Comprendre et dépanner CAPWAP Keepalives sur le WLC Catalyst 9800

Table des matières

Introduction

Ce document décrit les keepalives de contrôle CAPWAP, les keepalives de données et les retransmissions,

Contexte

CAPWAP fournit l'infrastructure de communication entre les points d'accès Cisco et le contrôleur LAN sans fil sur les plates-formes Cisco Catalyst 9800. Il utilise des canaux de contrôle et de données séparés, ainsi que des mécanismes de test d'activité et de retransmission définis pour maintenir la connectivité.

Cet article explique les keepalives de contrôle CAPWAP, les keepalives de données et les retransmissions, y compris les temporisateurs associés, le comportement de défaillance et les indicateurs de dépannage clés.

Deux Canaux Séparés, Deux Mécanismes Différents

CAPWAP fonctionne sur deux canaux UDP, et chacun a son propre mécanisme d'activité

Canal	Port par défaut (UDP)	Mécanisme D'Activité
Contrôle	5246	Requête d'écho / Réponse d'écho
Données	5247	Maintien de la connexion au canal de données

Contrôle CAPWAP Keepalive (écho / pulsation)

Objectif

Le keepalive de contrôle CAPWAP est utilisé pour confirmer que le point d'accès est toujours accessible sur le canal de contrôle.

Son objectif est très simple : il vérifie que la connexion de contrôle entre le point d'accès et le contrôleur est toujours active et réactive. Il n'est pas utilisé pour transporter des mises à jour de configuration ou des données client. Au lieu de cela, il sert de mécanisme d'activité pour le chemin de contrôle CAPWAP sur le port UDP 5246.

Qui envoie le Keepalive ?

Le point d'accès est le périphérique qui initialise activement le keepalive de contrôle.

Si nécessaire, le point d'accès envoie une requête d'écho CAPWAP au contrôleur. Le contrôleur répond ensuite avec une réponse d'écho correspondante. Cette réponse confirme que le contrôleur a reçu la requête et que le canal de contrôle fonctionne dans les deux sens.

La réponse correspond à la requête, ce qui permet au point d'accès de confirmer qu'il reçoit une réponse valide au keepalive qu'il a envoyé.

Ce comportement est important car il montre que le keepalive de contrôle n'est pas principalement un mécanisme d'interrogation piloté par le contrôleur. Au lieu de cela, l'AP est responsable de vérifier l'accessibilité du canal de contrôle, et le contrôleur répond en conséquence.

L'intervalle d'écho est basé sur l'inactivité, et non sur un planning fixe

C'est l'un des aspects les plus souvent mal compris des keepalives de contrôle CAPWAP.

Une hypothèse commune est que le point d'accès envoie une requête d'écho toutes les 30 secondes comme un battement de coeur périodique fixe. En pratique, cela ne fonctionne pas ainsi.

Le keepalive de contrôle est basé sur l'inactivité du canal de contrôle. Cela signifie que le point d'accès envoie une requête d'écho autonome uniquement lorsque le canal de contrôle a été inactif pendant l'intervalle configuré. Si un autre trafic de contrôle est déjà en cours d'échange entre le point d'accès et le contrôleur, il n'est pas nécessaire d'envoyer un paquet d'écho séparé.

Toute communication de contrôle valide entre le point d'accès et le contrôleur prouve efficacement que le canal de contrôle est actif. Pour cette raison, le trafic de contrôle CAPWAP normal réinitialise le compteur de test d'activité.

Voici des exemples de trafic de contrôle :

- échanges de configuration
- mises à jour de gestion des ressources radio
- Messages de contrôle liés au WLAN
- événements de contrôle liés au client
- statistiques ou messages de synchronisation d'état
- autres paquets de contrôle CAPWAP échangés dans le cadre d'un fonctionnement normal

Par conséquent, les requêtes d'écho autonomes ne sont visibles que pendant les périodes où le canal de contrôle est silencieux.

Exemple simple

Vous pouvez vérifier les compteurs d'écho CAPWAP sur l'AP en utilisant la commande show capwap client timer.

<#root>

```
Training-AP#

show capwap client timer


CAPWAP TIMERS Running Current / Max (seconds)
PATHMTU_TIMER : 23 / 30
MSG_CLIENT_STAT : 124 / 180
DATA_CHANNEL_KEEP_ALIVE_TIMER : 24 / 30
```

ECHO_INTERVAL_TIMER : 23 / 30

PERIODIC_ECHO_TIMER : 233 / 300

PRIMARY_DISCOVERY_TIMER : 50 / 120
CAPWAP_WDG_UPDATE_TIMER : 4 / 5
FLASH_WRITE_INTERVAL_TIMER : 21 / 60

Minuteurs

Le mécanisme de test d'activité de contrôle repose sur deux valeurs de synchronisation importantes.

Élément	Valeur par défaut	Description
Intervalle d'écho	30 secondes	Quantité d'inactivité du canal de contrôle avant que le point d'accès envoie une requête d'écho autonome
Minuteur d'arrêt de contrôle du contrôleur	90 secondes	La durée maximale que le contrôleur autorise sans recevoir de trafic de contrôle valide avant de déclarer la session de contrôle morte

Que se passe-t-il lorsque le contrôle Keepalive échoue ?

Si le canal de contrôle devient silencieux pendant plus longtemps que le délai d'attente autorisé, le contrôleur finit par déclarer la session de contrôle perdue.

À ce stade, le contrôleur traite le point d'accès comme n'étant plus accessible sur le canal de contrôle et commence l'interruption de session. Cela inclut généralement la fermeture de la connexion de contrôle et la suppression de l'état de session de contrôle actif de l'AP.

Le point d'accès peut alors tenter de rétablir la connectivité en redécouvrant et en rejoignant à nouveau le contrôleur, selon le scénario de défaillance.

Implications du dépannage

Il est essentiel de comprendre ce comportement de test d'activité lors du dépannage.

Les paquets d’écho manquants ne posent pas toujours problème

Si une capture de paquet n'affiche pas de requêtes d'écho toutes les 30 secondes, cela n'indique pas automatiquement une défaillance. Cela peut simplement signifier qu'assez d'autres trafics de contrôle CAPWAP circulent, donc aucun écho autonome n'est requis.

Un Espacement D'Écho Irrégulier Est Généralement Normal

Les échos apparaissent souvent à des intervalles non uniformes car ils sont déclenchés par l'inactivité. C’est un comportement attendu.

Concentration sur l'activité globale du canal de contrôle

Lors du dépannage des déconnexions AP, il est plus utile de demander :

- Le contrôleur reçoit-il toujours du trafic de contrôle CAPWAP du point d'accès ?
- Le canal de contrôle est-il silencieux suffisamment longtemps pour déclencher le minuteur d'arrêt ?
- Des problèmes réseau affectent-ils uniquement le chemin de contrôle ?
- La perte de paquets, le comportement du pare-feu, les problèmes NAT ou les abandons du plan de contrôle interrompent-ils le trafic UDP 5246 ?

Le vrai problème n'est pas l'absence de paquets d'écho périodiques par eux-mêmes. Le vrai problème est la perte de communication du canal de contrôle pendant assez longtemps pour que le contrôleur déclare le point d'accès inaccessible.

Maintien des données CAPWAP

Objectif

L'écho de contrôle CAPWAP confirme que le canal de contrôle fonctionne, mais il ne prouve pas que le canal de données est également sain. Puisque le chemin de contrôle et le chemin de données peuvent échouer indépendamment, CAPWAP utilise un keepalive de données séparé sur le port UDP 5247.

L'objectif du keepalive est de :

- Vérifiez que le point d'accès peut toujours atteindre le contrôleur sur le canal de données
- maintenir l'état du tunnel de données du point d'accès

- aider à détecter les pannes qui affectent uniquement le chemin de données

Qui l'envoie et comment le contrôleur répond

Le point d'accès envoie le keepalive de données, et le contrôleur répond avec une réponse Data Keepalive.

La réponse peut être chiffrée ou non, selon que le chiffrement du canal de données est activé ou non.

Si le keepalive est valide, le contrôleur l'utilise pour confirmer que le chemin de données de l'AP est toujours accessible. Si le paquet ne peut pas être mis en correspondance avec une session AP valide, ou si la réponse ne peut pas être envoyée, le keepalive est abandonné et le chemin de données peut finalement être considéré comme ayant échoué.

Comment le contrôleur le valide

Avant d'accepter le keepalive, le contrôleur effectue une validation de base pour s'assurer que le paquet appartient au point d'accès correct.

Le contrôleur vérifie que :

- le keepalive contient des informations CAPWAP valides
- l'adresse MAC radio du point d'accès est présente
- le paquet peut être mis en correspondance avec une session AP

Le contrôleur tente d'abord d'identifier la session à l'aide de l'adresse IP source et du port UDP source. Si cela échoue, il peut revenir à l'utilisation de l'adresse MAC radio AP.

