

Índice

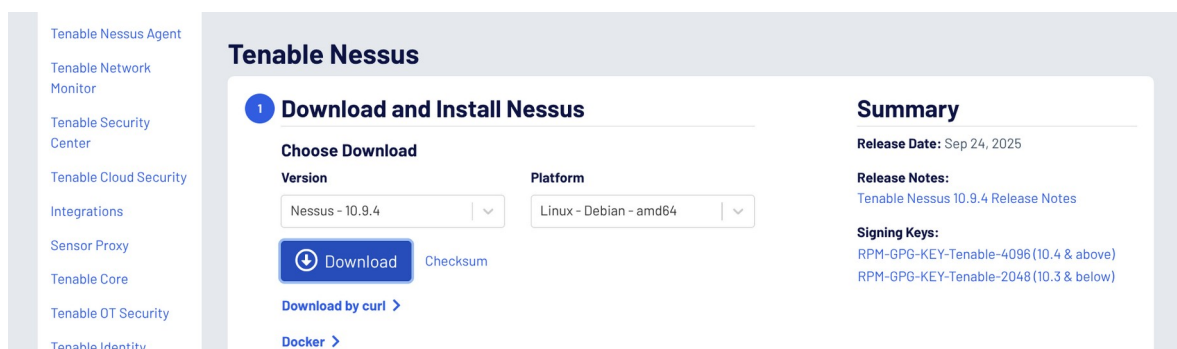
Nessus Home.....	2
Instalación y configuración inicial.....	2
Descubrimiento de hosts.....	5
Escaneo a un host descubierto.....	8
Alertas por correo electrónico.....	11
Prueba de uso en red.....	17
Escáner en máquina metasploitable.....	21
Informe.....	22

Nessus Home.

Nessus Home ya no existe y ahora es Nessus Essentials, una versión gratuita del escáner de vulnerabilidades Nessus de Tenable que permite escanear hasta 16 direcciones IP para encontrar fallos de seguridad en sistemas y redes. Es una herramienta para profesionales de ciberseguridad, educadores y estudiantes que desean realizar evaluaciones de vulnerabilidades de alta velocidad y en profundidad sin costo, aunque con algunas limitaciones en comparación con las versiones de pago.

Instalación y configuración inicial.

Para descargarnos el paquete nos vamos a la página de [Nessus](#):



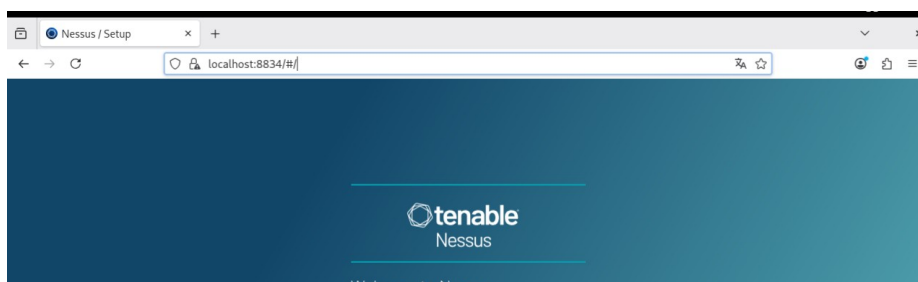
Una vez que tenemos el .deb lo descargamos con dpkg:

```
debian@marina:~$ sudo dpkg -i Nessus-10.9.4.deb
```

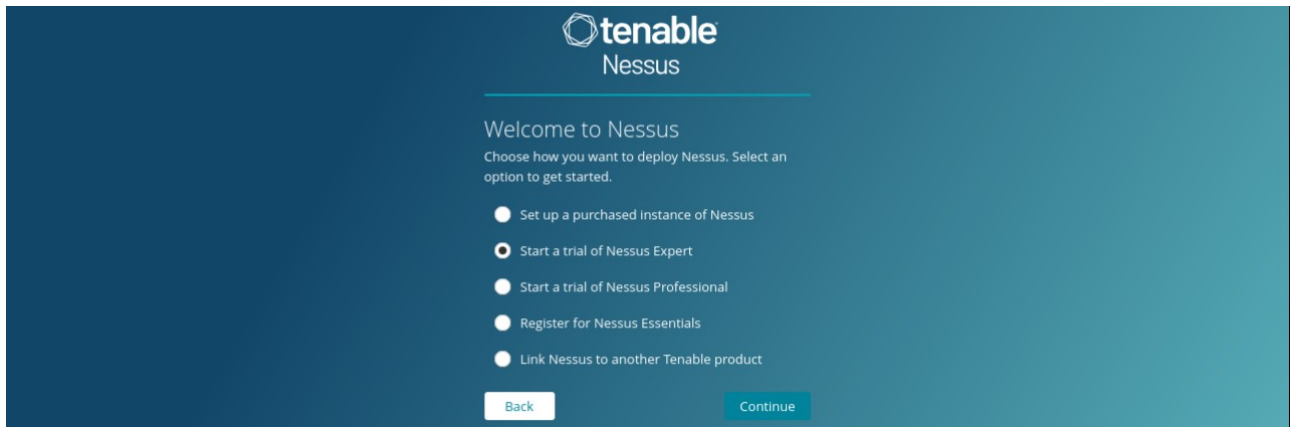
Activamos el servicio y comprobamos que escucha por el puerto 8834 que es el de nessus:

```
debian@marina:~$ sudo systemctl start nessusd.service
debian@marina:~$ sudo ss -tulnp | grep nessusd
tcp    LISTEN 0      1024
0.0.0.0:8834      0.0.0.0:*      users:
(("nessusd",pid=1423,fd=18))
tcp    LISTEN 0      1024
[::]:8834 [::]:*      users:(("nessusd",pid=1423,fd=19))
```

Nos vamos al navegador para poder configurar Nessus:



Nos ofrece una serie de opciones:



The image shows the 'Welcome to Nessus' screen. At the top is the Tenable Nessus logo. Below it, the text 'Welcome to Nessus' is followed by 'Choose how you want to deploy Nessus. Select an option to get started.' There are five radio button options: 'Set up a purchased instance of Nessus', 'Start a trial of Nessus Expert' (which is selected), 'Start a trial of Nessus Professional', 'Register for Nessus Essentials', and 'Link Nessus to another Tenable product'. At the bottom are 'Back' and 'Continue' buttons.

tenable
Nessus

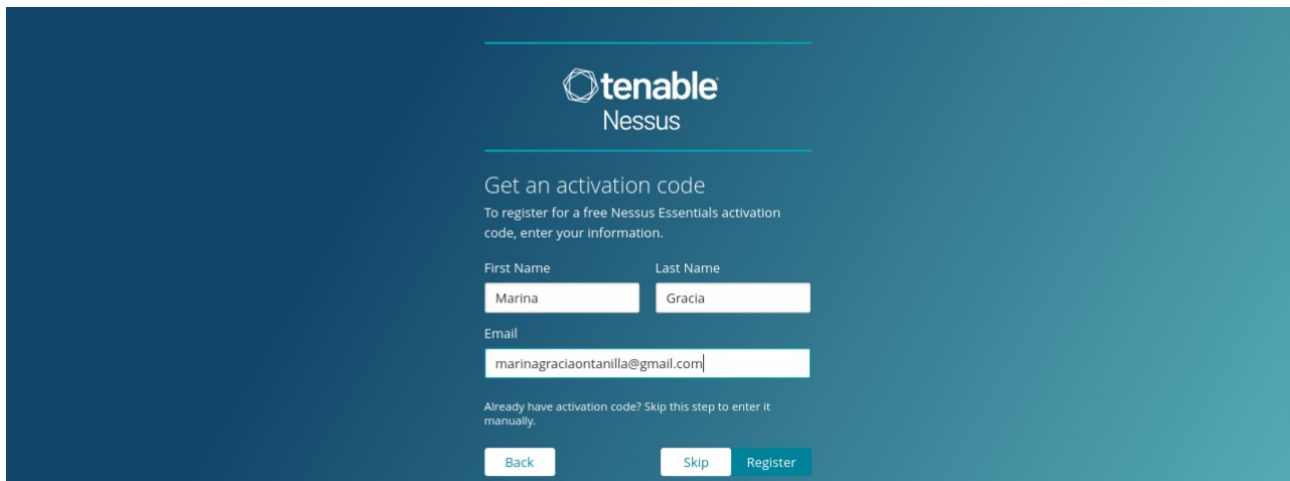
Welcome to Nessus

Choose how you want to deploy Nessus. Select an option to get started.

