

Trabalho Prático 1

Segurança em Tecnologias da Informação

Mestrado em Engenharia Informática

2022/2023

David Caetano - uc2018283431@student.uc.pt
Tomás Ventura - uc2018279147@student.uc.pt

Índice

1 Certificados	3
1.1 Criação da CA	3
1.2 Certificados VPN Gateway	3
1.3 Certificados VPN Client	3
1.4 Certificados Apache	3
2 Configuração de túneis VPN	4
2.1 VPN Gateway	4
2.2 VPN Client	5
2.3 Script verificação dos certificados (OCSP_check.sh)	6
2.4 Apache	7
2.4 Autenticação de 2 Fatores	8
3 Routing	8
3.1 IP das Máquinas Virtuais, suas interfaces, route Tables e comandos usados	8
3.2 Hosts Files em cada máquina	10
3.3 Testes das ligações entre máquinas	11
Apache -> VPN	11
VPN -> Client	11
VPN -> Apache	12
Cliente -> VPN	12
Client -> Apache (antes da criação do túnel o ping falha)	13
4 OCSP	13
4.1 Teste do OCSP	13
5 Objetivos	15
5.1 Criação do túnel entre Cliente e VPN	15
5.2 Aceder ao Apache	16
5.3 Confirmar estado dos Certificados	16
5.4 OTP	16

1 Certificados

1.1 Criação da CA

Uma Autoridade Certificadora (CA) é uma entidade responsável pela emissão, distribuição, renovação e revogação de certificados digitais. Sendo que, o propósito destes certificados é validar identidades na Internet.

Para a criação da nossa CA foram utilizados os seguintes comandos:

```
openssl genrsa -out ca.key -des3 1024
openssl req -new -key ca.key -out ca.csr
openssl x509 -req -days 365 -in ca.csr -out ca.crt -signkey ca.key
```

Inicialmente, geramos uma chave RSA denominada ca.key. Seguidamente, utilizamos essa mesma chave para gerar uma solicitação de assinatura do certificado, ca.csr, contendo todos os dados da entidade a ser certificada. Finalmente, o certificado é gerado através da assinatura por parte da chave da própria CA, tratando-se então de um certificado self-signed.

1.2 Certificados VPN Gateway

O processo para a criação do certificado VPN Gateway foi semelhante ao anterior, na qual o pedido de emissão de certificado foi assinado pela CA. Os seguintes comandos foram utilizados:

```
openssl genrsa -out vpn.key -des3 1024
openssl req -new -key vpn.key -out vpn.csr
openssl ca -in vpn.csr -cert ca.crt -keyfile ca.key -out vpn.crt
```

1.3 Certificados VPN Client

A criação do certificado VPN Client seguiu os mesmos princípios que o anterior. Os seguintes comandos foram utilizados:

```
openssl genrsa -out pessoa1.key -des3 1024
openssl req -new -key pessoa1.key -out pessoa1.csr
openssl ca -in pessoa1.csr -cert ca.crt -keyfile ca.key -out pessoa1.crt
```

1.4 Certificados Apache

Relativamente ao Apache, o processo foi exatamente igual ao da VPN Gateway e Client. Os seguintes comandos foram utilizados:

```
openssl genrsa -out apache.key -des3 1024
openssl req -new -key apache.key -out apache.csr
openssl ca -in apache.csr -cert ca.crt -keyfile ca.key -out apache.crt
```

Foi ainda necessário editar o ficheiro `ssl.conf`, na qual inserimos o *path* para o ficheiro `apache.crt` na secção `Server Certificate` e o *path* para o ficheiro `apache.key` na secção `Server Private Key`.

2 Configuração de túneis VPN

Para que possamos estabelecer uma ligação entre servidor (VPN Gateway) e cliente (VPN Client), tivemos que fazer várias alterações aos ficheiros `server.conf` e `client.conf` na pasta `openvpn`.

É de realçar que para facilitar estabelecer uma ligação entre as máquinas é necessário desativar a firewall em todas as máquinas através de:

```
systemctl stop firewalld
```

2.1 VPN Gateway

Estas foram as alterações realizadas:

```
- server.conf
    plugin openvpn-plugin-auth-pam.so "login login USERNAME password
    PASSWORD pin OTP"

    script-security 3
    tls-verify OCSF_check.sh

    local 10.5.0.1
    port 1194

    proto udp
    dev tun

    ca /home/dcaetano/STI/SSL-resultados/ca.crt
    cert /home/dcaetano/STI/SSL-resultados/vpn.crt
    key /home/dcaetano/STI/SSL-resultados/vpn.key # This file should be kept
    secret

    dh dh2048.pem
    server 10.8.0.0 255.255.255.0

    ifconfig-pool-persist ipp.txt
    push "route 10.6.0.0 255.255.255.0"

    keepalive 10 120
    tls-auth ta.key 0
    cipher AES-256-CBC
```

```
persist-key
persist-tun
status openvpn-status.log
verb 3
explicit-exit-notify 1
```

2.2 VPN Client

Estas foram as alterações realizadas:

