Configurare Wireshark e FreeRADIUS per decrittografare 802.11 WPA2-Enterprise/EAP/dot1x over-the-air Wireless Sniffer

Sommario

Introduzione

Questo documento descrive come decrittografare gli sniffer OTA (Wireless Over-The-Air) crittografati Wi-Fi Protected Access 2 - Enterprise (WPA2-Enterprise) o 802.1x (dot1x) con qualsiasi metodo EAP (Extensible Authentication Protocol).

Èrelativamente facile decrittografare l'acquisizione 802.11 OTA basata su PSK/WPA2-personal a condizione che vengano acquisiti tutti gli handshake EAP over LAN (EAPoL) a quattro vie. Tuttavia, non è sempre consigliabile utilizzare una chiave già condivisa (PSK) dal punto di vista della protezione. Cancellare una password hardcoded è solo una questione di tempo.

Molte aziende scelgono quindi il dot1x con il servizio RADIUS (Remote Authentication Dial-In User Service) come soluzione di protezione migliore per le reti wireless.

Prerequisiti

Requisiti

Cisco raccomanda la conoscenza dei seguenti argomenti:

- FreeRADIUS con **radsniff** installato
- Wireshark/Omnipeek o qualsiasi software in grado di decrittografare il traffico wireless 802.11
- Privilegio per ottenere il segreto condiviso tra Network Access Server (NAS) e Authenticator.

- Capacità di acquisire l'acquisizione di pacchetti radius tra NAS e autenticatore dalla prima richiesta di accesso (da NAS a autenticatore) all'ultima accettazione di accesso (da autenticatore a NAS) durante la sessione EAP
- Capacità di eseguire l'acquisizione OTA (Over-the-Air) contenente handshake EAPoL a quattro vie

Componenti usati

Le informazioni fornite in questo documento si basano sulle seguenti versioni software e hardware:

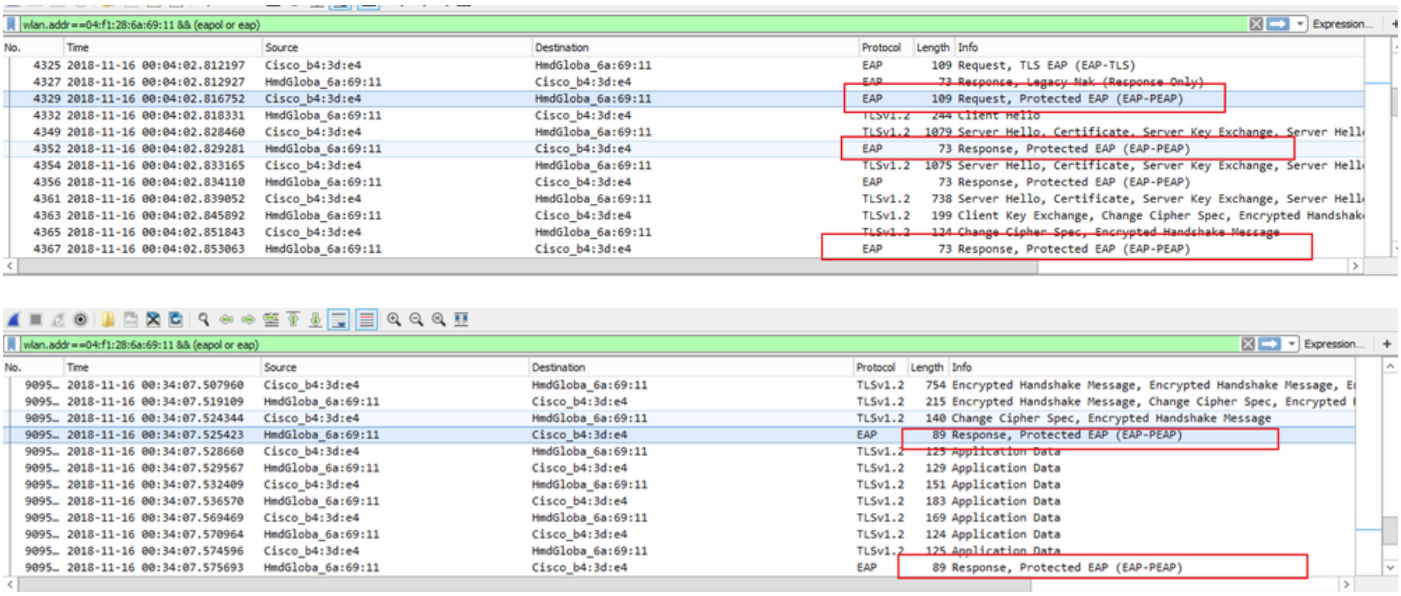
- Server Radius (FreeRADIUS o ISE)
- Dispositivo di acquisizione over-the-air
- Dispositivo Apple macOS/OS X o Linux

