Configuració de serveis

Jordi Serra Ruiz

PID_00204283

Universitat Oberta de Catalunya

Índex

Introducció

Una vegada s'ha instal·lat el servidor –el sistema operatiu– cal instal·lar o engegar, en els casos en què els serveis quedin instal·lats amb el sistema operatiu mateix, totes les aplicacions que ens serviran per a donar un servei addicional al nostre servidor.

Un exemple clar d'això són el servidor de fitxers i el servidor d'impressió. S'ha de configurar el servidor per a donar accés als directoris del servidor de fitxers amb la finalitat que els usuaris tinguin un directori personal en el disc d'usuaris ubicat en el servidor. D'aquesta manera, es podran fer molt millor les còpies de seguretat, es podran controlar millor i serà més ràpid fer-les.

Un altre servei molt important en una empresa és el correu electrònic. Cal configurar el servidor perquè rebi els missatges de correu electrònic i els guardi en discos fins que el client de correu electrònic de l'usuari hi accedeixi i els llegeixi. Aquí es veu com s'han de protegir els comptes de correu perquè només hi hagi un client autoritzat que accedeixi als missatges enviats als comptes de correu personals.

Una altra aplicació que requereix una configuració per a protegir les dades de l'empresa és el servidor web i sobretot el servidor d'FTP, que s'ha d'instal·lar i configurar correctament per a no deixar "forats" de seguretat pels quals puguin accedir els intrusos. Aquests protocols d'intercanvi de fitxers són àmpliament usats per a accedir de manera fraudulenta a la informació interna de l'empresa.

Finalment, s'han de vigilar els ports que es deixen oberts en el nostre sistema informàtic. Un port obert de manera errònia implica que un programa maliciós el pugui fer servir per a entrar remotament al sistema i obtenir informació interna de l'empresa.

Un cas típic el constitueix el port 25, que és el que fa servir el correu electrònic; si es coneixen les instruccions d'enviament de correu electrònic, es pot accedir a un servidor que tingui obert aquest port per a enviar correus electrònics com si els hagués enviat la màquina a què s'accedeix.

Objectius

En aquest mòdul pretenem que conegueu la manera de configurar de forma segura alguns dels serveis més populars de la Xarxa (servidor de fitxers, correu, web i FTP). En l'últim apartat d'aquest mòdul veurem com hem de protegir els ports dels nostres servidors.

Els objectius d'aquest mòdul són els següents:

1. Aprendre a instal·lar i configurar els serveis comuns de protocols i eines d'Internet.

2. Conèixer la configuració segura dels serveis més comuns de la xarxa.

3. Conèixer els riscos d'aquests serveis i com s'han de prevenir.

4. Aprendre a obrir i tancar els ports dels nostres servidors.

1. Servidor de fitxers i impressió

Els servidors de fitxers són els servidors que s'encarreguen d'oferir als usuaris de l'empresa un espai de disc, normalment a la xarxa local, per a guardar els documents interns de l'empresa. Un dels avantatges d'aquest tipus de servidors és que per fer la còpia de seguretat o *backup* dels documents dels usuaris només hem de fer una còpia de seguretat del disc del servidor on estan ubicats els directoris o carpetes dels usuaris.

Això també implica una tasca d'educació per part de l'administrador de l'empresa per a ensenyar als usuaris a guardar tots els documents en les unitats de xarxa i deixar el disc dur local de cada ordinador dels usuaris per a emmagatzemar el sistema operatiu i els programes instal·lats de manera local. Si es treballa en carpetes locals o per exemple amb l'escriptori, es fa molt més complicat controlar els documents i fer-ne còpies de seguretat.

Les empreses IBM i Sytek, Inc. van dissenyar l'any 1983 un sistema per a construir i comunicar xarxes petites (LAN). Aquest sistema incloïa una aplicació anomenada *sistema bàsic d'entrada-sortida* de xarxa o *network basic input-out put system* (NetBIOS). Aquesta aplicació estava carregada en la memòria de les màquines i proporcionava una interfície entre els programes. El sistema d'identificació que feia servir era un nom de 16 bytes. Les aplicacions actuals també publiquen a les xarxes en NetBIOS els seus serveis mitjançant aquests noms curts, així donen compatibilitat amb altres aplicacions i equips que hi pugui haver a la xarxa. Els noms NetBIOS han de ser identificatius, és a dir, han de ser únics en la subxarxa on són.

Posteriorment, Microsoft hi va afegir una sèrie de mecanismes per a permetre que el seu sistema operatiu MS-DOS fos capaç de tornar a encaminar les entrades i sortides del disc cap a les xarxes NetBIOS, cosa que va permetre compartir els discos mitjançant la xarxa. D'aquest sistema de compartició d'arxius després se'n va dir *bloc de missatge del servidor* o *server message block* (SMB). Actualment, aquest sistema es coneix com a *sistema d'arxius d'interfície comuna* o *common internate file system* (CIFS).

Per a saber on hi ha les màquines identificades pels noms NetBIOS, es va dissenyar el protocol *NetBIOS name service* (NBNS), que fa funcions anàlogues al servidor de noms de domini (DNS) en el protocol TCP/IP. Actualment, d'aquest servei se'n diu *Windows Internet naming service* (WINS), que és del tot compatible amb l'NBNS.

Les aplicacions SMB/CIFS funcionen acceptant o denegant la petició d'accés al recurs compartit segons els privilegis que tingui cada usuari i utilitza els ports TCP 137, 138 i 139; per tant, haurem de tenir en compte aquests ports a l'hora d'obrir i tancar els ports de les màquines.

1.1. Servidor de fitxers i impressió en el GNU/Linux

En el GNU/Linux hi ha diverses aplicacions que poden fer aquest tipus de serveis, però n'hi ha dues que destaquen per sobre de les altres per l'ús que se n'hi fa i per la utilitat que tenen.

- **Sistema d'arxiu de xarxa** o *network file system* (NFS). És un protocol que permet exportar parts de l'arbre de directoris a altres màquines. Les màquines que importen l'arbre de directoris el munten com si es tractés d'una partició local. Aquest tipus de protocol només permet exportar disc, però és molt adequat en entorns cent per cent GNU/Linux o Unix, encara que no permet exportar discos a entorns Microsoft.

- **SMB** (més conegut com a Samba). És una aplicació de font pública o *open source* (GNU) que permet compartir impressores, a més d'exportar els discos, fitxers, crear controladors de domini i fins i tot el directori actiu dels servidors de Microsoft. És la reimplementació en programari lliure dels protocols SMB/CIFS. Per tant, aquesta aplicació funciona en entorns mixtos (GNU/Linux i Windows).

L'aplicació Samba consisteix en dos o tres dimonis (*daemons*), depenent de les necessitats específiques de cada empresa. En el cas de la configuració mínima, en fan falta dos. Els dimonis que formen el servidor Samba són els següents:

1) `nmbd`: aquest dimoni maneja totes les peticions de registre i resolució de noms. És una eina primària involucrada en la navegació per la xarxa. Aquest dimoni és un dels dos que fan falta com a mínim en tota configuració del Samba.

2) `smbd`: aquest dimoni maneja totes les connexions basades en el protocol TCP/IP per als serveis de discos i de les impressores. També es responsabilitza de l'autenticació. Aquest és el segon dimoni imprescindible en qualsevol configuració del Samba.

3) `winbindd`: només hem d'iniciar aquest dimoni quan el servidor Samba és membre d'un domini en què no és el controlador de domini principal o *primary domain controller* (PDC). També utilitzem aquest dimoni quan hi ha relacions de confiança amb altres dominis. Si volem arrencar aquest dimoni, cal que configurem els paràmetres `idmap uid` i `udmap gid` en el fitxer `smb.conf`.

En aquesta secció veurem com s'ha d'instal·lar i configurar l'aplicació Samba per a treballar en entorns mixtos.

1.1.1. Instal·lació del Samba

Podem instal·lar aquesta aplicació de dues maneres diferents. La primera és aprofitant les ordres que ens ofereix la versió Debian que tenim instal·lada:

```
root# apt-get install samba
```

Si volem que el programa Samba sigui compatible amb l'*Active Directory*, abans d'instal·lar el programari del Samba hem d'instal·lar els paquets següents:

```
root# apt-get install libkrb5-dev
root# apt-get install krb5-user
```

Com que la creació de paquets Debian es fa després de l'aparició d'una versió nova, a causa de la manera en què es creen aquests paquets (s'instal·la la versió compilada i després, amb una eina de creació de paquets, hi indiquem quins fitxers o directoris formen part del paquet), si volem instal·lar l'última versió del Samba, ho hem de fer pel mètode tradicional: baixar el codi font, configurar el programari i finalment compilar tot el programa.

En primer lloc, hem de baixar la darrera versió del programari Samba que trobem al lloc web oficial d'aquesta aplicació (http://www.samba.org). A la part central d'aquest lloc web hi ha una secció anomenada *Current stable releases*, en què hi ha un enllaç cap a l'última versió estable del programari Samba, que s'anomena *Samba 4.0.3* (aquesta és la darrera versió en el moment de redactar aquests materials).

Una vegada tenim l'aplicació en local, només cal clicar i baixar en format comprimit el fitxer amb el programari del Samba i descomprimir-lo:

```
root# tar zxvf samba-4.0.3.tar.gz
```

Entrem al directori que acabem de crear i executem l'ordre:

```
root# cd samba-4.0.3
root# ./configure --help
```

Triem quines opcions de compilació hi volem afegir i executem l'ordre `configure` amb els paràmetres que ens interessa:

```
root# ./configure [... arguments ...]
```

Una vegada hem fet la configuració, si ens interessa l'*Active Directory*, ens hem d'assegurar que el fitxer `include/config.h` conté dues línies semblants a aquestes:

```
#defineix HAVE_KRB5 1
#defineix HAVE_LDAP 1
```

Arribats a aquest punt ja podem compilar:

```
root# make
```

Si la compilació es fa de manera satisfactòria, instal·lem l'aplicació. Per a fer-ho, es poden instal·lar només els executables, només els manuals o tots dos alhora. Les ordres per a fer aquestes instal·lacions són, respectivament, les següents:

```
root# make installbin
root# make installman
root# make install
```

Si la instal·lació que hem dut a terme és una actualització (*upgrade*) des d'una versió anterior i volem tornar enrere (tornar a la versió que estava instal·lada amb anterioritat), una vegada feta la instal·lació hem d'executar l'ordre següent:

```
root# make revert
```

Una vegada acabada la instal·lació ja podem configurar l'aplicació.

1.1.2. Configuració del Samba

En aquesta secció explicarem diferents nivells de configuració del Samba, des del model més senzill fins a la configuració del Samba com a controlador de domini en un entorn mixt entre màquina GNU/Linux i màquina Windows, cosa que ens permetrà controlar qui té accés a les diferents màquines, i per tant, fer més segur l'accés per part dels usuaris. Malgrat que explicarem els requisits necessaris per a fer servir eines de configuració del Samba utilitzant entorns gràfics, tots els exemples que es mostren al llarg d'aquesta secció es basen en l'edició dels fitxers.

La configuració del Samba és en el fitxer de configuració `smb.conf`. Trobarem aquest fitxer normalment en el directori `/etc/samba` o a `/usr/local/samba/lib`, depenent de la distribució que hàgim fet servir. Per a configurar el Samba podem editar aquest fitxer o utilitzar una eina gràfica basada en un entorn web anomenada *SWAT*, que està inclosa en la mateixa instal·lació del Samba.

El fitxer `smb.conf` està configurat en mode text, de manera que el podem editar amb el nostre editor preferit. La sintaxi és semblant als antics fitxers de configuració del Windows i consisteix en diverses seccions independents; cada secció comença amb una paraula entre claudàtors (`[ ]`) i representa un recurs compartit en el servidor. Hi ha una secció especial anomenada `[global]` que conté els paràmetres que afecten tots els recursos en general que es comparteixen amb el Samba. Les línies que hi ha dins de cada secció consisteixen en un parell de claus/valors separades per un símbol (=).

Els administradors de xarxes amb sistemes Microsoft sovint utilitzen la nomenclatura de controlador de domini, membre d'un domini, o servidor *standalone* per a referir-se al servidor que fa les funcions de controlador d'accés a la xarxa i al servidor. També podem configurar el Samba per actuar en tots aquests rols. Cada configuració només proveeix un tipus de servidor, però veurem tres configuracions diferents per a cobrir els tres tipus de serveis que pot oferir.

Modes de seguretat del Samba

En els entorns de xarxes SMB/CIFS només hi ha dos nivells de seguretat:

- *user security level*
- *shared security level*

La implantació d'aquests dos nivells del Samba s'ha fet de manera més extensa i proporciona millors funcionalitats a les inicials en SMB/CIFS. Actualment, el Samba ha implementat una versió de *shared security level*, però ha fet quatre implantacions diferents del nivell de seguretat d'usuari (*user security level*). Per a no confondre la nomenclatura, el Samba anomena la seva implantació dels nivells de seguretat d'SMB/CIFS *modes de seguretat*. Les quatre implementacions del nivell de seguretat d'usuari s'anomenen `user`, `domain`, `ads` i `server`.

Veurem com són els modes de seguretat del Samba abans de veure'n les configuracions.

1) ***Usermode***. El client envia una petició de configuració de la sessió al servidor que inclou l'usuari i la contrasenya. El servidor només pot acceptar o denegar l'accés. En aquesta fase de la negociació el servidor desconeix quin recurs compartit vol utilitzar el client i, per tant, l'acceptació o denegació només té en compte el parell usuari/contrasenya o el nom de màquina. Si el servidor accepta la connexió, és llavors quan el client intenta accedir als recursos compartits, però en totes les peticions d'ús d'aquests recursos no indica la contrasenya, ja que el client espera que tots els drets d'accés s'hagin negociat en la configuració de la sessió. Per a utilitzar aquest nivell de seguretat en el fitxer `smb.conf`, en la secció global, hi hem de posar el paràmetre `security = user`. Aquest és el mode de seguretat per defecte.

2) *Sharedmode*. El client s'autentica de manera independent en cadascun dels recursos compartits que vol utilitzar. Per fer-ho, envia la contrasenya en cada petició d'ús d'un recurs. En aquesta petició no hi ha l'usuari, ja que el client espera que amb les contrasenyes n'hi hagi prou. A diferència d'altres implantacions de nivell de seguretat *shared* que permeten autenticació amb el parell recurs_compartit/contrasenya, el Samba sempre treballa amb el parell usuari/contrasenya (esquema d'autenticació de GNU/Linux per defecte). Per sort, per al Samba, hi ha moltes aplicacions client que envien també l'usuari quan es negocia la connexió; així, el Samba recorda els usuaris i, quan rep la petició de connexió a un recurs compartit només amb la contrasenya, la comprova amb els usuaris que té anotats. Si la contrasenya d'algun d'aquests usuaris coincideix amb la que ha enviat el client, el servidor li permet accedir. Per fer servir aquest mode hem d'afegir en el fitxer `smb.conf`, en la secció global, el paràmetre `Security = shared`. És altament desaconsellable fer servir aquest mode perquè ens ofereix poques garanties de seguretat, ja que s'està enviant contínuament la contrasenya per la xarxa i en el cas de tenir dos usuaris amb la mateixa contrasenya es produirien problemes.

3) *Domainmode*. Quan el Samba opera en el mode de seguretat de domini (*domain mode*), el servidor Samba té un compte de confiança (un compte de la màquina), que és un entorn en què hi ha màquines Windows i en què el Samba no és el controlador del domini, un servidor de la família Windows. Això fa que totes les peticions d'autenticació hagin de passar pel controlador de domini; dit d'una altra manera, aquesta configuració fa que el servidor del Samba sigui membre del domini, però no el controlador. Per fer servir aquest mode de seguretat hem d'afegir en el fitxer `smb.conf` els paràmetres següents:

```
security = domain
workgroup = name (our name group)
```

Per treballar en aquest mode hem d'unir el servidor Samba al domini de seguretat del Windows:

- Fent servir el *Server Manager* en el controlador de domini Windows, afegim un compte de màquina per al servidor Samba.

- En el servidor Samba hem d'executar l'ordre fent servir el compte d'administrador del controlador de domini amb la contrasenya que té:

```
root# net rpc join -O administrador%password
```

Aquest mode també necessita l'assignació d'identificador d'usuari (UID) per als usuaris del Samba. Per fer-ho, tenim dues opcions: crear comptes en el servidor Samba o utilitzar el dimoni `windbindd`.

4) ***ADSmode***. El Samba es pot unir a un domini *Active Directory* de Microsoft. Això és possible si el domini s'executa en el mode natiu. Per què ens interessa accedir des d'una màquina GNU/Linux a un *Active Directory*? Si el nostre entorn està constituït per màquines del Microsoft Windows, fent servir el Kerberos necessitarem tenir accés a l'*Active Directory* per a acceptar els "tiquets" de Kerberos i tenir per tant accés a tots els equips de la xarxa. Per fer servir aquest mode de seguretat hem d'afegir en el fitxer `smb.conf` els paràmetres següents:

```
realm = our.real.kerberos security = ADS
password server = our.server.kerberos
```

5) ***Server mode***. Aquest mode de seguretat es feia servir quan el Samba no era capaç d'actuar com a membre d'un domini. És molt recomanable no fer-lo servir, ja que té molts problemes de seguretat i les altres opcions ja cobreixen aquest comportament.

El mode de seguretat de servidor (*server mode*) actua de "*bypass*" en mode usuari. És a dir, la petició que arriba del client per a accedir als recursos compartits és enviada al servidor principal del domini. Si el servidor respon amb una acceptació de la petició, el servidor Samba accepta la connexió del client. Si no, la denega.

Per fer servir aquest mode de seguretat hem d'afegir en el fitxer `smb.conf` els paràmetres següents:

```
encrypt password = yes
security = server
password server =
"Netbios_name_of_Domain_Controller(DCO)"
```

Configuració bàsica del Samba

Per a fer una configuració bàsica del servidor Samba, n'hi ha prou de posar les línies següents en el fitxer `smb.conf`:

```
[Global]
Workgroup = [real group]
Netbios name = [real name]

[Temp] Path = /tmp

[myhome] Path= /home/jordi
Comment = my home
```

Hi ha molts més paràmetres que veurem tot seguit i que fan referència a la seguretat, al tipus d'autenticació, etc.

Al llarg d'aquesta secció utilitzarem diverses versions d'aquest fitxer d'exemple, en les quals `[real group]` és el nom del grup de treball en què situarem aquesta màquina i `[real name]` és el nom NetBIOS del servidor. El camp `Path` indica l'emplaçament del directori que volem compartir, mentre que el camp `comment` és una explicació breu del contingut del recurs compartit.

És important que, una vegada modificat el fitxer `smb.conf` i abans d'iniciar el servei, executem l'ordre següent:

```
root# testparam /etc/samba/smb.conf
```

Aquesta ordre fa una "revisió mèdica" del fitxer `smb.conf` i ens dóna missatges d'avís si troba errors en la sintaxi o paràmetres desconeguts. Si l'ordre `testparam` s'executa de manera correcta indica que el fitxer de configuració és correcte.

Configuració del Samba com a controlador de domini principal

Per configurar el Samba com a PDC, el primer que hem de fer és modificar el fitxer `/etc/samba/smb.conf`. Aquest fitxer ha de tenir un aspecte semblant a aquest:

```
[global]
Netbios name = [real name]
Workgroup = [real group]
Domain logons = yes
Domain master = yes
Security = user

Passdb backend = tdbsam
Password server = *
Os level = 33
Preferred master = yes
Local master= yes
Logon path = \\%N\profiles\%o
Logon drive = H:
Logon home = \\homeserver\%o\winprofile
Logon script = logon.cmd<

[netlogon]
Path = /var/lib/samba/netlogon
Read only = yes
Write list = ntadmin

[profiles]
Path = /var/lib/samba/profiles
```

```
Read only = no
Create mask = 0600
Directory mask = 0700
```

Les cinc primeres línies d'aquest exemple són imprescindibles per a tenir el Samba com a PDC (*primary domain controller*). Els recursos compartits de `net logon` i `profiles` són necessaris en un entorn amb sistemes mixtos amb el Microsoft Windows. En el primer és on hi ha els *scripts* d'entrada al sistema (*logon*) i en el segon és on s'emmagatzemen els perfils d'usuari que s'envien a les màquines Windows perquè cada usuari tingui un perfil personalitzat. Com que cada usuari té el seu perfil propi i l'ha de poder modificar, el recurs compartit `profiles` ha de ser configurat amb permisos d'escriptura per als usuaris. És a dir, `Read only = no`, d'altra manera quedarien fixats i no podrien canviar res, ni tan sols els accessos directes de l'escriptori.

