

Confidencialidade
Integridade
Dispoñibilidade
Vulnerabilidades

Adrián Gómez Lois

Contenido

1. Obxectivos	3
2. Confidencialidade: Sistema EFS de Windows en carpetas e ficheiros e certificados	3-8
3. Confidencialidade: Sistema PGP en Linux	9-15
4. Confidencialidade: Sistema BitLocker en Windows e desencriptación da información	15-17
5. Integridade: Utilidade SFC en Windows.....	18-19
6. Integridade: Escaneo con Rootkit Hunter en Linux.....	19-21
7. Integridade: Verificación de hashes MD5 en ficheiros	22
8. Dispoñibilidade: Análises con Nmap en Windows e Linux.....	23-24
9. Vulnerabilidades: MBSA en Windows.....	25-26
10. Vulnerabilidades: Nessus en Linux.....	27-31
11. Conclusións	31

1. Obxectivos

O obxectivo destas tarefas son entender e saber manexarse cas diferentes ferramentas que os sistemas operativos (ou empresas de terceiros) incorporan para asegurar os nosos datos.

Tanto a confidencialidade, integridade e posibles vulnerabilidades que esto supón para o usuario final.

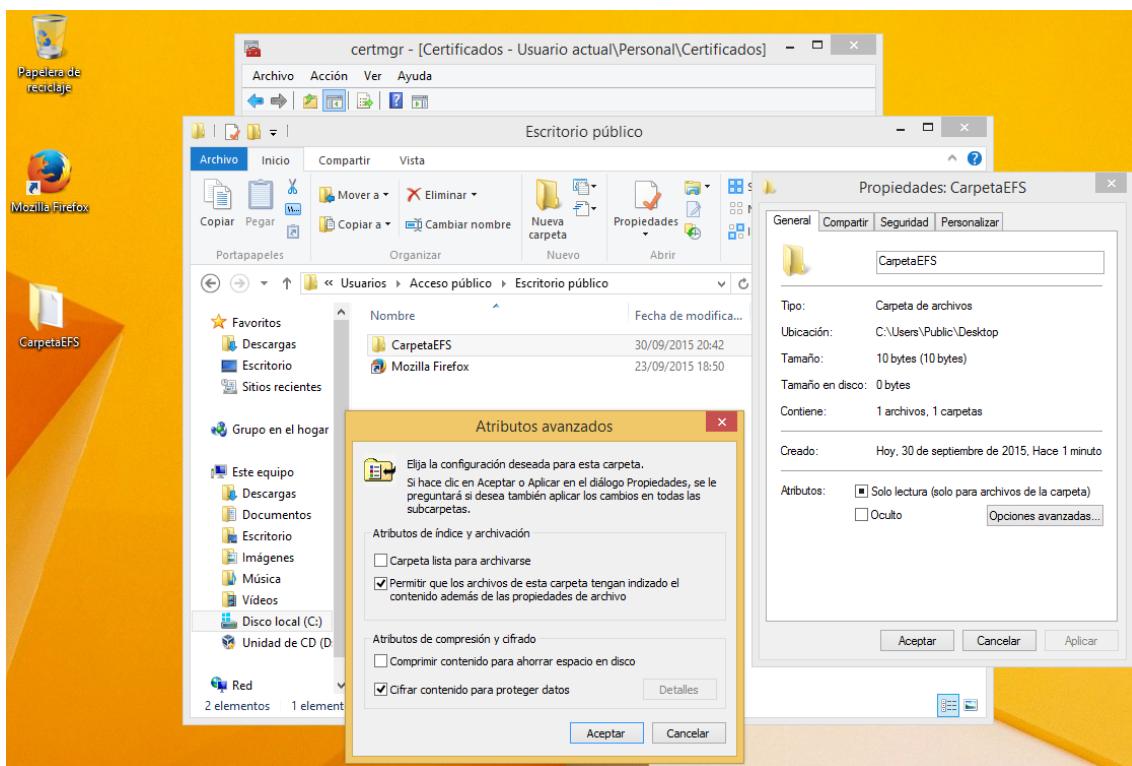
Saber que hay ferramentas e mecanismos para protexer a privacidade dos datos e os nosos sistemas.

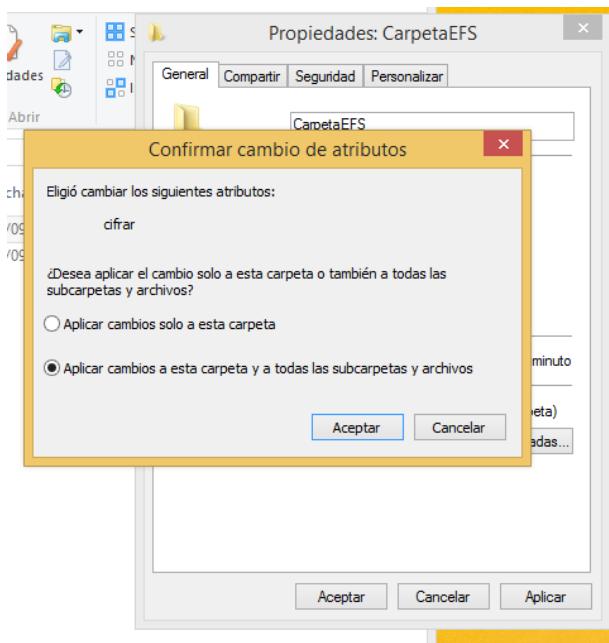
2. Confidencialidade: Sistema EFS de Windows en carpetas e ficheiros e certificados.

Nesta práctica en Windows cifraremos un cartafol con ficheiros con EFS (*Encrypting File System*).

Creamos un carpeta con un ficheiro de texto, sobre a carpeta raíz a cifrar pulsamos botón derecho > General > Opciones avanzadas > marcamos o checkbox “Cifrar contenido para proteger datos”. E aplicamos os cambios na carpeta, subcarpetas e ficheiros.

Automáticamente o cifrar un ficheiro con EFS en Windows, este crea un certificado de cifrado de ficheiros do usuario que cifrou o cartafol en cuestión.



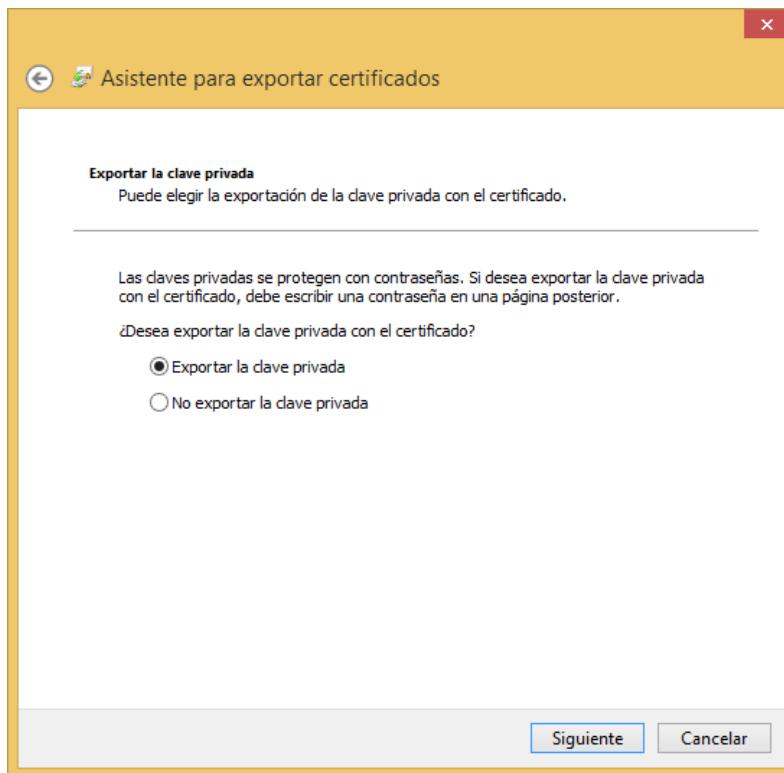


Unha vez cifrados veremos o **cartafol raíz de cor verde**, agora si queremos poder ver/editar este ficheiro desde calquera outro usuario ou equipo, teremos que exportar o certificado de clave privada.

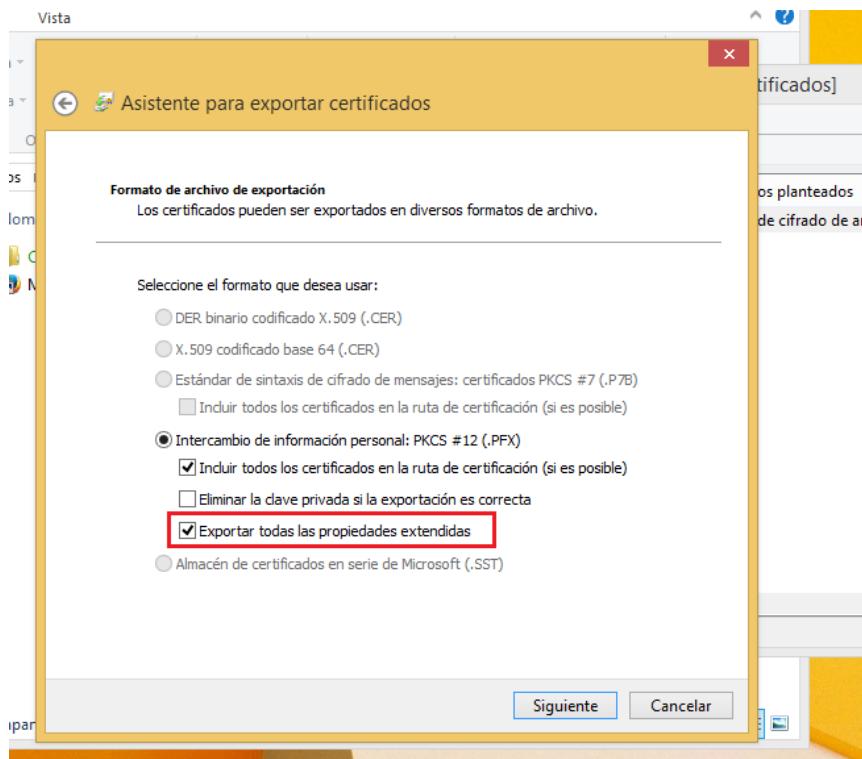
Para iso vamos a Microsoft Console (MSC) de Windows encarga de administrar os certificados instalados do usuario actual.

`certmgt.msc`

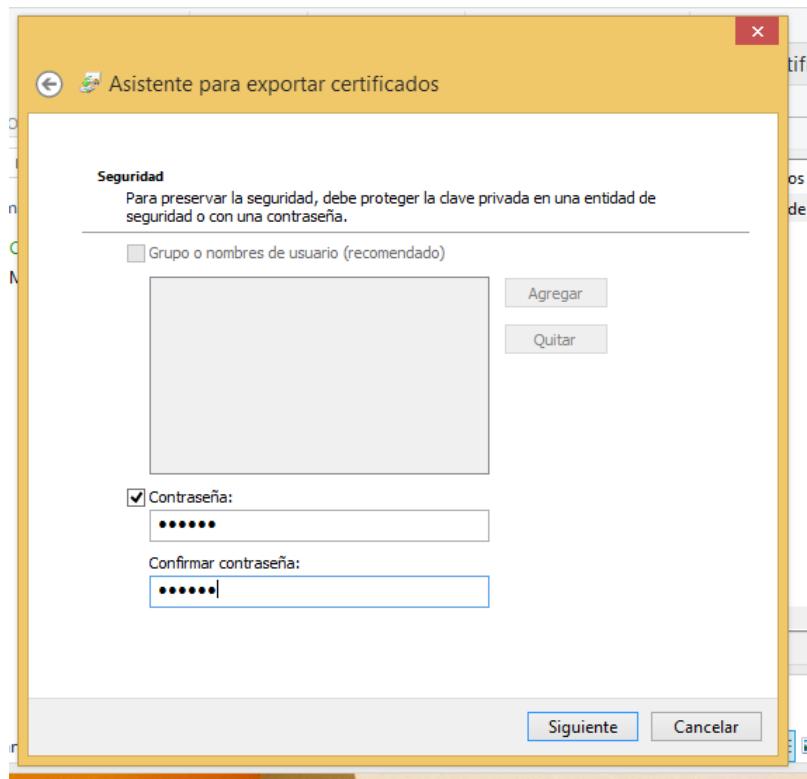
Unha vez ahí buscamos en “Personal” o certificado xenerado por Windows e o exportamos como chave privada.



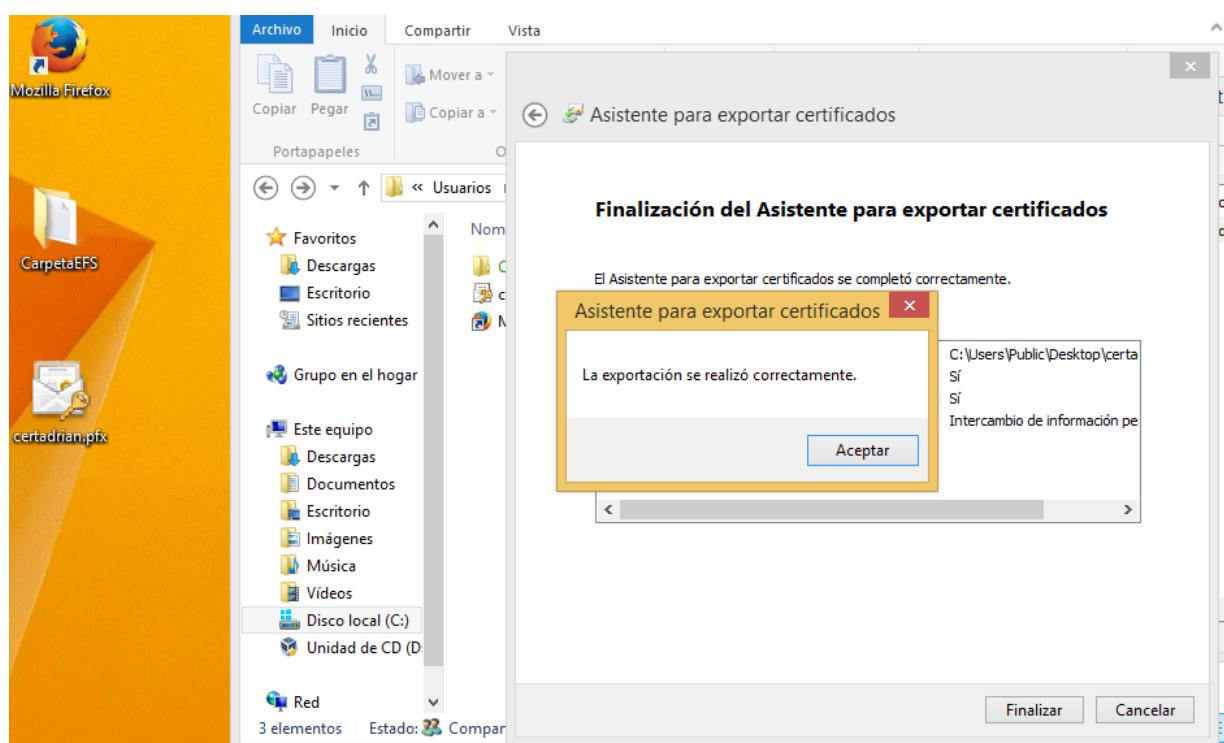
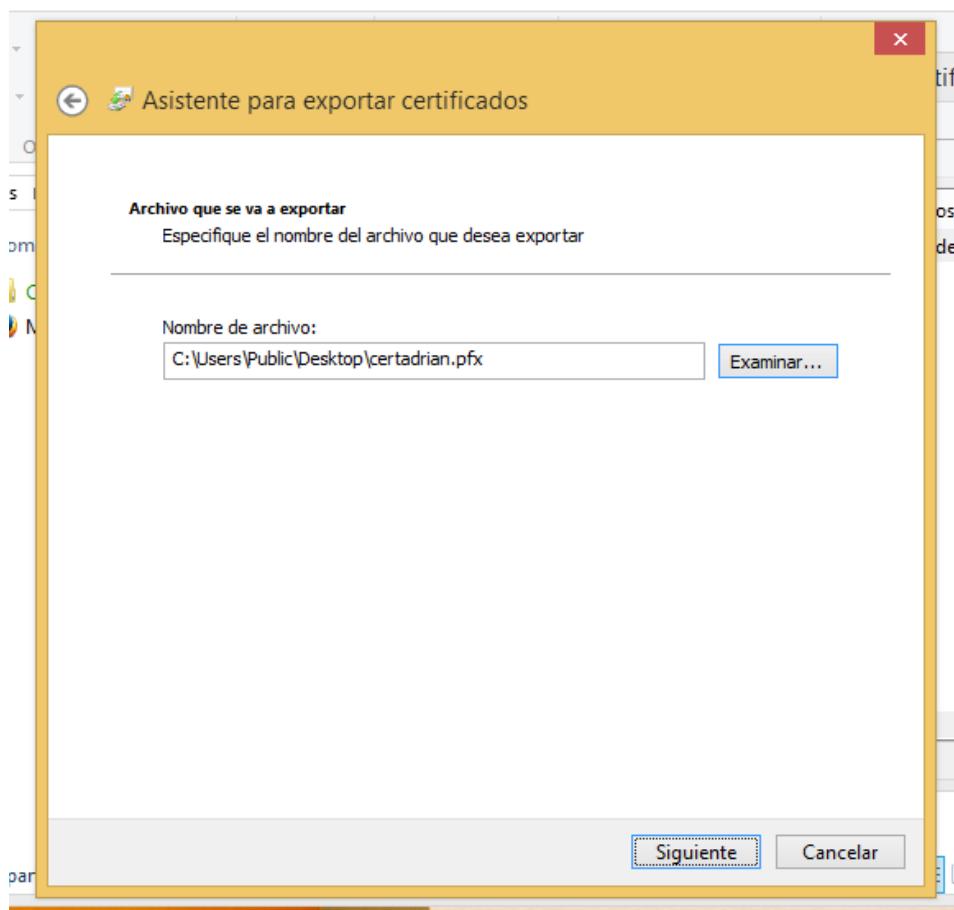
Neste punto é moi importante marcar o checkbox “Exportar todas las propiedades extendidas”.