Ceci est important pour les AP derrière NAT ou PAT, où le canal de données peut arriver d'un port UDP traduit différent du canal de contrôle. Dans ce cas, le contrôleur peut apprendre et mettre à jour le tuple de canal de données réel du point d'accès.

Si le paquet semble appartenir à un AP différent de la session déjà associée à cette combinaison de ports IP, le keepalive est rejeté pour empêcher un mappage de session incorrect.

Que se passe-t-il après validation

Une fois que le keepalive est accepté, le contrôleur le traite en fonction de l'état de session actuel de l'AP.

- Si la session de données est déjà établie, le keepalive actualise simplement l'activité.
- S'il s'agit du premier keepalive de données valide, le contrôleur peut l'utiliser pour apprendre ou mettre à jour les informations de canal de données de l'AP et compléter l'établissement du tunnel de données.

Ce premier keepalive est particulièrement important car il aide le contrôleur à programmer correctement le tunnel de données, y compris dans les cas où l'AP est derrière NAT ou PAT.

Une fois la session de données entièrement établie, les futures keepalives suivent le chemin normal de l'état stable et ne sont utilisées que pour maintenir l'activité.

Comportement important

Un keepalive de données valide ne fait pas que confirmer l'intégrité du chemin de données. Il actualise également l'activité de session globale du point d'accès sur le contrôleur.

Cela signifie que, sur le contrôleur, l'activité du canal de données contribue à l'intégrité globale de la session AP. Par conséquent, les mécanismes de contrôle et de transmission des données sont liés, même s'ils servent à des fins différentes.

Temporisateurs de maintien de connexion côté point d'accès

Le point d'accès contrôle l'intervalle de test d'activité des données et décide quand le tunnel de données doit être considéré comme hors service.

Élément	Valeur par défaut
Intervalle de conservation des données	30 secondes
Retour arrière de retransmission	3s, 6s, 12s, puis 15s
Intervalle d'inactivité des données	180 secondes

Dans des conditions normales, le point d'accès envoie un keepalive de données toutes les 30 secondes.

Si le point d'accès ne reçoit pas de réponse, il réessaie en utilisant un modèle de réémission temporisée. Si la défaillance continue pendant 180 secondes, le point d'accès déclare le tunnel de données arrêté.

Retransmissions CAPWAP

Principe fondamental : La fiabilité fonctionne dans les deux sens

La messagerie de contrôle CAPWAP utilise un modèle de requête-réponse, même si elle s'exécute sur UDP. Pour assurer la fiabilité, le périphérique qui envoie une requête est également responsable de la retransmission jusqu'à ce que la réponse attendue soit reçue.

Cela signifie que les retransmissions sont symétriques :

- Pour les demandes contrôleur à AP, telles qu'une mise à jour de configuration, le contrôleur gère la retransmission.
- Pour les demandes AP à contrôleur, telles que Discovery, Join, Configuration Status, les messages relatifs à l'image et les Echo Requests, l'AP gère la retransmission.

Il s'agit d'un point de dépannage important. Si vous voyez des retransmissions AP-à-contrôleur dans une capture de paquets, cela signifie généralement que l'AP est simplement en train de retenter sa propre requête parce qu'il n'a pas reçu la réponse attendue. Ceci est normal pendant la jointure et peut également se produire pendant l'activité d'écho de contrôle.

Comportement De Retransmission Côté Contrôleur

Sur le contrôleur, la retransmission est gérée par une machine d'état de fiabilité de transmission dédiée.

En termes simples, le contrôleur :

1. Accepte une requête nécessitant un accusé de réception
2. le place dans une file d'attente de transmission par point d'accès
3. envoie la demande
4. attend la réponse correspondante
5. le supprime de la file d'attente lorsque la réponse arrive ou le retransmet à l'expiration du minuteur

Cette logique est suivie séparément pour chaque session AP. Le contrôleur conserve également une fenêtre de transmission et un compte de messages de demande en attente.

Comment le contrôleur décide-t-il de retransmettre

Chaque fois que le temporisateur de retransmission expire, le contrôleur examine les entrées de requête en file d'attente et prend l'une des décisions suivantes :

1. Réponse déjà reçue hors service

Si la réponse à cette demande a déjà été reçue, même si elle est arrivée hors séquence, le contrôleur ne retransmet pas le message.

2. Limite de tentatives dépassée

Si la requête a déjà été retransmise plus de fois que le nombre autorisé, le contrôleur traite cela comme un échec et abandonne le processus de transmission pour cette session AP.

À ce stade, la session AP est terminée.

3. La demande n'est pas encore assez ancienne

Le contrôleur ne retransmet pas immédiatement chaque message en file d'attente à chaque événement de minuteur. Une demande doit rester dans la file d'attente pendant au moins l'intervalle de retransmission avant d'être à nouveau éligible.

Par défaut, cet intervalle de retransmission est de 3 secondes.

4. Retransmettre la requête

Si la demande est toujours en attente, a expiré suffisamment longtemps et n'a pas dépassé la limite de tentatives, le contrôleur la renvoie sur le canal de contrôle CAPWAP et incrémente le compteur de tentatives.

Cas particulier : Points d'accès câblés en série

Pour les déploiements maillés utilisant un chemin d'accès en chaîne câblé, le contrôleur permet un budget de tentatives plus important.

Dans ce cas, la limite normale de tentatives est effectivement triplée.

Avec le nombre de nouvelles tentatives par défaut de 5, cela signifie que le contrôleur peut recommencer jusqu'à 15 fois avant de déclarer l'échec.

Cette exception existe car ces topologies peuvent nécessiter une tolérance plus élevée pour le retard ou la perte de messages.

Comportement de retransmission côté point d'accès

Le point d'accès retransmet également ses propres requêtes CAPWAP, mais il utilise un modèle différent de celui du contrôleur.

Alors que le contrôleur utilise un intervalle de retransmission fixe, le point d'accès utilise un réémission temporisée exponentielle. Cela signifie que le temps d'attente augmente après chaque tentative échouée.

Avec les paramètres par défaut, le délai de nouvelle tentative du point d'accès est approximativement :

- 6 secondes
- 12 secondes
- 24 secondes
- 48 secondes
- 96 secondes

Ce comportement s'applique aux requêtes CAPWAP émises par AP telles que :

- Découverte
- Se Joindre
- échanges relatifs à la configuration
- transactions de contrôle chiffrées
- Demandes d'écho

Ces paramètres de nouvelle tentative sont appris du contrôleur dans le cadre de la configuration de jonction AP.

Empreinte de diagnostic dans les captures de paquets

Le modèle de retransmission lui-même est souvent un indice utile lors du dépannage.

Schéma De Retransmission	Source probable
Retransmissions régulièrement espacées, environ toutes les 3 secondes	Retransmission côté contrôleur

Schéma De Retransmission	Source probable
Retransmissions qui s'étendent plus loin dans le temps, telles que 6, 12, 24, 48 et 96	Retransmission côté point d'accès avec réémission exponentielle

Il s'agit d'une méthode pratique permettant de déterminer quel côté effectue une nouvelle tentative, en particulier lors d'échecs de jointure ou de problèmes liés à l'écho.

Que se passe-t-il lorsque les tentatives sont épuisées

Si les retransmissions se poursuivent sans recevoir la réponse attendue, les deux parties finissent par abandonner, mais leurs actions de récupération sont différentes.

Côté contrôleur

Si le contrôleur épuise son budget de nouvelles tentatives, il abandonne le processus de transmission et met fin à la session AP. Cela entraîne la fermeture des sessions de contrôle et de données CAPWAP.

côté point d'accès

Si le point d'accès épuise ses propres tentatives de nouvelle tentative, il abandonne ce contrôleur et recommence depuis le début, généralement en retournant à la phase de découverte et en essayant une rejointure complète.

Boutons de configuration et paramètres par défaut

Le comportement de retransmission est contrôlé par deux paramètres principaux dans le profil de jonction AP.

Commande	Valeur par défaut	Fonction
capwap retransmit count	5	Nombre maximal de tentatives de retransmission
intervalle de retransmission capwap	3 secondes	Intervalle de retransmission de base

Ces valeurs influencent à la fois la fiabilité de jointure et d'autres tentatives de contrôle CAPWAP d'origine AP, y compris les tentatives liées à l'écho.

Référence du minuteur consolidée

Ce tableau résume les principaux compteurs CAPWAP abordés dans cet article.