- ☐ Set up a purchased instance of Nessus
- ☒ Start a trial of Nessus Expert
- ☐ Start a trial of Nessus Professional
- ☐ Register for Nessus Essentials
- ☐ Link Nessus to another Tenable product

Back Continue

En este caso, vamos a elegir Nessus Essentials, ya que es la versión gratuita para uso personal/educativo. La limitación que encontraremos será el número de IPs que se pueden escanear. Nos registramos:



The image shows the 'Get an activation code' screen. It has the Tenable Nessus logo at the top. The text says 'Get an activation code' and 'To register for a free Nessus Essentials activation code, enter your information.' There are input fields for 'First Name' (Marina), 'Last Name' (Gracia), and 'Email' (marinagraciaontanilla@gmail.com). Below these is a link: 'Already have activation code? Skip this step to enter it manually.' At the bottom are 'Back', 'Skip', and 'Register' buttons.

tenable
Nessus

Get an activation code

To register for a free Nessus Essentials activation code, enter your information.

First Name Last Name

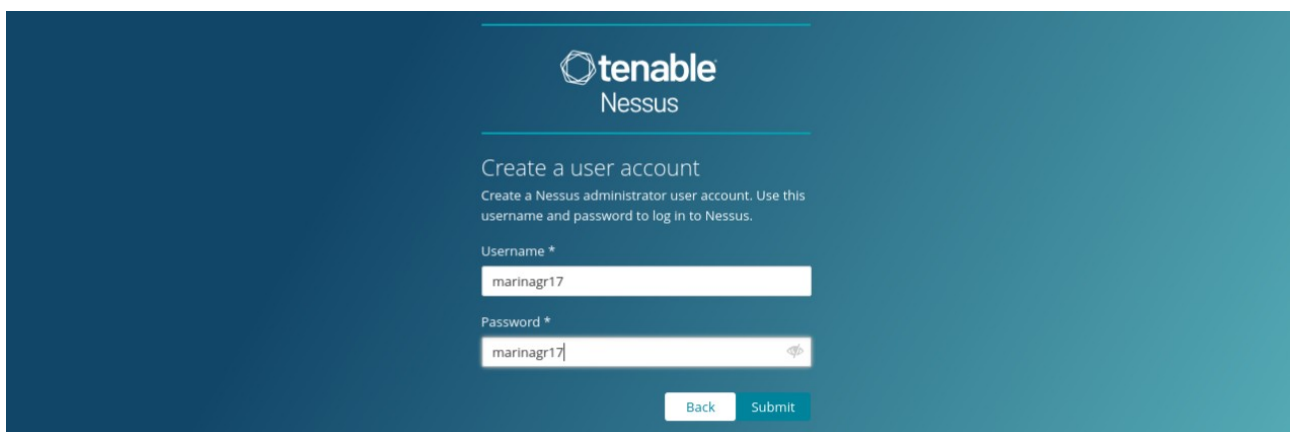
Marina Gracia

Email

marinagraciaontanilla@gmail.com

Already have activation code? Skip this step to enter it manually.

Back Skip Register



The image shows the 'Create a user account' screen. It has the Tenable Nessus logo at the top. The text says 'Create a user account' and 'Create a Nessus administrator user account. Use this username and password to log in to Nessus.' There are input fields for 'Username *' (marinagr17) and 'Password *' (marinagr17). At the bottom are 'Back' and 'Submit' buttons.

tenable
Nessus

Create a user account

Create a Nessus administrator user account. Use this username and password to log in to Nessus.

