

Practical Assignment #2

João Neto – 2023234004

Vasco Alves – 2022228207

24 de abril de 2026

Conteúdo

1	Introdução	2
2	Criação de certificados	2
3	Configuração da <i>Gateway</i> VPN	3
4	Configurar TOTP	3
4.1	Aceder ao código	3
5	Revocation e OCSP	3
5.1	Testar OSCP via revoke	3
6	Conclusion	4

1 Introdução

Este projecto tem como âmbito implementar, uma rede virtual privada (VPN) num cenário de road-warrior, configurar two-factor authentication com os serviços OpenVPN e Apache, e gerir certificados X.509 utilizando OCSP.

O nosso cenário vai envolver três máquinas, o cliente (road warrior), a gateway que utiliza OpenVPN e um servidor interno com OpenSSL e Apache. O OpenVPN utiliza two-factor authentication, recebendo o utilizador, e uma password que é uma junção de uma fixa, e de uma gerada pelo plugin google-authenticator. O servidor de Apache implementa a mesma autenticação. (!!!!!)

Os certificados utilizados foram certificados por uma autoridade central que não está no nosso cenário. A gestão da lista de revogação está a ser gerido pelo serviço OpenSSL que está na mesma máquina que o Apache. Num cenário real seria melhor dividir estes serviços por outras máquinas, mas os computadores que temos acesso estão limitados na quantidade de máquinas virtuais que consegue simular simultaneamente.

Temos então três máquinas virtuais:

Nome	Script	Rede
Road Warrior	VM_ROAD_WARRIOR.sh	Rede Externa 193.168.0.0/24
VPN Gateway	VM_OPENVPN_GATEWAY.sh	Router
OpenSSL / Apache	VM_OPENSSL_APACHE.sh	Rede Interna 10.60.0.0/24

2 Criação de certificados

Criar chaves com 2048 bits.

```
1 cert_ca="/C=PT/ST=Coimbra/L=Coimbra/O=UC/CN=CoimbraVPN"
2 cert_vpn="/C=PT/ST=Coimbra/L=Coimbra/O=UC/CN=gateway"
3 cert_user="/C=PT/ST=Coimbra/L=Coimbra/O=UC/CN=warrior"
4 cert_apache="/C=PT/ST=Coimbra/L=Coimbra/O=UC/CN=apache.coimbra"
5
6 openssl genrsa -out "ca.key" 2048
7 openssl req -x509 -nodes -days 365 -key "ca.key" -out "ca.crt"
  -subj "$cert_ca"
8 openssl genrsa -out "vpn.key" 2048
9 openssl req -new -key "vpn.key" -out "vpn.csr" -subj "$cert_vpn"
  "
10 openssl ca -batch -in "vpn.csr" -cert "ca.crt" -keyfile "ca.key"
  -out "vpn.crt" -config cheese.cfg
11 openssl dhparam -out "dh2048.pem" 2048
12 openvpn --genkey secret "ta.key"
13 openssl genrsa -out user.key
14 openssl req -new -key user.key -out user.csr -subj "$cert_user"
```



```

15 openssl ca -batch -in "user.csr" -cert "ca.crt" -keyfile "ca.
    key" -out "user.crt" -config cheese.cfg
16 openssl genrsa -out apache.key
17 openssl req -new -key apache.key -out apache.csr -subj "
    $cert_apache" -addext "subjectAltName = IP:10.60.0.1,DNS:
    apache"
18 openssl ca -batch -in "apache.csr" -cert "ca.crt" -keyfile "ca.
    key" -out "apache.crt" -config cheese.cfg

```

Criar chave secreta.

```
1 openssl --genkey secret ta.key
```

3 Configuração da *Gateway* VPN

4 Configurar TOTP

Foi criado o ficheiro `totp` com a configuração de autenticação a ser utilizada pelo plugin de PAM para o `openvpn`.

```
1 plugin /usr/lib64/openvpn/plugins/openvpn-plugin-auth-pam.so
    totp

```

4.1 Aceder ao código

Primeiro, na gateway, entramos como o utilizador desejado e obtemos a chave do gerador de palavras passas temporárias. Ao inserir a chave no `google authenticator` podemos obter a nossa primeira chave de 6 dígitos.

```

1 su john
2 google-authenticator

```

5 Revocation e OCSP

5.1 Testar OSCP via revoke

1. Conectar ao VPN e ver que funciona
2. Na máquina host, não nas vms, na repo mesmo.
3. revogar o certificado via `openssl -revoke user.crt -config cheese.cfg -keyfile ca.key -cert ca.crt`

4. Fechae OSCP e correr VM_OPENSSL novamente (copiar index.txt e serial?)
5. Tentar outra vez e ver que de facto falha

6 Conclusion

Conclusão!!!