- client.conf

```
client
dev tun
proto udp
```

```
remote vpn 1194
resolv-retry infinite
nobind
```

```
persist-key
persist-tun
```

```
ca /home/dcaetano/STI/SSL-resultados/ca.crt
cert /home/dcaetano/STI/SSL-resultados/pessoa1.crt
key /home/dcaetano/STI/SSL-resultados/pessoa1.key
```

```
auth-user-pass
static-challenge "Enter your OTP" 1
```

```
tls-auth ta.key 1
```

```
auth-user-pass
auth-nocache #n por a pass em cache
```

```
cipher AES-256-CBC
verb 3
explicit-exit-notify
```

2.3 Script verificação dos certificados (OCSP_check.sh)

```
#!/bin/bash
# if the depth is non-zero , continue processing
#echo "sstat 0x${tls_serial_0}"
[ "$1" -ne 0 ] && exit 0
issuer=/etc/pki/CA/ca.crt
CAfile=/etc/pki/CA/ca.crt
host=localhost
port=4444
if [ -n "${tls_serial_0}" ]
then
    status=$(openssl ocsp -issuer "${issuer}" -CAfile "${CAfile}" -host
"${host}" -port "${port}" -serial "${tls_serial_0}")
    #echo ${status} with openssl ocsp -issuer "${issuer}" -CAfile
"${CAfile}" -host "${host}" -port "${port}" -serial "${tls_serial_0}"
    if [ $? -eq 0 ]
    then
        # debug:
        #echo "OCSP status: $status"
        if echo "$status" | grep -Fq "${tls_serial_0}: good"
        then
            #echo "We exit gracefully"
            exit 0
        fi
    else
        # debug:
        echo "openssl ocsp command failed!"
    fi
fi
exit 1
```

2.4 Apache

Relativamente ao Apache, o seguinte ficheiro teve que ser alterado:

- ssl.conf

Listen 443 https

SSLPassPhraseDialog exec:/usr/libexec/httpd-ssl-pass-dialog

SSLSessionCache shmcb:/run/httpd/sslcache(512000)

SSLSessionCacheTimeout 300

SSLRandomSeed startup file:/dev/urandom 256

SSLRandomSeed connect builtin

SSLCryptoDevice builtin

<VirtualHost _default_:443>

ErrorLog logs/ssl_error_log

TransferLog logs/ssl_access_log

LogLevel warn

SSLEngine on

SSLProtocol all -SSLv2 -SSLv3

SSLCipherSuite HIGH:3DES:!aNULL:!MD5:!SEED:!IDEA

SSLCertificateFile /home/dcaetano/STI/SSL-resultados/apache.crt

SSLCertificateKeyFile /home/dcaetano/STI/SSL-resultados/apache.key

<Files ~ "\.(cgi|shtml|phtml|php3?)\$" >

SSLOptions +StdEnvVars

</Files>

<Directory "/var/www/cgi-bin">

SSLOptions +StdEnvVars

</Directory>

BrowserMatch "MSIE [2-5]" \

nokeepalive ssl-unclean-shutdown \

downgrade-1.0 force-response-1.0

CustomLog logs/ssl_request_log \

"%t %h %{SSL_PROTOCOL}x %{SSL_CIPHER}x \"%r\" %b"

</VirtualHost>

2.4 Autenticação de 2 Fatores

Autenticação por username e password estão a funcionar a 100%, no entanto 2FA nunca foi posta a funcionar, tentámos usar o plugin do google authenticator, que funcionou para gerar as OTP, porém quando este era listado nos processos de autenticação do linux, ele nunca aceitava os tokens estando estes corretos ou não.

As OTP eram associadas a cada usuário de linux ao entrar na sessão de cada um e executar o google-authenticator e seguir as instruções de criação serial key de 24 bits que depois era inserida na APP do Google Authenticator para gerar tokens baseados em tempo, o resultado final da geração da chave fica no root da pasta do respectivo user.

3 Routing

3.1 IP das Máquinas Virtuais, suas interfaces, route Tables e comandos usados

Cliente tem 1 interface para a rede 10.5.0.0/16 (Rede Pública)

Apache tem 1 interface para a rede 10.6.0.0/24 (Rede Privada)

VPN tem 2 interface uma para cada rede

VPN	(publica)	10.5.0.1	255.255.0.0	ip addr add 10.5.0.1/16 dev enp0s8
	(privada)	10.6.0.1	255.255.255.0	ip addr add 10.6.0.1/24 dev enp0s9
Client		10.5.0.2	255.255.0.0	ip addr add 10.5.0.2/16 dev enp0s8
Apache		10.6.0.2	255.255.255.0	ip addr add 10.6.0.2/24 dev enp0s8
				ip route add 10.8.0.0/24 via 10.6.0.2 dev enp0s8

Usados no VPN após o tun0 estar estabelecido com o cliente:

Mudar os IPs de saída e entrada que envolvam o tunel:

```
iptables -t nat -A POSTROUTING -s 10.8.0.0/24 -o enp0s9 -j MASQUERADE
iptables -t nat -A POSTROUTING -s 10.6.0.0/24 -o tun0 -j MASQUERADE
```

Encaminhar o que vem do tun0(cliente) para o enp0s9(apache) e vice-versa

```
sysctl -w net.ipv4.ip_forward=1
iptables -A FORWARD -i tun0 -o enp0s9 -j ACCEPT
iptables -A FORWARD -i enp0s9 -o tun0 -m state --state RELATED,ESTABLISHED -j ACCEPT
```


