++
++
2.1.1. Creación de las claves para la CA

1. Incluye el contenido del certificado de la CA en formato legible en la memoria.

poniendo este comando en la carpeta de pki, podemos ver el certificado de CA: openssl x509 -in ca.crt -text

```
Certificate:
    Data:
        Version: 3 (0x2)
        Serial Number: 18322321055141593085 (0xfe45f5ec805ffffd)
    Signature Algorithm: sha256WithRSAEncryption
        Issuer: CN=Easy-RSA CA
        Validity
            Not Before: Mar 20 11:04:16 2018 GMT
            Not After : Mar 17 11:04:16 2028 GMT
        Subject: CN=Easy-RSA CA
        Subject Public Key Info:
            Public Key Algorithm: rsaEncryption
                Public-Key: (2048 bit)
                Modulus:
                    00:ce:fa:09:f9:9a:b6:59:d5:11:e2:88:55:7d:8b:
                    1e:f0:c1:a4:7f:e5:cf:72:4d:cb:0b:ff:7b:0c:ab:
                    2e:2b:c6:8b:37:29:aa:34:bf:e3:c8:e2:4a:10:1a:
                    0d:4d:74:c5:c2:84:b4:50:7e:2a:a3:1f:14:d9:b7:
                    1f:3e:59:c5:5f:e6:b0:49:40:f1:c9:84:2e:84:c9:
                    c8:75:a4:31:4c:0c:e9:c8:50:8b:ce:06:0f:a6:a6:
                    90:86:a4:66:c1:6b:01:3c:bd:c5:7f:12:3d:c6:f1:
                    83:05:bf:3f:57:d1:2b:dc:63:c4:41:11:d9:36:16:
                    fa:9a:ec:dc:c7:b8:7c:f6:01:15:ce:18:2d:71:f5:
                    47:34:a2:92:fb:09:2e:ce:2a:a6:87:f1:88:1d:3d:
                    b8:1f:df:8e:07:80:9b:b9:f8:18:ab:de:75:39:3a:
                    96:76:45:a0:a9:dc:c4:a8:6f:69:6e:3a:e6:1a:73:
                    74:f2:89:ab:27:7e:73:af:54:67:cc:92:2c:c7:11:
                    14:1a:5d:0d:f3:b3:25:d1:36:e8:df:e5:56:a0:38:
                    19:d1:e3:67:52:bc:bb:ab:32:73:be:0a:ec:ad:ac:
                    f2:2b:3d:6b:cb:8d:2a:a0:c1:55:5f:42:57:cd:15:
                    84:2c:c9:87:04:e3:87:da:84:ea:33:82:9b:b3:a3:
                    1e:6d
                Exponent: 65537 (0x10001)
        X509v3 extensions:
            X509v3 Subject Key Identifier:

2E:CE:EB:8E:89:BC:8A:A6:85:38:17:62:4D:C9:68:F4:AF:8F:1D:A3
            X509v3 Authority Key Identifier:

keyid:2E:CE:EB:8E:89:BC:8A:A6:85:38:17:62:4D:C9:68:F4:AF:8F:1D:A3
                DirName:/CN=Easy-RSA CA
                serial:FE:45:F5:EC:80:5F:FF:FD

            X509v3 Basic Constraints:
```

```
                CA:TRUE
            X509v3 Key Usage:
                Certificate Sign, CRL Sign
    Signature Algorithm: sha256WithRSAEncryption
         01:6a:86:fc:e2:80:ff:7c:40:d9:ea:49:06:1c:c6:6c:51:41:
         9e:ef:c0:5f:a0:56:9a:75:7e:ca:e3:a9:bb:95:be:b9:48:4a:
         37:8b:32:7a:52:a3:b5:92:8b:94:01:fa:e3:e3:b6:19:b6:f2:
         ac:86:42:8e:9f:b4:62:fd:ca:89:be:d9:fa:99:a1:a0:68:6d:
         d4:08:ef:39:99:51:ab:de:e5:f1:32:c3:49:51:de:62:69:3e:
         25:46:b2:a8:91:de:8f:a9:fa:aa:30:af:45:8b:22:7f:8c:27:
         e2:7d:3a:f6:ee:c4:fd:1f:5f:3b:88:6b:fb:e4:f1:d2:ad:7b:
         16:4a:57:94:45:f4:c2:b8:4f:78:7e:c9:6c:12:9c:07:23:f6:
         8f:e8:ff:dd:b6:f9:3e:2e:31:98:ba:2e:3a:63:ab:49:ce:58:
         f4:f3:56:a6:04:c0:37:7e:b2:4d:91:e4:48:d8:e8:11:0c:a9:
         cb:7f:af:4d:c9:5c:15:9e:19:01:d4:91:8f:9a:6f:37:90:1e:
         37:05:32:4a:76:a9:be:54:2b:03:a8:44:58:ba:48:ff:ba:52:
         d6:8d:96:4d:8e:67:73:69:69:0e:7c:81:7c:fb:e1:9b:a2:c3:
         96:b6:8e:5a:09:02:35:42:61:c0:84:44:59:15:01:23:4e:79:
         71:99:4f:ef
-----BEGIN CERTIFICATE-----
MIIDNTCCAh2gAwIBAgIJAP5F9eyAX//9MA0GCSqGSIb3DQEBCwUAMBYxFDASBgNV
BAMMC0Vhc3ktUlNBIENBMB4XDTE4MDMyMDExMDQxNloXDTI4MDMxNzExMDQxNlow
FjEUMBIGA1UEAwwLRWFzeS1SU0EgQ0EwggEiMA0GCSqGSIb3DQEBAQUAA4IBDwAw
ggEKAoIBAQDO+gn5mrZZ1RHiiFV9ix7wwaR/5c9yTcsL/3sMqy4rxos3Kao0v+PI
4koQGg1NdMXChLRQfiqjHxTZtx8+WcVf5rBJQPHJhC6Eych1pDFMDOnIUIvOBg+m
ppCGpGbBawE8vcV/Ej3G8YMFvz9X0SvcY8RBEdk2Fvqa7NzHuHz2ARXOGC1x9Uc0
opL7CS7OKqaH8YgdPbgf344HgJu5+Bir3nU5OpZ2RaCp3MSob2luOuYac3Tyiasn
fnOvVGfMkizHERQaXQ3zsyXRNujf5VagOBnR42dSvLurMnO+CuytrPIrPWvLjSqg
wVVfQlfNFYQsyYcE44fahOozgpuzox5tAgMBAAGjgYUwgYIwHQYDVR0OBBYEFC7O
646JvIqmhTgXYk3JaPSvjx2jMEYGA1UdIwQ/MD2AFC7O646JvIqmhTgXYk3JaPSv
jx2joRqkGDAWMRQwEgYDVQQDDAtFYXN5LVJTQSBDQYIJAP5F9eyAX//9MAwGA1Ud
EwQFMAMBAf8wCwYDVR0PBAQDAgEGMA0GCSqGSIb3DQEBCwUAA4IBAQABaob84oD/
fEDZ6kkGHMZsUUGe78BfoFaadX7K46m7lb65SEo3izJ6UqO1kouUAfrj47YZtvKs
hkKOn7Ri/cqJvtn6maGgaG3UCO85mVGr3uXxMsNJUd5iaT4lRrKokd6PqfqqMK9F
iyJ/jCfifTr27sT9H187iGv75PHSrXsWSleURfTCuE94fslsEpwHI/aP6P/dtvk+
LjGYui46Y6tJzlj081amBMA3frJNkeRI2OgRDKnLf69NyVwVnhkB1JGPmm83kB43
BTJKdqm+VCsDqERYukj/ulLWjZZNjmdzaWkOfIF8++GbosOWto5aCQI1QmHAhERZ
FQEjTnlxmU/v
-----END CERTIFICATE-----
```

==========

2. ¿Cómo sabes que es un certificado autofirmado?

Podemos verlo en el certificado, ya que el firmante en la CA

```
Issuer: CN=Easy-RSA CA
.
.
.
Subject: CN=Easy-RSA CA
```

==========

3. ¿El certificado lleva algún campo que indique que pertenece a una CA?

```
Subject: CN=Easy-RSA CA Aqui se indica que pertece a la CA
```