Le informazioni discusse in questo documento fanno riferimento a dispositivi usati in uno specifico ambiente di emulazione. Su tutti i dispositivi menzionati nel documento la configurazione è stata ripristinata ai valori predefiniti. Se la rete è operativa, valutare attentamente eventuali conseguenze derivanti dall'uso dei comandi.

Premesse

Nell'esempio, due chiavi master Pairwise (PMK) vengono derivate da pacchetti Radius acquisiti da ISE 2.3, poiché il timeout della sessione su questo SSID è di 1800 secondi e l'acquisizione qui specificata è lunga 34 minuti (2040 secondi).

Come mostrato nell'immagine, EAP-PEAP è usato come esempio, ma può essere applicato a qualsiasi autenticazione wireless basata su dot1x.



Procedura

Passaggio 1. Decrittografare le chiavi PMK dal pacchetto di accettazione dell'accesso.

Eseguire il **radsniff** sull'acquisizione del raggio tra NAS e Authenticator per estrarre PMK. Il motivo per cui due pacchetti access-accept vengono estratti durante l'acquisizione è che il timer di timeout della sessione è impostato su 30 minuti su questo particolare SSID e l'acquisizione è lunga 34 minuti. L'autenticazione viene eseguita due volte.

```
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 = 0x52656175746853657373696f6e3a30613432346632613030303030356537356265653039373 2

Class =
0x434143533a30613432346632613030303030356537356265653039373a4953452d322d332f333238323731323338
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 = 0x52656175746853657373696f6e3a30613432346632613030303030356537356265653039373 2

Class =
0x434143533a30613432346632613030303030356537356265653039373a4953452d322d332f333238323731323338
2f33303434

EAP-Message = 0x03910004

Message-Authenticator = 0x81c572651679e15e54a900f3360c0aa9

MS-MPPE-Send-Key = 0xeae42cf7c6cd26371eee29856c51824fbb5bbb298874125928470114d009b5fb

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

Authenticator-Field = 0xa523dd9ec2ce93d19fe4fc2e21537a5d
```

> **Nota:** Rimuovere il tag LAN virtuale (VLAN) dell'acquisizione pacchetti Radius, altrimenti **radsniff** non riconosce il file pcap di input. Ad esempio, per rimuovere un tag VLAN, è possibile usare editcap.

Suggerimento: In genere, il runtime del comando **radsniff** rispetto a un file pcap RADIUS può essere conteggiato come una scala di secondi. Tuttavia, se il **radsniff** è bloccato in questo stato indicato nel log, collegare in serie questa acquisizione pacchetto (A) con un'altra acquisizione pacchetto più lunga (B) tra lo stesso NAS e l'autenticatore. Eseguire quindi il comando radsniff sul pacchetto a cascata (A+B). L'unico requisito dell'acquisizione del pacchetto (B) è la possibilità di eseguire il comando radsniff su di esso e visualizzare un risultato dettagliato.

```
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)
```

Nell'esempio, il control plane logging (A) del Wireless Lan Controller (WLC), acquisito tramite la funzione di registrazione dei pacchetti WLC, viene acquisito in cascata con una durata maggiore dal protocollo TCPdump (B) di ISE. La registrazione dei pacchetti WLC è usata come esempio perché di solito è di dimensioni molto piccole.