Hem de tenir present que, quan es configura el servidor Samba com a PDC, per motius de seguretat, hem de fer els passos següents perquè les màquines que fan servir Microsoft Windows puguin accedir al domini:

1) En el servidor Samba hem d'executar:

```
root# useradd -g machines -d /dev/null -c machine_name -s /bin/false machine_name$
root# passwd -l machine_name$
root# smbpasswd -a -m machine_name
```

2) Hem de tenir molt present que `machines` és un grup del sistema. Per tant, hi ha d'haver un grup a `/etc/group` anomenat `machines`, per tant, si fa falta s'haurà de crear amb anterioritat. També hem de saber que `machine_name` és el nom de la nostra màquina; hem de substituir el `machine_name` pel nom corresponent, respectant el signe $, ja que és important no treure aquest signe de l'ordre.

3) Hem de configurar tots els sistemes Microsoft Windows com a *domain members*. Si ens demana un nom d'usuari i una contrasenya per unir la màquina al domini, hem de fer servir el compte de *root* (o un compte amb privilegis) del servidor Samba; per fer-ho, abans s'ha d'haver inclòs l'usuari *root* en el fitxer `/etc/samba/smbpasswd` mitjançant l'ordre següent:

```
root# smbpasswd -a root
```

(posar la contrasenya de *root* per al Samba)

Depenent del mètode d'autenticació (LDAP, PAM, Unix, etc.), els usuaris han de tenir un compte creat en el servidor del Samba. És a dir, si fem servir el mètode d'autenticació Unix, els usuaris han de tenir un compte creat en el servidor i han d'estar donats d'alta en el fitxer `/etc/samba/smbpasswd`. Si fem servir LDAP o PAM, no cal crear-ne cap en el servidor.

Documentació addicional

En el cas de tenir màquines molt antigues que tinguin instal·lat Windows XP Home Edition o Windows 9x/ME s'ha de consultar la documentació següent:

http://us1.samba.org/samba/docs/man/samba-howto-collection/samba-pdc.html#id2518228

1.2. Servidor de fitxers i impressió en el Windows Server 2012

La instal·lació d'un servidor d'arxius en el sistema permet posar a disposició de tots els equips de la xarxa un conjunt de fitxers comuns ubicats en aquest equip. Els servidors d'arxius del Windows Server 2012 tenen, entre d'altres, les característiques següents:

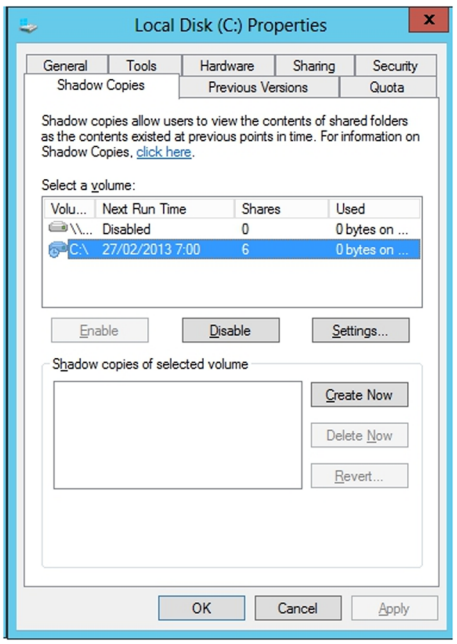
1) Arxius sense connexió: permet crear una còpia local dels arxius que hi ha en una carpeta de xarxa per utilitzar-los mentre s'està desconnectat. Els arxius s'actualitzen automàticament en tornar a establir la connexió.

2) Sistema d'arxius distribuït o *distributed file system* (DFS): permet que el conjunt d'arxius de xarxa estigui distribuït en diversos servidors d'arxius formant una estructura d'arbre, cosa que facilita molt la disponibilitat i l'administració dels fitxers a la xarxa.

3) Sistema d'arxius NTFS: admet xifrar arxius, agregar espai de disc a un volum NTFS sense reiniciar-lo i seguir els vincles distribuïts i les quotes de disc per usuari, que permeten controlar i limitar l'espai de disc de cada usuari i per tant també l'espai global que ocupen tots en el disc dur.

4) Millores en els permisos: se simplifica l'establiment i l'administració de permisos.

5) Servei d'instantànies de volum (VSS): crea una còpia exacta dels arxius guardats en carpetes de la xarxa compartides i, fins i tot, d'arxius que hi ha oberts. Les aplicacions poden continuar escrivint dades en el volum del disc durant el procés d'implementació de *shadow copies*. VSS no elimina la necessitat de fer còpies de seguretat completes i regulars. La figura següent mostra com es configura aquesta opció.



El servidor d'impressió del Server 2012 gestiona l'accés dels usuaris a les impressores connectades a la xarxa. El servidor d'impressió del Windows Server 2012 té, entre d'altres, les característiques següents:

- Augment de les impressores admeses: proporciona compatibilitat amb la majoria de les impressores actuals i relativament recents, cosa que permet utilitzar les capacitats d'impressió de diferents sistemes operatius.

- Grups d'impressores: permet configurar un servidor d'impressió per a compartir impressores a la xarxa.

- Protocol d'impressió a Internet o *Internet printing protocol* (IPP): permet als usuaris administrar impressores mitjançant un explorador web, imprimir mitjançant una pàgina web i veure la informació dels treballs d'impressió en format de llenguatge d'etiquetatge d'hipertext o *hypertext markup language* (HTML). També els permet connectar amb impressores mitjançant un explorador web, cosa que simplifica el procés d'establir connexions d'impressores.

En el cas que es vulgui compartir arxius i impressores amb clients molt antics com Windows 9x, cal instal·lar el protocol NetBEUI, i per a clients Windows XP, el protocol IPX/SPX/NetBIOS, d'aquesta manera el servidor es podrà "entendre" amb els clients més antics.

1.2.1. Configuració d'un servidor de fitxers

Creació de carpetes compartides

Les carpetes compartides constitueixen el mecanisme més fàcil de distribuir arxius en una xarxa local. Les carpetes o directoris compartits permeten posar a la disposició d'un usuari o de diversos usuaris de la xarxa local el contingut d'una carpeta situada en un altre equip de la mateixa xarxa local; per exemple, un servidor de fitxers.

Per configurar una carpeta com a carpeta compartida, obrim la finestra de propietats de la carpeta (fent clic amb el botó secundari del ratolí sobre la icona de la carpeta i seleccionant l'opció "Propietats"). En la pestanya "Compartir", seleccionem l'opció "Compartir aquesta carpeta" i posem un nom al recurs compartit: no cal que sigui el mateix que el de la carpeta real, però ha de ser únic en tot l'equip (no hi ha d'haver dos recursos compartits amb el mateix nom).

Si al final del nom del recurs compartit hi posem el signe $, aquest recurs compartit serà ocult en les cerques de carpetes o recursos compartits des d'un altre sistema. Això, si bé sembla que pot donar una mica de seguretat en el moment de compartir carpetes, no és del tot cert: per a usuaris "normals" és cert, ja que no veuran els recursos compartits, però per a eines de cerca especialitzades no ho serà, ja que són capaces de llistar tots els recursos compartits, tant si estan ocults amb el signe $ del final com si no.

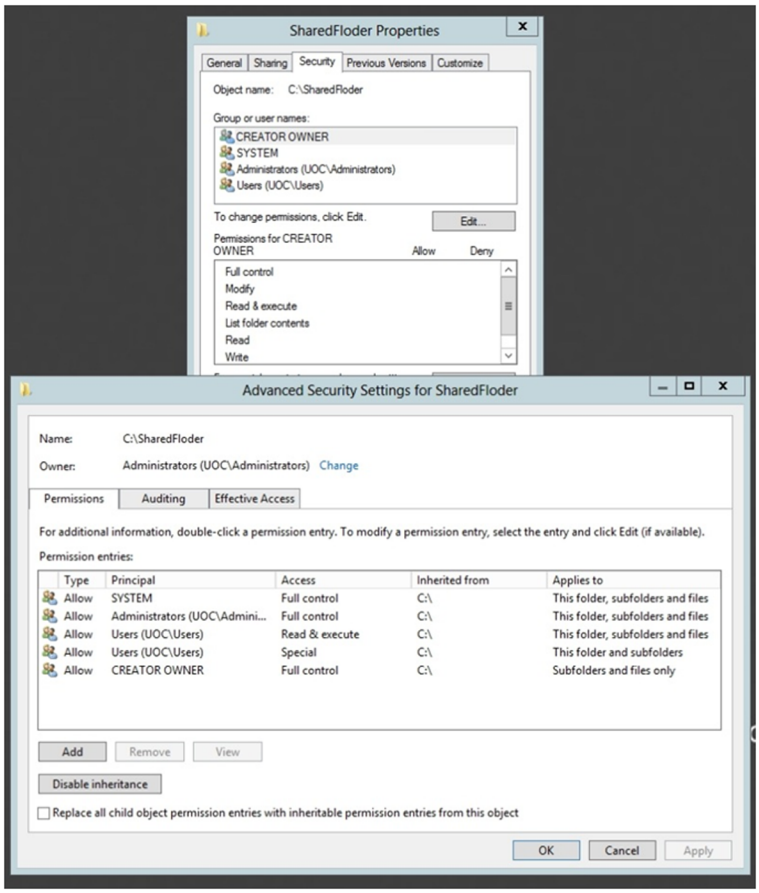
Permisos d'accés a carpetes compartides

A més d'especificar el nom i un comentari per a la carpeta compartida, podem limitar l'accés a un nombre màxim d'usuaris i fins i tot podem especificar quins usuaris en concret hi tenen accés mitjançant l'opció "Permisos". Dins les opcions de la carpeta compartida podrem incloure grups d'usuaris creats en el directori actiu, o directament certs usuaris, perquè tinguin drets de només lectura o també d'escriptura.

L'opció per defecte és compartir la carpeta amb tots els usuaris. Encara que sembli contradictori, amb aquesta opció n'hi ha prou, i normalment és la volguda per a no tenir problemes posteriorment, tot i que es pot afegir i restringir l'accés mitjançant els botons "Agregar" o "Treure". Per a no tenir problemes a l'hora de compartir dades, és recomanable assignar permisos per a NTFS (en la pestanya "Seguretat"), ja que sempre s'apliquen en primer lloc els permisos del recurs compartit i després els permisos NTFS. Per tant, el millor en molts casos –no sempre– és tenir accés a tothom, o a gairebé tothom, i després restringir amb més opcions els usuaris amb els permisos de la carpeta, i no del recurs compartit. Podríem tenir problemes i que un usuari tingués permisos per a NTFS, però no per a un recurs compartit. En aquest cas, l'usuari no pot accedir a les dades del recurs compartit. Per això, la quantitat de permisos NTFS és

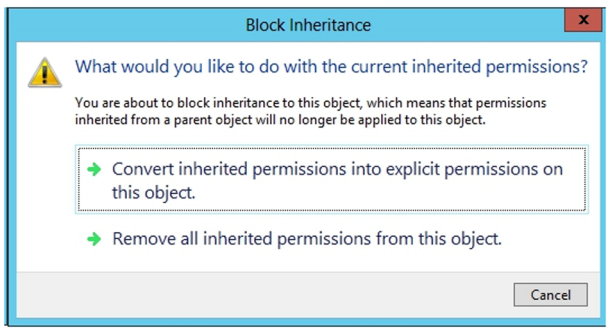
molt més gran (figura següent). Cal tenir en compte que, perquè ens surti la pestanya "Seguretat", la partició del disc ha d'estar formatada en NTFS. El sistema de fitxers FAT o FAT32 no proporciona control d'accés sobre els fitxers.

Permisos de la carpeta compartida



Els permisos d'accés a una carpeta compartida es poden assignar a usuaris individuals o a grups. Quan es canvien els permisos d'una carpeta principal, aquests permisos es propaguen a la resta de carpetes i arxius que conté la carpeta inicial. Per a evitar l'herència de permisos en una carpeta determinada s'ha de clicar el botó "Deshabilitar l'herència". Quan ho fem, surt un diàleg amb les opcions que mostrem en la figura següent.

Permisos heretats



- Copiar. Es copien els permisos heretats prèviament de la carpeta principal i es denega l'herència de permisos posteriors.

- Treure. Treu els permisos heretats prèviament de la carpeta principal i només manté els que s'han assignat explícitament a la carpeta.

Carpetes compartides sense connexió

L'opció "Memòria cau" de la finestra de propietats de la carpeta compartida permet activar o desactivar l'opció d'emmagatzemar en una memòria cau o *cache* local dels ordinadors client els documents per a utilitzar-los quan no es disposa de connexió amb el servidor; per exemple, en el cas de fer servir un portàtil, que en arribar a l'organització es connecta a la xarxa local i així es poden sincronitzar tots els documents que han estat modificats.

L'encarregat de sincronitzar els arxius de la xarxa local amb els del client és l'administrador de sincronització, que fa que la versió més recent de cada arxiu estigui disponible sempre, tant en el servidor com en el PC client. Es pot configurar l'administrador de sincronització perquè sincronitzi els arxius en determinats moments (en engegar o parar el sistema, a una hora concreta, quan el sistema és inactiu, etc.). Per sincronitzar manualment els arxius, seleccionem l'opció "Sincronitzar" del menú "Eines" en una finestra de l'explorador de Windows.

En la finestra de sincronització seleccionem els elements que es volem sincronitzar i prement el botó "Sincronitzar" començarà l'acció. El botó "Configurar" obre la finestra de configuració de l'administrador de sincronització, en la qual podem especificar en quin moment s'activarà l'administrador automàticament.

Quan dos usuaris fan canvis sobre el mateix fitxer, el segon usuari que intenta sincronitzar ha de decidir quina versió de l'arxiu vol mantenir o si se n'han de mantenir les dues versions.

Si bé és útil poder sincronitzar els fitxers per poder treballar des de fora de l'oficina, aquesta tècnica pot comportar en alguns casos problemes a l'hora de tenir la darrera versió del fitxer modificat, sobretot en els casos en què més d'una persona pot editar el fitxer.

En el cas de necessitar modificar documents que es guarden en el servidor i tenir molt més controlat l'accés i les versions, hi ha altres solucions, com tenir configurada una VPN per a accedir directament al fitxer, o un sistema de control de versions dels fitxers instal·lat en el servidor.

Accés a carpetes compartides

Per a accedir a les carpetes compartides creades en el servidor, s'ha d'obrir una finestra de l'explorador de Windows i escriure el nom del servidor en la barra d'adreces, seguit del nom de la carpeta compartida:

```
\\server_name\folder_name
```

Si l'accés al recurs compartit i la carpeta s'han configurat de manera que només certs usuaris hi tinguin accés, es demana l'usuari i la contrasenya corresponents quan s'hi vol accedir. Però si es tracta d'un recurs general, al qual tots els usuaris han de tenir accés, no caldrà restringir-ne l'accés ni demanar les credencials en entrar.

Si es vol que no surti una carpeta en fer la llista dels recursos compartits d'un servidor, `\\server_name`, cal compartir-la com a `folder_name$`. En aquest cas, per accedir-hi hem de posar `\\server_name\folder_name$`, però això té un desavantatge: l'efecte d'ocultar la carpeta es produeix per als sistemes operatius Windows client, però des d'un client amb GNU/Linux, amb el Samba configurat per a poder accedir a recursos compartits en un entorn Windows, encara que tingui el signe $ al final del nom, es continua mostrant i, per tant, no es produeix l'efecte buscat. Però encara que es mostri, cal disposar de permisos NTFS per a accedir-hi.

S'ha de tenir en compte que, en alguns casos, sempre es comparteix la unitat C$, de manera que s'oculta, però continua estant compartida.

Xifratge d'arxius i carpetes

Per a augmentar la protecció d'arxius i carpetes personals, es poden xifrar. Per fer-ho, obrim la finestra de propietats de la carpeta o arxiu que volem xifrar, premem el botó "Opcions avançades" que hi ha en la pestanya "General" i seleccionem l'opció "Xifrar contingut per a protegir dades". Si la carpeta xifrada conté arxius, hem de decidir si els arxius i subcarpetes que conté se xifren també recursivament. Els fitxers que es graven després en aquesta carpeta se xifren automàticament.

Per desxifrar un fitxer o una carpeta, desseleccionem l'opció "Xifrar contingut per a protegir dades" que s'ha activat en el pas anterior. Amb això es garanteix que els usuaris no puguin accedir a aquesta informació, que pot estar ubicada en carpetes compartides per més d'un usuari, però s'ha de tenir en compte que l'administrador del sistema té prou privilegis per a desxifrar tots els fitxers que han xifrat altres usuaris.

En moure un arxiu o una carpeta xifrats prèviament a una partició FAT, els arxius es desxifren automàticament, ja que el sistema de fitxers FAT no permet el xifratge.

La primera vegada que se xifra un arxiu o carpeta es crea un certificat autosignat que permet xifrar i desxifrar arxius; és important conservar aquest certificat, ja que si canviem de màquina o reinstal·lem el sistema es tornarà a generar un altre certificat diferent i no podrem desxifrar arxius xifrats amb el certificat anterior. Per tant, és molt important poder guardar en un lloc segur aquest certificat perquè es podria donar el cas de no poder accedir a la informació si no es té, malgrat que els fitxers estiguin xifrats.

Si es vol compartir un arxiu xifrat amb altres usuaris, cal distribuir també la clau pública del certificat perquè els destinataris el puguin desxifrar; en cas contrari, tot i poder tenir accés al fitxer, no el podran obrir.

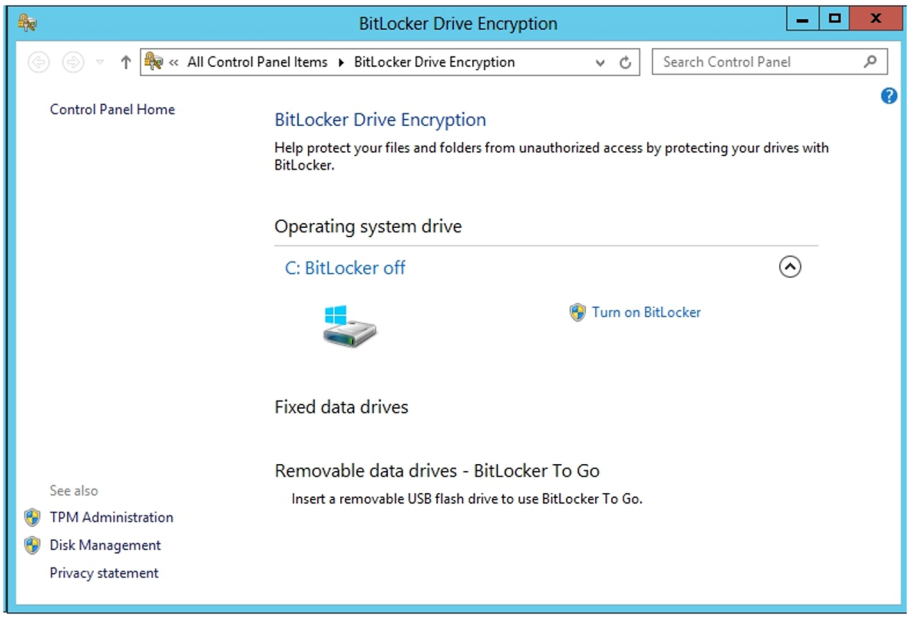
També es disposa de la nova eina Bitlocker, que permet xifrar i protegir d'una manera molt fàcil tot el sistema d'arxius o algun fitxer o carpeta.

En la darrera versió, incorporada al Windows Server 2012, se n'ha facilitat molt més l'ús per a grans empreses: per exemple, els administradors de sistemes no estan obligats a canviar el PIN d'accés de cada usuari quan no el recorden o poden desxifrar volums remotament a través de la xarxa, cosa que facilita la instal·lació remota de programari en equips que han d'estar xifrats.

Mitjançant les directives de grup l'administrador podrà decidir si es permet a l'usuari de l'ordinador xifrar alguns fitxers o carpetes, o si se xifra tot el disc, o només la part del disc que realment s'utilitza. Aquesta configuració es troba en la directiva: Configuració de l'equip / Directives / Plantilles administratives / Components de Windows / Xifratge d'unitat Bitlocker.

Instal·larem aquesta eina de xifratge, si no ho hem fet ja en algun dels passos anteriors, des de l'administrador del servidor, dins les característiques del servidor que s'ha d'instal·lar. La figura següent mostra la pantalla de configuració, on es pot activar el xifratge del disc o dels arxius en el cas que s'hagi configurat per poder xifrar fitxers individualment.