Client Final Route Table

```
etc : bash - Konsole
Ficheiro Editar Ver Favoritos Configuração Ajuda
[dcaetano@localhost etc]$ route
Kernel IP routing table
Destination Gateway Genmask Flags Metric Ref Use Iface
default gateway 0.0.0.0 UG 101 0 0 enp0s3
10.0.2.0 0.0.0.0 255.255.255.0 U 101 0 0 enp0s3
10.5.0.0 0.0.0.0 255.255.0.0 U 0 0 0 enp0s8
10.6.0.0 10.8.0.5 255.255.255.0 UG 0 0 0 tun0
10.8.0.1 10.8.0.5 255.255.255.255 UGH 0 0 0 tun0
10.8.0.5 0.0.0.0 255.255.255.255 UH 0 0 0 tun0
[dcaetano@localhost etc]$
```

VPN Final Route Table

```
sample-config-files : bash - Konsole
Ficheiro Editar Ver Favoritos Configuração Ajuda
[root@localhost sample-config-files]# route
Kernel IP routing table
Destination Gateway Genmask Flags Metric Ref Use Iface
default gateway 0.0.0.0 UG 101 0 0 enp0s3
10.0.2.0 0.0.0.0 255.255.255.0 U 101 0 0 enp0s3
10.5.0.0 0.0.0.0 255.255.0.0 U 0 0 0 enp0s8
10.6.0.0 0.0.0.0 255.255.255.0 U 0 0 0 enp0s9
10.8.0.0 10.8.0.2 255.255.255.0 UG 0 0 0 tun0
10.8.0.2 0.0.0.0 255.255.255.255 UH 0 0 0 tun0
[root@localhost sample-config-files]#
```

Apache Final Route Table

```
apache : bash - Konsole
Ficheiro Editar Ver Favoritos Configuração Ajuda
[root@localhost apache]# route
Kernel IP routing table
Destination Gateway Genmask Flags Metric Ref Use Iface
default gateway 0.0.0.0 UG 104 0 0 enp0s3
10.0.2.0 0.0.0.0 255.255.255.0 U 104 0 0 enp0s3
10.6.0.0 0.0.0.0 255.255.255.0 U 0 0 0 enp0s8
10.8.0.0 tomasventura 255.255.255.0 UG 0 0 0 enp0s8
[root@localhost apache]#
```

3.2 Hosts Files em cada máquina

VPN

```
sample-config-files : bash - Konsole

Ficheiro  Editar  Ver  Favoritos  Configuração  Ajuda

[root@localhost sample-config-files]# route
Kernel IP routing table
Destination Gateway      Genmask         Flags Metric Ref    Use Iface
default     gateway      0.0.0.0         UG    101    0      0 enp0s3
10.0.2.0     0.0.0.0      255.255.255.0   U     101    0      0 enp0s3
10.5.0.0     0.0.0.0      255.255.0.0     U     0      0      0 enp0s8
10.6.0.0     0.0.0.0      255.255.255.0   U     0      0      0 enp0s9
10.8.0.0     10.8.0.2     255.255.255.0   UG    0      0      0 tun0
10.8.0.2     0.0.0.0      255.255.255.255 UH    0      0      0 tun0
[root@localhost sample-config-files]#
```

Client

```
etc : bash - Konsole

Ficheiro  Editar  Ver  Favoritos  Configuração  Ajuda

[dcaetano@localhost etc]$ route
Kernel IP routing table
Destination Gateway      Genmask         Flags Metric Ref    Use Iface
default     gateway      0.0.0.0         UG    101    0      0 enp0s3
10.0.2.0     0.0.0.0      255.255.255.0   U     101    0      0 enp0s3
10.5.0.0     0.0.0.0      255.255.0.0     U     0      0      0 enp0s8
10.6.0.0     10.8.0.5     255.255.255.0   UG    0      0      0 tun0
10.8.0.1     10.8.0.5     255.255.255.255 UGH   0      0      0 tun0
10.8.0.5     0.0.0.0      255.255.255.255 UH    0      0      0 tun0
[dcaetano@localhost etc]$
```

Apache

```
sample-config-files : bash - Konsole

Ficheiro  Editar  Ver  Favoritos  Configuração  Ajuda

[root@localhost sample-config-files]# route
Kernel IP routing table
Destination Gateway      Genmask         Flags Metric Ref    Use Iface
default     gateway      0.0.0.0         UG    101    0      0 enp0s3
10.0.2.0     0.0.0.0      255.255.255.0   U     101    0      0 enp0s3
10.5.0.0     0.0.0.0      255.255.0.0     U     0      0      0 enp0s8
10.6.0.0     0.0.0.0      255.255.255.0   U     0      0      0 enp0s9
10.8.0.0     10.8.0.2     255.255.255.0   UG    0      0      0 tun0
10.8.0.2     0.0.0.0      255.255.255.255 UH    0      0      0 tun0
[root@localhost sample-config-files]#
```