++

++
2.1.2. Creación de las claves y el certificado firmado para el servidor r4

1. Incluye en la memoria el contenido de la solicitud del certificado en formato legible.

metiendonos en la carpeta de easy_r4, en pki/reqs, ejecutamos el siguiente comando: openssl req -noout -text -in r4.req

```
Certificate Request:
    Data:
        Version: 0 (0x0)
        Subject: CN=r4
        Subject Public Key Info:
            Public Key Algorithm: rsaEncryption
                Public-Key: (2048 bit)
                Modulus:
                    00:c9:ec:80:1e:ab:75:41:eb:02:92:26:01:41:f9:
                    41:8b:4e:1c:c5:45:fe:83:9a:7b:16:6e:fe:80:cf:
                    74:eb:78:fa:ec:27:66:8d:be:57:25:7e:fd:a9:7d:
                    77:98:4e:d7:cd:73:25:4e:4c:92:db:4d:a9:b8:9d:
                    79:8d:48:38:62:af:f4:88:52:a5:f1:4d:fa:64:49:
                    1f:eb:c8:f3:bf:15:77:41:71:11:93:7f:15:65:bb:
                    a3:48:86:77:f8:7c:51:8a:19:ff:f0:ab:32:6f:1d:
                    8a:f4:d7:a5:72:cf:16:a0:80:63:1e:99:a2:e7:f0:
                    eb:6a:9f:3d:33:23:d4:bc:31:df:7b:62:ef:d5:05:
                    b2:1b:b9:6c:ff:d3:c1:bb:81:cc:d1:0f:6a:1a:06:
                    c9:99:0c:9e:fa:ad:90:15:75:52:8e:42:5b:e8:56:
                    9b:58:24:cd:60:80:6e:cb:9b:5f:e8:88:9a:66:e0:
                    be:e8:c8:22:69:ce:e1:65:29:96:bc:06:30:f0:9b:
                    57:99:5c:61:92:45:9e:14:44:01:ba:3d:9e:06:df:
                    43:9b:b8:c3:38:05:37:8c:10:a8:ce:fc:a7:c3:33:
                    de:50:64:80:9f:0a:2d:c0:5c:0d:8b:f6:cc:95:4a:
                    3c:b0:72:c1:b0:d4:ff:ce:e6:ab:9f:7b:19:c8:9e:
                    77:1b
                Exponent: 65537 (0x10001)
        Attributes:
            a0:00
    Signature Algorithm: sha256WithRSAEncryption
         8d:45:ab:e7:44:64:fd:f6:21:01:ec:39:c8:2e:49:c8:88:1f:
         68:0a:3c:a7:9e:2a:73:90:d8:20:ad:06:49:47:81:b0:38:a9:
         57:2a:8b:94:3e:f2:7a:e0:d0:bf:96:a3:87:d6:7b:6c:c9:4d:
         ef:61:1d:99:b4:f7:ce:9c:a9:9d:86:61:28:6a:cf:7f:37:bb:
         15:37:1a:9a:f5:78:0e:d7:e9:72:e2:8b:b8:0a:cf:ff:92:ee:
         b3:77:b5:68:b6:bd:32:1c:ed:b2:e5:cb:46:76:6a:26:97:e7:
         9e:37:83:77:1a:d9:2f:a4:4a:6a:52:bf:21:2d:9f:a8:9e:6c:
         52:c7:70:6d:18:4e:62:11:06:11:1d:de:6c:e8:dd:8a:a4:1c:
         a0:96:c7:35:e9:89:e2:92:55:29:b4:1c:6b:b0:ba:6c:7c:89:
         d2:e6:73:21:fc:0b:a8:6e:6e:74:87:39:07:f0:58:06:bb:75:
         6e:7b:74:a1:2c:0e:8e:15:61:38:3d:76:c2:88:56:c2:a7:d9:
         fb:6f:e4:32:4d:41:76:ba:6b:37:a7:67:cf:bc:c5:f5:63:bb:
         c7:ed:0f:27:9c:11:56:10:8e:2f:c7:53:fc:c2:4d:ba:92:35:
         9c:fe:8c:e7:88:ce:2a:72:6e:29:ae:80:d9:fa:e7:ab:20:ce:
```

c1:57:dc:6b

==========

2. Genera los parámetros Diffie-Hellman que después se utilizarán para configurar openVPN. Copia el
resultado de visualizar el fichero en modo legible.

hacemos: cat dh.pem, en la carpeta de pki

```
-----BEGIN DH PARAMETERS-----
MIIBCAKCAQEA0LpwCDandUojr7lLa3xz6f4sGDCuP3NzKPua85xsoRaL3yJIz2hG
7Dse7ygvhR46oXo+XbmXIX2m3LPpzXfkphDw1bIqnW/lfLG9UmnwuRUglv5EVFZn
EQVi+Wja8tI6bstgJPplFaSRt/ycv13PewWBME0Azyw2sfMHt8RG7RRXj4h4AjcQ
w1DSX27NakWBlQgpufkiy905Z9gBgOw8Z24etN0dM5NLBzplyssldGK7DSoDTfJP
dxrVnP3sGeVC+vJ4FxDKtQPO4gqyGcwLAng/V9Wjn6ejGewTdE1f+XbnU77VES5p
yAgy7o33bnPoSyKugUjpgfH4KW3pvD310wIBAg==
-----END DH PARAMETERS-----
```

==========

3. Importa la solicitud de certificado desde la carpeta de la autoridad de certificación y comprueba que se
ha importado correctamente, visualizando la solicitud que acabas de importar. Copia el resultado en la
memoria (el campo Attributes: a0:00 significa que no hay atributos).

usamos el siguiente comando dentro de la carpeta easyrsa3: ./easyrsa import-req <carpeta/nombreMáqServidor>.req <nombreMáqServidor>

y este en la misma carpeta, para comprobar que hemos hecho bien los pasos: ./easyrsa show-req <nombreMáqServidor>, donde nombreMáqServidor es r4

```
Showing req details for 'r4'.
This file is stored at:
/home/david/Escritorio/Seguridad-en-Redes/vpn/generacion-claves/easy_CA/easy-rsa/easyrsa3/pki/reqs/r4.req

Certificate Request:
    Data:
        Version: 0 (0x0)
        Subject:
            commonName                = r4
        Attributes:
            a0:00
```

==========

4. Firma el certificado del servidor con la CA e incluye en la memoria el certificado del servidor firmado
en modo legible.

ejecutamos ./easyrsa sign-req server <nombreMáqServidor>, donde nombreMáqServidor es r4

hacemos un cat de r4.crt en la carpeta issued:

```
Certificate:
    Data:
        Version: 3 (0x2)
        Serial Number:
            17:03:74:b4:17:2f:b6:bc:7d:7b:60:79:54:a3:8c:86
    Signature Algorithm: sha256WithRSAEncryption
        Issuer: CN=Easy-RSA CA
        Validity
            Not Before: Mar 20 11:37:12 2018 GMT
            Not After : Mar 17 11:37:12 2028 GMT
        Subject: CN=r4
        Subject Public Key Info:
            Public Key Algorithm: rsaEncryption
                Public-Key: (2048 bit)
                Modulus:
                    00:c9:ec:80:1e:ab:75:41:eb:02:92:26:01:41:f9:
                    41:8b:4e:1c:c5:45:fe:83:9a:7b:16:6e:fe:80:cf:
                    74:eb:78:fa:ec:27:66:8d:be:57:25:7e:fd:a9:7d:
                    77:98:4e:d7:cd:73:25:4e:4c:92:db:4d:a9:b8:9d:
                    79:8d:48:38:62:af:f4:88:52:a5:f1:4d:fa:64:49:
                    1f:eb:c8:f3:bf:15:77:41:71:11:93:7f:15:65:bb:
                    a3:48:86:77:f8:7c:51:8a:19:ff:f0:ab:32:6f:1d:
                    8a:f4:d7:a5:72:cf:16:a0:80:63:1e:99:a2:e7:f0:
                    eb:6a:9f:3d:33:23:d4:bc:31:df:7b:62:ef:d5:05:
                    b2:1b:b9:6c:ff:d3:c1:bb:81:cc:d1:0f:6a:1a:06:
                    c9:99:0c:9e:fa:ad:90:15:75:52:8e:42:5b:e8:56:
                    9b:58:24:cd:60:80:6e:cb:9b:5f:e8:88:9a:66:e0:
                    be:e8:c8:22:69:ce:e1:65:29:96:bc:06:30:f0:9b:
                    57:99:5c:61:92:45:9e:14:44:01:ba:3d:9e:06:df:
                    43:9b:b8:c3:38:05:37:8c:10:a8:ce:fc:a7:c3:33:
                    de:50:64:80:9f:0a:2d:c0:5c:0d:8b:f6:cc:95:4a:
                    3c:b0:72:c1:b0:d4:ff:ce:e6:ab:9f:7b:19:c8:9e:
                    77:1b
                Exponent: 65537 (0x10001)
        X509v3 extensions:
            X509v3 Basic Constraints:
                CA:FALSE
            X509v3 Subject Key Identifier:

67:BE:EA:50:26:D1:F2:4E:B6:2E:9B:B1:5D:95:EA:23:93:20:80:58
            X509v3 Authority Key Identifier:

keyid:2E:CE:EB:8E:89:BC:8A:A6:85:38:17:62:4D:C9:68:F4:AF:8F:1D:A3
                DirName:/CN=Easy-RSA CA
                serial:FE:45:F5:EC:80:5F:FF:FD

            X509v3 Extended Key Usage:
                TLS Web Server Authentication
            X509v3 Key Usage:
                Digital Signature, Key Encipherment
            X509v3 Subject Alternative Name:
                DNS:r4
    Signature Algorithm: sha256WithRSAEncryption
```

```
        5a:5f:13:2a:11:8c:b7:32:52:9f:41:8f:3e:f2:3a:91:95:06:
        be:de:6f:93:bf:ff:e6:f9:9e:6e:63:72:d5:28:6c:2d:3d:e7:
        9a:d1:fa:ec:d5:2e:f0:45:36:94:e4:44:53:6e:36:6c:e7:6c:
        f5:d5:0e:9a:c3:31:41:13:35:4e:5e:88:51:97:cd:07:2b:6b:
        c8:0c:6f:f6:b7:dd:94:1b:46:a7:54:df:13:5e:e9:ca:bc:68:
        0b:b0:71:6b:71:5c:56:79:57:5c:3f:a8:95:73:d7:ad:0f:72:
        42:d2:fc:dd:c6:d1:28:e9:15:60:bf:b9:bb:32:a4:8a:11:0b:
        fa:eb:6c:2d:ca:e3:35:aa:8c:1c:71:aa:ad:ff:aa:2a:ec:ce:
        7d:c6:e4:7a:72:89:6d:aa:fe:db:9f:a5:5d:7a:c3:8a:30:fc:
        ac:eb:74:ff:9b:63:a3:a3:5d:6e:b3:e8:b8:bc:27:02:7b:e7:
        e0:53:04:13:0d:fa:7a:cf:b5:57:13:62:01:fe:fa:61:86:60:
        23:bc:4b:e4:cc:e5:2b:8a:fd:6b:0d:05:8f:34:9a:d9:87:0e:
        29:09:50:35:10:af:9c:19:ee:d1:34:b5:73:f6:5a:f5:34:a6:
        9d:fc:25:c5:69:8a:e2:73:cb:18:69:82:31:6c:d5:9f:0a:22:
        bd:9f:97:d0
-----BEGIN CERTIFICATE-----
MIIDVDCCAjygAwIBAgIQFwN0tBcvtrx9e2B5VKOMhjANBgkqhkiG9w0BAQsFADAW
MRQwEgYDVQQDDAtFYXN5LVJTQSBDQTAeFw0xODAzMjAxMTM3MTJaFw0yODAzMTcx
MTM3MTJaMA0xCzAJBgNVBAMMAnI0MIIBIjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIB
CgKCAQEAyeyAHqt1QesCkiYBQflBi04cxUX+g5p7Fm7+gM9063j67Cdmjb5XJX79
qX13mE7XzXMlTkyS202puJ15jUg4Yq/0iFKl8U36ZEkf68jzvxV3QXERk38VZbuj
SIZ3+HxRihn/8Ksybx2K9Nelcs8WoIBjHpmi5/Drap89MyPUvDHfe2Lv1QWyG7ls
/9PBu4HM0Q9qGgbJmQye+q2QFXVSjkJb6FabWCTNYIBuy5tf6IiaZuC+6Mgiac7h
ZSmWvAYw8JtXmVxhkkWeFEQBuj2eBt9Dm7jDOAU3jBCozvynwzPeUGSAnwotwFwN
i/bMlUo8sHLBsNT/zuarn3sZyJ53GwIDAQABo4GmMIGjMAkGA1UdEwQCMAAwHQYD
VR0OBBYEFGe+6lAm0fJOti6bsV2V6iOTIIBYMEYGA1UdIwQ/MD2AFC7O646JvIqm
hTgXYk3JaPSvjx2joRqkGDAWMRQwEgYDVQQDDAtFYXN5LVJTQSBDQYIJAP5F9eyA
X//9MBMGA1UdJQQMMAoGCCsGAQUFBwMBMAsGA1UdDwQEAwIFoDANBgNVHREEBjAE
ggJyNDANBgkqhkiG9w0BAQsFAAOCAQEAWl8TKhGMtzJSn0GPPvI6kZUGvt5vk7//
5vmebmNy1ShsLT3nmtH67NUu8EU2lOREU242bOds9dUOmsMxQRM1Tl6IUZfNBytr
yAxv9rfdlBtGp1TfE17pyrxoC7Bxa3FcVnlXXD+olXPXrQ9yQtL83cbRKOkVYL+5
uzKkihEL+utsLcrjNaqMHHGqrf+qKuzOfcbkenKJbar+25+lXXrDijD8rOt0/5tj
o6NdbrPouLwnAnvn4FMEEw36es+1VxNiAf76YYZgI7xL5MzlK4r9aw0FjzSa2YcO
KQlQNRCvnBnu0TS1c/Za9TSmnfwlxWmK4nPLGGmCMWzVnwoivZ+X0A==
-----END CERTIFICATE-----
```