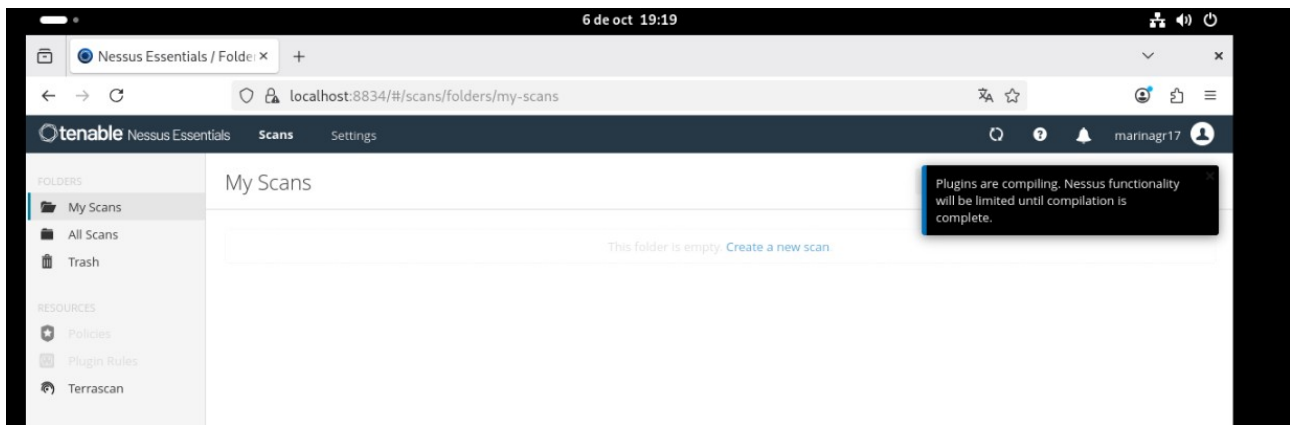
Username *

marinagr17

Password *

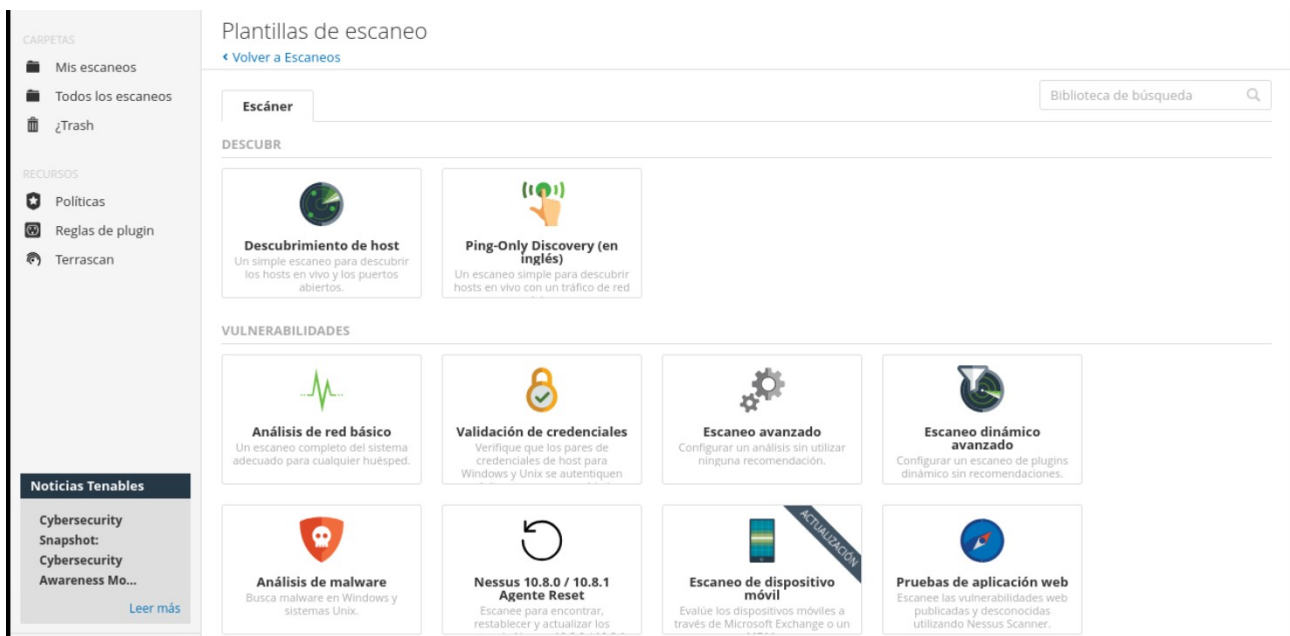
marinagr17

Back Submit



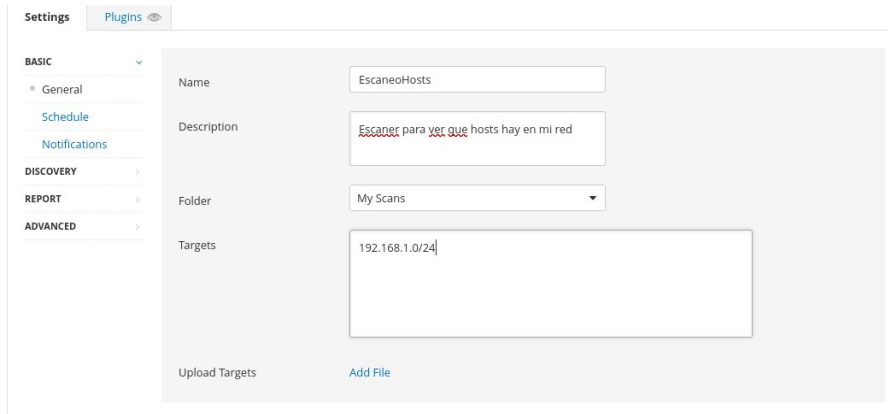
Una vez aquí, tenemos que esperar a que se descarguen todos los pluggins. Podemos verlo en esta sección:

Si le damos a la opción de nuevo escaneo vemos que hay diversos tipos:





Descubrimiento de hosts.

El primer escáner que aparece es el de 'Descubrimiento de host'. Este nos sirve para saber que hosts hay en mi red y es el primer paso para cualquier evaluación de vulnerabilidades.



Al lanzar el escaneo de descubrimiento de hosts, vemos además de que hosts están en nuestra red, su información asociada, como la dirección IP, FQDN, sistemas operativos y puertos abiertos, si está disponible. Después de tener una lista de host, podemos elegir cual escanear:

<input type="checkbox"/>	Nombre	Tipo de escaneo	Horario	Última exploración ▼	
<input type="checkbox"/>	EscaneoHost	Descubrimiento d...	Bajo demanda	✓ Today at 6:51 PM	 

EscaneoHosts

[Back to My Scans](#)

Configure

Hosts 5 Vulnerabilities 2 History 1


Filter Search Hosts 5 Hosts

Host ▼	FQDN	Ports	%
192.168.1.76	android-2.home		100%
192.168.1.47	debian.home		100%
192.168.1.11			100%
192.168.1.10			100%
192.168.1.1	liveboxfibra.home	139, 445	74%

Scan Details

Policy: Host Discovery
Status: Running
Severity Base: CVSS v3.0
Scanner: Local Scanner
Start: Today at 6:58 PM

Vulnerabilities



Critical

High

Medium



Low

Info

En la pestaña 'Hosts', visualiza los hosts que Nessus ha descubierto. Si accedemos a la pestaña 'vulnerabilities':

Hosts 5 Vulnerabilities 2 History 1

Filter Search Vulnerabilities 2 Vulnerabilities

Sev ▼	CVSS ▼	VPR ▼	EPSS ▼	Family ▲	Count ▼		can Details
INFO				Settings	5		Policy: Host Discovery Status: Completed
INFO				Port scanners	5		Severity Base: CVSS v3.0 Scanner: Local Scanner Start: Today at 6:58 PM

5

Settings: Proporciona información sobre la configuración del escaneo. No detecta vulnerabilidades, sino que reporta detalles operativos del proceso de escaneo, como las opciones de configuración usadas. El 5 indica que esto se aplica en los 5 hosts escaneados.

Output

Information about this scan :	
Nessus version : 10.9.4	
Nessus build : 20037	
Plugin feed version : 202510060158	
Scanner edition used : Nessus Home	
Scanner OS : LINUX	
Scanner distribution : debian10-x86-64	
Scan type : Normal	
more...	
To see debug logs, please visit individual host	
Port ▲	Hosts
N/A	192.168.1.10

La salida que nos muestra es:

```
Information about this scan :
Nessus version : 10.9.4
Nessus build : 20037
Plugin feed version : 202510060158
Scanner edition used : Nessus Home
Scanner OS : LINUX
Scanner distribution : debian10-x86-64
Scan type : Normal
Scan name : EscaneoHosts
Scan policy used : Host Discovery
Scanner IP : 192.168.122.252
WARNING : No port scanner was enabled during the scan. This may
lead to incomplete results.
Port range : default
Ping RTT : 28.348 ms
...
Scan for Unpatched Vulnerabilities : no
...
Max hosts : 256
Max checks : 5
...
Scan Start Date : 2025/10/7 18:58 CEST (UTC +02:00)
Scan duration : 51 sec
Scan for malware : no
```

Esta salida es como un resumen de configuración del proceso. Nos muestra datos básicos del sistema y la versión (Nessus version: 10.9.4 y Nessus build: 20037). Indica la fecha de actualización de los plugin (202510060158). Podemos ver también que

estamos usando la versión gratuita de Nessus (Scanner edition used : Nessus Home). El escáner corre en Linux, específicamente Debian 10 para procesadores de 64 bits.

La salida nos muestra el número máximo de hosts y de controles simultáneos:

```
Max anfitriones : 256
Controles máximos : 5
```

Esto establece un límite moderado para mantener el rendimiento del escaneo.

Port scanners: Incluye plugins dedicados al escaneo de puertos para descubrir hosts activos y servicios abiertos. Ejemplos comunes son "Nessus SYN Scanner" (escaneo TCP SYN para puertos abiertos) o "Nessus UDP Scanner". En esencia, Nessus envía paquetes a rangos de puertos para ver si responden, lo que ayuda a confirmar hosts vivos sin una conexión completa. El descubrimiento de hosts en Nessus depende en gran medida de escaneos de puertos (configurados en la sección "Network Port Scanners" de la política). Aparece como INFO porque reporta los resultados del escaneo de puertos (por ejemplo, puertos abiertos detectados), no vulnerabilidades.

Este plugin básicamente comprueba si mi host está activo en la red y lo hace mediante ping:

- Ping ARP: Funciona si el host está en la misma subred local y se usa Ethernet.
- Ping ICMP: Ping clásico.
- Ping TCP: Envía un paquete SYN a un puerto del host y espera a un SYN/ACK o RST.
- Ping UDP: Prueba usando servicios UDP conocidos como DNS, RCP o NTP.

La salida muestra:

Output

```
The remote host is up
The remote host replied with an ICMP unreachable packet sent in response to a TCP SYN
packet sent to port 497
```

To see debug logs, please visit individual host

Port ▲

Hosts

N/A

192.168.1.47

```
The remote host is up
The remote host replied to an ICMP echo packet
```

To see debug logs, please visit individual host

Port ▲

Hosts

N/A

192.168.1.1 192.168.1.10 192.168.1.11 192.168.1.76

Primera salida:

The remote host is up: El host remoto (192.168.1.47) está activo y respondiendo.