Configuració Bitlocker



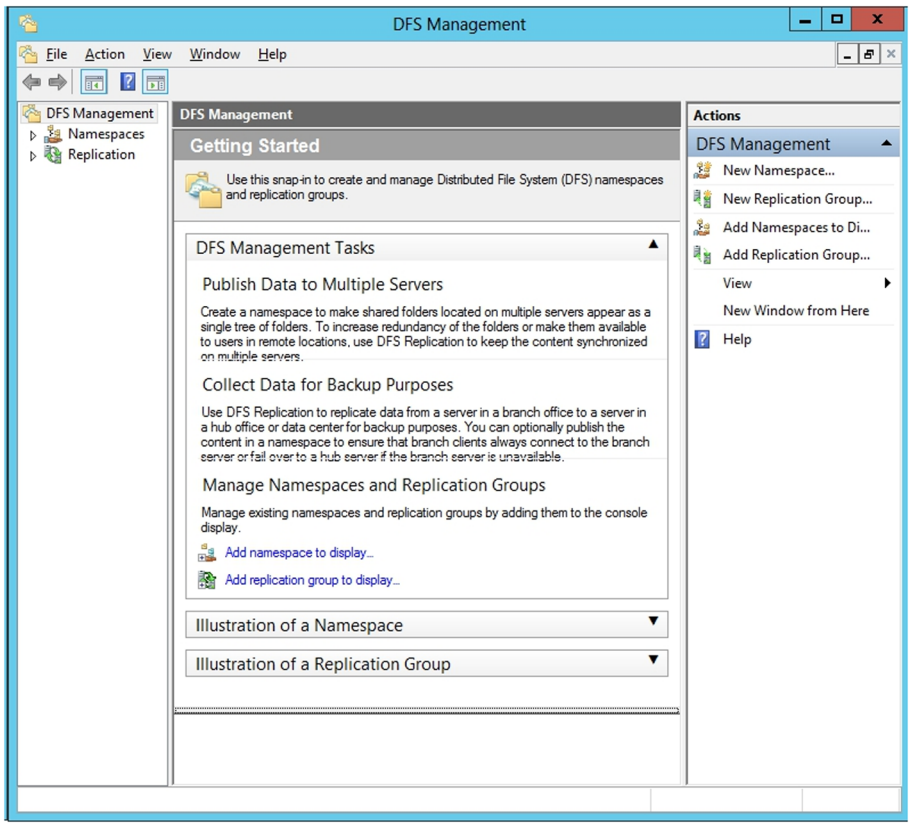
Sistema d'arxius distribuït (DFS)

El sistema d'arxius distribuït, que es troba dins el rol d'administrador de fitxers i que s'ha d'instal·lar a part d'aquest servidor de fitxers, permet definir una estructura virtual de directoris compartits entre diferents servidors de manera unificada en un punt únic de la xarxa local. Aquesta eina proporciona transparència sobre la localització física real dels arxius i els recursos compartits respectius en què se situen. D'aquesta manera, un usuari no ha de recordar l'emplaçament real o el servidor on hi ha cada recurs compartit ni el nom d'aquest recurs, sinó que en té prou de saber la ruta a l'arrel del DFS. A més, permet aprofitar parts de discos de servidors i equips que no es fan servir i fa que, si s'ha de moure informació d'un disc a un altre o a un altre servidor, sigui completament transparent i molt més segur des del punt de vista de l'usuari, que seguirà tenint la mateixa ruta per a arribar als fitxers.

Una altra característica del DFS que també es pot instal·lar és l'aplicació que farà que les dades d'un disc estiguin replicades en un altre disc d'un servidor ubicat en la mateixa xarxa local o fins i tot a través de la xarxa Internet. Així, podem tenir les dades replicades de manera automàtica, cosa que ens dóna molta més seguretat a l'hora de guardar els documents i poder-hi accedir ràpidament. Per exemple, en el cas de tenir dos seus que han d'accedir a fitxers comuns, una possible solució seria replicar aquests fitxers per a poder treballar molt més ràpidament.

El DFS defineix una estructura lògica de directoris en forma d'arbre que comença en un directori principal compartit, anomenat *arrel de fitxers*, a partir del qual s'organitzen els enllaços a les carpetes compartides anomenades *vincles DFS*. La figura següent mostra la configuració inicial del DFS.

Per a crear una nova arrel de fitxers, seleccionem l'opció "Nova arrel DFS" del menú de les accions. L'assistent ens demana que seleccionem el tipus d'arrel DFS que volem crear:



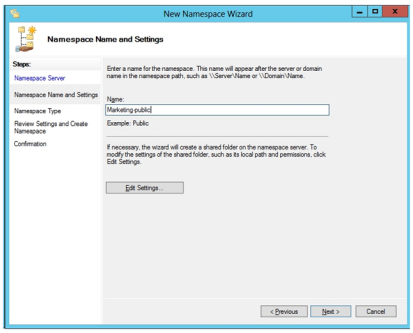
1) Arrel de domini: arrel DFS basada en el directori actiu; la informació del DFS s'emmagatzema amb la informació del domini. Permet fer duplicació automàtica i molts nivells de vincles.

2) Arrel independent: arrel independent de l'*Active Directory*. La duplicació s'ha de fer manualment i només permet un nivell de vincles.

La duplicació automàtica permet copiar l'estructura del DFS en una altra arrel en un altre servidor per a augmentar la disponibilitat i l'equilibri de càrrega. Una vegada creada l'arrel, hi podem afegir vincles. Un vincle és un enllaç virtual a un recurs compartit que hi ha en un servidor de la xarxa. Podem aconseguir que un vincle sigui tolerant a fallades i amb distribució automàtica de càrrega enllaçant un mateix vincle amb altres recursos compartits. El DFS s'encarrega de mantenir la replicació entre recursos compartits i d'assegurar que, en tot moment, el contingut de tots els recursos compartits que conté un vincle és idèntic.

L'assistent de configuració del DFS demana el nom del servidor (i el domini si triem una arrel de domini) on s'ha de situar el DFS, i a continuació el directori compartit en què volem situar l'arrel DFS. Finalment, hem de posar un nom únic a l'arrel DFS en el domini o a la xarxa, tal com es pot veure en la figura següent.

Nom de l'arrel de DFS



Per afegir una nova carpeta compartida a l'arrel DFS, seleccionem l'opció "Nou vincle..." del menú contextual de l'element arrel en la llista de l'esquerra de la pantalla. En la pantalla de creació del nou vincle, ens demanen el nom i comentari corresponents, i també la localització física a la qual serà dirigit l'usuari en seleccionar aquest vincle.

Es poden crear rèpliques addicionals d'una arrel DFS o d'un vincle DFS en concret per a augmentar la disponibilitat. La replicació d'una arrel DFS només es pot fer si es tracta d'una arrel de domini, i un mateix servidor només en pot emmagatzemar una.

Per crear una rèplica d'arrel DFS, premem el botó secundari del ratolí sobre l'element arrel, i seleccionem l'opció "Nova destinació d'arrel..." del menú contextual. L'assistent demanarà el servidor on es vol crear la rèplica, i la carpeta compartida a partir de la qual s'emmagatzemaran els vincles DFS.

Una vegada creada la rèplica, definim la política de replicació automàtica seleccionant l'opció "Directiva de rèplica" del menú contextual de l'element arrel. En la finestra de configuració veiem una llista de les rèpliques de l'arrel. Podem seleccionar una arrel i fer clic sobre "Habilitar" o "Deshabilitar" per incloure l'arrel DFS en el procés de replicació automàtica. Una de les arrels ha de marcar el contingut que han de tenir la resta d'arrels. D'aquesta arrel se'n diu *mestra*. Per a canviar l'arrel mestra, seleccionem la nova arrel i premem "Establir mestra".

D'altra banda, podem crear rèpliques d'un vincle DFS seleccionant l'opció "Nou recurs..." del menú contextual del vincle en la llista de l'esquerra i introduint les dades del nou recurs compartit que ha de contenir la rèplica del primer. L'existència d'aquesta rèplica té sentit només si és en un altre servidor

de la xarxa, ja que, si cau un dels dos servidors, sempre tindrem una de les rèpliques a punt. Podem definir quina és la còpia mestra (*master*) o la que replicarà sobre els altres vincles.

Si l'arrel DFS és independent, hem de copiar els arxius de manera manual, i també les possibles actualitzacions en el futur. En canvi, si l'arrel es basa en el directori actiu, seleccionem l'opció de rèplica automàtica, de manera que el servei de replicació d'arxius s'encarregui de copiar i actualitzar els arxius.

Si se selecciona la replicació automàtica, hem d'especificar quines rèpliques s'actualitzen automàticament i quina fa de mestra, com en la replicació de l'arrel DFS.

Per accedir a un recurs d'una arrel DFS des d'un client, obrim una finestra de l'explorador del Windows i escrivim el nom del servidor en la barra d'adreces, seguit del nom de l'arrel DFS. Exactament igual que si fos una carpeta compartida pel mateix servidor, per a l'usuari és completament transparent. A partir d'aquí podem navegar per tota l'estructura lògica DFS, sense saber a quin servidor i a quina carpeta concreta estem accedint. Els permisos d'accés a les carpetes reals físiques determinen si un usuari té accés a un vincle DFS o no. Si és una arrel DFS de domini, n'hi ha prou que accedim al nom de domini:

```
\\domain_name\namespace
```

1.2.2. Configuració del servidor d'impressió

Igual que es poden compartir carpetes i fitxers en una xarxa local, també es poden compartir impressores, de manera que estiguin disponibles per a alguns usuaris de l'empresa. La impressora s'afegeix a un servidor (servidor d'impressió) i a continuació es comparteix a la xarxa.

Instal·lar i compartir impressores

Per a instal·lar i administrar impressores d'un servidor d'impressió, utilitzem l'eina normal d'instal·lar les impressores, l'eina "Impressores i dispositius" del tauler de control. Per agregar una impressora, fem doble clic sobre la icona "Agregar impressora", i ens surt l'assistent corresponent. En la finestra "Impressora local o de xarxa" seleccionem si la impressora està connectada amb el servidor o és una impressora amb interfície de xarxa. Darrerament, el més usual és tenir les impressores connectades directament a la xarxa local.

Un cop instal·lada la impressora, s'ha de compartir perquè els altres ordinadors la puguin veure en el cas que estigui connectada directament al servidor per USB o amb port paral·lel. En el cas de tenir-la en xarxa, no fa falta compartir-la, perquè ja ho està pel fet d'estar connectada directament a la xarxa.

A més de compartir la impressora, hem d'assignar els permisos dels usuaris que la poden utilitzar des de la pestanya "Seguretat". No tothom podrà imprimir en totes les impressores que hi ha en una organització i, per tant, no caldrà que li surtin totes les impressores, únicament cal que es pugui instal·lar les que té més a prop, les que siguin del seu departament, o les que tinguin les característiques adients per a la feina que fa.

Una vegada configurada, els clients del domini poden connectar la impressora des de qualsevol explorador amb `\\server_name` i fent doble clic sobre la impressora compartida. Els programes controladors necessaris s'instal·len de manera automàtica.

Especificació de la carpeta de cua d'impressió

Quan els clients envien treballs d'impressió a una impressora, les dades que s'envien s'emmagatzemen temporalment (es posen a la cua) en la carpeta de cua d'impressió, que és per defecte en el directori `system32\spool\printers` del directori d'instal·lació del Windows Server 2012.

Com que els arxius temporals d'impressió i els arxius del sistema operatiu són en la mateixa unitat física, és possible que es redueixi el rendiment del Windows i el del servei d'impressió. Per això és aconsellable canviar l'emplaçament de la carpeta de cua d'impressió a una altra unitat que tingui un controlador de disc independent i prou espai lliure per a emmagatzemar tots els arxius temporals d'impressió.

Per a canviar la carpeta de cua d'impressió obrim l'eina "Impressores i dispositius" del tauler de control i seleccionem l'opció "Propietats del servidor" del menú "Arxiu". En la pestanya "Opcions avançades" especifiquem la unitat i el directori en els quals es guardaran els arxius temporals, i també més opcions relacionades amb els esdeveniments que s'han de registrar en el servidor i notificacions d'impressió.

Permisos d'accés a impressores

Els permisos d'accés a una impressora es configuren de la mateixa manera que els permisos d'accés a una carpeta. Per fer-ho, obrim l'eina "Dispositius i Impressores", premem el botó secundari del ratolí sobre la impressora i seleccionem l'opció "Propietats". En la pestanya "Seguretat" configurem els usuaris o grups que tenen permisos sobre la impressora i quins permisos tenen concretament.

Impressores compartides amb l'*Active Directory*

Quan s'instal·len i comparteixen impressores en una xarxa des d'un servidor Windows Server 2012, aquest les publica per defecte en l'*Active Directory*, cosa que facilita la localització de les impressores segons el tipus o emplaçament que tenen.

Una vegada oberta l'eina "Usuaris i equips d'*Active Directory*", fem clic amb el botó secundari del ratolí sobre l'element en el qual volem publicar la impressora i seleccionem l'opció "Nova" i, a continuació, "Impressora". Finalment, introduïm l'emplaçament de la impressora.

Aquesta eina és útil en entorns amb moltes impressores compartides i en emplaçaments distants. Permet localitzar les impressores compartides per equips del domini de manera ràpida utilitzant criteris de cerca, com ara impressores amb possibilitat de color, d'impressió a dues cares, etc.

Anàlisi de riscos i prevenció

Ja s'ha explicat com es configura un servidor per a oferir diferents serveis de manera segura. Aquest apartat introdueix el concepte del tallafoc o *firewall*. Un tallafoc és un dispositiu de xarxa que exerceix un paper molt important en la seguretat: és l'encarregat de deixar passar un paquet de dades de la xarxa cap a l'interior de xarxa o bloquejar-lo i, per tant, no deixar-lo passar ni cap a fora de la institució ni cap a dins si la procedència és externa. La configuració d'aquests equips ha de permetre que tots els serveis públics que ofereix l'empresa o institució siguin visibles des d'Internet. Per tant, com que els serveis oferts no poden ser protegits pel tallafoc, els ha de protegir el sistema.

El cas del servei de directori és diferent. Aquests serveis no estan pensats per a oferir-los per Internet, sinó per a facilitar la feina de compartir documents entre els treballadors de l'empresa i oferir un espai de disc als usuaris per a guardar els documents importants, tant els compartits com els personals. Per tant, no podem accedir a aquests serveis de manera directa des d'Internet. El tallafoc bloqueja –o hauria de bloquejar– tots els intents d'accedir-hi remotament. És a dir, no podem utilitzar el servidor de discos de l'oficina des de casa nostra.

Per accedir remotament a l'espai de disc de l'empresa s'han desenvolupat altres serveis que ens permeten fer aquesta connexió remota. Aquests serveis són, d'una banda, l'FTP (clarament insegur) i el seu homòleg que utilitza ports segurs SFTP, i d'altra banda, les VPN, que permeten un accés segur a l'espai de disc des de fora de l'empresa o institució. El cas del protocol FTP el veurem més endavant. El cas de les VPN l'hem tractat en el mòdul anterior.

Així, doncs, quins són els riscos d'un servidor de discos si no podem accedir des d'Internet a aquest servei?

Hem de tenir present que, sia per treballadors descontents, per errors humans a l'hora de tractar la informació o per competència deslleial, la majoria dels atacs seriosos que pateix una empresa tenen l'origen dins de l'empresa. Tot i que també rep molts intents d'atac de l'exterior, si la seguretat està mínimament configurada molt pocs podran arribar a obtenir dades de l'empresa o entitat.

En la majoria dels casos, un atac des de dins es produeix pels motius següents:

- Una configuració errònia en el servidor (l'usuari pot accedir a documents als quals en principi no hauria de poder accedir).

- Un treballador sap la contrasenya d'un altre usuari de l'empresa.

Aquests atacs són molt difícils de prevenir, ja que per funcionament de l'empresa hem de donar servei a tots els treballadors. Per tant, no es poden utilitzar tallafocs, ni restriccions de ports, ni claus d'autenticació si no estan ben configurats, ja que podríem tallar l'accés a la informació als usuaris legítims.

Per prevenir aquests tipus d'atacs ens hem d'assegurar dels punts següents:

- Tots els treballadors de l'empresa tenen els privilegis que han de tenir dins el directori actiu.

- Tots els usuaris donats d'alta en el servidor són els treballadors actuals de l'empresa. És a dir, quan un treballador deixa l'empresa, se n'elimina l'usuari i se'n treuen els privilegis d'accés a tots els recursos.

- La política de contrasenyes és prou restrictiva per a no permetre contrasenyes simples i obligar els usuaris a canviar-les sovint.

Encara que apliquem tots aquests punts preventius, no podem assegurar que un treballador no aconseguirà la contrasenya d'un altre treballador, o que utilitzarà un ordinador que no és el seu mentre el company ha abandonat l'ordinador temporalment.

2. Tallafoc

Un tallafoc és un dispositiu de xarxa que filtra el trànsit entre dues xarxes. És un element bàsic en qualsevol xarxa amb un mínim de seguretat. L'objectiu principal és analitzar els paquets que hi travessen i decidir, segons un conjunt de regles, si s'ha de descartar el paquet o s'ha d'acceptar que continuï cap a la destinació que té.

En una màquina Linux podem implementar un tallafoc local o bé un tallafoc que protegeix un conjunt de màquines d'una xarxa interna. Per fer-ho, necessitem l'eina *iptables*, que es comunica amb el nucli o *kernel* de la màquina per indicar-hi quina acció ha de fer amb els paquets mitjançant un conjunt de regles.

La versió de Debian que s'està fent servir ja té el suport integrat en el nucli per a *iptables* i, a més, ja té instal·lades les eines que s'executen en espai d'usuari *iptables*. En el cas que no estiguin instal·lades, n'hi ha prou d'executar l'ordre següent:

```
root# apt-get install iptables
```

L'eina *iptables* ofereix diferents funcionalitats, com traducció d'adreces IP internes (NAT) o priorització de paquets, entre d'altres. Ens centrarem únicament en el filtratge de paquets per a implementar un senzill tallafoc.

2.1. Conceptes bàsics

Abans de començar la part més pràctica d'aquesta eina, introduirem alguns conceptes que més endavant ajudaran a entendre la configuració d'un *script* (fitxer d'instruccions) amb *iptables*:

- Regla: defineix una sèrie d'atributs que ha de contenir el paquet de dades de la xarxa perquè s'hi apliqui l'acció que està associada a aquesta regla. Aquests atributs corresponen a valors de les capçaleres IP o TCP/UDP/ICMP del paquet que s'està tractant.

- Acció: hi ha una acció (*target*) per cada regla. L'acció indica com s'ha de procedir amb el paquet si el conjunt d'atributs definits en la regla coincideixen amb el paquet de què es tracta (*match*). Així, per exemple, es pot descartar un paquet o acceptar-lo i passar-lo a nivells superiors, entre d'altres.

- Taula: en *iptables* es defineixen tres taules, cadascuna amb un objectiu diferent:

  - Taula *Filter*. Per al filtratge de paquets. És la taula que es fa servir per a crear el tallafoc.

  - Taula NAT. Per a traduir adreces IP.

  - Taula *Mangle*. Per a modificar alguns camps del paquet de dades del protocol IP.

- Cadena: una cadena conté un conjunt de regles de filtratge que s'apliquen als paquets que travessen aquesta cadena. Cada cadena té un objectiu concret. La taula *Filter* conté les cadenes d'*input*, *output* i *forward*.

- Política: és el comportament per defecte que té una cadena si no hi ha cap regla que indiqui l'acció concreta que s'ha de fer sobre el paquet concret. Hi ha dues polítiques generals que, depenent del cas, es fan servir de manera contrària. La primera descarta tots els paquets excepte els que s'indiquen explícitament. La segona fa tot el contrari, és a dir, accepta tots els paquets per defecte i rebutja els que s'indiquen explícitament. En general, s'utilitza la primera política, perquè és la més segura per defecte.