3.3 Testes das ligações entre máquinas

Apache -> VPN

```
Ficheiro  Editar  Ver  Favoritos  Configuração  Ajuda
[root@localhost apache]# ping 10.6.0.1
PING 10.6.0.1 (10.6.0.1) 56(84) bytes of data.
64 bytes from 10.6.0.1: icmp_seq=1 ttl=64 time=0.862 ms
64 bytes from 10.6.0.1: icmp_seq=2 ttl=64 time=0.858 ms
64 bytes from 10.6.0.1: icmp_seq=3 ttl=64 time=1.12 ms
^C
--- 10.6.0.1 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2009ms
rtt min/avg/max/mdev = 0.858/0.948/1.124/0.124 ms
[root@localhost apache]#
```

VPN -> Client

```
[root@localhost sample-config-files]# ping 10.5.0.2
PING 10.5.0.2 (10.5.0.2) 56(84) bytes of data.
64 bytes from 10.5.0.2: icmp_seq=1 ttl=64 time=4.27 ms
64 bytes from 10.5.0.2: icmp_seq=2 ttl=64 time=1.15 ms
64 bytes from 10.5.0.2: icmp_seq=3 ttl=64 time=0.477 ms
^C
--- 10.5.0.2 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2003ms
rtt min/avg/max/mdev = 0.477/1.968/4.276/1.655 ms
[root@localhost sample-config-files]#
```



client.conf - KWrite
On Ecrã 1

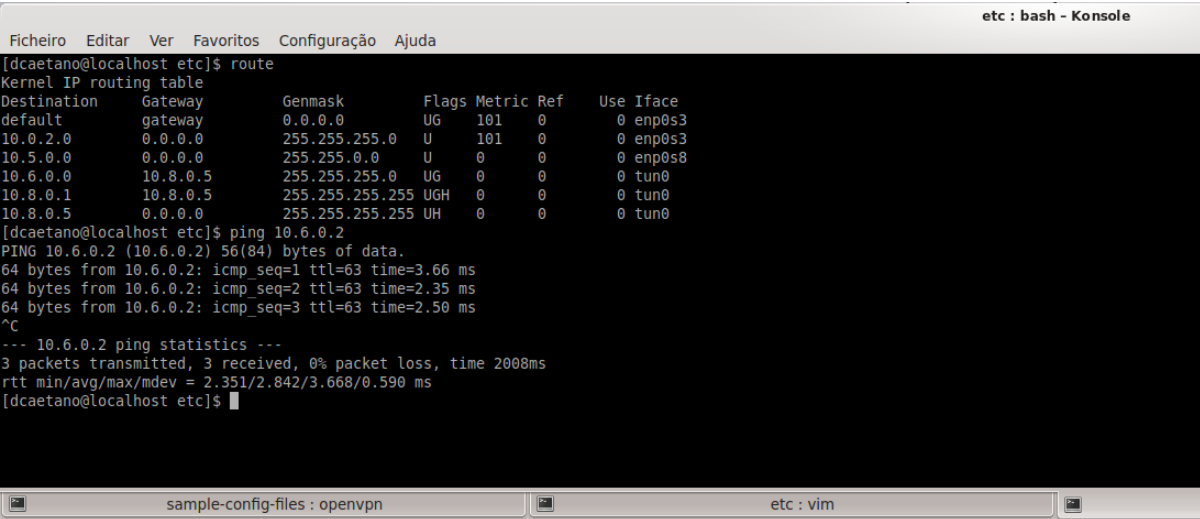
VPN -> Apache

```
[root@localhost sample-config-files]# ping 10.6.0.2
PING 10.6.0.2 (10.6.0.2) 56(84) bytes of data.
64 bytes from 10.6.0.2: icmp_seq=1 ttl=64 time=0.707 ms
64 bytes from 10.6.0.2: icmp_seq=2 ttl=64 time=0.552 ms
64 bytes from 10.6.0.2: icmp_seq=3 ttl=64 time=0.867 ms
^C
--- 10.6.0.2 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2001ms
rtt min/avg/max/mdev = 0.552/0.708/0.867/0.132 ms
[root@localhost sample-config-files]#
```

Cliente -> VPN

```
Ficheiro  Editar  Ver  Favoritos  Configuração  Ajuda
[root@localhost sample-config-files]# ping 10.5.0.1
PING 10.5.0.1 (10.5.0.1) 56(84) bytes of data.
64 bytes from 10.5.0.1: icmp_seq=1 ttl=64 time=0.949 ms
64 bytes from 10.5.0.1: icmp_seq=2 ttl=64 time=0.737 ms
64 bytes from 10.5.0.1: icmp_seq=3 ttl=64 time=0.884 ms
^C
--- 10.5.0.1 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2008ms
rtt min/avg/max/mdev = 0.737/0.856/0.949/0.094 ms
[root@localhost sample-config-files]#
```

Client -> Apache (antes da criação do túnel o ping falha)