The remote host replied with an ICMP unreachable packet sent in response to a TCP SYN packet sent to port 497: Nessus detectó el host enviando un paquete TCP SYN al puerto 497. El host contestó con un mensaje ICMP “puerto no disponible”, pero esa respuesta sirve para saber que el host no lo ignoró. Es un truco para encontrar hosts que bloquean pings normales.

Segunda salida:

The remote host is up: Estos hosts están activos (192.168.1.1, 192.168.1.10, 192.168.1.11 y 192.168.1.76).

The remote host replied to an ICMP echo packet: Detección directa vía "ping" (ICMP echo request). Nessus envió un ping simple, y estos hosts respondieron con un echo reply, confirmando que están activos.

Escaneo a un host descubierto.

Si nos vamos a la sección hosts del anterior escaneo, podemos seleccionar que host queremos escanear:

EscaneoHosts

Back to My Scans

Hosts 5 Vulnerabilities 2 History 1

Filter Search Hosts 5 Hosts (1 Selected) Clear Selected Item

Host	FQDN	Ports
<input type="checkbox"/> 192.168.1.76	android-2.home	x
<input checked="" type="checkbox"/> 192.168.1.47	debian.home	x
<input type="checkbox"/> 192.168.1.11		x
<input type="checkbox"/> 192.168.1.10		x

Scan Details

Policy: Host Discovery
Status: Completed
Severity Base: CVSS v3.0
Scanner: Local Scanner
Start: Today at 6:58 PM
End: Today at 6:59 PM
Elapsed: 2 minutes

Se nos abre otra vez la lista de plantillas para seleccionar una. Tenable Nessus llena automáticamente la lista de Targets con los hosts que hemos seleccionado.

En este caso, vamos a seleccionar ‘Análisis de malware’.

New Scan / Malware Scan

Back to Scan Templates

Settings Credentials Plugins

BASIC

General

Schedule

Notifications

DISCOVERY

ASSESSMENT

REPORT

ADVANCED

Name: EscanerMalware

Description: Escaner de malware a mi pc

Folder: My Scans

Targets: 192.168.1.47

Este tipo de escaneo necesita credenciales autenticadas para acceder al interior del host y buscar malware en archivos, registros o procesos. Sin ellas, Nessus no puede hacer un chequeo profundo y falla.

Lo primero será generar un par de claves en la máquina del servidor:


```
nessus@debian:~/.ssh$ ssh-keygen
. . .
nessus@debian:~/.ssh$ ssh-copy-id -i nessus.pub
marina@192.168.1.47
```

A continuación, añadimos credenciales:

The screenshot displays the Nessus web interface for configuring credentials. On the left, a sidebar shows 'Host' selected under 'CATEGORIES'. The main area is titled 'SSH User: root, Auth method: public key'. It contains several configuration fields: 'Authentication method' is set to 'public key'; 'Username' is 'marina'; 'Private key' is 'nessus'; 'Private key passphrase' is masked with dots; 'Elevate privileges with' is set to 'Nothing'; and 'Targets to prioritize credentials' is '192.168.1.76'. Below this is the 'Global Credential Settings' section, which includes: 'known_hosts file' set to 'known_hosts'; 'Preferred port' set to '22'; 'Client version' set to 'OpenSSH_5.0'; and 'Attempt least privilege' which is unchecked. Each field has a brief description or warning below it.

Guardamos y probamos.

Hosts 1 Vulnerabilities 6 History 2

Filter ▼ Search Hosts 1 Host

<input type="checkbox"/>	Host	Auth	Vulnerabilities ▼
<input type="checkbox"/>	192.168.1.47	Pass	23

Scan Details

Policy: Malware Scan
 Status: Completed
 Severity Base: CVSS v3.0
 Scanner: Local Scanner

Hosts 1 Vulnerabilities 6 History 2

Filter ▼ Search Vulnerabilities 6 Vulnerabilities

<input type="checkbox"/>	Sev ▼	CVSS ▼	VPR ▼	EPSS ▼	Family ▲	Count ▼	
<input type="checkbox"/>	INFO				Port scanners	18	
<input type="checkbox"/>	INFO				General	1	
<input type="checkbox"/>	INFO				General	1	
<input type="checkbox"/>	INFO				Settings	1	
<input type="checkbox"/>	INFO				General	1	
<input type="checkbox"/>	INFO				General	1	

Scan Details

Policy: Malware Scan
 Status: Completed
 Severity Base: CVSS v3.0
 Scanner: Local Scanner
 Start: Today at 8:13 PM
 End: Today at 8:17 PM
 Lapsed: 4 minutes

Vulnerabilities



Ahora Nessus hizo un escaneo profundo (leyó directorios como /tmp, chequeó procesos, etc.), no solo superficial.

Se han detectado un total de 6 vulnerabilidades, todas calificadas como INFO (informativas, no críticas). En un escaneo de malware esto es común si no hay amenazas reales.

Port scanners: Nos da información de que puertos están abiertos. Nessus usa SSH para ejecutar comandos como netstat o ss directamente en mi máquina. No es una vulnerabilidad real, solo información sobre los puertos que están escuchando.

Puertos detectados:

```

Port 22/tcp was found to be open → SSH
Port 53/udp was found to be open Port 53/tcp was found to be open → DNS
Port 67/udp was found to be open → DHCP
Port 1521/tcp was found to be open → ORACLE
Port 1716/udp was found to be open Port 1716/tcp was found to be open → KDE
Port 5353/udp was found to be open → mDNS
Port 5500/tcp was found to be open → HTTP
Port 10056/udp was found to be open
Port 11166/udp was found to be open
  
```

```
Port 11745/udp was found to be open
Port 15176/udp was found to be open
Port 27017/tcp was found to be open    → MongoDB
Port 33655/tcp was found to be open
Port 39523/udp was found to be open
Port 45375/udp was found to be open
Port 48515/udp was found to be open
```

Hay varios puertos que están abiertos en mi pc y no se a que pertenecen. Para mirar el proceso al que están asociados:

```
[0]marina@debian:~$ sudo ss -tulnp | grep 10056
udp    UNCONN 0      0      0.0.0.0:10056      0.0.0.0:*
users: (("kdeconnectd",pid=3428,fd=23))
```

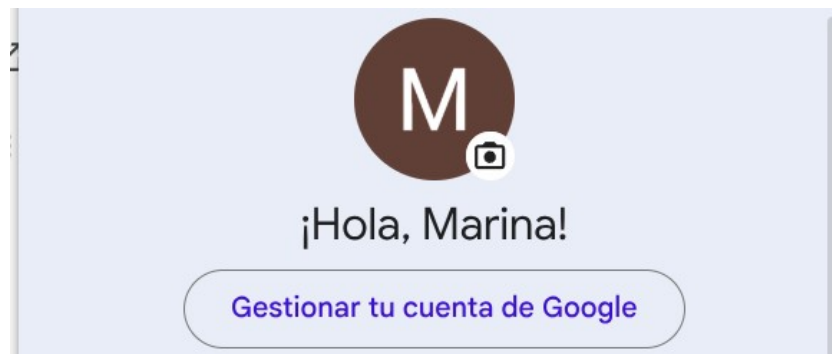
Todos los puertos corresponden con KDE. Esto me ha servido para saber que puertos de mi pc cerrar ya que yo no he instalado KDE en mi pc ni he configurado eso.

El resto de vulnerabilidades que aparecen en la lista nos proporcionan información como el hostname de la máquina, el FQDN.

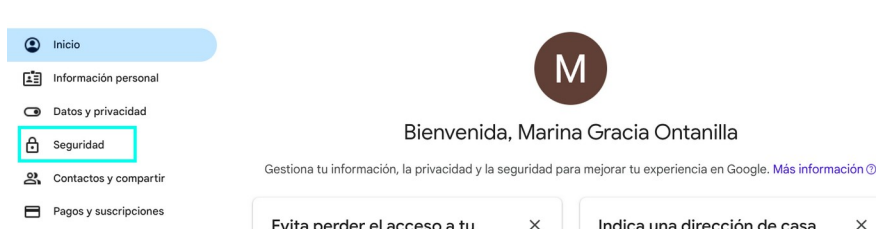
Alertas por correo electrónico.