==========

5. Fıjate en el certificado, ¿se puede saber que no pertenece a una autoridad de certificación?

Ha sido firmado por nuestra autoridad de certificacion, pero aparece un campo: CA:FALSE, el cual nos avisa de que no es una autentica CA

++++++++++

++++++++++

2.1.3. Creación de las claves y certificado firmado para el cliente pc3

1. Incluye en la memoria el contenido de la solicitud del certificado en formato legible.

Al igual que en el apartado anterior:

comando: openssl req -noout -text -in pc3.req

Certificate Request:

```
    Data:
        Version: 0 (0x0)
        Subject: CN=pc3
        Subject Public Key Info:
            Public Key Algorithm: rsaEncryption
                Public-Key: (2048 bit)
                Modulus:
                    00:bc:20:29:94:b6:e9:63:79:93:64:50:17:fd:c3:
                    11:19:ec:1c:90:9d:c5:a7:e0:11:6c:cc:70:5a:dc:
                    91:c3:6f:ea:05:c8:87:2f:b0:0d:a5:64:b4:16:6f:
                    54:76:5c:b4:7d:0b:29:d5:85:28:dc:28:c7:13:95:
                    b4:37:06:ba:31:9a:e8:3f:03:9f:7e:2c:a7:9f:ab:
                    fc:d9:79:a2:a1:20:34:01:a4:ab:65:61:3f:0c:45:
                    2e:e4:27:d1:16:76:69:30:f8:40:49:e0:2d:e9:3c:
                    f0:38:ba:1a:22:0a:b2:fb:e2:e1:f8:e8:80:26:5a:
                    e0:13:84:61:f1:ae:f5:06:fe:60:8c:51:8b:92:6e:
                    50:f4:84:c2:83:7b:6b:70:d8:05:99:9c:a6:2d:71:
                    2c:dc:3e:2d:14:d1:0f:3e:25:c3:44:17:d7:3e:c0:
                    5d:38:30:48:c5:e2:0d:11:08:47:5c:80:aa:17:e5:
                    dc:6d:a1:55:ed:1b:8d:fe:a7:4a:b5:c6:52:19:91:
                    e5:1d:2d:cb:9f:3e:5d:64:d6:b5:22:d1:bf:aa:28:
                    46:0f:f4:26:5d:df:c6:59:84:07:bd:bd:bd:01:a0:
                    d6:1b:b0:bf:b4:78:ca:0b:8e:27:5a:57:af:85:08:
                    7f:b9:1c:0f:81:af:4e:2c:2a:b9:52:d9:e4:9b:28:
                    18:c5
                Exponent: 65537 (0x10001)
        Attributes:
            a0:00
    Signature Algorithm: sha256WithRSAEncryption
         5d:a2:62:0c:7e:f0:a1:84:bf:df:14:23:74:96:79:ac:be:e1:
         69:1a:d7:2c:e7:3d:14:b7:4c:de:57:f0:a2:65:10:19:8f:14:
         5c:8f:66:b5:76:24:63:7b:64:67:b9:4f:fb:ac:3b:87:5c:55:
         b5:c8:26:0e:ce:53:9c:af:33:00:38:af:75:76:41:ea:27:cd:
         15:e6:f0:e9:c9:19:cf:0f:41:09:89:23:5d:d1:41:dc:04:9a:
         ed:b0:3e:05:6a:bf:b4:af:b6:33:22:a1:92:74:84:c0:a0:20:
         fb:43:77:53:13:47:ec:b6:2d:19:ef:d9:0d:8c:83:10:dc:ff:
         4f:62:df:71:25:29:7c:ca:c7:06:08:5d:f6:d8:28:7a:69:88:
         38:eb:9e:52:7a:8a:4c:ae:42:5b:0e:e8:c7:69:3b:69:cf:74:
         5e:f7:42:85:08:ed:4f:bf:a8:5a:cc:56:a0:0a:f7:5e:53:f3:
         a0:53:f3:8c:aa:a9:ac:d9:ea:c0:a6:f3:d5:35:2e:d2:ea:1e:
         d9:e1:b9:17:d4:7d:ca:0b:9d:88:b6:bc:c3:94:13:74:4f:c1:
         b9:6a:0b:77:1a:c7:5f:de:1a:3a:ee:a8:cf:3c:bb:09:5b:2a:
         16:f1:83:dc:1a:d4:6d:b1:2d:d1:62:ed:d6:af:9a:bb:a5:48:
         bd:dd:85:8d
```

==========

2. Importa la solicitud de certificado desde la carpeta de la autoridad de certificación y comprueba que se
ha importado correctamente, visualizando la solicitud que acabas de importar. Copia el resultado en la
memoria.

al igual que en el apartado anterior:
(dentro de easy_pc3): ./easyrsa show-req pc3

```
Showing req details for 'pc3'.
This file is stored at:
/home/david/Escritorio/Seguridad-en-Redes/vpn/generacion-
claves/easy_CA/easy-rsa/easyrsa3/pki/reqs/pc3.req


Certificate Request:
    Data:
        Version: 0 (0x0)
        Subject:
            commonName                = pc3
        Attributes:
            a0:00
```

==========

3. Firma el certificado del cliente con la CA e incluye en la memoria el certificado del cliente firmado en
modo legible.

ejecutamos: ./easyrsa sign-req client <nombreMáqServidor>, donde nombreMáqServidor es pc3