The screenshot shows a terminal window titled "etc : bash - Konsole". The user is logged in as "dcaetano@localhost" in the "etc" directory. They run the command "route" to display the kernel IP routing table. The output shows a table with columns: Destination, Gateway, Genmask, Flags, Metric, Ref, Use, and Iface. The routes include a default gateway at 0.0.0.0 and specific routes for 10.0.2.0, 10.5.0.0, 10.6.0.0, 10.8.0.1, and 10.8.0.5. After displaying the routing table, the user runs "ping 10.6.0.2". The output shows three successful ping requests with response times around 3.66 ms, 2.35 ms, and 2.50 ms. Finally, the user runs "ping -c 1 10.6.0.2" to get a single statistics summary, which shows 3 packets transmitted, 3 received, 0% packet loss, and a time of 2008ms.

```
[dcaetano@localhost etc]$ route
Kernel IP routing table
Destination Gateway Genmask Flags Metric Ref Use Iface
default gateway 0.0.0.0 UG 101 0 0 enp0s3
10.0.2.0 0.0.0.0 255.255.255.0 U 101 0 0 enp0s3
10.5.0.0 0.0.0.0 255.255.0.0 U 0 0 0 enp0s8
10.6.0.0 10.8.0.5 255.255.255.0 UG 0 0 0 tun0
10.8.0.1 10.8.0.5 255.255.255.255 UGH 0 0 0 tun0
10.8.0.5 0.0.0.0 255.255.255.255 UH 0 0 0 tun0
[dcaetano@localhost etc]$ ping 10.6.0.2
PING 10.6.0.2 (10.6.0.2) 56(84) bytes of data.
64 bytes from 10.6.0.2: icmp_seq=1 ttl=63 time=3.66 ms
64 bytes from 10.6.0.2: icmp_seq=2 ttl=63 time=2.35 ms
64 bytes from 10.6.0.2: icmp_seq=3 ttl=63 time=2.50 ms
^C
--- 10.6.0.2 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2008ms
rtt min/avg/max/mdev = 2.351/2.842/3.668/0.590 ms
[dcaetano@localhost etc]$
```

4 OCSP

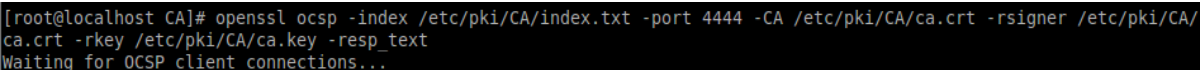
O OCSP é um protocolo utilizado para verificar o estado (validado ou revogado) de certificados utilizados por entidades.

4.1 Teste do OCSP

Para a configuração do OCSP é necessário, primeiramente, iniciar o serviço através do seguinte comando:

```
openssl ocsp -index /etc/pki/CA/index.txt -port 4444 -CA /etc/pki/CA/ca.crt -rsigner /etc/pki/CA/ca.crt -rkey /etc/pki/CA/ca.key -resp_text
```

Desta forma, o OCSP fica a aguardar que se estabeleçam ligações com clientes.



The screenshot shows a terminal window where the user runs "openssl ocsp" with the same flags as in the previous block. The output shows "Waiting for OCSP client connections..."

```
[root@localhost CA]# openssl ocsp -index /etc/pki/CA/index.txt -port 4444 -CA /etc/pki/CA/ca.crt -rsigner /etc/pki/CA/ca.crt -rkey /etc/pki/CA/ca.key -resp_text
Waiting for OCSP client connections...
```

Foi também criado um shell script, referenciado no server.conf através do tls-verify, de forma a verificar a validade do certificado.

```
#!/bin/bash
#if the depth is non-zero , continue processing
#echo "sstat 0x${tls_serial_0}"
[ "$1" -ne 0 ] && exit 0
issuer=/etc/pki/CA/ca.crt
CAfile=/etc/pki/CA/ca.crt
host=localhost
port=4444
if [ -n "${tls_serial_0}" ]
then
    status=$(openssl ocsp -issuer "${issuer}" -CAfile "${CAfile}" -host "${host}" -port "${port}" -serial "${tls_serial_0}")
    #echo ${status} with openssl ocsp -issuer "${issuer}" -CAfile "${CAfile}" -host "${host}" -port "${port}" -serial "${tls_serial_0}"
    if [ $? -eq 0 ]
    then
        # debug:
        #echo "OCSP status: $status"
        if echo "$status" | grep -Fq "${tls_serial_0}: good"
        then
            echo "We exit gracefully"
            exit 0
        fi
    else
        # debug:
        echo "openssl ocsp command failed!"
    fi
fi
exit 1
```

O script recebe o serial do certificado do cliente e faz um pedido de verificação do certificado ao OCSP. Aquando da ligação entre cliente e servidor, o certificado do cliente será verificado e, se não ocorrer nenhum erro e o estado ser válido, a seguinte mensagem aparecerá no terminal:

```
Waiting for OCSP client connections...
OCSP Response Data:
  OCSP Response Status: successful (0x0)
  Response Type: Basic OCSP Response
  Version: 1 (0x0)
  Responder Id: C = PT, ST = Coimbra, L = Coimbra, O = UC, OU = DEI, CN = CA
  Produced At: Mar 10 15:22:32 2023 GMT
  Responses:
  Certificate ID:
    Hash Algorithm: sha1
    Issuer Name Hash: ACD9FDAF22597E6170EC4A0F2B1BB96CB4102204
    Issuer Key Hash: BD0AA5273EC03FC274EB89B3390134D2D4E72B5B
    Serial Number: 04
  Cert Status: good
  This Update: Mar 10 15:22:32 2023 GMT
```

É possível fazer a revogação de um certificado através do seguinte comando:

```
openssl ca -keyfile ca.key -cert ca.crt -revoke exemplo.crt
```

