GOBIERNO DE ESPAÑA
VICEPRESIDENCIA TERCERA DEL GOBIERNO
MINISTERIO DE ASUNTOS ECONÓMICOS Y TRANSFORMACIÓN DIGITAL
SECRETARÍA DE ESTADO DE DIGITALIZACIÓN E INTELIGENCIA ARTIFICIAL

Webinar: Linux hardening basic

Exercises

017 TU AYUDA EN CIBERSEGURIDAD
incibe_ INSTITUTO NACIONAL DE CIBERSEGURIDAD

INDEX

FRAME INDEX

1. PRACTICAL EXERCISE

The objective of the exercise is to configure the boot and firewall of a server with the default installation.

The following configuration is required for the startup:

- The server must ask for a password to start.

Firewall setup:

- Deny all incoming and outgoing traffic.
- Allow incoming traffic to DNS, HTTP, HTTPS, SNMP and SSH services.
- Allow outgoing traffic to DNS and SYSLOG services.

Resolution of the exercise:

To configure the boot, first create a password hash with the command:

- `grub-mkpasswd-pbkdf2`

Requests a password, enter for example "*iPSK=BZ]aav*El^N*" and it returns a string.

The output of the command would be a string like the following:

```
Enter password:

Reenter password:

PBKDF2 hash of your password is
grub.pbkdf2.sha512.10000.FB11E8E745C23174644A5A14726ABA1883A296AB181DEFA
33055AE739B44D91022D7EB5CDF4A2B5568EF0959220319C1BD2BB82E6D760BA84D55F95
CFDBCA86E.D23381F3EEB6E7B1F19230DFDBA2209EA0551365B13A36711CC1079E36A3D0
1494DC796BD5F6D94057E1A72FD629D5BA567A47343D985246667584BE45427FB3
```

Create and edit the file */etc/grub.d/init-pwd* and add the following lines:

```
cat <<EOF

set superusers="root"

password_pbkdf2 root
grub.pbkdf2.sha512.10000.FB11E8E745C23174644A5A14726ABA1883A296AB181DEFA
33055AE739B44D91022D7EB5CDF4A2B5568EF0959220319C1BD2BB82E6D760BA84D55F95
CFDBCA86E.D23381F3EEB6E7B1F19230DFDBA2209EA0551365B13A36711CC1079E36A3D0
1494DC796BD5F6D94057E1A72FD629D5BA567A47343D985246667584BE45427FB3

EOF
```

Save and we give you execution permits:

```
chmod +x /etc/grub.d/init-pwd
```

To configure the FW, to deny all traffic we must execute the following commands:

```
ufw default deny incoming
```

```
ufw default deny outgoing
ufw default deny routed
```

To enable services on the server: DNS, HTTP, HTTPS, SNMP and SSH

```
ufw allow in 53/tcp para DNS
ufw allow in 53/udp para DNS
ufw allow in 80/tcp para HTTP
ufw allow in 443/tcp para HTTPS
ufw allow in 161/udp para SNMP
ufw allow in 22/tcp para SSH
```

And finally to enable access to DNS and SYSLOG services:

```
ufw allow out to any port 53
ufw allow out to any port 514
```

2. RESEARCH INVESTIGATION

Given the following sudo configuration file, the basic user incibe would have root permissions to execute only the command */usr/bin/vim*. Could the user incibe obtain a command console as *root* and execute any command as such? If possible, what measures should be taken to avoid this type of vulnerability?

```
# User privilege specification
root    ALL=(ALL:ALL) ALL
incibe ALL=(ALL:ALL) /usr/bin/vim

# Members of the admin group may gain root privileges
%admin ALL=(ALL) ALL

# Allow members of group sudo to execute any command
%sudo   ALL=(ALL:ALL) ALL
```

Frame 1 Sudo configuration

- **Hint**: Review options for the /usr/bin/vim command

Resolution of the exercise:

The user incibe has permissions to execute the binary */usr/bin/vim* with elevation of privileges. "Vim" is a text editor that allows the option to execute a command console from it. To do this, execute the command:

- `sudo /usr/bin/vim prueba.txt`

Once in the text editor we run:

- `:sh`

And we get a command console as root.

Another option is to edit the file /etc/shadow with:

- `sudo /usr/bin/vim /etc/shadow`

And directly change the root password, elevate privileges with your and get interactive console with the *root* user.

To avoid these types of vulnerabilities, you should always ensure that the command or commands that you allow a user to execute as *root*, does not allow you to obtain dynamic *shells* or a parameter that allows you to execute commands, or edit sensitive system files.