2.2. Recorregut d'un paquet IP

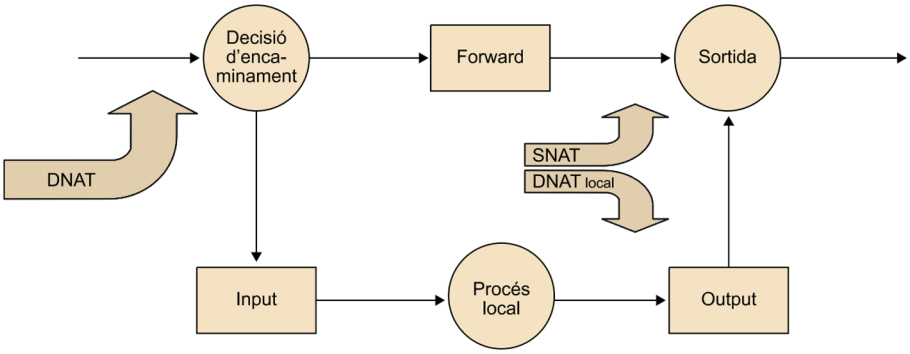
Quan un paquet de dades entra per la interfície de xarxa, passa per un conjunt d'estats. D'entrada es comprova l'adreça IP destinació del paquet. Si coincideix amb l'adreça IP de la màquina, indica que és un paquet destinat a la màquina pròpia, i per tant passa pel conjunt de regles de la cadena d'*input*. Si s'accepta el paquet, el procés que l'espera, és a dir, el servei o aplicació que està escoltant per rebre un determinat paquet, el rep i el processa.

Si, en canvi, l'adreça IP n'és una altra, el paquet es mou cap a la cadena de *forward*. Si la màquina té l'encaminament (*forwarding*) deshabilitat, el paquet és descartat automàticament, mentre que si el té habilitat, el conjunt de regles en aquesta cadena decideixen el futur del paquet i és encaminat o no cap a la destinació final que té.

Finalment, qualsevol aplicació que genera trànsit de xarxa envia els paquets directament a la cadena d'*output*. Si les regles accepten el paquet, és dirigit cap a la interfície de sortida.

La figura següent resumeix el recorregut que pot arribar a fer un paquet des que entra per la interfície de xarxa fins que es decideix quina acció s'hi fa:

Recorregut dels paquets IP



Les sigles *DNAT* i *SNAT* indiquen en el diagrama anterior quan es pot fer una traducció d'adreces IP (NAT), sia de l'adreça d'origen (SNAT) o de l'adreça de destinació (DNAT).

2.3. Passos per a la creació d'un tallafoc en GNU/Linux

En aquest apartat es mostra com s'ha de configurar un tallafoc local en una màquina, per exemple, el servidor que s'està configurant. Com que la màquina té l'encaminament deshabilitat, és a dir, no s'està fent de pont entre altres màquines, no cal configurar res dins de la cadena de *forward*. Per tant, tota la configuració se centra en les cadenes d'*input* i *output* de la taula *Filter*.

Com ja hem comentat, els paquets passen per un conjunt de regles, amb una sèrie de condicions que s'han de complir per a aplicar l'acció que està associada a la regla. Si al final s'han aplicat totes les regles sense que n'hi hagi cap que coincideixi amb el paquet que s'està tractant, s'aplica a la cadena la política per defecte, normalment tancar la resta de ports que no s'han obert amb una regla específica per al port en qüestió.

Web recomanat

Si voleu més informació sobre aquests aspectes, consulteu la pàgina web següent:
http://www.netfilter.org/documentation/index.html

En aquest exemple inicial, configurem un *script* sense cap regla explícita, i canviem la política per defecte en les cadenes d'*input* i *output* a *drop* (descartat). Fent això indiquem que, per defecte, es descartin tots els paquets que entren i surten de la màquina. Aquest *script* no és un tallafoc pròpiament, ja que descartem tot el trànsit i impedim, alhora, l'accés a Internet a la nostra pròpia màquina. Però en molts casos en què es necessita poder aïllar completament la màquina per un incident de seguretat és una bona opció a tenir en compte, ja que la màquina queda aïllada de la xarxa Ethernet.

```
root# cat script.sh
#!/bin/bash

/sbin/iptables -P INPUT DROP
/sbin/iptables -P OUTPUT DROP
/sbin/iptables -P FORWARD DROP
```

Mitjançant el paràmetre `-P` configurem la política per defecte de la cadena, que pot ser *accept* o *drop*. Cal destacar que en aplicar aquest *script* perdem connectivitat amb l'exterior, de manera que cal executar-lo des de la consola local de la màquina i mai remotament, ja que no es podrà tornar a recuperar la connexió. Per a executar l'*script*, n'hi ha prou de donar-hi permisos d'execució i executar-lo amb privilegis de *root*:

```
root# chmod +x script.sh
root# ./script.sh
```

Ara, afegim a l'*script* anterior les regles perquè la màquina pugui servir pàgines HTTP/HTTPS i a més permeti connexions entrants del protocol SSH.

```
root# cat script.sh
#!/bin/bash

/sbin/iptables -P INPUT DROP
/sbin/iptables -P OUTPUT DROP
/sbin/iptables -P FORWARD DROP

/sbin/iptables -A INPUT -p tcp --dport 80 -j ACCEPT
/sbin/iptables -A INPUT -p tcp --dport 443 -j ACCEPT
/sbin/iptables -A INPUT -p tcp --dport 22 -j ACCEPT
```

Amb el paràmetre `-A` (*append*) afegim una nova regla al final de la cadena. El paràmetre `-p` indica el tipus de protocol superior, que en aquest cas és `TCP`. El paràmetre `--dport` indica el port de destinació del paquet, i en aquest cas hi ha una regla per als ports d'HTTP, HTTPS i SSH que es volen configurar (80, 443 i 22, respectivament). Finalment, el paràmetre `-j` indica l'acció que s'ha de fer amb els paquets que arribin per cada un dels ports configurats, que pot ser *accept*, *drop*, *log*, etc.

D'aquesta manera tot el que no estigui configurat com a *accept* en les darreres regles es descartarà automàticament per a les darreres regles de l'*iptables*.

Per visualitzar el conjunt de regles que tenim configurades fins ara, executem la línia següent:

```
root# iptables -L -n
```

El paràmetre `-L` fa que l'ordre faci la llista del conjunt de regles que s'han aplicat i el paràmetre `-n` ens omet la resolució d'adreces IP a noms. Podem consultar altres paràmetres mitjançant l'ordre `man iptables`. El resultat d'aquesta ordre és el següent:

```
root# iptables -L -n
Chain INPUT (policy DROP)
```

```
target  prot  opt   source  destination
ACCEPT  tcp   --   0.0.0.0/0 0.0.0.0/0   tcp dpt:80
ACCEPT  tcp   --   0.0.0.0/0 0.0.0.0/0   tcp dpt:443
ACCEPT  tcp   --   0.0.0.0/0 0.0.0.0/0   tcp dpt:22
Chain FORWARD (policy DROP)
target  prot  opt   source   destination
Chain OUTPUT (policy DROP)
target  prot  opt   source   destination
```

Cal destacar que, com que no tenim cap regla sobre els paquets de sortida (*output*) de la màquina cap a la xarxa i com que tenim la política per defecte a *drop*, el trànsit de resposta de la nostra màquina queda descartat a la sortida. Per tant, si bé pot rebre paquets dels protocols que s'han obert, no pot enviar peticions ni res més pels mateixos ports perquè estan tancats. S'hi ha d'afegir una regla que permeti el trànsit de resposta. Per exemple una regla general:

```
/sbin/iptables -P OUTPUT ACCEPT
```

Amb aquesta regla canviem la política per defecte a *accept*, cosa que permet tot el trànsit de sortida de la nostra màquina a l'exterior, i per tant això inclou les respostes a les peticions d'entrada en els ports 80, 443 i 22. Però no hi ha cap control de ports de sortida, és a dir, és com si no existís el tallafoc, i per tant l'alternativa millor és ser una mica més restrictius i permetre sortir únicament el trànsit que hem deixat entrar:

```
/sbin/iptables -A OUTPUT -p tcp --sport 80 -j ACCEPT
/sbin/iptables -A OUTPUT -p tcp --sport 22 -j ACCEPT
/sbin/iptables -A OUTPUT -p tcp --sport 25 -j ACCEPT
```

Finalment, l'opció més còmoda i segura consisteix a deixar sortir únicament el que hem deixat entrar, és a dir, el trànsit que pertany a connexions que ja hi ha establertes:

```
/sbin/iptables -A OUTPUT -m state --state ESTABLISHED -j ACCEPT
```

Cal dir també que no s'ha especificat cap interfície de xarxa en cap de les regles, de manera que s'apliquen les regles amb independència de la interfície de xarxa per on entren o surten els paquets.

"Només hi ha dues regles per a escriure: tenir res a dir i dir-ho." Oscar Wilde.

Si visualitzem les regles, tenim la sortida següent:

```
root# iptables -L -n
Chain INPUT (policy DROP)
target  prot  opt   source   destination
ACCEPT  tcp   --   0.0.0.0/0 0.0.0.0/0   tcp dpt:80
```

```
ACCEPT  tcp   --   0.0.0.0/0 0.0.0.0/0   tcp dpt:443
ACCEPT  tcp   --   0.0.0.0/0 0.0.0.0/0   tcp dpt:22
Chain FORWARD (policy DROP)
target  prot  opt   source   destination
Chain OUTPUT (policy DROP)
target  prot  opt   source
ACCEPT  all   --   0.0.0.0/0   0.0.0.0/0 state ESTABLISHED
```

Les execucions successives de l'*script* aniran afegint noves regles al final de les cadenes d'*input* i *output*, de manera que abans de tornar a executar l'*script* cal esborrar les regles anteriors. Per a esborrar les regles s'utilitza el paràmetre `-F` (*flush*). Aquest pas és important, ja que si s'està provant amb diferents configuracions i no s'esborren les regles anteriors pot ser que el sistema no respongui correctament al que volem fer amb una regla en concret.

L'*script* queda de la manera següent:

```
root# cat script.sh
#!/bin/bash

/sbin/iptables -F
/sbin/iptables -P INPUT DROP
/sbin/iptables -P OUTPUT DROP
/sbin/iptables -P FORWARD DROP

/sbin/iptables -A INPUT -p tcp --dport 80 -j ACCEPT
/sbin/iptables -A INPUT -p tcp --dport 443 -j ACCEPT
/sbin/iptables -A INPUT -p tcp --dport 22 -j ACCEPT
/sbin/iptables -A OUTPUT -m state --state ESTABLISHED -j ACCEPT
```

El fet de tenir la política per defecte a *drop* sobre el trànsit de sortida implica que impedim que la màquina tingui connectivitat a l'exterior. Això pot comportar un problema, ja que, per exemple, no podrem rebre actualitzacions de programari, però és un avantatge, ja que, en cas d'un accés no autoritzat a la màquina per a robar la contrasenya, evitem la baixada de programari maliciós.

2.4. Altres ordres d'*iptables*

Amb el senzill exemple anterior hem vist algunes de les ordres d'*iptables* especificades amb els paràmetres `-F`, `-A`, `-L`, etc.

Un altre paràmetre molt útil que acompanya el paràmetre `-L` és `-v`, que permet veure els comptadors de quants paquets i bytes han coincidit en cadascuna de les regles. Finalment, el paràmetre `-Z` (zero) posa els comptadors a zero.

```
root# iptables -L -n -V
Chain INPUT (policy DROP 6 packets, 1353 bytes)
```

```
 pkts bytes target  prot opt in out source  destination
  1  60 ACCEPT  tcp  --  *  * 0.0.0.0/0 0.0.0.0/0 tcp dpt:80
  0   0 ACCEPT  tcp  --  *  * 0.0.0.0/0 0.0.0.0/0 tcp dpt:443
  27 3066 ACCEPT  tcp  --  *  * 0.0.0.0/0 0.0.0.0/0 tcp dpt:22

Chain FORWARD (policy DROP 0 packets, 0 bytes)
 pkts bytes target  prot opt in out source  destination

Chain OUTPUT (policy DROP 23 packets, 7028 bytes)
 pkts bytes target  prot opt in out source  destination
  25 4670 ACCEPT  all  --  *  * 0.0.0.0/0 0.0.0.0/0 state ESTABLISHED

root# iptables -Z
```

En la inserció de regles, l'únic paràmetre que s'ha vist fins ara ha estat el `-A` (*append*), però també es pot fer servir el paràmetre `-I` per a inserir una regla nova en una posició concreta de la cadena, o el paràmetre `-D` per a eliminar una regla de la cadena. A continuació veurem una sèrie d'exemples que utilitzen les ordres anteriors. D'entrada tenim el següent conjunt de regles en la cadena d'*input*:

```
root# iptables -L -n
Chain INPUT (policy DROP)
Target  prot opt source  destination
ACCEPT  tcp  --  0.0.0.0/0 0.0.0.0/0 tcp dpt:80
ACCEPT  tcp  --  0.0.0.0/0 0.0.0.0/0 tcp dpt:443
ACCEPT  tcp  --  0.0.0.0/0 0.0.0.0/0 tcp dpt:22
```

Tot seguit eliminem la regla referent al port 80 i visualitzem el resultat obtingut:

```
root# /sbin/iptables -D input 1

root# iptables -L -n
Chain INPUT (policy DROP)
Target  prot opt source  destination
ACCEPT  tcp  --  0.0.0.0/0 0.0.0.0/0 tcp dpt:443
ACCEPT  tcp  --  0.0.0.0/0 0.0.0.0/0 tcp dpt:22
```

Finalment, en comptes d'afegir la regla referent al port 80 al final de la cadena, la inserim en segona posició:

```
root# iptables -L -n
Chain INPUT (policy DROP)
Target prot opt source destination
ACCEPT tcp -- 0.0.0.0/0 0.0.0.0/0 tcp dpt:443
ACCEPT tcp -- 0.0.0.0/0 0.0.0.0/0 tcp dpt:80
```

```
ACCEPT tcp -- 0.0.0.0/0 0.0.0.0/0 tcp dpt:22
```

Una altra opció molt útil és la redirecció de ports, que es pot fer servir per a amagar de manera senzilla els ports estàndard de comunicació, tot i que no seria del tot segura però sí senzilla, i també en el cas de tenir un servidor amb màquines virtuals per a poder tenir connexió d'entrada des de la xarxa cap a les màquines sense disposar d'IP externa de les màquines virtuals.

En aquest cas el que s'ha de fer és crear una regla en el PREROUTING, és a dir, que es processarà abans que la resta, amb el canvi de ports i adreces IP que són necessaris. Per exemple, si tenim una màquina virtual que té el port d'escolta del ssh configurat en el port 45022, i a més es vol que el port d'escolta de la màquina real per a respondre a aquest port sigui el 55022, es pot fer de la manera següent:

```
root# iptables -t nat -A PREROUTING -p tcp --dport 55022 -j DNAT --to-destination
192.168.122.23:45022
```

Així, la màquina virtual que té l'adreça IP 192.168.122.23 estarà escoltant pel port 45022 el que la màquina *host* està rebent pel port 55022. Això permet tenir diferents màquines que facin servir el mateix port, per exemple el port 22 o 80, i tenir el NAT en la màquina allotjadora (*host*) amb diferents ports i la mateixa IP (la de la màquina allotjadora).

Per veure les regles configurades per a fer NAT en la màquina allotjadora s'aplica l'ordre:

```
root# iptables -t nat -L
```

Les regles que es van declarant en la configuració del tallafoc es van guardant en el fitxer `/etc/iptables/rules.v4`; per tant, es pot modificar aquest fitxer directament per a modificar les regles del tallafoc.

Amb l'ordre `iptables-save` es pot guardar la configuració de les regles del tallafoc que hi ha en funcionament i amb l'ordre `iptables-restore` es poden recuperar les regles a partir d'un fitxer.

Per restaurar les regles que es tenen guardades en un fitxer propi o en el fitxer `rules.v4`, només cal fer `iptables-restore < rules.v4`.

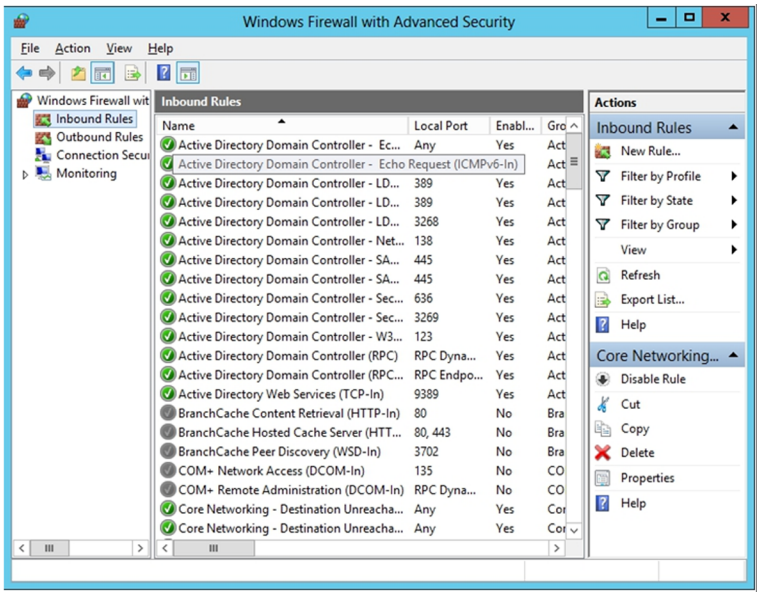
2.5. Configuració del tallafoc en el Windows Server

La darrera versió del Windows Server 2012 ja disposa del tallafoc instal·lat i en funcionament des del mateix moment de fer la instal·lació. Per defecte, i no com en les primeres versions, en què no en tenia, el tallafoc ja ve configurat

per a mirar quins són els serveis que estan configurats i adapta la configuració. Així, si s'instal·la un servei nou sobre el servidor, aquest ja modifica les regles del tallafoc perquè tot funcioni correctament.

Podem trobar el tallafoc en el menú d'eines de l'administrador del servidor.

Regles del tallafoc



Tant les regles d'entrada com les de sortida es poden configurar per separat. Les regles d'entrada són les que el servidor rep des de fora del sistema i són iniciades, per tant, des d'un altre equip informàtic, com per exemple les validacions al directori actiu, les peticions DNS, l'accés a pàgines web, etc.

Per tal de poder respondre a les peticions externes, s'ha de disposar d'una regla oberta per cada port o aplicació que està escoltant les peticions de la xarxa. Per tant, per tenir més segur el sistema, tot allò que no es fa servir, tots els ports que no cal tenir oberts, no tindran una regla oberta en el tallafoc i per tant es bloquejarà tot el trànsit que vulgui arribar al servidor per aquests ports o aplicacions que no estan permesos.

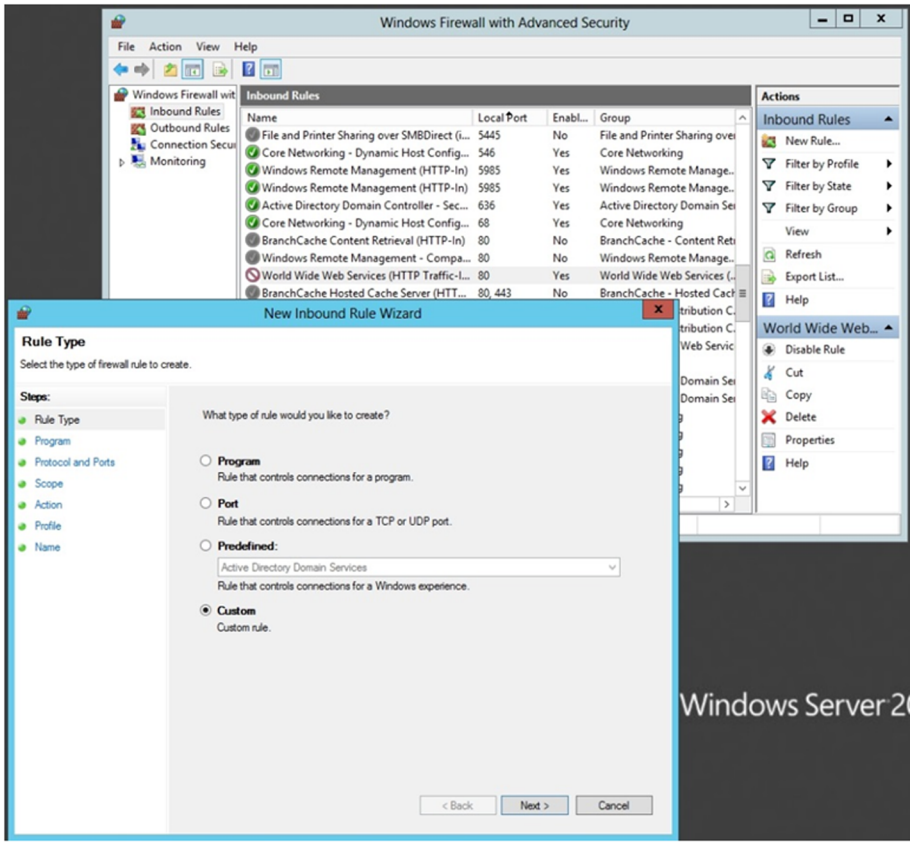
Les regles de sortida funcionen al revés que les d'entrada, són les que permeten o impedeixen sortir cap a l'exterior.