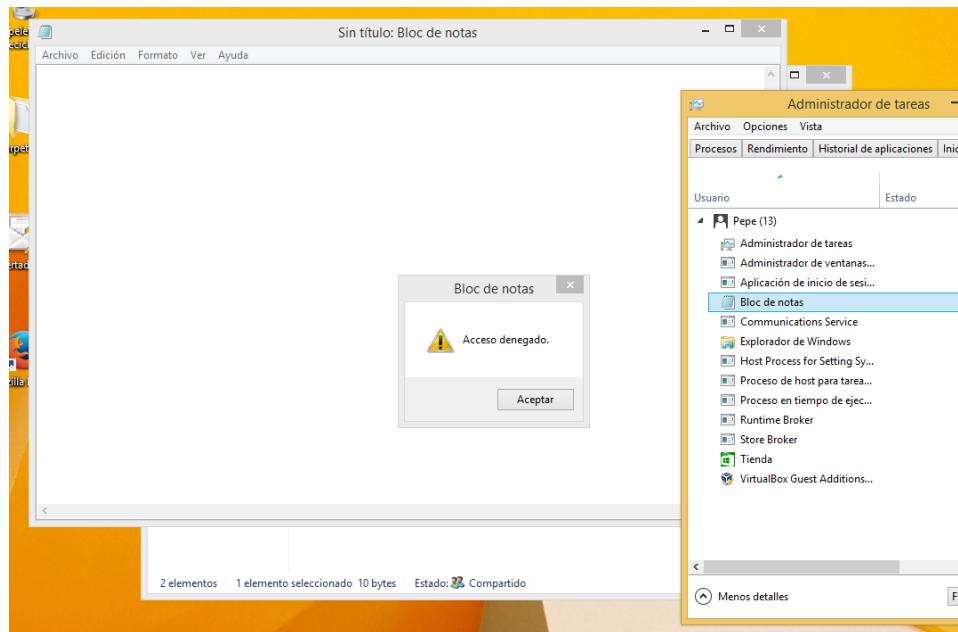
Establecemos unha contrasinal para o certificado de chave privada.



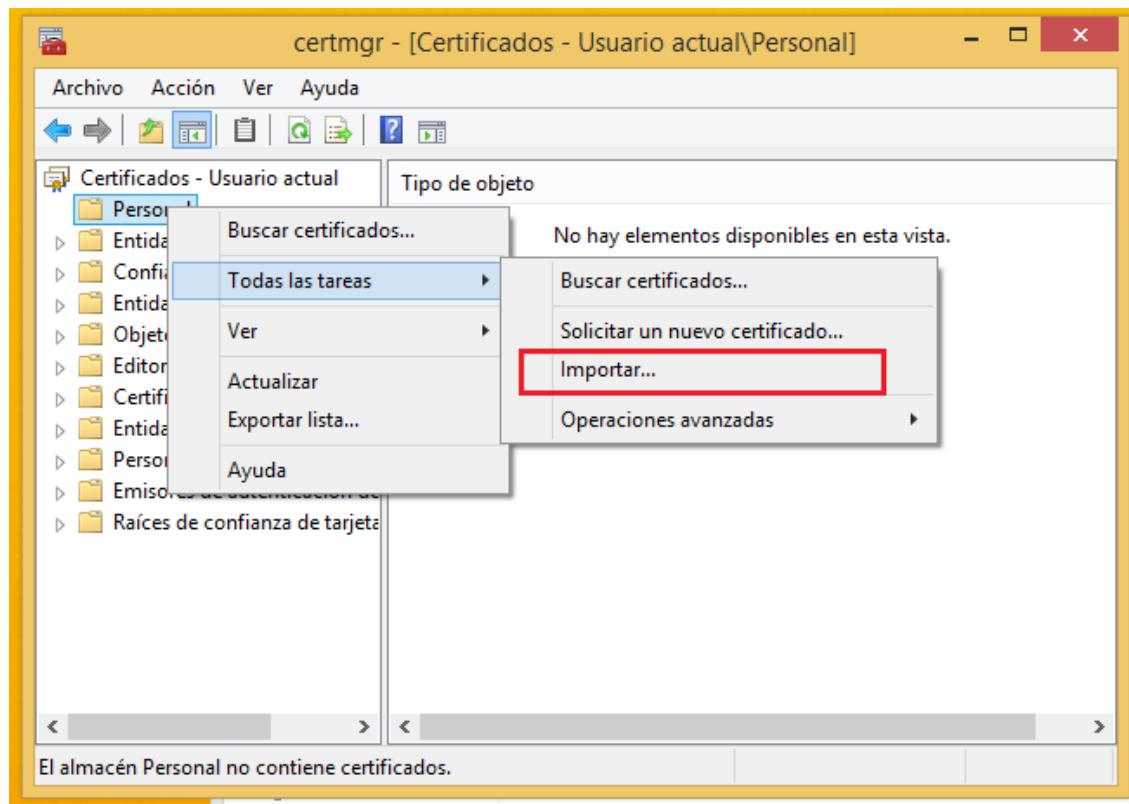
Asignamos unha ruta de exportación para o certificado tipo .pfx.



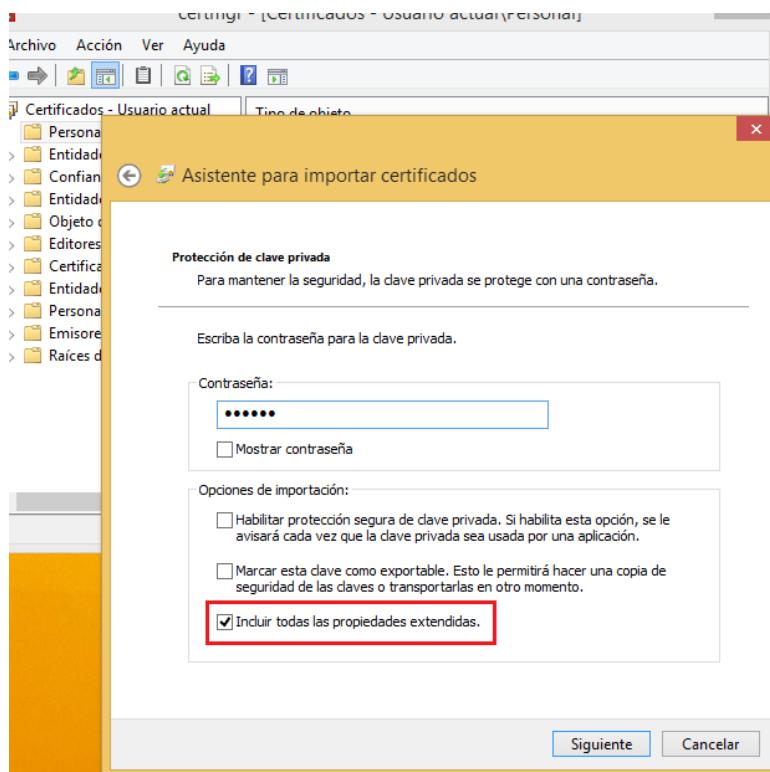
Para probar a seguridad de EFS, iniciamos sesión con outro usuario (previamente xa creado), e tentamos abrir o ficheiro creado no cartafol cifrado, o resultado e o máis normal é que non podamos abrilo.



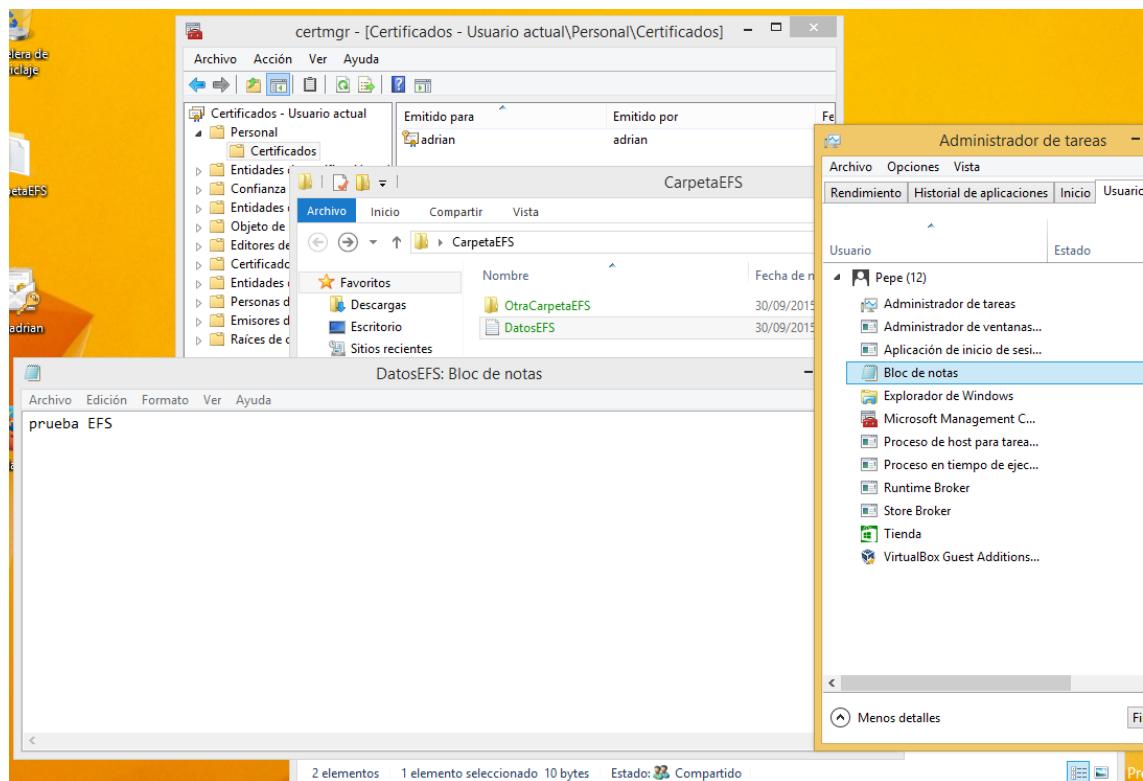
Dende este outro usuario non veremos o certificado que autoriza a autenticación para poder ver/editar o ficheiro anterior, polo que na mesma consola anterior msc (certmgr.msc). Importamos o certificado .pfx ca chave privada do usuario incial que cifrou o cartafol.



Cando no lo pida, no asistente de importación introducimos a contrasinal que establecimos para o certificado e moi importante marcamos o checkbox “*Incluir todas las propiedades extendidas*”.



Unha vez acabada a importación, veremos o certificado en “Personal” xa importado (creado co usuario “adrian”), e podemos ver como agora xa podemos ver/editar o ficheiro “DatosEFS” dende o usuario “Pepe”.

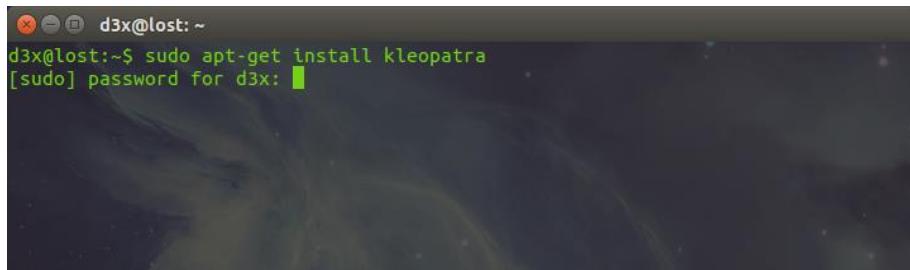


3. Confidencialidade: Sistema PGP en Linux.

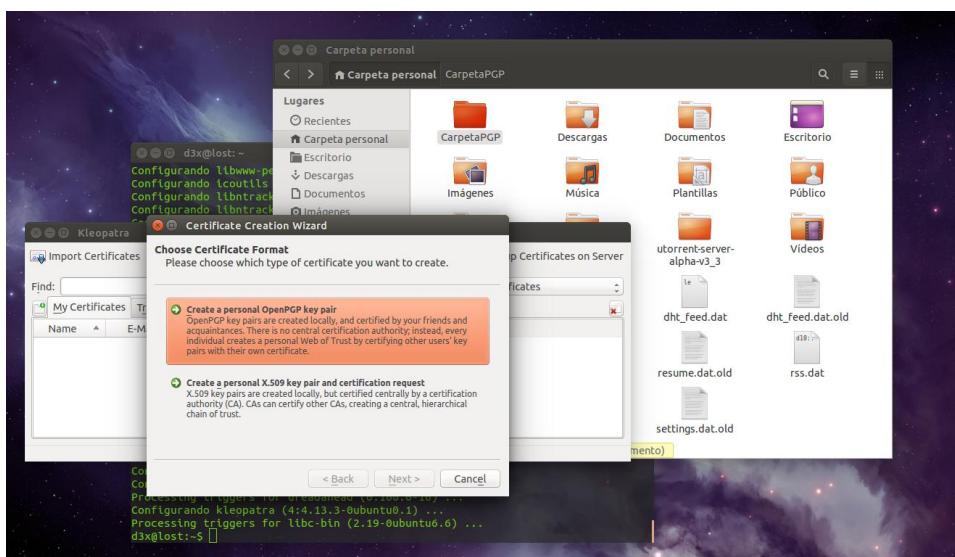
No meu caso para administración de certificados PGP (*Pretty Good Privacy*) fareño co software para Linux **Kleopatra**. Ainda que posteriormente faremos uso dos plugins de **seahorse** e **gpg** para importar a chave privada.