Minuteur	Valeur par défaut	Plan	Propriétaire	Description
Intervalle d'écho de contrôle	30 secondes	Contrôle	POINT D'ACCÈS	Si le point d'accès ne transmet pas de trafic de contrôle CAPWAP pendant 30 secondes, il envoie une requête d'écho.
Contrôler le minuteur d'arrêt de pulsation	90 secondes	Contrôle	Contrôleur	Le contrôleur attend un trafic de contrôle valide dans cette fenêtre. Si rien n'est reçu, la session de contrôle est considérée comme inactive.
Contrôler l'intervalle de retransmission	3 secondes	Contrôle	Contrôleur	Le contrôleur retransmet ses propres demandes CAPWAP sans réponse à un intervalle fixe.
Contrôler le nombre de retransmissions	5 tentatives	Contrôle	Contrôleur	Nombre maximal de tentatives pour les demandes CAPWAP émises par le contrôleur. Dans les scénarios câblés en guirlande, ce nombre peut atteindre 15 tentatives.
Retransmission de désactivation du contrôle AP	6, 12, 24, 48, 96 secondes	Contrôle	POINT D'ACCÈS	Le point d'accès retransmet ses propres requêtes CAPWAP en utilisant un réémission temporisée exponentielle.
Intervalle de conservation des données	30 secondes	Données	POINT D'ACCÈS	Dans des conditions normales, le point d'accès envoie un keepalive de données toutes les 30 secondes.
Data keepalive retry backoff	3, 6, 12, 15, 15	Données	POINT D'ACCÈS	Si une réponse de test d'activité des données est manquée, le point d'accès

Minuteur	Valeur par défaut	Plan	Propriétaire	Description
	secondes			réessaie en utilisant la réémission temporisée, avec un délai maximal de 15 secondes.
Intervalle d'inactivité du canal de données	180 secondes	Données	POINT D'ACCÈS	Si le point d'accès ne parvient pas à maintenir l'échange de données keepalive pendant cette période, il déclare le tunnel de données arrêté.

Différences côte à côte

Keepalive ou retransmission

Bien qu'ils soient parfois confus, le keepalive et la retransmission servent des objectifs différents.

Aspect	Keepalive	Retransmission
Objectif principal	Vérifie que l'homologue est toujours accessible	Relance une requête spécifique lorsqu'aucune réponse n'est reçue
Portée	Par session ou par canal	Par message
Déclencheur	Délai d'inactivité ou d'activité	Une demande reste sans réponse
Méthode de suivi	Basé sur le temps	Basé sur le nombre de tentatives
Défaillance, signification	Le canal ou la session ne sont plus accessibles	Un échange CAPWAP spécifique a échoué à plusieurs reprises
Résultat de l'échec	La session peut être déclarée inactive	Le contrôleur peut mettre fin à la session ou le point d'accès peut redémarrer la connexion

Chaînes du journal de suivi RA par mécanisme

Pour le traçage spécifique à l'AP, collectez les journaux pour l'AP et recherchez ces chaînes exactes.

Collecte des journaux

Utilisation:

- debug wireless mac <ap-radio-mac>
- show logging profile wireless filter mac <ap-radio-mac> to-file bootflash:<fichier>

Contrôle keepalive / heartbeat

Rechercher :

- Traitement de la demande Echo
- Générer une réponse d'écho
- Réponse d’écho envoyée pour la requête avec le numéro de séquence <N>.
- Heartbeat Timer redémarré avec une nouvelle valeur <ms>
- Contact du minuteur Heartbeat invoqué
- Expiration du minuteur Heartbeat

Maintien des données

Rechercher :

- Maintien des données reçu
- données déjà établies
- DTLS de données non établi
- Mise à jour des entrées de tunnel de données CAPWAP
- Échec du traitement de la conservation des données. Motif : Session CAPWAP non en état d'exécution ou de configuration
- Message keepalive non valide, aucun bloc capwap
- Message de test d'activité non valide, aucun MAC WTP
- Échec du traitement du keepalive, session CAPWAP introuvable
- Le maintien de connexion des données est reçu sur un port différent de celui du port de contrôle : <port>
- Le maintien de connexion des données est reçu pour différents points d’accès <mac>
- Échec de la mise à jour des informations de tuple de données dans la table de session WTP

pour l'hôte distant : <ip>
- Impossible de mettre à jour l'entrée du tunnel de données CAPWAP

Gestion de keepalive du plan de données

Rechercher :

- Les données reçues sont conservées de <src> à <dst>
- Réponse de maintien de connexion de données <enc/plain> envoyée
- Suppression du paquet keep-alive de l'adresse IP : <ip>, session inconnue
- Suppression du paquet keep-alive de l'adresse IP : <ip>, échec de l'obtention de l'état DTLS des données
- Échec de l'envoi de la réponse de maintien de connexion des données
- Échec du message de maintien de connexion des données d'équilibrage de charge

Retransmission

Rechercher :

- retransmission timer callback
- Retransmission du séqnum <N> car il est en file d'attente et doit encore être renvoyé
- Retransmission du message : <msg-name> tentative de retransmission : <M>
- entrée avec numéro de séquence : <N> est en file d'attente pendant <S> secondes. Non retransmission
- Fin de session CAPWAP, retransmissions max. expirées pour : <msg> ...
- Nettoyage des ressources de session CAPWAP en cas de congé AP.

Désactivation de session

Rechercher :

- Arrêt de la session AP CAPWAP.
- Fermez la session CAPWAP DTLS.
- capwap-dtls control session close
- capwap-dtls fermeture de session de données
- Dernier paquet de contrôle reçu <S> secondes auparavant.
- Dernier paquet Data Keep Alive reçu <S> secondes auparavant.
- Session DTLS CAPWAP fermée pour AP, cause : <motif>

Exemples:

Scénario de travail

Vérification du débogage AP

Cette commande de débogage AP peut être utilisée pour surveiller le contrôle CAPWAP et la communication de maintien de la connexion de données entre l'AP et le WLC.

```
#debug capwap client event
```

Ces journaux de débogage montrent une séquence de communication CAPWAP réussie.

À 13:11:44, le point d'accès a transmis une requête d'écho CAPWAP au WLC sur le port UDP 5246. Au cours du même intervalle, le point d'accès a également transmis un paquet CAPWAP Data Keepalive sur le port UDP 5247. Les journaux confirment que le WLC a répondu avec succès aux deux demandes.

Les horodatages indiquent un cycle de communication CAPWAP normal :

- 13:11:44.0917 - Demande d'écho transmise.
- 13:11:44.0918 - Transmission du maintien de la connexion des données.
- 13:11:44.0926 - Données Keepalive reçues du WLC.
- 13:11:44.0940 - Réponse d'écho reçue du WLC.

Ces horodatages confirment que les canaux de contrôle et de données CAPWAP fonctionnent comme prévu avec une latence d'aller-retour négligeable.

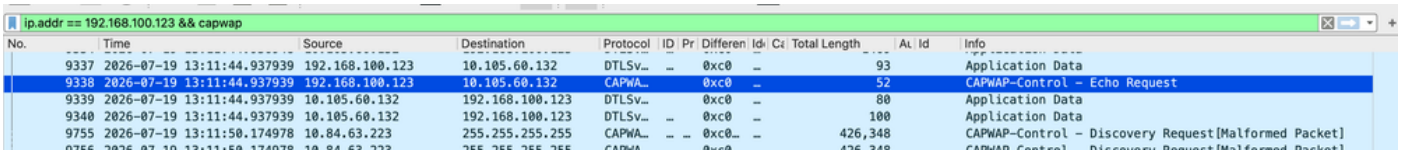
```
[*07/19/2026 13:11:14.0817] [RX]KEEPALIVE: RoundTripTime=0.001 sec
[*07/19/2026 13:11:44.0917] Echo Request: Send count 22
[*07/19/2026 13:11:44.0917] [TX]Echo Request: Sent to 10.105.60.132
[*07/19/2026 13:11:44.0918] [TX]KEEPALIVE: Send to 10.105.60.132-5247
[*07/19/2026 13:11:44.0918] [TX]KEEPALIVE: Schedule for Retransmit in 3 sec sec_drop_count=0
[*07/19/2026 13:11:44.0918] capwap_ap_upstream_send ip: 192.168.100.123, port: 5272 => ip: 10.105.60.13
[*07/19/2026 13:11:44.0926] chatter: [RX]KEEPALIVE: Count 164
[*07/19/2026 13:11:44.0927] Sending KEEPALIVE to WTP SM
[*07/19/2026 13:11:44.0927] Capwap data keep-alive Msg.
[*07/19/2026 13:11:44.0927] [RX]KEEPALIVE: Session ID 3221108139, Next scheduled for TX in 30 sec
[*07/19/2026 13:11:44.0927] [RX]KEEPALIVE: RoundTripTime=0.001 sec
[*07/19/2026 13:11:44.0940] [RX]Echo Response from 10.105.60.132
[*07/19/2026 13:11:44.0004] [RX]Echo Response from 10.105.60.132 RttCount 1
[*07/19/2026 13:12:14.0004] [TX]KEEPALIVE: Send to 10.105.60.132-5247
[*07/19/2026 13:12:14.0005] [TX]KEEPALIVE: Schedule for Retransmit in 3 sec sec_drop_count=0
[*07/19/2026 13:12:14.0005] capwap_ap_upstream_send ip: 192.168.100.123, port: 5272 => ip: 10.105.60.13
```

Validation des paquets de contrôle CAPWAP (port UDP 5246)

Analyse de capture de paquets

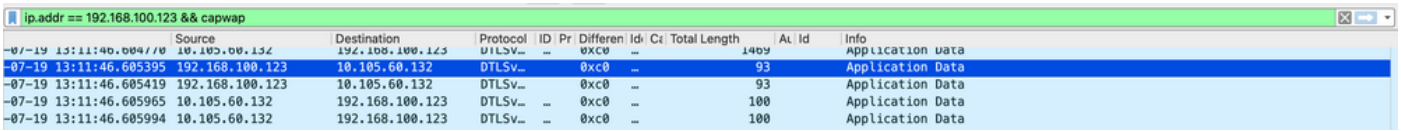
La capture de paquets confirme que le point d'accès a généré une requête d'écho CAPWAP à 13:11:44 sur le port UDP 5246.

