752-3346-49ba-86ef-47b1ec206bc2:11:0, id 5 (outbound)
- > Ethernet II, Src: Microsof..., Dst: 1...
- > Internet Protocol Version 4, Src: 10.100.9.15, Dst: 172.16.26.235
- > Data (222 bytes)

Azure End의 캡처

패킷이 잘못된 순서로 도착하고 물리적 노드가 비순차적 프레임을 거부하도록 프로그래밍되므로 패킷이 즉시 삭제됩니다. 이 중단으로 인해 인증 프로세스가 실패하고 클라이언트가 인증 단계를 초과하여 진행할 수 없게 됩니다.

무선 컨트롤러 쪽에서 제안하는 해결 방법:

버전 17.11.1부터 Radius/AAA 패킷의 점보 프레임 지원을 구현하고 있습니다. 이 기능을 사용하면 다음 컨피그레이션이 컨트롤러에 설정되어 있는 경우 c9800 컨트롤러에서 AAA 패킷의 단편화를 방지할 수 있습니다. 이러한 패킷의 단편화를 완전히 방지하려면 AAA 서버를 비롯한 모든 네트워크 홉이 점보 프레임 패킷과 호환되는지 확인해야 합니다. ISE의 경우 점보 프레임 지원은 버전 3.1 이상부터 시작합니다.

무선 컨트롤러의 인터페이스 구성:

```
C9800-CL(config)#interface
```

```
C9800-CL(config-if) # mtu
```

```
C9800-CL(config-if) # ip mtu
```

```
[1500 to 9000]
```

무선 컨트롤러의 AAA 서버 구성:

```
C9800-CL(config)# aaa group server radius
```

```
C9800-CL(config-sg-radius) # server name
```

```
C9800-CL(config-sg-radius) # ip radius source-interface
```

다음은 WLC(Wireless LAN Controller)에서 MTU(Maximum Transmission Unit)가 3000바이트로 구성된 경우의 Radius 패킷에 대한 간략한 설명입니다. 3000바이트보다 작은 패킷은 단편화 없이 원활하게 전송되었습니다.