hacemos un cat de pc3.crt en la carpeta issued:

```
Certificate:
    Data:
        Version: 3 (0x2)
        Serial Number:
            93:8b:0b:f6:b8:29:a4:1c:68:f4:dc:4e:c9:e7:0a:b4
    Signature Algorithm: sha256WithRSAEncryption
        Issuer: CN=Easy-RSA CA
        Validity
            Not Before: Mar 20 11:43:33 2018 GMT
            Not After : Mar 17 11:43:33 2028 GMT
        Subject: CN=pc3
        Subject Public Key Info:
            Public Key Algorithm: rsaEncryption
                Public-Key: (2048 bit)
                Modulus:
                    00:bc:20:29:94:b6:e9:63:79:93:64:50:17:fd:c3:
                    11:19:ec:1c:90:9d:c5:a7:e0:11:6c:cc:70:5a:dc:
                    91:c3:6f:ea:05:c8:87:2f:b0:0d:a5:64:b4:16:6f:
                    54:76:5c:b4:7d:0b:29:d5:85:28:dc:28:c7:13:95:
                    b4:37:06:ba:31:9a:e8:3f:03:9f:7e:2c:a7:9f:ab:
                    fc:d9:79:a2:a1:20:34:01:a4:ab:65:61:3f:0c:45:
                    2e:e4:27:d1:16:76:69:30:f8:40:49:e0:2d:e9:3c:
                    f0:38:ba:1a:22:0a:b2:fb:e2:e1:f8:e8:80:26:5a:
                    e0:13:84:61:f1:ae:f5:06:fe:60:8c:51:8b:92:6e:
                    50:f4:84:c2:83:7b:6b:70:d8:05:99:9c:a6:2d:71:
                    2c:dc:3e:2d:14:d1:0f:3e:25:c3:44:17:d7:3e:c0:
                    5d:38:30:48:c5:e2:0d:11:08:47:5c:80:aa:17:e5:
                    dc:6d:a1:55:ed:1b:8d:fe:a7:4a:b5:c6:52:19:91:
                    e5:1d:2d:cb:9f:3e:5d:64:d6:b5:22:d1:bf:aa:28:
```

```
                    46:0f:f4:26:5d:df:c6:59:84:07:bd:bd:bd:01:a0:
                    d6:1b:b0:bf:b4:78:ca:0b:8e:27:5a:57:af:85:08:
                    7f:b9:1c:0f:81:af:4e:2c:2a:b9:52:d9:e4:9b:28:
                    18:c5
                Exponent: 65537 (0x10001)
        X509v3 extensions:
            X509v3 Basic Constraints:
                CA:FALSE
            X509v3 Subject Key Identifier:

01:A9:0D:06:6A:54:47:ED:91:8C:D6:F1:41:05:EB:E4:1B:92:3E:98
            X509v3 Authority Key Identifier:

keyid:2E:CE:EB:8E:89:BC:8A:A6:85:38:17:62:4D:C9:68:F4:AF:8F:1D:A3
                DirName:/CN=Easy-RSA CA
                serial:FE:45:F5:EC:80:5F:FF:FD

            X509v3 Extended Key Usage:
                TLS Web Client Authentication
            X509v3 Key Usage:
                Digital Signature
    Signature Algorithm: sha256WithRSAEncryption
         0d:7e:8e:48:e6:56:7c:3a:14:a2:93:b9:c7:a3:f1:c4:74:37:
         30:75:0b:22:59:ed:c0:8d:0c:df:d3:22:da:d2:79:f1:d9:b6:
         b0:52:13:94:f2:bf:e2:bd:ed:b1:79:2e:f6:3e:b6:53:fb:1c:
         a3:84:df:0c:75:17:03:84:8b:2e:75:9f:84:e9:82:0d:1e:45:
         19:57:d4:2b:4e:07:dc:8d:10:3f:cb:88:1a:ad:59:95:b4:1c:
         59:eb:5d:2b:cf:08:5c:18:8a:02:ba:96:7c:2b:94:d7:5a:6a:
         43:ae:f4:87:04:76:cf:17:e4:ad:7a:d4:e7:d7:75:82:e9:38:
         14:7c:d3:73:ed:9b:33:94:d4:fd:09:02:f9:a0:be:d3:fd:49:
         08:82:16:ff:23:f5:75:7a:02:26:0b:fa:9f:c8:4f:ad:aa:18:
         78:62:b6:75:33:3f:da:42:46:07:4f:25:f1:80:a4:d2:05:4e:
         56:ca:5e:84:ef:a4:4c:92:ec:54:06:d1:34:c2:4f:4c:d5:ed:
         35:c6:f0:f6:45:be:f5:0b:5a:bd:06:84:76:b6:18:c9:cc:6a:
         17:bc:54:a1:ad:36:d5:a7:4b:70:69:67:f0:d9:22:72:3d:c1:
         62:64:68:7e:e1:db:08:eb:29:2f:47:7e:5e:58:e2:4f:9b:8a:
         d3:f7:b7:56
-----BEGIN CERTIFICATE-----
MIIDRzCCAi+gAwIBAgIRAJOLC/a4KaQcaPTcTsnnCrQwDQYJKoZIhvcNAQELBQAw
FjEUMBIGA1UEAwwLRWFzeS1SU0EgQ0EwHhcNMTgwMzIwMTE0MzMzWhcNMjgwMzE3
MTE0MzMzWjAOMQwwCgYDVQQDDANwYzMwggEiMA0GCSqGSIb3DQEBAQUAA4IBDwAw
ggEKAoIBAQC8ICmUtuljeZNkUBf9wxEZ7ByQncWn4BFszHBa3JHDb+oFyIcvsA2l
ZLQWb1R2XLR9CynVhSjcKMcTlbQ3Broxmug/A59+LKefq/zZeaKhIDQBpKtlYT8M
RS7kJ9EWdmkw+EBJ4C3pPPA4uhoiCrL74uH46IAmWuAThGHxrvUG/mCMUYuSblD0
hMKDe2tw2AWZnKYtcSzcPi0U0Q8+JcNEF9c+wF04MEjF4g0RCEdcgKoX5dxtoVXt
G43+p0q1xlIZkeUdLcufPl1k1rUi0b+qKEYP9CZd38ZZhAe9vb0BoNYbsL+0eMoL
jidaV6+FCH+5HA+Br04sKrlS2eSbKBjFAgMBAAGjgZcwgZQwCQYDVR0TBAIwADAd
BgNVHQ4EFgQUAakNBmpUR+2RjNbxQQXr5BuSPpgwRgYDVR0jBD8wPYAULs7rjom8
iqaFOBdiTclo9K+PHaOhGqQYMBYxFDASBgNVBAMMC0Vhc3ktUlNBIENBggkA/kX1
7IBf//0wEwYDVR0lBAwwCgYIKwYBBQUHAwIwCwYDVR0PBAQDAgeAMA0GCSqGSIb3
DQEBCwUAA4IBAQANfo5I5lZ8OhSik7nHo/HEdDcwdQsiWe3AjQzf0yLa0nnx2baw
UhOU8r/ive2xeS72PrZT+xyjhN8MdRcDhIsudZ+E6YINHkUZV9QrTgfcjRA/y4ga
rVmVtBxZ610rzwhcGIoCupZ8K5TXWmpDrvSHBHbPF+StetTn13WC6TgUfNNz7Zsz
lNT9CQL5oL7T/UkIghb/I/V1egImC/qfyE+tqhh4YrZ1Mz/aQkYHTyXxgKTSBU5W
```

```
yl6E76RMkuxUBtE0wk9M1e01xvD2Rb71C1q9BoR2thjJzGoXvFShrTbVp0twaWfw
2SJyPcFiZGh+4dsI6ykvR35eWOJPm4rT97dW
-----END CERTIFICATE-----
```

++
++

2.1.4. Creación de las claves y certificado firmado para el cliente r1

1. Incluye en la memoria el contenido de la solicitud del certificado en formato legible.

comando: openssl req -noout -text -in r1.req

```
Certificate Request:
    Data:
        Version: 0 (0x0)
        Subject: CN=r1
        Subject Public Key Info:
            Public Key Algorithm: rsaEncryption
                Public-Key: (2048 bit)
                Modulus:
                    00:c3:f1:63:80:84:0e:96:4c:4b:5c:27:57:79:34:
                    25:fe:2b:51:05:f1:1b:36:97:d8:c2:4b:63:32:b0:
                    b5:a2:08:28:5b:16:d3:c1:ae:23:40:05:9c:63:03:
                    f6:95:6a:51:b0:83:6e:47:13:8f:49:74:fc:43:9f:
                    f9:29:4c:f3:37:5c:b2:8a:98:24:5a:61:e5:c1:ae:
                    3a:36:5d:3e:6c:4f:5e:bb:e5:c1:94:74:52:48:48:
                    e5:80:68:e2:d7:23:98:05:7b:fa:89:40:1b:a3:5b:
                    f2:df:d0:39:5b:77:8e:08:24:b9:82:f4:c6:ad:d4:
                    f6:3e:e0:0d:3e:23:dc:39:71:c2:d7:a3:31:5b:ce:
                    1c:25:c8:ed:30:c7:7c:d7:bb:c7:f3:41:82:1c:c9:
                    f4:1c:61:08:ab:e8:f7:af:50:1f:c4:a3:1b:30:5e:
                    88:37:6c:fe:83:49:b3:4a:fc:69:a4:52:05:ba:2f:
                    6a:d1:e8:a2:03:76:2a:f5:9a:65:60:87:a0:60:52:
                    78:18:61:79:1e:94:84:b4:32:2f:02:1b:23:cd:8f:
                    ea:9d:70:43:55:13:4a:af:c6:c1:3e:f3:c5:f3:9e:
                    e4:cc:47:84:b5:5a:88:19:41:f2:3f:d0:0c:8b:f1:
                    9b:b6:69:5b:fc:36:08:f8:62:a5:b5:dc:83:5a:95:
                    3f:47
                Exponent: 65537 (0x10001)
        Attributes:
            a0:00
    Signature Algorithm: sha256WithRSAEncryption
         61:aa:5e:e4:42:13:8d:2a:e2:d4:f7:ba:0f:61:39:e1:8f:e5:
         47:ad:57:a0:17:6b:1f:41:d5:6c:54:b3:10:4f:31:e9:ce:36:
         28:53:33:17:6a:a4:79:33:a8:44:22:c9:6f:a7:9f:11:c8:8f:
         7c:04:b7:f4:70:a6:f0:68:3e:ad:75:66:2a:b6:e4:72:bf:17:
         b6:bb:33:63:6f:d9:f7:14:7d:a5:81:c0:21:31:99:3c:5c:62:
         84:47:ae:59:ce:47:a4:08:47:c4:f9:55:35:dc:05:c0:d5:23:
         2f:ad:4c:85:17:f8:f6:0d:16:ae:15:6f:dd:76:2b:42:27:fd:
         13:d3:91:9d:48:1a:29:42:be:f8:4e:71:0b:56:2d:9e:25:46:
         cb:52:8d:35:c9:71:35:75:84:eb:39:da:3e:b2:aa:6b:2e:48:
         9b:47:7a:e1:f0:cd:08:b8:ed:05:c2:bc:81:79:2c:55:40:fb:
         df:3c:fb:57:15:50:05:83:f1:bd:c0:e5:c1:be:68:63:a4:33:
```

```
          65:62:f7:17:5b:e5:ef:41:6a:6e:f8:69:8d:5d:38:3a:f4:30:
          6d:bb:cf:e4:ba:40:a0:55:bf:ab:b0:68:20:8c:8a:1d:3f:e2:
          37:9c:8a:54:67:02:b2:91:e3:fb:65:b4:dd:cf:42:8a:33:b9:
          a5:24:b2:98
```

==========

2. Importa la solicitud de certificado desde la carpeta de la autoridad de certificación y comprueba que se ha importado correctamente, visualizando la solicitud que acabas de importar. Copia el resultado en la memoria.