Le WLC a reçu le paquet, a traité la demande et a immédiatement généré la réponse d'écho correspondante. Comme le canal de contrôle CAPWAP est protégé à l'aide du chiffrement DTLS, la réponse apparaît sous la forme de données d'application chiffrées dans la capture de paquets.



Vérification des paquets du commutateur

La capture de paquets du commutateur confirme que les paquets de contrôle CAPWAP chiffrés ont été reçus avec succès du WLC et transférés vers le point d'accès.

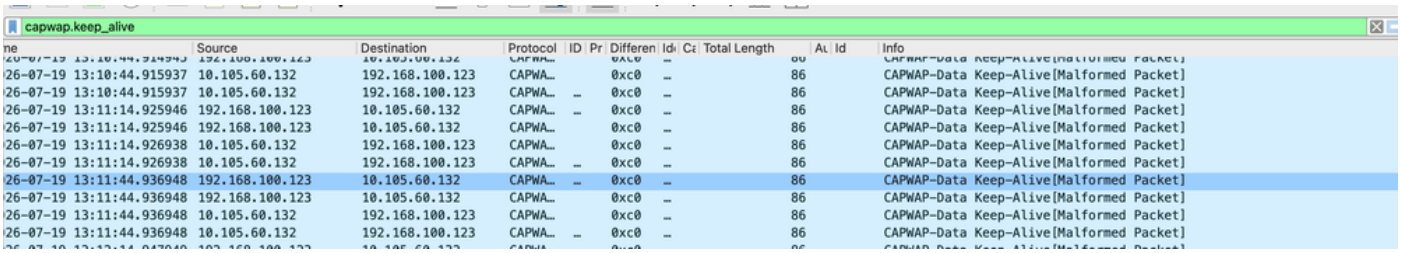


Validation du maintien des données CAPWAP (port UDP 5247)

Le point d'accès transmet périodiquement des paquets CAPWAP Data Keepalive sur le port UDP 5247 pour vérifier l'état du tunnel de données CAPWAP.

À 13:11:44, le point d'accès a transmis un paquet Data Keepalive vers le WLC. Le WLC a reçu le paquet avec succès et a immédiatement répondu avec la réponse keepalive correspondante.

Cet échange réussi confirme que le chemin de données CAPWAP reste opérationnel et que la communication bidirectionnelle entre l'AP et le WLC fonctionne normalement

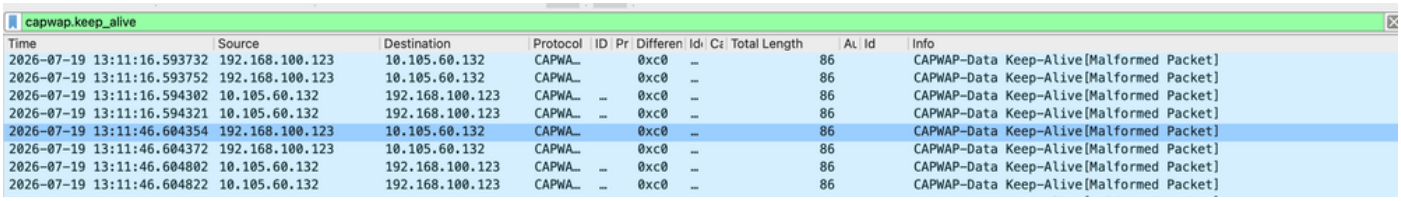


Vérification des paquets du commutateur

La capture de paquets du commutateur confirme en outre que les paquets CAPWAP Data Keepalive ont été transférés avec succès entre le point d'accès et le WLC sans interruption.

Le flux de paquets observé confirme que :

- Les paquets Data Keepalive ont atteint le WLC.
- Le WLC a généré la réponse keepalive attendue.
- Le commutateur a renvoyé la réponse au point d’accès.
- Aucune perte de paquets n’a eu lieu sur le chemin de transfert.



Paquets de contrôle (UDP 5246) abandonnés sur le commutateur de liaison ascendante AP

Ce test démontre le comportement d'un AP quand le port de contrôle CAPWAP (UDP 5246) est abandonné sur le commutateur de liaison ascendante AP.

L'objectif est de valider le comportement du point d'accès, du WLC et du réseau lorsque les paquets de contrôle CAPWAP ne peuvent pas atteindre le point d'accès, malgré le traitement réussi et la réponse réussie du WLC aux requêtes.

Dans ce scénario :

- Le point d'accès continue à transmettre des requêtes d'écho CAPWAP vers le WLC.
- Le WLC reçoit et traite avec succès chaque requête d'écho.
- Le WLC génère la réponse d'écho correspondante.
- Le commutateur de liaison ascendante AP reçoit la réponse mais ne la retransmet pas au

point d'accès.
- Comme le point d'accès ne reçoit jamais la réponse d'écho, il finit par dépasser le nombre maximal de retransmissions et redémarre la machine d'état CAPWAP.

Analyse de débogage AP

À 12:11:55.4568, le point d'accès a transmis une requête d'écho CAPWAP vers le WLC sur le port UDP 5246.

Demande d'écho : Nombre d'envois 0

Contrairement au scénario de travail normal, aucune réponse d'écho n'a été reçue. Par conséquent, le point d'accès a initié des retransmissions selon le minuteur CAPWAP par défaut.

Les retransmissions se sont produites à ces horodatages :

Heure	Événement
12:12:00.2587	Nombre de retransmissions = 1
12:12:03.2599	Nombre de retransmissions = 2
12:12:06.2610	Nombre de retransmissions = 3
12:12:09.2624	Nombre de retransmissions = 4
12:12:12.2637	Nombre de retransmissions = 5

Après la cinquième retransmission infructueuse, le point d'accès a déclaré la session de contrôle CAPWAP inaccessible.

À 12:12:15.2647, le point d'accès a signalé :

Nombre maximal de retransmissions dépassé, retour au mode DISCOVER.

Immédiatement après, l'AP a redémarré la machine d'état CAPWAP pour lancer un nouveau processus de découverte.

```
[*07/17/2026 12:11:54.2577] [RX]KEEPALIVE: Session ID 4068929293, Next scheduled for TX in 30 sec
[*07/17/2026 12:11:54.2577] [RX]KEEPALIVE: RoundTripTime=0.001 sec
[*07/17/2026 12:11:55.4568] Echo Request: Send count 0
[*07/17/2026 12:11:55.4568] [TX]Echo Request: Sent 1 Lost 269
[*07/17/2026 12:12:00.2587] Re-Tx Count=1, Max Re-Tx Value=5, SendSeqNum=63, NumofPendingMsgs=3
[*07/17/2026 12:12:03.2599] Re-Tx Count=2, Max Re-Tx Value=5, SendSeqNum=63, NumofPendingMsgs=3
[*07/17/2026 12:12:06.2610] Re-Tx Count=3, Max Re-Tx Value=5, SendSeqNum=63, NumofPendingMsgs=3
[*07/17/2026 12:12:09.2624] Re-Tx Count=4, Max Re-Tx Value=5, SendSeqNum=63, NumofPendingMsgs=3
[*07/17/2026 12:12:12.2637] Re-Tx Count=5, Max Re-Tx Value=5, SendSeqNum=63, NumofPendingMsgs=3
[*07/17/2026 12:12:15.2647] Max retransmission count exceeded, going back to DISCOVER mode.
[*07/17/2026 12:12:15.2647] Failed to reach capwap down with retransmission 3 times
```

Les traces RA du WLC confirment que le contrôleur n'a pas rencontré de problèmes de traitement.

