Résolution des problèmes de fragmentation : Affectation du contrôleur sans fil c9800 avec Azure

Table des matières

Introduction

Symptômes

Erreur sur le serveur ISE

Analyse détaillée du journal :

- Contrôleur sans fil EPC :
- Vidages TCP ISE
- Azure Side Capture avec analyse :

Solution suggérée du côté du contrôleur sans fil :

Solution :

Introduction

Ce document décrit un problème connu avec la plate-forme Azure qui entraîne la perte de paquets en raison d'une mauvaise gestion des fragments hors séquence.

Symptômes

Produits affectés : Contrôleur sans fil Catalyst 9800-CL hébergé sur Azure ou Identity Service Engine hébergé sur Azure.

Configuration SSID : Configuré pour 802.1x EAP-TLS avec authentification centrale.

Conduite : Lors de l'utilisation du 9800-CL hébergé sur la plate-forme Azure avec un SSID basé sur EAP-TLS, vous pouvez rencontrer des problèmes de connectivité. Les clients peuvent rencontrer des difficultés pendant la phase d'authentification.

Erreur sur le serveur ISE

Code d'erreur 5411 indiquant que le demandeur a cessé de communiquer avec ISE pendant l'échange de certificats EAP-TLS.

Analyse détaillée du journal :

Voici une illustration de l'une des configurations concernées : Dans le contrôleur sans fil 9800, le SSID est configuré pour 802.1x et le serveur AAA est configuré pour EAP-TLS. Lorsqu'un client tente une authentification, en particulier pendant la phase d'échange de certificats, il envoie un certificat qui dépasse la taille maximale de l'unité de transmission (MTU) sur le contrôleur sans fil. Le contrôleur sans fil 9800 fragmente ensuite ce paquet volumineux et envoie les fragments séquentiellement au serveur AAA. Cependant, ces fragments n'arrivent pas dans le bon ordre sur l'hôte physique, ce qui entraîne l'abandon des paquets.

Voici les traces d'annonce de routeur du contrôleur sans fil lorsque le client tente de se connecter : Le client entre dans l'état d'authentification L2 et le processus EAP est démarré

```
2023/04/12 16:51:27.606414 {wncd_x_R0-0}{1} : [dot1x] [19224] : (info) :
[Client_MAC:capwap_90000004] Saisie de l'état de la demande
2023/04/12 16:51:27.606425 {wncd_x_R0-0}{1} : [dot1x] [19224] : (info) :
[0000.0000.0000:capwap_90000004] Envoi du paquet EAPOL
2023/04/12 16:51:27.606494 {wncd_x_R0-0}{1} : [dot1x] [19224] : (info) :
[Client_MAC:capwap_90000004] Paquet EAPOL envoyé - Version : 3,Type
EAPOL : EAP, longueur de la charge utile : 1008, EAP-Type = EAP-TLS
2023/04/12 16:51:27.606496 {wncd_x_R0-0}{1} : [dot1x] [19224] : (info) :
[Client_MAC:capwap_90000004] Paquet EAP - REQUEST, ID : 0x25
2023/04/12 16:51:27.606536 {wncd_x_R0-0}{1} : [dot1x] [19224] : (info) :
[Client_MAC:capwap_90000004] Paquet EAPOL envoyé au client
2023/04/12 16:51:27.640768 {wncd_x_R0-0}{1} : [dot1x] [19224] : (info) :
[Client_MAC:capwap_90000004] Paquet EAPOL reçu - Version : 1,Type EAPOL
: EAP, longueur de la charge utile : 6, EAP-Type = EAP-TLS
2023/04/12 16:51:27.640781 {wncd_x_R0-0}{1} : [dot1x] [19224] : (info) :
[Client_MAC:capwap_90000004] Paquet EAP - RÉPONSE, ID : 0x25
```