(dentro de la CA): ./easyrsa show-req r1

```
Showing req details for 'r1'.
This file is stored at:
/home/david/Escritorio/Seguridad-en-Redes/vpn/generacion-
claves/easy_CA/easy-rsa/easyrsa3/pki/reqs/r1.req

Certificate Request:
    Data:
        Version: 0 (0x0)
        Subject:
            commonName                = r1
        Attributes:
            a0:00
```

==========

3. Firma el certificado del cliente con la CA e incluye en la memoria el certificado del cliente firmado en modo legible.

ejecutamos: ./easyrsa sign-req client <nombreMáqServidor>, donde nombreMáqServidor es r1

hacemos un cat de r1.crt en la carpeta issued:

```
Certificate:
    Data:
        Version: 3 (0x2)
        Serial Number:
            51:fc:20:b1:26:73:cb:be:1b:3d:56:93:43:96:a3:b9
    Signature Algorithm: sha256WithRSAEncryption
        Issuer: CN=Easy-RSA CA
        Validity
            Not Before: Mar 20 11:43:21 2018 GMT
            Not After : Mar 17 11:43:21 2028 GMT
        Subject: CN=r1
        Subject Public Key Info:
            Public Key Algorithm: rsaEncryption
                Public-Key: (2048 bit)
                Modulus:
                    00:c3:f1:63:80:84:0e:96:4c:4b:5c:27:57:79:34:
                    25:fe:2b:51:05:f1:1b:36:97:d8:c2:4b:63:32:b0:
```

```
                    b5:a2:08:28:5b:16:d3:c1:ae:23:40:05:9c:63:03:
                    f6:95:6a:51:b0:83:6e:47:13:8f:49:74:fc:43:9f:
                    f9:29:4c:f3:37:5c:b2:8a:98:24:5a:61:e5:c1:ae:
                    3a:36:5d:3e:6c:4f:5e:bb:e5:c1:94:74:52:48:48:
                    e5:80:68:e2:d7:23:98:05:7b:fa:89:40:1b:a3:5b:
                    f2:df:d0:39:5b:77:8e:08:24:b9:82:f4:c6:ad:d4:
                    f6:3e:e0:0d:3e:23:dc:39:71:c2:d7:a3:31:5b:ce:
                    1c:25:c8:ed:30:c7:7c:d7:bb:c7:f3:41:82:1c:c9:
                    f4:1c:61:08:ab:e8:f7:af:50:1f:c4:a3:1b:30:5e:
                    88:37:6c:fe:83:49:b3:4a:fc:69:a4:52:05:ba:2f:
                    6a:d1:e8:a2:03:76:2a:f5:9a:65:60:87:a0:60:52:
                    78:18:61:79:1e:94:84:b4:32:2f:02:1b:23:cd:8f:
                    ea:9d:70:43:55:13:4a:af:c6:c1:3e:f3:c5:f3:9e:
                    e4:cc:47:84:b5:5a:88:19:41:f2:3f:d0:0c:8b:f1:
                    9b:b6:69:5b:fc:36:08:f8:62:a5:b5:dc:83:5a:95:
                    3f:47
                Exponent: 65537 (0x10001)
        X509v3 extensions:
            X509v3 Basic Constraints:
                CA:FALSE
            X509v3 Subject Key Identifier:

5D:15:65:52:EA:A4:94:6A:7A:24:6A:3B:BE:1A:06:BE:36:EA:00:9D
            X509v3 Authority Key Identifier:

keyid:2E:CE:EB:8E:89:BC:8A:A6:85:38:17:62:4D:C9:68:F4:AF:8F:1D:A3
                DirName:/CN=Easy-RSA CA
                serial:FE:45:F5:EC:80:5F:FF:FD

            X509v3 Extended Key Usage:
                TLS Web Client Authentication
            X509v3 Key Usage:
                Digital Signature
    Signature Algorithm: sha256WithRSAEncryption
         ac:b2:08:5f:a9:a5:91:16:4e:76:bf:4d:c6:7c:33:b4:33:14:
         dd:42:f5:1a:57:f8:91:77:14:47:40:bd:cb:7e:0b:22:dc:16:
         e5:12:d0:e8:1b:8d:87:a2:1c:06:4f:83:8c:7e:81:c6:09:31:
         e7:f9:b3:e9:6c:a9:39:5a:01:37:ed:58:c8:92:10:4c:79:10:
         7f:2e:45:77:7e:32:a4:df:f0:22:0c:55:a3:1a:05:b5:5b:ee:
         7c:17:e4:2d:3e:c5:b3:de:4c:ca:2e:64:9f:78:19:d0:e3:79:
         6f:90:e1:9d:80:7b:6b:f1:6d:f8:a4:48:35:22:47:c5:c1:2f:
         bd:11:52:de:95:2c:9f:a6:90:05:6e:0a:69:bd:d6:09:1b:34:
         fd:09:9d:2c:44:6e:d1:0b:61:77:d4:83:13:ca:24:ca:7c:00:
         2d:40:ae:18:47:d9:ca:91:03:29:be:5d:67:54:06:a9:7d:4b:
         c3:5e:35:1c:09:ff:34:64:48:6b:1a:fc:9a:cc:63:02:d6:b7:
         3a:af:6a:5e:dd:50:28:78:3f:9a:00:2c:4d:de:5f:ac:e5:bc:
         13:11:4b:1b:37:71:f6:aa:d3:df:20:11:d5:fd:e8:aa:a7:82:
         ad:01:34:0a:95:de:a8:d9:29:d5:b8:42:d3:d5:3e:d4:37:4f:
         b1:93:cf:f2
-----BEGIN CERTIFICATE-----
MIIDRTCCAi2gAwIBAgIQUfwgsSZzy74bPVaTQ5ajuTANBgkqhkiG9w0BAQsFADAW
MRQwEgYDVQQDDAtFYXN5LVJTQSBDQTAeFw0xODAzMjAxMTQzMjFaFw0yODAzMTcx
MTQzMjFaMA0xCzAJBgNVBAMMAnIxMIIBIjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIB
CgKCAQEAw/FjgIQOlkxLXCdXeTQl/itRBfEbNpfYwktjMrC1oggoWxbTwa4jQAWc
```

YwP2lWpRsINuRxOPSXT8Q5/5KUzzN1yyipgkWmHlwa46Nl0+bE9eu+XBlHRSSEjl
gGji1yOYBXv6iUAbo1vy39A5W3eOCCS5gvTGrdT2PuANPiPcOXHC16MxW84cJcjt
MMd817vH80GCHMn0HGEIq+j3r1AfxKMbMF6IN2z+g0mzSvxppFIFui9q0eiiA3Yq
9ZplYIegYFJ4GGF5HpSEtDIvAhsjzY/qnXBDVRNKr8bBPvPF857kzEeEtVqIGUHy
P9AMi/Gbtmlb/DYI+GKltdyDWpU/RwIDAQABo4GXMIGUMAkGA1UdEwQCMAAwHQYD
VR0OBBYEFF0VZVLqpJRqeiRqO74aBr426gCdMEYGA1UdIwQ/MD2AFC7O646JvIqm
hTgXYk3JaPSvjx2joRqkGDAWMRQwEgYDVQQDDAtFYXN5LVJTQSBDQYIJAP5F9eyA
X//9MBMGA1UdJQQMMAoGCCsGAQUFBwMCMAsGA1UdDwQEAwIHgDANBgkqhkiG9w0B
AQsFAAOCAQEArLIIX6mlkRZOdr9NxnwztDMU3UL1Glf4kXcUR0C9y34LItwW5RLQ
6BuNh6IcBk+DjH6Bxgkx5/mz6WypOVoBN+1YyJIQTHkQfy5Fd34ypN/wIgxVoxoF
tVvufBfkLT7Fs95Myi5kn3gZ0ON5b5DhnYB7a/Ft+KRINSJHxcEvvRFS3pUsn6aQ
BW4Kab3WCRs0/QmdLERu0Qthd9SDE8okynwALUCuGEfZypEDKb5dZ1QGqX1Lw141
HAn/NGRIaxr8msxjAta3Oq9qXt1QKHg/mgAsTd5frOW8ExFLGzdx9qrT3yAR1f3o
qqeCrQE0CpXeqNkp1bhC09U+1DdPsZPP8g==
-----END CERTIFICATE-----

++
++
2.2. Almacenar los ficheros adecuadamente para cada máquina

Copia sólo los ficheros de claves y certificados necesarios dentro de la carpeta del escenario:
lab-openvpn/<nombreMáq>/etc/openvpn
Indica en la memoria qué ficheros has almacenado en la carpeta del escenario y en qué carpetas los has
almacenado.

En el server (lab-openvpn/r4/etc/openvpn) he metido:
--> dh.pem
--> r4.crt
--> CA.crt
--> r4.key

En el cliente r1 (lab-openvpn/r1/etc/openvpn) he metido:
--> r1.crt
--> CA.crt
--> r1.key

En el otro cliente, pc3 (lab-openvpn/pc3/etc/openvpn) he metido:
--> pc3.crt
--> CA.crt
--> pc3.key

++
++
2.3. Configuración del extremo servidor r4
Configura el fichero server.conf dentro de lab-openvpn/r4/etc/openvpn de r4 para crear una configu-
ración UDP en el puerto 1194. El servidor openVPN deberá asignar a las máquinas que se comunican a través
de la VPN el siguiente rango de subred: 10.X.8.0/24
Incluye en la memoria las lıńeas que no estén comentadas del fichero server.conf

NOTA: las lineas que tienen varios #, son las que he editado yo. Son para saber, en caso de que algo falle, que ahi es donde puedo mirar.

```
port 1194
proto udp
dev tun
ca /etc/openvpn/ca.crt       ##############
cert /etc/openvpn/r4.crt     ##############
key /etc/openvpn/r4.key  # This file should be kept secret #######
dh /etc/openvpn/dh.pem
topology subnet
server 10.18.8.0 255.255.255.0 ############################
ifconfig-pool-persist /etc/openvpn/ipp.txt
keepalive 10 120
persist-key
persist-tun
status /var/log/openvpn/openvpn-status.log
log         /var/log/openvpn/openvpn.log
verb 6
```

++
++

2.4. Configuración del extremo cliente r1

Configura el fichero client.conf dentro de lab-openvpn/r1/etc/openvpn de r1 para que se conecte al
servidor de r4.