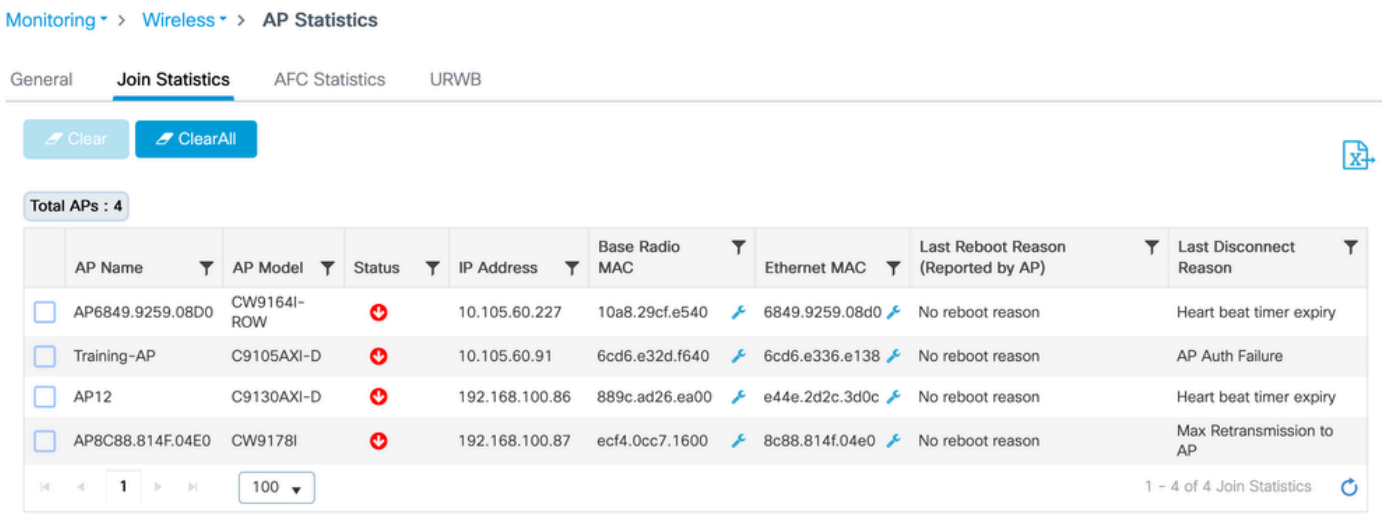
À 12:11:58.731835712, le WLC a reçu avec succès la demande d'écho CAPWAP transmise par l'AP. Ces journaux démontrent que le WLC a traité la demande avec succès et a généré la réponse appropriée. Plus tard, à 12:12:19.802183814, le WLC a reçu une DTLS Close Notify de l'AP. Le point d'accès s'est déconnecté parce qu'il n'a jamais reçu les réponses d'écho transmises par le contrôleur. Par conséquent, le WLC a terminé la session DTLS et a enregistré la dissociation du point d'accès.

<#root>

```
2026/07/17 12:12:19.802274790 {wncd_x_R0-1}{2}: [ewlc-dtls-sess] [16522]: (info):

 Remote Host: 192.168.100.120[5256] MAC: 889c.ad26.ea00  dtls session closed


2026/07/17 12:12:19.802279648 {wncd_x_R0-1}{2}: [ewlc-infra-capwap-dgram] [16522]: (debug): dgram handl
2026/07/17 12:12:19.802317470 {wncd_x_R0-1}{2}: [ewlc-capwapmsg-sess] [16522]: (debug): Encrypted DTLS
2026/07/17 12:12:19.802321202 {wncd_x_R0-1}{2}: [capwapac-smgr-srvr] [16522]: (debug): Mac: 889c.ad26.e
2026/07/17 12:12:19.802376588 {wncd_x_R0-1}{2}: [ap-join-info-db] [16522]: (note): MAC: 889c.ad26.ea00

AP disconnect initiated. Name : AP12, Ethernet mac : e44e.2d2c.3d0c, Reason: DTLS close alert from peer,
```

<#root>

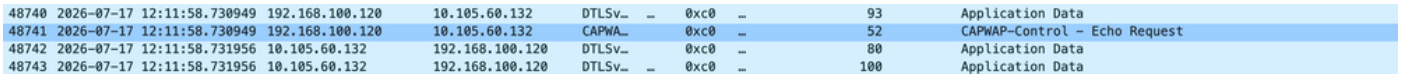
```
VK-WLC#show wireless stats ap history | i AP12


AP12                            889c.ad26.ea00  Joined     07/17/26 12:24:18  NA
AP12                            889c.ad26.ea00  Disjoined  07/17/26 12:12:19  NA
AP12                            889c.ad26.ea00  Joined     07/17/26 12:09:25  NA

AP12                          889c.ad26.ea00  Disjoined  07/17/26 12:08:33  NA                  Heart be
```

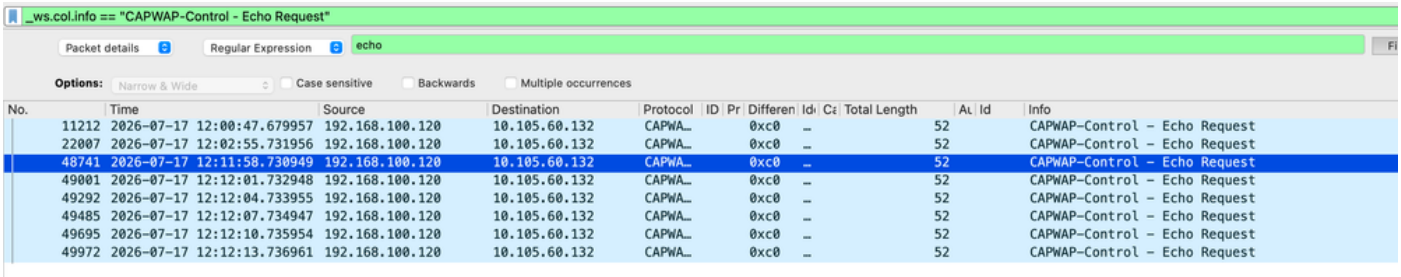


L'EPC WLC confirme que :

- Au même horodatage, la requête d'écho CAPWAP a réussi à atteindre le WLC.
- Le WLC a généré une réponse d'écho CAPWAP chiffrée.
- La réponse apparaît sous la forme Application Data, puisque le trafic de contrôle CAPWAP est protégé par le chiffrement DTLS.

L'EPC confirme donc que le contrôleur a transmis la réponse avec succès, ce qui élimine le WLC comme source du problème.





Les captures de paquets collectées sur le commutateur de liaison ascendante AP fournissent la preuve finale.

Les captures démontrent que :

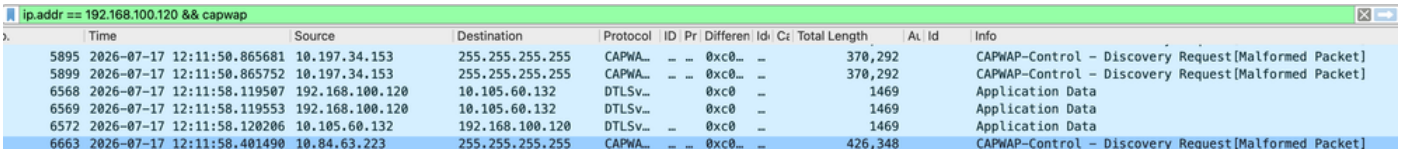
- Le commutateur a reçu avec succès la réponse d'écho chiffrée transmise par le WLC.

- La réponse est visible sous la forme Application Data, qui est attendue en raison du chiffrement DTLS.
- La réponse n'a pas été transmise au point d'accès.

Ceci explique pourquoi :

- L'AP n'a jamais reçu la réponse d'écho CAPWAP.
- Le point d'accès a continué à retransmettre les requêtes d'écho.
- Le compteur de retransmission a finalement dépassé le seuil configuré.
- L'AP a redémarré la machine d'état CAPWAP.

Les captures de paquets identifient clairement le commutateur comme le point où le trafic de contrôle CAPWAP a été interrompu


ip.addr == 192.168.100.120 && capwap

	Time	Source	Destination	Protocol	ID	Pr	Differen	Id	Ca	Total Length	Au	Id	Info
5895	2026-07-17 12:11:50.865681	10.197.34.153	255.255.255.255	CAPWA…	…	…	0xc0…	…		370,292			CAPWAP-Control - Discovery Request[Malformed Packet]
5899	2026-07-17 12:11:50.865752	10.197.34.153	255.255.255.255	CAPWA…	…	…	0xc0…	…		370,292			CAPWAP-Control - Discovery Request[Malformed Packet]
6568	2026-07-17 12:11:58.119507	192.168.100.120	10.105.60.132	DTLSv…			0xc0	…		1469			Application Data
6569	2026-07-17 12:11:58.119553	192.168.100.120	10.105.60.132	DTLSv…			0xc0	…		1469			Application Data
6572	2026-07-17 12:11:58.120206	10.105.60.132	192.168.100.120	DTLSv…	…		0xc0	…		1469			Application Data
6663	2026-07-17 12:11:58.401490	10.84.63.223	255.255.255.255	CAPWA…	…	…	0xc0…	…		426,348			CAPWAP-Control - Discovery Request[Malformed Packet]

Port de données CAPWAP (UDP 5247) abandonné sur le commutateur de liaison ascendante AP

Ce test valide le comportement du point d'accès lorsque le port de données CAPWAP (UDP 5247) est abandonné sur le commutateur de liaison ascendante du point d'accès tandis que le port de contrôle CAPWAP (UDP 5246) reste opérationnel.

Contrairement au scénario précédent, le point d'accès continue à maintenir la connexion de contrôle CAPWAP avec le WLC en échangeant avec succès des messages d'écho CAPWAP sur le port UDP 5246. Cependant, puisque les paquets de maintien de connexion de données CAPWAP ne peuvent pas terminer le voyage aller-retour, le point d'accès déclare finalement le chemin de données CAPWAP comme inaccessible et lance un redémarrage CAPWAP.