Instalamos Kleopatra, escribimos nunha terminal:

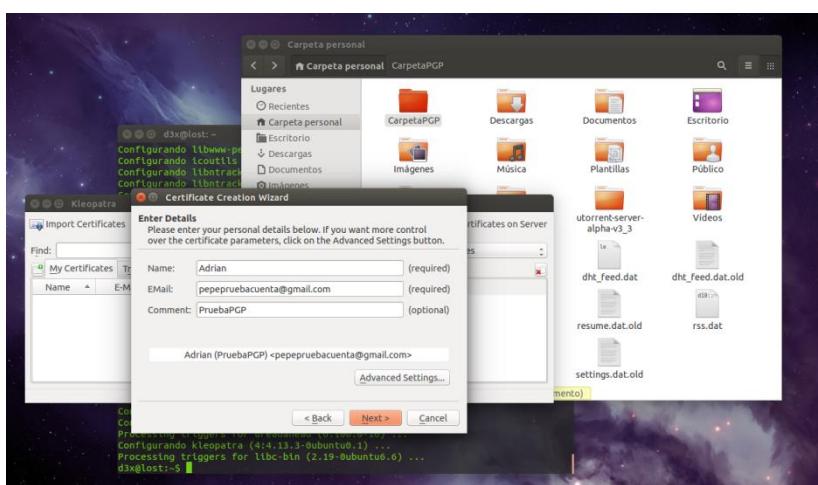
```
sudo apt-get install kleopatra
```

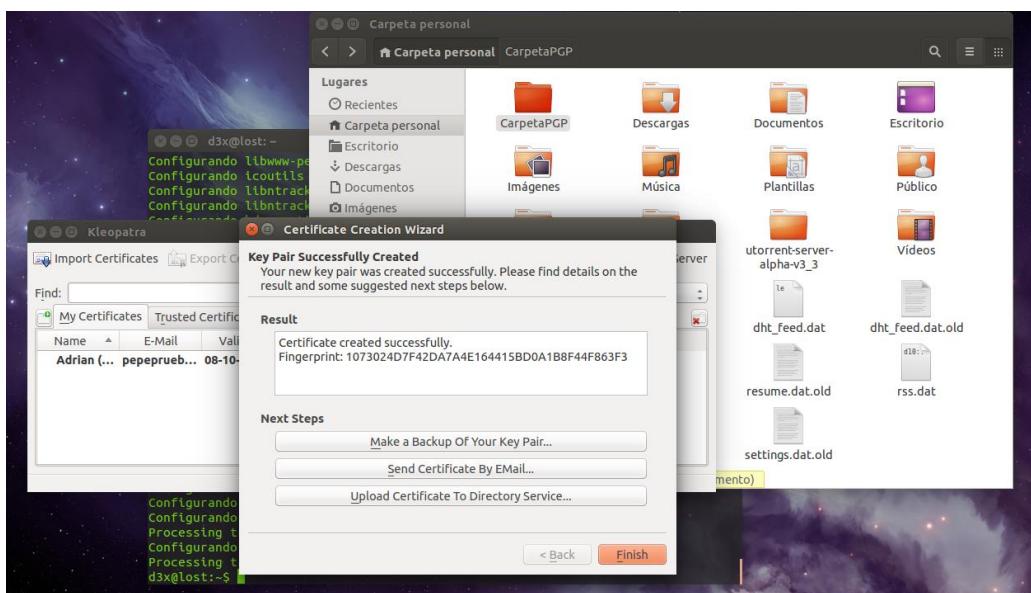
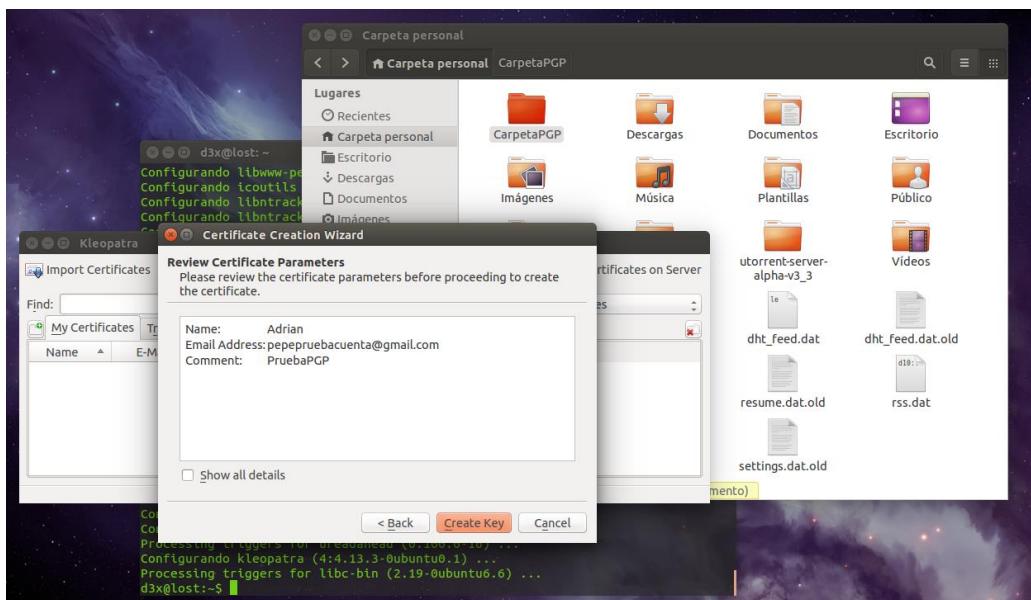


Unha vez acabe de instalar, abrimos Kleopatra e creamos un certificado persoal PGP de chaves asimétricas (chave pública e chave privada) co asistente de Kleopatra para xenerar certificados.



Introducimos o nome, email e un comentario (opcional).

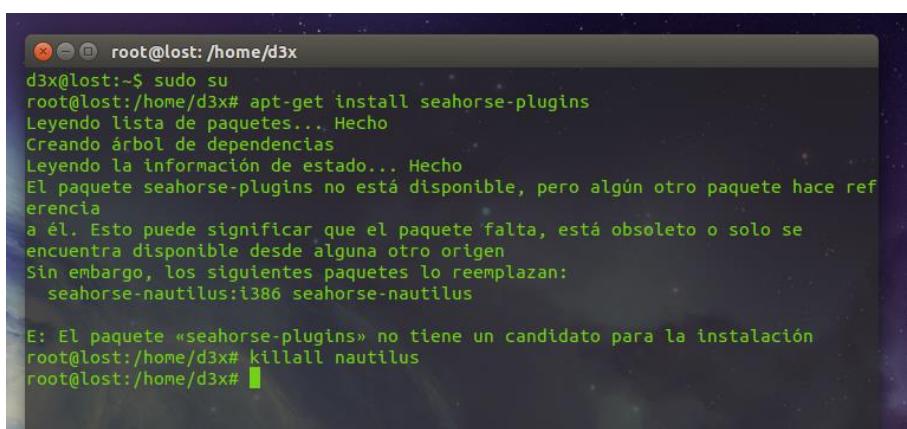




Agora temos que instalar o plugin para nautilus o cal permítenos cifrar ficheiros ca opción, “botón derecho > Cifrar”.

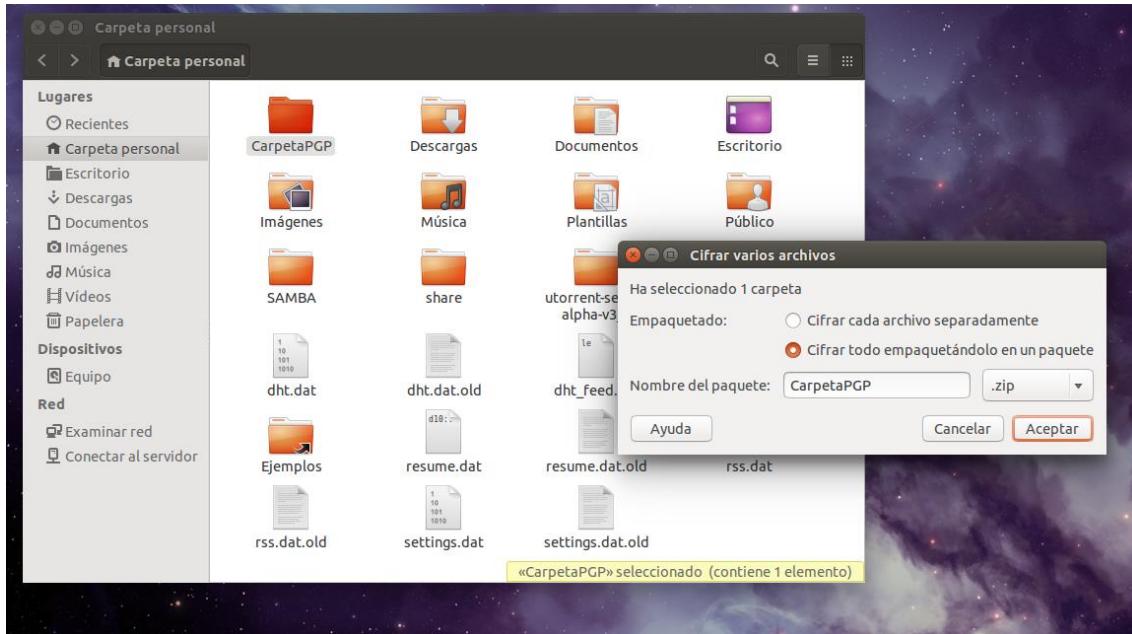
Para iso, escribimos nunha terminal:

```
sudo apt-get install seahorse-plugins
killall nautilus
```



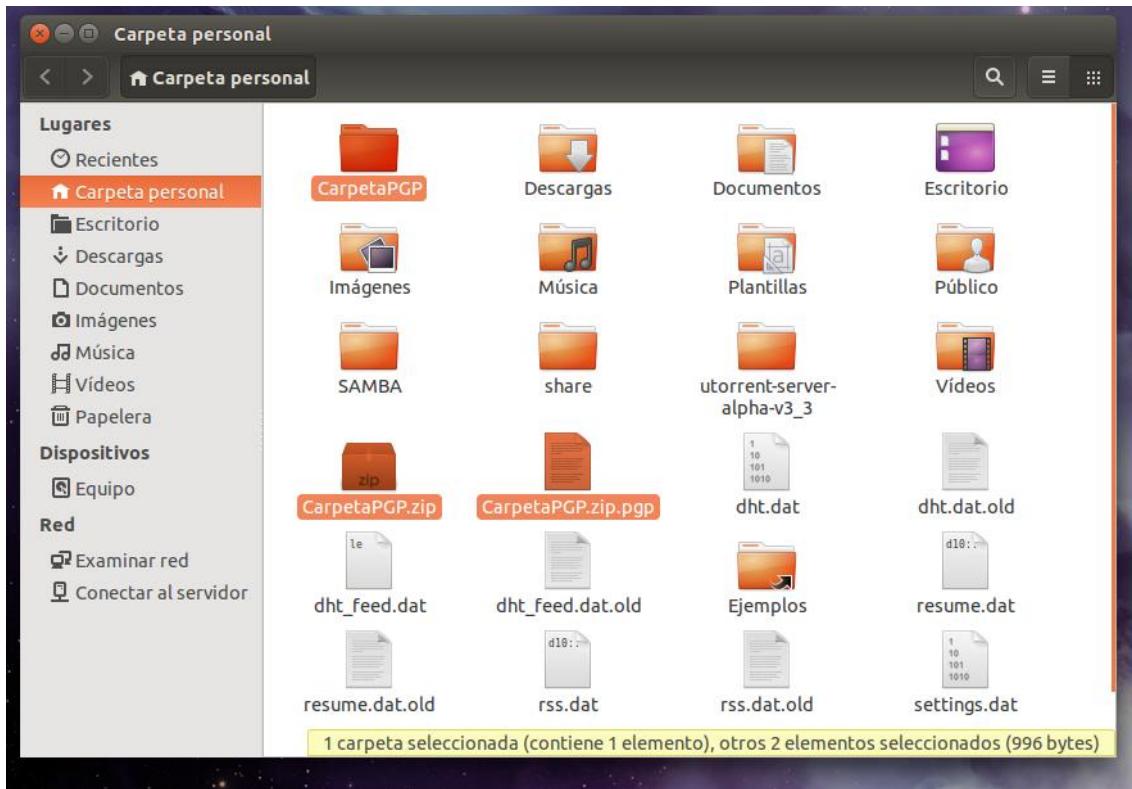
Cerramos sesión e volvemos iniciar para aplicar os cambios dos plugins de seahorse instalados en nautilus.

Ca opción cifrar, neste exemplo, ciframos a carpeta “CarpetaPGP” ca opción de empaquetala nun paquete .zip.

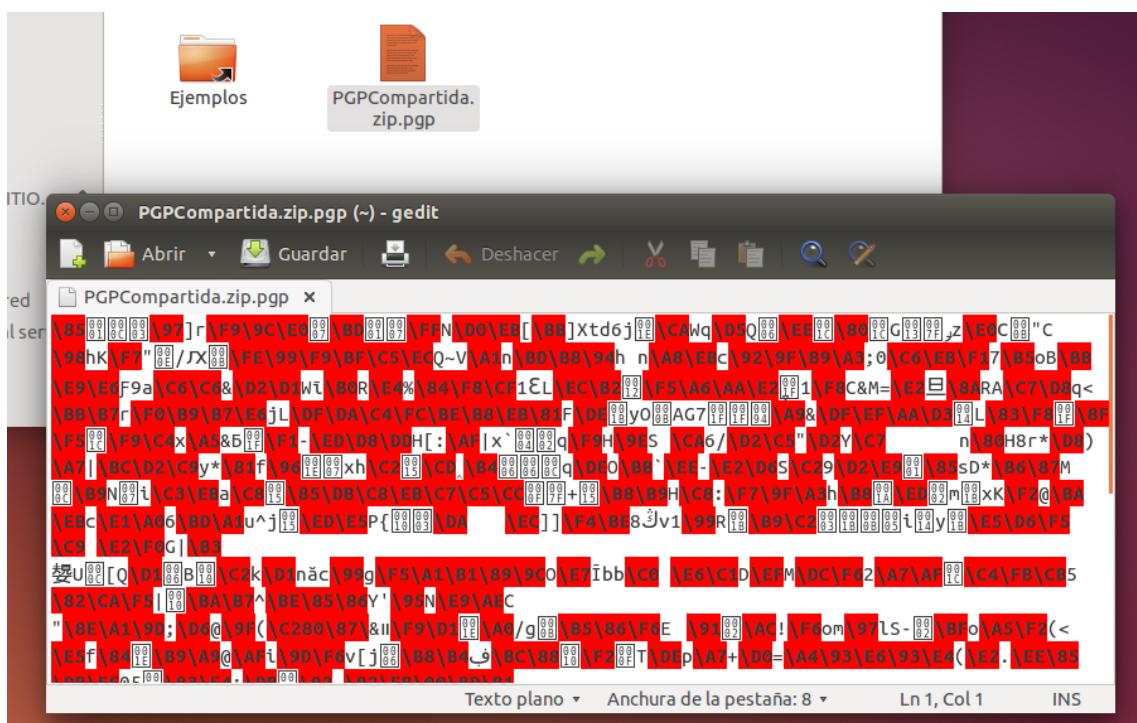


Unha vez acabe de cifrar a carpeta, esta xenera dous ficheiros máis a maiores. Una e a carpeta empaquetada nun .zip e outro e o propio cifrado empaquetado o .zip nun ficheiro cifrado .pgp.