Registrazione pacchetti WLC (A)

ISE Tcpdump (B)

Unito (A+B)

Eseguire quindi il **radsniff** sulla cuffia unita (A+B) per visualizzare l'output dettagliato.

```
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>
```

Passaggio 2. Estrarre i file PMK.

Viene quindi presentata l'eliminazione del campo 0x in ogni **MS-MPPE-Recv-Key** dall'output dettagliato e le PMK necessarie per la decodifica del traffico wireless.

MS-MPPE-Recv-Key = 0xdb0b09a7d6980515825950b5929d02f236799f3e8a87f163c8ca41a066d8b3b

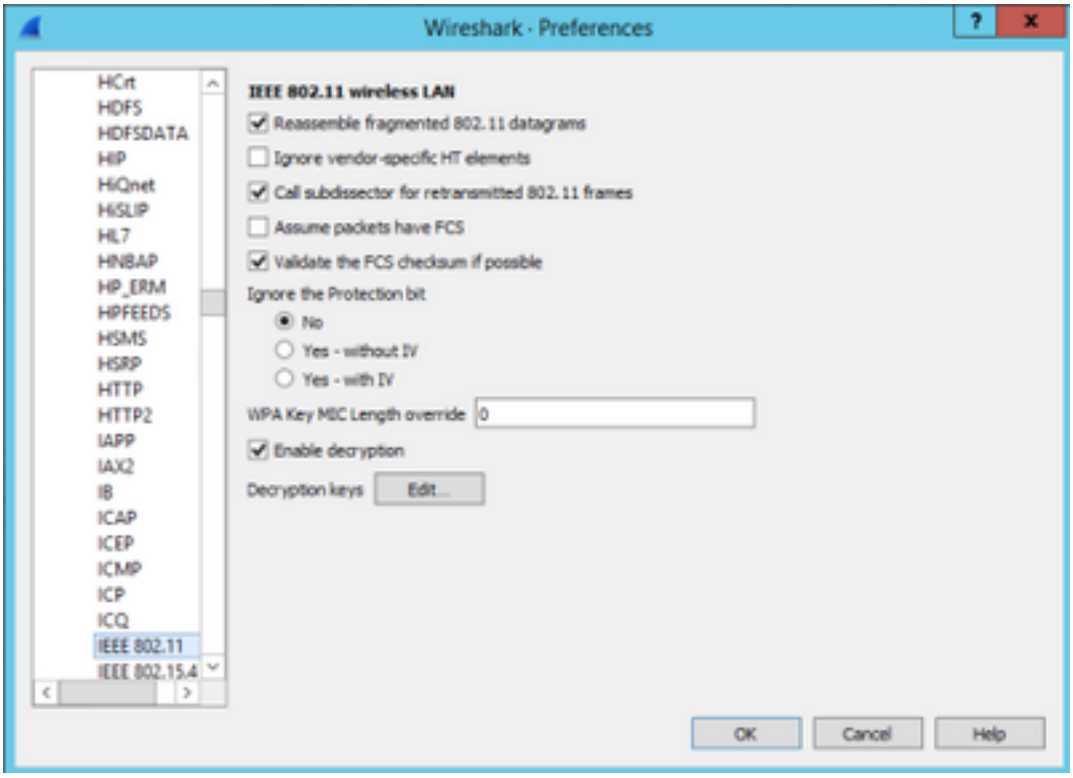
```
PMK:
ddb0b09a7d6980515825950b5929d02f236799f3e8a87f163c8ca41a066d8b3b
```

MS-MPPE-Recv-Key =
0x7cce47eb82f48d8c0a91089ef7168a9b45f3d798448816a3793c5a4dfb1cfb0e