Nuestro objetivo es configurar alertas automáticas que nos lleguen directamente al correo electrónico.

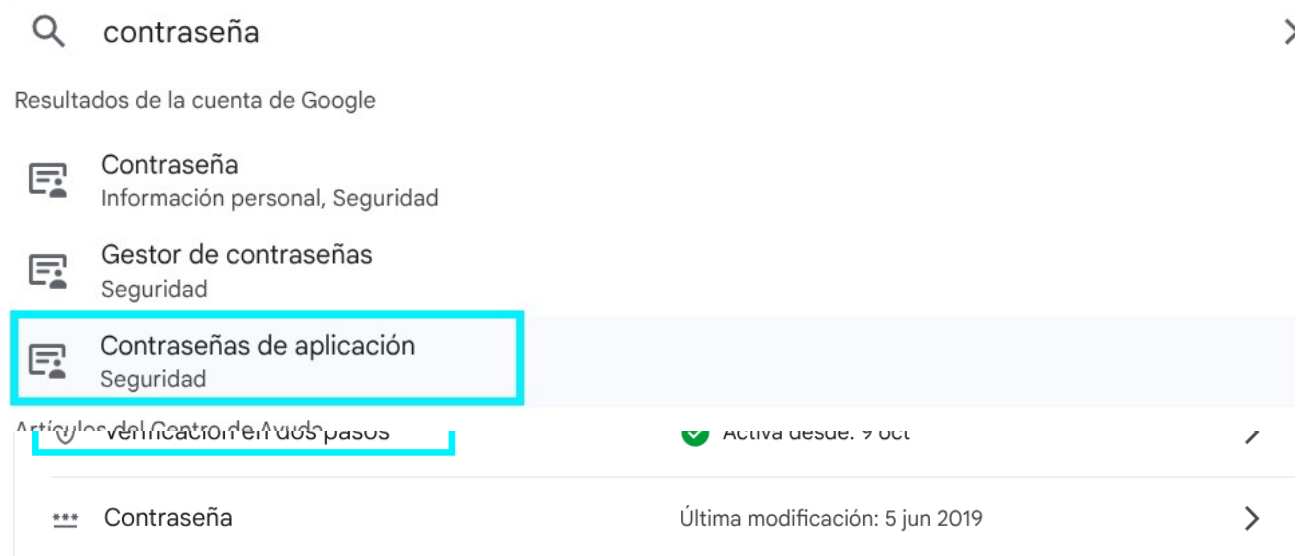
Voy a usar Postfix para configurar el servidor de correo de Nessus. Para empezar Vamos a configurar la certificación en dos pasos de nuestro correo:



Al meternos en 'Gestionar tu cuenta de Google', se nos abre una pestaña, donde nos podemos meter en el apartado de seguridad:



Una vez activada, vamos al buscador e introducimos 'contraseña de aplicaciones':



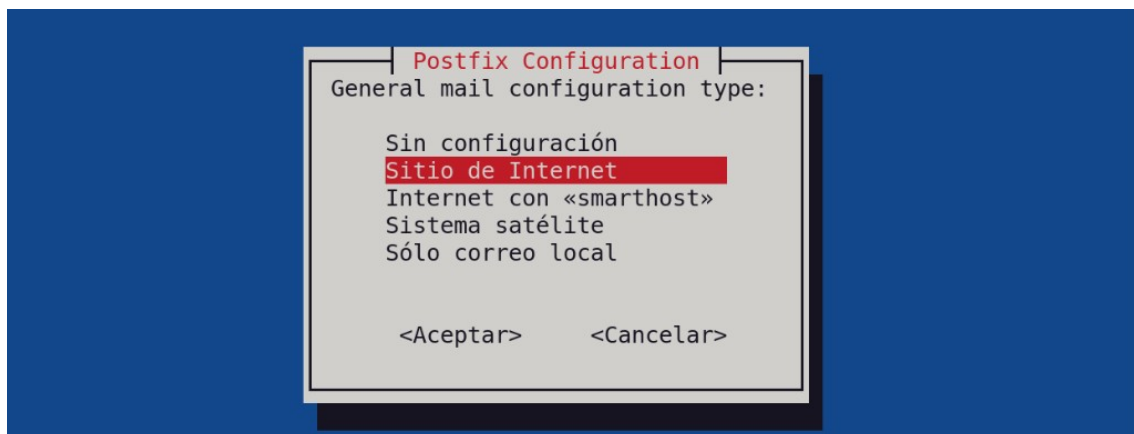
Para crear una contraseña específica de la aplicación, escribe el nombre de la aplicación a continuación...

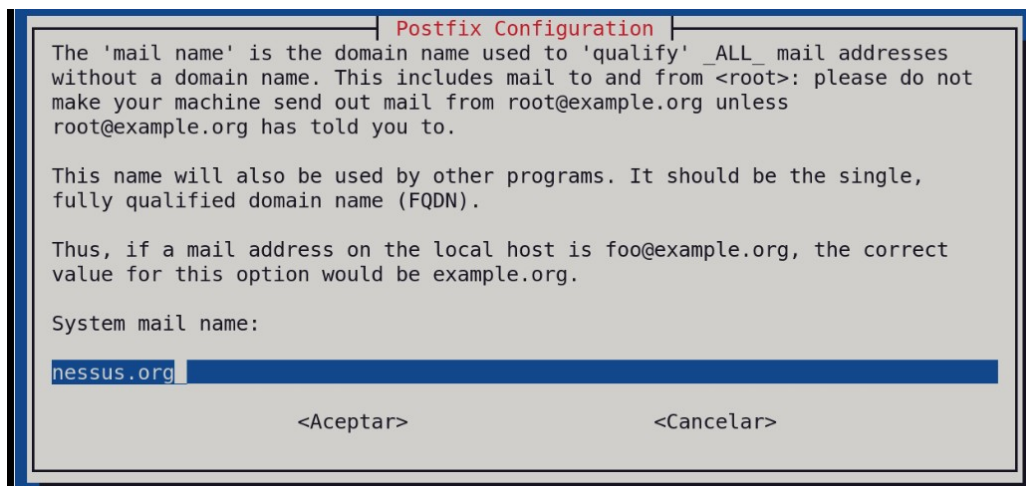
Nombre de la aplicación
Postfix

Nos genera una contraseña. Ahora instalamos postfix y lo configuramos:

```
nessus@debian:~$ sudo apt install postfix -y
```

En este apartado introducimos el FQDN de nuestra máquina.