Les journaux de débogage AP confirment que le canal de contrôle CAPWAP est resté opérationnel tout au long du test.

Au début de la capture, les requêtes d'écho CAPWAP transmises sur le port UDP 5246 ont continué à recevoir des réponses d'écho valides du WLC, confirmant la communication ininterrompue du plan de contrôle.

Cependant, à 14:30:15, le point d'accès a transmis un paquet CAPWAP Data Keepalive sur le port UDP 5247. Comme aucune réponse Data Keepalive correspondante n'a été reçue, le point

d'accès a lancé le mécanisme de nouvelle tentative. Les retransmissions peuvent être observées à ces horodatages :

Horodatage	Événement
14:30:15	Transmission de la conservation initiale des données
14:30:19	Réessayer 1
14:30:25	Réessayer 2
14:30:37	Réessayer 3
14:30:49	Réessayer 4
14:31:01	Réessayer 5
14:31:13	Nouvelle tentative finale

Bien que les réponses d'écho CAPWAP aient continué à être reçues pendant cette période, le point d'accès n'a reçu aucune réponse pour les paquets de maintien de connexion de données. Après avoir épuisé les nouvelles tentatives, le point d'accès a signalé un délai de maintien de connexion de données non chiffrées et a initié un redémarrage de CAPWAP. Vers 14:31:16, le point d'accès a terminé la session CAPWAP existante et est retourné au processus de détection.

```
AP12#[*07/19/2026 14:30:15.9995] [TX]KEEPALIVE: Send to 10.105.60.132-5247
[*07/19/2026 14:30:15.9995] [TX]KEEPALIVE: Schedule for Retransmit in 3 sec sec_drop_count=0
[*07/19/2026 14:30:15.9996] capwap_ap_upstream_send ip: 192.168.100.123, port: 5272 => ip: 10.105.60.13
[*07/19/2026 14:30:19.0002] [TX]KEEPALIVE: Send to 10.105.60.132-5247
[*07/19/2026 14:30:19.0002] [TX]KEEPALIVE: Schedule for Retransmit in 6 sec sec_drop_count=0
[*07/19/2026 14:30:19.0003] capwap_ap_upstream_send ip: 192.168.100.123, port: 5272 => ip: 10.105.60.13
[*07/19/2026 14:30:25.0023] [TX]KEEPALIVE: Send to 10.105.60.132-5247
[*07/19/2026 14:30:25.0024] [TX]KEEPALIVE: Schedule for Retransmit in 12 sec sec_drop_count=0
[*07/19/2026 14:30:25.0024] capwap_ap_upstream_send ip: 192.168.100.123, port: 5272 => ip: 10.105.60.13
[*07/19/2026 14:30:37.0066] [TX]KEEPALIVE: Send to 10.105.60.132-5247
[*07/19/2026 14:30:37.0066] [TX]KEEPALIVE: Schedule for Retransmit in 12 sec sec_drop_count=0
[*07/19/2026 14:30:37.0067] capwap_ap_upstream_send ip: 192.168.100.123, port: 5272 => ip: 10.105.60.13
[*07/19/2026 14:30:49.0109] [TX]KEEPALIVE: Send to 10.105.60.132-5247
[*07/19/2026 14:30:49.0109] [TX]KEEPALIVE: Schedule for Retransmit in 12 sec sec_drop_count=0
[*07/19/2026 14:30:49.0109] capwap_ap_upstream_send ip: 192.168.100.123, port: 5272 => ip: 10.105.60.13
[*07/19/2026 14:31:01.0151] [TX]KEEPALIVE: Send to 10.105.60.132-5247
[*07/19/2026 14:31:01.0151] [TX]KEEPALIVE: Schedule for Retransmit in 12 sec sec_drop_count=0
[*07/19/2026 14:31:01.0152] capwap_ap_upstream_send ip: 192.168.100.123, port: 5272 => ip: 10.105.60.13
[*07/19/2026 14:31:13.0195] [TX]KEEPALIVE: Send to 10.105.60.132-5247
```

```
[*07/19/2026 14:31:13.0195] [TX]KEEPALIVE: Schedule for Retransmit in 12 sec sec_drop_count=0
[*07/19/2026 14:31:13.0196] capwap_ap_upstream_send ip: 192.168.100.123, port: 5272 => ip: 10.105.60.13
[*07/19/2026 14:31:16.0207] Warning, unencrypted data keepalive failed
[*07/19/2026 14:31:16.0207] Going to restart CAPWAP (reason : data keepalive not received)...
```

Pour le canal de données CAPWAP, les traces indiquent que le contrôleur n'a pas reçu les paquets Data Keepalive attendus. Finalement, après que le point d'accès ait déclaré l'échec Data Keepalive, le WLC a enregistré la fin de la session DTLS et l'événement de disjonction AP. La séquence observée dans les traces RA confirme que le contrôleur est resté opérationnel jusqu'à ce que le point d'accès se déconnecte volontairement en raison du délai d'expiration Data Keepalive.

Suivi RA avec AP ethernet mac :

<#root>

```
2026/07/19 14:31:16.842212773 {fman_rp_R0-0}{2}: [source] [20448]: (debug): ipc(mqipc/wncd_2/wncd-fmrp)
2026/07/19 14:31:16.842250177 {wncd_x_R0-2}{2}: [errmsg] [16638]: (note): %CAPWAPAC_SMGR_TRACE_MESSAGE-
2026/07/19 14:31:16.842251233 {wncd_x_R0-2}{2}: [capwapac-smgr-sess-fsm] [16638]: (note): Mac: 889c.ad2
```

Last Data Keep Alive Packet received 90 seconds ago.

```
2026/07/19 14:31:16.843318439 {wncmgrd_R0-0}{2}: [loadbalance-algo] [16262]: (note): Algo counter decre
2026/07/19 14:31:16.843318599 {wncd_x_R0-2}{2}: [wsa-core] [16638]: (debug): WSA AP EVT Create Populate
2026/07/19 14:31:16.843320409 {wncd_x_R0-2}{2}: [wsa-core] [16638]: (debug): WSA AP EVT Create Populate
```

DISJOIN - AP Mac:889c.ad26.ea00 , new ap disconnect reason 'DTLS close alert from peer' (26)

Suivi RA avec Radio mac :

<#root>

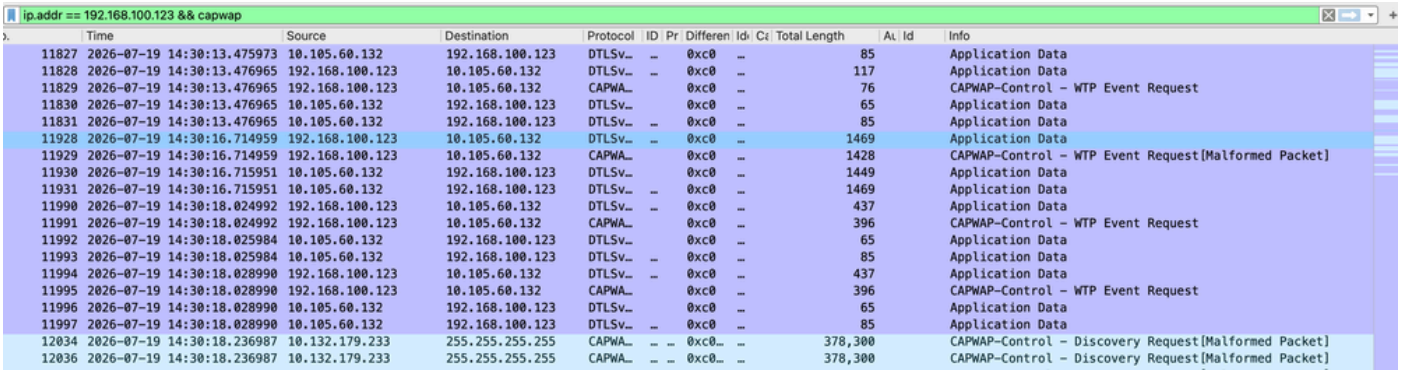
```
2026/07/19 14:31:16.737070835 {wncd_x_R0-2}{2}: [capwapac-smgr-sess] [16638]: (debug): Mac: 889c.ad26.e
2026/07/19 14:31:16.737072831 {wncd_x_R0-2}{2}: [capwapac-smgr-srvr] [16638]: (debug): Mac: 889c.ad26.e
2026/07/19 14:31:16.737076419 {wncd_x_R0-2}{2}: [msc-fsm] [16638]: (debug): @msc_event {"entity":"/capw
2026/07/19 14:31:16.737078137 {wncd_x_R0-2}{2}: [msc-fsm] [16638]: (debug): @msc_event {"entity":"/capw
2026/07/19 14:31:16.841870791 {wncd_x_R0-2}{2}: [ap-join-info-db] [16638]: (note): MAC: 889c.ad26.ea00
```

