Comprensión de CAPWAP AP PMTU Discovery

Contenido

Introducción

Este documento describe el mecanismo de detección de la Unidad de transmisión máxima (PMTU) de la trayectoria del punto de acceso CAPWAP en IOS® XE y COS, problemas y resolución.

Escenario y alcance

Normalmente, se observan problemas de PMTU cuando un punto de acceso CAPWAP (AP) de un sitio remoto se registra en un controlador de LAN inalámbrica (WLC) a través de una WAN, especialmente cuando la ruta incluye VPN, GRE o cualquier segmento de red con una MTU inferior a los 1500 bytes estándar.

También examinamos la autenticación mediante el protocolo de autenticación extensible Transport Layer Security (EAP-TLS). Debido a que EAP-TLS intercambia certificados de gran tamaño, una MTU de ruta reducida aumenta el riesgo de fragmentación.

Todos los registros se capturaron en la versión de código 17.9.3. Los resultados se truncan para mostrar sólo las líneas relevantes.

Control CAPWAP frente a datos (qué se negocia)

Control CAPWAP:
El canal de control maneja mensajes de administración críticos como solicitudes de unión, intercambios de configuración y señales de señal de mantenimiento. Estos mensajes se protegen

mediante DTLS y son el objetivo principal del proceso de negociación de MTU de ruta (PMTU) para garantizar una comunicación del plano de control fiable y eficiente.

Datos CAPWAP:
Este canal transporta tráfico de cliente encapsulado, normalmente también protegido por DTLS en la mayoría de las implementaciones. Mientras que la negociación de PMTU ocurre en el canal de control, los valores de PMTU resultantes determinan indirectamente el tamaño máximo del paquete para la encapsulación del plano de datos, lo que afecta a la confiabilidad y fragmentación de la transmisión de datos del cliente.

Examples

- Paquetes de control: Unir solicitudes y respuestas, actualizaciones de configuración y mensajes de eco/keepalive.
- Paquetes de datos: Tramas de cliente encapsuladas transmitidas entre el punto de acceso (AP) y el controlador de LAN inalámbrica (WLC).

Hechos: Paquete CAPWAP de tamaño máximo

IOS AP (ejemplo)

Tamaño de paquete de PMTU enviado: 1499 bytes = Ethernet + PMTU CAPWAP

- Ethernet = 14 bytes
- PMTU CAPWAP = 1485 bytes
    - IP externa = 20 bytes
    - UDP = 25 bytes
    - DTLS = 1440 bytes

AP-COS (ejemplo)

Tamaño de paquete de PMTU enviado: 1483 bytes = Ethernet + PMTU CAPWAP

- Ethernet = 14 bytes
- PMTU CAPWAP = 1469 bytes
    - IP externa = 20 bytes
    - UDP = 25 bytes
    - DTLS = 1424 bytes

Comprobaciones de PMTU en tres fases

Ambas plataformas sondean tres valores PMTU codificados de forma rígida: 576, 1005 y 1485. La diferencia es cómo cada plataforma cuenta el encabezado Ethernet:

- Los AP IOS no incluyen el encabezado Ethernet en los valores 576/1005/1485.
    - Trama total = Ethernet (14) + PMTU (576/1005/1485) ⇒ 590, 1019, 1499 bytes (tamaño de cable).

- AP-COS incluye el encabezado Ethernet en los valores 576/1005/1485.
    - Trama total = PMTU (ya incluye Ethernet). Estos paquetes son 14 bytes más pequeños en el cable que los equivalentes de IOS AP.

Mecanismo de detección de PMTU CAPWAP

Comportamiento de IOS AP

Fase de unión de PA

Durante la unión CAPWAP, el AP negocia una PMTU CAPWAP máxima de 1485 bytes con el bit DF configurado. Espera 5 segundos por una respuesta.

- Si no hay respuesta o llega un ICMP "Fragmentation Needed", el AP retrocede a 576 bytes para completar la unión rápidamente, luego intenta elevar la PMTU después de que alcanza RUN.

Captura de paquetes (ejemplo)

Número de paquete 106 Verá una sonda de 1499 bytes (conjunto DF). Ninguna respuesta del mismo tamaño indica que el paquete no pudo atravesar la trayectoria sin fragmentación. A continuación, verá ICMP "Fragmentation Needed" (Se necesita fragmentación).