Una vez instalado podemos proceder a configurarlo, para ello modificaremos el archivo:

```
GNU nano 8.4 /etc/postfix/main.cf
GNU nano 8.4
/etc/postfix/main.cf *
# =====
# CONFIGURACIÓN PRINCIPAL DE POSTFIX
# Adaptado para usar el SMTP de Gmail
# FQDN del servidor: nessus.org
# =====

# Nivel de compatibilidad (por defecto en versiones recientes de
Postfix)
compatibility_level = 3.9

# Nombre completo (FQDN) del servidor
myhostname = nessus.org

# Dominio que aparecerá como origen de los correos salientes
myorigin = /etc/mailname

# Interfaces en las que Postfix escuchará (todas)
inet_interfaces = all

# Protocolos IP (soporta IPv4 e IPv6)
inet_protocols = all

# Máquinas de confianza (solo este host podrá enviar)
mynetworks_style = host
mynetworks = 127.0.0.0/8 [::1]/128

# Destinos locales (dominios que este servidor considera
propios)
mydestination = $myhostname, localhost.$mydomain, localhost
```

```

# No limitar el tamaño del buzón (0 = ilimitado)
mailbox_size_limit = 0

# Separador entre nombre de usuario y extensión
(usuario+info@dominio)
recipient_delimiter = +

# Archivos de alias locales
alias_maps = hash:/etc/aliases
alias_database = hash:/etc/aliases

# Desactivar notificaciones tipo "biff"
biff = no

# =====
# CONFIGURACIÓN PARA ENVIAR CORREOS VIA GMAIL
# =====

# Servidor SMTP de Gmail y su puerto (STARTTLS en 587)
relayhost = [smtp.gmail.com]:587

# Activar el uso de TLS
smtp_use_tls = yes
smtp_tls_security_level = encrypt
smtp_tls_CAfile = /etc/ssl/certs/ca-certificates.crt
smtp_tls_session_cache_database =
btree:${data_directory}/smtp_scache

# Activar autenticación SASL para Gmail
smtp_sasl_auth_enable = yes

# Archivo con usuario y contraseña (creado por ti en
/etc/postfix/sasl_passwd)
smtp_sasl_password_maps = hash:/etc/postfix/sasl_passwd

# Opciones de seguridad SASL
smtp_sasl_security_options = noanonymous
smtp_sasl_tls_security_options = noanonymous
smtp_sasl_mechanism_filter = plain, login

# =====
# CONFIGURACIÓN DEL SERVIDOR LOCAL
# =====

# Banner de presentación cuando alguien conecta por SMTP
smtpd_banner = $myhostname ESMTTP $mail_name (Debian)

```

```
# Nivel de seguridad TLS del servidor local (no afecta al envío)
smtpd_tls_security_level = may
smtpd_tls_cert_file = /etc/ssl/certs/ssl-cert-snakeoil.pem
smtpd_tls_key_file = /etc/ssl/private/ssl-cert-snakeoil.key

# Restricciones básicas para el reenvío
smtpd_relay_restrictions = permit_mynetworks
permit_sasl_authenticated defer_unauth_destination

# =====
# FIN DEL ARCHIVO
# =====
```

A continuación, editamos, o creamos si no existe, el siguiente archivo:

```
nessus@debian:~$ sudo nano /etc/postfix/sasl_passwd
GNU nano 8.4 /etc/postfix/sasl_passwd *
[smtp.gmail.com]:587
marinagraciaontanilla@gmail.com:contraseña
```

En contraseña pondremos la contraseña de aplicación que se nos generó anteriormente. La contraseña aparece con espacios, en este fichero la colocamos sin espacios.

Guardamos el archivo y le cambiamos los permisos:

```
nessus@debian:~$ sudo postmap /etc/postfix/sasl_passwd
nessus@debian:~$ sudo chmod 400 /etc/postfix/sasl_passwd
```

postmap es para convertir el archivo en un formato de base de datos que Postfix puede leer.

Reiniciamos postfix:

```
nessus@debian:~$ sudo systemctl restart postfix
```

Mandamos un correo de prueba para verificar que se ha configurado correctamente:

```
nessus@debian:~$ echo "Esto es un correo de prueba desde
Nessus/Postfix" | mail -s "Prueba de correo"
mgracial411@gmail.com
nessus@debian:~$ echo "Esto es un correo de prueba desde
Nessus/Postfix" | mail -s "Prueba de correo"
marinagraciaontanilla@gmail.com
```

Ahora pasamos a configurar el SMTP de nessus:

Simple Mail Transfer Protocol (SMTP) is an industry standard for sending and receiving email. Once configured for SMTP, scan results will be emailed to the list of recipients specified in a scan's "Email Notifications" configuration. These results can be custom tailored through filters and require an HTML compatible email client.

Host:

Port:

From (sender email):

Encryption:

Hostname (for email links):

Auth Method:

En el host tenemos que poner el nombre de nuestro host, ya que si ponemos localhost nos da un error. Para ello debemos hacer el siguiente cambio en el /etc/hosts:

```
GNU nano 8.4 /etc/hosts
127.0.0.1    nessus.org
. . .
```

Puerto 25 porque Postfix escucha en ese puerto, que es el puerto estándar de SMTP. Esto es porque usamos Postfix local, si fuese gmail externo sería el 587.

En el correo debemos poner el que usamos en '/etc/postfix/sasl_passwd'.

Como Nessus habla con Postfix local, en encryption vamos a seleccionar NONE.

Hostname sirve para los enlaces dentro de los emails.

En auth method no ponemos nada, ya que Postfix acepta mensajes desde localhost sin credenciales.

A continuación, he programado un nuevo escáner que se ha ejecutado, y al que he configurado para que cuando termine me notifique por correo.

Settings | Credentials | Plugins

BASIC

- General
- Schedule
- Notifications**

DISCOVERY

ASSESSMENT

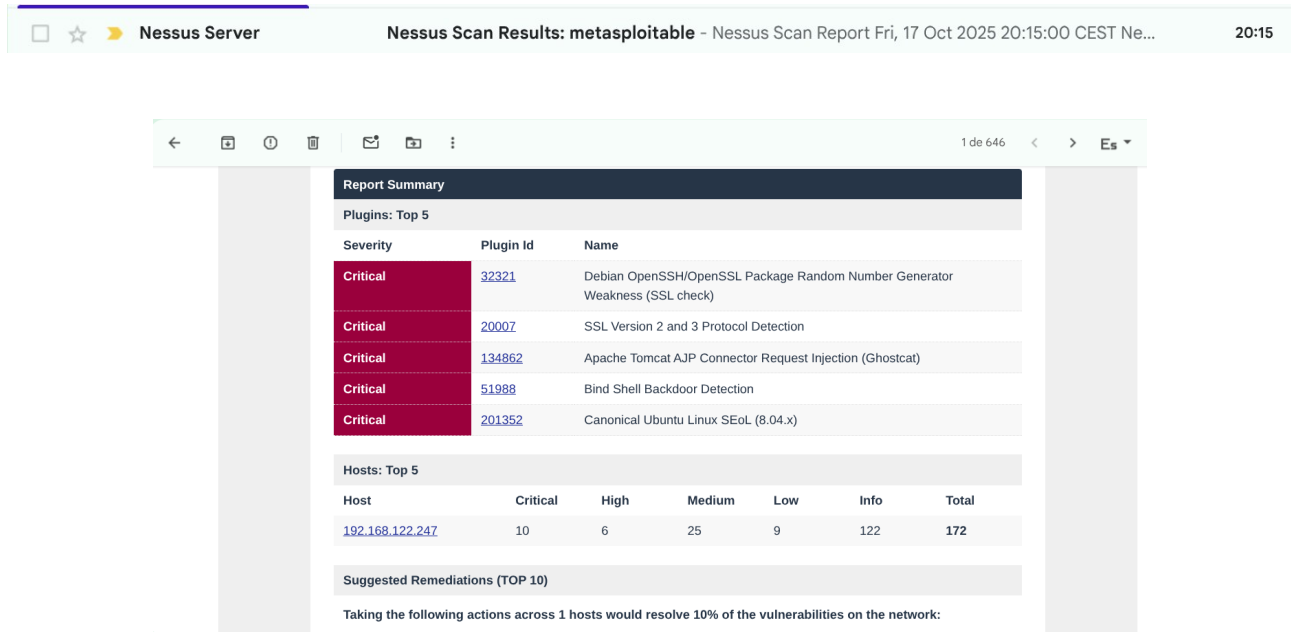
REPORT

ADVANCED

Email Recipient(s):

Result Filters:

Una vez finalizado el escáner, comprobamos nuestro correo:



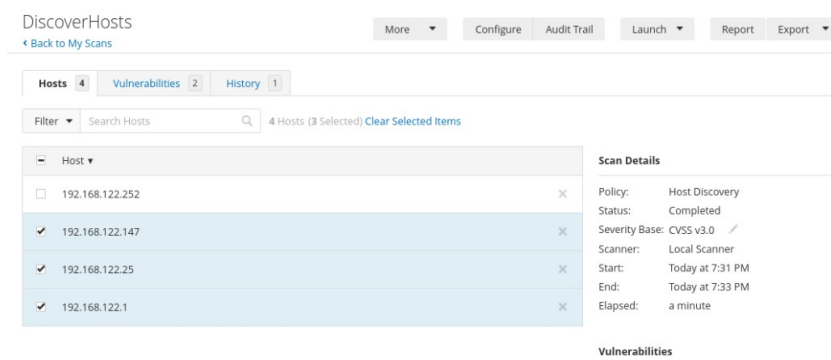
El escaneo evalúa los servicios y puertos abiertos del equipo, identificando posibles fallos de configuración, software desactualizado y debilidades de seguridad que podrían ser explotadas por un atacante.

