Cisco WAP561 Wireless Access Point

Report generated by Tenable Nessus™ Thu, 19 Feb 2026 10:36:52 EST

TABLE OF CONTENTS

Vulnerabilities by Host

Vulnerabilities by Host

192.168.1.145

0	0	2	1	4
CRITICAL	HIGH	MEDIUM	LOW	INFO

Host Information

DNS Name:	wapc79f40.int[redacted].com
IP:	192.168.1.145
MAC Address:	E0:AC[redacted]
OS:	Dell iDRAC Controller, KYOCERA Printer, Linux Kernel 2.6

Vulnerabilities

51192 - SSL Certificate Cannot Be Trusted

Synopsis

The SSL certificate for this service cannot be trusted.

Description

The server's X.509 certificate cannot be trusted. This situation can occur in three different ways, in which the chain of trust can be broken, as stated below :

- First, the top of the certificate chain sent by the server might not be descended from a known public certificate authority. This can occur either when the top of the chain is an unrecognized, self-signed certificate, or when intermediate certificates are missing that would connect the top of the certificate chain to a known public certificate authority.

- Second, the certificate chain may contain a certificate that is not valid at the time of the scan. This can occur either when the scan occurs before one of the certificate's 'notBefore' dates, or after one of the certificate's 'notAfter' dates.

- Third, the certificate chain may contain a signature that either didn't match the certificate's information or could not be verified. Bad signatures can be fixed by getting the certificate with the bad signature to be re-signed by its issuer. Signatures that could not be verified are the result of the certificate's issuer using a signing algorithm that Nessus either does not support or does not recognize.

If the remote host is a public host in production, any break in the chain makes it more difficult for users to verify the authenticity and identity of the web server. This could make it easier to carry out man-in-the-middle attacks against the remote host.

See Also

https://www.itu.int/rec/T-REC-X.509/en

https://en.wikipedia.org/wiki/X.509

Solution

Purchase or generate a proper SSL certificate for this service.

Risk Factor

Medium

CVSS v3.0 Base Score

6.5 (CVSS:3.0/AV:N/AC:L/PR:N/UI:N/S:U/C:L/I:L/A:N)

CVSS v2.0 Base Score

6.4 (CVSS2#AV:N/AC:L/Au:N/C:P/I:P/A:N)

Plugin Information

Published: 2010/12/15, Modified: 2025/06/16

Plugin Output

tcp/443/www

```
The following certificate was part of the certificate chain
sent by the remote host, but it has expired :

|-Subject   : CN=192.168.1.[redacted]OU=Cisco /L=San Jose/ST=California/C=US/C=US/CN=192.168.1.[redacted]
O=Cisco
|-Not After : Dec 26 12:00:02 2019 GMT

The following certificate was at the top of the certificate
chain sent by the remote host, but it is signed by an unknown
certificate authority :

|-Subject : CN=192.168.1.[redacted]/OU=Cisco /L=San Jose/ST=California/C=US/C=US/CN=192.168.1[redacted]O=Cisco
|-Issuer  : CN=192.168.1.[redacted]/OU=Cisco /L=San Jose/ST=California/C=US
```

15901 - SSL Certificate Expiry

Synopsis

The remote server's SSL certificate has already expired.

Description

This plugin checks expiry dates of certificates associated with SSL- enabled services on the target and reports whether any have already expired.

Solution

Purchase or generate a new SSL certificate to replace the existing one.

Risk Factor

Medium

CVSS v3.0 Base Score

5.3 (CVSS:3.0/AV:N/AC:L/PR:N/UI:N/S:U/C:N/I:L/A:N)

CVSS v2.0 Base Score

5.0 (CVSS2#AV:N/AC:L/Au:N/C:N/I:P/A:N)

Plugin Information

Published: 2004/12/03, Modified: 2025/12/08

Plugin Output

tcp/443/www

```
The SSL certificate has already expired :

  Subject          : CN=192.168.1.[redacted], OU=Cisco , L=San Jose, ST=California, C=US, C=US,
 CN=192.168.1.[redacted], O=Cisco
  Issuer           : CN=192.168.1.[redacted], OU=Cisco , L=San Jose, ST=California, C=US
  Not valid before : Dec 31 12:00:02 1999 GMT
  Not valid after  : Dec 26 12:00:02 2019 GMT
```

69551 - SSL Certificate Chain Contains RSA Keys Less Than 2048 bits

Synopsis

The X.509 certificate chain used by this service contains certificates with RSA keys shorter than 2048 bits.

