Configurez Wireshark et FreeRADIUS afin de déchiffrer l'analyseur sans fil 802.11 WPA2-Enterprise/EAP/dot1x en direct

Contenu

Introduction

Ce document décrit la procédure à suivre pour déchiffrer l'analyseur d'accès sans fil en direct (OTA) crypté Wi-Fi Protected Access 2 - Enterprise (WPA2-Enterprise) ou 802.1x (dot1x), avec toute méthode EAP (Extensible Authentication Protocol).

Il est relativement facile de décrypter la capture OTA 802.11 basée sur PSK/WPA2 personnel, à condition que les échanges EAPoL à quatre voies complets soient capturés. Cependant, la clé prépartagée (PSK) n'est pas toujours recommandée du point de vue de la sécurité. Il ne s'agit que d'une question de temps.

Par conséquent, de nombreuses entreprises choisissent dot1x avec Remote Authentication Dial-In User Service (RADIUS) comme meilleure solution de sécurité pour leur réseau sans fil.

Conditions préalables

Conditions requises

Cisco vous recommande de prendre connaissance des rubriques suivantes :

- FreeRADIUS avec **radsniff** installé
- Wireshark/Omnipeek ou tout logiciel capable de déchiffrer le trafic sans fil 802.11
- Privilège pour obtenir le secret partagé entre le serveur d'accès au réseau (NAS) et Authenticator
- Possibilité de capturer la capture de paquets radius entre le NAS et l'authentificateur à partir

de la première demande d'accès (du NAS à l'authentificateur) jusqu'à la dernière acceptation d'accès (de l'authentificateur au NAS) tout au long de la session EAP
- Possibilité d'effectuer une capture OTA (Over-the-Air) contenant des échanges EAPoL à quatre voies

Components Used

Les informations contenues dans ce document sont basées sur les versions de matériel et de logiciel suivantes :

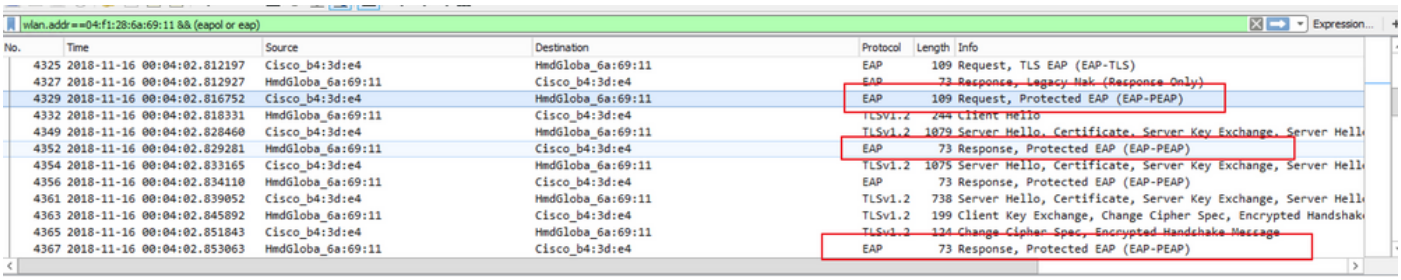
- Serveur Radius (FreeRADIUS ou ISE)
- Périphérique de capture en vol
- Appareil Apple macOS/OS X ou Linux

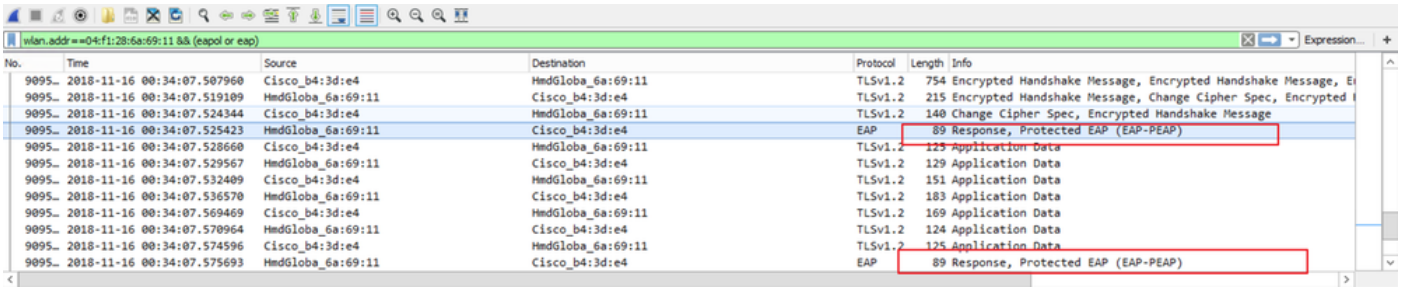
The information in this document was created from the devices in a specific lab environment. All of the devices used in this document started with a cleared (default) configuration. Si votre réseau est en ligne, assurez-vous de bien comprendre l'incidence possible des commandes.