Nessus analiza estos elementos mediante plugins especializados, los cuales comparan las versiones de los servicios detectados con bases de datos de vulnerabilidades conocidas.

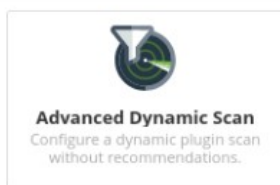
En este caso, el informe muestra un resumen con las vulnerabilidades más graves encontradas, clasificadas por nivel de severidad (Crítica, Alta, Media, Baja e Informativa).

Prueba de uso en red.

Para hacer una prueba de uso en red, vamos a seleccionar varios hosts que me han aparecido en un nuevo escaneo de descubrimiento de hosts. He creado varias máquinas virtuales para poder ejecutar el escáner con credenciales SSH y poder realizarlo completo.



Copiamos la clave publica en las tres máquinas y creamos un escáner nuevo como hemos hecho anteriormente:



En este caso, voy a elegir esta plantilla de escáner. Hace un escaneo que se adapta al host y la red mientras se ejecuta.

Configuramos:

Settings | Credentials | Dynamic Plugins

BASIC

- General
- Schedule
- Notifications

DISCOVERY

ASSESSMENT

REPORT

ADVANCED

Name: EscanoAjustable

Description: escaneo de red

Folder: My Scans

Targets: 192.168.1.47, 192.168.1.25, 192.168.1.11

Settings | Credentials | Dynamic Plugins

BASIC

- General
- Schedule
- Notifications

Email Recipient(s): mgracia1411@gmail.com

En el apartado “Dynamic Pluggins” podemos especificar la ID de vulnerabilidad que queremos que el escaneo busque.

New Scan / Advanced Dynamic Scan

[Back to Scan Templates](#)

Settings | Credentials | Dynamic Plugins

Match: All of the following:

CVE is equal to CVE-2021-44228

Preview Plugins

Debian Local Security Checks (3)

Plugin Name	Plugin ID
Debian DLA-2842-1 : apache-log4j2 - LTS security update	156018
Debian DSA-5020-1 : apache-log4j2 - security update	156015
Debian DSA-5022-1 : apache-log4j2 - security update	156124

Save Cancel

Esta ID es una vulnerabilidad conocida en Log4j que permite la ejecución remota de un código. Como podemos ver Nessus tiene pluggins dedicados para detectarla. Guardamos y lanzamos el escaneo.

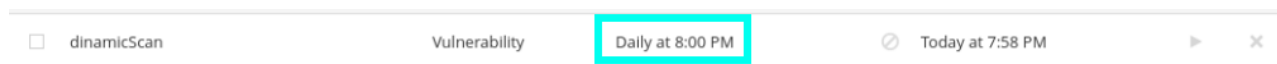


En la pestaña 'schedule' vamos a programarlo:

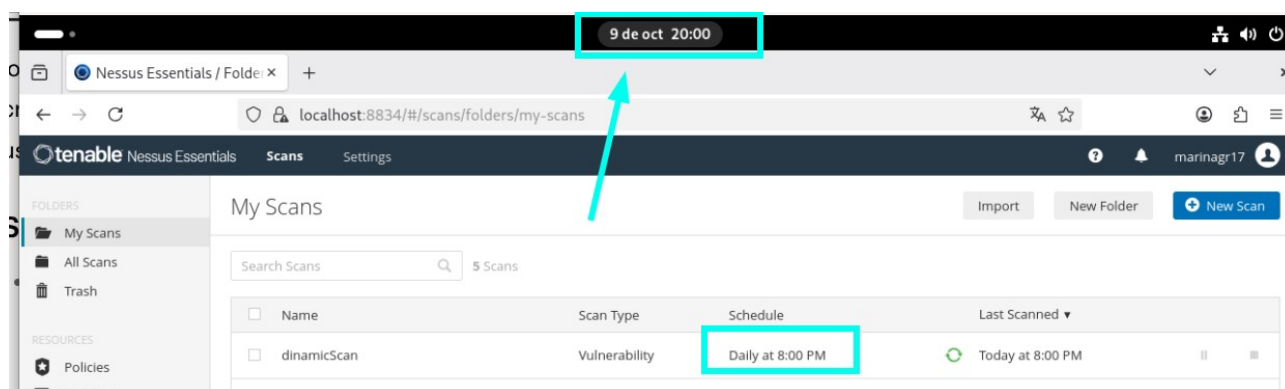
Solo puedo tener una programación porque tengo la versión gratuita de Nessus.

Configuramos las credenciales. Una para el pc con username: Marina y otra para los pc con username: debian.

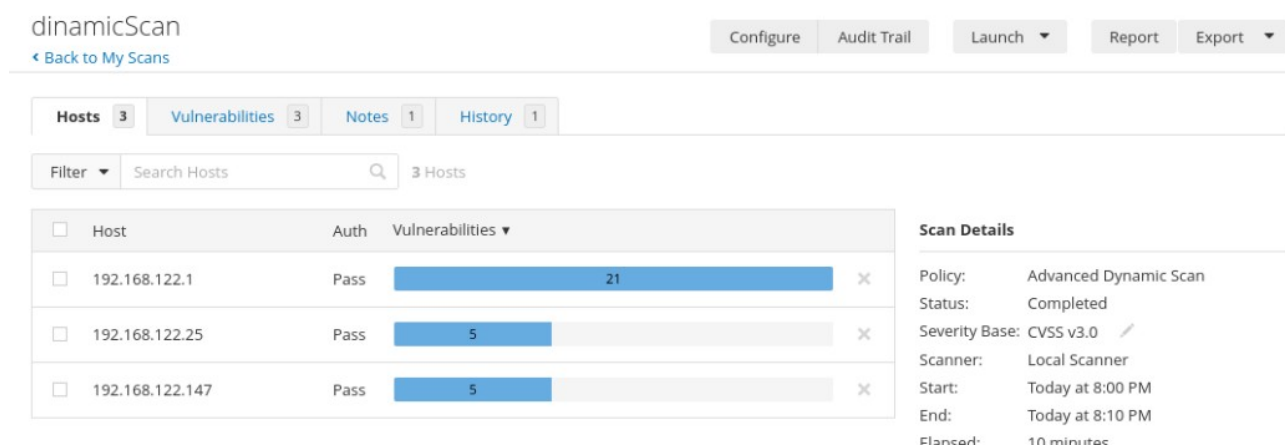
En la opción 'Targets to prioritize credentials' ponemos la ip de los hosts. Esto agiliza el proceso ya que no las busca, sino que se las indicamos.



Debe de lanzarse automáticamente:



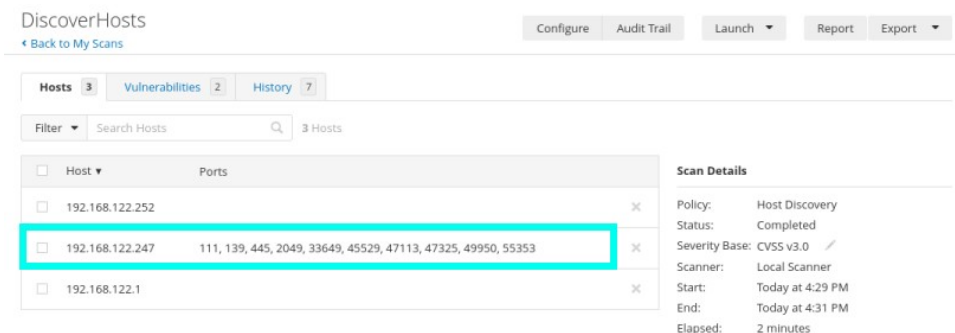
Lo ideal sería programar también un escáner de puertos para saber que puertos están en la red, pero como solo me deja programar un escáner y conozco todos los puertos que quiero escanear lo dejamos así.