Lorsque le contrôleur sans fil envoie la demande d'accès au serveur AAA et que la taille du paquet est inférieure à 1 500 octets (ce qui est la MTU par défaut sur le contrôleur sans fil), la demande d'accès est reçue sans aucune complication.

```
2023/04/12 16:51:27.641094 {wncd_x_R0-0}{1} : [rayon] [19224] : (info) :
RADIUS: Envoyer la demande d'accès à 172.16.26.235:1812 id 0/6, len 552
2023/04/12 16:51:27.644693 {wncd_x_R0-0}{1} : [rayon] [19224] : (info) :
RADIUS: Reçu de id 1812/6 172.16.26.235:0, Access-Challenge, len 1141
```

Parfois, un client peut envoyer son certificat pour authentification. Si la taille du paquet dépasse la MTU, il sera fragmenté avant d'être envoyé.

```
2023/04/12 16:51:27.758366 {wncd_x_R0-0}{1} : [rayon] [19224] : (info) :
RADIUS: Envoyer la demande d'accès à 172.16.26.235:1812 id 0/8, len 2048
2023/04/12 16:51:37.761885 {wncd_x_R0-0}{1} : [rayon] [19224] : (info) :
RADIUS: Délai d'attente de 5 secondes démarré
2023/04/12 16:51:42.762096 {wncd_x_R0-0}{1} : [rayon] [19224] : (info) :
RADIUS: Retransmettre à (172.16.26.235:1812,1813) pour l'ID 0/8
```

```
2023/04/12 16:51:32.759255 {wncd_x_R0-0}{1} : [rayon] [19224] : (info) :
RADIUS: Retransmettre à (172.16.26.235:1812,1813) pour l'ID 0/8
2023/04/12 16:51:32.760328 {wncd_x_R0-0}{1} : [rayon] [19224] : (info) :
RADIUS: Délai d'attente de 5 secondes démarré
2023/04/12 16:51:37.760552 {wncd_x_R0-0}{1} : [rayon] [19224] : (info) :
RADIUS: Retransmettre à (172.16.26.235:1812,1813) pour l'ID 0/8
2023/04/12 16:51:42.762096 {wncd_x_R0-0}{1} : [rayon] [19224] : (info) :
RADIUS: Retransmettre à (172.16.26.235:1812,1813) pour l'ID 0/8
```

