INTRODUCCIÓN AL HARDENING LINUX

TALLER PRÁCTICO

HackOn 2023

ÍNDICE

0 **¿Quiénes somos?**
¡Conócenos!

1 **Introducción**
Qué es el hardening y por qué es importante

2 **Arquitectura**
Diseño del laboratorio

3 **Comprobaciones**
Puntos de control

4 **Tests de hardening**
Desarrolla tus tests

5 **Scripts de hardening**
Ahora securiza tus sistemas

6 **Tests de nuevo**
¿Funcionan ya?

7 **Soluciones**
Así lo hemos hecho nosotros

¿QUIÉNES SOMOS?

Ingenieros en Ciberseguridad

Javier Sánchez

Cloud Security Architect

@Javi_sanchez04
kaputt4.github.io

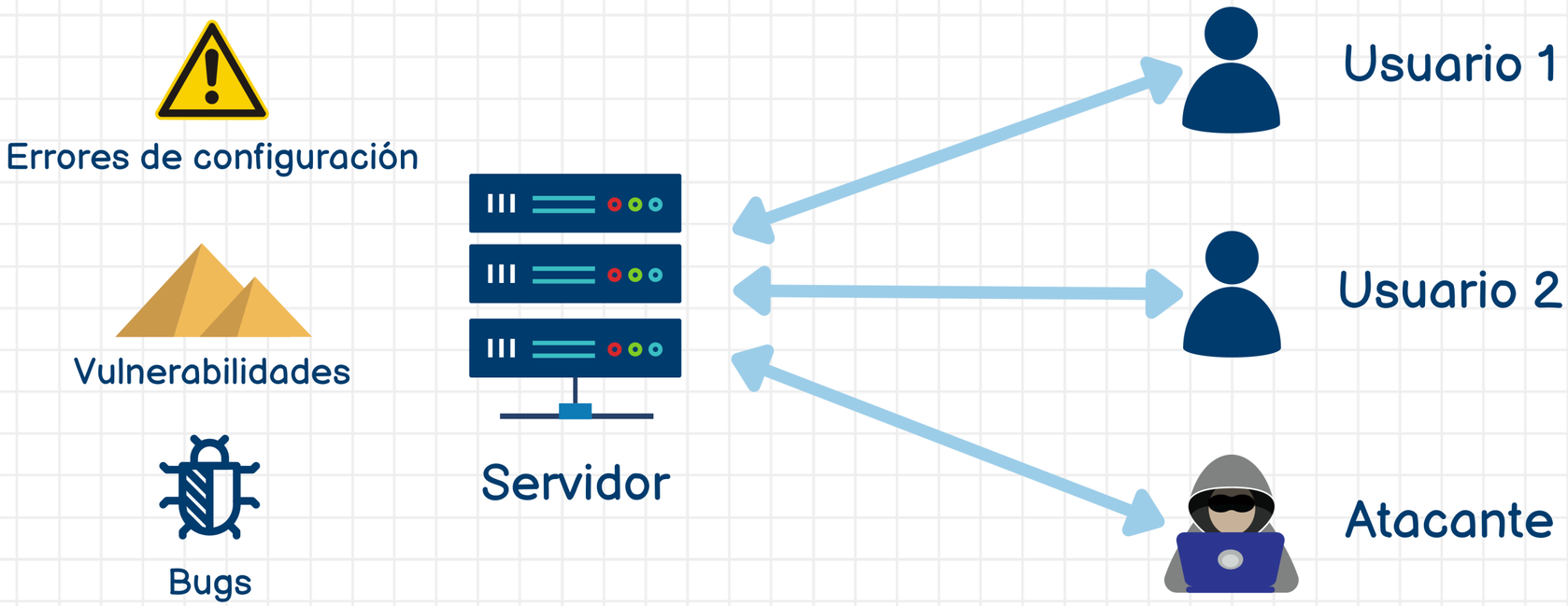
Alejandro Bermejo

Cybersecurity Engineer at GMV

@R00tedSec

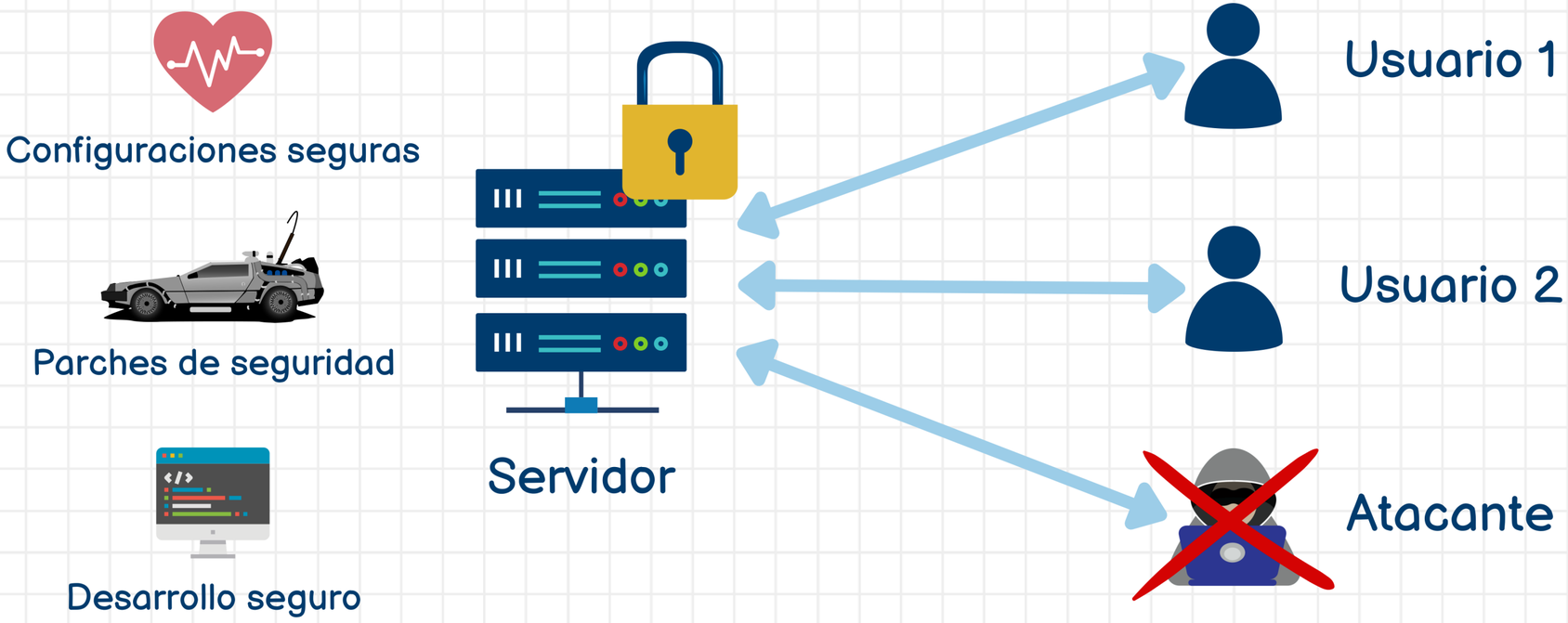
PONGAMOS UN EJEMPLO

Disponemos de un sistema donde alojamos nuestro servidor web



PROTECCIÓN

Necesitamos aplicar medidas de protección



HARDENING

Conjunto de actividades que se realizan en un sistema con el objetivo de protegerlo

- DISPONIBILIDAD
- CONFIDENCIALIDAD
- INTEGRIDAD
- CONTROL DE ACCESO
- NO REPUDIO

TAREAS DE HARDENING

Cerrar puertos
Desinstalar servicios
Eliminar usuarios
Modificar configuraciones
Actualizar paquetes
...