Si queremos podemos borrar a carpeta original “CarpetaPGP” e o comprimido .zip. De modo que solo nos quedemos co ficherio .zip.pgp. Xa que este conten o comprimido .zip cifrado co certificado persoal creado anteriormente.

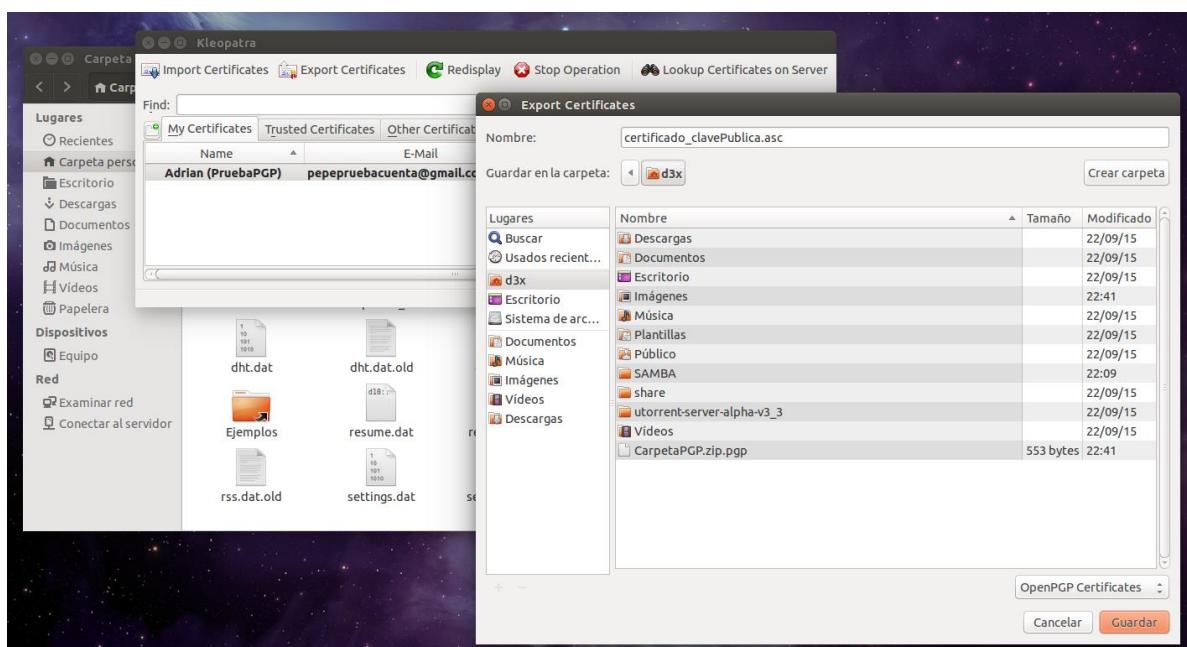


Se tentamos abrir con un gedit o ficheiro cifrado veremos o seguinte.

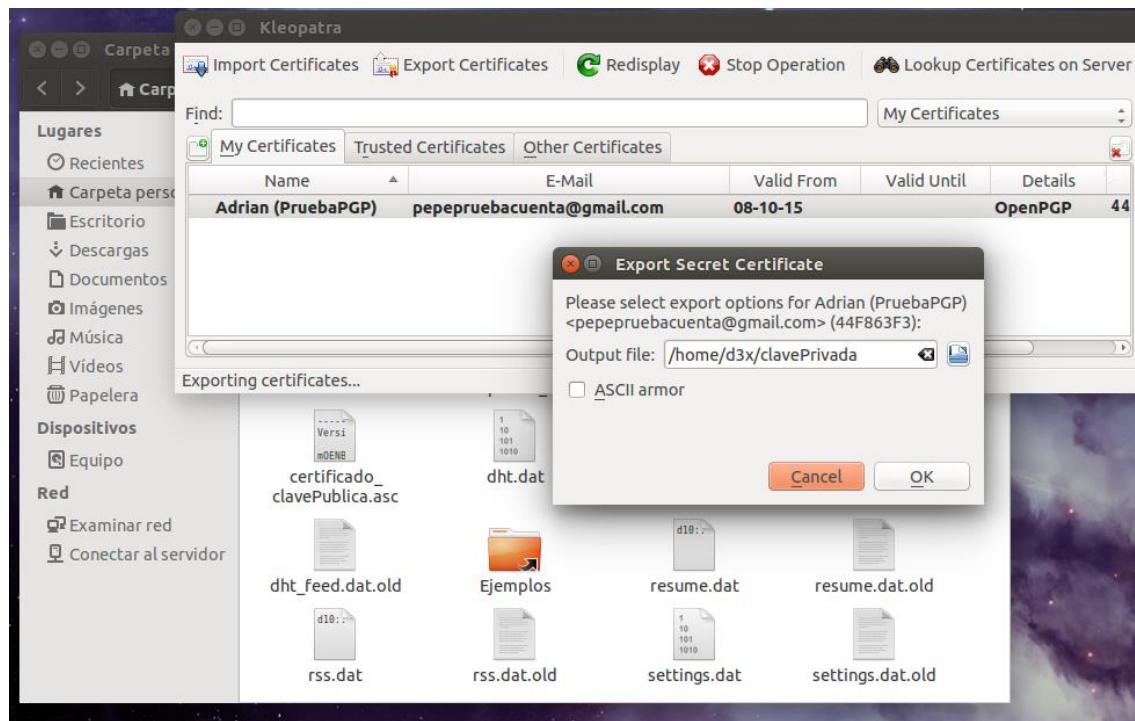


Neste xa tendremos o ficheiro cifrado, pero si queremos abrir este ficheiro con outro usuario ou noutro equipo teremos que levar o certificado persoal pgp, polo que temos que exportar o certificado para logo impórtalo no novo usuario ou equipo no queiramos visualizado ou editalo.

Polo que exportamos o certificado que contén a chave pública, como resultado xerará un ficheiro .asc. Neste caso chameíño “certificado_ClavePublica.asc”



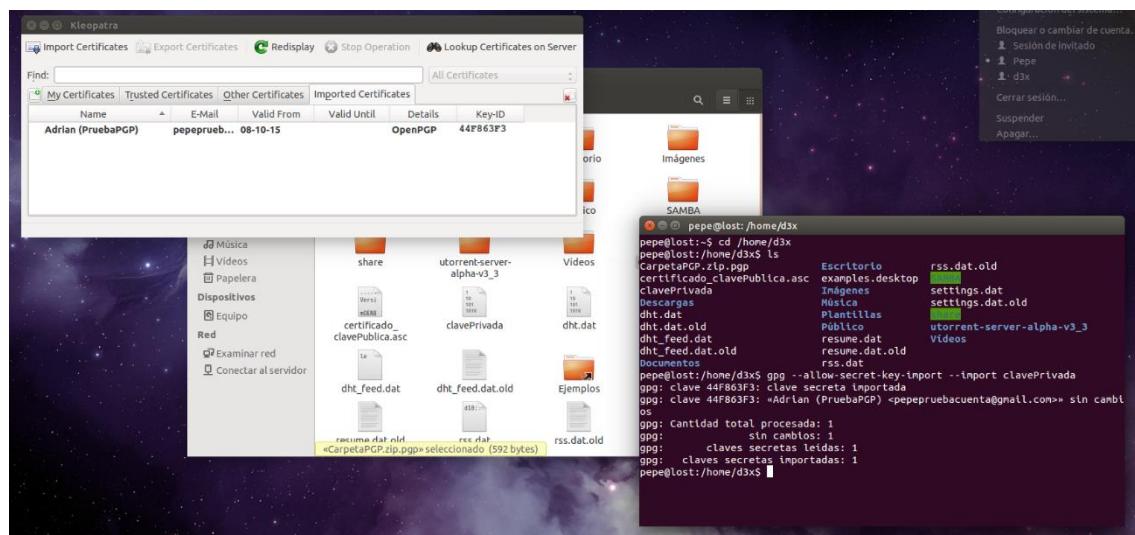
A continuación exportamos a chave privada (secret key). Neste caso chameino “clavePrivada”.



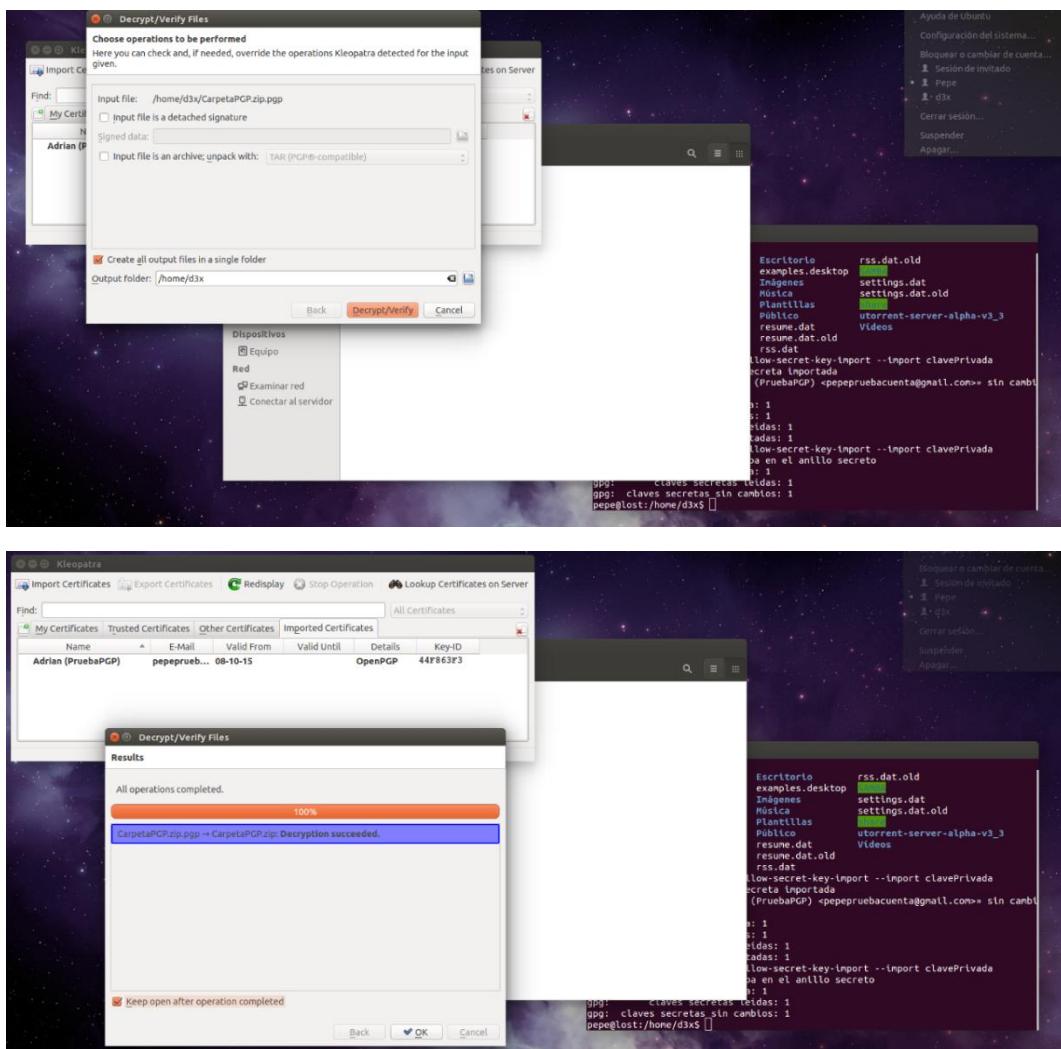
Agora iniciamos sesión con outro usuario (previamente xa creado), neste caso o usuario “Pepe”, neste usuario abrimos Kleopatra e importamos o certificado que contén a chave pública .asc.

E despois facendo uso da ferramenta gpg mediante liña de comandos importamos a chave privada.

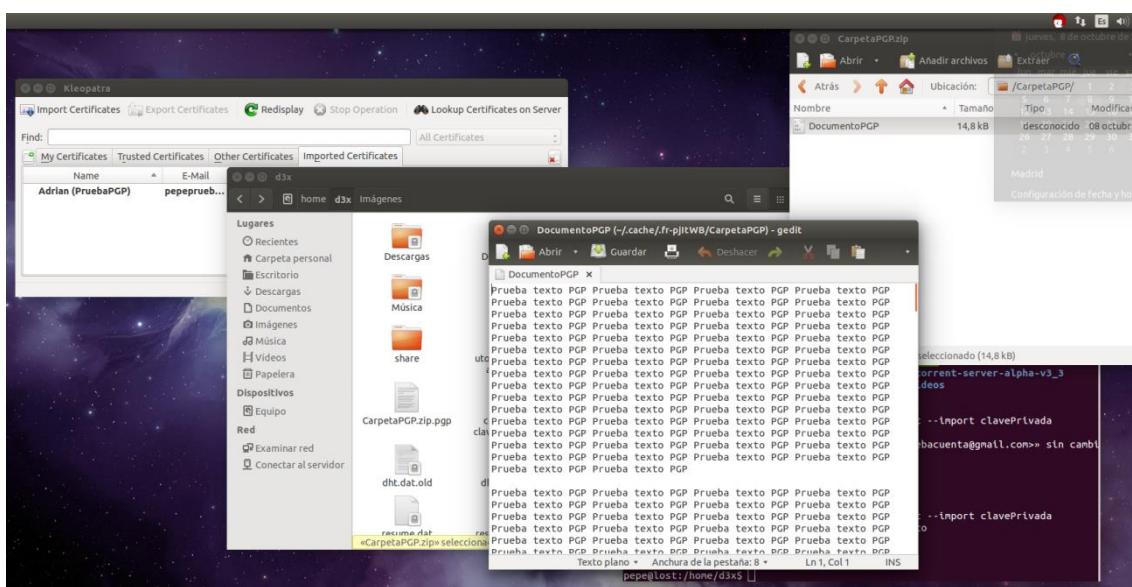
```
gpg --allow-secret-key-import --import clavePrivada
```



Por último desde Kleopatra iremos a opción de decrypt/verify files, na que seleccionaremos o ficheiro empaquetado e cifrado con PGP, neste caso CarpetaPGP.zip.pgp.



Unha vez descifrando o ficheiro desde o usuario "Pepe", xa podemos abrilo sen problemas podendo visualizalo e editalo.

