FCTUC FACULDADE DE CIÊNCIAS E TECNOLOGIA
UNIVERSIDADE DE COIMBRA

Trabalho Prático #1

Segurança em Tecnologias de Informação
Departamento de Engenharia Informática
Ano letivo 2019/2020 - 2º Semestre

Trabalho realizado por:

- Francisco Miguel Almeida Monteiro, 2016241480
- Ricardo José Monteiro Paiva, 2016253100

Introdução:

Para este trabalho, foi-nos pedido para implementar um cenário VPN com os conhecimentos adquiridos nas aulas práticas. O presente relatório descreve o cenário implementado e todo o procedimento executado na criação de todas as unidades que o constituem nomeadamente a Certificate Authority (CA), todos os certificados X.509, OCSP e a implementação de autenticação por dois fatores com o auxílio do Google Authenticator.
A versão do CentOS utilizada para este projeto foi a 8.

1. Cenário

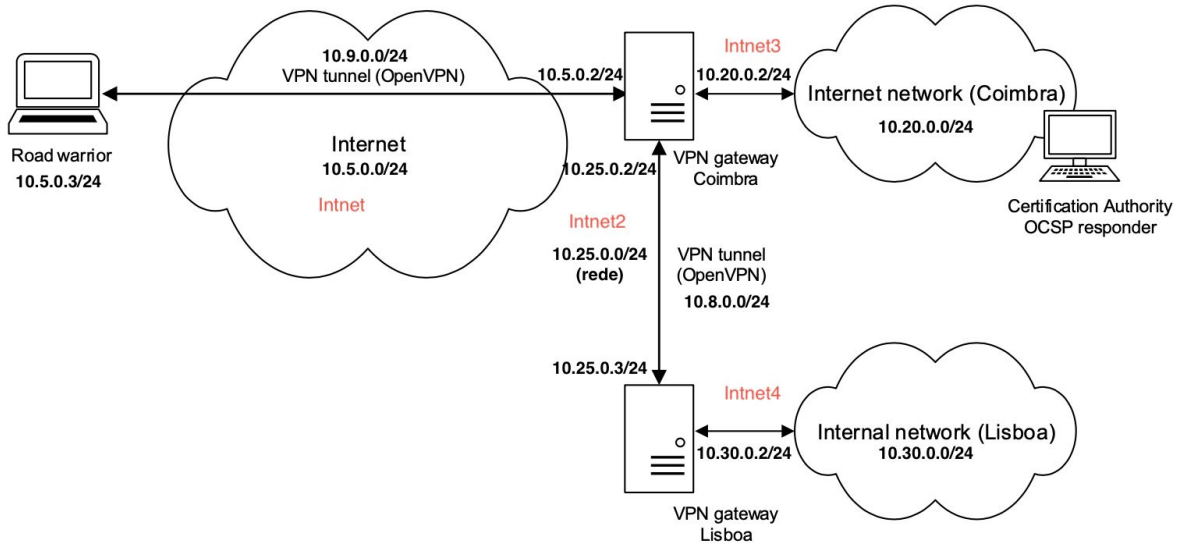

Figura 1. Cenário implementado

Como podemos verificar através da análise figura anterior, foram configurados dois túneis VPN, um entre o road warrior (cliente) e o gateway de Coimbra e o segundo entre o gateway de Coimbra e o gateway de Lisboa. Para conseguir estabelecer estas ligações, foram clonadas 3 máquinas virtuais. Em cada uma das máquinas, foram criadas e configuradas manualmente as redes internas necessárias para a comunicação. O esquema de configuração está presente em baixo:

Gateway de Coimbra:

- **3 redes internas + NAT**
  - IP: 10.5.0.2 Mask: 255.255.255.0
  - IP: 10.20.0.2 Mask:255.255.255.0
  - IP: 10.25.0.2 Mask: 255.255.255.0

Gateway de Lisboa:

- **2 redes internas + NAT**
  - IP: 10.25.0.3 Mask: 255.255.255.0
  - IP: 10.30.0.2 Mask: 255.255.255.0

Road Warrior:

- **1 rede interna + NAT**
  - IP: 10.5.0.3 Mask: 255.255.255.0

É de salientar que foi necessário desativar sempre a firewall em todas as máquinas virtuais para podermos executar os diferentes comandos. Para o fazer utilizámos os seguinte comandos:

```
# Verificar o estado da firewall
firewall-cmd --state

# Desligar a firewall
systemctl stop firewalld
```

2. Procedimento

Inicialmente começámos por criar a chave associada à autoridade de certificação, o CSR (Certificate Signing Request) e o certificado correspondente. Esta autoridade foi criada no gateway de Coimbra dentro da diretoria **/etc/pki/projeto** antes de ser feita a clonagem das VMs que viriam a ser utilizadas no trabalho. Os comandos utilizados foram os seguintes:

```
#1 - Chave
openssl genrsa -out CA.key 2048

#2 - CSR
openssl req -new -key CA.key -out CA.csr

#3 - Certificado
openssl x509 -req -days 3650 -in CA.csr -out CA.crt -signkey CA.key
```

Nota: No comando da criação da chave era aconselhável utilizar encriptação (3DES), porém para este caso optámos por não utilizar visto não ser o objetivo do trabalho.

De seguida, foram criadas as chaves e os respetivos certificados para cada um dos componentes, nomeadamente o **road warrior**, o **gateway de Coimbra** e o **gateway de Lisboa**. Os comandos utilizados foram semelhantes aos apresentados em cima pelo que só foram mudados os nomes dos ficheiros e o comando da criação do certificado que neste caso tem de estar associado à CA.

```
# Chave do Gateway de Coimbra
openssl genrsa -out coimbra.key 2048

# Chave do Gateway de Lisboa
openssl genrsa -out lisboa.key 2048

# Chave do Cliente (RoadWarrior)
openssl genrsa -out client.key 2048

# CSR de Coimbra
openssl req -new -key coimbra.key -out coimbra.csr

# CSR de Lisboa
openssl req -new -key lisboa.key -out lisboa.csr

# CSR do Cliente (RoadWarrior)
openssl req -new -key client.key -out client.csr

# Certificado do Gateway Coimbra associado à CA
openssl ca -in coimbra.csr -cert CA.crt -keyfile CA.key -out coimbra.crt

# Certificado do Gateway Lisboa associado à CA
openssl ca -in lisboa.csr -cert CA.crt -keyfile CA.key -out lisboa.crt

# Certificado do Cliente (RoadWarrior) associado à CA
openssl ca -in client.csr -cert CA.crt -keyfile CA.key -out client.crt
```

Posto isto, e como referido anteriormente, passámos então à clonagem da VM original de forma a obtermos um total de **3 máquinas** que nos permitissem simular o cenário pretendido.

Tendo as máquinas clonadas, tivemos de configurar as interfaces de cada com os endereços IP (IPv4) mencionados anteriormente. Podemos observar uma ilustração deste passo na figura seguinte

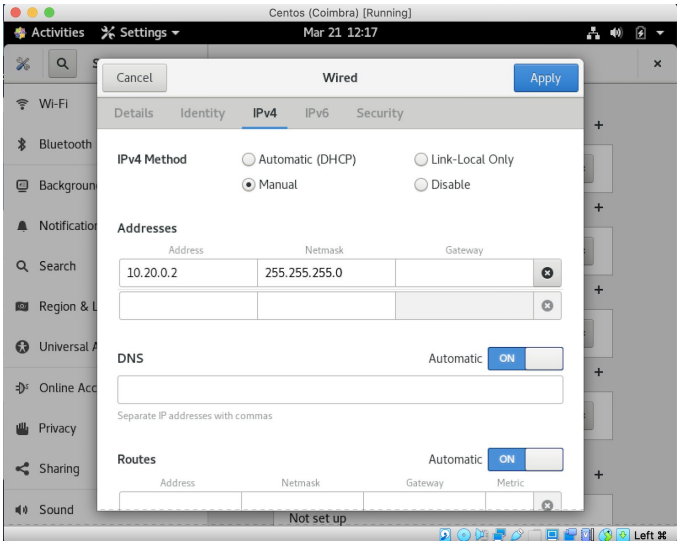

Figura 2. Configuração de uma rede interna

De seguida, com as máquinas com os respectivos endereços atribuídos, encontramos-nos em condições de proceder à configuração dos ficheiros OpenVPN de cada uma.