Es poden configurar noves regles en el tallafoc que siguin d'entrada o sortida del sistema, a partir de l'aplicació que es vol autoritzar a fer l'entrada i sortida, o directament autoritzant a un port en concret la comunicació en un sentit o un altre, és a dir, qui origina la comunicació. Es poden definir regles a partir dels ports que necessitem obrir, o fins i tot per serveis que ja estan predeterminats i per tant són molt més fàcil de configurar, tant per les regles d'entrada com per les regles de sortida.

El mateix tallafoc permet monitorar els esdeveniments que va detectant, en la pestanya de monitor es poden veure les regles actives i els registres (*logs*) que va deixant les peticions des de fora.

És molt important tenir completament tancats tots els ports i aplicacions que no es facin servir, tant d'entrada com de sortida, ja que així minimitzem els riscos d'atac i tenim molt més controlat el que està passant a la xarxa local. La regla general és tenir-ho tot tancat a menys que es necessiti tenir-ho obert perquè hi ha un programa o protocol que fa servir un port concret de comunicacions.

Configuració de les regles pròpies



3. Servidor de correu

Actualment, el correu electrònic és una de les aplicacions d'Internet que té més popularitat. En l'ús del correu electrònic intervenen diversos protocols ja fixats. El protocol d'accés als missatges d'Internet o *Internet message acces protocol* (IMAP) i la versió 3 del protocol d'oficina de correus o *post office protocol 3* (POP3) fan les funcions d'enviar els missatges als usuaris.

L'IMAP utilitza el port TCP número 143. Aquest protocol deixa tots els missatges i les carpetes on s'emmagatzemen aquests missatges en el servidor, cosa que permet accedir al compte de correu des de més d'un ordinador (el de casa, el de l'oficina, etc.).

El POP3, en canvi, utilitza el port TCP número 110. Aquest protocol esborra els missatges del servidor en el mateix moment en què el client els baixa. Per tant, si intentem consultar el correu des de més d'una màquina, ens trobem que en cadascuna de les màquines hi ha una part de tots els nostres missatges. Per solucionar aquest problema, el POP3 incorpora actualment una opció que permet deixar els missatges en el servidor durant un cert temps.

Quin avantatge té l'IMAP sobre el POP3? Amb l'IMAP podem deixar en el servidor no solament els missatges, sinó també una estructura de carpetes on es van emmagatzemant els missatges, mentre que amb l'POP3 només podem deixar els missatges en l'única carpeta que hi ha, ja que les carpetes es creen en local.

Hem de tenir una cosa molt clara: tant l'IMAP com el POP3 només són protocols per a rebre missatges. Si volem enviar un correu electrònic hem d'utilitzar el protocol simple de transferència de correu o *simple mail transfer protocol* (SMTP). Aquest protocol fa servir el port TCP número 25.

Els protocols que es fan servir per a enviar els missatges utilitzen canals no xifrats; si volem executar aquestes funcions amb protocols xifrats, hem de treballar amb l'IMAPS i el POP3S, que fan servir certificats SSL per a xifrar el canal de comunicació. Aquests protocols usen els ports TCP 993 en el cas de l'IMAPS i 995 en el cas del POP3S.

Aquestes configuracions s'hauran de tenir en compte a l'hora de configurar correctament el tallafoc, ja que si no creem una regla per a l'*iptables* que deixi passar cap a fora els paquets pel port 993 (IMAP segur), no es podran enviar correus electrònics de d'aquella màquina o des dels clients que facin servir aquest servidor per a fer de servidor d'enviament de correus electrònics.

3.1. Anàlisi de riscos i prevenció

En aquests últims anys, el correu electrònic s'ha convertit en una eina de comunicació molt arrelada, tant a l'empresa o en l'entorn domèstic. Per tant, qualsevol incidència en aquest servei repercuteix molt en els usuaris.

El principal risc del correu electrònic és que amb la lectura o execució d'un missatge es compromet la seguretat de la màquina des d'on llegim el correu (no del servidor). Aquest mena d'atacs resulta especialment perillós perquè la vulnerabilitat, com que està adjuntada a un missatge de correu llegit dins l'organització, pot travessar els tallafocs i accedir a la xarxa interna de l'empresa. A més, els atacs moderns s'autopropaguen, és a dir, una vegada infectada una màquina, es distribueixen (mitjançant el correu electrònic i els discos en xarxa) cap a altres màquines o usuaris. Els atacs que han tingut una difusió a escala mundial han estat el *iloveyou* i el *blaster*, que han estat distribuïts per correu electrònic.

Per acabar-ho de complicar més per a l'administrador, els usuaris del correu electrònic esperen que, quan envien un missatge, el receptor el rebi gairebé a l'acte. Abans era així, però el retard que es dóna entre l'enviament i la recepció és cada vegada més gran. Molts cops es deu a les mesures de seguretat que han pres els administradors de sistemes de les màquines que intervenen en el procés de comunicació.

Tots aquests punts fan que la solució a aquesta mena de problemes no sigui gens trivial. Els mètodes preventius moderns es basen en motors de comparació de patrons que busquen possibles atacs (tant si són virus com cucs o cavalls de Troia) i algorismes bayesians que intenten detectar quins són els missatges bons i quins són brossa. Aquests mètodes, com que són empírics, generen alguns "falsos positius", és a dir, cataloguen missatges com a virus o com a correu brossa sense que ho siguin realment. Per tant, ens podem trobar que els missatges enviats no arriben a la destinació que tenen a causa de les mesures de seguretat que s'han pres.

Els falsos positius augmenten o disminueixen proporcionalment respecte a les mesures de seguretat. Si aquestes mesures són molt restrictives, augmenten els falsos positius, però en canvi disminueix el risc de rebre atacs de l'exterior. Si les mesures de seguretat són molt permissives, disminueixen els falsos positius, però augmenta el risc que un atac tingui èxit. Per tant, actualment no hi ha una solució de molta seguretat i pocs falsos positius. La feina de supervisar freqüentment aquestes eines de prevenció perquè tinguin el nivell de compromís de seguretat adequat a la política de seguretat de l'empresa és de l'administrador de sistemes i de les màquines que allotgen el servidor de correu. És a dir, el que determina el nivell de seguretat d'aquestes eines és la política de seguretat de l'empresa, i la responsabilitat de l'administrador és adequar aquesta política al servidor de correu.

No solament tenim problemes en la recepció dels missatges, sinó que també podem tenir problemes amb el servidor de correu mateix. Antigament, qualsevol servidor de correu enviava el missatge de sortida fent servir el protocol SMTP de manera directa a la destinació; el port SMTP era obert completament i no es vigilava. Això va motivar que es fessin servir aquests ports oberts per a enviar el típic correu brossa a altres usuaris. Aquest fet tenia conseqüències negatives per al nostre servidor: d'una banda, la sobrecàrrega de la cua pel fet d'haver d'enviar tants missatges i, de l'altra, els servidors que tenien regles antiinundació consideraven el nostre servidor de correu una màquina *non grata*, cosa que feia que els missatges dels usuaris reals no poguessin arribar a la destinació que tenien. Per tant, no solament hem d'introduir mesures de seguretat a les cues de missatges entrants, sinó també als fitxers de configuració dels serveis de missatgeria.

Configurar el nostre SMTP per a enviar i rebre missatges únicament de les màquines que considera de confiança fa que hi hagi certa jerarquia entre màquines i que, en conseqüència, els missatges es concentrin. És a dir, si tots els missatges que envia el nostre servidor no els envia als destinataris sinó al servidor de correu del nostre proveïdor d'ISP, pot fer que augmenti el temps d'espera dels missatges en les cues entrants.

Afegir al nostre SMTP un motor de comparació que rastreja les cues de missatges a la recerca de diferents patrons predefinits (els antivirus). Com que cada dia els atacs són més sofisticats, els antivirus actuals no solament busquen virus, sinó que han ampliat el radi d'acció a cavalls de Troia, cucs i fins i tot algun tipus de correu brossa.

Si tenim en compte que actualment la majoria de servidors de correu tenen mecanismes de seguretat, pot arribar a passar que, si el nostre servidor és considerat no grat, no podrem lliurar tots els missatges que envien els usuaris. Per tant, els administradors s'han de preocupar no solament de no rebre cap tipus de missatge no volgut, sinó també del possible error de no enviar-lo. En resum, els motors de comparació de patrons no solament recorren la cua de missatges entrants, sinó que en molts casos també recorren la cua de missatges sortints.

4. Servidor de web i FTP

Avui dia és difícil trobar algú que no s'hagi connectat a Internet alguna vegada; la majoria d'aquestes persones que utilitzen la Xarxa de comunicacions descriurien el funcionament d'una petició de la manera següent: "el meu navegador genera una connexió a un servidor web, sol·licita una pàgina i la rep". Aquesta visió de funcionament d'una sol·licitud d'una pàgina web, malgrat que és correcta, és molt superficial.

Imaginem-nos que volem visualitzar la pàgina web següent:

http://www.uoc.edu/web/cat/index.html

El procediment vist més exhaustivament és el següent:

1) L'URL del navegador es divideix en tres parts:

2) El protocol utilitzat (HTTP).

3) El nom del servidor (www.uoc.edu).

4) El camí que s'ha de seguir (/web/cat/) fins al fitxer volgut (index.html).

5) El primer que fa el navegador és traduir el nom del servidor a una adreça IP. Per fer-ho, duu a terme una petició DNS al servidor que té configurat per a aquesta finalitat. Depenent de l'ISP que utilitzem, aquest servidor canvia. I rebem la resposta amb l'adreça IP del servidor web.

6) Llavors, el navegador fa una connexió al port 80 de la màquina identificada per l'adreça IP corresponent al nom (www.uoc.edu). És a dir, fa una connexió TCP al servidor web de la Universitat Oberta de Catalunya (UOC).

7) Mitjançant el protocol HTTP, el navegador sol·licita obtenir l'arxiu index.html.

8) El servidor envia al client l'arxiu.

9) El navegador del client interpreta les ordres de l'arxiu index.html i les mostra per pantalla.

En aquest exemple de connexió, a més del funcionament d'una petició, veiem que el model de connexió que utilitza el servidor web és de dues capes: la capa de presentació (que és en el client) i la capa de domini (que és en el servidor).

Tot i que sembla un servei senzill, el servidor web té més funcions a banda de servir pàgines web. Entre aquestes funcions, hi ha processar informació mitjançant *scripts* CGI i afegir algun nivell de seguretat en les tasques que té.

Es parla de processament d'informació quan en una pàgina web surt un formulari per a introduir text i s'obté un resultat diferent segons el text introduït. Un exemple de processament d'informació és una cerca en una pàgina web.

Si visitem una pàgina web i el navegador ens mostra una finestra de diàleg en què ens demana el nom d'usuari i la contrasenya, aquesta protecció de pàgines per contrasenya també la fa el servidor web. Els servidors antics permetien que l'administrador de la màquina mantingués una llista amb els usuaris i les contrasenyes respectives per a accedir a aquestes pàgines protegides. Els servidors es responsabilitzaven d'autenticar els paràmetres introduïts pels usuaris, comparant-los amb els que hi havia en la llista d'accés.

Avui dia, aquests nivells bàsics de seguretat han quedat obsolets, ja que els servidors moderns, per a fer funcions d'autenticació i de privadesa, utilitzen canals xifrats amb SSL. El port TCP de servei web xifrat és el 443.

El protocol de transferència de fitxers o *file transfer protocol* (FTP), que pertany a la família de protocols TCP/IP, va començar essent una utilitat inclosa en el mateix sistema operatiu Unix que es feia servir per a transferir arxius entre els equips connectats a una xarxa. Actualment, ja és un protocol estàndard. El funcionament de l'FTP es basa en la comunicació client-servidor.

A diferència d'altres protocols, l'FTP utilitza dos ports de TCP per a fer la comunicació:

- El port de control, on es fa el diàleg entre el client i el servidor. Aquest port és el número 21.

- El port de dades, on es fa la transferència d'informació entre el client i el servidor. És el port 20.

El protocol FTP permet copiar fitxers entre dues màquines mitjançant la xarxa. Hi ha dues modalitats d'FTP:

- L'FTP pròpiament dit: cal tenir en compte d'usuari en la màquina a la qual es pretén accedir. Se sol fer servir per a accedir als "nostres" fitxers ubicats en el servidor. És a dir, nosaltres, com a usuaris, tenim un espai de disc situat en el servidor per a guardar els documents; quan treballem a l'empresa accedim a aquest espai mitjançant les unitats compartides (o unitats de xarxa) i quan som a casa hi accedim mitjançant l'FTP.

- L'FTP anònim: no hi cal tenir compte. Se sol fer servir per a aconseguir programes de domini públic (programari de prova o *shareware*). Com a

nom d'usuari, la identificació és *anonymous*; com a clau es fa servir el nom de domini i normalment també el correu electrònic de l'usuari que demana la connexió.

Finalment, hem de saber que l'FTP és capaç de transferir fitxers binaris i de text.

4.1. Servidor web en el GNU/Linux

4.1.1. Instal·lació de l'Apache + SSL

Partim de la base que ja està instal·lat el servidor de pàgines web, en tot cas només cal fer la instal·lació bàsica a partir del següent:

```
root# apt-get install apache2
```

Amb aquesta senzilla ordre s'instal·la el servidor web i configura una pàgina web que ja és accessible. Però a continuació es mostrarà com fer que aquesta pàgina web passi a ser segura, és a dir, que faci servir el protocol HTTPS en lloc del normal amb HTTP.

El primer que cal fer és activar el mòdul SSL de l'Apache, fent:

```
root# a2enmod ssl
```

I reiniciar posteriorment el servidor de pàgines web Apache, fent:

```
root# /etc/init.d/apache2 restart
```

Això farà que agafi ja la configuració del servei SSL i per tant aquest procés de servei de pàgines escolti peticions amb el protocol segur, i no solament amb el protocol HTTP. Podem veure si realment està escoltant o no pel port del protocol HTTPS fent servir l'ordre següent i mirar si realment existeix l'HTTPS en la resposta.

```
root# netstat -tap | grep https
....
tcp6   0  0 [::]:https    [::]:*    LISTEN   1238/apache2
root#
```

Com podem veure en la resposta, el servidor web també està escoltant les peticions per HTTPS, tal com esperàvem que es produís.

A partir d'ara crearem el vhost `www.uoc-test.net` per fer la configuració d'aquesta pàgina web al servidor perquè funcioni amb el protocol segur d'HTTPS. El primer que cal fer és crear el directori on s'ha de col·locat tota la informació relacionada amb aquest portal.

```
root# mkdir /var/www/www.uoc-test.net
```

El servidor Apache agafa la configuració per defecte que té en el fitxer `/etc/apache2/sites-available/default-ssl`, però necessitem canviar algunes coses perquè funcioni per al nou lloc (*site*) que estem construint. Així, podem partir de la base d'aquest fitxer i editar-lo per modificar el necessari.

```
root# cp /etc/apache2/sites-available/default-ssl /etc/apache2/sites-available/
www.uoc-test.net-ssl
root# joe /etc/apache2/sites-available/www.uoc-test.net-ssl
```

i modificarem el contingut perquè quedi de la manera següent, on es modifica el correu electrònic i nom del servidor, a més dels directoris on estan ubicats els documents de les pàgines web.

```
<IfModule mod_ssl.c>
<VirtualHost _default_:443>
#   DocumentRoot /srv/www/mydomain.com/public_html/
#   ErrorLog /srv/www/mydomain.com/logs/error.log
#   CustomLog /srv/www/mydomain.com/logs/access.log combined



 ServerAdmin webmaster@localhost
  ServerName www.uoc-test.net:443

 DocumentRoot /var/www/www.uoc-test.net
 <Directory />
  Options FollowSymLinks
  AllowOverride None
 </Directory>
 <Directory /var/www/www.uoc-test.net/>
  Options Indexes FollowSymLinks MultiViews
  AllowOverride None
  Order allow,deny
  allow from all
 </Directory>

 ScriptAlias /cgi-bin/ /usr/lib/cgi-bin/
 <Directory "/usr/lib/cgi-bin">
  AllowOverride None
  Options +ExecCGI -MultiViews +SymLinksIfOwnerMatch
  Order allow,deny
```

```
 Allow from all
</Directory>

ErrorLog ${APACHE_LOG_DIR}/error.log

# Possible values include: debug, info, notice, warn, error, crit,
# alert, emerg.
LogLevel warn

CustomLog ${APACHE_LOG_DIR}/ssl_access.log combined

#  SSL Engine Switch:
#  Enable/Disable SSL for this virtual host.
SSLEngine on

#  A self-signed (snakeoil) certificate can be created by installing
#  the ssl-cert package. See
#  /usr/share/doc/apache2.2-common/README.Debian.gz for more info.
#  If both key and certificate are stored in the same file, only the
#  SSLCertificateFile directive is needed.
SSLCertificateFile  /etc/ssl/certs/ssl-cert-snakeoil.pem
 SSLCertificateKeyFile /etc/ssl/private/ssl-cert-snakeoil.key

#  Server Certificate Chain:
#  Point SSLCertificateChainFile at a file containing the
#  concatenation of PEM encoded CA certificates which form the
#  certificate chain for the server certificate. Alternatively
#  the referenced file can be the same as SSLCertificateFile
#  when the CA certificates are directly appended to the server
#  certificate for convinience.
#SSLCertificateChainFile /etc/apache2/ssl.crt/server-ca.crt

#  Certificate Authority (CA):
#  Set the CA certificate verification path where to find CA
#  certificates for client authentication or alternatively one
#  huge file containing all of them (file must be PEM encoded)
#  Note: Inside SSLCACertificatePath you need hash symlinks
#     to point to the certificate files. Use the provided
#     Makefile to update the hash symlinks after changes.
#SSLCACertificatePath /etc/ssl/certs/
#SSLCACertificateFile /etc/apache2/ssl.crt/ca-bundle.crt

#  Certificate Revocation Lists (CRL):
#  Set the CA revocation path where to find CA CRLs for client
#  authentication or alternatively one huge file containing all
#  of them (file must be PEM encoded)
#  Note: Inside SSLCARevocationPath you need hash symlinks
```

```
#     to point to the certificate files. Use the provided
#     Makefile to update the hash symlinks after changes.
#SSLCARevocationPath /etc/apache2/ssl.crl/
#SSLCARevocationFile /etc/apache2/ssl.crl/ca-bundle.crl

#  Client Authentication (Type):
#  Client certificate verification type and depth. Types are
#  none, optional, require and optional_no_ca. Depth is a
#  number which specifies how deeply to verify the certificate
#  issuer chain before deciding the certificate is not valid.
#SSLVerifyClient require
#SSLVerifyDepth 10

#  Access Control:
#  With SSLRequire you can do per-directory access control based
#  on arbitrary complex boolean expressions containing server
#  variable checks and other lookup directives. The syntax is a
#  mixture between C and Perl. See the mod_ssl documentation
#  for more details.
#<Location />
#SSLRequire (  %{SSL_CIPHER} !~ m/^(EXP|NULL)/ \
#      and %{SSL_CLIENT_S_DN_O} eq "Snake Oil, Ltd." \
#      and %{SSL_CLIENT_S_DN_OU} in {"Staff", "CA", "Dev"} \
#      and %{TIME_WDAY} >= 1 and %{TIME_WDAY} <= 5 \
#      and %{TIME_HOUR} >= 8 and %{TIME_HOUR} <= 20    ) \
#      or %{REMOTE_ADDR} =~ m/^192\.76\.162\.[0-9]+$/
#</Location>

#  SSL Engine Options:
#  Set various options for the SSL engine.
#  o FakeBasicAuth:
#   Translate the client X.509 into a Basic Authorisation. This means that
#   the standard Auth/DBMAuth methods can be used for access control. The
#   user name is the `one line' version of the client's X.509 certificate.
#   Note that no password is obtained from the user. Every entry in the user
#   file needs this password: `xxj31ZMTZzkVA'.
#  o ExportCertData:
#   This exports two additional environment variables: SSL_CLIENT_CERT and
#   SSL_SERVER_CERT. These contain the PEM-encoded certificates of the
#   server (always existing) and the client (only existing when client
#   authentication is used). This can be used to import the certificates
#   into CGI scripts.
#  o StdEnvVars:
#   This exports the standard SSL/TLS related `SSL_*' environment variables.
#   Per default this exportation is switched off for performance reasons,
#   because the extraction step is an expensive operation and is usually
#   useless for serving static content. So one usually enables the
```

```
    #   exportation for CGI and SSI requests only.
    #  o StrictRequire:
    #   This denies access when "SSLRequireSSL" or "SSLRequire" applied even
    #   under a "Satisfy any" situation, i.e. when it applies access is denied
    #   and no other module can change it.
    #  o OptRenegotiate:
    #   This enables optimized SSL connection renegotiation handling when SSL
    #   directives are used in per-directory context.
    #SSLOptions +FakeBasicAuth +ExportCertData +StrictRequire
    <FilesMatch "\.(cgi|shtml|phtml|php)$">
     SSLOptions +StdEnvVars
    </FilesMatch>
    <Directory /usr/lib/cgi-bin>
     SSLOptions +StdEnvVars
    </Directory>

    #  SSL Protocol Adjustments:
    #  The safe and default but still SSL/TLS standard compliant shutdown
    #  approach is that mod_ssl sends the close notify alert but doesn't wait for
    #  the close notify alert from client. When you need a different shutdown
    #  approach you can use one of the following variables:
    #  o ssl-unclean-shutdown:
    #   This forces an unclean shutdown when the connection is closed, i.e. no
    #   SSL close notify alert is send or allowed to received. This violates
    #   the SSL/TLS standard but is needed for some brain-dead browsers. Use
    #   this when you receive I/O errors because of the standard approach where
    #   mod_ssl sends the close notify alert.
    #  o ssl-accurate-shutdown:
    #   This forces an accurate shutdown when the connection is closed, i.e. a
    #   SSL close notify alert is send and mod_ssl waits for the close notify
    #   alert of the client. This is 100% SSL/TLS standard compliant, but in
    #   practice often causes hanging connections with brain-dead browsers. Use
    #   this only for browsers where you know that their SSL implementation
    #   works correctly.
    #  Notice: Most problems of broken clients are also related to the HTTP
    #  keep-alive facility, so you usually additionally want to disable
    #  keep-alive for those clients, too. Use variable "nokeepalive" for this.
    #  Similarly, one has to force some clients to use HTTP/1.0 to workaround
    #  their broken HTTP/1.1 implementation. Use variables "downgrade-1.0" and
    #  "force-response-1.0" for this.
    BrowserMatch "MSIE [2-6]" \
     nokeepalive ssl-unclean-shutdown \
     downgrade-1.0 force-response-1.0
    # MSIE 7 and newer should be able to use keepalive
    BrowserMatch "MSIE [17-9]" ssl-unclean-shutdown

</VirtualHost>
```

```
</IfModule>
```

Com es pot veure, en el codi anterior la pàgina web està signada amb els certificat autosignats que té per defecte Debian. Ara provarem si funcionen realment els certificats i el servidor mitjançant el servei HTTPS.