17	07:41:47.427848	0.002187	10.201.166.185	10.201.234.34	CAPWAP-Cont…	264	Set	CAPWAP-Control - Discovery Request[Malformed Packet]
88	07:42:45.435367	58.0075…	10.201.166.185	10.201.234.34	DTLSv1.0	117	Set	Client Hello
92	07:42:45.437784	0.002417	10.201.166.185	10.201.234.34	DTLSv1.0	137	Set	Client Hello
98	07:42:45.667215	0.229431	10.201.166.185	10.201.234.34	DTLSv1.0	590	Set	Certificate (Fragment)
99	07:42:45.667260	0.000045	10.201.166.185	10.201.234.34	DTLSv1.0	590	Set	Certificate (Fragment)
100	07:42:45.667293	0.000033	10.201.166.185	10.201.234.34	DTLSv1.0	178	Set	Certificate (Reassembled)
101	07:42:45.667316	0.000023	10.201.166.185	10.201.234.34	DTLSv1.0	329	Set	Client Key Exchange
102	07:42:45.667347	0.000031	10.201.166.185	10.201.234.34	DTLSv1.0	329	Set	Certificate Verify
103	07:42:45.667372	0.000025	10.201.166.185	10.201.234.34	DTLSv1.0	60	Set	Change Cipher Spec
104	07:42:45.667394	0.000022	10.201.166.185	10.201.234.34	DTLSv1.0	123	Set	Encrypted Handshake Message
106	07:42:45.674895	0.007501	10.201.166.185	10.201.234.34	DTLSv1.0	1499	Set	Application Data
107	07:42:45.675288	0.000393	10.201.166.161	10.201.166.185	ICMP	70	Not set,Set	Destination unreachable (Fragmentation needed)
112	07:42:50.671019	4.995731	10.201.166.185	10.201.234.34	DTLSv1.0	411	Set	Application Data
114	07:42:50.718532	0.047513	10.201.166.185	10.201.234.34	DTLSv1.0	539	Set	Application Data
115	07:42:50.718571	0.000039	10.201.166.185	10.201.234.34	DTLSv1.0	539	Set	Application Data

La depuración de nivel de AP correspondiente ("debug capwap client path-mtu") muestra que el AP intentó primero con 1485 bytes y esperó 5 segundos para obtener una respuesta. Si no hay respuesta, envía otro paquete de solicitud de unión con una longitud menor, ya que aún está en la fase de unión y no tenemos tiempo que perder. Va al valor mínimo para conseguir que el AP se una al WLC, como se indica en el registro del debug:

```
*Jul 11 18:27:15.000: CAPWAP_PATHMTU: CAPWAP_DTLS_SETUP: MTU = 1485
*Jul 11 18:27:15.000: CAPWAP_PATHMTU: Setting default MTU: MTU discovery can start with 576
*Jul 11 18:27:15.235: %CAPWAP-5-DTLSREQSUCC: DTLS connection created sucessfully peer_ip: 10.201.234.34
*Jul 11 18:27:15.235: CAPWAP_PATHMTU: Sending Join Request Path MTU payload, Length 1376, MTU 576
*Jul 11 18:27:15.235: %CAPWAP-5-SENDJOIN: sending Join Request to 10.201.234.34
...
*Jul 11 18:27:20.235: %CAPWAP-5-SENDJOIN: sending Join Request to 10.201.234.34
*Jul 11 18:27:21.479: %CAPWAP-5-JOINEDCONTROLLER: AP has joined controller c9800-CL
```

Y si ejecuta #show capwap client rcb en este momento, verá que CAPWAP AP MTU tiene 576 bytes:

```
3702-AP#show capwap client rcb
AdminState : ADMIN_ENABLED
Primary SwVer : 17.9.3.50
..
MwarName : c9800-CL
MwarApMgrIp : 10.201.234.34
OperationState : JOIN
CAPWAP Path MTU : 576
```

Fase de estado de ejecución

Después de que el AP se una con éxito al controlador del Wireless LAN. puede ver el Mecanismo de detección de PMTU en juego, donde después de 30 segundos puede ver que el AP comienza a negociar un valor de PMTU más alto enviando otro paquete CAPWAP con un conjunto de bits DF de ese tamaño del siguiente valor de PMTU más alto.

