Engenharia de Segurança

Trabalho Prático 2

Afonso Fontes (pg35389)

Bruno Carvalho (a67847)

Mariana Carvalho (a67635)

26 de Fevereiro de 2018

Universidade do Minho

Blind Signatures

O exercício relativo a blind signatures está resolvido no repositório, na directoria *p1_blindSignature*.

Protocolo SSL/TLS

Pergunta 2.1

Escolha quatro sites de empresas não bancárias cotadas na Bolsa Portuguesa e pertencentes ao PSI 20

i Anexe os resultados do SSL Server test à sua resposta.

Os resultados dos sites das quatro empresas escolhidas (Amorim, EDP Renováveis, REN, Sonae) encontram-se em formato pdf no repositório, na directoria *resultados_ssl*.

ii Analise o resultado do SSL Server test relativo ao site escolhido com pior rating. Que comentários pode fazer sobre a sua segurança. Porquê?

O site que tem pior rating é de longe o site da Sonae (*www.sonae.pt*), com um rating de **C**.

O site tem vários problemas de segurança, por exemplo, o não suporte do protocolo TLS mais recente, sendo suportado apenas o protocolo TLS 1.0. Esta versão não contem várias melhorias e correções de segurança introduzidas nas versões 1.1 e 1.2 do protocolo, por exemplo, a utilização de funções de hash mais modernas (SHA-256) como funções pseudo-aleatórias, em vez de funções de hash já ultrapassadas (MD5, SHA-1) [1]. Para além disso, a versão 1.0 do protocolo é vulnerável a ataques ao modo de operação CBC, mais concretamente, os últimos bits da última mensagem cifrada são usados como vector de inicialização da cifra, o que em alguns cenários poderá permitir a um atacante ter conhecimento indevido do vector de inicialização utilizado e comprometer a segurança da cifra [2]. Esta vulnerabilidade foi corrigida na versão 1.1 do protocolo [3].

É também importante referir que o site ainda suporta a cifra RC4 que é demonstravelmente insegura e o seu uso foi inclusive proibido pela IETF [4].

iii É natural que tenha reparado na seguinte informação: "OpenSSL CCS vuln. (CVE-2014-0224)"na secção de detalhe do protocolo. O que significa, para efeitos práticos?

Significa que a versão do openSSL utilizada no servidor não verifica e restringe adequadamente as mensagens do tipo *ChangeCipherSpec*, o que permite forçar o cliente e servidor a utilizar chaves fracas/vulneráveis para comunicação. Mais concretamente, um atacante pode prever o material utilizado para gerar as chaves, enviando mensagens inválidas durante o processo de handshake [5]. Esta vulnerabilidade pode ser explorada através de um ataque man-in-the-middle para decifrar ou falsificar mensagens.

Apenas algumas versões do openSSL são afectadas por esta vulnerabilidade e as versões mais recentes (a partir da 1.0.1h) não o são, pelo que é recomendável actualizar o openSSL a qualquer serviço que se encontre actualmente vulnerável.

Protocolo SSH

Pergunta 3.1

Anexe os resultados ssh-audit à sua resposta

Neste ponto escolhemos dois servidores ssh de empresas cotadas na Bolsa Portuguesa, respetivamente a **NOS** e a **PT Comunicações**, como proposto no enunciado com recurso à *https://www.shodan.io/* pesquisamos por serviços correspondentes a cada uma das entidades à escuta na porta 22. No caso da **NOS** escolhemos o servidor identificado pelo **IP** 88.157.176.86 , enquanto que no caso da **PT Comunicações** foi escolhido o servidor com o **IP** 2.81.46.154. Podemos visualizar o resultado após aplicar o **ssh audit** a cada um dos **IPs** indicados em Anexo.

Indique o software e versão utilizada pelos servidores ssh

Como podemos visualizar através dos resultados do ssh-audit aplicados ao IP 88.157.176.86 (NOS) o software utilizado é **Cisco IOS/PIX sshd 1.25** enquanto que o software utilizado no IP 2.81.46.154 é o **OpenSSH 6.6.1p1**. Ambos utilizam a versão 2.0 do protocolo SSH.

Qual dessas versões de software tem mais vulnerabilidades?