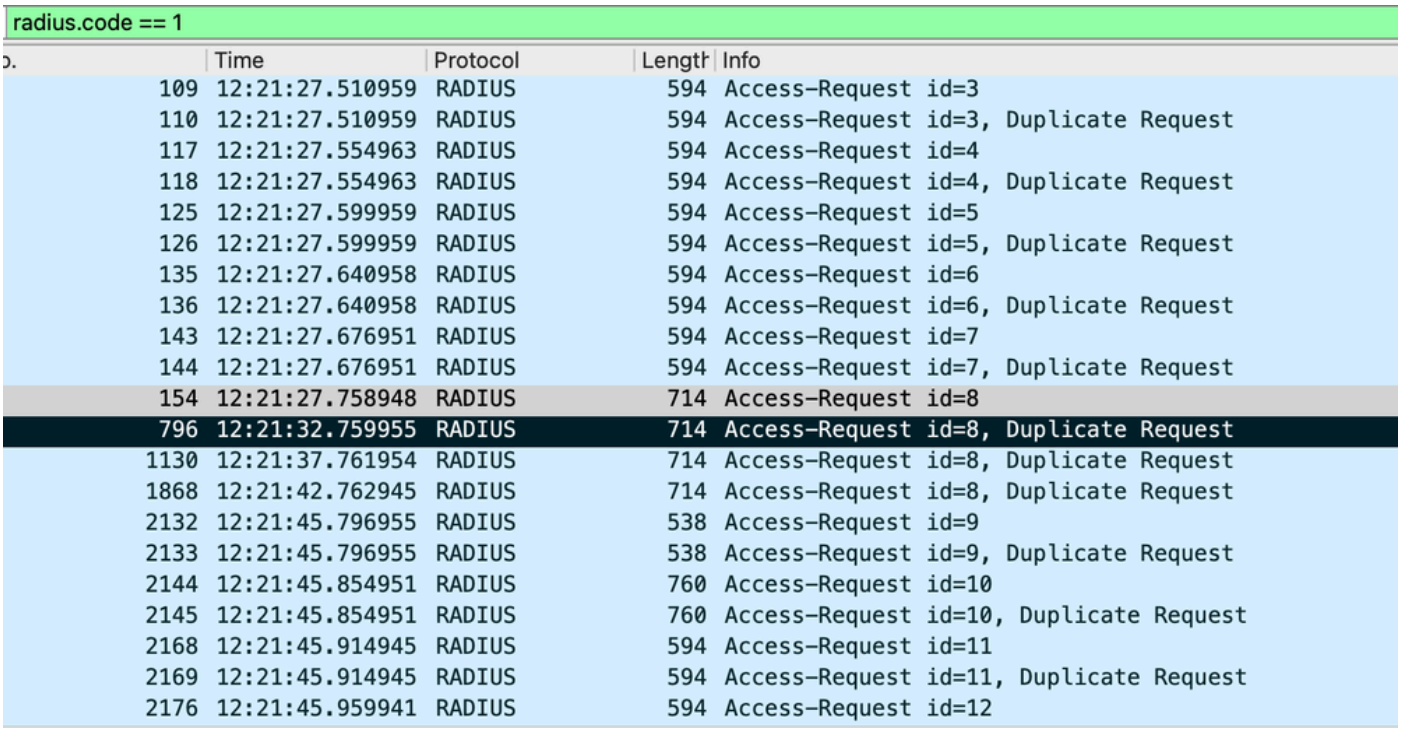
Nous avons remarqué que la taille du paquet est 2048, ce qui dépasse la MTU par défaut. Par conséquent, il n'y a pas eu de réponse du serveur AAA. Le contrôleur sans fil renvoie la demande d'accès de façon permanente jusqu'à ce qu'il atteigne le nombre maximal de tentatives. En l'absence de réponse, le contrôleur sans fil réinitialise finalement le processus EAPOL.

```
2023/04/12 16:51:45.762890 {wncd_x_R0-0}{1} : [dot1x] [19224] : (info) :
[Client_MAC : capwap_90000004] Publication d'EAPOL_START sur le client
2023/04/12 16:51:45.762956 {wncd_x_R0-0}{1} : [dot1x] [19224] : (info) :
[Client_MAC:capwap_90000004] Saisie de l'état d'initialisation
2023/04/12 16:51:45.762965 {wncd_x_R0-0}{1} : [dot1x] [19224] : (info) :
[Client_MAC : capwap_90000004] Publication de !AUTH_ABORT sur le client
2023/04/12 16:51:45.762969 {wncd_x_R0-0}{1} : [dot1x] [19224] : (info) :
[Client_MAC:capwap_90000004] Passage à l'état de redémarrage
```

Ce processus se déroule en boucle et le client est bloqué uniquement en phase d'authentification.

La capture de paquets intégrée capturée sur le contrôleur sans fil montre qu'après plusieurs demandes d'accès et échanges de demande de confirmation avec un MTU inférieur à 1500 octets, le contrôleur sans fil envoie une demande d'accès supérieure à 1500 octets, qui contient le certificat du client. Ce paquet plus volumineux subit une fragmentation. Cependant, il n'y a pas de réponse à cette demande d'accès particulière. Le contrôleur sans fil continue de renvoyer cette requête jusqu'à ce qu'il atteigne le nombre maximal de tentatives, après quoi la session EAP-TLS redémarre. Cette séquence d'événements se répète constamment, ce qui indique qu'une boucle EAP-TLS se produit lorsque le client tente de s'authentifier. Reportez-vous aux captures de paquets simultanées du contrôleur sans fil et de l'ISE fournies ci-dessous pour une meilleure compréhension.