Incluye en la memoria las lıńeas que no estén comentadas del fichero client.conf de r1.

NOTA: las lineas que tienen varios #, son las que he editado yo. Son para saber, en caso de que algo falle, que ahi es donde puedo mirar.

```
client
dev tun
proto udp
remote 100.18.4.4 1194 ########
resolv-retry infinite
nobind
persist-key
persist-tun
ca /etc/openvpn/ca.crt
cert /etc/openvpn/r1.crt
key /etc/openvpn/r1.key
remote-cert-tls server
verb 6
```

++
++

2.5. Configuración del extremo cliente pc3

Configura el fichero client.conf dentro de lab-openvpn/pc3/etc/openvpn de pc3 para que se conecte al
servidor de r4.

Incluye en la memoria las lıńeas que no estén comentadas del fichero client.conf de pc3.

NOTA: las lineas que tienen varios #, son las que he editado yo. Son para saber, en caso de que algo falle, que ahi es donde puedo mirar.

```
client
dev tun
proto udp
remote 100.18.4.4 1194 ########
resolv-retry infinite
nobind
persist-key
persist-tun
ca /etc/openvpn/ca.crt
cert /etc/openvpn/pc3.crt
key /etc/openvpn/pc3.key
remote-cert-tls server
verb 6
```

++
++
2.6. Túnel entre pc3 y r4

1. Copia en la memoria la configuración de la interfaz tun0 que se ha creado en el lado servidor y explica
las direcciones IP que se muestran ¿para qué sirven?

```
tun0      Link encap:UNSPEC  HWaddr 00-00-00-00-00-00-00-00-00-00-00-00-00-
00--00
          inet addr:10.18.8.1  P-t-P:10.18.8.1  Mask:255.255.255.0
          inet6 addr: fe80::4893:c84e:7235:de06/64 Scope:Link
          UP POINTOPOINT RUNNING NOARP MULTICAST  MTU:1500  Metric:1
          RX packets:0 errors:0 dropped:0 overruns:0 frame:0
          TX packets:7 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:100
          RX bytes:0 (0.0 B)  TX bytes:336 (336.0 B)
```

Vemos que para cualquier conexion vpn, se realizara a traves de la interfaz tun0. La direccion 10.18.8.1 es la de r4 en la conexion vpn

==========
2. Mira la tabla de encaminamiento de r4 y copia las entradas que hacen referencia a la interfaz tun0.
Explica su significado

```
root@alonsod:~/Escritorio/Seguridad-en-Redes/vpn/lab-openvpn# route
Kernel IP routing table
Destination     Gateway         Genmask         Flags Metric Ref    Use
Iface
default         100.18.4.3      0.0.0.0         UG    0      0        0
r4-eth0
10.18.2.0       *               255.255.255.0   U     0      0        0
r4-eth1
10.18.8.0       *               255.255.255.0   U     0      0        0
tun0
```

```
100.18.4.0      *               255.255.255.0   U     0      0        0
r4-eth0
```

Como hemos visto, cualquier conexion vpn se realizara a traves de tun0, a las direcciones 10.18.8.0

==========

3. Copia en la memoria la configuración de la interfaz tun que se ha creado en el lado cliente y explica las
direcciones IP que se muestran ¿para qué sirven?

```
tun0      Link encap:UNSPEC  HWaddr 00-00-00-00-00-00-00-00-00-00-00-00-
00-00-00-00
          inet addr:10.18.8.2  P-t-P:10.18.8.2  Mask:255.255.255.0
          inet6 addr: fe80::1d6:54a6:1bce:c84d/64 Scope:Link
          UP POINTOPOINT RUNNING NOARP MULTICAST  MTU:1500  Metric:1
          RX packets:0 errors:0 dropped:0 overruns:0 frame:0
          TX packets:8 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:100
          RX bytes:0 (0.0 B)  TX bytes:384 (384.0 B)
```

Vemos que para cualquier conexion vpn, se realizara a traves de la interfaz tun0. La direccion 10.18.8.2 es la de pc3 en la conexion vpn

==========

4. Mira la tabla de encaminamiento de pc3 y copia las entradas que hacen referencia a la interfaz tun0.
Explica su significado.

```
root@alonsod:~/Escritorio/Seguridad-en-Redes/vpn/lab-openvpn# route
Kernel IP routing table
Destination     Gateway         Genmask         Flags Metric Ref    Use
Iface
default         100.18.7.5      0.0.0.0         UG    0      0        0
pc3-eth0
10.18.8.0       *               255.255.255.0   U     0      0        0
tun0
100.18.7.0      *               255.255.255.0   U     0      0        0
pc3-eth0
```

Como hemos visto, cualquier conexion vpn se realizara a traves de tun0, a las direcciones 10.18.8.0

==========

5. Realiza un ping desde pc3 hacia r4 enviando 3 paquetes ICMP Echo Request sin usar el túnel. ¿Qué
dirección IP destino has especificado para enviar los paquetes sin utilizar el túnel? ¿Por qué?

```
root@alonsod:~/Escritorio/Seguridad-en-Redes/vpn/lab-openvpn# ping -c 3
100.18.4.4
PING 100.18.4.4 (100.18.4.4) 56(84) bytes of data.
64 bytes from 100.18.4.4: icmp_seq=1 ttl=62 time=13.8 ms
```

```
64 bytes from 100.18.4.4: icmp_seq=2 ttl=62 time=6.72 ms
64 bytes from 100.18.4.4: icmp_seq=3 ttl=62 time=0.908 ms

--- 100.18.4.4 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2003ms
rtt min/avg/max/mdev = 0.908/7.159/13.844/5.290 ms
```

He utilizado la direccion 100.18.4.4, que es la de la interfaz publica de r4 (se ve haciendo ifconfig en r4)

==========

6. Realiza un ping desde pc3 hacia r4 enviando 3 paquetes ICMP Echo Request usando el túnel. ¿Qué
dirección IP destino has especificado para enviar los paquetes utilizando el túnel? ¿Por qué?

```
root@alonsod:~/Escritorio/Seguridad-en-Redes/vpn/lab-openvpn# ping -c 3
10.18.8
.1
PING 10.18.8.1 (10.18.8.1) 56(84) bytes of data.
64 bytes from 10.18.8.1: icmp_seq=1 ttl=64 time=0.554 ms
64 bytes from 10.18.8.1: icmp_seq=2 ttl=64 time=1.91 ms
64 bytes from 10.18.8.1: icmp_seq=3 ttl=64 time=1.71 ms

--- 10.18.8.1 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2019ms
rtt min/avg/max/mdev = 0.554/1.396/1.919/0.601 ms
```

He utilizado la direccion 10.18.8.1, que es la de la interfaz del tunel vpn tun0 de r4 (se ve haciendo ifconfig en r4)

==========

7. Interrumpe la captura que has realizado (Ctrl+C). Abre la captura openvpn-01.cap en Wireshark y
explica el contenido:

Cliente: 100.18.7.30 (PC3)
Servidor: 100.18.4.4 (r4)

b) Observa el campo Message Packet-ID indica cuál es su valor incial para el cliente y para el servidor.
Explica cómo va cambiando en cada uno de los paquetes P_CONTROL que se envıán.

En el cliente, el valor inicial es cero: Message Packet-ID: 0
Y en el servidor, tambien el valor inicial es cero: Message Packet-ID: 0

El siguiente paquete P_CONTROL (en el lado del cliente), ha aumentado en 1: Message Packet-ID: 1
Y en el lado del servidor, pasa lo mismo: Message Packet-ID: 1

c) ¿Por qué los mensajes P_ACK no llevan el campo Message Packet-ID?

Los mensajes P_ACK llevan el ID de sesion de la maquina, y del otro extremo, pero no es necesario que lleven el Message Packet-ID porque segun llega un mensaje, el asentimiento es enviado de vuelta al destino.

d) ¿En qué paquete se asiente el mensaje P_CONTROL_HARD_RESET_CLIENT_V2? ¿Cómo lo sabes?

En el primer ack enviado por el servidor, ya que solamente se ha enviado el P_CONTROL del cliente. Ademas, en el campo Message Packet-ID Array Length: 1 ,indica cuantos paquetes ha asentido

e) ¿En qué paquete se asiente el mensaje P_CONTROL_HARD_RESET_SERVER_V2? ¿Cómo lo sabes?

En el primer ack enviado por el cliente, ya que solamente se ha enviado el P_CONTROL del servidor. Ademas, en el campo Message Packet-ID Array Length: 1 ,indica cuantos paquetes ha asentido

f) El primer mensaje que envıá el cliente al servidor del SSL/TLS handshake es Client Hello. ¿Alguno de los campos viaja cifrado?

En principio no, ya que el cliente envıa una lista de los algoritmos de cifrado soportados, lista de metodos de compresion y extensiones. Se adjunta un identificador de sesion SessionId.

g) Localiza el primer mensaje que envıá el servidor al cliente para establecer SSL/TLS handshake, es Server Hello. ¿Alguno de los campos viaja cifrado? Indica el algoritmo de cifrado que se va a usar.