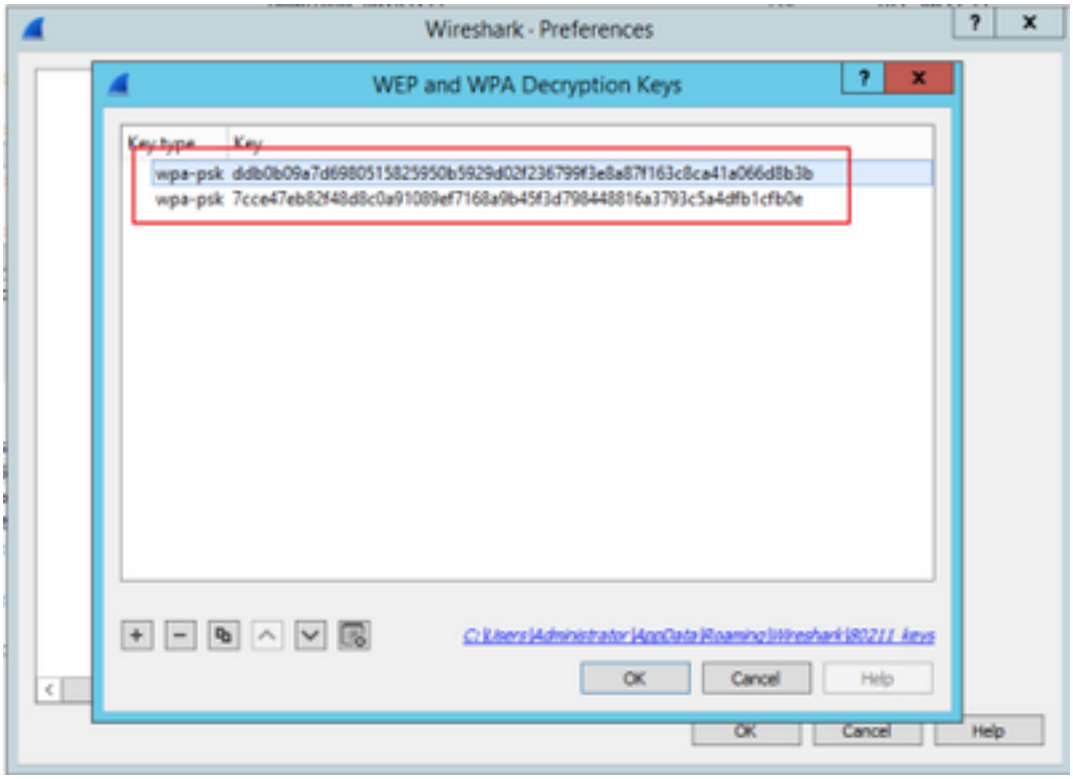
PMK:
7cce47eb82f48d8c0a91089ef7168a9b45f3d798448816a3793c5a4dfb1cfb0e

Passaggio 3. Decrittografare lo sniffer OTA.