Description

At least one of the X.509 certificates sent by the remote host has a key that is shorter than 2048 bits. According to industry standards set by the Certification Authority/Browser (CA/B) Forum, certificates issued after January 1, 2014 must be at least 2048 bits.

Some browser SSL implementations may reject keys less than 2048 bits after January 1, 2014. Additionally, some SSL certificate vendors may revoke certificates less than 2048 bits before January 1, 2014.

Note that Nessus will not flag root certificates with RSA keys less than 2048 bits if they were issued prior to December 31, 2010, as the standard considers them exempt.

See Also

https://www.cabforum.org/wp-content/uploads/Baseline_Requirements_V1.pdf

Solution

Replace the certificate in the chain with the RSA key less than 2048 bits in length with a longer key, and reissue any certificates signed by the old certificate.

Risk Factor

Low

Plugin Information

Published: 2013/09/03, Modified: 2018/11/15

Plugin Output

tcp/443/www

```
The following certificates were part of the certificate chain
sent by the remote host, but contain RSA keys that are considered
to be weak :

|-Subject        : CN=192.168.1.[redacted]/OU=Cisco /L=San Jose/ST=California/C=US/C=US/CN=192.168.1.[redacted]/
O=Cisco
|-RSA Key Length : 1024 bits
```

10863 - SSL Certificate Information

Synopsis

This plugin displays the SSL certificate.

Description

This plugin connects to every SSL-related port and attempts to extract and dump the X.509 certificate.

Solution

n/a

Risk Factor

None

Plugin Information

Published: 2008/05/19, Modified: 2021/02/03

Plugin Output

tcp/443/www

```
Subject Name:

Common Name: 192.168.1.245
Organization Unit: Cisco
Locality: San Jose
State/Province: California
Country: US
Country: US
Common Name: 192.168.1.[redacted]
Organization: Cisco

Issuer Name:

Common Name: 192.168.1.[redacted]
Organization Unit: Cisco
Locality: San Jose
State/Province: California
Country: US

Serial Number: 1D 3F E1 8C

Version: 3

Signature Algorithm: SHA-1 With RSA Encryption

Not Valid Before: Dec 31 12:00:02 1999 GMT
Not Valid After: Dec 26 12:00:02 2019 GMT

Public Key Info:

Algorithm: RSA Encryption
Key Length: 1024 bits
```

```
Public Key: 00 C5 C2 92 22 8C 5B 98 22 2C C5 BD 15 26 97 9C 14 6D 68 7C
            A2 A0 77 CF 4C 62 D4 BB EC 27 24 0D 57 81 E2 33 5F 8A 8C 88
            44 0C F1 22 4C AE 5A A3 BE E9 FB FC FE 05 A6 86 5B 9F 64 69
            DB 89 CD C4 5A 95 17 E4 BD 20 E5 79 84 12 08 60 71 41 10 C1
            79 2A 53 96 C6 74 F0 3A 8A 08 53 30 A1 BD C4 D8 2B 9F 81 1F
            76 77 D9 C9 85 F0 26 48 E6 9E 82 39 83 F5 4B 7A 72 FD E3 34
            0C 43 AE 92 23 31 C8 9B 8F
Exponent: 01 00 01

Signature Length: 128 bytes / 1024 bits
Signature: 00 [redacted]

Fingerprints :

SHA-256 Fingerprint: [redacted]FF 17 F5 1B 5C 1B 56 98
                     35 31 77 88 51 30 DB 48 FA 10 65 BB
SHA-1 Fingerprint: 57[redacted] 8D 9D
MD5 Fingerprint: CB E0 [redacted]


PEM certificate :

-----BEGIN CERTIFICATE-----
[redacted]
 [...]
```

70544 - SSL Cipher Block Chaining Cipher Suites Supported

Synopsis

The remote service supports the use of SSL Cipher Block Chaining ciphers, which combine previous blocks with subsequent ones.

Description

The remote host supports the use of SSL ciphers that operate in Cipher Block Chaining (CBC) mode. These cipher suites offer additional security over Electronic Codebook (ECB) mode, but have the potential to leak information if used improperly.