Na secção seguinte vamos falar de alguns dos comandos utilizados **(os mais importantes)** e a função de cada um deles. Estes ficheiros de configuração encontram-se também em anexo juntamente com este relatório.

2.1. Máquina Virtual de Coimbra

Esta VM tem dois ficheiros de configuração, ficheiros estes que seguem um "template" fornecido pelo próprio OpenVPN (server.conf).

2.1.1. coimbra.conf

Responsável por comunicar com o gateway de Lisboa. Este recebe os pedidos dos clientes da rede interna de Lisboa.

```
# Atribuição do IP da interface do servidor
local 10.25.0.2

# Porto
port 1195

# Protocolo de transporte
proto udp

# Usamos uma interface TUN visto estarmos a configurar um túnel VPN
dev tun
```

```
# Certificado da Autoridade de Certificação, certificado do servidor,
chave privada do servidor, respectivamente
ca /etc/pki/projeto/CA.crt
cert /etc/pki/projeto/coimbra.crt
key /etc/pki/projeto/coimbra.key

# Gama de IPs que vão ser atribuídos aos clientes VPN
server 10.8.0.0 255.255.255.0

# Redes que passam a ser alcançáveis
push "route 10.20.0.0 255.255.255.0"
push "route 10.5.0.0 255.255.255.0"
push "route 10.9.0.0 255.255.255.0"

# Para a rede interna conseguir ser acedida pelo VPN
client-config-dir /etc/openvpn/ccd
route 10.30.0.0 255.255.255.0
```

2.1.2 rw_server.conf

A configuração desta interface é semelhante à anterior, mudando apenas alguns comandos. Este recebe os pedidos dos road warriors e comunica com o OCSP para verificar a validade dos certificados.

```
local 10.5.0.2
port 1194
proto udp
dev tun
ca /etc/pki/projeto/CA.crt
cert /etc/pki/projeto/coimbra.crt
key /etc/pki/projeto/coimbra.key
server 10.9.0.0 255.255.255.0
push "route 10.20.0.0 255.255.255.0"
push "route 10.25.0.0 255.255.255.0"
push "route 10.8.0.0 255.255.255.0"
push "route 10.30.0.0 255.255.255.0"

# Para permitir a execução de scripts externos
script-security 2
```

```
# Bash script associado ao OCSP
tls-verify /etc/openvpn/ocsp.sh

# Plugin relativamente ao 2 factor authentication (Google Authenticator)
plugin /usr/lib64/openvpn/plugins/openvpn-plugin-auth-pam.so openvpn
```

2.2. Máquina Virtual de Lisboa

2.2.1 lisboa.conf

Estabelece uma ligação com a interface de Coimbra e serve como ponto de acesso à sua rede interna (de Lisboa).

```
# Para indicar que este se trata de um cliente
client

dev tun
proto udp

# IP e porto da interface do servidor a que o cliente se vai ligar
remote 10.25.0.2 1195

# Certificado da Autoridade de Certificação, certificado do gateway de
Lisboa, chave privada do gateway de Lisboa, respectivamente
ca /etc/pki/projeto/CA.crt
cert /etc/pki/projeto/lisboa.crt
key /etc/pki/projeto/lisboa.key
```

2.3. Máquina Virtual do RoadWarrior

2.3.1. rw.conf

É o cliente deste cenário, que comunica com a gateway de Coimbra através da interface do servidor do RoadWarrior. Utiliza uma autenticação por dois fatores para estabelecer esta ligação.

```
client
dev tun
proto udp

# IP e porto da interface do servidor (rw_server) a que o cliente (RW)
se vai ligar
remote 10.5.0.2 1194

# Certificado da Autoridade de Certificação, certificado do RoadWarrior
(cliente), chave privada RoadWarrior, respectivamente
```

```
ca /etc/pki/projeto/CA.crt
cert /etc/pki/projeto/client.crt
key /etc/pki/projeto/client.key

script-security 2


# Para pedir username e password quando se liga o cliente
auth-user-pass
```