Informations générales

Dans cet exemple, deux clés maître par paire (PMK) sont dérivées de paquets Radius capturés à partir de ISE 2.3, car le délai d'attente de session sur ce SSID est de 1800 secondes et la capture donnée ici est de 34 minutes (2040 secondes).

Comme l'illustre l'image, EAP-PEAP est utilisé comme exemple, mais cela peut être appliqué à toute authentification sans fil basée sur dot1x.





Procédure

Étape 1 : déchiffrement des PMK à partir du paquet Access-accept

Exécutez le **radsniff** contre la capture radius entre NAS et Authenticator afin d'extraire PMK. La raison pour laquelle deux paquets d'acceptation d'accès sont extraits au cours de la capture est que le délai d'expiration de la session est défini sur 30 minutes sur ce SSID particulier et la

capture dure 34 minutes. L'authentification est effectuée deux fois.

```
FRLU-M-51X5:pcaps frlu$ radsniff -I /Users/frlu/Downloads/radius_novlan_merged.pcapng -
s <shared-secret between NAS and Authenticator> -x

<snip>

2018-11-16 11:39:01.230000 (24) Access-Accept Id 172
/Users/frlu/Downloads/radius_novlan_merged.pcapng:10.66.79.42:32771 <- 10.66.79.36:1812 +0.000
+0.000

User-Name = "frlu_2"

State = 0x52656175746853657373696f6e3a3061343234663261303030303035653735626565303937323

Class =
0x434143533a3061343234663261303030303035653735626565303937323a4953452d322d332f333238323731323338
2f33303432

EAP-Message = 0x03c50004

Message-Authenticator = 0x38c67b9ba349842c9624889a45cabdfb

MS-MPPE-Send-Key = 0xa464cc15c0df8f09edc249c28711eb13a6db2d1a176f1196edcc707579fd6793

MS-MPPE-Recv-Key =
0xddb0b09a7d6980515825950b5929d02f236799f3e8a87f163c8ca41a066d8b3b<<<<<<<<<<<<<<<<<<PMK

Authenticator-Field = 0x6cd33b4d4dde05c07d9923e17ad6c218

<snip>

2018-11-16 11:39:01.470000 (48) Access-Accept Id 183
/Users/frlu/Downloads/radius_novlan_merged.pcapng:10.66.79.42:32771 <- 10.66.79.36:1812 +0.000
+0.000

User-Name = "frlu_2"

State = 0x52656175746853657373696f6e3a3061343234663261303030303035653735626565303937323

Class =
0x434143533a3061343234663261303030303035653735626565303937323a4953452d322d332f333238323731323338
2f33303434

EAP-Message = 0x03910004

Message-Authenticator = 0x81c572651679e15e54a900f3360c0aa9

MS-MPPE-Send-Key = 0xeae42cf7c6cd26371eee29856c51824fbb5bbb298874125928470114d009b5fb

MS-MPPE-Recv-Key =
0x7cce47eb82f48d8c0a91089ef7168a9b45f3d798448816a3793c5a4dfb1cfb0e<<<<<<<<<<<<<<<<<<PMK

Authenticator-Field = 0xa523dd9ec2ce93d19fe4fc2e21537a5d
```

Note: Supprimez toute étiquette de réseau local virtuel (VLAN) de la capture de paquets Radius, sinon **radsniff** ne reconnaîtra pas le fichier pcap d'entrée. Afin de supprimer toute balise VLAN, par exemple, editcap peut être utilisé.

