



Suites de cifrado TLS/SSL (Parte II)

Por mikiminoru el 15 sept 2020 con 0 comentarios

Good meowrning! Dado que en el anterior post estuvimos comentando en qué consistía cada sección de una suite de cifrado, en este vamos a listar un conjunto de herramientas que nos ayudan a comprobar las suites de cifrado que acepta un servidor SSL/TLS, y si dicha configuración contiene vulnerabilidades asociadas.

- Con herramientas online como:

- [SSL Labs](#)
- [crt.sh](#)
- [Toolbox](#)
- [Cipherli.st](#)

- Con herramientas a importar en nuestra máquina Kali/Parrot/Arch, tenemos las siguientes, que nos permiten solicitar un mayor nivel de detalle de las vulnerabilidades obtenidas.

- Burp Suite [SSL Scanner](#).
- Nmap

```
root@miau# nmap --script=ssl-enum-ciphers -p 443 target
```

```

PORT      STATE SERVICE REASON
443/tcp open  https  syn-ack ttl 64
ssl-enum-ciphers:
  TLSv1.0:
    ciphers:
      TLS_RSA_WITH_AES_128_CBC_SHA (rsa 2048) - A
      TLS_RSA_WITH_AES_256_CBC_SHA (rsa 2048) - A
      TLS_RSA_WITH_3DES_EDE_CBC_SHA (rsa 2048) - C
      TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA (secp256r1) - A
      TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA (secp256r1) - A
    compressors:
      NULL
    cipher preference: server
    warnings:
      64-bit block cipher 3DES vulnerable to SWEET32 attack
  TLSv1.1:
    ciphers:
      TLS_RSA_WITH_AES_128_CBC_SHA (rsa 2048) - A
      TLS_RSA_WITH_AES_256_CBC_SHA (rsa 2048) - A
      TLS_RSA_WITH_3DES_EDE_CBC_SHA (rsa 2048) - C
      TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA (secp256r1) - A
      TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA (secp256r1) - A
    compressors:
      NULL
    cipher preference: server
    warnings:
      64-bit block cipher 3DES vulnerable to SWEET32 attack
  TLSv1.2:
    ciphers:
      TLS_RSA_WITH_AES_128_CBC_SHA256 (rsa 2048) - A
      TLS_RSA_WITH_AES_128_CBC_SHA (rsa 2048) - A
      TLS_RSA_WITH_AES_256_CBC_SHA256 (rsa 2048) - A
      TLS_RSA_WITH_AES_256_CBC_SHA (rsa 2048) - A
      TLS_RSA_WITH_3DES_EDE_CBC_SHA (rsa 2048) - C
      TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA256 (secp256r1) - A
      TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA (secp256r1) - A
      TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA (secp256r1) - A
    compressors:
      NULL
    cipher preference: server
    warnings:
      64-bit block cipher 3DES vulnerable to SWEET32 attack
_ least strength: C

```

- [ssllscan](#)

```
root@miau# ssllscan target -p 443
```

```

Supported Server Cipher(s):
Preferred TLSv1.2 128 bits AES128-SHA256
Accepted TLSv1.2 128 bits AES128-SHA
Accepted TLSv1.2 256 bits AES256-SHA256
Accepted TLSv1.2 256 bits AES256-SHA
Accepted TLSv1.2 112 bits DES-CBC3-SHA
Accepted TLSv1.2 128 bits ECDHE-RSA-AES128-SHA256 Curve P-256 DHE 256
Accepted TLSv1.2 128 bits ECDHE-RSA-AES128-SHA Curve P-256 DHE 256
Accepted TLSv1.2 256 bits ECDHE-RSA-AES256-SHA Curve P-256 DHE 256
Preferred TLSv1.1 128 bits AES128-SHA
Accepted TLSv1.1 256 bits AES256-SHA
Accepted TLSv1.1 112 bits DES-CBC3-SHA
Accepted TLSv1.1 128 bits ECDHE-RSA-AES128-SHA Curve P-256 DHE 256
Accepted TLSv1.1 256 bits ECDHE-RSA-AES256-SHA Curve P-256 DHE 256
Preferred TLSv1.0 128 bits AES128-SHA
Accepted TLSv1.0 256 bits AES256-SHA
Accepted TLSv1.0 112 bits DES-CBC3-SHA
Accepted TLSv1.0 128 bits ECDHE-RSA-AES128-SHA Curve P-256 DHE 256
Accepted TLSv1.0 256 bits ECDHE-RSA-AES256-SHA Curve P-256 DHE 256