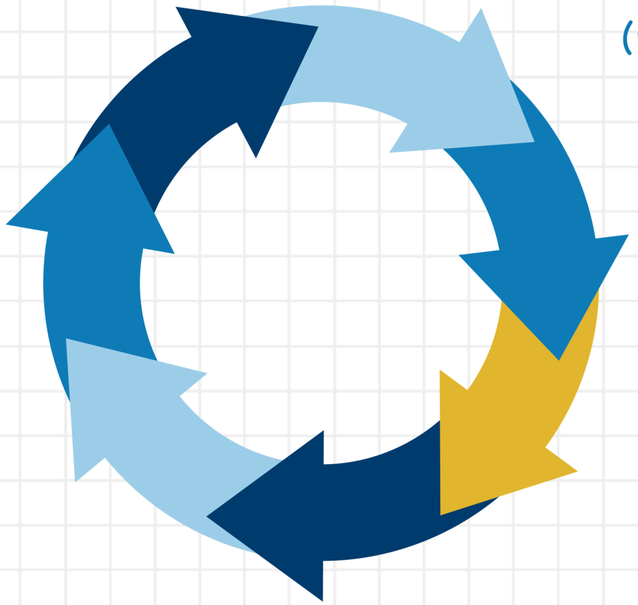
CICLO DE HARDENING

Fase 1

Evaluación del sistema a proteger

Fase 2

Búsqueda de información (Guías de hardening, configuraciones, herramientas ....)



Fase 3.A

Generacion de los test de hardening (verificación)

Fase 3.B

Generación de los script de hardening

Fase 4

Ejecución de los script de hardening

Fase 5

Verificación del estado de hardening

CICLO DE HARDENING

Tests

Verificar el estado de seguridad de cada sistema.

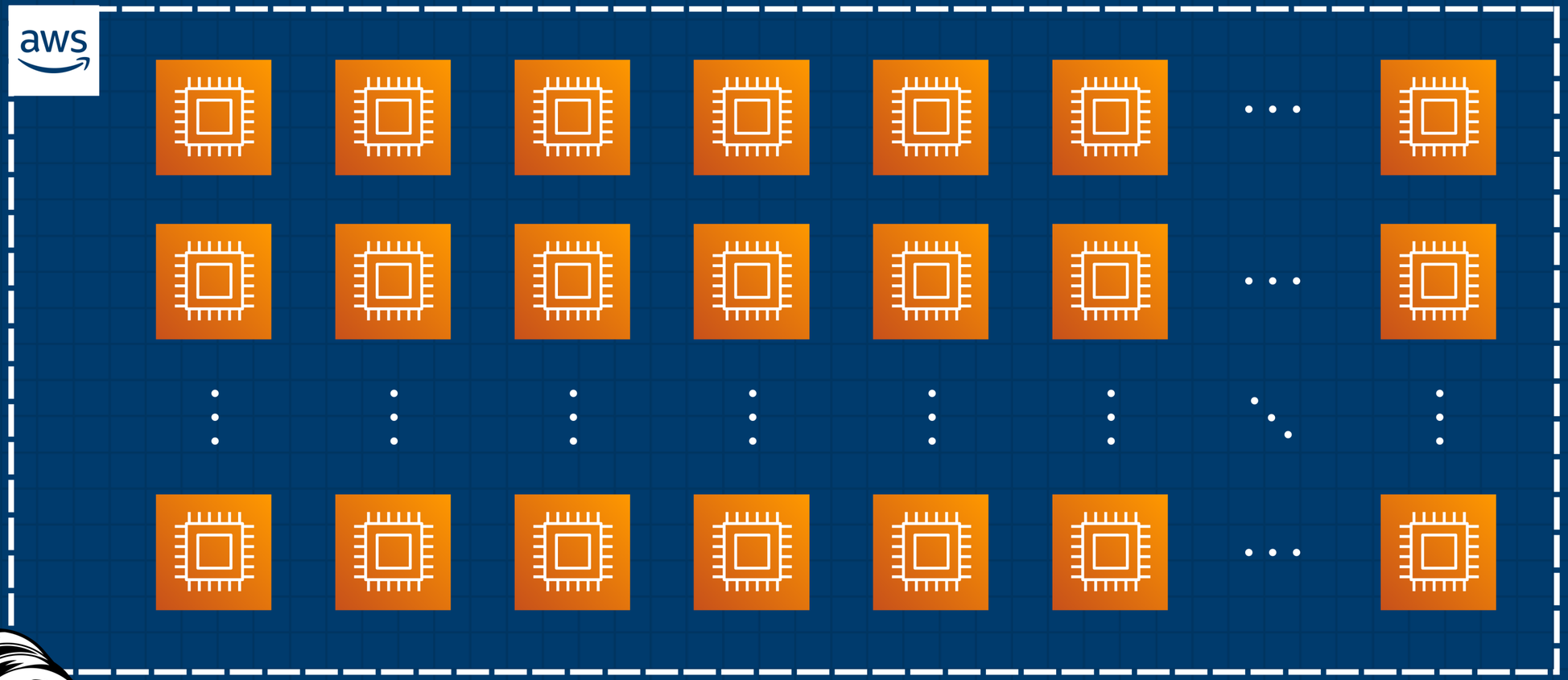
Ej.: comprobar que las configuraciones se mantienen seguras, los paquetes actualizados, los puertos cerrados....