2.4. OCSP

Para a configuração do OCSP, que se encontra no gateway de Coimbra, foi utilizado um ficheiro (bash script) retirado de um repositório do OpenVPN (ver referências). Este ficheiro foi adaptado de acordo com o nosso cenário e colocado na pasta **/etc/open-vpn**. Este serve para verificar a validade dos diferentes certificados associados à AC. As mudanças efetuadas são apresentadas de seguida:

```
# Endereço de IP e porto em que o servidor OCSP vai estar à escuta
ocsp_url="http://10.5.0.2:1196"

# Autoridade de certificação que emite os certificados
issuer="/etc/pki/projeto/CA.crt"

# Autoridade de certificação que vai verificar a validade dos
certificados
verify="/etc/pki/projeto/CA.crt"
```

O servidor OCSP é executado num novo terminal da seguinte forma:

```
# Comando para executar o servidor OCSP
openssl ocsp -index /etc/pki/CA/index.txt -port 1196 -CA
/etc/pki/projeto/CA.crt -rsigner /etc/pki/projeto/CA.crt -rkey
/etc/pki/projeto/CA.key -resp_text
```

2.5. Autenticação por dois fatores

De forma a tornar a autenticação do cliente mais segura, foi nos pedido para implementar uma autenticação por dois fatores através do **Google Authenticator.**

Para tal, foi criado um cliente, com **username** e **password**, que de seguida teve de introduzir uma chave na aplicação de autenticação da Google, de forma a obter um token de confirmação, que se auto revalida passado algum tempo.

Para efetuar o login é necessário inserir o username e posteriormente a password sucedida de um **token** de 6 algarismos gerado na aplicação.

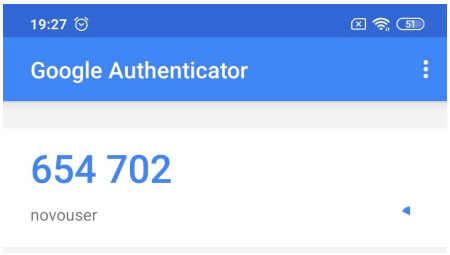

Figura 3. Google Authenticator

3. Testes efetuados

Para verificar se a nossa implementação estava de acordo com os requisitos, foram executados diferentes testes, dois dos quais utilizaram uma ferramenta WireShark. Esta ferramenta permite verificar o envio e receção de pacotes através dos túneis VPN criados.

3.1. Túneis

De forma a verificar se os túneis VPN configurados estavam a ser utilizados, colocámos o WireShark à escuta nos dois túneis presentes no nosso cenário.

- **RW ↔ Coimbra**

Neste primeiro teste, através do Road Warrior (cliente), fizemos ping à rede interna de Coimbra. Como se verifica na **Figura 4**, os pacotes para chegarem a esta rede interna, atravessaram o túnel VPN **(tun0)**, o que indica a sua correta configuração.

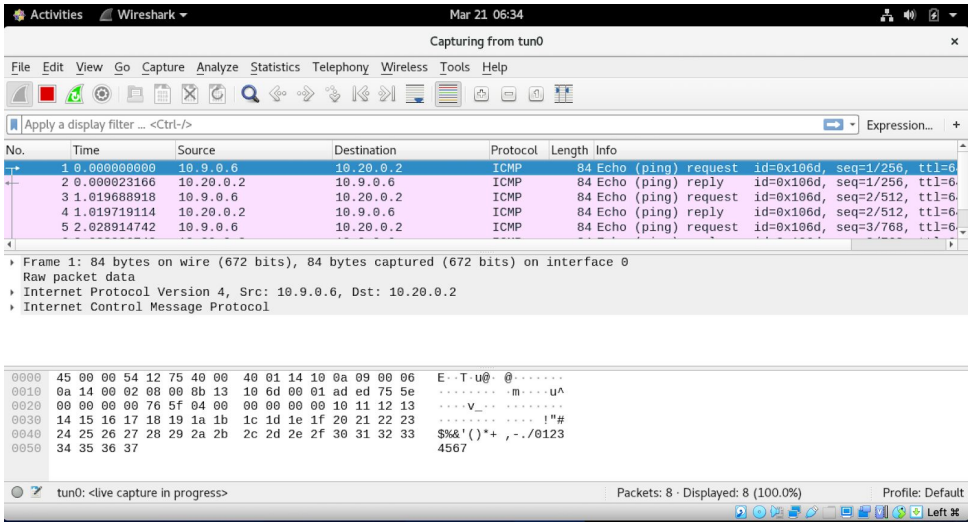

Figura 4. WireShark Gateway Coimbra (tun0)

- **RW ↔ Lisboa**

Neste teste, e de forma análoga ao anterior, fizemos ping desde o Road Warrior até à rede interna de Lisboa de forma a poder comprovar se os pacotes passam pelo túnel VPN.