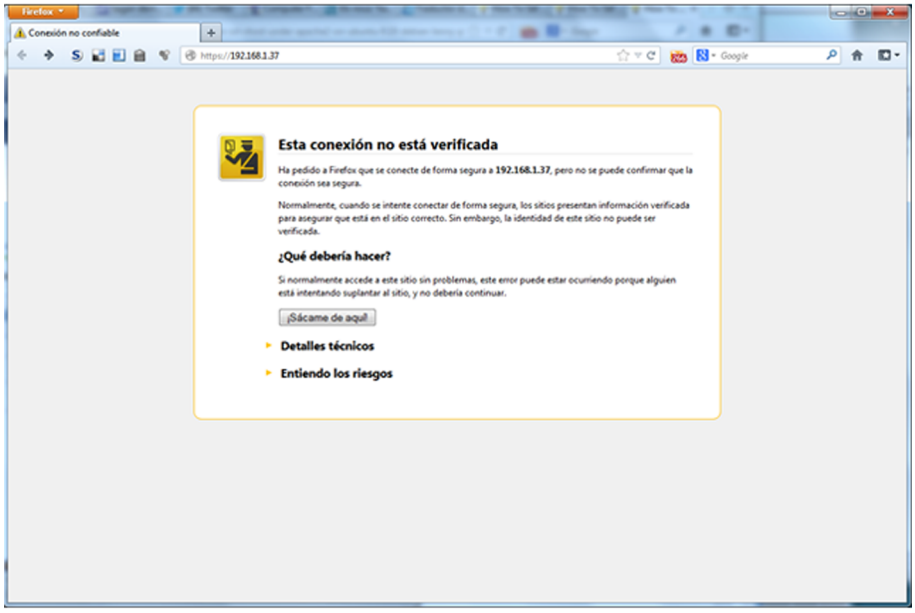
El que hem de fer és canviar el vhost per defecte pel que tenim configurat.

```
root# a2dissite default-ssl
root# a2ensite www.hostmauritius.com-ssl
root# /etc/init.d/apache2 reload
```

Ara ja podem obrir un navegador i mirar què passa quan fem la crida a la màquina que acabem de configurar. Com que s'estan fent servir els certificats autosignats de Debian, no verificats per cap organisme de certificació, el navegador avisarà d'això, i ens deixa seguir o no sota la responsabilitat de la persona que vol visitar la pàgina web: si pensa que aquesta pàgina web, signada amb certificats propis, és segura o no.

Per tant, com mostra la figura següent, tenir una pàgina web amb un certificat propi és relativament senzill i ràpid de fer, però cada persona que ha de saber si l'accés a aquesta pàgina web és segura o no.

Pàgina web creada



Mostrem ara com s'han de crear els certificats propis que després farem servir per a accedir de manera segura a la pàgina web.

En primer lloc, cal instal·lar el programari que crea els certificats per a ssl:

```
root# apt-get install ssl-cert
```

I crearem els certificats propis fent l'ordre següent:

```
root# make-ssl-cert /usr/share/ssl-cert/ssleay.cnf /etc/ssl/private/www.uoc-test.net.crt
```

En l'execució demanarà el nom de la pàgina web, on haurem de posar la nostra (en aquest cas de prova seria `www.uoc-test.net`). Això crearà un fitxer `/etc/ssl/private/www.uoc-test.net.crt` amb els dos certificats, tant el públic com el privat, i per tant els haurem de separar en els dos fitxers corresponents.

Si mostrem el contingut del certificat, veurem clarament on comença l'un i on l'altre; per tant, copiant i editant els dos fitxers podem eliminar la part que no interessa de cada certificat. La part privada del certificat (la primera) la guardarem en el fitxer `/etc/ssl/private/www.uoc-test.net.key` i el certificat autosignat públic el deixarem en el fitxer `/etc/ssl/certs/www.uoc-test.net.pem`. És important que només tingui accés a la part privada del certificat l'usuari *root* de la màquina, perquè ningú la pugui copiar o manipular. Per això cal fer l'ordre següent, que ens assegura que ningú té accés al fitxer.

```
root# chmod 600 /etc/ssl/private/www.uoc-test.net.key
```

I també s'ha d'esborrar completament el fitxer generat inicialment, ja que té la part privada del certificat inclosa i per tant es podria extreure d'allà molt fàcilment:

```
root# rm -f /etc/ssl/private/www.uoc-test.net.crt
```

Ara ja únicament cal canviar els vhost de l'SSL perquè tinguin els nous certificats generats per nosaltres mateixos. Tot i que no solucionarem el problema de la confiança en el certificat, no seran els certificats que per defecte fa servir Debian i que qualsevol persona pot obtenir molt fàcilment i que podrien suplantar la nostra pàgina web amb el mateix certificat que estiguéssim fent servir en aquesta pàgina web. Per tant, només cal editar el fitxer:

```
root# joe /etc/apache2/sites-available/www.uoc-test.net-ssl
```

I fer els canvis perquè tinguem en compte els certificats següents:

```
    #  A self-signed (snakeoil) certificate can be created by installing
    #  the ssl-cert package. See
    #  /usr/share/doc/apache2.2-common/README.Debian.gz for more info.
    #  If both key and certificate are stored in the same file, only the
    #  SSLCertificateFile directive is needed.
  SSLCertificateFile /etc/ssl/certs/www.uoc-test.net.pem
  SSLCertificateKeyFile /etc/ssl/private/www.uoc-test.net.key
```

Ara reiniciarem el servei Apache fent que torni a llegir tota la configuració:

```
root# /etc/init.d/apache2 reload
```

Ara ja només queda crear un certificat que es pugui enviar a una entitat certificadora com pot ser Verising, Thawte o Comodo perquè certifiquin que el domini és legítim i tenim un certificat per al nostre domini. Cal notar que aquestes entitats certificadores cobren per fer aquest servei, per la qual cosa només mostrarem els passos fins a enviar el certificat a aquestes entitats.

Crearem un directori per poder treballar còmodament.

```
root# mkdir /etc/ssl/csr
```

A partir de la clau privada del certificat que s'ha creat abans, crearem un nou certificat, que serà el que es farà servir per a les pàgines web:

```
root# openssl req -new -key /etc/ssl/private/www.uoc-test.net.key -out/etc/ssl/csr/
www.uoc-test.net.csr
```

Anirà preguntant dades que s'han d'omplir correctament per tal que surti bé la informació en el certificat; en cas contrari, l'entitat certificadora podria no fer el certificat.

És important per tal de verificar el domini que en el camp de "Common name" es posi el nom del domini que es té (en el nostre cas `www.uoc-test.net`) i que la resta d'informació sigui correcta.

Al final del procés disposarem del certificat que s'ha d'enviar al fitxer `/etc/ssl/csr/www.uoc-test.net.com.csr`.

L'entitat certificadora ens enviarà un nou certificat públic, amb extensió .PEM, que s'haurà de copiar en el directori on tenim ara el certificat autosignat antic, a: `/etc/ssl/certs/www.uoc-test.net.pem`.

Hi ha entitats certificadores que, a més del certificat que envien, també fan que s'hagi d'instal·lar un certificat propi per tal de poder fer la certificació. En aquest cas, només caldrà copiar el certificat en la mateixa carpeta i modificar el fitxer dels vhost on tenim el camí dels nostres certificats per incloure el certificat de l'entitat:

```
root# joe /etc/apache2/sites-available/www.uoc-test.net-ssl

...
SSLCertificateFile  /etc/ssl/certs/www.uoc-test.net.pem
SSLCertificateKeyFile /etc/ssl/private/www.uoc-test.net.key

#  Server Certificate Chain:
#  Point SSLCertificateChainFile at a file containing the
```

```
#  concatenation of PEM encoded CA certificates which form the
#  certificate chain for the server certificate. Alternatively
#  the referenced file can be the same as SSLCertificateFile
#  when the CA certificates are directly appended to the server
#  certificate for convinience.
SSLCertificateChainFile /etc/ssl/certs/CAcert_chain.pem
...
```

4.1.2. Configuració Apache

Acabada la configuració inicial del servidor de pàgines web Apache, mirem el fitxer de configuració del servidor per veure que tot està configurat correctament. Aquesta configuració pot estar dividida en diferents fitxers de configuració. Per garantir el funcionament correcte de les versions antigues d'Apache, s'ha optat per continuar tenint els mateixos fitxers i afegir la configuració de les noves versions en altres fitxers. Així, tenim els fitxers `httpd.conf` per a les versions antigues, el fitxer `apache2.conf`, i els fitxers de configuració dels vhost que s'han mostrat anteriorment i que estan ubicats per cada un dels allotjadors virtuals que es tinguin configurats en el servidor Apache. En el cas d'instal·lar una nova versió de l'Apache, el fitxer `httpd.conf` estarà completament buit i tota la configuració es posarà en el fitxer `apache2.conf`.

Trobem també les variables d'entorn del servidor web en un fitxer a part (`/etc/apache2/envvars`), on hi haurà totes les variables que es poden modificar i que afecten el funcionament del servidor de pàgines web. Una de les més importants des del punt de vista de la seguretat és l'usuari amb què s'executa l'aplicació Apache. Antigament, si l'Apache s'instal·lava des de l'usuari *root* el servidor s'executava amb aquest i, per tant, quan s'explotava una vulnerabilitat i s'obtenia el control de l'aplicació Apache, es podia tenir accés al sistema com a usuari *root*.

Actualment, durant la instal·lació de l'aplicació Apache es crea un usuari i un grup que es dediquen únicament a executar l'aplicació. L'usuari i grup són el www-data, i es pot comprovar que realment s'està fent servir aquest usuari i grup en el fitxer de variables `/etc/apache2/envvars`, on ha d'estar declarat l'usuari i el grup corresponents:

```
export APACHE_RUN_USER=www-data
export APACHE_RUN_GROUP=www-data
```

4.2. Servidor d'FTP en el GNU/Linux

El servidor FTP permet compartir de manera ràpida fitxers entre diferents equips ubicats a la xarxa. És un protocol que no és segur, ja que la transferència dels fitxers no està xifrada i per tant es poden obtenir si algú té monitorada la xarxa i pot llegir tot el que passa per ella.

Si el que es necessita és poder compartir fitxers entre usuaris de la mateixa entitat, el millor és instal·lar el servei SSH/SSL en el servidor de fitxers, i fer així que la compartició d'aquests arxius sigui xifrada i que únicament els usuaris validats puguin veure el contingut del servidor. Amb un client d'SSH podrem iniciar sessió en el servidor, i amb un client d'SFTP es podrà entrar al servidor i obtenir tots els arxius necessaris.

Però si el que es necessita és tenir una mena de repositori, on tothom pugui entrar i obtenir els fitxers que hi posem, llavors s'ha d'optar per tenir un servidor FTP, ja que no cal validar els usuaris i això fa molt més àgil la transferència de fitxers.

4.2.1. Instal·lació del servidor d'FTP

El protocol FTP és un dels protocols més antics de la família TCP/IP. Una prova d'això és la multitud de servidors FTP que hi ha. En aquesta secció mostrarem la instal·lació del VSFTPD.

Mirem la descripció:

```
root# apt-cache search vsftpd
vsftpd - lightweight, efficient FTP server written for security
root#
```

Per fer la instal·lació, hem d'executar l'ordre següent:

```
root# apt-get install vsftpd
```

Una vegada acabada la breu instal·lació, configurem el servidor.

4.2.2. Configuració del servidor FTP a GNU/Linux

El fitxer de configuració del servidor FTP és `/etc/vsftpd.conf`. Editem aquest arxiu i descomentem, si no ho estan ja, les línies següents:

```
# Allow anonymous FTP? (Beware - allowed by default if you comment this out).
anonymous_enable = YES

# Uncomment this to allow local users to log in.
local_enable=YES
```

El primer paràmetre ens permet fer un FTP de tipus anònim, és a dir, que no cal tenir usuari en el servidor, i el segon permet als usuaris que tenen compte en el servidor accedir-hi per FTP. Si no volem que s'hi pugui accedir de manera anònima, s'ha de posar el símbol # per comentar aquesta línia.

Si volem modificar el missatge de benvinguda que surt als usuaris quan es connecten per FTP, hem de modificar el contingut del paràmetre `ftp_banner` i posar-hi el missatge:

```
# You may fully customise the login banner string:
ftpd_banner = New welcome to blah FTP service.
```

Amb la configuració feta fins ara ja n'hi hauria prou perquè els usuaris, quan s'autentiquen, accedeixen al seu `home_directory`, però no hi ha res que els impedeixi navegar per la resta de directoris del servidor. Si volem que els nostres usuaris no puguin accedir a cap directori que no sigui `home_directory`, hem de descomentar el paràmetre següent:

```
# You may restrict local users to their home directories. See the FAQ for
# the possible risks in this before using chroot_local_user or
# chroot_list_enable below.
chroot_local_user=YES
```

Si volem deixar que els nostres usuaris no solament baixin arxius del seu directori, sinó que també hi posin dades, hem de configurar els paràmetres següents:

```
# Uncomment this to enable any form of FTP write command.
write_enable = YES

# By default the server will pretend to allow ASCII mode but in fact ignore
# the request. Turn on the below options to have the server actually do ASCII
# mangling on files when in ASCII mode.
# Beware that on some FTP servers, ASCII support allows a denial of service
# attack (DoS) via the command "SIZE /big/file" in ASCII mode. vsftpd
# predicted this attack and has always been safe, reporting the size of the
# raw file.
# ASCII mangling is a horrible feature of the protocol.
ascii_upload_enable = YES
ascii_download_enable = YES
```

Podem limitar el nombre de connexions actives simultànies, tant des de les IP com dels clients amb les opcions següents, així serà més segur, ja que únicament permetem que un usuari es connecti des d'una IP.

```
# If vsftpd is in standalone mode, this is the maximum number of clients
# which may be connected. Any additional clients connecting will get
# an error message.
max_clients=5

# If vsftpd is in standalone mode, this is the maximum number of clients
# which may be connected from the same source internet address. A client
```

```
# will get an error message if they go over this limit.
max_per_ip=1
```

Podem limitar quins usuaris no es poden autenticar en el servidor TFP, així ens assegurem que aquests usuaris no podran entrar-hi mai. Per exemple aquests:

```
root# cat /etc/ftpusers
# /etc/ftpusers: list of users disallowed FTP access. See ftpusers(5).

root
daemon
bin
sys
sync
games
man
lp
mail
news
uucp
nobody
www-data
root#
```

El directori per defecte que crea per als usuaris anònims està ubicat a `/srv/ftp`, i és aquí on s'hauran de posar tots els fitxers que es vulguin compartir directament.

Hi ha moltes més opcions de configuració, com limitar l'amplada de banda per als usuaris, per a les connexions anònimes, fer que els usuaris, en lloc de ser del sistema, estiguin en una base de dades, etc. Aquestes opcions ens permeten configurar el servidor una mica més segur. Si volem saber com s'activen aquestes opcions, hem de consultar:

```
root# man 5 vsftpd.conf
```

Finalment, tan sols queda indicar com arrencar el servidor. Quan hem fet la instal·lació, s'ha creat un arxiu d'arrencada a `/etc/init.d` anomenat `vsftpd`. Per a carregar totes les noves opcions cal parar i arrencar el servidor executant les ordres següents:

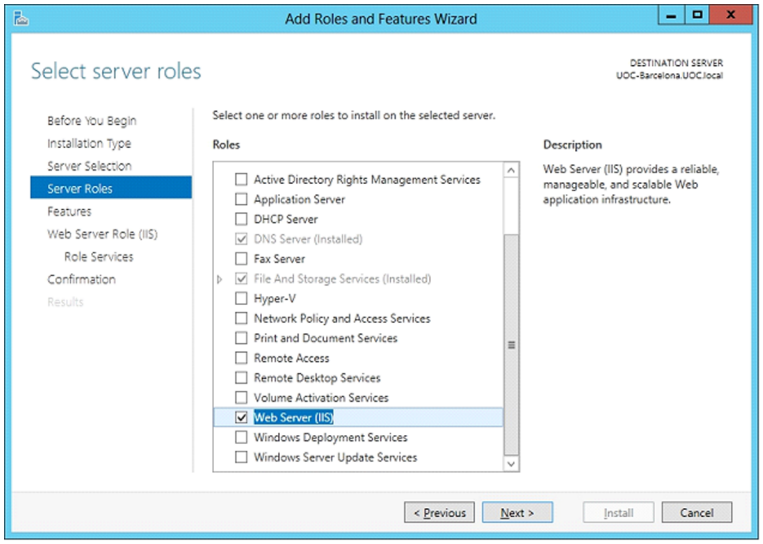
```
root# /etc/init.d/vsftpd restart
```

4.3. Servidor web i FTP en el Windows Server 2012

4.3.1. Servidor d'informació d'Internet

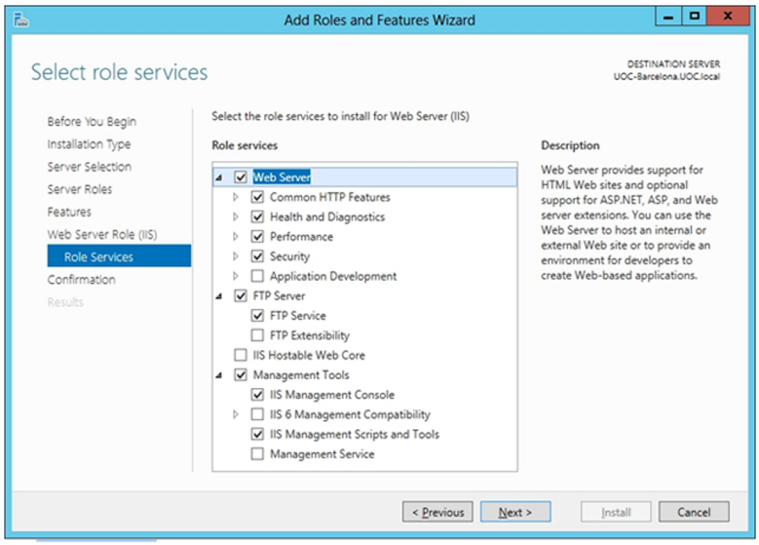
IIS és el conjunt de serveis d'Internet que proporciona el Windows Server 2012. Entre aquests serveis, hi ha el servidor de pàgines web i el de transferència de fitxers per protocol FTP, i també el suport per a pàgines dinàmiques amb ASP o ASP.NET, el servei de protocol simple de transferència de correu (SMTP), el servei de protocol de transport de notícies de xarxa (NNTP), la impressió mitjançant protocol HTTP (protocol d'impressió per Internet o *Internet printing protocol*, IPP), les extensions de transferència intel·ligent de fitxers (BITS) i els serveis de reproducció en temps real o *streaming* de mitjans Microsoft Windows Media (difusió de presentacions multimèdia d'alta qualitat per Internet).