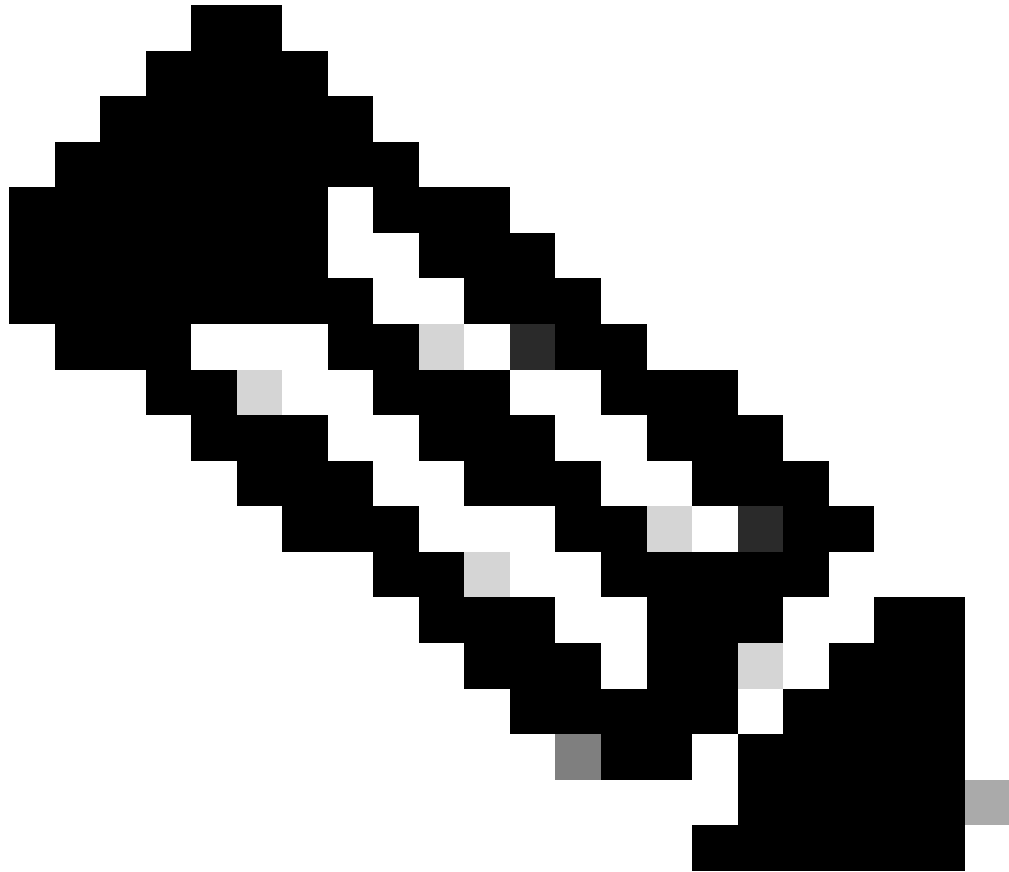
1020	10:08:11.177984	RADIUS	2075	Access-Request id=199
1021	10:08:11.177984	RADIUS	2075	Access-Request id=199, Duplicate Request
1119	10:08:16.194981	RADIUS	2075	Access-Request id=199, Duplicate Request
1120	10:08:16.194981	RADIUS	2075	Access-Request id=199, Duplicate Request
1223	10:08:21.179983	RADIUS	2075	Access-Request id=199, Duplicate Request
1224	10:08:21.179983	RADIUS	2075	Access-Request id=199, Duplicate Request
1451	10:08:26.180990	RADIUS	2075	Access-Request id=199, Duplicate Request
1452	10:08:26.180990	RADIUS	2075	Access-Request id=199, Duplicate Request
2470	10:08:31.181982	RADIUS	2075	Access-Request id=199, Duplicate Request

MTU가 증가한 WLC의 패킷 캡처

무선 컨트롤러는 이러한 방식으로 컨피그레이션을 설정하여 패킷을 단편화하지 않고 그대로 전송합니다. 그러나 Azure 클라우드는 점보 프레임 지원하지 않으므로 이 솔루션을 구현할 수 없습니다.

해결책:

- 무선 컨트롤러의 EPC(Encapsulated Packet Capture)에서 패킷이 올바른 순서로 전송되는 것을 확인했습니다. 그런 다음 수신 호스트가 적절히 리어셈블하고 처리를 계속해야 하며, 이 경우 Azure 측에서 발생하지 않습니다.
- 순서가 잘못된 UDP 패킷의 문제를 해결하려면 `enable-udp-fragment-reordering` Azure에서 옵션을 활성화해야 합니다.
- 이 문제에 대한 도움을 받으려면 Azure 지원 팀에 문의해야 합니다. Microsoft에서 이 문제를 확인했습니다.



참고: 이 문제는 WLC(Wireless LAN Controller)에만 국한되지 않습니다. ISE, Forti Authenticator 및 RTSP 서버를 비롯한 여러 RADIUS 서버에서, 특히 Azure 환경에서 작동할 때 순서가 잘못된 UDP 패킷과 유사한 문제가 발생했습니다.

이 번역에 관하여

Cisco는 전 세계 사용자에게 다양한 언어로 지원 콘텐츠를 제공하기 위해 기계 번역 기술과 수작업 번역을 병행하여 이 문서를 번역했습니다. 아무리 품질이 높은 기계 번역이라도 전문 번역가의 번역 결과물만큼 정확하지는 않습니다. Cisco Systems, Inc.는 이 같은 번역에 대해 어떠한 책임도 지지 않으며 항상 원본 영문 문서(링크 제공됨)를 참조할 것을 권장합니다.