Contrôleur sans fil EPC :



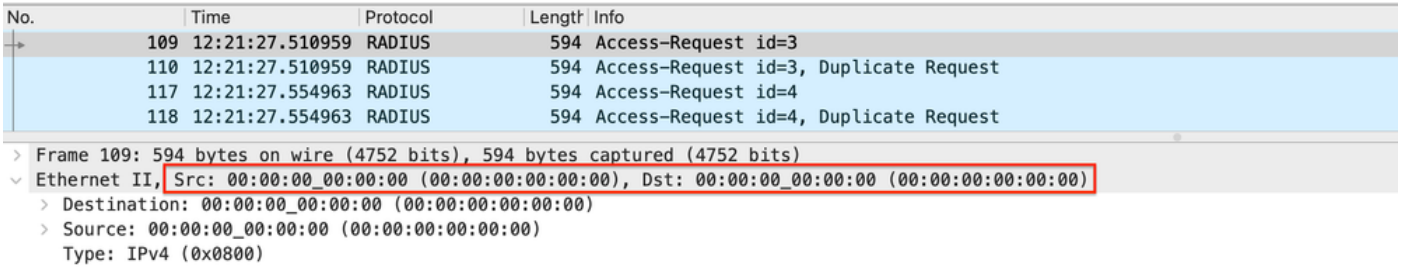
radius.code == 1

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
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

Capture de paquets sur WLC

Nous observons que le contrôleur sans fil envoie plusieurs demandes en double pour un ID de demande d'accès particulier = 8

Remarque : Sur l'EPC, nous remarquons également qu'il existe une seule demande dupliquée pour d'autres ID. La question suivante se pose : Un tel dédoublement est-il prévu ? La réponse à la question de savoir si cette duplication est attendue est oui, elle l'est. La raison en est que la capture a été effectuée à partir de l'interface utilisateur graphique du contrôleur sans fil avec l'option « Monitor Control Plane » sélectionnée. Par conséquent, il est normal d'observer plusieurs instances de paquets RADIUS puisqu'ils sont dirigés vers le processeur. Dans ce cas, les demandes d'accès doivent être vues avec les adresses MAC source et de destination définies sur 00:00:00.

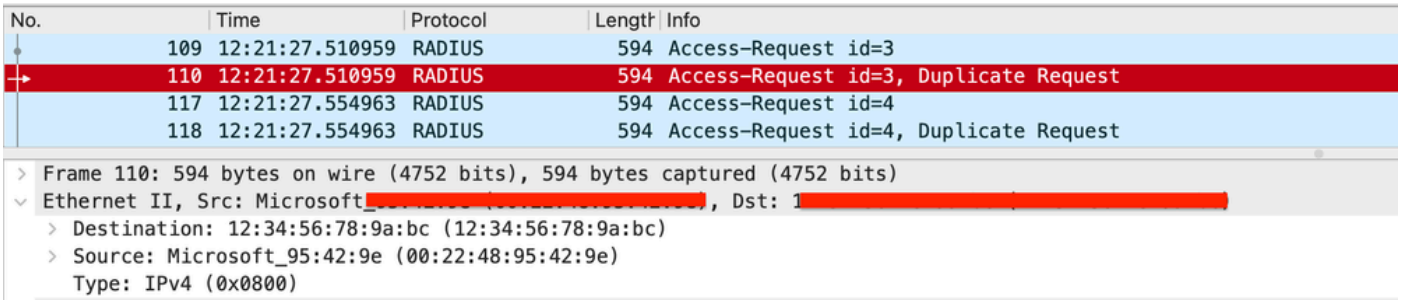


No.	Time	Protocol	Length	Info
109	12:21:27.510959	RADIUS	594	Access-Request id=3
110	12:21:27.510959	RADIUS	594	Access-Request id=3, Duplicate Request
117	12:21:27.554963	RADIUS	594	Access-Request id=4
118	12:21:27.554963	RADIUS	594	Access-Request id=4, Duplicate Request

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

Demande d'accès Radius dirigée vers le processeur sur le WLC

Seules les demandes d'accès avec les adresses MAC source et de destination spécifiées doivent réellement être envoyées à partir du contrôleur sans fil.