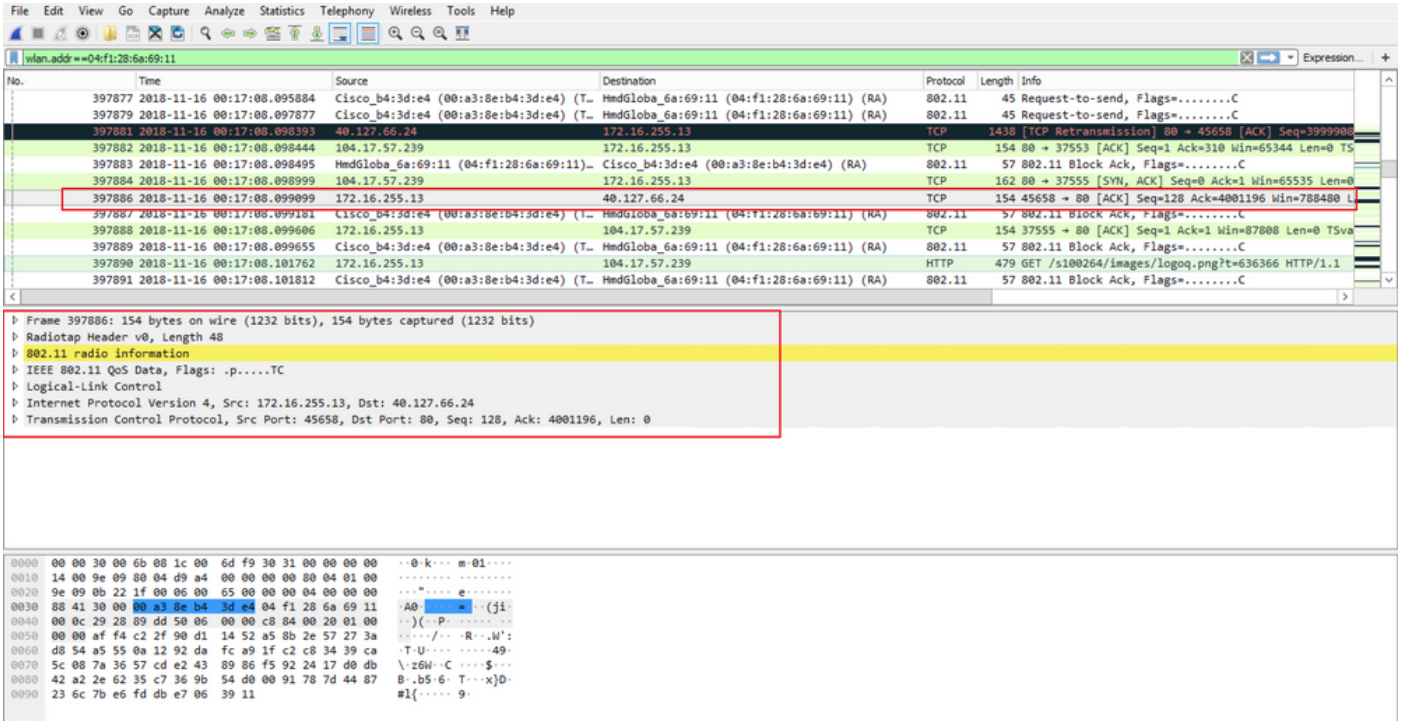
Passare a **Wireshark > Preferenze > Protocolli > IEEE 802.11.** Quindi selezionare **Abilita decrittografia** e fare clic sul pulsante **Modifica** accanto a **Chiavi di decrittografia**, come mostrato nell'immagine.



Selezionare quindi **wpa-psk** come tipo di chiave, inserire le PMK derivate nel campo **Chiave** e fare clic su **OK**. Al termine, l'acquisizione OTA deve essere decriptata ed è possibile visualizzare informazioni sul livello superiore (3+).