En principio tampoco, ya que solamente va a responder al cliente con el metodo de cifrado que se usara en las comunicaciones posteriores.
Se usara el cifrado AES-256-CBC: Cipher Suite:
TLS_DHE_RSA_WITH_AES_256_CBC_SHA (0x0039)

h) A continuación verás en la captura que Wireshark no reconoce como mensajes de tipo openVPN: un
mensaje del servidor y 2 del cliente. Estos mensajes son el intercambio de certificados y parámetros
Diffie Hellman. La negociación TLS termina con el mensaje del servidor Change Cipher Spec y 4
mensajes Application Data, 2 del cliente y 2 del servidor, que van cifrados.

Change Cipher Spec: numero 123
Application Data: server --> 130, 139
client --> 135, 143

i) Observa el contenido de los mensajes P_DATA. ¿Por qué no llevan el campo Message Packet-ID?

Porque no es necesario ningun tipo de ack de respuesta como que se ha asentido dicho paquete, por eso ese identificador lo se incluye.

j) ¿Qué crees que hay dentro de los mensajes P_DATA?

Al estar cifrados no podemos saber con certeza su cntenido. Tal vez sean algun keepalive o alguna consulta de MTU

==========

8. Comprueba el contenido del fichero ipp.txt en el servidor y explica su contenido.
El servidor guarda en este archivo la asociacion entre maquina y direccion IP asignada para el tunel (no es la direccion publica de los clientes!)
Para ver el contenido del fichero:
(un vez estamos en la carpeta openvpn de r4): sudo cat ipp.txt
Nos saldra esto por pantalla: pc3,10.18.8.2

++++++++++

++++++++++

2.7. Conectividad entre pc3 y pc2

1. ¿Qué crees que ocurrirá si desde pc3 se envıá un ping a la dirección IP de pc2?

Como no se han anunciado las subredes de r1, no se puede realizar el ping entre pc3 y pc2

==========

2. Modifica la configuración de r4 para que anuncie sus subredes internas y desde pc3 funcione un ping a la dirección IP de pc2 a través del túnel OpenVPN. Explica en la memoria qué has modificado.

Paramos el servidor y el cliente si los teniamos lanzados. Dentro de la carpeta openvpn de r4, modificamos el fichero server.conf, donde le añadimos la linea push "route 10.18.2.0 255.255.255.0" (aparece comentada dentro del archivo, pero con otra direccion)
Guardamos el archivo con el nuevo cambio, y lanzamos de nuevo el servidor y el cliente.
Hacemos en pc3: ping -c 3 10.18.2.20
Y vemos que ahora funciona

```
PING 10.18.2.20 (10.18.2.20) 56(84) bytes of data.
64 bytes from 10.18.2.20: icmp_seq=1 ttl=63 time=0.617 ms
64 bytes from 10.18.2.20: icmp_seq=2 ttl=63 time=1.69 ms
64 bytes from 10.18.2.20: icmp_seq=3 ttl=63 time=1.97 ms

--- 10.18.2.20 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2031ms
rtt min/avg/max/mdev = 0.617/1.428/1.978/0.587 ms
```

==========

3. Realiza una captura en r4(eth0) (fichero openvpn-02.cap) y en pc2 (fichero openvpn-03.cap) mientras ejecutas el ping y explica las capturas.

Ejecutamos en PC2(eth0) : tcpdump -i pc2-eth0 -s 0 -w openvpn-03.cap

Ejecutamos en r4(eth0): tcpdump -i r4-eth0 -s 0 -w openvpn-02.cap
Ejecutamos en PC3: ping -c 10 10.18.2.20

En la captura de r4 vemos que hay paquetes de openvpn, entre el cliente y el servidor. Pero en la captura de pc2 no observamos eso; vemos que aparecen mensajes icmp echo request del ping de pc3 a pc2. Por tanto, el tunel funciona correctamente.

==========
4. Explica la tabla de encaminamiento que tiene pc3 y las diferencias con la tabla de encaminamiento que viste en el apartado anterior.

Si ejecutamos "route" en pc3 una vez si funciona el ping, vemos que se ha añadido a su tabla de encaminamiento lo siguiente:

```
root@alonsod:~/Escritorio/Seguridad-en-Redes/vpn/lab-openvpn# route
Kernel IP routing table
Destination     Gateway         Genmask         Flags Metric Ref    Use Iface
default         100.18.7.5      0.0.0.0         UG    0      0        0 pc3-eth0
10.18.2.0       10.18.8.1       255.255.255.0   UG    0      0        0 tun0
10.18.8.0       *               255.255.255.0   U     0      0        0 tun0
100.18.7.0      *               255.255.255.0   U     0      0        0 pc3-eth0
```

como vemos, para poder alcanzar a las direcciones de la subred de r4, debemos ir por la 10.18.8.1

==========
5. ¿Crees que pc2 sabe que se está usando un túnel openVPN? ¿Por qué?
No, porque si analizamos la captura de pc2, vemos que solo hay mensajes echo request del ping de pc3. En ningun momento le llegan mensajes de openvpn.

++
++
2.8. Túnel entre r1 y r4

En este apartado se va a poner a prueba el túnel OpenVPN entre r1 y r4. Mantén la configuración realizada anteriormente y ahora:
+Inicia una captura de tráfico en r4(eth0) de forma que se capture todo el tráfico en el fichero openvpn-04.cap.
+Arranca el cliente en r1.
+Realiza un ping desde r1 hacia r4 enviando 3 paquetes ICMP Echo Request sin usar el túnel.
+Realiza un segundo ping desde r1 hacia r4 enviando 3 paquetes ICMP Echo Request usando el túnel.
+Interrumpe la captura que has realizado (Ctrl+C)

Vemos que ambos ping funcionan:

```
root@alonsod:~/Escritorio/Seguridad-en-Redes/vpn/lab-openvpn# ping -c 3
100.18
.4.4
PING 100.18.4.4 (100.18.4.4) 56(84) bytes of data.
64 bytes from 100.18.4.4: icmp_seq=1 ttl=62 time=4.92 ms
64 bytes from 100.18.4.4: icmp_seq=2 ttl=62 time=1.44 ms
64 bytes from 100.18.4.4: icmp_seq=3 ttl=62 time=0.219 ms

--- 100.18.4.4 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2004ms
rtt min/avg/max/mdev = 0.219/2.194/4.920/1.991 ms


root@alonsod:~/Escritorio/Seguridad-en-Redes/vpn/lab-openvpn# ping -c 3
10.18.
8.1
PING 10.18.8.1 (10.18.8.1) 56(84) bytes of data.
64 bytes from 10.18.8.1: icmp_seq=1 ttl=64 time=0.703 ms
64 bytes from 10.18.8.1: icmp_seq=2 ttl=64 time=1.46 ms
64 bytes from 10.18.8.1: icmp_seq=3 ttl=64 time=1.68 ms

--- 10.18.8.1 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2013ms
rtt min/avg/max/mdev = 0.703/1.282/1.684/0.420 ms
```

1. Comprueba el contenido del fichero ipp.txt en el servidor y explıćalo.

Nos vamos a r4/etc/openvpn y ejecutamos lo siguiente:

```
root@alonsod:~/Escritorio/Seguridad-en-Redes/vpn/lab-
openvpn/r4/etc/openvpn# cat ipp.txt
pc3,10.18.8.2
r1,10.18.8.3
```

vemos que salen dos direcciones, las de los dos clientes que utilizan el tunel.

==========

2. ¿Qué crees que ocurrirá si desde r1 se enviá un ping a pc2? Incluye la tabla de encaminamiento de r1.

estando en r1, ejecutamos lo siguiente:

```
root@alonsod:~/Escritorio/Seguridad-en-Redes/vpn/lab-openvpn# route
Kernel IP routing table
Destination     Gateway         Genmask         Flags Metric Ref    Use
Iface
default         100.18.2.2      0.0.0.0         UG    0      0        0
r1-eth1
10.18.1.0       *               255.255.255.0   U     0      0        0
r1-eth0
10.18.2.0       10.18.8.1       255.255.255.0   UG    0      0        0
tun0
```

```
10.18.8.0       *               255.255.255.0   U     0      0        0
tun0
100.18.2.0      *               255.255.255.0   U     0      0        0
r1-eth1
```

Si se tratase de hacer un ping a pc2 desde r1, en principio si se podria ya que r4 ha anunciado a los miembros del tunel sus subredes.
Lo comprobamos y efectivamente se puede.

```
root@alonsod:~/Escritorio/Seguridad-en-Redes/vpn/lab-openvpn# ping -c 3
10.18.2
.20
PING 10.18.2.20 (10.18.2.20) 56(84) bytes of data.
64 bytes from 10.18.2.20: icmp_seq=1 ttl=63 time=1.79 ms
64 bytes from 10.18.2.20: icmp_seq=2 ttl=63 time=4.41 ms
64 bytes from 10.18.2.20: icmp_seq=3 ttl=63 time=2.47 ms

--- 10.18.2.20 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2003ms
rtt min/avg/max/mdev = 1.797/2.896/4.419/1.111 ms
```

==========

3. ¿Qué crees que ocurrirá si desde pc1 se envıá un ping a pc2? ¿Por qué?