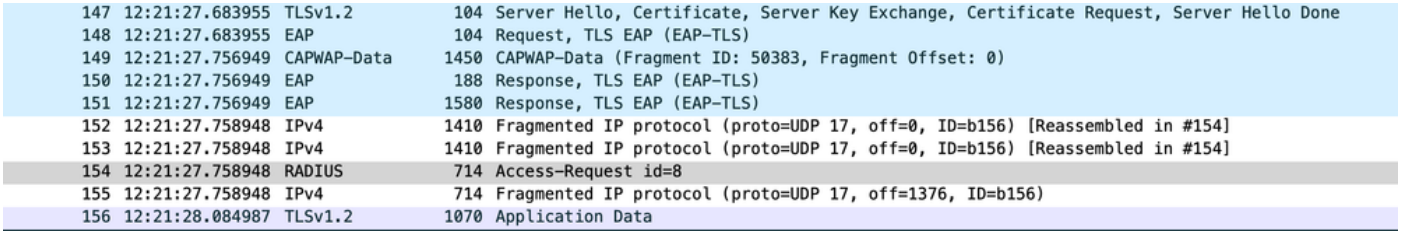


No.	Time	Protocol	Length	Info
109	12:21:27.510959	RADIUS	594	Access-Request id=3
110	12:21:27.510959	RADIUS	594	Access-Request id=3, Duplicate Request
117	12:21:27.554963	RADIUS	594	Access-Request id=4
118	12:21:27.554963	RADIUS	594	Access-Request id=4, Duplicate Request

```
> Frame 110: 594 bytes on wire (4752 bits), 594 bytes captured (4752 bits)
v Ethernet II, Src: Microsoft[redacted], Dst: 1[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)
```

Demande d'accès Radius envoyée au serveur AAA

Les demandes d'accès en question, identifiées par ID = 8, qui sont envoyées plusieurs fois et pour lesquelles aucune réponse n'a été vue du serveur AAA. Après un examen plus approfondi, nous avons observé que pour l'ID de demande d'accès = 8, la fragmentation UDP se produit en raison de la taille dépassant la MTU, comme illustré ci-dessous :



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

Fragmentation en cours sur la capture de paquets WLC

```
> Frame 152: 1410 bytes on wire (11280 bits), 1410 bytes captured (11280 bits)
v 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)
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]
> Data (1376 bytes)
```

Paquet fragmenté - 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]
```

Paquet fragmenté - 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]
```

Paquet Réassemblé

Pour effectuer une vérification croisée, nous avons examiné les journaux ISE et découvert que la demande d'accès, qui avait été fragmentée sur le contrôleur sans fil, n'était pas reçue par l'ISE du tout.