Astuce : En règle générale, le temps d'exécution de la commande **radsniff** par rapport à un

fichier pcap RADIUS peut être compté comme une échelle de secondes. Cependant, si le **radsniff** est coincé dans cet état indiqué dans le journal, mettez en cascade cette capture de paquets (A) avec une autre capture de paquets plus longue (B) entre le même NAS et l'authentificateur. Ensuite, exécutez la commande radsniff sur le paquet en cascade (A+B). La seule exigence de la capture de paquets (B) est que vous pouvez exécuter la commande radsniff contre elle et voir le résultat verbose.

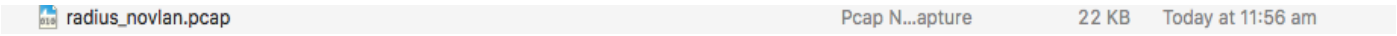
```
FRLU-M-51X5:pcaps frlu$ radsniff -I /Users/frlu/Downloads/radius_novlan.pcap -s Cisco123 -x

Logging all events

Sniffing on (/Users/frlu/Downloads/radius_novlan.pcap)
```

Dans cet exemple, la journalisation du plan de contrôle du contrôleur de réseau local sans fil (WLC) (A) capturée via la fonctionnalité de journalisation des paquets du WLC, est en cascade avec une capture plus longue à partir du TCPdump (B) d'ISE. La journalisation des paquets WLC est utilisée comme exemple car elle est généralement de très petite taille.

Journalisation des paquets WLC (A)



Tcpdump ISE (B)

Fusion (A+B)

Ensuite, exécutez le **radsniff** contre le pcap fusionné (A+B) et vous pourrez voir la sortie du verbose.

```
FRLU-M-51X5:pcaps frlu$ radsniff -I /Users/frlu/Downloads/radius_novlan_merged.pcapng -s
<shared-secret between NAS and Authenticator> -x

<snip>

2018-11-16 11:39:01.230000 (24) Access-Accept Id 172
/Users/frlu/Downloads/radius_novlan_merged.pcapng:10.66.79.42:32771 <- 10.66.79.36:1812 +0.000
+0.000

<snip>
```

Étape 2. Extraire les PMK.

La suppression de 0x dans chaque **MS-MPPE-Recv-Key** à partir de la sortie détaillée et des PMK nécessaires au décodage du trafic sans fil est ensuite présentée.

MS-MPPE-Recv-Key = 0xddb0b09a7d6980515825950b5929d02f236799f3e8a87f163c8ca41a066d8b3b

```
PMK:
ddb0b09a7d6980515825950b5929d02f236799f3e8a87f163c8ca41a066d8b3b
```

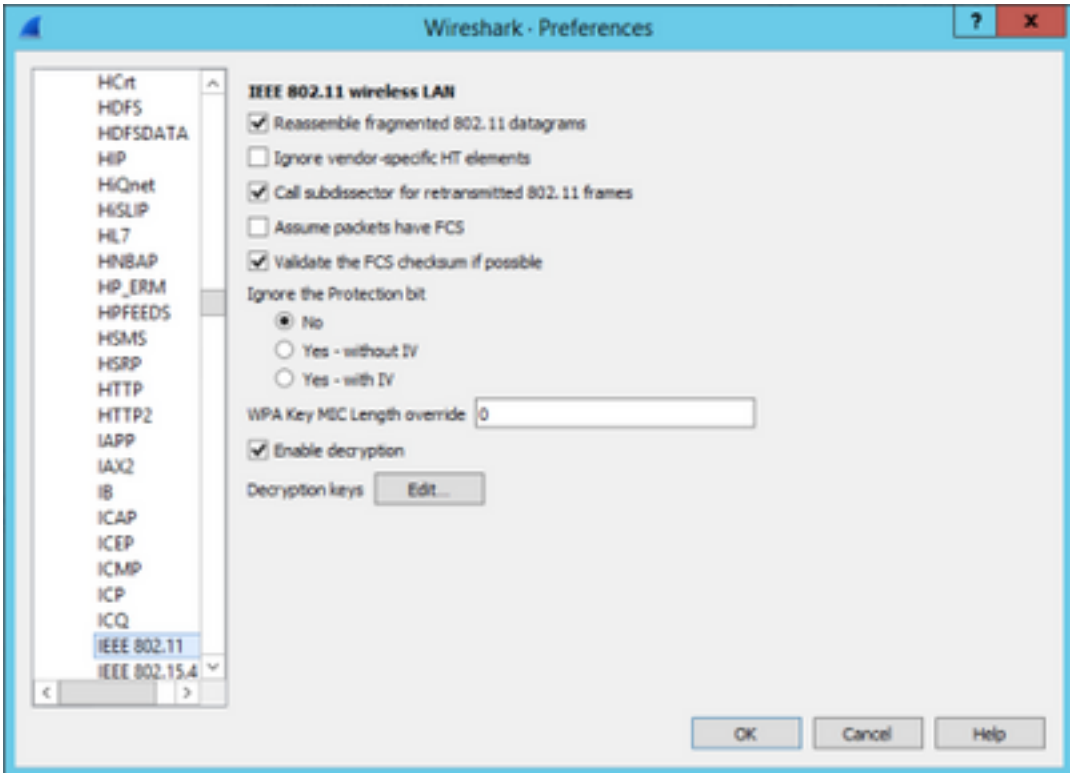
MS-MPPE-Recv-Key = 0x7cce47eb82f48d8c0a91089ef7168a9b45f3d79848816a3793c5a4dfb1cb