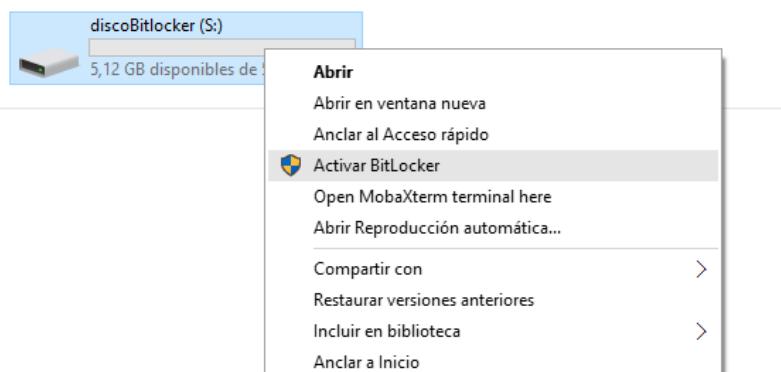


Conclusións: Se comparamos nas prácticas anteriores os cífrados EFS (Windows) co cífrado PGP (Linux, ainda que tamén pódese usar en Windows), chegamos a conclusión de que e outro tipo de filosofía de cífrado de ficheiros, a idea de EFS e que tendo o certificado podemos descifrar e editar o mesmo ficheiro sin realizar ninguna outra modificación. Sin embargo a filosofía de PGP e que temos que levar o certificado con clave pública e privada para descifralo e solo levaremos un ficheiro empaquetado comprimido cifrado con pgp. Cada vez que descifremos o ficheiro para editalo teremos que volvelo empaquetar de novo.

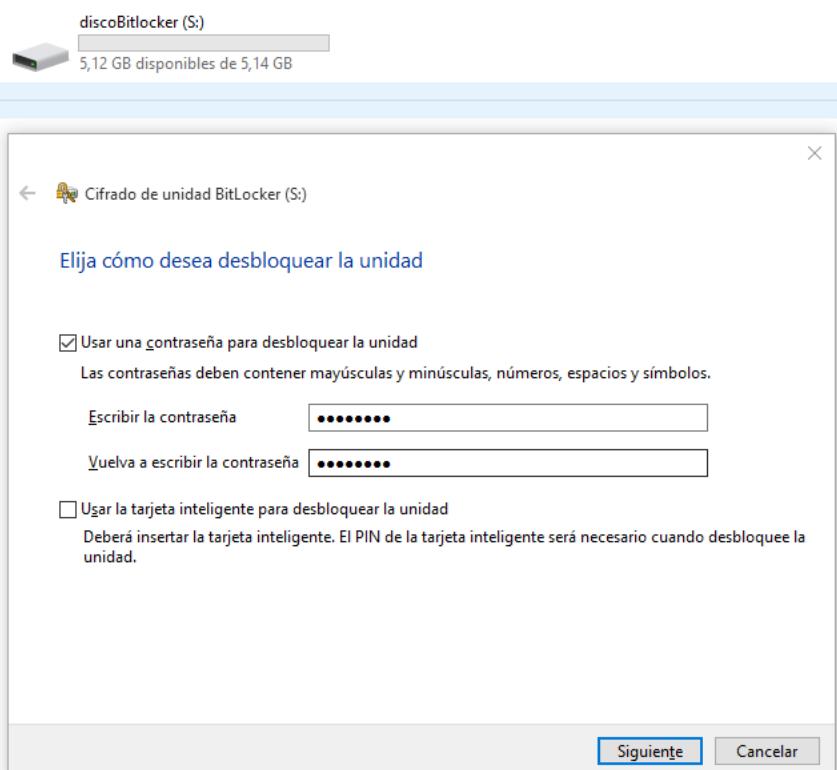
4. Confidencialidade: Sistema BitLocker en Windows e desencriptación da información.

Creamos unha partición aproximadamente de 5GiB para este exemplo.

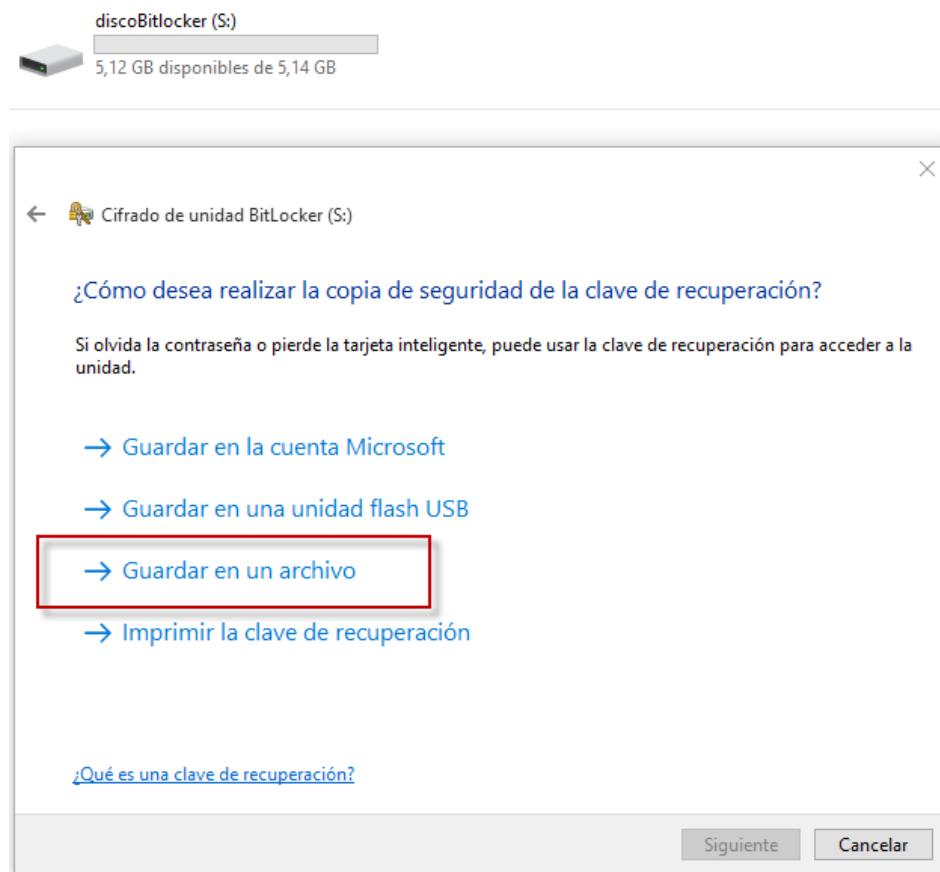
Activamos o cífrado da partición BitLocker. Esto sería exactamente igual si se tratase dun disco HDD ou SSD de forma íntegra ou dun dispositivo USB tipo pendrive.



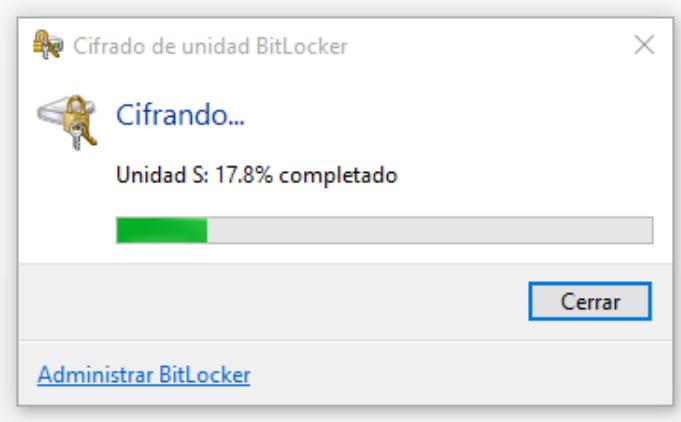
Establecemos unha contrasinal para o cífrado da partición.



Seleccionamos onde queremos guardar a clave de recuperación, no caso de esquecemos da nosa contrasinal establecida. No meu caso vou elixir guardar dita chave nun ficheiro de texto.



Despois eliximos que queremos cifrar si a unidade enteira ou solo cifrar o espacio de xa usado. No meu caso cifrarei a unidade enteira. E iniciamos o cifrado.



Unha vez cifrado pediranos a contrasinal cada vez que iniciemos sesión para desbloquear esta partición en concreto, tamén decir que desde administración de BitLocker dispomos de varias opcións de xestión.

The screenshot shows the Windows Control Panel under 'Sistema y seguridad' (System and Security) and then 'Cifrado de unidad BitLocker' (BitLocker Drive Encryption). It lists two drives:

- Unidad de sistema operativo:** C: BitLocker desactivado (Operating system drive: BitLocker disabled). A blue link 'Cifrar ahora' (Encrypt now) is visible.
- Unidades de datos fijas:** discoBitlocker (S:) BitLocker activado (Fixed data drive: BitLocker activated). A blue link 'Desactivar BitLocker' (Turn off BitLocker) is visible. To the left of the drive letter is a small icon of a USB drive with a lock symbol.

Below the drives, there is a section for removable drives: 'Unidades de datos extraíbles: BitLocker To Go'. It lists 'HDD_500GB (E): BitLocker activado' (Removable drive: BitLocker activated).

Existe algunha ferramenta que nos permita recuperar a información cifrada con Bitlocker ou EFS, no caso de que nos atopemos sen SO, sen axentes de recuperación e sen certificados exportados?.

Con ferramentas propias do sistema non hay unha forma conocida de poder descifrar un EFS ou un BitLocker. Sin embargo si existen ferramentas de terceiros de pago, as cales pódenos axudar a recuperar o acceso a os contenedores cifrados.

Cando os contenedores son descrifrados esta clave queda almacenada en memoria, mentres o equipo non se reinicie ou de algúun modo non borre esa caché da chave da sua memoria, poderíamos facer un volcado de memoria co fin de obter a chave de acceso.

Para BitLocker, EFS ou PGP, podemos facer uso de ferramentas como:

- Elcomsoft Forensic Disk Decryptor (299€)
- Passware Kit Forensic (875€)
- Advanced EFS Data Recovery (299€)

5. Integridade: Utilidade SFC en Windows.

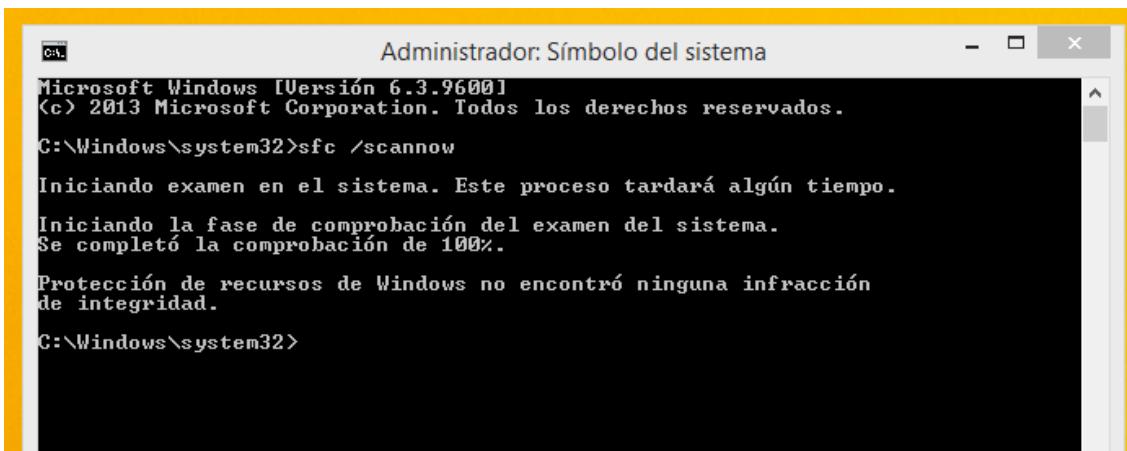
SFC (*System File Checker*) é o sistema de comprobación que ten Windows para garantizar a integridade dos ficheiros, este escanea os ficheiros propios de Windows y se ve alguma variación ou algún cambio neles, este restaura o ficheiro afectado a sua versión orixinal.

Si escribimos nunha consola de Windows:

```
sfc /scannow
```

(O cal escanea e repara, OLLO! Nun entorno empresarial y no caso de realizar esta operación nun Windows Servidor hay que ter coidado en primeiro SOLO escanear antes de reparar... polo que nese caso usaremos o atributo: `sfc /verifyonly` (solo verificar)).

Este escaneará e intentará reparar esos posibles ficheiros dañados.



```
Administrator: Símbolo del sistema
Microsoft Windows [Versión 6.3.9600]
(c) 2013 Microsoft Corporation. Todos los derechos reservados.

C:\Windows\system32>sfc /scannow

Iniciando examen en el sistema. Este proceso tardará algún tiempo.

Iniciando la fase de comprobación del examen del sistema.
Se completó la comprobación de 100%.
Protección de recursos de Windows no encontró ninguna infracción
de integridad.

C:\Windows\system32>
```

Para simular un caso dun arquivo dañado, renombrei o ficheiro **msprivs.dll.mui** (un ficheiro do paquete de idiomas integrado no sistema por defecto), e vemos como sfc reparou (restaurou mellor dito) a integridade deste ficheiro.



```
Seleccionar Administrador: Símbolo del sistema
Microsoft Windows [Versión 10.0.10240]
(c) 2015 Microsoft Corporation. Todos los derechos reservados.

C:\WINDOWS\system32>sfc /scannow

Iniciando examen en el sistema. Este proceso tardará algún tiempo.

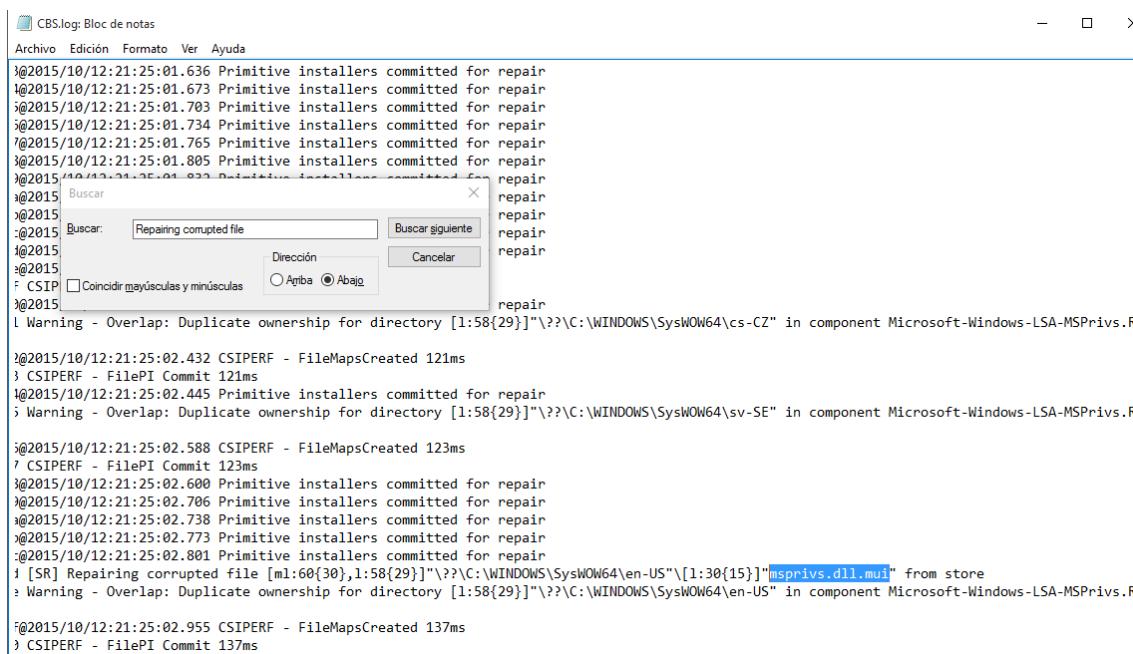
Iniciando la fase de comprobación del examen del sistema.
Se completó la comprobación de 100%.
Protección de recursos de Windows encontró archivos dañados y los reparó
correctamente. Para obtener más detalles, consulte CBS.Log windir\Logs\CMS\CBS.log. Por
ejemplo, C:\Windows\Logs\CMS\CBS.log. Tenga en cuenta que actualmente no se
admite el inicio de sesión en los escenarios de instalación sin conexión.

C:\WINDOWS\system32>
```

Para poder consultar de forma detallada o escaneo de sfc podemos ver o log almacenado no siguiente path:

C:\Windows\Logs\CMS\CMS.log

As entradas do tipo “Repaired file” ou “Repairing corrupted file” reparan un ficheiro copiando a versión orixinal traído do almacén do sistema.