AP disconnect initiated. Name : AP12, Ethernet mac : e44e.2d2c.3d0c, Reason: DTLS close alert from peer,

```
2026/07/19 14:31:16.841890831 {wncd_x_R0-2}{2}: [capwapac-smgr-srvr] [16638]: (debug): MAC: 889c.ad26.e
```

EPC collecté sur le WLC valide le traitement des paquets côté contrôleur. Les captures confirment

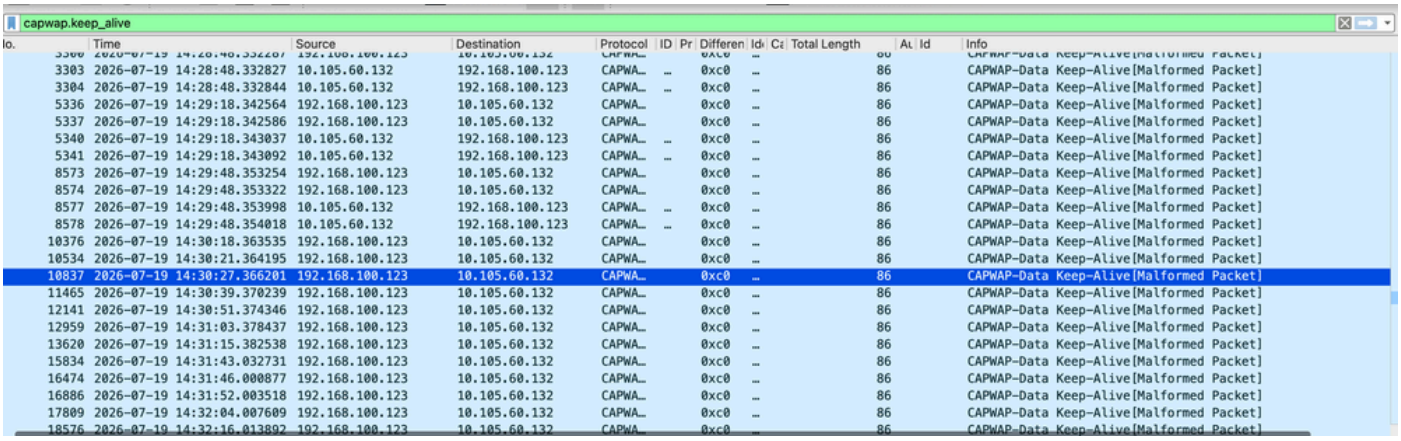
que les paquets de contrôle CAPWAP ont continué à être échangés avec succès tout au long du test, mais qu'il n'y a pas eu de paquet de maintien de la connexion des données reçu sur le wlc. Cette observation s'aligne sur les journaux de débogage du point d'accès et démontre que le mécanisme de maintien de la connexion des données a échoué malgré le fait que le canal de contrôle reste actif.


ip.addr == 192.168.100.123 && capwap

No.	Time	Source	Destination	Protocol	ID	Pr	Differen	Id	Ca	Total Length	At	Id	Info
11827	2026-07-19 14:30:13.475973	10.105.60.132	192.168.100.123	DTLSv…	…		0xc0	…		85			Application Data
11828	2026-07-19 14:30:13.476965	192.168.100.123	10.105.60.132	DTLSv…	…		0xc0	…		117			Application Data
11829	2026-07-19 14:30:13.476965	192.168.100.123	10.105.60.132	CAPWA…			0xc0	…		76			CAPWAP-Control - WTP Event Request
11830	2026-07-19 14:30:13.476965	10.105.60.132	192.168.100.123	DTLSv…			0xc0	…		65			Application Data
11831	2026-07-19 14:30:13.476965	10.105.60.132	192.168.100.123	DTLSv…	…		0xc0	…		85			Application Data
11928	2026-07-19 14:30:16.714959	192.168.100.123	10.105.60.132	DTLSv…	…		0xc0	…		1469			Application Data
11929	2026-07-19 14:30:16.714959	192.168.100.123	10.105.60.132	CAPWA…			0xc0	…		1428			CAPWAP-Control - WTP Event Request[Malformed Packet]
11930	2026-07-19 14:30:16.715951	10.105.60.132	192.168.100.123	DTLSv…			0xc0	…		1449			Application Data
11931	2026-07-19 14:30:16.715951	10.105.60.132	192.168.100.123	DTLSv…	…		0xc0	…		1469			Application Data
11990	2026-07-19 14:30:18.024992	192.168.100.123	10.105.60.132	DTLSv…	…		0xc0	…		437			Application Data
11991	2026-07-19 14:30:18.024992	192.168.100.123	10.105.60.132	CAPWA…			0xc0	…		396			CAPWAP-Control - WTP Event Request
11992	2026-07-19 14:30:18.025984	10.105.60.132	192.168.100.123	DTLSv…			0xc0	…		65			Application Data
11993	2026-07-19 14:30:18.025984	10.105.60.132	192.168.100.123	DTLSv…	…		0xc0	…		85			Application Data
11994	2026-07-19 14:30:18.028990	192.168.100.123	10.105.60.132	DTLSv…	…		0xc0	…		437			Application Data
11995	2026-07-19 14:30:18.028990	192.168.100.123	10.105.60.132	CAPWA…			0xc0	…		396			CAPWAP-Control - WTP Event Request
11996	2026-07-19 14:30:18.028990	10.105.60.132	192.168.100.123	DTLSv…			0xc0	…		65			Application Data
11997	2026-07-19 14:30:18.028990	10.105.60.132	192.168.100.123	DTLSv…	…		0xc0	…		85			Application Data
12034	2026-07-19 14:30:18.236987	10.132.179.233	255.255.255.255	CAPWA…	…	…	0xc0…	…		378,300			CAPWAP-Control - Discovery Request[Malformed Packet]
12036	2026-07-19 14:30:18.236987	10.132.179.233	255.255.255.255	CAPWA…	…	…	0xc0…	…		378,300			CAPWAP-Control - Discovery Request[Malformed Packet]

Commutateur :

Les captures de paquets collectées sur le commutateur de liaison ascendante AP indiquent si les paquets étaient présents sur le commutateur, mais qu'ils n'ont pas été transférés au wlc


capwap.keep_alive

No.	Time	Source	Destination	Protocol	ID	Pr	Differen	Id	Ca	Total Length	At	Id	Info
3303	2026-07-19 14:28:48.332827	10.105.60.132	192.168.100.123	CAPWA…	…		0xc0	…		86			CAPWAP-Data Keep-Alive[Malformed Packet]
3304	2026-07-19 14:28:48.332844	10.105.60.132	192.168.100.123	CAPWA…	…		0xc0	…		86			CAPWAP-Data Keep-Alive[Malformed Packet]
5336	2026-07-19 14:29:18.342564	192.168.100.123	10.105.60.132	CAPWA…			0xc0	…		86			CAPWAP-Data Keep-Alive[Malformed Packet]
5337	2026-07-19 14:29:18.342586	192.168.100.123	10.105.60.132	CAPWA…			0xc0	…		86			CAPWAP-Data Keep-Alive[Malformed Packet]
5340	2026-07-19 14:29:18.343037	10.105.60.132	192.168.100.123	CAPWA…	…		0xc0	…		86			CAPWAP-Data Keep-Alive[Malformed Packet]
5341	2026-07-19 14:29:18.343092	10.105.60.132	192.168.100.123	CAPWA…	…		0xc0	…		86			CAPWAP-Data Keep-Alive[Malformed Packet]
8573	2026-07-19 14:29:48.353254	192.168.100.123	10.105.60.132	CAPWA…			0xc0	…		86			CAPWAP-Data Keep-Alive[Malformed Packet]
8574	2026-07-19 14:29:48.353322	192.168.100.123	10.105.60.132	CAPWA…			0xc0	…		86			CAPWAP-Data Keep-Alive[Malformed Packet]
8577	2026-07-19 14:29:48.353998	10.105.60.132	192.168.100.123	CAPWA…	…		0xc0	…		86			CAPWAP-Data Keep-Alive[Malformed Packet]
8578	2026-07-19 14:29:48.354018	10.105.60.132	192.168.100.123	CAPWA…	…		0xc0	…		86			CAPWAP-Data Keep-Alive[Malformed Packet]
10376	2026-07-19 14:30:18.363535	192.168.100.123	10.105.60.132	CAPWA…			0xc0	…		86			CAPWAP-Data Keep-Alive[Malformed Packet]
10534	2026-07-19 14:30:21.364195	192.168.100.123	10.105.60.132	CAPWA…			0xc0	…		86			CAPWAP-Data Keep-Alive[Malformed Packet]
10837	2026-07-19 14:30:27.366201	192.168.100.123	10.105.60.132	CAPWA…			0xc0	…		86			CAPWAP-Data Keep-Alive[Malformed Packet]
11465	2026-07-19 14:30:39.370239	192.168.100.123	10.105.60.132	CAPWA…			0xc0	…		86			CAPWAP-Data Keep-Alive[Malformed Packet]
12141	2026-07-19 14:30:51.374346	192.168.100.123	10.105.60.132	CAPWA…			0xc0	…		86			CAPWAP-Data Keep-Alive[Malformed Packet]
12959	2026-07-19 14:31:03.378437	192.168.100.123	10.105.60.132	CAPWA…			0xc0	…		86			CAPWAP-Data Keep-Alive[Malformed Packet]
13620	2026-07-19 14:31:15.382538	192.168.100.123	10.105.60.132	CAPWA…			0xc0	…		86			CAPWAP-Data Keep-Alive[Malformed Packet]
15834	2026-07-19 14:31:43.032731	192.168.100.123	10.105.60.132	CAPWA…			0xc0	…		86			CAPWAP-Data Keep-Alive[Malformed Packet]
16474	2026-07-19 14:31:46.000877	192.168.100.123	10.105.60.132	CAPWA…			0xc0	…		86			CAPWAP-Data Keep-Alive[Malformed Packet]
16886	2026-07-19 14:31:52.003518	192.168.100.123	10.105.60.132	CAPWA…			0xc0	…		86			CAPWAP-Data Keep-Alive[Malformed Packet]
17809	2026-07-19 14:32:04.007609	192.168.100.123	10.105.60.132	CAPWA…			0xc0	…		86			CAPWAP-Data Keep-Alive[Malformed Packet]
18576	2026-07-19 14:32:16.013892	192.168.100.123	10.105.60.132	CAPWA…			0xc0	…		86			CAPWAP-Data Keep-Alive[Malformed Packet]