En este ejemplo, el AP probó el valor de 1005 bytes. Debido a que el IOS excluye la Ethernet del campo PMTU, verá 1019 bytes en el cable. Si el WLC responde, el AP actualiza la PMTU a 1005 bytes. Si no es así, espera 30 segundos y vuelve a intentarlo.

Esta captura de pantalla muestra una negociación de AP exitosa de PMTU 1005 (consulte paquetes #268 y #269). Observe que estos paquetes tienen diferentes tamaños, lo que se debe a que el WLC tiene un algoritmo diferente para el cálculo de PMTU.

266	08:36:06.777257	21.0865…	10.201.166.185	10.201.234.34	DTLSv1.0	123 Set	Application Data
267	08:36:06.778067	0.000810	10.201.234.34	10.201.166.185	DTLSv1.0	139 Set	Application Data
268	08:36:12.689324	5.911257	10.201.166.185	10.201.234.34	DTLSv1.0	1019 Set	Application Data
269	08:36:12.690257	0.000933	10.201.234.34	10.201.166.185	DTLSv1.0	987 Set	Application Data
270	08:36:12.700439	0.010182	10.201.166.185	10.201.234.34	DTLSv1.0	155 Set	Application Data
271	08:36:12.701442	0.001003	10.201.234.34	10.201.166.185	DTLSv1.0	139 Set	Application Data

Aquí, la depuración de nivel de AP correspondiente (debug capwap client pmtu) muestra dónde el AP negoció exitosamente la PMTU de 1005 bytes y actualizó el valor de PMTU de AP.

```
*Jul 11 18:28:39.911: CAPWAP_PATHMTU: PMTU Timer Expired: Trying to send higher MTU packet 576
*Jul 11 18:28:39.911: CAPWAP_PATHMTU: PMTU Timer:Sending Path MTU packet of size 1005
*Jul 11 18:28:39.911: CAPWAP_PATHMTU: MTU = 1005 for current MTU path discovery
*Jul 11 18:28:39.911: CAPWAP_PATHMTU: Ap Path MTU payload with MTU 1005 sent 888
*Jul 11 18:28:39.911: CAPWAP_PATHMTU: Stopping the message timeout timer
*Jul 11 18:28:39.911: CAPWAP_PATHMTU: Setting MTU to : 1005, it was 576
*Jul 11 18:28:39.911: CAPWAP_PATHMTU: Updating MTU to DPAA
*Jul 11 18:28:39.915: CAPWAP_PATHMTU: Sending MTU update to WLC
*Jul 11 18:28:39.915: CAPWAP_PATHMTU: MTU = 1005 for current MTU path discovery
*Jul 11 18:28:39.915: CAPWAP_PATHMTU: Ap Path MTU payload with MTU 1005 sent 21
```

Y si lo hace (#show capwap client rcb) en este momento encuentra que CAPWAP AP MTU en

1005 bytes, Aquí está el resultado show:

```
3702-AP#show capwap client rcb
AdminState : ADMIN_ENABLED
Primary SwVer : 17.9.3.50
Name : 3702-AP
MwarName : c9800-CL
MwarApMgrIp : 10.201.234.34
OperationState : UP
CAPWAP Path MTU : 1005
```

Después de 30 segundos, el AP intenta de nuevo negociar el valor más alto siguiente de 1485 bytes, sin embargo, el AP recibió ICMP inalcanzable mientras el estado del AP está en el estado RUN. El ICMP inalcanzable tiene un valor de salto siguiente, y el AP honra este valor y lo utiliza para calcular su propia PMTU como podemos ver en los debugs.

```
*Jul 11 18:29:45.911: CAPWAP_PATHMTU: PMTU Timer:Sending Path MTU packet of size 1485
*Jul 11 18:29:45.911: CAPWAP_PATHMTU: MTU = 1485 for current MTU path discovery
*Jul 11 18:29:45.911: CAPWAP_PATHMTU: Ap Path MTU payload with MTU 1485 sent 1368
*Jul 11 18:29:45.911: CAPWAP_PATHMTU: Received ICMP Dst unreachable
*Jul 11 18:29:45.911: CAPWAP_PATHMTU: Src port:5246 Dst Port:60542, SrcAddr:10.201.166.185 Dst Addr:10.
*Jul 11 18:29:45.911: CAPWAP_PATHMTU: Calculated MTU 1293, last_icmp_mtu 1300
*Jul 11 18:29:48.911: CAPWAP_PATHMTU: Path MTU message could not reach WLC, Removing it from the Reliab
```