Vidages TCP ISE

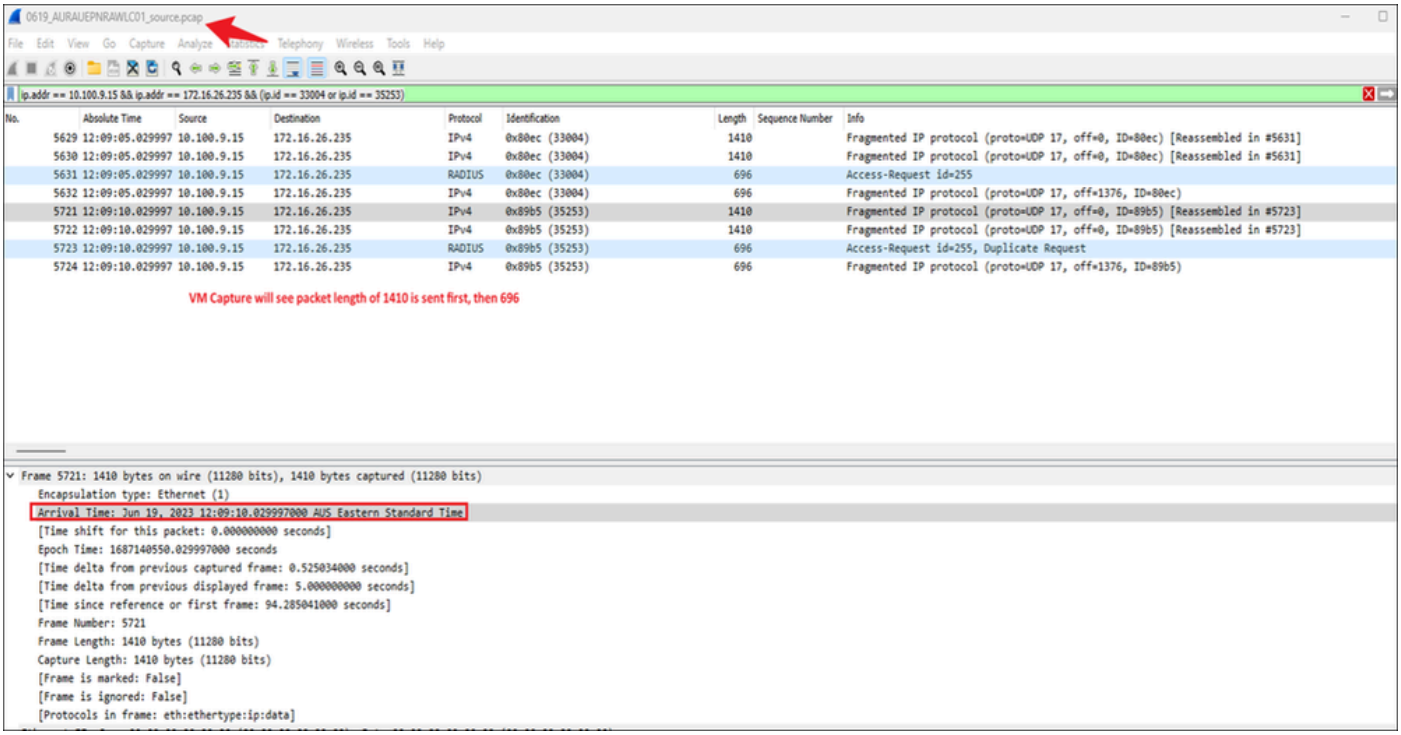
radius.code == 1

o.	Time	Protocol	Lengtr	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

Captures du côté ISE

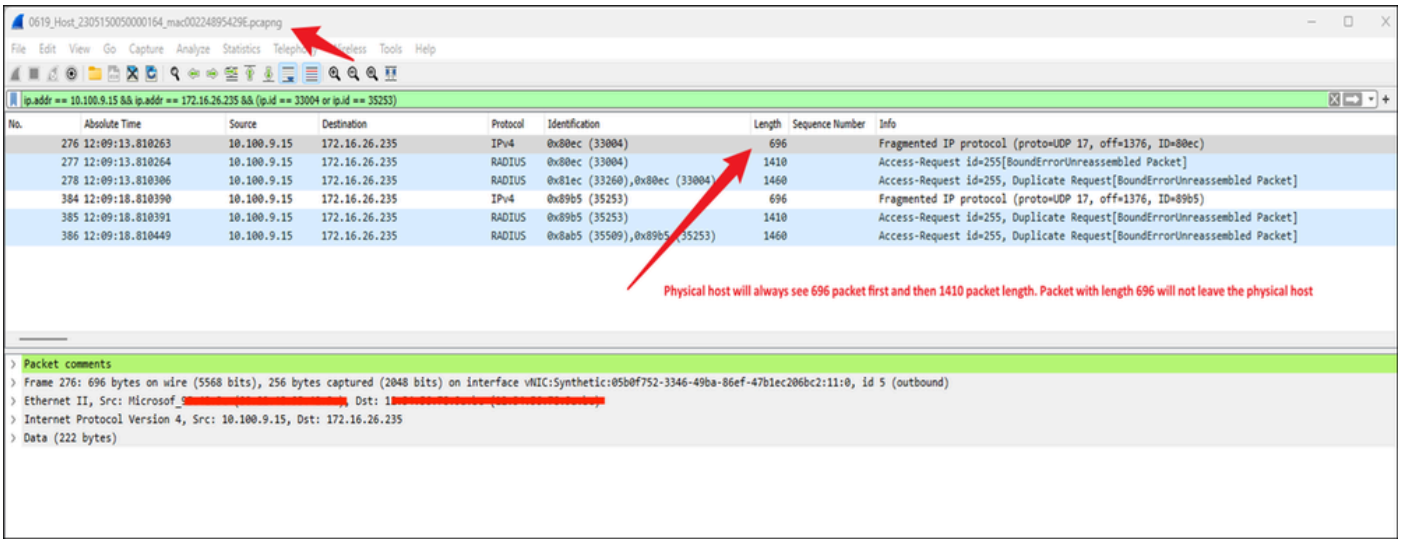
Azure Side Capture avec analyse :