CAPWAP Data Keepalive Disable

Cette fonctionnalité a été introduite sous l'ID de bogue Cisco CSCvs66015

Cette fonctionnalité est utile pour le dépannage de ces scénarios.

1. Dépanner les déconnexions AP : Après avoir désactivé capwap data keep-alive, nous pouvons voir si AP est toujours déconnecté en raison de capwap control keep-alive et identifier la cause racine des déconnexions AP keep-alive.

2. Évitez les déconnexions dues à capwap data-keepalive comme solution de contournement jusqu'à ce que la raison de capwap data keep-alive soit identifiée.
3. Analyser les points d'accès dans les déploiements flexibles des filiales connectés via une liaison WAN à faible débit. Si nous voulons collecter des captures de paquets OTA en configurant l'un des AP pour être en mode renifleur, alors tous les paquets capturés sont transmis au WLC via la liaison WAN en utilisant le port de données capwap. Si cela se produit, la liaison WAN est submergée et la succursale complète en pâtit.
4. Si vous désactivez la conservation des données capwap pour le point d'accès en mode renifleur et créez une liste de contrôle d'accès dans la branche pour supprimer les paquets de données capwap de ce point d'accès concret, le point d'accès reste connecté puisque le contrôle capwap est correct et vous pouvez alors collecter l'OTA du port de commutation du point d'accès sans saturer la liaison WAN avec des captures de paquets.

Déclenchez COS-AP data keepalive enable/disable à partir du WLC 9800. Par défaut, le keepalive des données dans le WLC et le point d'accès est activé.

Commande WLC :

1) Affichez l'état avec la chaîne « Données non cryptées maintenues actives »

```
show ap config general
```

2) Activer/Désactiver le keepalive des données avec le nom ap

```
ap name AP-NAME keepalive
ap name AP-NAME no keepalive
```

Déclenchez COS-AP data keepalive enable/disable à partir du point d'accès lui-même. Par défaut, le keepalive des données dans le point d'accès est activé.

Commande AP :

1) Affichez l'état avec la chaîne "Unencryption Data Keep Alive"

```
show capwap client config
```

2) Activer/Désactiver la conservation des données dans AP

```
capwap ap unencrypted_data_keepalive enable
capwap ap unencrypted_data_keepalive disable
```

Exemple :

Vérification de keepalive désactivée au niveau du point d'accès :

```
AP12#capwap ap unencrypted_data_keepalive disable
```

<#root>

```
AP12#show capwap client configuration
```

```
AdminState                          : ADMIN_ENABLED(1)
Name                                : AP12
Location                            : default location
Primary controller name             : VK-WLC
Primary controller IP               : 10.105.60.132
Secondary controller name           : 9800-demo
Secondary controller IP             : 10.106.39.156
Tertiary controller name            :
ssh status                          : Enabled
ApMode                              : Local
ApSubMode                           : Not Configured
Link-Encryption                     : Disabled

Unencrypted Data Keep Alive         : Disabled

OfficeExtend AP                     : Disabled
Discovery Timer                     : 10
```

Débogages AP :

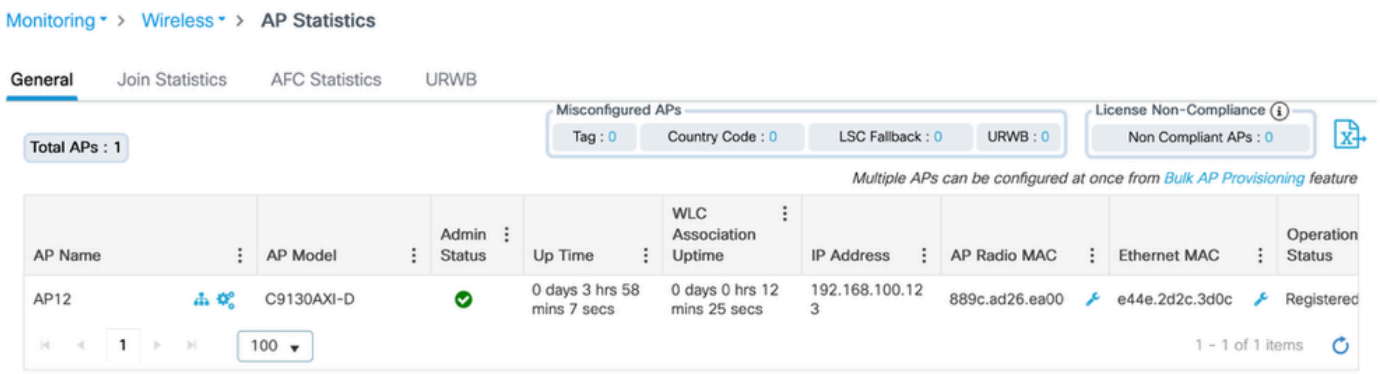
Dans le débogage d'AP, nous pouvons voir qu'il n'y a pas d'échange de paquets de données keepalive entre l'AP et le wlc, nous voyons juste l'échange de paquets de contrôle.

AP12#debug capwap client keepalive

```
AP12#[*07/19/2026 15:24:33.4087] Echo Request: Send count 0
[*07/19/2026 15:24:33.4087] [TX]Echo Request: Sent to 10.105.60.132
[*07/19/2026 15:24:33.4112] [RX]Echo Response from 10.105.60.132
```

```
[*07/19/2026 15:24:34.0006] [RX]Echo Response from 10.105.60.132 RttCount 1
[*07/19/2026 15:25:34.0032] Echo Request: Send count 1
[*07/19/2026 15:25:34.0032] [TX]Echo Request: Sent 2 Lost 2
[*07/19/2026 15:25:34.0056] [RX]Echo Response from 10.105.60.132
[*07/19/2026 15:25:34.0006] [RX]Echo Response from 10.105.60.132 RttCount 2
[*07/19/2026 15:27:34.0327] Echo Request: Send count 2
[*07/19/2026 15:27:34.0327] [TX]Echo Request: Sent 3 Lost 2
[*07/19/2026 15:27:34.0347] [RX]Echo Response from 10.105.60.132
[*07/19/2026 15:27:34.0004] [RX]Echo Response from 10.105.60.132 RttCount 1
[*07/19/2026 15:28:34.0025] Echo Request: Send count 3
[*07/19/2026 15:28:34.0025] [TX]Echo Request: Sent 4 Lost 2
[*07/19/2026 15:28:34.0050] [RX]Echo Response from 10.105.60.132
[*07/19/2026 15:28:34.0022] [RX]Echo Response from 10.105.60.132 RttCount 2
AP12#[*07/19/2026 15:29:43.5772] capwap_ap_upstream_send ip: 192.168.100.123, port: 5272 => ip: 10.105.
[*07/19/2026 15:30:04.0140] Echo Request: Send count 4
[*07/19/2026 15:30:04.0140] [TX]Echo Request: Sent 5 Lost 2
[*07/19/2026 15:30:04.0164] [RX]Echo Response from 10.105.60.132
[*07/19/2026 15:30:04.0011] [RX]Echo Response from 10.105.60.132 RttCount 1
[*07/19/2026 15:30:27.7695] capwap_ap_upstream_send ip: 192.168.100.123, port: 5272 => ip: 10.105.60.13
```

Même si les paquets de test d'activité des données étaient abandonnés, mais puisque nous avons désactivé la vérification de test d'activité des données, nous pouvons voir que l'AP reste stable sur le wlc sans être affecté par l'abandon des paquets de test d'activité.



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.