De referir que neste caso de teste, os pacotes têm de atravessar dois túneis VPN, um dos quais referido no teste anterior.

Como podemos observar na **Figura 5** e **Figura 6**, verificamos que os pacotes passam, como seria de esperar, nos dois túneis **(tun0 e tun1)**.

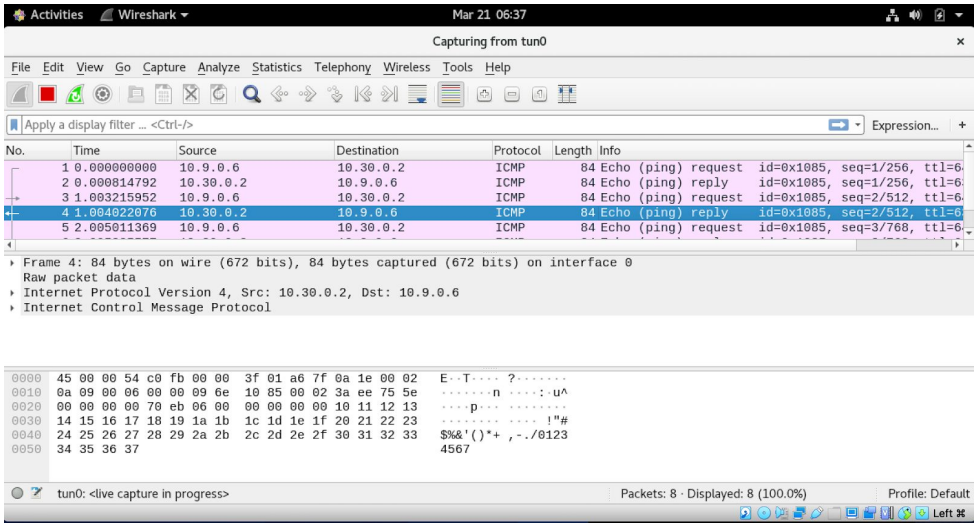

Figura 5. WireShark Gateway Coimbra (tun0)

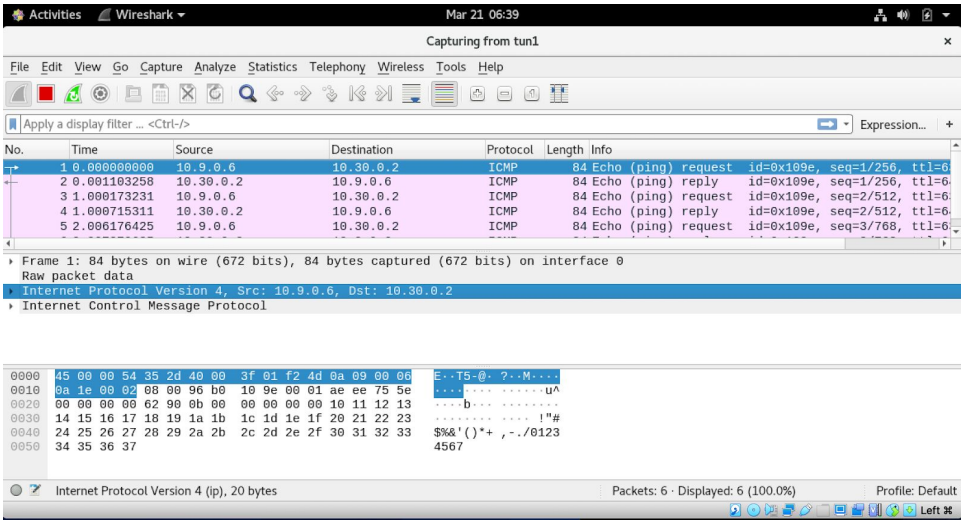

Figura 6. WireShark Gateway Lisboa (tun1)