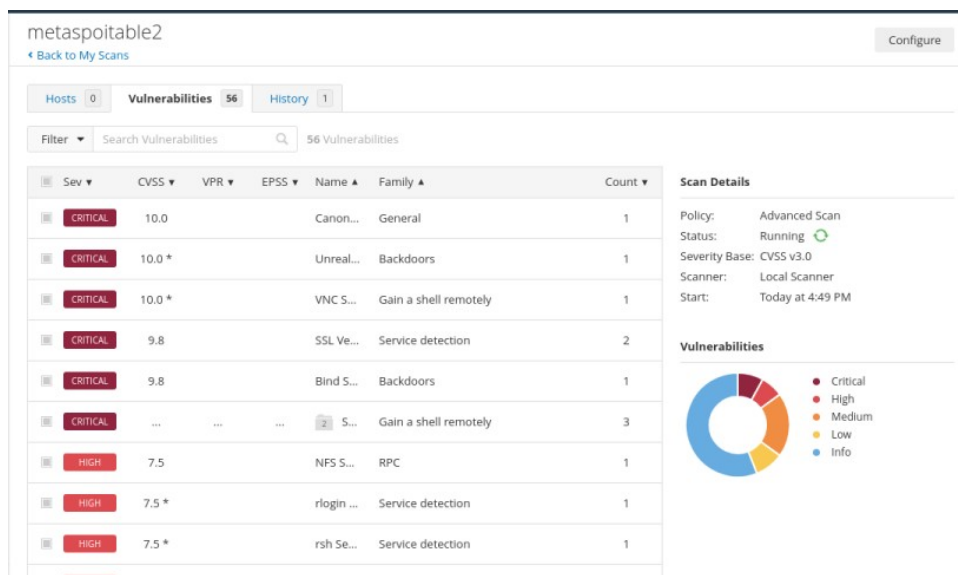
No ha encontrado vulnerabilidades, nos ha dado datos informativos.

Escáner en máquina metasploitable

Mediante un escáner de DiscoveryHost encontramos la ip de nuestra metasploitable:

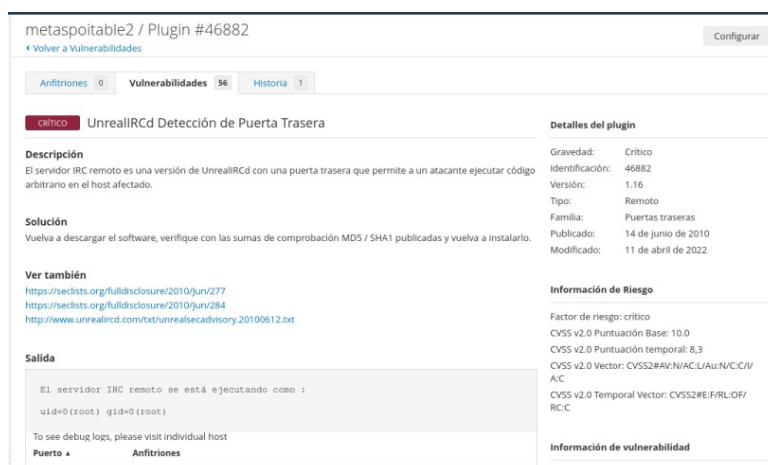


Una vez tenemos la dirección, lanzamos un nuevo escáner para ver las vulnerabilidades. Configuramos servidor de correo, pluggins que no hacen falta los desactivamos para no cargar tanto el escáner y que sea más rápido:



Como vemos tenemos varias vulnerabilidades, y las que faltan porque sigue en proceso.

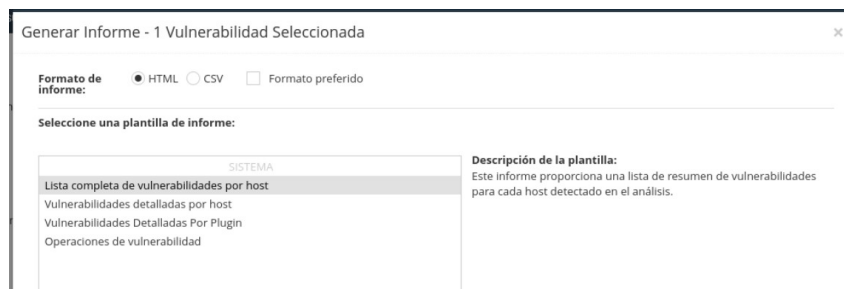
Si accedemos a las vulnerabilidades, nos aparece un informe:



Además de un informe, nos indica la solución del problema.

Informe.

Podemos exportar los informes del escáner:



Nos genera informes según clasifica las vulnerabilidades.

Entre las más destacadas aparecen:

- Debilidad en el generador de números aleatorios de OpenSSH/OpenSSL (Debian), que compromete la fortaleza de las claves criptográficas.
- Protocolos SSL 2.0 y 3.0 activos, que ya no son seguros y deben deshabilitarse.
- Vulnerabilidad "Ghostcat" en Apache Tomcat, que permite la inyección de peticiones a través del conector AJP.
- Puerta trasera detectada (Bind Shell Backdoor) que indica una posible intrusión.
- Sistema operativo Ubuntu 8.04.x sin soporte (EoL), lo que lo hace vulnerable al no recibir actualizaciones.

El host analizado (192.168.122.247) presenta un total de 172 vulnerabilidades, de las cuales 10 son críticas. El informe también incluye un apartado de remediaciones sugeridas, donde se recomiendan acciones para mitigar los riesgos detectados, como actualizar servicios (BIND, Samba), eliminar software malicioso o reinstalar aplicaciones comprometidas. En conjunto, este informe proporciona una visión general del estado de seguridad del sistema, permitiendo priorizar las correcciones necesarias para reducir la superficie de ataque.

Escaneo por plugin:

10205 (1) - detección de servicios de registro

Síntesis
El servicio de registro se ejecuta en el host remoto.

Descripción
El servicio de registro se ejecuta en el host remoto. Este servicio es vulnerable ya que los datos se pasan entre el cliente de registro y el servidor en texto claro. Un atacante de man-in-the-middle puede explotar esto para detectar inicios de sesión y contraseñas. Además, puede permitir inicios de sesión mal autenticados sin contraseñas. Si el host es vulnerable a la adivinación del número de secuencia TCP (desde cualquier red) o a la suplantación de IP (incluido el secuestro de ARP en una red local), entonces puede ser posible evitar la autenticación. Finalmente, login es una manera fácil de convertir el acceso de escritura de archivos en inicios de sesión completos a través de los archivos .rhosts o rhosts.equiv.

Solución
Comenta la línea 'login' en /etc/inetd.conf y reinicia el proceso inetd. Alternativamente, desactive este servicio y use SSH en su lugar.

Factor de riesgo
Alto

CVSS v2.0 Puntuación de base
7.5 (CVSS2#AV:N/AC:L/Au:C/P:II/A:P)

Referencias

Escaneo por host:

192.168.122.247

0

CRITICAL

1

HIGH

0

MEDIUM

0

LOW

0

INFO

Scan Information

Start time:

Thu Oct 16 19:07:24 2025

End time:

Thu Oct 16 19:26:11 2025

Host Information

IP:

192.168.122.247

MAC Address:

52:54:00:CA:AD:4E

OS:

Linux Kernel 2.6 on Ubuntu 8.04 (hardy)

Vulnerabilities

10205 - rlogin Service Detection -