fb0e

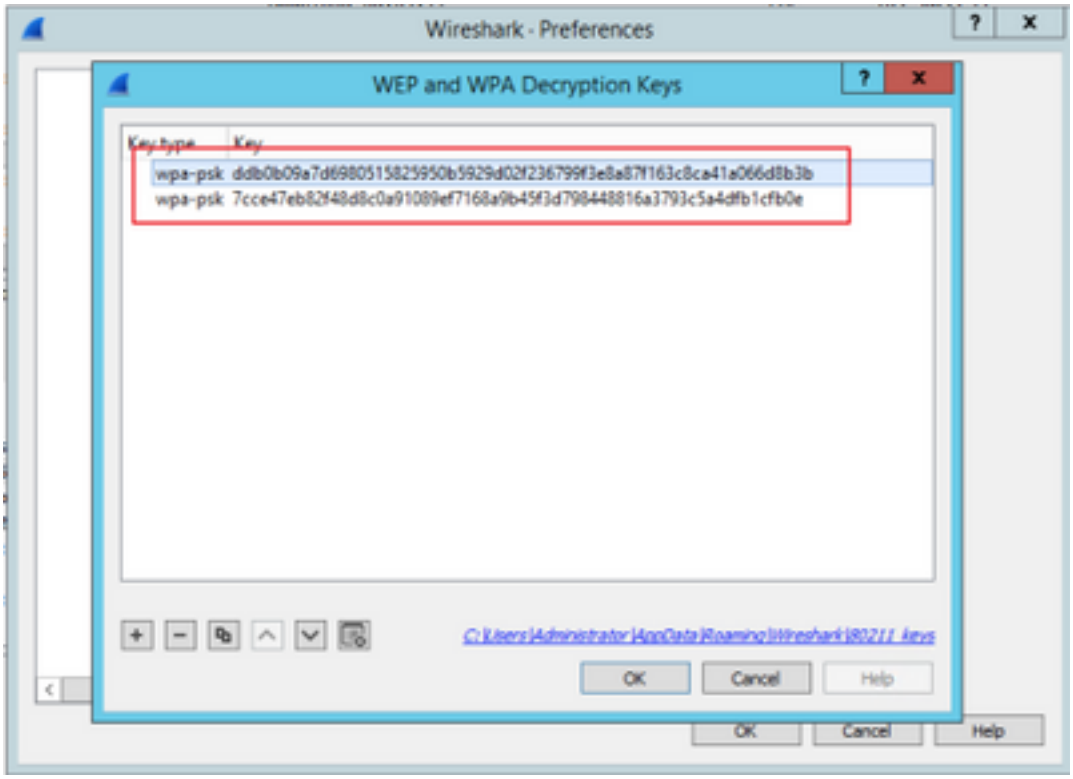
```
PMK:
7cce47eb82f48d8c0a91089ef7168a9b45f3d798448816a3793c5a4dfb1cfb0e
```