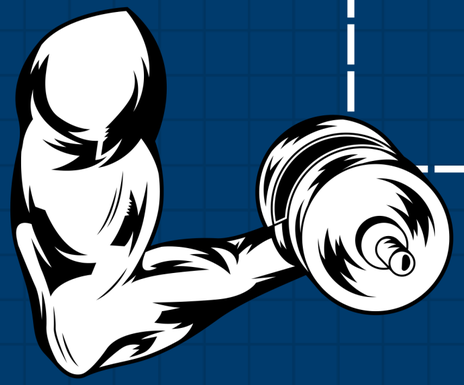
Scripts

Aplicar las medidas de seguridad necesarias.

Ej.: cambios en configuraciones, instalación de servicios de log, despliegue de servicios de monitorización...

Si tenemos una cantidad elevada de servidores...
¿quién se ofrece a securizarlos uno a uno a mano?





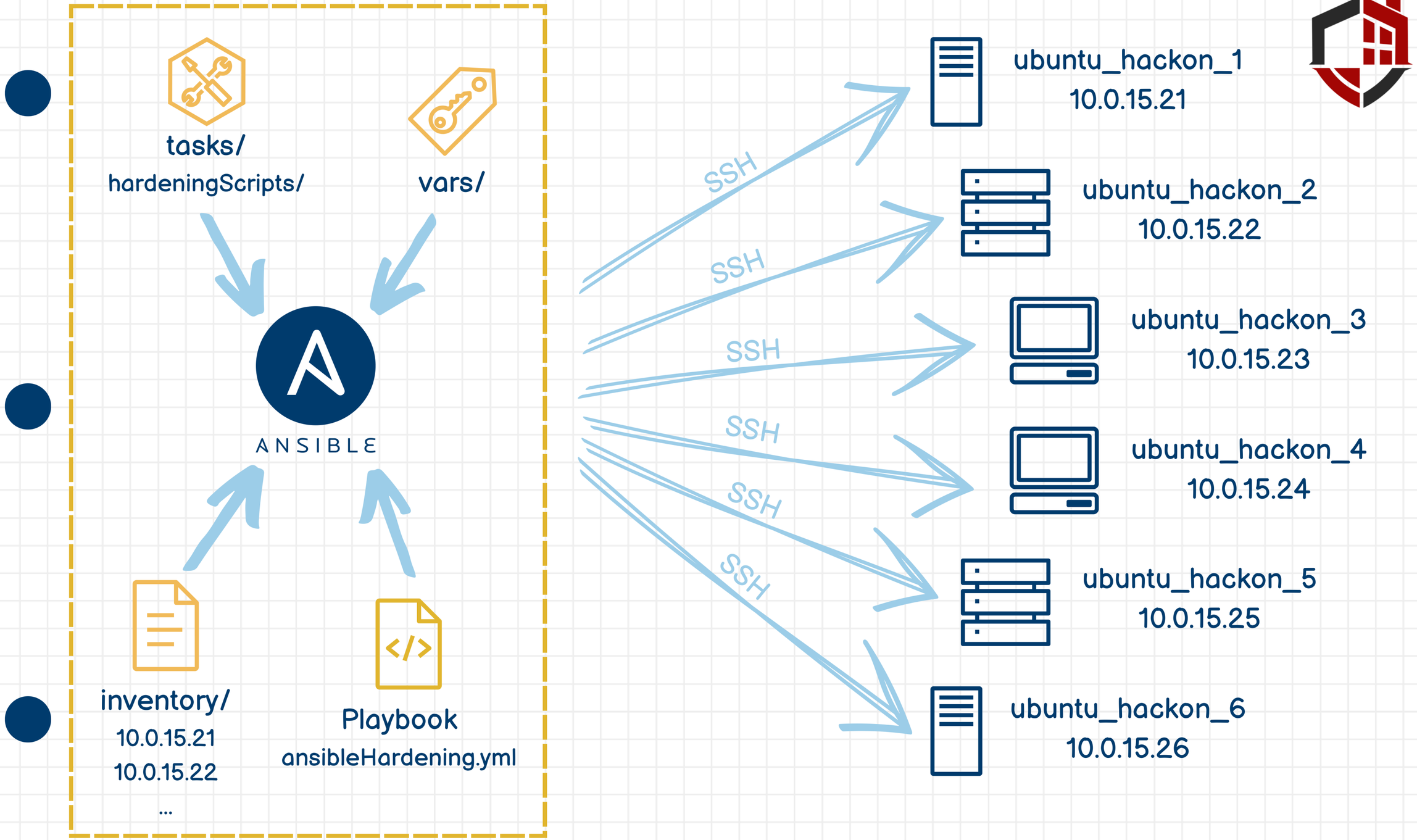
HERRAMIENTAS AUTOMATICAS

Tests: Lynis

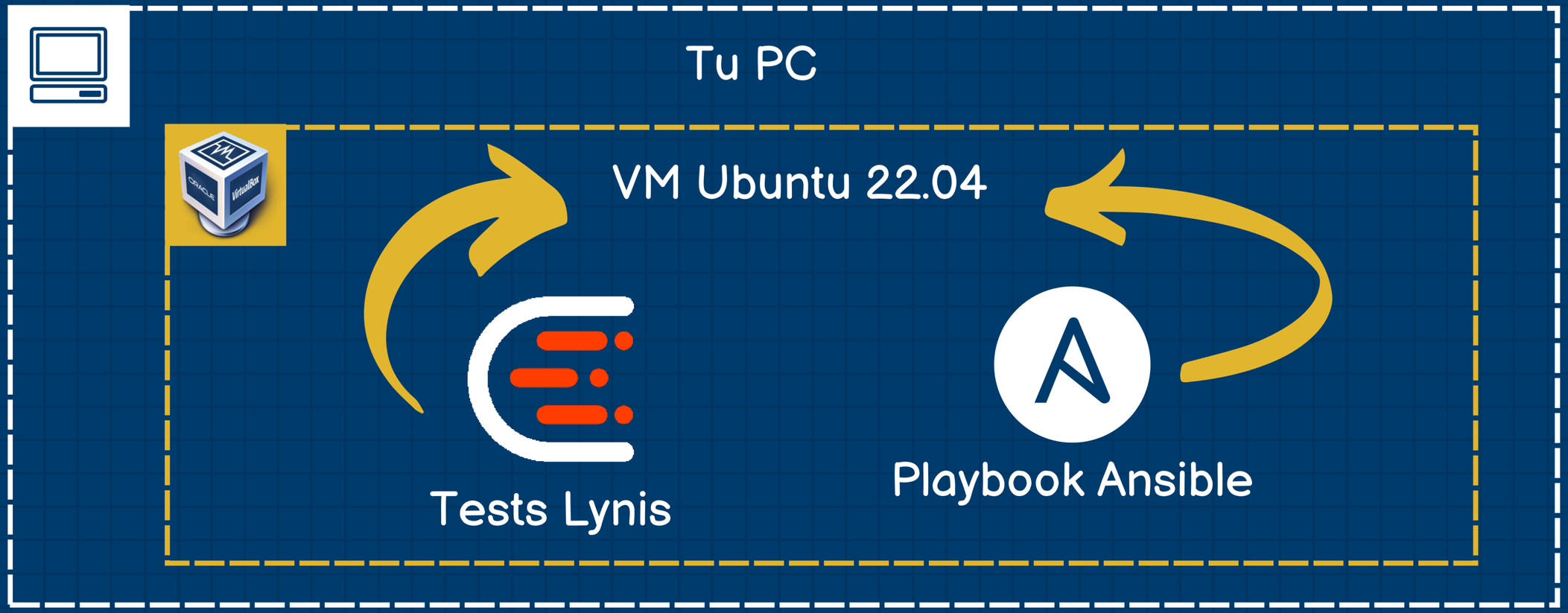
Herramienta Open Source desarrollada por Michael Boelen que se utiliza para realizar auditorías, ver el nivel de configuración de activos, pruebas de cumplimiento...