3.2. Certificados

Para verificar se o OSCP estava a verificar a revogação dos diferentes certificados, foi criado um certificado de teste associado ao road warrior (Cliente). O certificado deste cliente foi então revogado na CA com auxílio do comando:

```
# Comando para revogar um certificado
openssl ca -keyfile /etc/pki/projeto/CA.key -cert
/etc/pki/projeto/CA.crt -revoke /etc/pki/projeto/newuser.cert
```

Ao estar revogado, tentámos iniciar uma nova conexão com o servidor, porém verificámos que este imprimiu uma mensagem de erro onde se pode verificar que detetou que o certificado está revogado. Podemos verificar essa mensagem na figura seguinte:

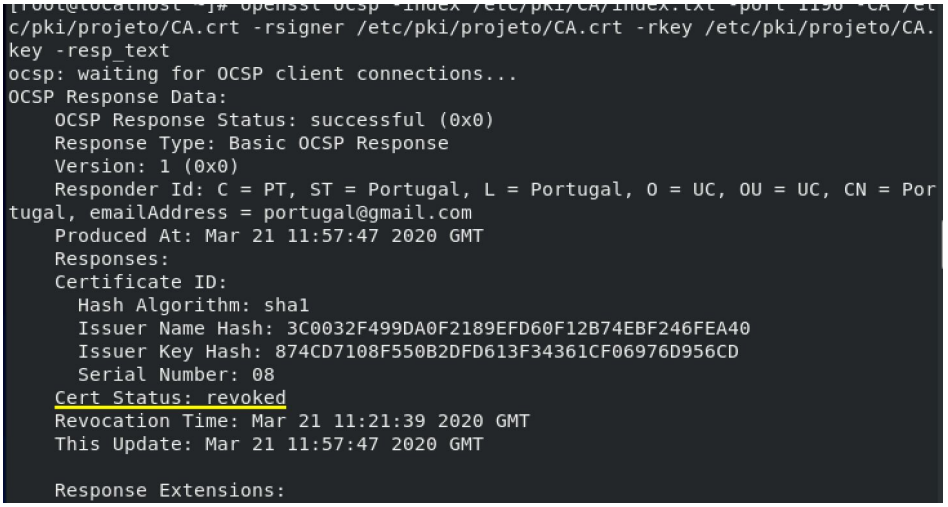

Figura 7. Output OCSP

Com o certificado revogado, o cliente não se consegue autenticar ao servidor visto que este não aceita a conexão.

3.3. Autenticação por dois fatores

Para podermos avaliar o funcionamento da autenticação por dois fatores tentámos fazer login através do road warrior ao servidor de coimbra utilizando uma palavra passe incorreta ou mesmo a palavra passe correta, mas sem o token. Como era de esperar, a autenticação não foi bem sucedida, como podemos observar na imagem seguinte.

Figura 8. Output Two Factor Authentication

4. Entrega com PGP

Para a entrega do projeto, foi necessário utilizar a nossa chave PGP para assinar o arquivo que contém todos os ficheiros de configuração e o presente relatório. Para isso usámos o seguinte comando onde "entrega" corresponde ao arquivo a ser entregue.

```
gpg --sign entrega
```

5. Referências

https://openvpn.net/community-resources/how-to/#openvpn-quickstart
https://backreference.org/2010/05/09/ocsp-verification-with-openssl/
https://medium.com/@bhashineen/create-your-own-ocsp-server-ffb212df8e63
https://linuxize.com/post/how-to-stop-and-disable-firewalld-on-centos-7/
https://medium.com/@alexeysamoshkin/how-ssl-certificate-revocation-is-broken-in-practice-af3b63b9cb3
https://github.com/OpenVPN/openvpn/blob/master/contrib/OCSP_check/OCSP_check.sh
https://linuxconfig.org/install-wireshark-on-linux-redhat-8
https://nethack.ch/2016/12/08/setup-an-openvpn-server-with-certificate-and-two-factor-authentication-on-centos-7/