Após pesquisar por vulnerabilidades presentes nos diferentes softwares apuramos que aquele que tem maior número de vulnerabilidades conhecidas é o **OpenSSH 6.6.1p1**.

E qual tem a vulnerabilidade mais grave (de acordo com o CVSS score identificado no CVE details)?

Aquele que apresentou a vulnerabilidade mais grave foi o **OpenSSH 6.6.1p1** identificada pelo **CVE-2016-1908** com um **CVSS score (version 3)** de 9.8, caracterizada como uma vulnerabilidade crítica.

Para efeitos práticos, a vulnerabilidade indicada no ponto anterior é grave? Porquê?

A vulnerabilidade indicada no ponto anterior é grave, pois de acordo com a descrição obtida pela consulta do **CVE** indicado, esta vulnerabilidade pode permitir que uma aplicação 'X11' maliciosa use esta falha para estabelecer uma conexão confiável com o servidor 'X11' local.

Anexos

- PT.png

```
user@CSI:~/Desktop/Tools/ssh-audit$ python ssh-audit.py 2.81.46.154
# general
(gen) banner: SSH-2.0-OpenSSH_6.6.1p1 Ubuntu-2ubuntu2.10
(gen) software: OpenSSH 6.6.1p1
(gen) compatibility: OpenSSH 6.5-6.6, Dropbear SSH 2013.62+ (some functionality from 0.52)
(gen) compression: enabled (zlib@openssh.com)

# key exchange algorithms
(kex) curve25519-sha256@libssh.org          -- [info] available since OpenSSH 6.5, Dropbear SSH 2013.62
(kex) ecdh-sha2-nistp256                    -- [fail] using weak elliptic curves
                                            `- [info] available since OpenSSH 5.7, Dropbear SSH 2013.62
(kex) ecdh-sha2-nistp384                    -- [fail] using weak elliptic curves
                                            `- [info] available since OpenSSH 5.7, Dropbear SSH 2013.62
(kex) ecdh-sha2-nistp521                    -- [fail] using weak elliptic curves
                                            `- [info] available since OpenSSH 5.7, Dropbear SSH 2013.62
(kex) diffie-hellman-group-exchange-sha256  -- [warn] using custom size modulus (possibly weak)
                                            `- [info] available since OpenSSH 4.4
(kex) diffie-hellman-group-exchange-sha1    -- [fail] removed (in server) since OpenSSH 6.7, unsafe algorithm
                                            `- [warn] using weak hashing algorithm
                                            `- [info] available since OpenSSH 2.3.0
(kex) diffie-hellman-group14-sha1           -- [warn] using weak hashing algorithm
                                            `- [info] available since OpenSSH 3.9, Dropbear SSH 0.53
(kex) diffie-hellman-group1-sha1            -- [fail] removed (in server) since OpenSSH 6.7, unsafe algorithm
                                            `- [fail] disabled (in client) since OpenSSH 7.0, logjam attack
                                            `- [warn] using small 1024-bit modulus
                                            `- [warn] using weak hashing algorithm
                                            `- [info] available since OpenSSH 2.3.0, Dropbear SSH 0.28

# host-key algorithms
(key) ssh-rsa                               -- [info] available since OpenSSH 2.5.0, Dropbear SSH 0.28
(key) ssh-dss                               -- [fail] removed (in server) and disabled (in client) since OpenSSH 7.0,
lgorithm
                                            `- [warn] using small 1024-bit modulus
                                            `- [warn] using weak random number generator could reveal the key
                                            `- [info] available since OpenSSH 2.1.0, Dropbear SSH 0.28
(key) ecdsa-sha2-nistp256                   -- [fail] using weak elliptic curves
                                            `- [warn] using weak random number generator could reveal the key
                                            `- [info] available since OpenSSH 5.7, Dropbear SSH 2013.62
```

Figura 1: ssh-audit aplicado ao servidor da PT Comunicações (1)

