Wireless KRACK attack client side workaround and detection

Contents

Introduction

On October 16, a set of vulnerabilities widely known as KRACK affecting different protocols used in WiFi networks have been made public. They affect security protocols used on WPA/WPA2 networks, which could compromise the data privacy or integrity when it is transmitted over a wireless connection.

The practical level of impact varies significantly on each scenario, plus not all client side implementations are affected in the same way.
The attacks use different clever scenarios of “negative testing” where state transitions not properly defined on the wireless standards are tried, and in most cases, not handled properly by the affected device. It is not against the crypto algorithms used to protect WPA2, but on how the authentication and protocol negotiations are done during the securing of the wireless connection.

Most of the vulnerabilities scenarios have been reported for clients, where the possible typical attack will use fake Aps as “man in the middle” to intercept and inject specific frames during the security negotiations between the client and the real AP (CVE-2017-13077, CVE-2017-13078, CVE-2017-13079, CVE-2017-13080, CVE-2017-13081). These are the focus of this document

One scenario has been described attacking the AP infrastructures that provide 802.11r (FT) fast roaming services (CVE-2017-1382), which is fixed on recently released AireOS code

There are 4 remaining attacks against client specific protocols: STK, TDLS, WNM, which are not directly supported by the AireOS infrastructure (CVE-2017-13084 CVE-2017-13086 CVE-2017-13087 CVE-2017-13088), and are outside the scope of this document

In practical terms, an attacker could decrypt traffic for the affected session, or inject frames in one or two directions . It does not provide a way to decode previously existing traffic, prior to the attack, nor it will provide a mechanism to “obtain” the encryption kays of all devices in a given SSID or their PSK or 802.1x passwords

The vulnerabilities are real, and have a significant impact, but they do not mean that WPA2 protected networks are “affected forever”, as the issue can be fixed by improving the implementations on both client and AP side, to work properly in those *negative test scenarios* that are currently not handled in a robust way

What should a customer do:

- For AP side vulnerabilities: Upgrade is the recomended action if using FT. if FT is not needed for voice/video services, evaluate if FT feature should be disabled until the upgrade to fixed code is done. If using voice, evaluate if CCKM is feasible (client side needs to support), or upgrade to fixed code. If no FT/802.11r is in use, there is no need to upgrade at this time
- For client side vulnerabilities, improve your visibility: ensure that rogue detection is enabled, covering all channels, and a rule to report “managed SSID” as malicious is created. Additionally, implement EAPoL retry configurations changes that can limit or entirely block the attacks to be performed, as described in this docuemnt

The main reference advisory is at https://tools.cisco.com/security/center/content/CiscoSecurityAdvisory/cisco-sa-20171016-wpa. T

Components used

This document focuses on Wireless Controllers running releases 8.0 or later.

Requirements

Knowledge of the content covered by the security advisory mentioned above is required.

For the WPA KRACK attacks, there are 2 main actions we can take to protect the clients that have not been patched yet.

1. EAPoL (EAP over LAN) retry protection

2. Rogue detection and Access Point (AP) impersonation features, to detect if the attack tools are being used

EAPoL Attack protections

For vulnerabilities-2017-13077 to 81, it is relatively easy to prevent clients to be affected, using an EAPoL retry counter set to zero. This configuration is available in all WLC versions

Why this works

The attack needs at minimum one additional EAPoL retry generated by the authenticator during the 4 way handshake , or during the broadcast key rotation. If we block the generation of retries, the attack can't be applied against Pairwise Transient Key (PTK)/Groupwise Transient Key (GTK).

Possible impact

1. Clients which are slow or may drop initial processing of EAPoL M1 (i.e. the first message of the 4 way key exchange). This is seen on some small clients or some phones, which may receive the M1, and not be ready to process it after the dot1x authentication phase, or do it too slow to meet a short retransmission timer

2. Scenarios with bad RF environment, or WAN connections between AP and WLC, that may cause a packet drop at some point on transmission towards client.

In both scenarios, the outcome would be that an EAPoL exchange failure may be reported, and client will be deauthenticated, it will have to restart the association and authentication processes.

To decrease the likelihood of incurring into this issue, a longer timeout should be used (1000 msec), to allow more time for slow clients to respond. The default is 1000msec, but could have been changed to a lower value manually so it is to be verified.

Configuration

There are two mechanisms available to configure this change.