6. Integridade: Escaneo con Rootkit Hunter en Linux.

Un Rootkit e un conxunto de ferramentas que permite a un atacante ter acceso continuo a un sistema, este permite ocultar a sua presencia e ocultar outros programas ou procesos que o atacante usa para ter acceso o sistema da vítima de forma ilícita.

Rootkit Hunter, é unha ferramenta para Linux que a sua función e detectar rootkits con comprobacións de firmas de hashes MD5 en ficheiros do sistema, busca directorios por defecto donde se hospedan os rootkits, permisos incorrectos, ficheiros ocultos, etc.

Para probar rkhunter, instalámolo en Ubuntu. Nunha terminal escribimos:

```
sudo apt-get install rkhunter
```

```
d3x@lost: ~
d3x@lost:~$ sudo apt-get install rkhunter
[sudo] password for d3x:
Leyendo lista de paquetes... Hecho
Creando árbol de dependencias
Leyendo la información de estado... Hecho
Se instalarán los siguientes paquetes extras:
  libruby1.9.1 libyaml-0.2 postfix ruby ruby1.9.1 unhide.rb
Paquetes sugeridos:
  procmail postfix-mysql postgresql postfix-ldap postfix-pcre sasl2-bin
  dovecot-common postfix-cdb postfix-doc bsd-mailx mailutils heirloom-mailx
  mailx tripwire libdigest-whirlpool-perl ri ruby-dev ruby1.9.1-examples
  rii-9.1 graphviz ruby1.9.1-dev ruby-switch
Paquetes recomendados:
  default-mta mail-transport-agent
Se instalarán los siguientes paquetes NUEVOS:
  libruby1.9.1 libyaml-0.2 postfix rkhunter ruby ruby1.9.1 unhide.rb
0 actualizados, 7 se instalarán, 0 para eliminar y 60 no actualizados.
Necesito descargar 4.037 kB de archivos.
Se utilizarán 17,4 MB de espacio de disco adicional después de esta operación.
¿Desea continuar? [S/n] ■
```

Actualizamos as firmas de comprobación da base de datos de rkhunter:

```
rkhunter --propupd
```

```
Processing triggers for ufw (0.34~rc-0ubuntu2) ...
Processing triggers for ureadahead (0.100.0-16) ...
Processing triggers for rkhunter (1.4.0-3) ...
[ Rootkit Hunter version 1.4.0 ]
File updated: searched for 168 files, found 136
d3x@lost:~$ sudo su
root@lost:/home/d3x# rkhunter --propupd
[ Rootkit Hunter version 1.4.0 ]
File updated: searched for 168 files, found 136
root@lost:/home/d3x#
```

Iniciamos o escaneo e a comprobación de posibles rootkits.

```
root@lost:/home/d3x
Processing triggers for libc-bin (2.19-0ubuntu6.6) ...
Processing triggers for ufw (0.34~rc-0ubuntu2) ...
Processing triggers for ureadahead (0.100.0-16) ...
Processing triggers for rkhunter (1.4.0-3) ...
[ Rootkit Hunter version 1.4.0 ]
File updated: searched for 168 files, found 136
d3x@lost:~$ sudo su
root@lost:/home/d3x# rkhunter --propupd
[ Rootkit Hunter version 1.4.0 ]
File updated: searched for 168 files, found 136
root@lost:/home/d3x# rkhunter --check
[ Rootkit Hunter version 1.4.0 ]

Checking system commands...
  Performing 'strings' command checks
    Checking 'strings' command [ OK ]
  Performing 'shared libraries' checks
    Checking for preloading variables [ None found ]
    Checking for preloaded libraries [ None found ]
    Checking LD_LIBRARY_PATH variable [ Not found ]

  Performing file properties checks
    Checking for prerequisites
      /usr/sbin/adduser [ OK ]
      /usr/sbin/chroot [ OK ]
      /usr/sbin/cron [ OK ]
      /usr/sbin/groupadd [ OK ]
      /usr/sbin/groupdel [ OK ]
      /usr/sbin/groupmod [ OK ]
      /usr/sbin/grpck [ OK ]
      /usr/sbin/nologin [ OK ]
      /usr/sbin/pwck [ OK ]
      /usr/sbin/rsyslogd [ OK ]
      /usr/sbin/tcpd [ OK ]
      /usr/sbin/useradd [ OK ]
      /usr/sbin/userdel [ OK ]
      /usr/sbin/usermod [ OK ]
      /usr/sbin/vipw [ OK ]
      /usr/bin/awk [ OK ]
      /usr/bin/basename [ OK ]
      /usr/bin/chattr [ OK ]
      /usr/bin/curl [ OK ]
```

O acabar o proceso de escaneo, este xenera un informe almacenado nun log no seguinte path:

/var/log/rkhunter.log

O cal podemos abrir con nano desde a terminal para examinar con detalle o escaneo.

The screenshot shows a terminal window titled "root@lost:/home/d3x". The title bar also displays "GNU nano 2.2.6" and "Archivo: /var/log/rkhunter.log". The main area of the terminal shows the contents of the rkhunter.log file. The log file contains numerous entries from the rkhunter audit process. Most entries are in green, indicating successful or informational checks. Some entries are in yellow, such as "[Skipped]" and "[None found]". The log includes sections for kernel module names, network ports (including TCP and UDP), hidden ports, and network interfaces. The bottom of the terminal window shows the standard nano key bindings.

```
[00:11:00] Checking kernel module names [ OK ]
[00:17:25]
[00:17:25] Info: Starting test name 'network'
[00:17:25] Checking the network...
[00:17:25]
[00:17:25] Performing checks on the network ports
[00:17:25] Info: Starting test name 'ports'
[00:17:25] * Performing check for backdoor ports
[00:17:25]     Checking for TCP port 1524 [ Not found ]
[00:17:25]     Checking for TCP port 1984 [ Not found ]
[00:17:25]     Checking for UDP port 2001 [ Not found ]
[00:17:25]     Checking for TCP port 2006 [ Not found ]
[00:17:25]     Checking for TCP port 2128 [ Not found ]
[00:17:25]     Checking for TCP port 6666 [ Not found ]
[00:17:25]     Checking for TCP port 6667 [ Not found ]
[00:17:25]     Checking for TCP port 6668 [ Not found ]
[00:17:25]     Checking for TCP port 6669 [ Not found ]
[00:17:26]     Checking for TCP port 7000 [ Not found ]
[00:17:26]     Checking for TCP port 13000 [ Not found ]
[00:17:26]     Checking for TCP port 14856 [ Not found ]
[00:17:26]     Checking for TCP port 25000 [ Not found ]
[00:17:26]     Checking for TCP port 29812 [ Not found ]
[00:17:26]     Checking for TCP port 31337 [ Not found ]
[00:17:26]     Checking for TCP port 32982 [ Not found ]
[00:17:26]     Checking for TCP port 33369 [ Not found ]
[00:17:26]     Checking for TCP port 47107 [ Not found ]
[00:17:26]     Checking for TCP port 47018 [ Not found ]
[00:17:26]     Checking for TCP port 60922 [ Not found ]
[00:17:26]     Checking for TCP port 62883 [ Not found ]
[00:17:26]     Checking for TCP port 65535 [ Not found ]
[00:17:26]     Checking for backdoor ports [ None found ]
[00:17:26]
[00:17:26] Info: Starting test name 'hidden_ports'
[00:17:26] Checking for hidden ports [ Skipped ]
[00:17:26] Info: Unable to find the 'unhide-tcp' command
[00:17:26]
[00:17:26] Performing checks on the network interfaces
[00:17:26] Info: Starting test name 'promisc'
[00:17:26]     Checking for promiscuous interfaces [ None found ]
[00:17:26]

^G Ver ayuda ^O Guardar ^R Leer Fich ^Y RePág. ^K Cortar Texto ^C Pos actual
^X Salir ^J Justificar ^W Buscar ^V Pág. Sig. ^U PegarTxt ^T Ortografía
```

7. Integridade: Verificación de hashes MD5 en ficheiros.

Comprobar a integridade dun ficheiro diranos si o ficheiro descargado foi modificado ou non o a cal asegúranos a integridade do mesmo, e con isto aseguramos de que o ficheiro e o verdadeiro do desarrollador de cuestión.

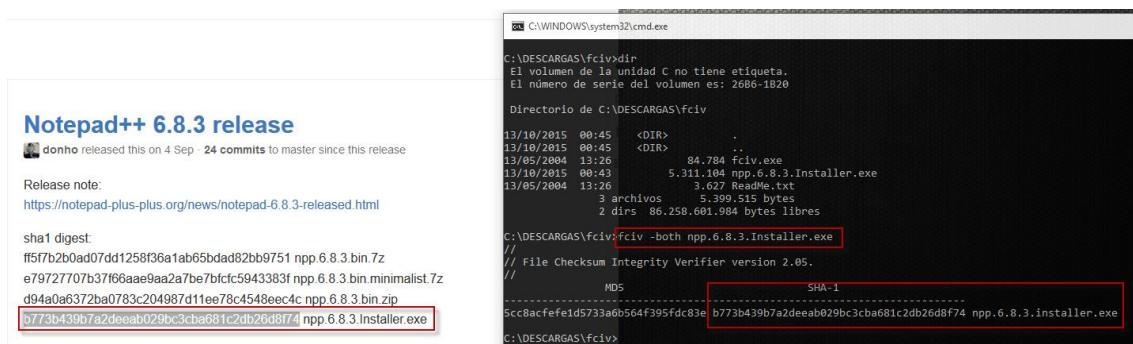
FCIV (File Checksum Integrity Verifier) é unha utilidade de incorporada non por defecto en Windows, pero que podemos descargar a parte e integrala no sistema, a cal comprobaremos a integridade dun ficheiro descargado.

No seguinte exemplo faremos a comprobación dun hash SHA-1 da aplicación Notepad++ v6.8.3 release.

Simplemente escribimos nunha consola de Windows:

fciv -md5 -sha1 <nomeDoFicheiro> ou fciv -both <nomeDoFicheiro>

Donde –both indica a comprobación tanto de hashes MD5 como de SHA-1.



```
C:\WINDOWS\system32\cmd.exe
C:\DESCARGAS\fcivxdir
El volumen de la unidad C no tiene etiqueta.
El n mero de serie del volumen es: 26B6-1B20
Directorio de C:\DESCARGAS\fciv
13/10/2015 00:45 <DIR> .
13/10/2015 00:45 <DIR> ..
13/05/2004 13:26 84.784 fciv.exe
13/10/2015 00:43 5.311.104 npp.6.8.3.Installer.exe
13/05/2004 13:26 3.627 ReadMe.txt
               2 archivos      5.399.515 bytes
                  2 dirs  86.258.601.984 bytes libres
C:\DESCARGAS\fciv:fciv -both npp.6.8.3.Installer.exe
// File Checksum Integrity Verifier version 2.05.
// MD5                               SHA-1
5cc8acfef1d5733aa6b564f395fd83e b773b439b7a2deeb629bc3cba681c2db26d8f74 npp.6.8.3.installer.exe
C:\DESCARGAS\fciv>
```

Outras tools para realizar comprobac ns de ficheiros son:

- MD5summer
- FSUM 2.52
- MD5Fourmilab
- MD5Checker 2.3.1
- pySum (para Linux)

Fai un tempo escribira un post sobre isto:

<http://www.zonasystem.com/2012/07/calcular-o-comprobar-los-checksum-crc.html>

8. Disponibilidade: Análises con Nmap en Windows e Linux.

Instalamos **Nmap en Windows**, descargámolo desde a web oficial e instalámolo.

En Zenmap escanemos o host remoto (10.10.13.56) un servizo FTP.

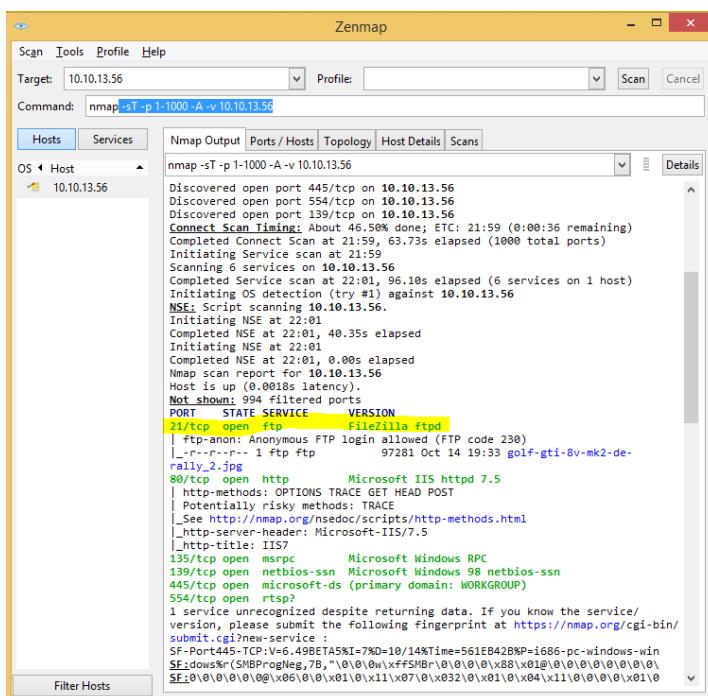
```
nmap -v -sT -p 1-1000 10.10.13.56
```

-v: información detallada.

-sT: auxilia o escaneo dos primeiros 1000 portos.

-p 1-1000: escanear un rango de portos entre 1 e 1000.

10.10.13.56: dirección ip remota a escanear.