Instal·lació de l'IIS



L'IIS no s'instal·la per defecte i per tant si s'hi vol instal·lar, s'ha de fer amb el configurador del servidor, instal·lant el rol corresponent a l'IIS. Un cop seleccionat, ens apareixerà una pàgina per incloure tots els serveis que es vulguin tenir en el servidor, com pot ser l'FTP.

Serveis que es volen instal·lar

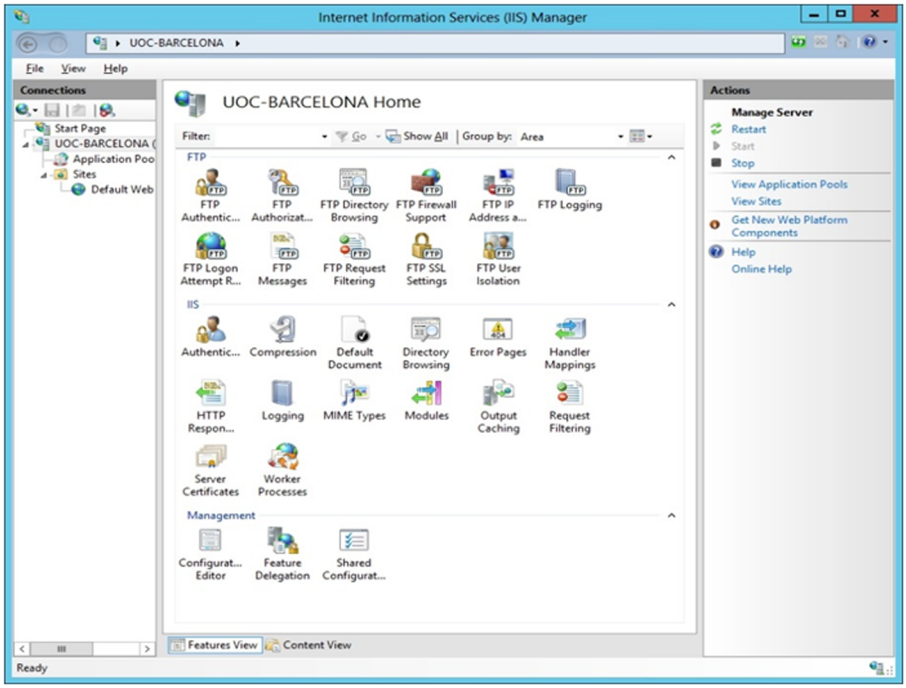


És important seleccionar només els serveis o components de l'IIS que són necessaris, ja que com més serveis hi ha instal·lats, més gran és la superfície d'atac del nostre servidor i, per tant, més gran és la probabilitat de patir un atac de seguretat. Serveis com FTP, SMTP o NNTP, que no són gaire habituals, val més no instal·lar-los tret que siguin necessaris per a l'organització.

La instal·lació de l'IIS crea en el disc dur, en el qual hi ha instal·lat el sistema operatiu, una estructura de carpetes dins del directori `Inetpub`. Dins d'aquest directori trobem dues carpetes (`wwwroot` i `ftproot`) que, com indica el nom, són l'arrel dels directoris públics per defecte de web i FTP.

Per a configurar i administrar l'estructura web i FTP, utilitzem l'eina "Administrador d'Internet Information Services", que trobarem dins el menú d'eines de l'administrador del servidor. Aquesta eina mostra d'una manera molt ràpida i gràfica tots els components instal·lats i com s'han de configurar.

Configuració de l'IIS



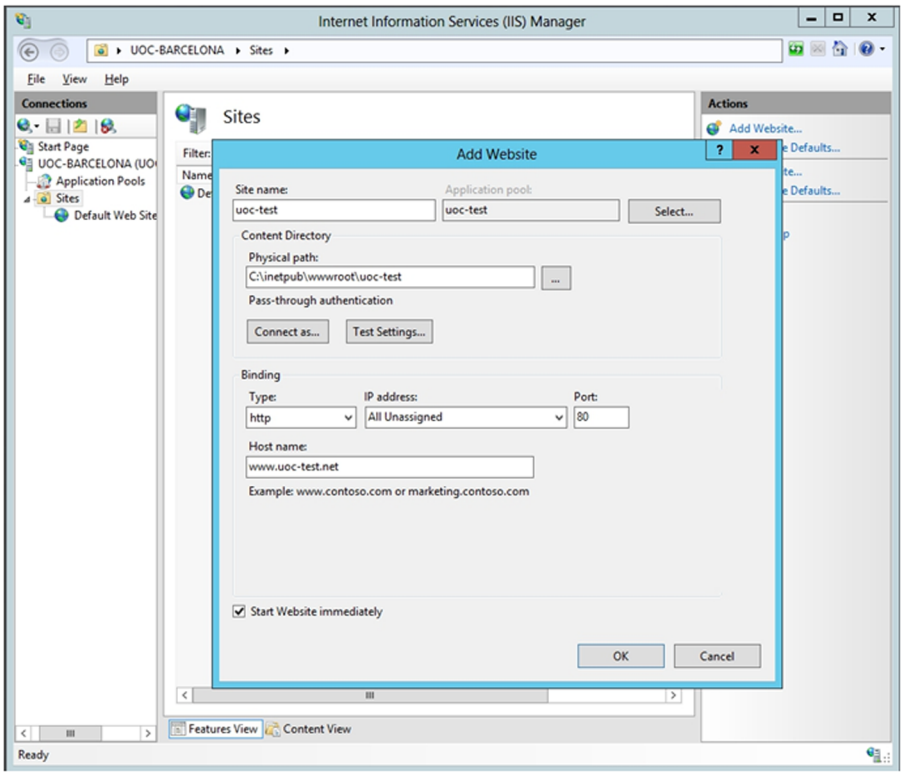
Configuració d'un lloc web

L'IIS crea un lloc web predeterminat situat a `Inetpub\wwwroot`. Podem crear altres llocs web addicionals en altres directoris del disc dur del servidor, però només un lloc web pot escoltar sobre un port TCP alhora.

Per crear un lloc web, seleccionem l'element *sites* que conté tots els llocs web del servidor, que està dins el servidor local, i amb el botó dret obrim el menú contextual i seleccionem l'opció "Afegir lloc web". L'assistent de creació de llocs ens sol·licita la informació necessària per a crear-los: en primer lloc, una descripció del contingut del lloc web i, a continuació, l'adreça IP i el port TCP pel qual s'hi accedirà. El port per defecte és el 80 i, encara que es pot canviar, això implica que els usuaris no podran accedir directament al lloc web des del navegador si no saben el port concret que utilitza el servei.

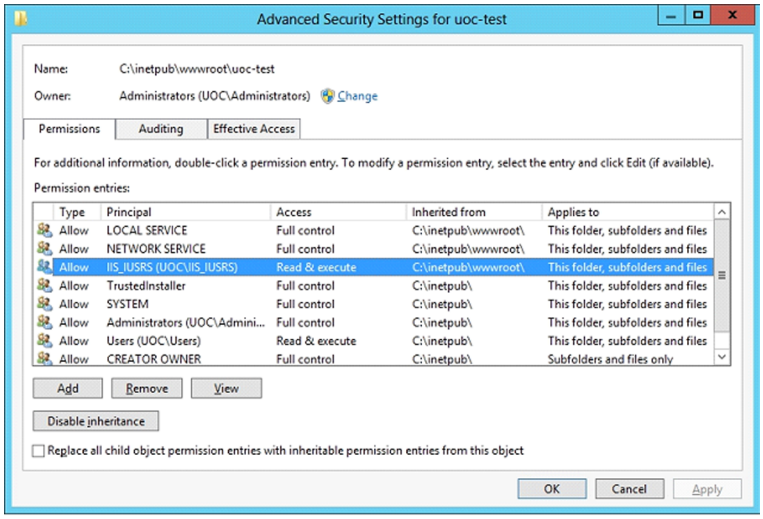
En la figura següent podem veure que també es pot configurar perquè sigui una pàgina segura fent servir HTTPS en lloc d'HTTP: en aquest cas s'haurà de seleccionar un certificat validat per una entitat certificadora, com s'ha vist en l'apartat anterior.

Afegint un nou lloc



Escollirem també la ruta d'accés arrel al lloc web, a partir de la qual se situen la resta d'arxius del lloc web. Amb això ja es crea la infraestructura dins el servidor per tal de tenir una pàgina web. Podem mirar que els usuaris que accedeixin a la web ho facin amb els permisos pertinents, i ho podrem veure en les propietats del nou lloc que acabem de configurar. Tret que sigui necessari algun altre permís, els únics que han d'estar seleccionats perquè sigui més segur són el de "Lectura" i el d'"Execució" si es tracta de pàgines ASP o ASP.NET. És important mirar això i treure tots els usuaris que no hi han de tenir accés d'escriptura, i deixar únicament els que sí que en necessiten.

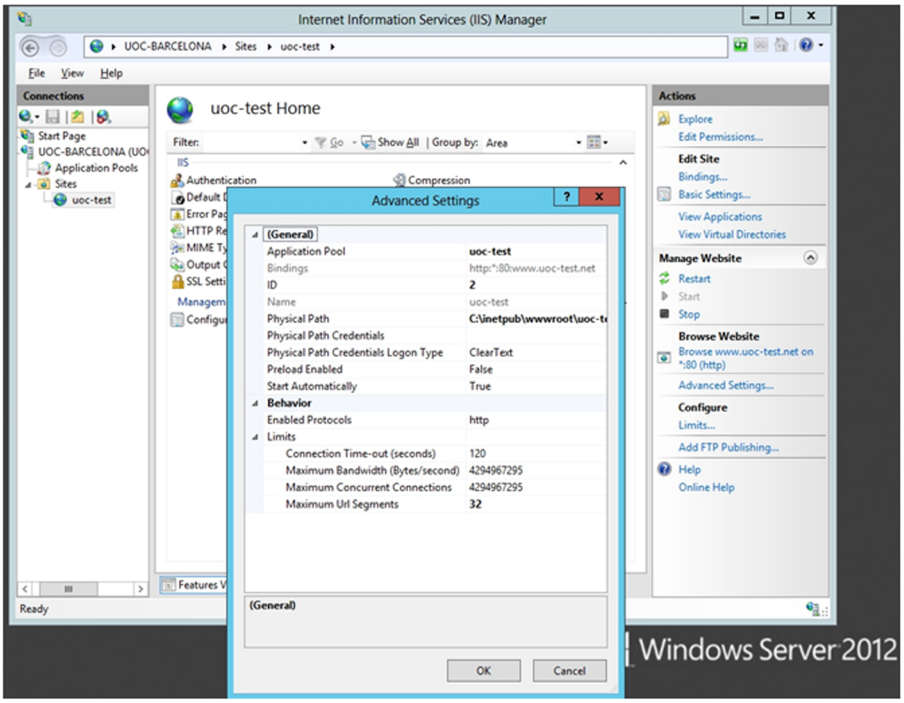
Permisos per defecte



Una vegada tenim el lloc web configurat inicialment, podem afegir carpetes i fitxers dins de la carpeta arrel d'aquest lloc; aquestes carpetes i aquests fitxers es poden explorar des de la llista de l'esquerra i el contingut que tenen es pot visualitzar a mà dreta. També hi podem afegir directoris virtuals, que són en altres carpetes del disc dur (no dins de la carpeta arrel) o en altres equips. L'assistent demana un àlies per identificar el directori virtual, la localització concreta i els permisos que té assignats; s'ha de tenir cura amb aquests permisos, ja que són directoris externs a l'arbre del lloc web.

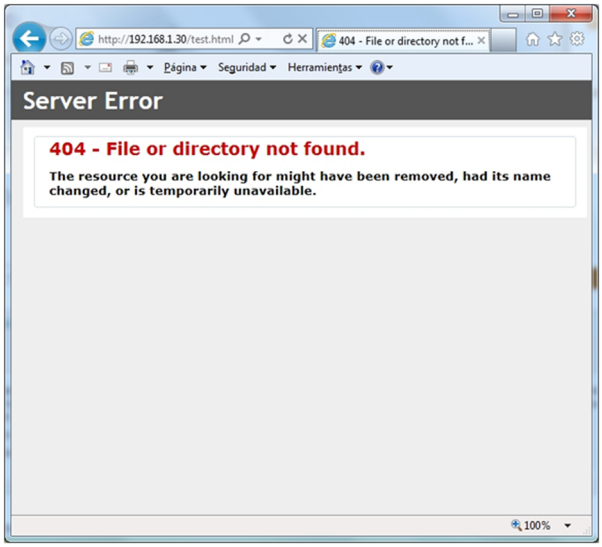
Podem modificar també la configuració avançada, on es pot modificar l'amplada de banda que es vol oferir, el nombre màxim de connexions o el *time-out*.

Configuracions avançades



Un dels aspectes que s'han de tenir en compte és la informació que es dóna de manera automàtica als atacants. És important minimitzar o fins i tot anul·lar la versió del sistema operatiu i de l'IIS i canviar els missatges d'error que retornen els servidors web i les bases de dades. Per tant, es pot canviar la pàgina dels errors perquè siguin les que vénen per defecte.

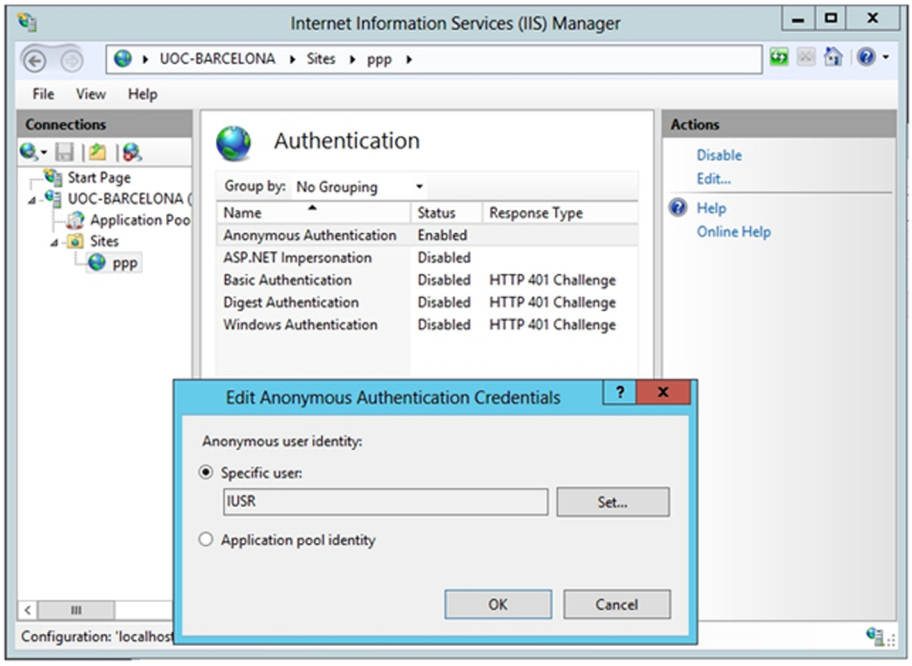
Pàgina per defecte de l'error no trobat (*not found*)



4.3.2. Mecanismes d'autenticació

Hi ha diversos mètodes d'autenticació en l'IIS i els trobarem en el rol de l'IIS, dins la branca de seguretat en el moment d'instal·lar el rol. Podem canviar el mecanisme d'autenticació d'un element des de la finestra del lloc web, en l'enllaç d'autenticació.

Configuració d'accés a la pàgina web



Com podem veure, és important que l'usuari que executa les pàgines web dels usuaris anònims, és a dir, dels que no es validen en el servidor, ho faci amb l'usuari IUSR del sistema, ja que té únicament privilegis de lectura i execució de pàgines web i no té cap privilegi sobre el sistema. Això donarà més seguretat al sistema en el cas d'explotar una vulnerabilitat en el servidor IIS.

L'autenticació anònima proporciona accés a les àrees públiques del lloc web sense sol·licitar el nom d'usuari ni la contrasenya. S'assigna l'usuari al compte **IUSR**, que està inclòs en el grup de convidats, sotmès a unes restriccions de seguretat imposades pels permisos d'accés als directoris i carpetes del web. Si fem clic sobre "Modificar", podem modificar el compte que s'associa a l'usuari anònim (per defecte, IUSR). Si no és necessari l'ús de l'accés anònim, es recomana desactivar-lo, ja que és un punt d'entrada al sistema que pot provocar problemes de seguretat; això farà, però, que la pàgina web no sigui accessible si no es disposa d'un usuari vàlid del sistema.

També es disposa d'altres mètodes d'autenticació:

- Autenticació bàsica. Se sol·licita un nom d'usuari i una contrasenya abans d'entrar al web. L'usuari i la contrasenya han de correspondre a un compte del Windows vàlid, és a dir, que només podran accedir al web els usuaris interns del sistema. Les contrasenyes es transmeten per la Xarxa sense xifrar, de manera que no es recomana aquest mecanisme tret que sigui necessari. En aquest cas, es recomana acompanyar aquest tipus d'autenticació amb el nivell de capa de sòcol segur (*secure sockets layer*, SSL).

- Autenticació implícita. Té les mateixes característiques que l'autenticació bàsica, però el nom d'usuari i la contrasenya es transmeten mitjançant un procés de resum (*hashing*). Aquest tipus d'autenticació només és disponible en servidors que pertanyen a un mateix domini.

- Autenticació integrada del Windows. És un mecanisme segur d'autenticació, ja que sí que es protegeix el nom d'usuari i la contrasenya. La informació d'autenticació d'un usuari és la utilitzada per a accedir a l'equip client des del qual s'inicia la petició. Aquest mecanisme només és compatible amb el Microsoft Explorer.

A més, també podem afegir filtres d'entrada i sortida al lloc web, fent que, per exemple, només es pugui accedir a les pàgines web des de dins de la mateixa institució, o de contingut, etc.

4.3.3. Configuració d'un lloc FTP

La configuració d'un lloc FTP en l'IIS és anàloga a la d'un lloc web. Es crea un lloc FTP inicial amb l'opció "Afegir lloc FTP nou" de l'element corresponent al servidor en l'arbre de l'esquerra. Alhora, podem crear carpetes o directoris virtuals dins del lloc FTP i configurar-ne les propietats, els permisos i els mecanismes d'autorització de la mateixa manera que en la configuració del lloc web.

Es pot forçar que l'FTP sigui a través del protocol SSH, i així tenir una mica més de seguretat. Però això fa que el servei deixi de ser anònim i que els usuaris necessitin un usuari en el servidor. Igual que abans, es poden afegir filtres i restriccions d'accés, i també l'amplada de banda que es vol donar a aquest servei.

4.3.4. Registre de l'IIS

Cada accés al servidor de pàgines web o al d'FTP crea una entrada en el registre d'esdeveniments de l'IIS, i també els accessos fallits, pàgines no trobades o usuaris no autoritzats. Per habilitar l'ús del registre obrim la finestra de propietats del lloc web i en la pestanya "Lloc web" seleccionem l'opció "Logs" i podrem configurar tot el que es necessiti: on es guarda, com, què es guarda, etc. Tot i que si s'obre un fitxer de registres de l'IIS es veurà que guarda cada una de les peticions que es fa al servidor, per tant fa molt difícil estudiar-los, es necessitaran eines especialitzades a extreure informació per a poder entendre ràpidament el que està passat amb el servidor IIS.