Étape 3. Décryptage de l'analyseur OTA

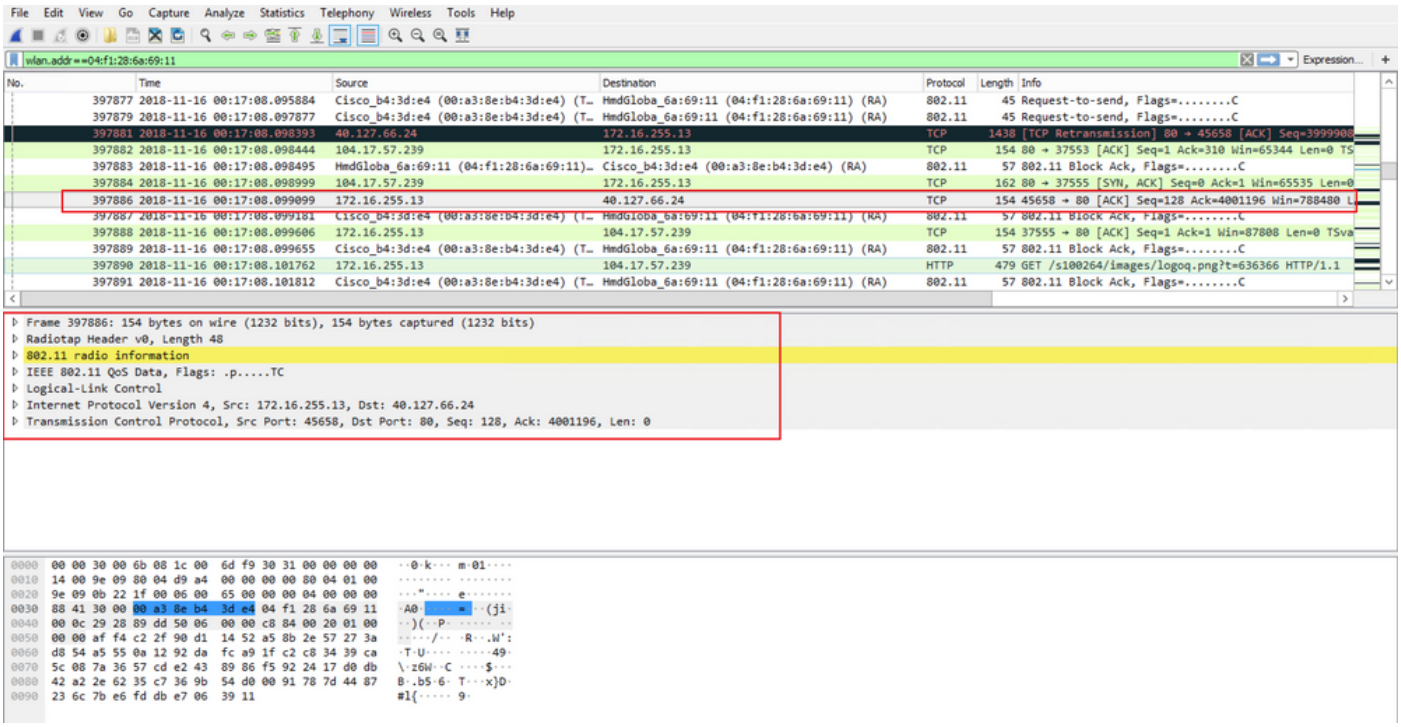
Accédez à **Wireshark > Préférences > Protocoles > IEEE 802.11.** Ensuite, cochez **Enable Decryption** et cliquez sur le bouton **Edit** en regard de **Decryption Keys**, comme indiqué dans l'image.



Ensuite, sélectionnez **wpa-psk** comme type de clé, puis placez les PMK dérivés dans le champ **Key**, puis cliquez sur **OK**. Une fois cette opération terminée, la capture OTA doit être déchiffrée et vous pouvez voir des informations de couche supérieure (3+).