Capturas del nivel de AP correspondiente

Observe el paquete ICMP inalcanzable número 281 y luego el AP intenta negociar una PMTU con honores del valor de salto siguiente ICMP en 1300 bytes en los paquetes número 288 y respuesta en 289:

280	08:36:42.691876	23.9733…	10.201.166.185	10.201.234.34	DTLSv1.0	1499 Set	Application Data
281	08:36:42.692200	0.000324	10.201.166.161	10.201.166.185	ICMP	70 Not set,Set	Destination unreachable (Fragmentation needed)
282	08:36:45.695098	3.002898	10.201.166.185	10.201.234.34	CAPWAP-Data	92 Set	CAPWAP-Data Keep-Alive[Malformed Packet]
283	08:36:45.695533	0.000435	10.201.166.185	10.201.234.34	DTLSv1.0	139 Set	Application Data
284	08:36:45.695785	0.000252	10.201.234.34	10.201.166.185	CAPWAP-Data	92 Set	CAPWAP-Data Keep-Alive[Malformed Packet]
285	08:36:45.695931	0.000146	10.201.234.34	10.201.166.185	DTLSv1.0	123 Set	Application Data
286	08:36:45.696416	0.000485	10.201.166.185	10.201.234.34	DTLSv1.0	155 Set	Application Data
287	08:36:45.696981	0.000565	10.201.234.34	10.201.166.185	DTLSv1.0	139 Set	Application Data
288	08:36:48.695568	2.998587	10.201.166.185	10.201.234.34	DTLSv1.0	1307 Set	Application Data
289	08:36:48.696456	0.000888	10.201.234.34	10.201.166.185	DTLSv1.0	1275 Set	Application Data
290	08:36:48.706641	0.010185	10.201.166.185	10.201.234.34	DTLSv1.0	155 Set	Application Data
291	08:36:48.707636	0.000995	10.201.234.34	10.201.166.185	DTLSv1.0	139 Set	Application Data

Comportamiento de COS AP

Hay diferencias en el mecanismo de detección para AP-COS AP. Empezamos en la unión AP.

Fase de unión de PA

Al unirse, el AP envía una solicitud de unión con el valor máximo y espera cinco segundos.

Si no hay respuesta, vuelve a intentarlo y espera otros cinco segundos.

Si aún no hay respuesta, envía otra solicitud de unión con 1005 bytes. Si lo consigue, actualiza la PMTU y continúa (por ejemplo, descarga de imágenes). Si la sonda DF de 1005 bytes todavía no puede alcanzar el controlador, cae al mínimo 576 y vuelve a intentar.