- Global, available in all releases
- Per WLAN, available from 7.6 to latest

The global option is simpler, and can be done in all releases, the impact is across all WLANs in the WLC.

Per WLAN configuration setting allows a more granular control, with the possibility to limit which SSID gets impacted, so the changes could be applied per device types, etc, if they are grouped on specific wlans. This is available from version 7.6

For example, it could be applied to a generic 802.1x WLAN, but not into a voice specific WLAN, where it may have a larger impact

#1 Global Config:

```
config advanced eap eapol-key-retries 0
```

(CLI only option)

The value can be validated with:

```
(2500-1-ipv6) >show advanced eap

EAP-Identity-Request Timeout (seconds)........... 30

EAP-Identity-Request Max Retries................. 2

EAP Key-Index for Dynamic WEP.................... 0

EAP Max-Login Ignore Identity Response........... enable

EAP-Request Timeout (seconds).................... 30

EAP-Request Max Retries.......................... 2

EAPOL-Key Timeout (milliseconds)................. 1000
```

```
EAPOL-Key Max Retries............................ 0

EAP-Broadcast Key Interval....................... 3600
```

#2 Per WLAN Config

X=WLAN ID

```
config wlan security eap-params enable X

config wlan security eap-params eapol-key-retries 0 X
```

How to identify if a client is deleted due to zero retransmissions

Client would be deleted due to max EAPoL retries reached, and deauthenticated. The retransmit count is 1, as the initial frame is counted

```
 *Dot1x_NW_MsgTask_6: Oct 19 12:44:13.524: 28:34:a2:82:41:f6 Sending EAPOL-Key Message to mobile 28:34:a2:82:41:f6
state PTKINITNEGOTIATING (message 3), replay counter 00.00.00.00.00.00.00.01
..
*osapiBsnTimer: Oct 19 12:44:14.042: 28:34:a2:82:41:f6 802.1x 'timeoutEvt' Timer expired for station 28:34:a2:82:41:f6 and for message = M3
*Dot1x_NW_MsgTask_6: Oct 19 12:44:14.042: 28:34:a2:82:41:f6 Retransmit failure for EAPOL-Key M3 to mobile 28:34:a2:82:41:f6, retransmit count 1, mscb deauth count 0
..
*Dot1x_NW_MsgTask_6: Oct 19 12:44:14.043: 28:34:a2:82:41:f6 Sent Deauthenticate to mobile on BSSID 58:ac:78:89:b4:19 slot 1(caller 1x_ptsm.c:602)
```

Rogue Detection

Several of the attack techniques for the vulnerabilities against the client PMK/GTK encryption, need to “present” a fake AP with the same SSID as the infrastructure AP, but operating on a different channel. This can be easily detected and the network administrator can take physical actions based on it, as it is a visible activity.

There are 2 ways proposed so far to do the EAPoL attacks :

- Faking infrastructure AP, in other words, acting as rogue AP, using same mac address, of a real AP, but on a different channel. Easy to do for the attacker but visible
- Injecting frames into a valid connection, forcing the client to react. This is a lot less visible, but detectable under some conditions, it may need very careful timing to be successful

The combination of AP impersonation features and rogue detection can detect if a "fake ap" is being placed in the network.

Configuration

- Validate that rogue detection is enabled on the access points. This is enabled by default, but could have been disabled manually by the admin, so it is to be verified.

- Create rule to flag rogues using “managed SSIDs” as malicious:

- Ensure that channel monitoring is set to “all channels” for both 802.11a/b networks. The base attack is designed to be near from RF perspective, the client, on a different channel from what is used on the infrastructure APs. This is why it is important to ensure that all possible channels are scanned:

AP impersonation

On default configuration, the infrastructure can detect if the attack tool is using one of our AP mac addresses. This is reported as an SNMP trap and would be indication that the attack is taking place.

```
Impersonation of AP with Base Radio MAC bc:16:65:13:a0:40 using source address of
bc:16:65:13:a0:40 has been detected by the AP with MAC Address: bc:16:65:13:a0:40 on its
802.11b/g radio whose slot ID is 0
```

References

Security advisory notice

Rogue Management in a Unified Wireless Network using v7.4 - Cisco

Cisco Wireless LAN Controller Configuration Best Practices - Cisco

Rogue Detection under Unified Wireless Networks - Cisco