-PT.png

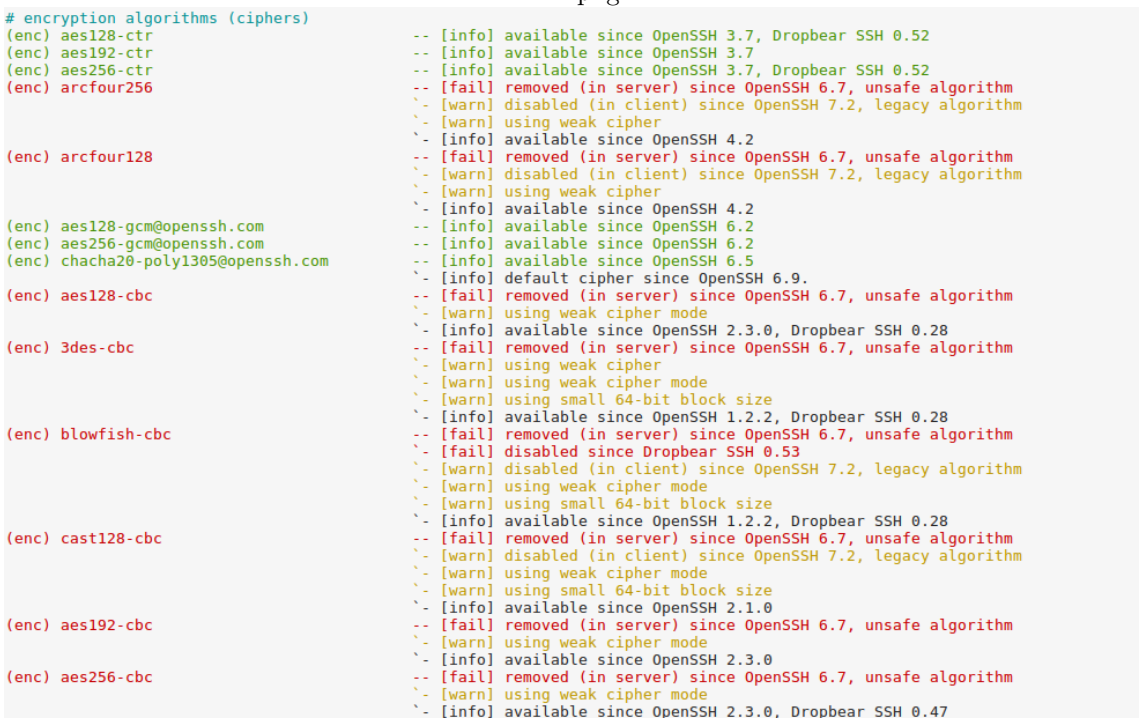

Figura 2: ssh-audit aplicado ao servidor da PT Comunicações (2)