Aquí está el debug capwap client pmtu en el nivel AP:

```
Jul 11 19:06:10 kernel: [*07/11/2023 19:06:10.7065] AP_PATH_MTU_PAYLOAD_msg_enc_cb: request pmtu 1485,
Jul 11 19:06:10 kernel: [*07/11/2023 19:06:10.7066] Sending Join request to 10.201.234.34 through port
Jul 11 19:06:10 kernel: [*07/11/2023 19:06:10.7066] Sending Join Request Path MTU payload, Length 1376
..
Jul 11 19:06:15 kernel: [*07/11/2023 19:06:15.3235] AP_PATH_MTU_PAYLOAD_msg_enc_cb: request pmtu 1485,
Jul 11 19:06:15 kernel: [*07/11/2023 19:06:15.3235] Sending Join request to 10.201.234.34 through port
Jul 11 19:06:15 kernel: [*07/11/2023 19:06:15.3235] Sending Join Request Path MTU payload, Length 1376
Jul 11 19:06:15 kernel: [*07/11/2023 19:06:15.3245] chatter: chkcapwapicmpneedfrag :: CheckCapwapICMPNe
..
Jul 11 19:06:20 kernel: [*07/11/2023 19:06:20.0794] AP_PATH_MTU_PAYLOAD_msg_enc_cb: request pmtu 1005,
Jul 11 19:06:20 kernel: [*07/11/2023 19:06:20.0794] Sending Join request to 10.201.234.34 through port
Jul 11 19:06:20 kernel: [*07/11/2023 19:06:20.0794] Sending Join Request Path MTU payload, Length 896
Jul 11 19:06:20 kernel: [*07/11/2023 19:06:20.0831] Join Response from 10.201.234.34, packet size 917
Jul 11 19:06:20 kernel: [*07/11/2023 19:06:20.0832] AC accepted previous sent request with result code:
Jul 11 19:06:20 kernel: [*07/11/2023 19:06:20.0832] Received wlcType 0, timer 30
Jul 11 19:06:20 kernel: [*07/11/2023 19:06:20.5280] WLC confirms PMTU 1005, updating MTU now.
Jul 11 19:06:20 kernel: [*07/11/2023 19:06:20.5702] PMTU: Set capwap_init_mtu to TRUE and dcb's mtu to
Jul 11 19:06:20 kernel: [*07/11/2023 19:06:20.5816] CAPWAP State: Image Data
Jul 11 19:06:20 kernel: [*07/11/2023 19:06:20.5822] AP image version 17.9.3.50 backup 17.6.5.22, Contro
```

Observe que el tamaño del paquete es de 1483 bytes, que es el valor de pmtu sin el encabezado ethernet como se esperaba para AP-COS. Puede ver esto en el paquete número 1168 aquí:

1135	09:13:33.358475	0.000768	10.201.166.187	10.201.234.34	CAPWAP-Control	298 Set	CAPWAP-Control - Discovery Request[Malformed Packet]
1136	09:13:33.359044	0.000569	10.201.234.34	10.201.166.187	CAPWAP-Control	143 Set	CAPWAP-Control - Discovery Response
1151	09:13:38.172586	4.813542	Cisco_93:84:60	Cisco_93:84:60	WLCCP	290 Set	U, func=UI; SNAP, OUI 0x004096 (Cisco Systems, Inc), PID 0x0000
1153	09:13:42.905529	4.732943	10.201.166.187	10.201.234.34	DTLSv1.2	272 Set	Client Hello
1154	09:13:42.906900	0.001371	10.201.234.34	10.201.166.187	DTLSv1.2	94 Set	Hello Verify Request
1155	09:13:42.907727	0.000827	10.201.166.187	10.201.234.34	DTLSv1.2	292 Set	Client Hello
1156	09:13:42.909930	0.002203	10.201.234.34	10.201.166.187	DTLSv1.2	558 Set	Server Hello, Certificate[Reassembly error, protocol DTLS: New f
1157	09:13:42.909963	0.000033	10.201.234.34	10.201.166.187	DTLSv1.2	558 Set	Certificate[Reassembly error, protocol DTLS: New fragment overla
1158	09:13:42.909990	0.000027	10.201.234.34	10.201.166.187	DTLSv1.2	558 Set	Certificate[Reassembly error, protocol DTLS: New fragment overla
1159	09:13:42.910032	0.000042	10.201.234.34	10.201.166.187	DTLSv1.2	558 Set	Certificate[Reassembly error, protocol DTLS: New fragment overla
1160	09:13:42.910060	0.000028	10.201.234.34	10.201.166.187	DTLSv1.2	558 Set	Certificate[Reassembly error, protocol DTLS: New fragment overla
1161	09:13:42.910087	0.000027	10.201.234.34	10.201.166.187	DTLSv1.2	121 Set	Certificate Request[Reassembly error, protocol DTLS: New fragmen
1162	09:13:42.928659	0.018572	10.201.166.187	10.201.234.34	DTLSv1.2	590 Set	Certificate[Reassembly error, protocol DTLS: New fragment overla
1163	09:13:42.942614	0.013955	10.201.166.187	10.201.234.34	DTLSv1.2	590 Set	Certificate[Reassembly error, protocol DTLS: New fragment overla
1164	09:13:43.552554	0.609940	10.201.166.187	10.201.234.34	DTLSv1.2	459 Set	Client Key Exchange[Reassembly error, protocol DTLS: New fragmen
1165	09:13:43.554047	0.001493	10.201.234.34	10.201.166.187	DTLSv1.2	121 Set	Change Cipher Spec, Encrypted Handshake Message
1168	09:13:48.216965	4.662918	10.201.166.187	10.201.234.34	DTLSv1.2	1483 Set	Application Data
1169	09:13:48.217294	0.000329	10.201.166.161	10.201.166.187	ICMP	70 Not set,Set	Destination unreachable (Fragmentation needed)
1173	09:13:52.972786	4.755492	10.201.166.187	10.201.234.34	DTLSv1.2	1003 Set	Application Data
1174	09:13:52.975783	0.002997	10.201.234.34	10.201.166.187	DTLSv1.2	1000 Set	Application Data
1179	09:13:53.939451	0.963668	10.201.166.187	10.201.234.34	DTLSv1.2	955 Set	Application Data
1180	09:13:53.939497	0.000046	10.201.166.187	10.201.234.34	DTLSv1.2	955 Set	Application Data
1181	09:13:53.939526	0.000029	10.201.166.187	10.201.234.34	DTLSv1.2	955 Set	Application Data
1182	09:13:53.939555	0.000029	10.201.166.187	10.201.234.34	DTLSv1.2	527 Set	Application Data
1183	09:13:53.941676	0.002121	10.201.234.34	10.201.166.187	DTLSv1.2	370 Set	Application Data