Sin ver de primeras la tabla de encaminamiento de pc1, como r1 establece el tunel con r4 y hemos comprobado que funciona, podriamos decir que deberia funcionar el ping entre pc1 y pc2.
probamos dicho ping pero vemos que no funciona porque r4 no tiene incluida en su tabla de encaminamiento las subredes de r1.

```
root@alonsod:~/Escritorio/Seguridad-en-Redes/vpn/lab-openvpn# ping -c 3
10.18.2.20
PING 10.18.2.20 (10.18.2.20) 56(84) bytes of data.

--- 10.18.2.20 ping statistics ---
3 packets transmitted, 0 received, 100% packet loss, time 2054ms
```

para llegar a las subredes de r1, esta puesta la ruta por defecto, la cual no utiliza el tunel.

```
root@alonsod:~/Escritorio/Seguridad-en-Redes/vpn/lab-
openvpn/r4/etc/openvpn/ccd # route
Kernel IP routing table
Destination     Gateway         Genmask         Flags Metric Ref    Use
Iface
default         100.18.4.3      0.0.0.0         UG    0      0        0
r4-eth0
10.18.2.0       *               255.255.255.0   U     0      0        0
r4-eth1
10.18.8.0       *               255.255.255.0   U     0      0        0
tun0
100.18.4.0      *               255.255.255.0   U     0      0        0
r4-eth0
```

Para solucionarlo hacemos lo siguiente:
El servidor debe incluir en su fichero server.conf:
client-config-dir /etc/openvpn/ccd
route 10.18.1.0 255.255.255.0

creamos una carpeta en r4/etc/openvpn/ llamada ccd.
nos metemos dentro y creamos un archivo con el nombre del cliente (touch r1)
una vez dentro, añadimos la siguiente linea: iroute 10.18.1.0 255.255.255.0

por ultimo, añadimos la siguiente direccion en r4: route add -net 10.18.1.0 gw 10.18.8.1 netmask 255.255.255.0 dev tun0

```
root@alonsod:~/Escritorio/Seguridad-en-Redes/vpn/lab-openvpn/r4/etc/openvpn/ccd # route
Kernel IP routing table
Destination     Gateway         Genmask         Flags Metric Ref    Use Iface
default         100.18.4.3      0.0.0.0         UG    0      0        0 r4-eth0
10.18.1.0       10.18.8.1       255.255.255.0   UG    0      0        0 tun0
10.18.2.0       *               255.255.255.0   U     0      0        0 r4-eth1
10.18.8.0       *               255.255.255.0   U     0      0        0 tun0
100.18.4.0      *               255.255.255.0   U     0      0        0 r4-eth0
```

hacemos un ping desde pc1 hasta pc2, y vemos que funciona:

```
root@alonsod:~/Escritorio/Seguridad-en-Redes/vpn/lab-openvpn# ping -c 3 10.18.2.20
PING 10.18.2.20 (10.18.2.20) 56(84) bytes of data.
64 bytes from 10.18.2.20: icmp_seq=1 ttl=62 time=6.37 ms
64 bytes from 10.18.2.20: icmp_seq=2 ttl=62 time=6.53 ms
64 bytes from 10.18.2.20: icmp_seq=3 ttl=62 time=2.52 ms

--- 10.18.2.20 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2003ms
rtt min/avg/max/mdev = 2.520/5.143/6.539/1.856 ms
```

++
++
2.9. Conectividad entre entre pc3 y pc1

1. ¿Qué crees que ocurrirá si desde pc3 se envıá un ping a r1? ¿Por qué?

En principio deberia funcionar, ya que vemos en su tabla de encaminamiento que para conexiones hacia 10.18.8.0, se utilice la interfaz de tun0.
Hacemos dicho ping desde pc3 hasta r1:

```
root@alonsod:~/Escritorio/Seguridad-en-Redes/vpn/lab-openvpn# ping -c 3
10.18.8
.3
PING 10.18.8.3 (10.18.8.3) 56(84) bytes of data.
From 10.18.8.1: icmp_seq=1 Redirect Host(New nexthop: 10.18.8.3)
64 bytes from 10.18.8.3: icmp_seq=1 ttl=63 time=1.95 ms
From 10.18.8.1: icmp_seq=2 Redirect Host(New nexthop: 10.18.8.3)
64 bytes from 10.18.8.3: icmp_seq=2 ttl=63 time=3.48 ms
From 10.18.8.1: icmp_seq=3 Redirect Host(New nexthop: 10.18.8.3)
64 bytes from 10.18.8.3: icmp_seq=3 ttl=63 time=3.78 ms

--- 10.18.8.3 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2003ms
rtt min/avg/max/mdev = 1.958/3.076/3.785/0.801 ms
```

==========

2. ¿Qué crees que ocurrirá si desde pc3 se envıá un ping a pc1? ¿Por qué?

En principio no deberia funcionar ya que pc3 no tiene en su tabla de encaminamiento la direccion de pc1, es decir, que no se la ha anunciado r4.

```
 root@alonsod:~/Escritorio/Seguridad-en-Redes/vpn/lab-openvpn# ping -c 3
10.18.1.10
PING 10.18.1.10 (10.18.1.10) 56(84) bytes of data.
^C
--- 10.18.1.10 ping statistics ---
2 packets transmitted, 0 received, 100% packet loss, time 1027ms
```

Vemos que no funciona como pensabamos.

==========

3. Modifica la configuración en r4 para permitir que r4 anuncie las subredes internas del cliente r1 (Subred1) al cliente pc3. No olvides guardar el fichero que has modificado en la máquina virtual.

añadimos en r4 (server.conf) lo siguiente:

```
client-to-client
push "route 10.18.1.0 255.255.255.0"
```

De esta forma, r4 anunciara las subredes de r1 a todo cliente que se quiera utilizar el servidor.

```
root@alonsod:~/Escritorio/Seguridad-en-Redes/vpn/lab-openvpn# ping -c 3
10.18.1.10
PING 10.18.1.10 (10.18.1.10) 56(84) bytes of data.
64 bytes from 10.18.1.10: icmp_seq=1 ttl=63 time=2.60 ms
64 bytes from 10.18.1.10: icmp_seq=2 ttl=63 time=3.87 ms
64 bytes from 10.18.1.10: icmp_seq=3 ttl=63 time=2.93 ms

--- 10.18.1.10 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2004ms
rtt min/avg/max/mdev = 2.600/3.134/3.870/0.541 ms
```

si hacemos route en pc3:

```
root@alonsod:~/Escritorio/Seguridad-en-Redes/vpn/lab-openvpn# route
Kernel IP routing table
Destination     Gateway         Genmask         Flags Metric Ref    Use
Iface
default         100.18.7.5      0.0.0.0         UG    0      0        0
pc3-eth0
10.18.1.0       10.18.8.1       255.255.255.0   UG    0      0        0
tun0
10.18.2.0       10.18.8.1       255.255.255.0   UG    0      0        0
tun0
10.18.8.0       *               255.255.255.0   U     0      0        0
tun0
100.18.7.0      *               255.255.255.0   U     0      0        0
pc3-eth0
```

vemos que se ha añadido una nueva ruta para poder llegar hasta las subredes de r1.

==========

4. Realiza una captura de tráfico (openvpn-05.cap) en r4(eth0) que muestre que funciona un ping desde pc3 a pc1. Explica los paquetes capturados.

En la captura vemos el primer mensaje enviado por pc3 (numero 4), el siguiente es un paquete enviado por r4 hacia r1 con destino pc2 de pc3, y el siguiente es un mensaje de respuesta de pc2 a traves de r1, hacia pc3 a traves de r4.

==========

5. Explica la tabla de encaminamiento que tienen pc3 y r1 y las diferencias con la tablas de encaminamiento que tenıán previamente.

tabla de encaminamiento de pc3 antes:

```
root@alonsod:~/Escritorio/Seguridad-en-Redes/vpn/lab-openvpn# route
Kernel IP routing table
Destination     Gateway         Genmask         Flags Metric Ref    Use
Iface
default         100.18.7.5      0.0.0.0         UG    0      0        0
pc3-eth0
10.18.2.0       10.18.8.1       255.255.255.0   UG    0      0        0
tun0
10.18.8.0       *               255.255.255.0   U     0      0        0
tun0
100.18.7.0      *               255.255.255.0   U     0      0        0
pc3-eth0
```

tabla de encaminamiento de pc3 ahora:

```
root@alonsod:~/Escritorio/Seguridad-en-Redes/vpn/lab-openvpn# route
Kernel IP routing table
Destination     Gateway         Genmask         Flags Metric Ref    Use
Iface
default         100.18.7.5      0.0.0.0         UG    0      0        0
pc3-eth0
```

```
10.18.1.0       10.18.8.1       255.255.255.0   UG    0      0        0
tun0
10.18.2.0       10.18.8.1       255.255.255.0   UG    0      0        0
tun0
10.18.8.0       *               255.255.255.0   U     0      0        0
tun0
100.18.7.0      *               255.255.255.0   U     0      0        0
pc3-eth0
```

tabla de encaminamiento de r1 antes:

```
root@alonsod:~/Escritorio/Seguridad-en-Redes/vpn/lab-openvpn# route
Kernel IP routing table
Destination     Gateway         Genmask         Flags Metric Ref    Use
Iface
default         100.18.2.2      0.0.0.0         UG    0      0        0
r1-eth1
10.18.1.0       *               255.255.255.0   U     0      0        0
r1-eth0
10.18.2.0       10.18.8.1       255.255.255.0   UG    0      0        0
tun0
10.18.8.0       *               255.255.255.0   U     0      0        0
tun0
100.18.2.0      *               255.255.255.0   U     0      0        0
r1-eth1
```

tabla de encaminamiento de r1 ahora:

```
root@alonsod:~/Escritorio/Seguridad-en-Redes/vpn/lab-openvpn# route
Kernel IP routing table
Destination     Gateway         Genmask         Flags Metric Ref    Use
Iface
default         100.18.2.2      0.0.0.0         UG    0      0        0
r1-eth1
10.18.1.0       *               255.255.255.0   U     0      0        0
r1-eth0
10.18.2.0       10.18.8.1       255.255.255.0   UG    0      0        0
tun0
10.18.8.0       *               255.255.255.0   U     0      0        0
tun0
100.18.2.0      *               255.255.255.0   U     0      0        0
r1-eth1
```

La tabla de encaminamiento de pc3 ha añadido una nueva ruta para poder llegar a las subredes de r1, qe se las ha anunciado a r4 y asu vez a pc3. Pero la tabla de encaminamiento de r1, en mi caso no han variado respecto al apartado anterior.