Podemos facer o mesmo desde liña de comandos desde a consola de Windows ca propia ferramenta Nmap. De igual forma que no anterior tarefa.

```
C:\Windows\system32\cmd.exe
C:\Users\adrian>cd C:\Program Files\Nmap
C:\Program Files\Nmap>nmap -sT -p 1-1000 -A -v 10.10.13.56
Starting Nmap 6.49BETA5 < https://nmap.org > at 2015-10-14 21:59 Hora de verano
romance
NSE: Loaded 122 scripts for scanning.
NSE: Script Pre-scanning.
Initiating NSE at 21:59
Completed NSE at 21:59. 0.00s elapsed
Initiating NSE at 21:59
Completed NSE at 21:59. 0.00s elapsed
Initiating ARP Ping Scan at 21:59
Scanning 10.10.13.56 [1 port]
Completed ARP Ping Scan at 21:59. 0.02s elapsed (1 total hosts)
Initiating Parallel DNS resolution of 1 host. at 21:59
Completed Parallel DNS resolution of 1 host. at 21:59. 0.00s elapsed
Initiating Connect Scan at 21:59
Scanning 10.10.13.56 [1000 ports]
Discovered open port 135/tcp on 10.10.13.56
Discovered open port 139/tcp on 10.10.13.56
Discovered open port 445/tcp on 10.10.13.56
Discovered closed port 21/tcp on 10.10.13.56
Discovered open port 445/tcp on 10.10.13.56
Discovered open port 554/tcp on 10.10.13.56
Completed Connect Scan at 22:00. 41.28s elapsed (1000 total ports)
Initiating Service scan at 22:00
Scanning 6 services on 10.10.13.56
Completed Service scan at 22:01. 96.11s elapsed (6 services on 1 host)
Initiating OS detection [try #1] against 10.10.13.56
NSE: Script scanning 10.10.13.56.
Initiating NSE at 22:01
Completed NSE at 22:02. 40.61s elapsed
Initiating NSE at 22:02
Completed NSE at 22:02. 0.01s elapsed
Nmap scan report for 10.10.13.56
Host is up (0.0013s latency).
Not shown: 994 filtered ports
PORT      STATE SERVICE      VERSION
21/tcp    open  ftp          FileZilla ftpd
|_ftp-anon: Anonymous FTP login allowed (FTP code 230)
|  _-rwxr--r-- 1 ftp  ftp      97281 Oct 14 19:33 golf-gti-8v-mk2-de-rally_2.jp
g
80/tcp    open  http         Microsoft IIS httpd 7.5
|_http-methods: OPTIONS TRACE GET HEAD POST
|_http-risky-methods: TRACE
| See https://nmap.org/nse/doc/scripts/http-methods.html
|_http-server-header: Microsoft-IIS/7.5
|_http-title: IIS7
135/tcp   open  msrpc        Microsoft Windows RPC
139/tcp   open  netbios-ssn  Microsoft Windows 98 netbios-ssn
```

Nmap en Linux (Ubuntu), instalamos o paquete Nmap desde consola:

```
apt-get install nmap
```

```
root@adrian-VirtualBox: /home/adrian
adrian@adrian-VirtualBox:~$ sudo su
[sudo] password for adrian:
root@adrian-VirtualBox:/home/adrian# apt-get install nmap
Leyendo lista de paquetes... Hecho
Creando árbol de dependencias
Leyendo la información de estado... Hecho
Se instalarán los siguientes paquetes extras:
  libblas3 liblinear-tools liblinear1
Paquetes sugeridos:
  libsvm-tools liblinear-dev
Se instalarán los siguientes paquetes NUEVOS:
  libblas3 liblinear-tools liblinear1 nmap
0 actualizados, 4 se instalarán, 0 para eliminar y 114 no actualizados.
Necesito descargar 4.102 kB de archivos.
Se utilizarán 18,3 MB de espacio de disco adicional después de esta operación.
¿Desea continuar? [S/n] s
Des:1 http://es.archive.ubuntu.com/ubuntu/ trusty/main libblas3 i386 1.2.2011041
9-7 [183 kB]
Des:2 http://es.archive.ubuntu.com/ubuntu/ trusty/main liblinear1 i386 1.8+dfsg-
1ubuntu1 [32,0 kB]
Des:3 http://es.archive.ubuntu.com/ubuntu/ trusty/main liblinear-tools i386 1.8+
dfsg-1ubuntu1 [18,1 kB]
Des:4 http://es.archive.ubuntu.com/ubuntu/ trusty/main nmap i386 6.40-0.2ubuntu1
[3.869 kB]
```

O igual que os parámetros postos nas dúas tarefas anteriores en Windows, poñemos os mismos parámetros para facer uso da ferramente Nmap y descubrir o servidor FTP do host remoto (10.10.13.56).

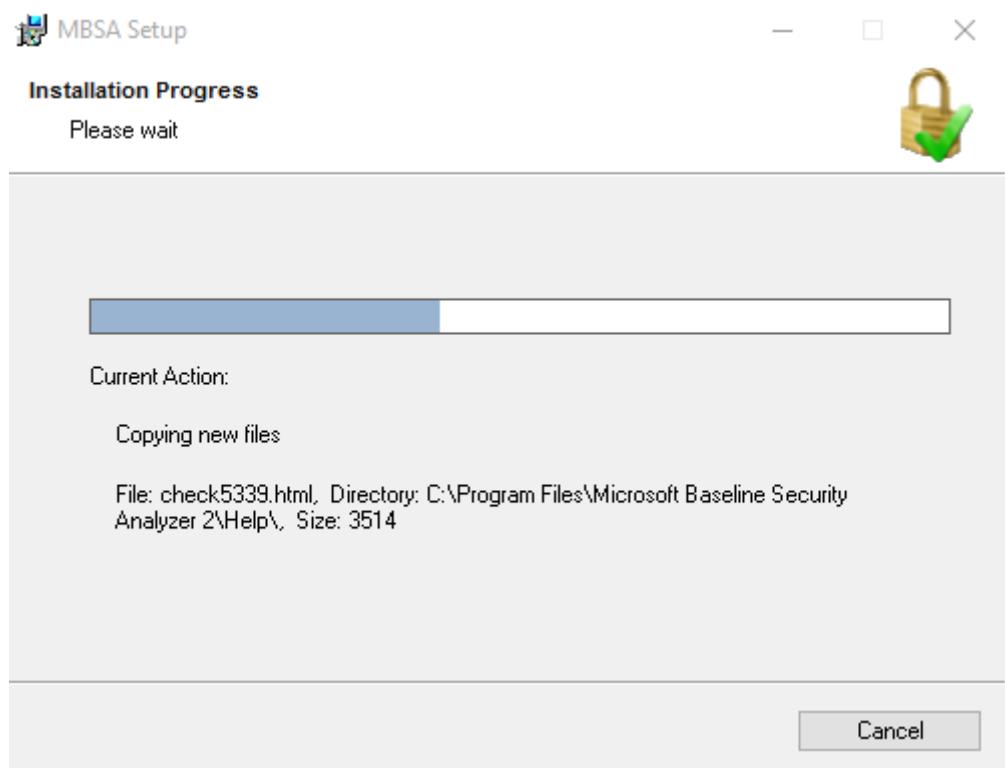
```
root@adrian-VirtualBox: /home/adrian#
root@adrian-VirtualBox:/home/adrian# nmap -v -sT -p 1-1000 10.10.13.56
Starting Nmap 6.40 ( http://nmap.org ) at 2015-10-14 21:56 CEST
Initiating ARP Ping Scan at 21:56
Scanning 10.10.13.56 [1 port]
Completed ARP Ping Scan at 21:56, 0.20s elapsed (1 total hosts)
Initiating Parallel DNS resolution of 1 host. at 21:56
Completed Parallel DNS resolution of 1 host. at 21:56, 0.00s elapsed
Initiating Connect Scan at 21:56
Scanning usuario-pc.acarballeira.local (10.10.13.56) [1000 ports]
Discovered open port 135/tcp on 10.10.13.56
Discovered open port 80/tcp on 10.10.13.56
Discovered open port 554/tcp on 10.10.13.56
Discovered open port 21/tcp on 10.10.13.56
Discovered open port 139/tcp on 10.10.13.56
Discovered open port 445/tcp on 10.10.13.56
Completed Connect Scan at 21:56, 12.23s elapsed (1000 total ports)
Nmap scan report for usuario-pc.acarballeira.local (10.10.13.56)
Host is up (0.0010s latency).
Not shown: 994 filtered ports
PORT      STATE SERVICE
21/tcp    open  ftp
80/tcp    open  http
135/tcp   open  msrpc
139/tcp   open  netbios-ssn
445/tcp   open  microsoft-ds
554/tcp   open  rtsp
MAC Address: 08:00:27:CB:D0:66 (Cadmus Computer Systems)

Read data files from: /usr/bin/../share/nmap
Nmap done: 1 IP address (1 host up) scanned in 12.52 seconds
          Raw packets sent: 1 (28B) | Rcvd: 1 (28B)
root@adrian-VirtualBox:/home/adrian#
```

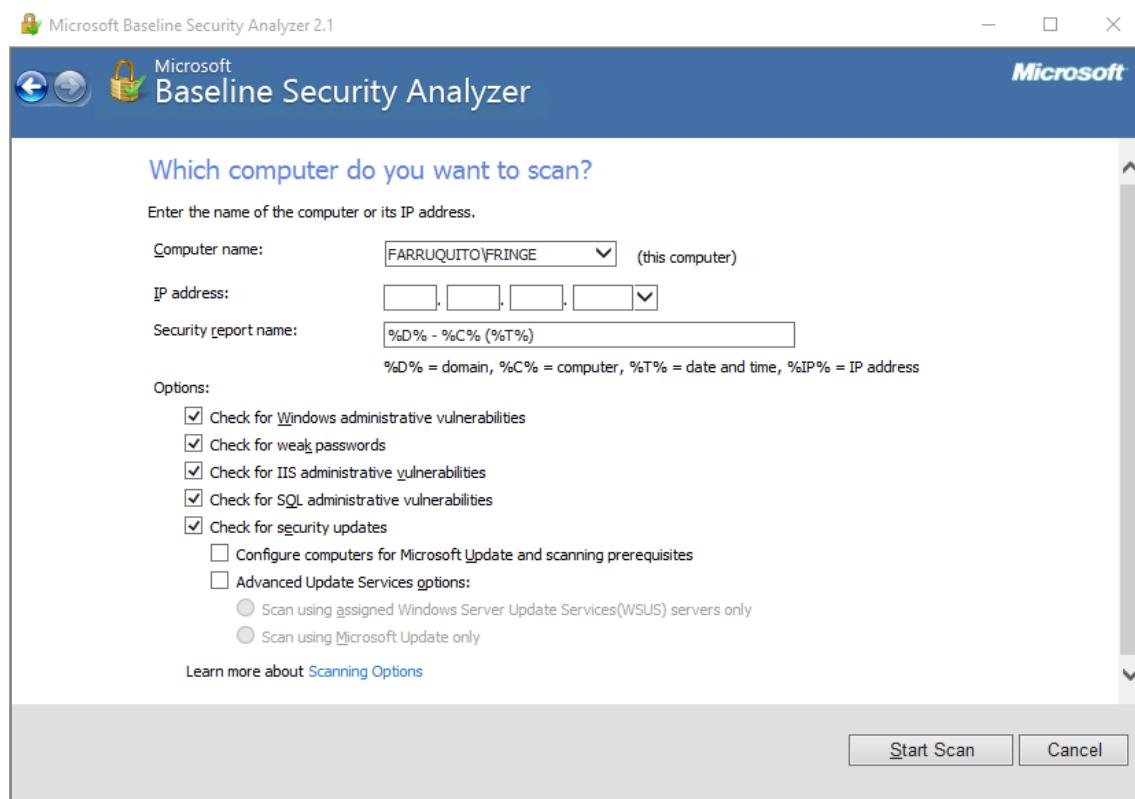
9. Vulnerabilidades: MBSA en Windows.

MBSA (*Microsoft Baseline Security Analyzer*) é unha ferramenta que nos permitirá determinar o estado de seguridade do noso equipo.

Instalamos MBSA descarga da web oficial de Microsoft.



Escaneamos o equipo local (no meu caso “Fringe”). No apartado de “Security report name”, vemos as variables que define e que significan.



Unha vez acabe a escaneo, veremos unha lista detallada na que según Microsoft recoméndanos correxir certos aspectos para mellorar a seguridade.

The screenshot shows the Microsoft Baseline Security Analyzer 2.1 interface. At the top, there's a toolbar with icons for back, forward, search, and help, followed by the title "Microsoft Baseline Security Analyzer". Below the title, the main content area displays several sections of audit results:

- Additional System Information:** A table with columns "Score", "Issue", and "Result". It lists findings such as "Guest Account" (disabled), "Restrict Anonymous" (properly restricted), and "Administrators" (no more than 2 found). Each item includes links to "What was scanned" and "Result details".
- Internet Information Services (IIS) Scan Results:** A section titled "Administrative Vulnerabilities" with a table showing items like "IIS Lockdown Tool" (not needed), "Sample Applications" (not installed), and "IISAdmin Virtual Directory" (not present).
- Additional System Information:** A table showing a single item: "IIS Logging Enabled" (all sites using recommended options).

At the bottom of the interface, there are buttons for "Print this report", "Copy to clipboard", "Previous security report", "Next security report", and "OK".

10. Vulnerabilidades: Nessus en Linux.

Nessus é un software de escaneo de vulnerabilidades. Este escanea portos, servizos, e examina a configuración de rede para posteriormente intentar atáculo con exploits escritos en NASL (*Nessus Attack Scripting Language*).

Existen duas versións de Nessus: Home e Work, esta última é de pago. Neste caso farei uso da versión Home.

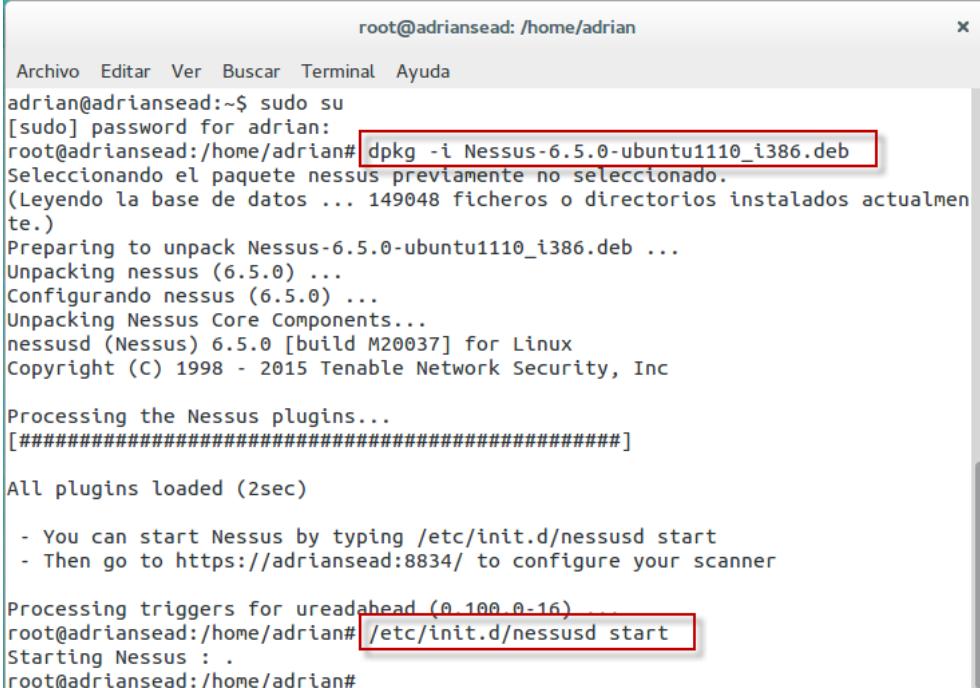
Primeiramente vamos a web oficial de Nessus, eliximos o tipo de producto (Home) e rexistramos unha conta de usuario asociada a un email para posteriormente recibir un código de activación do producto elixido e así poder traballar con Nessus. Este software é multiplataforma polo que admite diversos OS. Neste exemplo traballeremos con Nessus en Linux.