Fase de estado de ejecución

Después de que el AP alcance el estado RUN. continúa intentando mejorar la PMTU cada 30 segundos, enviando paquetes CAPWAP con DF configurado y el siguiente valor codificado.

Aquí está el debug del nivel AP (debug capwap client pmtu)

```
Jul 11 19:08:15 kernel: [*07/11/2023 19:08:15.1341] wtpEncodePathMTUPayload: Total Packet Size:
Jul 11 19:08:15 kernel: [*07/11/2023 19:08:15.1341] wtpEncodePathMTUPayload: Capwap Size is 137
Jul 11 19:08:15 kernel: [*07/11/2023 19:08:15.1341] [ENC]AP_PATH_MTU_PAYLOAD: pmtu 1485, len 13
Jul 11 19:08:15 kernel: [*07/11/2023 19:08:15.1341] capwap_build_and_send_pmtu_packet: packet l
Jul 11 19:08:15 kernel: [*07/11/2023 19:08:15.1343] Ap Path MTU payload sent, length 1368
Jul 11 19:08:15 kernel: [*07/11/2023 19:08:15.1343] WTP Event Request: AP Path MTU payload sent
Jul 11 19:08:15 kernel: [*07/11/2023 19:08:15.1351] pmtu icmp pkt(ICMP_NEED_FRAG) from click re
Jul 11 19:08:15 kernel: [*07/11/2023 19:08:15.1351] chatter: chkcapwapicmpneedfrag :: CheckCapwa
Jul 11 19:08:15 kernel: [*07/11/2023 19:08:15.1351] PMTU data: dcb->mtu 1005, pmtu_overhead:118
Jul 11 19:08:15 kernel: [*07/11/2023 19:08:15.1351] PMTU: Last try for next hop MTU failed
Jul 11 19:08:17 kernel: [*07/11/2023 19:08:17.9850] wtpCleanupPMTUPacket: PMTU: Found matching
..
Jul 11 19:08:43 kernel: [*07/11/2023 19:08:43.6435] wtpEncodePathMTUPayload: Total Packet Size:
Jul 11 19:08:43 kernel: [*07/11/2023 19:08:43.6435] wtpEncodePathMTUPayload: Capwap Size is 137
Jul 11 19:08:43 kernel: [*07/11/2023 19:08:43.6436] [ENC]AP_PATH_MTU_PAYLOAD: pmtu 1485, len 13
Jul 11 19:08:43 kernel: [*07/11/2023 19:08:43.6436] capwap_build-and-send_pmtu_packet: packet l
Jul 11 19:08:43 kernel: [*07/11/2023 19:08:43.6437] Ap Path MTU payload sent, length 1368
Jul 11 19:08:43 kernel: [*07/11/2023 19:08:43.6438] WTP Event Request: AP Path MTU payload sent
Jul 11 19:08:43 kernel: [*07/11/2023 19:08:43.6446] pmtu icmp pkt(ICMP_NEED_FRAG) from click re
Jul 11 19:08:43 kernel: [*07/11/2023 19:08:43.6446] chatter: chkcapwapicmpneedfrag :: CheckCapwa
Jul 11 19:08:43 kernel: [*07/11/2023 19:08:43.6446] PMTU data: dcb->mtu 1005, pmtu_overhead:118
Jul 11 19:08:43 kernel: [*07/11/2023 19:08:43.6447] PMTU: Last try for next hop MTU failed
Jul 11 19:08:46 kernel: [*07/11/2023 19:08:46.4945] wtpCleanupPMTUPacket: PMTU: Found matching
```