See Also

https://www.openssl.org/docs/manmaster/man1/ciphers.html

http://www.nessus.org/u?cc4a822a

https://www.openssl.org/~bodo/tls-cbc.txt

Solution

n/a

Risk Factor

None

Plugin Information

Published: 2013/10/22, Modified: 2021/02/03

Plugin Output

tcp/443/www

```
Here is the list of SSL CBC ciphers supported by the remote server :

  High Strength Ciphers (>= 112-bit key)

    Name                          Code              KEX           Auth      Encryption              MAC
    ----------------------        ----------        ---           ----      ----------------------  ---
    ECDHE-RSA-AES128-SHA256       0xC0, 0x27        ECDHE         RSA       AES-CBC(128)
 SHA256
    ECDHE-RSA-AES256-SHA384       0xC0, 0x28        ECDHE         RSA       AES-CBC(256)
 SHA384
    RSA-AES128-SHA256             0x00, 0x3C        RSA           RSA       AES-CBC(128)
 SHA256
    RSA-AES256-SHA256             0x00, 0x3D        RSA           RSA       AES-CBC(256)
 SHA256

The fields above are :
```

```
{Tenable ciphername}
{Cipher ID code}
Kex={key exchange}
Auth={authentication}
Encrypt={symmetric encryption method}
MAC={message authentication code}
{export flag}
```

21643 - SSL Cipher Suites Supported

Synopsis

The remote service encrypts communications using SSL.

Description

This plugin detects which SSL ciphers are supported by the remote service for encrypting communications.

See Also

https://www.openssl.org/docs/man1.0.2/man1/ciphers.html

http://www.nessus.org/u?e17ffced

Solution

n/a

Risk Factor

None

Plugin Information

Published: 2006/06/05, Modified: 2024/09/11

Plugin Output

tcp/443/www

```
Here is the list of SSL ciphers supported by the remote server :
Each group is reported per SSL Version.

SSL Version : TLSv12
  High Strength Ciphers (>= 112-bit key)

    Name                          Code              KEX           Auth     Encryption              MAC
    ----------------------        ----------        ---           ----     --------------------    ---
    ECDHE-RSA-AES128-SHA256       0xC0, 0x2F        ECDHE         RSA      AES-GCM(128)
 SHA256
    ECDHE-RSA-AES256-SHA384       0xC0, 0x30        ECDHE         RSA      AES-GCM(256)
 SHA384
    RSA-AES128-SHA256             0x00, 0x9C        RSA           RSA      AES-GCM(128)
 SHA256
    RSA-AES256-SHA384             0x00, 0x9D        RSA           RSA      AES-GCM(256)
 SHA384
    ECDHE-RSA-AES128-SHA256       0xC0, 0x27        ECDHE         RSA      AES-CBC(128)
 SHA256
    ECDHE-RSA-AES256-SHA384       0xC0, 0x28        ECDHE         RSA      AES-CBC(256)
 SHA384
    RSA-AES128-SHA256             0x00, 0x3C        RSA           RSA      AES-CBC(128)
 SHA256
```

```
    RSA-AES256-SHA256             0x00, 0x3D       RSA           RSA      AES-CBC(256)
 SHA256

The fields above are :

  {Tenable ciphername}
  {Cipher ID code}
  Kex={key exchange}
  Auth={authentication}
  Encrypt={symmetric encryption method}
  MAC={message authentication code}
  {export flag}
```

57041 - SSL Perfect Forward Secrecy Cipher Suites Supported

Synopsis

The remote service supports the use of SSL Perfect Forward Secrecy ciphers, which maintain confidentiality even if the key is stolen.

Description

The remote host supports the use of SSL ciphers that offer Perfect Forward Secrecy (PFS) encryption. These cipher suites ensure that recorded SSL traffic cannot be broken at a future date if the server's private key is compromised.

See Also

https://www.openssl.org/docs/manmaster/man1/ciphers.html

https://en.wikipedia.org/wiki/Diffie-Hellman_key_exchange

https://en.wikipedia.org/wiki/Perfect_forward_secrecy

Solution

n/a

Risk Factor

None

Plugin Information

Published: 2011/12/07, Modified: 2021/03/09

Plugin Output

tcp/443/www

```
Here is the list of SSL PFS ciphers supported by the remote server :

  High Strength Ciphers (>= 112-bit key)

    Name                          Code              KEX           Auth      Encryption             MAC
    ----------------------        ----------        ---           ----      ---------------------  ---
    ECDHE-RSA-AES128-SHA256       0xC0, 0x2F        ECDHE         RSA       AES-GCM(128)
 SHA256
    ECDHE-RSA-AES256-SHA384       0xC0, 0x30        ECDHE         RSA       AES-GCM(256)
 SHA384
    ECDHE-RSA-AES128-SHA256       0xC0, 0x27        ECDHE         RSA       AES-CBC(128)
 SHA256
    ECDHE-RSA-AES256-SHA384       0xC0, 0x28        ECDHE         RSA       AES-CBC(256)
 SHA384

The fields above are :
```

```
{Tenable ciphername}
{Cipher ID code}
Kex={key exchange}
Auth={authentication}
Encrypt={symmetric encryption method}
MAC={message authentication code}
{export flag}
```