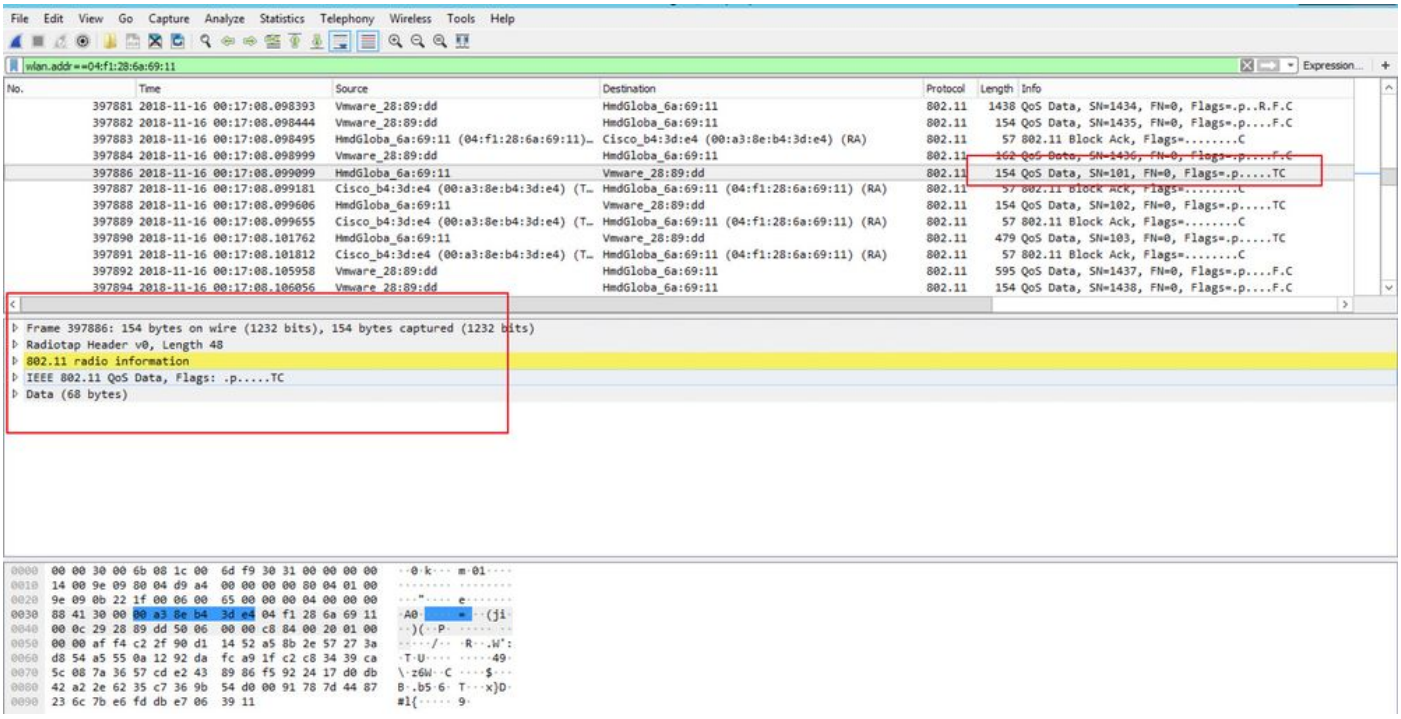


Exemple de paquet 802.11 déchiffré



Si vous comparez le deuxième résultat où le PMK n'est pas inclus, avec le premier résultat, où le PMK est inclus, le paquet 397886 est déchiffré en données QoS 802.11.

Exemple de paquet 802.11 chiffré



Attention : Vous pouvez rencontrer un problème avec Wireshark lors du déchiffrement, et dans ce cas, même si le bon PMK est fourni (ou si PSK est utilisé, SSID et PSK sont fournis), Wireshark ne déchiffre pas la capture OTA. La solution de contournement consiste à désactiver Wireshark et à le désactiver plusieurs fois jusqu'à ce que des informations de couche supérieure puissent être obtenues et que les paquets 802.11 ne soient plus affichés comme des données QoS, ou à utiliser un autre PC/Mac sur lequel Wireshark est installé.

Conseil : Un code C++ appelé pmkXtract est joint dans le premier billet de la rubrique Informations connexes. Les tentatives de compilation ont été réussies et un fichier exécutable est obtenu, mais le programme exécutable ne semble pas effectuer le déchiffrement correctement pour certaines raisons inconnues. En outre, un script Python qui tente d'extraire PMK est publié dans la zone des commentaires sur le premier billet, qui peut être exploré plus avant si les lecteurs sont intéressés.

Informations connexes

- Effacer le lien faible d'EAP : retirer les PMK WiFi de RADIUS avec pmkXtract
- Comment décoder Radius MS-MPPE-Recv-Key
- Support et documentation techniques - Cisco Systems