Aquí están las capturas de AP correspondientes. observe los paquetes número 1427 y 1448:

1424	09:15:13.511489	0.000057	Cisco_93:84:60	Cisco_93:84:60	WLCCP	671 Set	U, func=UI; SNAP, OUI 0x000000 (Officially Xer
1425	09:15:19.805660	6.294171	10.201.166.187	10.201.234.34	DTLSv1.2	1483 Set	Application Data
1427	09:15:19.806104	0.000444	10.201.166.161	10.201.166.187	ICMP	70 Not set,Set	Destination unreachable (Fragmentation needed)
1428	09:15:19.806515	0.000411	10.201.234.34	10.201.166.187	CAPWAP-Data	100 Set	CAPWAP-Data Keep-Alive[Malformed Packet]
1433	09:15:21.462377	1.655862	Cisco_93:84:60	Cisco_93:84:60	WLCCP	122 Set	U, func=UI; SNAP, OUI 0x000000 (Officially Xer
1434	09:15:21.462413	0.000036	Cisco_93:84:60	Cisco_93:84:60	WLCCP	122 Set	U, func=UI; SNAP, OUI 0x000000 (Officially Xer
1435	09:15:21.850913	0.388500	Cisco_93:84:60	Cisco_93:84:60	WLCCP	122 Set	U, func=UI; SNAP, OUI 0x000000 (Officially Xer
1438	09:15:32.161352	10.3104…	10.201.166.187	10.201.234.34	DTLSv1.2	107 Set	Application Data
1439	09:15:32.162037	0.000685	10.201.234.34	10.201.166.187	DTLSv1.2	114 Set	Application Data
1440	09:15:33.665648	1.503611	10.201.166.187	10.201.234.34	DTLSv1.2	571 Set	Application Data
1441	09:15:33.666353	0.000705	10.201.234.34	10.201.166.187	DTLSv1.2	99 Set	Application Data
1443	09:15:37.533517	3.867164	Cisco_93:84:60	Cisco_93:84:60	WLCCP	122 Set	U, func=UI; SNAP, OUI 0x000000 (Officially Xer
1444	09:15:38.122776	0.589259	Cisco_93:84:60	Cisco_93:84:60	WLCCP	122 Set	U, func=UI; SNAP, OUI 0x000000 (Officially Xer
1445	09:15:38.171399	0.048623	Cisco_93:84:60	Cisco_93:84:60	WLCCP	290 Set	U, func=UI; SNAP, OUI 0x004096 (Cisco Systems,
1447	09:15:40.684943	2.513544	Cisco_93:84:60	Cisco_93:84:60	WLCCP	122 Set	U, func=UI; SNAP, OUI 0x000000 (Officially Xer
1448	09:15:48.314752	7.629809	10.201.166.187	10.201.234.34	DTLSv1.2	1483 Set	Application Data
1450	09:15:48.315088	0.000336	10.201.166.161	10.201.166.187	ICMP	70 Not set,Set	Destination unreachable (Fragmentation needed)
1451	09:15:48.315397	0.000309	10.201.234.34	10.201.166.187	CAPWAP-Data	100 Set	CAPWAP-Data Keep-Alive[Malformed Packet]
1452	09:15:48.563890	0.248493	Cisco_93:84:60	Cisco_93:84:60	WLCCP	266 Set	U, func=UI; SNAP, OUI 0x000000 (Officially Xer

Conclusión (resumen del algoritmo)

En resumen, el algoritmo CAPWAP PMTUD en los puntos de acceso funciona así.