Scripts: Ansible

Herramienta para realizar acciones en distintos sistemas sin instalar agentes ni dependencias, de manera idempotente y pudiendo diferenciar por grupos de hosts, SOs...
Desarrollada por Michael DeHaan y adquirida por RedHat en 2015, es Open Source


tasks/
hardeningScripts/
vars/
ANSIBLE
inventory/
10.0.15.21
10.0.15.22
...
Playbook
ansibleHardening.yml
SSH
SSH
SSH
SSH
SSH
SSH
ubuntu_hackon_1
10.0.15.21
ubuntu_hackon_2
10.0.15.22
ubuntu_hackon_3
10.0.15.23
ubuntu_hackon_4
10.0.15.24
ubuntu_hackon_5
10.0.15.25
ubuntu_hackon_6
10.0.15.26

LABORATORIO



VM OVA: https://hackon.es/talleres/Alejandro_y_Javi.html

PUNTOS DE CONTROL

/etc/apache2/apache2.conf

/etc/apache2/mods-enabled/ssl.conf

Firewall

Puertos

/etc/passwd

$_

root login

Log level

TCP Forwarding

LINUX HARDENING

Firewall

Puertos

/etc/passwd

TESTS LINUX – LYNIS – EJEMPLO

Objetivo

Verificar que los permisos del archivo /etc/passwd son solo de lectura y lectura y escritura para el propietario (644)

```
# Test        : Linux-0001
# Description : Ensure permissions on /etc/passwd are configured
# Reference   : https://www.tenable.com/...

Register --test-no Linux-0000 --weight L --network NO --category security --description "Ensure ..."
if [ ${SKIPTEST} -eq 0 ]; then
    PERMISSIONS=$(stat -c "%a" /etc/passwd)
    if [ "$PERMISSIONS" = "644" ]; then
        Display --indent 2 --text "- /etc/passwd permissions" --result "${STATUS_OK}" --color GREEN
    else
        Display --indent 2 --text "- /etc/passwd permissions" --result "${STATUS_ERROR}" --color RED
    fi
fi
```

TESTS LINUX – ANSIBLE – EJEMPLO

Objetivo

Modificar los permisos de dicho
archivo para que sean 644 (-rw-r--r--)

```
ubuntu@ubuntu-2204:~/Desktop/HackOn2023/ansibleScripts$ ls -la /etc/passwd
-rwxrwxrwx 1 root root 3028 Jan  8 13:22 /etc/passwd
```

```yaml
- name: Correct permissions on /etc/passwd (644)
  ansible.builtin.file:
    path: /etc/passwd
    owner: root
    group: root
    mode: '0644'
```

TESTS LINUX – LYNIS – 1

Objetivo

Verificar que la expiración de las contraseñas
de los usuarios es de 365 días o menos

```
# Test        : Linux-0002
# Description : Ensure password expiration is 365 days or less - users
# Reference   : https://www.tenable.com/...

Register --test-no Linux-0002 --weight L --network NO --category security --description "Ensure ..."
if [ ${SKIPTEST} -eq 0 ]; then
    # CODE
fi
```

TESTS LINUX – LYNIS – 1

Solución

MAX_DAYS=$(grep "^ *PASS_MAX_DAYS" /etc/login.defs | awk '{printf $2}')

$MAX_DAYS -lt "366"

```
# Test        : Linux-0002
# Description : Ensure password expiration is 365 days or less - users
# Reference   : https://www.tenable.com/...

Register --test-no Linux-0002 --weight L --network NO --category security --description "Ensure ..."
if [ ${SKIPTEST} -eq 0 ]; then
    MAX_DAYS=$(grep "^ *PASS_MAX_DAYS" /etc/login.defs | awk '{print $2}')
    if [ $MAX_DAYS -lt "366" ];then
        Display --indent 2 --text "- Ensure password expiration is 365 days or less" --result "${STATUS_OK}" --color GREEN
    else
        Display --indent 2 --text "- Ensure password expiration is 365 days or less" --result "${STATUS_ERROR}" --color RED
    fi
fi
```