- PT.png

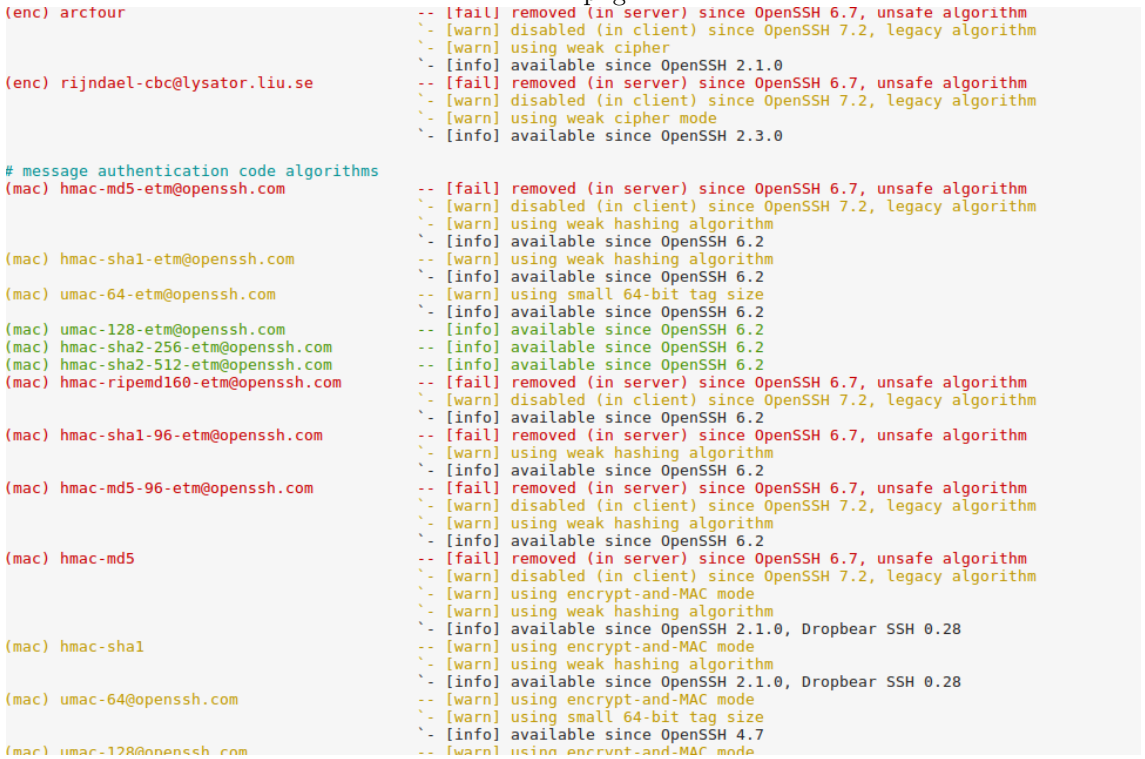

Figura 3: ssh-audit aplicado ao servidor da PT Comunicações (3)

- PT.png

```
# algorithm recommendations (for OpenSSH 6.6.1)
(rec) -diffie-hellman-group14-sha1          -- kex algorithm to remove
(rec) -diffie-hellman-group-exchange-sha1   -- kex algorithm to remove
(rec) -diffie-hellman-group1-sha1           -- kex algorithm to remove
(rec) -ecdh-sha2-nistp256                   -- kex algorithm to remove
(rec) -ecdh-sha2-nistp521                   -- kex algorithm to remove
(rec) -ecdh-sha2-nistp384                   -- kex algorithm to remove
(rec) -ecdsa-sha2-nistp256                  -- key algorithm to remove
(rec) -ssh-dss                              -- key algorithm to remove
(rec) +ssh-ed25519                          -- key algorithm to append
(rec) -arcfour                              -- enc algorithm to remove
(rec) -rijndael-cbc@lysator.liu.se          -- enc algorithm to remove
(rec) -blowfish-cbc                         -- enc algorithm to remove
(rec) -3des-cbc                             -- enc algorithm to remove
(rec) -aes256-cbc                           -- enc algorithm to remove
(rec) -arcfour256                           -- enc algorithm to remove
(rec) -cast128-cbc                          -- enc algorithm to remove
(rec) -aes192-cbc                           -- enc algorithm to remove
(rec) -arcfour128                           -- enc algorithm to remove
(rec) -aes128-cbc                           -- enc algorithm to remove
(rec) -hmac-sha2-512                        -- mac algorithm to remove
(rec) -hmac-md5-96                          -- mac algorithm to remove
(rec) -hmac-md5-etm@openssh.com             -- mac algorithm to remove
(rec) -hmac-sha1-96-etm@openssh.com         -- mac algorithm to remove
(rec) -hmac-ripemd160-etm@openssh.com       -- mac algorithm to remove
(rec) -hmac-md5-96-etm@openssh.com          -- mac algorithm to remove
(rec) -hmac-sha2-256                        -- mac algorithm to remove
(rec) -hmac-ripemd160                       -- mac algorithm to remove
(rec) -umac-128@openssh.com                 -- mac algorithm to remove
(rec) -hmac-sha1-96                         -- mac algorithm to remove
(rec) -umac-64@openssh.com                  -- mac algorithm to remove
(rec) -hmac-md5                             -- mac algorithm to remove
(rec) -hmac-ripemd160@openssh.com           -- mac algorithm to remove
(rec) -hmac-sha1                            -- mac algorithm to remove
(rec) -hmac-sha1-etm@openssh.com            -- mac algorithm to remove
(rec) -umac-64-etm@openssh.com              -- mac algorithm to remove
```