Paso 1. La PMTU CAPWAP inicial se negocia durante la fase de unión de AP.

Paso 2. 30 segundos después, el AP intenta mejorar la PMTU CAPWAP actual enviando el siguiente valor superior predefinido (576, 1005, 1485 bytes).

Paso 3 (opción 1). Si el WLC responde, ajuste la PMTU CAPWAP actual al nuevo valor y repita el paso 2.

Paso 3 (opción 2). Si no hay respuesta, mantenga la PMTU CAPWAP actual y repita el paso 2.

Paso 3 (opción 3). Si no hay respuesta y un ICMP inalcanzable (tipo 3, código 4) incluye una MTU de siguiente salto, ajuste la PMTU CAPWAP a ese valor y repita el paso 2.

NOTE: Vea las correcciones para asegurarse de que se utilice la PMTU CAPWAP correcta cuando se proporcione un valor de salto siguiente ICMP.

CDET relacionados

Número de problema 1:

ID de bug de Cisco CSCwf52815

AP-COS AP que no honran el valor de siguiente salto ICMP inalcanzable cuando fallan los sondeos de mayor valor.
Correcciones: 8.10.190.0, 17.3.8, 17.6.6, 17.9.5, 17.12.2.
Los AP del IOS honran el valor del salto siguiente y actualizan la PMTU.

Número de problema 2:

ID de bug de Cisco CSCwc05350

La MTU asimétrica (WLC→AP difiere de AP→WLC) llevó a la inestabilidad de PMTU cuando el ICMP no reflejaba la PMTU bidireccional máxima.
Correcciones: 8.10.181.0, 17.3.6, 17.6.5, 17.9.2 y 17.10.1.
Solución alternativa: configure la misma MTU en ambas direcciones en los dispositivos que controlan la MTU (router, firewall, concentrador VPN) entre el WLC y el AP.

ID de bug de Cisco del lado AP relacionado CSCwc05364: El COS-AP mejora el mecanismo de PMTU para poder identificar el tamaño máximo de MTU direccional para las MTU asimétricas

ID de bug de Cisco del lado WLC relacionado CSCwc48316: Mejore los cálculos de PMTU para que el AP pueda tener dos MTU diferentes, una ascendente y otra (marcada como Cerrada por DE ya que no tienen planes para abordar esto)

Número de problema 3:

ID de bug de Cisco CSCwf91557

AP-COS detiene la detección de PMTU después de alcanzar el valor máximo codificado por hardware.

Fijo en 17.13.1; también mediante Fixed mediante el identificador de bug Cisco CSCwf52815 en 17.3.8, 17.6.6, 17.9.5, 17.12.2.

Número de problema 4:

ID de bug de Cisco CSCwk70785

AP-COS no actualiza el valor de MTU para la sonda PMTU, lo que causa desconexiones.

corregido en Cisco bug ID CSCwk90660 - APSP6 17.9.5] Target 17.9.6, 17.12.5, 17.15.2, 17.16.

Número de problema 5:

ID de bug de Cisco CSCvv53456

Configuración de MTU de ruta CAPWAP estática 9800 (paridad con AireOS).

Esto permite que el 9800 tenga una MTU de trayectoria de CAPWAP estática configurada en función del perfil de unión de AP. Pasando a las 17:17.

Acerca de esta traducción
Cisco ha traducido este documento combinando la traducción automática y los recursos humanos a fin de ofrecer a nuestros usuarios en todo el mundo contenido en su propio idioma.

Tenga en cuenta que incluso la mejor traducción automática podría no ser tan precisa como la proporcionada por un traductor profesional.

Cisco Systems, Inc. no asume ninguna responsabilidad por la precisión de estas traducciones y recomienda remitirse siempre al documento original escrito en inglés (insertar vínculo URL).