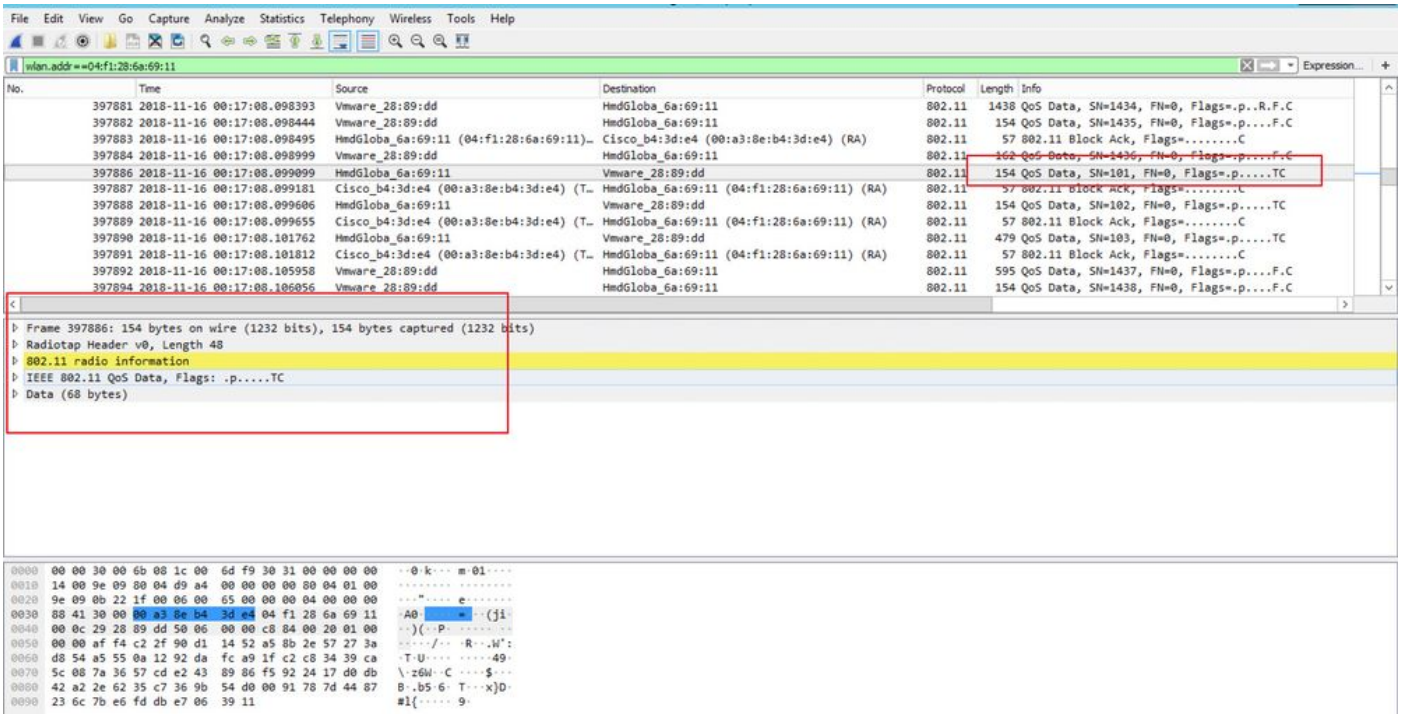


Esempio di pacchetto 802.11 decrittografato



Se si confronta il secondo risultato dove PMK non è incluso, con il primo risultato, dove PMK è incluso, il pacchetto 397886 viene decriptato come dati QoS 802.11.

Esempio di pacchetto 802.11 crittografato



Attenzione: È possibile che si verifichino problemi con Wireshark in fase di decrittografia e in tal caso, anche se viene fornito il PMK corretto (o se viene utilizzato il PSK, vengono forniti sia il SSID che il PSK), Wireshark non decripta l'acquisizione OTA. Per ovviare al problema, spegnere e riaccendere Wireshark alcune volte fino a quando non si ottengono informazioni sul layer più elevato e i pacchetti 802.11 non vengono più visualizzati come dati QoS, oppure utilizzare un altro PC/Mac con Wireshark installato.

Suggerimento: un codice C++ chiamato pmkXtract è allegato al primo post in Informazioni correlate. I tentativi di compilazione sono stati completati correttamente e viene ottenuto un file eseguibile, ma il programma eseguibile non sembra eseguire correttamente la decrittografia per motivi sconosciuti. Inoltre, uno script Python che tenta di estrarre PMK è pubblicato nell'area commenti al primo post, che può essere ulteriormente esplorato se i lettori sono interessati.

Informazioni correlate

- Modificare il collegamento debole di EAP - risucchiare i PMK WiFi da RADIUS con pmkXtract
- Come decodificare il raggio MS-MPPE-Recv-Key
- Documentazione e supporto tecnico – Cisco Systems