Seguretat en IIS

Han passat molts anys des que Microsoft va presentar la primera versió de l'IIS i ha millorat el producte a mesura que en presentava versions noves. De fet, la versió inclosa en el Windows 2000, l'IIS 5, va ser famosa pels cèlebres forats de seguretat que va tenir. Per tant, va tenir una gran repercussió gràcies als virus o cucs que explotaven les vulnerabilitats que presentava. Després de tot el que va passar, Microsoft va decidir començar de zero i desenvolupar l'IIS 6.0 des dels fonaments. No es va aprofitar ni una línia de codi. El resultats posteriors, amb les darreres versions més noves, és un producte que està preparat contra els atacs més comuns, que tolera fallades o caigudes del servei i que és autorecuperable, ja que els processos són capaços de reciclar-se per si mateixos.

A més, es va millorar l'arquitectura interna, de tal manera que, en cas d'atac i caiguda del servei IIS, el servidor no quedi compromès gràcies a un sistema d'aïllament d'aplicacions, basat en un conjunt d'aplicacions (*pools*) que s'executen amb *Worker Process* separades per aplicació.

S'ha d'anar amb compte amb els caràcters Unicode: com que es poden codificar els caràcters (/) i (..) dins de l'URL amb caràcters Unicode, quan el servidor IIS avaluï els permisos de les carpetes no hi trobarà cap problema, però en traduir els caràcters Unicode es poden formar adreces que permetin accedir a recursos del sistema.

Per tant, per a augmentar la seguretat d'IIS cal tenir en compte certes recomanacions, algunes de les quals ja hem comentat:

- Eliminar l'accés anònim si no és necessari o, si ho és, delimitar-lo als directoris virtuals que ho requereixin.

- Utilitzar SSH amb l'autenticació.

- No instal·lar els serveis de l'IIS que no siguin necessaris.

- Eliminar els fitxers d'exemple que s'instal·len juntament amb l'IIS.

És molt important, sobretot, tenir el sistema actualitzat amb els últims pedaços que genera Microsoft a mesura que es troben fallades en la seguretat de l'IIS o, en general, del sistema. En el capítol següent tractarem amb més deteniment el tema de les actualitzacions.

4.4. Anàlisi de riscos i prevenció

4.4.1. Web

Durant els últims anys, qualsevol empresa que es vol donar a conèixer mínimament està connectada a Internet i té un lloc web, en el qual l'empresa tracta de mostrar una imatge corporativa o vendre algun producte. Si el que volem és que la gent conegui l'empresa (o institució), hem de permetre que el lloc web sigui accessible des de qualsevol banda. Això implica que els ports TCP on s'executa un servidor web no han de tenir cap mena de restricció d'accés. Ara bé, hem de ser conscients que el fet de tenir una porta oberta al món també vol dir tenir una porta oberta als possibles atacants. Hi ha molts atacs que ens poden arribar pels mateixos ports per on escolta el servidor web.

Els atacs de modificació dels llocs web són molt "vistosos", sobretot si es tracta d'una empresa més o menys gran o una entitat coneguda per molta gent. La notícia que un lloc web d'aquests ha estat piratejat corre per Internet en qüestió de minuts. Fins i tot, hi ha recopiladors de llocs web piratejats. En algun d'aquests casos, depenent de la importància de l'empresa, la notícia pot arribar fins i tot a la premsa i la imatge de l'empresa o entitat que ha estat atacada es pot veure greument danyada. Hi ha un altre tipus d'atacs que, encara que no són tan freqüents, resulten més perillosos. Aquest altre tipus d'atac no pretén modificar el lloc web, sinó que, mitjançant una vulnerabilitat del servidor, pretén accedir a la màquina, al servidor. La majoria d'aquests atacs tenen èxit a causa d'una configuració errònia o defectuosa del servidor, encara que també és possible que siguin deguts a vulnerabilitats (errors o *bugs*) del servidor mateix i, moltes vegades, corregibles actualitzant el sistema (*Windows Update*).

En les empreses grans, els servidors web solen ser molt complexos: alta disponibilitat, redundància de servidors, balanceig de càrrega, gestió de continguts dinàmics, etc. Aquests sistemes són complexos d'administrar i protegir i és possible que, si no es duu a terme de manera correcta, es produeixin errors de

configuració. Les empreses petites moltes vegades utilitzen un servidor simple que requereix molt poc manteniment i que és fàcil d'administrar, però, al seu torn, aquests sistemes simples no solen tenir sistemes de seguretat gaire complicats.

Qualsevol analitzador de vulnerabilitats que puguem executar contra els sistemes és capaç de mostrar molta informació que ens pot resultar molt útil a l'hora de reforçar la seguretat dels nostres servidors.

Un exemple d'execució d'un analitzador contra un servidor web és el següent:

```
root# nmap -sV 192.168.1.30
Starting Nmap 5.00 ( http://nmap.org ) at 19:24 CEST
Interesting ports on 192.168.1.30:
Not shown: 983 filtered ports
PORT   STATE SERVICE   VERSION
53/tcp  open domain    Microsoft DNS
80/tcp  open http     Microsoft IIS webserver 8.0
88/tcp  open kerberos-sec Microsoft Windows kerberos-sec
135/tcp  open msrpc    Microsoft Windows RPC
139/tcp  open netbios-ssn
389/tcp  open ldap
445/tcp  open netbios-ssn
464/tcp  open kpasswd5?
593/tcp  open ncacn_http  Microsoft Windows RPC over HTTP 1.0
636/tcp  open tcpwrapped
3268/tcp open ldap
3269/tcp open tcpwrapped
49154/tcp open msrpc    Microsoft Windows RPC
49155/tcp open msrpc    Microsoft Windows RPC
49157/tcp open ncacn_http  Microsoft Windows RPC over HTTP 1.0
49158/tcp open msrpc    Microsoft Windows RPC
49175/tcp open msrpc    Microsoft Windows RPC
MAC Address: 00:21:5A:FD:00:00 (Hewlett Packard)
Service Info: OS: Windows

Service detection performed. Please report any incorrect results at http://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 66.10 seconds
```

Com apreciem, els analitzadors ens poden mostrar les versions de les parts del servidor (*http*, *dns*, *ldap*, *ssl*, *php*, *perl*, etc.) que tenim instal·lades. Fins i tot ens poden informar de directoris que tenim oberts en el servidor web. Quan l'atacant aconsegueix aquesta informació, només ha de buscar entre les vulnerabilitats dels nostres serveis instal·lats.

Un primer pas per a evitar aquests problemes consisteix a eliminar del servidor qualsevol directori o interfície comuna de passarel·la (*common gateway interface*, CGI) que s'instal·li de manera automàtica (per defecte) en el servidor. Aquest tipus de directoris solen ser d'exemple o de documentació, que no són especialment crítics. El cas de les CGI és diferent, ja que algun d'aquests *scripts* pot arribar a obrir forats importants en el servidor.

El pas següent per a afegir seguretat al nostre servei és deshabilitar el *Directory Indexing* del nostre servidor. Aquest paràmetre permet fer una llista del contingut d'un directori en cas de no trobar un fitxer `index.html`. En principi, en el `DirectoryRoot` tan sols hi ha d'haver els fitxers que són imprescindibles per a visualitzar correctament les pàgines web del nostre servidor, però la majoria de vegades hi trobem arxius comprimits que són els codis font d'alguna aplicació que s'executa en la pàgina web. Si els atacants saben que hi ha aquests fitxers, els poden baixar i obtenir-ne una informació molt útil a l'hora d'atacar el servidor. La millor manera d'evitar aquest tipus d'atacs és que no hi hagi res que ens pugui comprometre sota el `DirectoryRoot`, però si hi ha alguna cosa (per necessitats del sistema), hi hem de treure els permisos de lectura (perquè no ho pugui llegir l'usuari que executa el servidor web) i deshabilitar el *Directory Indexing* perquè ningú no sàpiga que aquests fitxers són aquí.

El punt següent que s'ha de tenir en compte és l'usuari amb el qual s'executa el servidor web. És molt desaconsellable que aquest usuari sigui `root`. Hem de fer servir un usuari sense cap privilegi (per exemple, `www-data`). Per evitar que ens modifiquin els fitxers `.html`, l'usuari `www-data` tan sols ha de tenir els privilegis de lectura i execució (només quan sigui necessari) sobre tots els fitxers que hi ha per sota del `DocumentRoot`, d'aquesta manera no podrà reescriure el contingut.

Per acabar, és recomanable fer servir el protocol HTTPS per a dur a terme les tasques d'autenticació d'usuaris i d'accés a informació confidencial per la Xarxa.

4.4.2. FTP

L'FTP és una eina de transferència de fitxers molt pràctica. Hem de saber, però, en quins entorns podem utilitzar aquesta aplicació. Si volem oferir un servei de baixada d'algun tipus de programari, l'FTP és la nostra eina, ja que ho fem amb l'FTP anònim. El client de baixada és el mateix navegador web, cosa que ofereix molta flexibilitat.

Si volem que els nostres usuaris es puguin connectar des de casa per accedir al seu espai de disc i baixar-ne els documents, llavors l'FTP no és l'eina que hem de fer servir, malgrat que satisfà les necessitats dels usuaris, ja que no té cap mena de xifratge del canal; per tant, l'autenticació dels usuaris i la transferència de fitxers es fa en pla, és a dir, sense xifrar.

Avui dia trobem servidors d'FTP que xifren el canal de comunicacions. El protocol que xifra la comunicació s'anomena *SFTP*. Els servidors d'SSH contenen un subsistema que permet fer transferències de fitxers.

A més, l'FTP és, tal com hem comentat, un protocol degà dins de les comunicacions TCP/IP, i això implica que és molt conegut i, per tant, també està molt estudiat. Els servidors antics d'FTP tenen molts forats per a entrar a la màquina. Per sort, les versions més modernes d'aquests servidors ofereixen mesures de seguretat molt més bones.

És important que, malgrat que l'FTP permet transferències en tots dos sentits, només permetem baixar fitxers del servidor. És potencialment molt perillós permetre escriure en el servidor, ja que no sabem qui hi ha a l'altra banda de la comunicació i, per tant, no sabem quines intencions té.

5. Protecció de ports

Al llarg d'aquest material hem parlat molt sobre els ports TCP. En aquesta secció veurem com hem de tancar determinats ports, com els hem d'obrir i com els hem de protegir.

Abans, però, de començar-ne a parlar, hem de saber quins ports tenim oberts en la nostra màquina. Hi ha certs ports que han d'estar oberts, i són els que coincideixen amb els serveis que oferim. En algunes versions de sistemes operatius, però, tenim més ports oberts. Com podem saber quins ports tenim oberts?

5.1. Protecció de ports en el GNU/Linux

En un mòdul anterior hem explicat com s'ha d'instal·lar l'eina de comprovació nmap. Per veure quins ports tenim oberts, hem d'executar l'ordre:

```
root# nmap host -sV
```

Com veiem en la figura següent, aquesta instrucció ens mostra els ports que tenim oberts i quina aplicació hi ha associada a aquest port.

```
root# nmap -sV localhost

Starting Nmap 5.00 ( http://nmap.org ) at 2013-04-01 19:58 CEST
Interesting ports on localhost (127.0.0.1):
Not shown: 989 closed ports
PORT   STATE SERVICE VERSION
21/tcp  open ftp  vsftp
25/tcp  open smtp   Postfix smtpd
80/tcp  open http   Apache httpd 2.2.16 ((Debian))
110/tcp open pop3   Courier pop3d
111/tcp open rpcbind
143/tcp open imap   Courier Imapd (released 2010)
443/tcp open ssl/http Apache httpd 2.2.16 ((Debian))
631/tcp open ipp   CUPS 1.4
993/tcp open ssl/imap Courier Imapd (released 2010)
995/tcp open ssl/pop3 Courier pop3d
3306/tcp open mysql  MySQL 5.1.66-0+squeeze1

Service detection performed. Please report any incorrect results at http://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 23.68 seconds
```

Per tancar un port, normalment, hem de parar el servei que hi ha escoltant en aquest port. La majoria dels serveis són a `/etc/init.d`. Una vegada som en aquest directori, hem d'executar el fitxer que té el mateix nom que el servei que volem parar seguit de la paraula `stop`. Per exemple, si volem parar `portmap` –un servei encarregat d'escoltar les crides a procediments remots o *remote procedure calls* (RPC) que es fa servir, entre altres coses, per a serveis de sistemes d'arxius de xarxa o *network file system* (NFS) i sistemes d'informació de xarxa o *network information service* (NIS)–, hem de fer el següent:

```
root# /etc/init.d/portmap stop
```

Per obrir un port hem d'engegar el servidor que s'executa en aquest port. Per fer-ho, seguim els mateixos passos que per parar-lo, però executem el fitxer seguit de la paraula `start`:

```
root# /etc/init.d/portmap start
```

Els ports, però, no s'obren i es tanquen tots d'aquesta manera. Hi ha determinats ports que es controlen des del superservidor i per tant no es poden modificar ni des del compte de superusuari.

5.1.1. Xinetd

En la majoria de les versions actuals del GNU/Linux, entre els paquets que tenen disponibles hi ha el `Xinetd`. D'aquesta aplicació també se'n diu *superservidor*. La funció principal d'aquesta aplicació consisteix a escoltar molts ports i, quan sent una petició per a un d'aquests ports, despertar el dimoni que hi està associat; d'aquesta manera, tenim una sola eina que escolta tots els ports en lloc de tenir una aplicació a cada port. El `Xinetd`, a diferència del seu antecessor (`Inetd`), té per defecte tots els ports tancats. Si tenim instal·lat l'`Inetd`, és molt recomanable (per motius de seguretat interna de l'aplicació) que actualitzem el superservidor en la versió `Xinetd`.

Si volem instal·lar el Xinetd, hem d'executar l'ordre següent:

```
root# apt-get install xinetd
```

Per a configurar aquest servei s'ha d'editar el fitxer `/etc/xinetd.conf`. Aquest fitxer indica quins serveis arrenquen quan s'engega el servidor. La configuració de cadascun dels serveis es troba en el directori `/etc/xinetd.d`.

Si es vol habilitar un servei, s'ha d'editar el fitxer corresponent i modificar el paràmetre següent:

```
disable = yes
```

I modificar-lo d'una de les maneres següents:

```
disable = no
# disable = yes
```

Un fitxer de configuració de serveis (el de telnet, per exemple) té un aspecte semblant al següent:

```
service telnet
{
flags = REUSE socket_type = stream wait = no
user = root
server = /usr/sbin/in.telnetd log_on_failure += USERID disable = no
}
```

Una vegada hem fet els canvis en els fitxers, hem de reiniciar el dimoni `Xinetd`; per fer-ho, executem l'ordre següent:

```
root# /etc/init.d/xinetd restart
```

5.1.2. Restricció de ports

A més d'obrir els ports, hi podem restringir l'accés. Aquest accés es controla amb els fitxers `/etc/hosts.allow` i `/etc/hosts.deny`. En el primer s'especifiquen to tes les màquines que hi tenen accés, i en el segon es posen totes les màquines que no hi han de tenir accés. Aquests dos fitxers ens permeten dissenyar diferents "polítiques" d'accés a les màquines: des de polítiques molt permissives (permetre-ho tot i denegar algun servei), fins a polítiques molt restrictives (denegar-ho tot i permetre només alguns serveis).

Per a fer servir aquests fitxers cal que els servidors (de les diferents aplicacions) siguin compatibles amb TCP Wrapper. TCP Wrapper és el servei que verifica l'origen de les connexions amb la seva base de dades `/etc/hosts.allow` (equips autoritzats) i `/etc/hosts.deny` (equips als quals es denega la connexió).

La sintaxi de tots dos fitxers és la mateixa: `Servei: màquines`.

El servei és el nom del dimoni o del servidor (`sshd`, `telnet`, `FTP`, etc.), mentre que el nom de la màquina pot ser una adreça IP, el nom o un rang de xarxa. Si volem especificar més d'una màquina, han de sortir separades per comes.

Aquests fitxers admeten també un nombre reduït de paraules:

1) `All`: tot (es pot referir tant a servei com a màquines).

2) `Local`: les màquines del domini (en el nom no ha de sortir un punt).

3) `Paranoid`: només les màquines donades d'alta en el DNS.

4) `Known`: màquines conegudes (són en el fitxer `/etc/hosts`).

5) `Unkown`: màquines desconegudes.

Un exemple de política molt restrictiva és el següent:

```
Host.allow
SSHD: LOCAL

Host.deny
ALL: ALL
```

Amb aquesta configuració només permetem accedir al servidor d'SSH a les màquines locals.

Actualment, hi ha poques màquines que configurin l'accés als seus serveis mitjançant aquests dos fitxers. La majoria dels controls d'accés es fan amb tallafocs, ja que permeten tenir més versatilitat en les configuracions de xarxes i el tipus d'accés.

Segons quina sigui la nostra configuració de la xarxa, podem tenir un tallafoc que permeti l'accés a la xarxa a determinades màquines i després podem tenir un segon nivell de protecció en cadascun dels servidors. Aquest segon nivell s'implementa amb l'aplicació `iptables` (en la majoria dels casos, aquesta aplicació es fa servir per a implantar un tallafoc programari) per a definir, entre totes les màquines de la xarxa, quines tenen accés a determinats serveis.

També hi ha l'opció d'implementar el port *knocking*, en què tots els ports estan amagats darrere d'una seqüència establerta de comunicació. Això vol dir que tindrem els diferents ports tancats des de fora, però seguirem tenint els serveis escoltant pel port correcte. Quan el port *knocking* rebi una seqüència vàlida de ports obrirà el port corresponent. Per a implementar aquesta tècnica cal instal·lar el dimoni `knockd` i configurar alguna de les crides perquè obri el port corresponent.

```
root# apt-get install knockd
```

Cal configurar-lo perquè es pugui fer servir. Tenim la configuració a `/etc/knockd.conf`, on tenim per defecte el protocol SSH configurat. Es pot veure que si arrenquem el servidor, el port del servei ssh es mostrarà tancat a menys

que es faci una crida prèvia als ports 7000, 8000 i 9000 en aquest ordre. I es mantindrà obert fins que es faci una altra seqüència amb els ports 9000, 8000 i 7000.

```
[options]
    UseSyslog

[openSSH]
    sequence  = 7000,8000,9000
    seq_timeout = 5
    command   = /sbin/iptables -A INPUT -s %IP% -p tcp --dport 22 -j ACCEPT
    tcpflags  = syn

[closeSSH]
    sequence  = 9000,8000,7000
    seq_timeout = 5
    command   = /sbin/iptables -D INPUT -s %IP% -p tcp --dport 22 -j ACCEPT
    tcpflags  = syn
```

Per a arrencar el servei de `knockd` cal editar el fitxer `/etc/default/knockd` i, tal com mostra, canviar el paràmetre a 1 perquè arrenqui el servei des de l'inici.

```
root# joe /etc/default/knockd
##################################################
#
# knockd's default file, for generic sys config
#
##################################################
# control if we start knockd at init or not
# 1 = start
# anything else = don't start
#
# PLEASE EDIT /etc/knockd.conf BEFORE ENABLING
START_KNOCKD=0

# command line options
#KNOCKD_OPTS="-i eth1"
```

Per poder obrir els ports hi ha aplicacions tant en GNU/Linux com en Windows que gestionen les crides als diferents ports per obrir el que realment es vol tenir operatiu. En GNU/Linux el mateix servei `knockd` té `knock`, que fa les crides, i en Windows existeix l'aplicació `KnockKnock`, que es pot configurar per a fer el mateix.

Tot i ser una bona tècnica per a amagar ports, a vegades resulta poc efectiva, ja que cal obrir abans els ports i a més és també molt fàcil deixar-los oberts. Per tant, s'ha d'anar amb compte amb aquesta tècnica i no deixar oberts els ports.

5.2. Protecció de ports en el Windows Server 2012

El servei web i el servei FTP de l'IIS són aplicacions que reben i envien dades per un determinat port TCP o UDP de la màquina. Hi ha molts altres serveis o aplicacions del mateix sistema o d'altres fabricants que utilitzen diferents ports de la màquina per a comunicar-se.

Com que un port és un punt d'entrada al sistema, és important desactivar els ports que no necessitarem en el servidor. Per a saber quins ports hi ha actius actualment en el sistema, podem utilitzar l'eina de consola d'ordres `netstat`. Per fer-ho, obrim una finestra de l'intèrpret d'ordres i hi escrivim el següent:

```
netstat
```

L'execució d'aquesta ordre ens mostra una llista de ports actius, el protocol (TCP o UDP) que utilitza i l'estat actual.

Una manera completa de configurar ports és utilitzant el tallafoc que ve amb el Windows Server 2012 o instal·lant un tallafoc maquinari o programari que permeti, a més de filtrar els ports, filtrar segons la IP origen de la petició i configurar també els accessos entrants i les peticions que surten del servidor mateix.