```

- [testssl](#)

root@miau# testssl target

```

* Certificates Information:
  Hostname sent for SNI:      target
  Number of certificates detected: 1

Certificate #0 ( _RSAPublicKey )
  SHA1 Fingerprint:          0dcf07f2705edff1a770827b344beadeb00f2cf0
  Common Name:               *.hack.me
  Issuer:                    CN=Danilo Oliveira Gomes,Rio de Janeiro, BR
  Serial Number:             4261100402479490415
  Not Before:                2019-09-11
  Not After:                 2021-09-11
  Public Key Algorithm:      _RSAPublicKey
  Signature Algorithm:       sha256
  Key Size:                  2048
  Exponent:                  65537
  DNS Subject Alternative Names: ["*.hack.me", "hack.me"]

Certificate #0 - Trust
  Hostname Validation:       OK - Certificate matches server hostname
  Android CA Store (9.0.0-r9): OK - Certificate is trusted
  Apple CA Store (iOS 13, iPadOS 13, macOS 10.15, watchOS 6, and tvOS 13): OK - Certificate is trusted
  Java CA Store (jdk-13.0.2): OK - Certificate is trusted
  Mozilla CA Store (2020-06-21): OK - Certificate is trusted
  Windows CA Store (2020-05-04): OK - Certificate is trusted
  Symantec 2020 Deprecations: OK - Not a Symantec-issued certificate
  Received Chains:           * hack.me ==> CN=Danilo Oliveira Gomes, Rio de Janeiro, BR ==> CN=Global Root Certificate Authority, BR
  Verified Chains:            * hack.me ==> CN=Global Root Certificate Authority, BR ==> CN=Global Root Certificate Authority, BR
  Received Chain Contains Anchor: OK - Anchor certificate not sent
  Received Chain Order:      OK - Order is valid
  Verified Chain Contains SHA1: OK - No SHA1-signed certificate in the verified certificate chain

Certificate #0 - Extensions
  OCSP Must-Staple:          NOT SUPPORTED - Extension not found
  Certificate Transparency:   OK - 3 SCTs included

```

- [sslyze](#)

root@miau# sslyze --regular target

```
* Certificates Information:
  Hostname sent for SNI:         localhost
  Number of certificates detected: 1

Certificate #0 ( _RSAPublicKey )
  SHA1 Fingerprint:              00cf07f2705eddf14a770827b344beadeb00f2cf0
  Common Name:                   *.hack.me
  Issuer:                         CN = Domain Controller Corpus Ricane, C=Germany, OU = IT
  Serial Number:                 4361190492679490415
  Not Before:                    2019-09-11
  Not After:                     2021-09-11
  Public Key Algorithm:          _RSAPublicKey
  Signature Algorithm:           sha256
  Key Size:                      2048
  Exponent:                      65537
  DNS Subject Alternative Names: 1 = *.hack.me, 'localhost'

Certificate #0 - Trust
  Hostname Validation:           OK - Certificate matches server hostname
  Android CA Store (9.0.0-99):   OK - Certificate is trusted
  Apple CA Store (10S 13, 1P0006 13, macOS 10.15, watchOS 6, and tvOS 13): OK - Certificate is trusted
  Java CA Store (jdk-13.0.2):    OK - Certificate is trusted
  Mozilla CA Store (2020-06-21): OK - Certificate is trusted
  Windows CA Store (2020-05-04): OK - Certificate is trusted
  Symantec 2020 Deprecation:     OK - Not a Symantec-issued certificate
  Received Chain:                = *.hack.me == CN = Domain Controller Corpus Ricane, C=Germany, OU = IT ==> CN = Hackle Root Certificate Authority = OK
  Verified Chain Contains Anchor: OK - Anchor certificate not sent
  Received Chain Order:          OK - Order is valid
  Verified Chain contains SHA1:   OK - No SHA1-signed certificate in the verified certificate chain

Certificate #0 - Extensions
  OCSP Must-Staple:              NOT SUPPORTED - Extension not found
  Certificate Transparency:      OK - 3 SCTs included
```

En la última parte de este hilo hablaremos de las vulnerabilidades que obtenemos ante configuraciones incorrectas :)

Muchos maullidos!

M

 SSL , TLS

Otros artículos relacionados

TLS1.3: ¿Qué hay de nuevo? (Parte I)

30 DE OCTUBRE DE 2020



Suites de TLS/SSL (Parte III)

21 DE SEPTIEMBRE DE 2020