Figura 4: ssh-audit aplicado ao servidor da PT Comunicações (4)

```
user@CSI:~/Desktop/Tools/ssh-audit$ python ssh-audit.py 88.157.176.86
# general
(gen) banner: SSH-2.0-Cisco-1.25
(gen) software: Cisco IOS/PIX sshd 1.25
(gen) compatibility: OpenSSH 3.9-6.6, Dropbear SSH 0.53+
(gen) compression: disabled

# key exchange algorithms
(kex) diffie-hellman-group-exchange-sha1  -- [fail] removed (in server) since OpenSSH 6.7, unsafe algorithm
                                          `- [warn] using weak hashing algorithm
                                          `- [info] available since OpenSSH 2.3.0
(kex) diffie-hellman-group14-sha1         -- [warn] using weak hashing algorithm
                                          `- [info] available since OpenSSH 3.9, Dropbear SSH 0.53
(kex) diffie-hellman-group1-sha1          -- [fail] removed (in server) since OpenSSH 6.7, unsafe algorithm
                                          `- [fail] disabled (in client) since OpenSSH 7.0, logjam attack
                                          `- [warn] using small 1024-bit modulus
                                          `- [warn] using weak hashing algorithm
                                          `- [info] available since OpenSSH 2.3.0, Dropbear SSH 0.28

# host-key algorithms
(key) ssh-rsa                             -- [info] available since OpenSSH 2.5.0, Dropbear SSH 0.28

# encryption algorithms (ciphers)
(enc) aes128-cbc                          -- [fail] removed (in server) since OpenSSH 6.7, unsafe algorithm
                                          `- [warn] using weak cipher mode
                                          `- [info] available since OpenSSH 2.3.0, Dropbear SSH 0.28
(enc) 3des-cbc                            -- [fail] removed (in server) since OpenSSH 6.7, unsafe algorithm
                                          `- [warn] using weak cipher
                                          `- [warn] using weak cipher mode
                                          `- [warn] using small 64-bit block size
                                          `- [info] available since OpenSSH 1.2.2, Dropbear SSH 0.28
(enc) aes192-cbc                          -- [fail] removed (in server) since OpenSSH 6.7, unsafe algorithm
                                          `- [warn] using weak cipher mode
                                          `- [info] available since OpenSSH 2.3.0
(enc) aes256-cbc                          -- [fail] removed (in server) since OpenSSH 6.7, unsafe algorithm
                                          `- [warn] using weak cipher mode
                                          `- [info] available since OpenSSH 2.3.0, Dropbear SSH 0.47
```

Figura 5: ssh-audit aplicado ao servidor da NOS (1)

```
# message authentication code algorithms
(mac) hmac-sha1                             -- [warn] using encrypt-and-MAC mode
                                            `- [warn] using weak hashing algorithm
                                            `- [info] available since OpenSSH 2.1.0, Dropbear SSH 0.28
(mac) hmac-sha1-96                          -- [fail] removed (in server) since OpenSSH 6.7, unsafe algorithm
                                            `- [warn] disabled (in client) since OpenSSH 7.2, legacy algorithm
                                            `- [warn] using encrypt-and-MAC mode
                                            `- [warn] using weak hashing algorithm
                                            `- [info] available since OpenSSH 2.5.0, Dropbear SSH 0.47

# algorithm recommendations (for OpenSSH 3.9)
(rec) -diffie-hellman-group1-sha1           -- kex algorithm to remove
(rec) -diffie-hellman-group-exchange-sha1   -- kex algorithm to remove
(rec) -aes192-cbc                           -- enc algorithm to remove
(rec) -aes128-cbc                           -- enc algorithm to remove
(rec) -3des-cbc                             -- enc algorithm to remove
(rec) -aes256-cbc                           -- enc algorithm to remove
(rec) +aes256-ctr                           -- enc algorithm to append
(rec) +aes128-ctr                           -- enc algorithm to append
(rec) +aes192-ctr                           -- enc algorithm to append
(rec) -hmac-sha1-96                         -- mac algorithm to remove
```

Figura 6: ssh-audit aplicado ao servidor da NOS (2)

Referências

[1] The Transport Layer Security (TLS) Protocol Version 1.2

`https://tools.ietf.org/html/rfc5246`

[2] CVE-2011-3389

`https://access.redhat.com/security/cve/cve-2011-3389`

[3] Beat the BEAST with TLS 1.1/1.2 and More

`https://blogs.cisco.com/security/beat-the-beast-with-tls`

[4] Prohibiting RC4 Cipher Suites

`https://tools.ietf.org/html/rfc7465`

[5] CSS Injection Vulnerability

`http://ccsinjection.lepidum.co.jp/`