5 Objetivos

5.1 Criação do túnel entre Cliente e VPN

Na VM da VPN iniciamos o servidor com:

openvpn server.conf

```
sample-config-files : openvpn - Konsole
[root@localhost sample-config-files]# openvpn server.conf
Mar 12 14:29:41 2023 WARNING: file 'ta.key' is group or others accessible
Mar 12 14:29:41 2023 OpenVPN 2.4.12 x86_64-redhat-linux-gnu [Fedora EPEL patched] [SSL (OpenSSL)] [LZO] [LZ4] [EPOLL] [PKCS11] [MH/TKTFINFO] [AEAD] built on Mar 17 2022
Mar 12 14:29:41 2023 Library versions: OpenSSL 1.0.2k-fips 26 Jan 2017, LZO 2.06
Mar 12 14:29:41 2023 PLUGIN_INIT: POST openvpn-plugin-auth-pam.so [login] [login] [USERNAME] [password] [PASSWORD] [pin] [OTP]: intercepted=PLUGIN_AUTH_USER_PASS_VERIFY
Mar 12 14:29:41 2023 Diffie-Hellman initialized with 2048 bit key
Mar 12 14:29:41 2023 Outgoing Control Channel Authentication: Using 160 bit message hash 'SHA1' for HMAC authentication
Mar 12 14:29:41 2023 Incoming Control Channel Authentication: Using 160 bit message hash 'SHA1' for HMAC authentication
Mar 12 14:29:41 2023 ROUTE GATEWAY 10.0.2.2/255.255.255.0 IFACE=enp0s3 HWADDR=08:00:27:e9:09:b8
Mar 12 14:29:41 2023 TUN/TAP device tun0 opened
Mar 12 14:29:41 2023 TUN/TAP TX queue length set to 100
Mar 12 14:29:41 2023 /sbin/ip link set dev tun0 up mtu 1500
Mar 12 14:29:41 2023 /sbin/ip addr add dev tun0 local 10.8.0.1 peer 10.8.0.2
Mar 12 14:29:41 2023 /sbin/ip route add 10.8.0.0/24 via 10.8.0.2
Mar 12 14:29:41 2023 Could not determine IPv4/IPv6 protocol. Using AF_INET
Mar 12 14:29:41 2023 Socket Buffers: R=[212992->212992] S=[212992->212992]
Mar 12 14:29:41 2023 UDPV4 Link local (bound): [AF_INET]10.5.0.1:1194
Mar 12 14:29:41 2023 UDPV4 Link remote: [AF_UNSPEC]
Mar 12 14:29:41 2023 MULTI: multi init called, rctx6 v=256
Mar 12 14:29:41 2023 IFCONFIG POOL: base=10.8.0.4 size=62, ipv6=0
Mar 12 14:29:41 2023 ifconfig_pool_read(), in='david,10.8.0.4', TOOO: IPv6
Mar 12 14:29:41 2023 succeeded -> ifconfig_pool_set()
Mar 12 14:29:41 2023 IFCONFIG POOL LIST
Mar 12 14:29:41 2023 david,10.8.0.4
Mar 12 14:29:41 2023 Initialization Sequence Completed
Mar 12 14:29:46 2023 10.5.0.2:58051 TLS: Initial packet from [AF_INET]10.5.0.2:58051, sid=9231ceff b6e5f150
Mar 12 14:29:46 2023 10.5.0.2:58051 VERIFY OK: depth=1, C=PT, ST=Coimbra, L=Coimbra, O=CNC, OU=DEI, CN=CA
Mar 12 14:29:46 2023 10.5.0.2:58051 VERIFY OK: depth=0, C=PT, ST=Coimbra, O=CNC, OU=DEI, CN=david
Mar 12 14:29:46 2023 10.5.0.2:58051 peer info: IV_VER=2.4.12
Mar 12 14:29:46 2023 10.5.0.2:58051 peer info: IV_PLAT=linux
Mar 12 14:29:46 2023 10.5.0.2:58051 peer info: IV_PROTO=2
Mar 12 14:29:46 2023 10.5.0.2:58051 peer info: IV_NCP=2
Mar 12 14:29:46 2023 10.5.0.2:58051 peer info: IV_CIPHER=AES-256-GCM:AES-128-GCM:AES-256-CBC
Mar 12 14:29:46 2023 10.5.0.2:58051 peer info: IV_LZ4=1
Mar 12 14:29:46 2023 10.5.0.2:58051 peer info: IV_LZ4v2=1
Mar 12 14:29:46 2023 10.5.0.2:58051 peer info: IV_LZO=1
Mar 12 14:29:46 2023 10.5.0.2:58051 peer info: IV_COMP_STUB=1
Mar 12 14:29:46 2023 10.5.0.2:58051 peer info: IV_COMP_STUBv2=1
Mar 12 14:29:46 2023 10.5.0.2:58051 peer info: IV_TCPNL=1
Mar 12 14:29:46 2023 10.5.0.2:58051 PLUGIN_CALL: POST openvpn-plugin-auth-pam.so/PLUGIN_AUTH_USER_PASS_VERIFY status=0
Mar 12 14:29:46 2023 10.5.0.2:58051 TLS: Username/Password authentication succeeded for username 'pessoal'
Mar 12 14:29:46 2023 10.5.0.2:58051 Control Channel: TLSv1.2, cipher TLSv1/SSLv3 ECDHE-RSA-AES256-GCM-SHA384, 1024 bit RSA
Mar 12 14:29:46 2023 10.5.0.2:58051 [david] Peer Connection Initiated with [AF_INET]10.5.0.2:58051
Mar 12 14:29:46 2023 david/10.5.0.2:58051 MULTI sva: pool returned IPv4=10.8.0.6, IPv6=(Not enabled)
Mar 12 14:29:46 2023 david/10.5.0.2:58051 MULTI: Learn: 10.8.0.6 -> david/10.5.0.2:58051
Mar 12 14:29:46 2023 david/10.5.0.2:58051 MULTI: primary virtual IP for david/10.5.0.2:58051: 10.8.0.6
Mar 12 14:29:47 2023 david/10.5.0.2:58051 PUSH: Received control message: 'PUSH REQUEST'
Mar 12 14:29:47 2023 david/10.5.0.2:58051 SENT CONTROL [david]: 'PUSH_REPLY,route 10.6.0.0 10.6.0.0 255.255.255.0,route 10.8.0.1,topology net30,ping 10,ping-restart 120,ifconfig 10.8.0.6 10.8.0.5,peer-id 0,cipher AES-256-GCM' (status=1)
sample-config-files : openvpn dcaetano : wireshark sample-config-files : bash
```