Unha vez descargamos o paquete correspondente .deb. Instalámolo desde un super-usuario nunha terminal:

```
dpkg -i <NomeDoPaquete.deb>
```

Despois inicializamos o servizo de nessusd:

```
/etc/init.d/nessusd start
```



The terminal window shows the following session:

```
root@adriansead: /home/adrian
Archivo Editar Ver Buscar Terminal Ayuda
adrian@adriansead:~$ sudo su
[sudo] password for adrian:
root@adriansead:/home/adrian# dpkg -i Nessus-6.5.0-ubuntu1110_i386.deb
Seleccionando el paquete nessus previamente no seleccionado.
(Leyendo la base de datos ... 149048 ficheros o directorios instalados actualmen-
te.)
Preparing to unpack Nessus-6.5.0-ubuntu1110_i386.deb ...
Unpacking nessus (6.5.0) ...
Configurando nessus (6.5.0) ...
Unpacking Nessus Core Components...
nessusd (Nessus) 6.5.0 [build M20037] for Linux
Copyright (C) 1998 - 2015 Tenable Network Security, Inc

Processing the Nessus plugins...
[########################################]

All plugins loaded (2sec)

- You can start Nessus by typing /etc/init.d/nessusd start
- Then go to https://adriansead:8834/ to configure your scanner

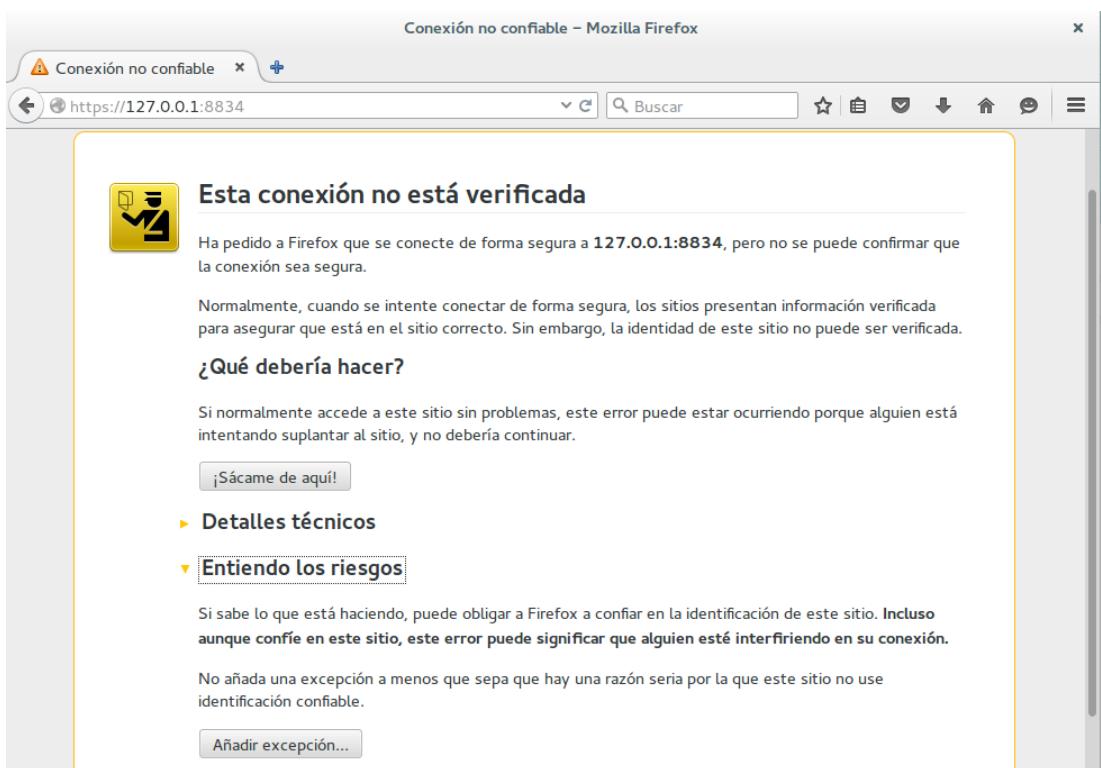
Processing triggers for ureadahead (0.100.0-16)
root@adriansead:/home/adrian# /etc/init.d/nessusd start
Starting Nessus : .
root@adriansead:/home/adrian#
```

Para poder acceder o panel de configuración de Nessus terémolo que facer a través do navegador web, xa que este corre como si se tratara dunha aplicación web local.

Na barra de dirección do navegador web escribimos a dirección de loopback seguido do porto que usa Nessus (8834).

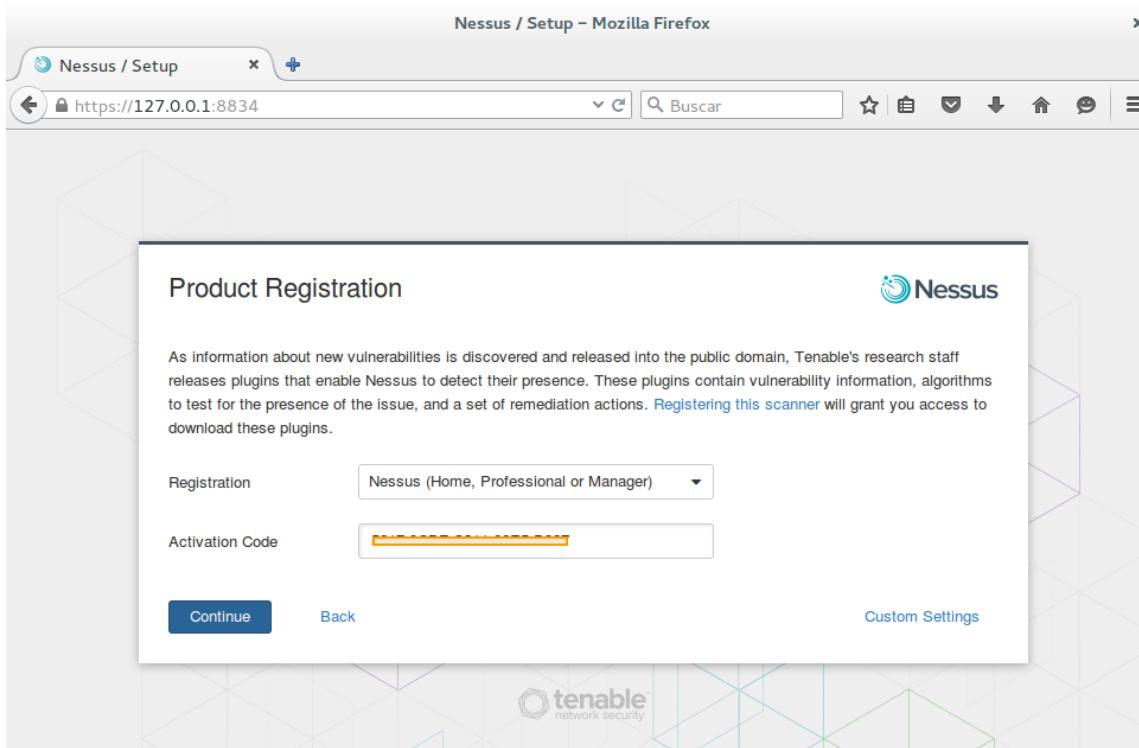
<https://127.0.0.1:8834>

Esta dirección corre a través de protocolo seguro HTTPS, polo o navegador móstranos unha alerta de si estamos seguros de continuar xa que este certificado SSL non está rexistrado na sua biblioteca por defecto. Añadimos a excepción e confirmamos que confiamos no certificado.

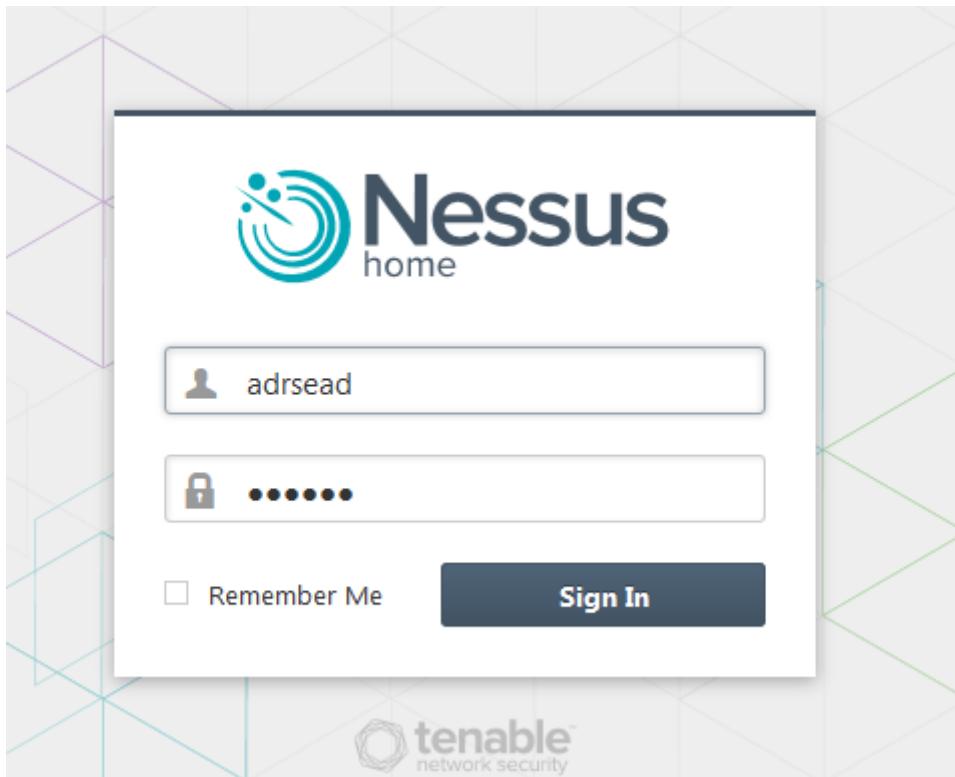


A continuación creamos un usuario con unha contrasinal para acceder o panel de Nessus.

Seleccionamos o tipo de producto e introducimos o código de activación que recibimos na dirección email que puxéramos cando nos rexistramos na web oficial de Nessus (este punto está comentado o principio de esta práctica, apartado 10).



Esperamos a que descargue e se inicie o proceso para o acceso a Nessus. Introducimos o usuario e contrasinal que creamos anteriormente.



Unha vez accedimos o panel de configuración de Nessus, podemos ver os servizos que temos dispoñibles ca conta de tipo Home. Xa que o resto de servizos tendríamos que actualizar a unha conta Work para poder facer uso deles.

The screenshot shows the 'Scanner Templates' section of the Nessus web interface. It displays a grid of 15 templates, each with a title, icon, and a brief description. Some templates have a purple 'UPGRADE' banner in the top right corner. The templates include:

- Advanced Scan**: Configure a scan without using any recommendations.
- Audit Cloud Infrastructure**: Audit the configuration of third-party cloud services.
- Bash Shellshock Detection**: Remote and local checks for CVE-2014-6271 and related vulnerabilities.
- Basic Network Scan**: A full system scan suitable for any host.
- Credentialed Patch Audit**: Authenticate to hosts and enumerate missing updates.
- GHOST (glibc) Detection**: Local checks for CVE-2015-0235.
- Host Discovery**: A simple scan to discover live hosts and open ports.
- Internal PCI Network Scan**: Perform an internal PCI DSS (11.2.1) vulnerability scan.
- MDM Config Audit**: Audit the configuration of mobile device managers.
- Mobile Device Scan**: Assess mobile devices via Microsoft Exchange or an MDM.
- Offline Config Audit**: Audit the configuration of network devices.
- PCI Quarterly External Scan**: Approved for quarterly external scanning as required by PCI.
- Policy Compliance Auditing**: Audit system configurations against a known baseline.
- SCAP and OVAL Auditing**: Audit systems using SCAP and OVAL definitions.
- Web Application Tests**: Scan for published and unknown web vulnerabilities.

Neste exemplo crearei dous novos escaneos; un escaneo básico de rede e un escaneo de aplicacíons web.

NOTA: A partir deste punto por problemas de conexión de Internet non puiden seguir con Nessus en Linux, polo que **instaleino nun Windows**. Igualmente neste punto administración do panel de xestión web de Nessus e a misma en calqueira OS.

The screenshot shows the 'Scans' section of the Nessus web interface. It displays a list of existing scans under the 'My Scans' tab. The list includes:

- webaplicaciones (On Demand, 07:07 PM)
- adrSEAD_red (On Demand, 06:46 PM)

Below the list are buttons for '+ New Scan' and 'New Folder'. At the bottom of the page, there is a copyright notice: '© 1998 - 2015 Tenable Network Security®. All Rights Reserved. Nessus Home v. 6.5.0'.

Resultado de escaneo de aplicacíons web:

The screenshot shows the results of the 'webaplicaciones' scan. The main table displays vulnerabilities categorized by severity (INFO) and plugin name. The table includes columns for Severity, Plugin Name, Plugin Family, Count, and Host Details. The Host Details pane shows the following information for the target host:

- IP: [REDACTED]
- DNS: [REDACTED]
- OS: Microsoft Windows 7 Enterprise
- Start: Today at 7:00 PM
- End: Today at 7:07 PM
- Elapsed: 7 minutes
- KB: Download

Below the table is a pie chart titled 'Vulnerabilities' with one segment labeled 'Info'.

Resultado de escaneo básico de red:

The screenshot shows the Nessus interface with the following details:

- Hosts:** adrSEAD_red
- Scans:** CURRENT RESULTS: TODAY AT 6:48 PM
- Vulnerabilities:** 50
- Severity Legend:**
 - Critical (Red)
 - Medium (Orange)
 - Low (Green)
 - Info (Blue)
- Plugin Name:**
 - MS14-066: Vulnerability in Schannel Could Allow Remote Code Execution (2992611) (unprivileged check)
 - SSL Certificate Cannot Be Trusted
 - Microsoft Windows Remote Desktop Protocol Server Man-in-the-Middle Weakness
 - SMB Signing Required
 - SSL Certificate Signed Using Weak Hashing Algorithm
 - SSL RC4 Cipher Suites Supported (Bar Mitzvah)
 - SSL Self-Signed Certificate
 - Terminal Services Doesn't Use Network Level Authentication (NLA) Only
 - Terminal Services Encryption Level is Medium or Low
 - Terminal Services Encryption Level is not FIPS-140 Compliant
 - netstat portscanner (SSH)
 - DCE Services Enumeration
 - Service Detection
 - Microsoft Windows SMB Service Detection
 - SSL / TLS Versions Supported
 - SSL Certificate Information
 - SSL Cipher Block Chaining Cipher Suites Supported

Como podemos ver nos escaneos, recoméndanos que debemos protexer da configuración do noso sistema actual.

11. Conclusións.

As conclusións que saco con todo isto é que ningún sistema operativo é seguro. Calquer sistema poder estar más ou menos protexido pero tódolos sistemas teñen un mínimo de parámetros configurables para securizalo. Que un sistema sexa amais vulnerable ca outro lévame a pensar de que a razón é a demanda, e dicir, si un sistema operativo é usado polo 70% da poboación o máis normal e que sexa ese sistema o máis demandado para atacar, de ahí que MS Windows sexa hoxe en día o máis atacado, pero si fose o sistema máis popular calquera outro sería ese outro o máis atacado e polo tanto o máis vulnerable.

Esto ten que ver despois con que as compañías propietarias de ditos sistemas tomen más ou menos medidas de seguridade para os seus produtos.

Cada sistema trae as suas ferramentas axeitadas para poder aplicar políticas seguras con fin de salvagardar a información, unhas más robustas ou confiables e outras algo menos, o que está claro que ten que ser o usuario final o que se conciencie do importante que pode ser a privacidade ou integridade dos seus datos, e ten que ser él o que realmente cos mecanismos expostos faga uso das ferramentas apropiadas.