L'équipe Azure a effectué une capture sur l'hôte physique dans Azure. Les données capturées sur le commutateur virtuel au sein de l'hôte Azure indiquent que les paquets UDP arrivent hors séquence. Étant donné que ces fragments UDP ne sont pas dans le bon ordre, Azure les rejette. Voici les captures de l'extrémité Azure et du contrôleur sans fil, prises simultanément pour l'ID de demande d'accès = 255, où le problème de désordre des paquets est clairement évident :
La capture de paquets encapsulés (EPC) sur le contrôleur sans fil affiche la séquence dans laquelle les paquets fragmentés quittent le contrôleur sans fil.



Séquence de paquets fragmentés sur le WLC

Sur l'hôte physique, les paquets n'arrivent pas dans l'ordre approprié



Captures sur Azure End

Comme les paquets arrivent dans le mauvais ordre et que le noeud physique est programmé pour rejeter toutes les trames désordonnées, les paquets sont immédiatement abandonnés. Cette interruption entraîne l'échec du processus d'authentification, ce qui empêche le client de passer au-delà de la phase d'authentification.

Solution suggérée du côté du contrôleur sans fil :

À partir de la version 17.11.1, nous implémentons la prise en charge des trames Jumbo dans les paquets Radius/AAA. Cette fonctionnalité permet au contrôleur c9800 d'éviter la fragmentation des paquets AAA, à condition que la configuration suivante soit définie sur le contrôleur. Veuillez noter que pour éviter la fragmentation complète de ces paquets, il est essentiel de s'assurer que chaque saut de réseau, y compris le serveur AAA, est compatible avec les paquets de trame Jumbo. Pour ISE, la prise en charge des trames Jumbo commence avec la version 3.1.
Configuration d'interface sur le contrôleur sans fil :

C9800-CL(config)#interface

C9800-CL(config-if) # mtu

C9800-CL(config-if) # ip mtu

[1500 to 9000]

Configuration du serveur AAA sur le contrôleur sans fil :

C9800-CL(config)# aaa group server radius

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

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

Voici un bref aperçu d'un paquet Radius lorsque le MTU (Maximum Transmission Unit) est configuré sur 3000 octets sur un contrôleur LAN sans fil (WLC). Les paquets de moins de 3 000 octets ont été envoyés de manière transparente sans nécessiter de fragmentation :



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

Capture de paquets sur WLC avec MTU augmenté

En définissant la configuration de cette manière, le contrôleur sans fil transmet les paquets sans les fragmenter, en les envoyant intacts. Toutefois, comme le cloud Azure ne prend pas en charge les trames Jumbo, cette solution ne peut pas être implémentée.

Solution :

- À partir de l'EPC (Encapsulated Packet Capture) du contrôleur sans fil, nous avons observé que les paquets sont envoyés dans le bon ordre. Il incombe alors à l'hôte récepteur de les réassembler correctement et de poursuivre le traitement, ce qui, dans ce cas, ne se produit pas du côté Azure.
- Pour résoudre le problème des paquets UDP dans le désordre, l'option doit être activée sur Azure, enable-udp-fragment-reordering.
- Vous devez contacter l'équipe d'assistance Azure pour obtenir de l'aide sur ce sujet. Microsoft a reconnu ce problème.



Remarque : Il convient de noter que ce problème n'est pas exclusif à l'interopérabilité avec le contrôleur LAN sans fil (WLC) Cisco 9800. Des problèmes similaires avec des paquets UDP désordonnés ont été rencontrés sur différents serveurs RADIUS, y compris les serveurs ISE, Forti Authenticator et RTSP, en particulier lorsqu'ils fonctionnent dans l'environnement Azure.

À propos de cette traduction
Cisco a traduit ce document en traduction automatisée vérifiée par une personne dans le cadre d'un service mondial permettant à nos utilisateurs d'obtenir le contenu d'assistance dans leur propre langue.

Il convient cependant de noter que même la meilleure traduction automatisée ne sera pas aussi précise que celle fournie par un traducteur professionnel.