TESTS LINUX – ANSIBLE – 1

Solución

Modificar el archivo /etc/login.defs , cambiar el valor del parámetro PASS_MAX_DAYS a 365

```
- name: Ensure password expiration and /etc/passwd permissions
  # CODE
```

TESTS LINUX – ANSIBLE – 1

Solución

Modificar el archivo /etc/login.defs , cambiar el valor del parámetro PASS_MAX_DAYS a 365

```yaml
- name: Ensure password expiration and /etc/passwd permissions
  lineinfile:
    path: /etc/login.defs
    state: present
    regexp: '^PASS_MAX_DAYS'
    line: 'PASS_MAX_DAYS 365'
```

TESTS LINUX – LYNIS – 2

Objetivo

Comprobar que el firewall (UFW) está funcionando y bloqueando todos los puertos salvo el 22 y 443

```
# Test        : Linux-0003
# Description : Ensure there is a firewall running and blocking all ports except 22 and 443

Register --test-no Linux-0003 --weight L --network NO --category security --description "Ensure ..."
if [ ${SKIPTEST} -eq 0 ]; then
    # CODE
fi
```

TESTS LINUX – LYNIS – 2

Solución

```
# Test        : Linux-0003
# Description : Ensure there is a firewall running and blocking all ports except 22 and 443

Register --test-no Linux-0003 --weight L --network NO --category security --description "Ensure ..."
if [ ${SKIPTEST} -eq 0 ]; then
  FIREWALL=$(ufw status)
  if [ $? -eq 127 ];then
    Display --indent 2 --text "- UFW not installed" --result "${STATUS_ERROR}" --color RED
  else
    FIREWALL_ENABLED=$(ufw status)
    if [ "$FIREWALL_ENABLED" = "Status: inactive" ];then
      Display --indent 2 --text "- Firewall disabled" --result "${STATUS_ERROR}" --color RED
    else
      FIREWALL_OPENPORTS=$(ufw status | grep ALLOW | cut -d " " -f1-2  | grep -v "v6" | cut -d "/" -f1 | uniq)
      DISALLOWED_PORT=0
      for i in $FIREWALL_OPENPORTS; do
        if [ "$i" ≠ "443" -a "$i" ≠ "22" ];then
          Display --indent 2 --text "- Allowed $i port " --result "${STATUS_ERROR}" --color RED
          DISALLOWED_PORT=1
        fi
      done
      if [ $DISALLOWED_PORT -eq 0 ];then
       Display --indent 2 --text "- Firewall enabled and correctly configured" --result "${STATUS_OK}" --color GREEN
      fi
    fi
  fi
fi
```

TESTS LINUX – ANSIBLE – 2

```yaml
- name: Reset firewall
  # CODE

- name: Start UFW
  # CODE

- name: Open port 22
  # CODE

- name: Open port 443
  # CODE
```

Solución

Resetear y activar UFW, y luego abrir puertos 22 y 443

TESTS LINUX – ANSIBLE – 2

```yaml
- name: Reset firewall
  community.general.ufw:
    state: reset

- name: Start UFW
  community.general.ufw:
    state: enabled

- name: Open port 22
  community.general.ufw:
    rule: allow
    port: '22'
    proto: tcp

- name: Open port 443
  community.general.ufw:
    rule: allow
    port: '443'
    proto: tcp
```

Solución

Resetear y activar UFW, y luego abrir puertos 22 y 443

SSH HARDENING

$_

root login

Log level

TCP Forwarding

SSH – LYNIS – EJEMPLO

Objetivo

Evitar que un atacante se pueda conectar con las credenciales de ROOT

```
# Test        : SSH-0001
# Description : Ensure SSH root login is disabled
# Reference   : https://www.tenable.com/audits/items/CIS_Ubuntu_22.04_LTS_v1.0.0_Server_L1.audit:13f14fcee36064a2ef722bb4d40f20d7

if [ ${SSH_RUNNING} -eq 1 ]; then PREQS_MET="YES"; SKIPREASON=""; else PREQS_MET="NO"; SKIPREASON="SSH not installed, or not running"; fi
Register --test-no SSH-0001 --preqs-met ${PREQS_MET} --skip-reason "${SKIPREASON}" --weight L --network NO --category security --description "Ensure SSH root login is disabled"
if [ ${SKIPTEST} -eq 0 ]; then
    SSH_ROOT=$(grep "^\s*PermitRootLogin no\s*$" /etc/ssh/sshd_config)
    if [ -z "$SSH_ROOT" ];then
        Display --indent 2 --text "- SSH root login enabled" --result "${STATUS_ERROR}" --color RED
    else
        Display --indent 2 --text "- SSH root login disabled" --result "${STATUS_OK}" --color GREEN
    fi
fi
```