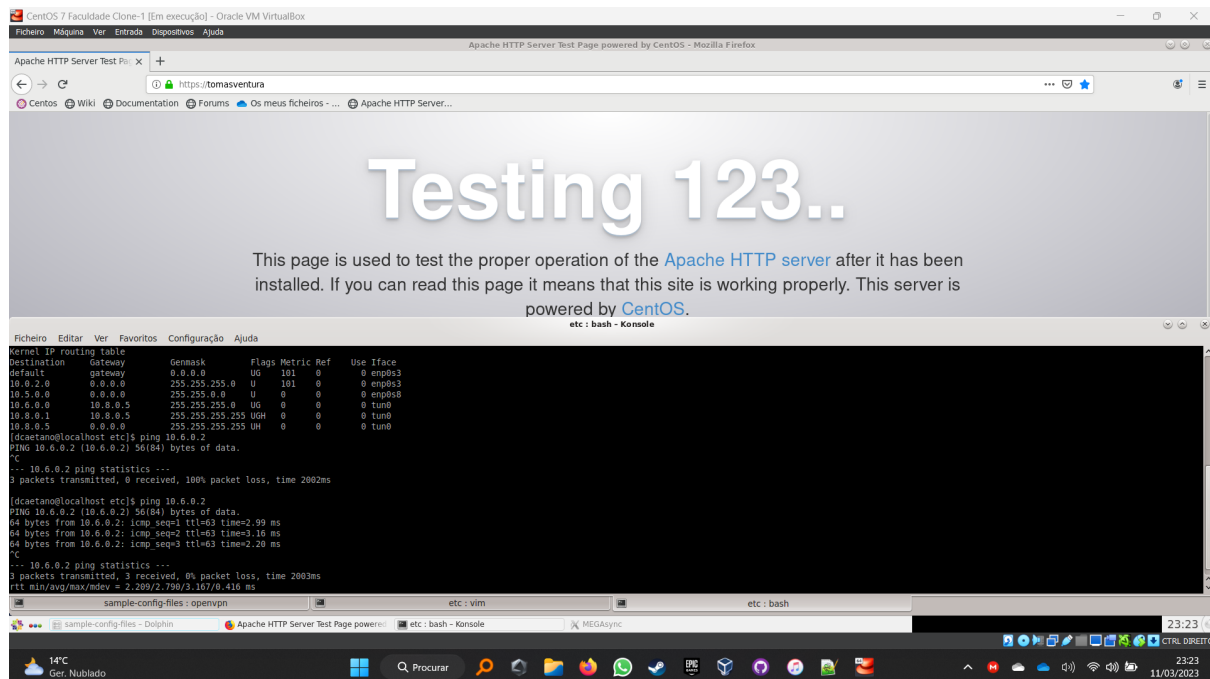
Na VM do cliente iniciamos o cliente com:

openvpn client.conf e inserimos o **username**, **password** e a **OTP**

```
sample-config-files : openvpn - Konsole
[Ficheiro Editar Ver Favoritos Configuração Ajuda]
[root@localhost sample-config-files]# openvpn client.conf
Sun Mar 12 14:26:44 2023 WARNING: file '/home/dcaetano/STI/SSL-resultados/pessoal.key' is group or others accessible
Sun Mar 12 14:26:44 2023 OpenVPN 2.4.12 x86_64-redhat-linux-gnu [Fedora EPEL patched] [SSL (OpenSSL)] [LZO] [LZ4] [EPOLL] [PKCS11] [MH/TKTFINFO] [AEAD] built on Mar 17 2022
Sun Mar 12 14:26:44 2023 Library versions: OpenSSL 1.0.2k-fips 26 Jan 2017, LZO 2.06
Enter Auth Username: pessoal
Enter Auth Password: ****
CHALLENGE: Enter your OTP
Sun Mar 12 14:26:53 2023 WARNING: No server certificate verification method has been enabled. See http://openvpn.net/howto.html#mitm for more info.
Sun Mar 12 14:26:53 2023 Outgoing Control Channel Authentication: Using 160 bit message hash 'SHA1' for HMAC authentication
Sun Mar 12 14:26:53 2023 Incoming Control Channel Authentication: Using 160 bit message hash 'SHA1' for HMAC authentication
Sun Mar 12 14:26:53 2023 TCP/UDP: Preserving recently used remote address: [AF_INET]10.5.0.1:1194
Sun Mar 12 14:26:53 2023 Socket Buffers: R=[212992->212992] S=[212992->212992]
Sun Mar 12 14:26:53 2023 UDP Link local (not bound)
Sun Mar 12 14:26:53 2023 UDP Link remote: [AF_INET]10.5.0.1:1194
Sun Mar 12 14:26:53 2023 TLS: Initial packet from [AF_INET]10.5.0.1:1194, sid=64705d37 2ed30328
Sun Mar 12 14:26:53 2023 VERIFY OK: depth=1, C=PT, ST=Coimbra, L=Coimbra, O=CNC, OU=DEI, CN=CA
Sun Mar 12 14:26:53 2023 VERIFY OK: depth=0, C=PT, ST=Coimbra, O=CNC, OU=DEI, CN=vpn
Sun Mar 12 14:26:53 2023 Control Channel: TLSv1.2, cipher TLSv1/SSLv3 ECDHE-RSA-AES256-GCM-SHA384, 1024 bit RSA
Sun Mar 12 14:26:53 2023 [vpn] Peer Connection Initiated with [AF_INET]10.5.0.1:1194
Sun Mar 12 14:26:54 2023 SENT CONTROL [vpn]: 'PUSH REQUEST' (status=1)
Sun Mar 12 14:26:54 2023 PUSH: Received control message: 'PUSH_REPLY,route 10.6.0.0 255.255.255.0,route 10.8.0.1,topology net30,ping 10,ping-restart 120,ifconfig 10.8.0.6 10.8.0.5,peer-id 0,cipher AES-256-GCM'
Sun Mar 12 14:26:54 2023 Options error: Unrecognized option or missing or extra parameter(s) in [PUSH-OPTIONS]:2: 10.6.0.0 (2.4.12)
Sun Mar 12 14:26:54 2023 OPTIONS IMPORT: timers and/or timeouts modified
Sun Mar 12 14:26:54 2023 OPTIONS IMPORT: --ifconfig/up options modified
Sun Mar 12 14:26:54 2023 OPTIONS IMPORT: route options modified
Sun Mar 12 14:26:54 2023 OPTIONS IMPORT: peer-id set
Sun Mar 12 14:26:54 2023 OPTIONS IMPORT: adjusting link mtu to 1024
Sun Mar 12 14:26:54 2023 OPTIONS IMPORT: data channel crypto options modified
Sun Mar 12 14:26:54 2023 Data Channel: using negotiated cipher 'AES-256-GCM'
Sun Mar 12 14:26:54 2023 Outgoing Data Channel: Cipher 'AES-256-GCM' initialized with 256 bit key
Sun Mar 12 14:26:54 2023 Incoming Data Channel: Cipher 'AES-256-GCM' initialized with 256 bit key
Sun Mar 12 14:26:54 2023 ROUTE GATEWAY 10.0.2.2/255.255.255.0 IFACE=enp0s3 HWADDR=08:00:27:b8:27:ae
Sun Mar 12 14:26:54 2023 TUN/TAP device tun0 opened
Sun Mar 12 14:26:54 2023 TUN/TAP TX queue length set to 100
Sun Mar 12 14:26:54 2023 /sbin/ip link set dev tun0 up mtu 1500
Sun Mar 12 14:26:54 2023 /sbin/ip addr add dev tun0 local 10.8.0.6 peer 10.8.0.5
Sun Mar 12 14:26:54 2023 /sbin/ip route add 10.6.0.0/24 via 10.8.0.5
Sun Mar 12 14:26:54 2023 /sbin/ip route add 10.8.0.1/32 via 10.8.0.5
Sun Mar 12 14:26:54 2023 Initialization Sequence Completed
sample-config-files : openvpn etc : vim etc : bash
```

5.2 Aceder ao Apache

Aceder ao site do apache com https através do cliente (após a criação do tun0 como VPN)



5.3 Confirmar estado dos Certificados

Quando o script de verificação dos certificados está descomentado no servidor e o OCPS está a correr na VM, o programa fica parado nunca finalizado o processo de verificação, tendo por isso sido comentado nos testes finais das ligações, no entanto como visto nas imagens de testes do OSCP, a validação e revogação dos certificados é feita de forma correta.

5.4 OTP

Como mencionado anteriormente, a OTP é pedida a todos os usuários assim como as sua senha, mesmo que elas não tenham nenhuma associada às suas contas, nesses casos são ignoradas.

Mas não foi possível por a validação das OTP a funcionarem corretamente, como tal não importa o input dado durante o login do cliente. Tentámos usar o system-auth do linux para chamar o módulo do google authenticator, com a tag "required", mas sortia resultados incorretos, por isso optámos por focar no resto do projeto.