SSH – ANSIBLE– EJEMPLO

Solución

Modificar el contenido del fichero /etc/ssh/sshd_config, añadiendo la línea PermitRootLogin no

```yaml
- name: Ensure SSH root login is disabled
  lineinfile:
    path: /etc/ssh/sshd_config
    state: present
    regexp: '^PermitRootLogin'
    line: 'PermitRootLogin no'
```

SSH – LYNIS– 1

Objetivo

Evitar que un atacante consiga usar ssh para crear túneles ssh.

```
# Test        : SSH-0002
# Description : Ensure SSH AllowTcpForwarding is disabled
# Reference   : https://www.tenable.com/...

if [ ${SSH_RUNNING} -eq 1 ]; then PREQS_MET="YES"; SKIPREASON=""; else PREQS_MET="NO"; SKIPREASON="SSH not installed, or not running"; fi
Register --test-no SSH-0002 --preqs-met ${PREQS_MET} --skip-reason "${SKIPREASON}" --weight L --network NO --category security --description "Ensure SSH AllowTcpForwarding is disabled"
if [ ${SKIPTEST} -eq 0 ]; then
    # CODE
fi
```

SSH – LYNIS– 1

Solución

Buscar en el fichero /etc/ssh/sshd_config

```
# Test        : SSH-0002
# Description : Ensure SSH AllowTcpForwarding is disabled
# Reference   : https://www.tenable.com/...

if [ ${SSH_RUNNING} -eq 1 ]; then PREQS_MET="YES"; SKIPREASON=""; else PREQS_MET="NO"; SKIPREASON="SSH not installed, or not running"; fi
Register --test-no SSH-0002 --preqs-met ${PREQS_MET} --skip-reason "${SKIPREASON}" --weight L --network NO --category security --description "Ensure SSH AllowTcpForwarding is disabled"
if [ ${SKIPTEST} -eq 0 ]; then
    ALLOW_TCP=$(grep "^\s*AllowTcpForwarding no\s*$" /etc/ssh/sshd_config)
    if [ -z "$ALLOW_TCP" ];then
        Display --indent 2 --text "- AllowTcpForwarding enabled" --result "${STATUS_ERROR}" --color RED
    else
        Display --indent 2 --text "- AllowTcpForwarding disabled" --result "${STATUS_OK}" --color GREEN
    fi
fi
```

SSH – ANSIBLE – 1

Objetivo

Cambiar el valor de AllowTcpForwarding

```
- name: Ensure SSH AllowTcpForwarding is disabled
  # CODE
```

SSH – ANSIBLE – 1

Solución

Modificar el contenido del fichero /etc/ssh/sshd_config, añadiendo la línea AllowTcpForwarding no

```yaml
- name: Ensure SSH AllowTcpForwarding is disabled
  lineinfile:
    path: /etc/ssh/sshd_config
    state: present
    regexp: '^AllowTcpForwarding'
    line: 'AllowTcpForwarding no'
```

SSH – LYNIS– 2

Objetivo

Comprobar que el nivel de logging de SSH es el correcto

```
# Test        : SSH-0003
# Description : Ensure SSH LogLevel is appropriate
# Reference   : https://www.tenable.com/...

if [ ${SSH_RUNNING} -eq 1 ]; then PREQS_MET="YES"; SKIPREASON=""; else PREQS_MET="NO"; SKIPREASON="SSH not installed, or not running"; fi
Register --test-no SSH-0003 --preqs-met ${PREQS_MET} --skip-reason "${SKIPREASON}" --weight L --network NO --category security --description "Ensure SSH LogLevel is appropriate"
if [ ${SKIPTEST} -eq 0 ]; then
    # CODE
fi
```

SSH – LYNIS– 2

Solución

Comprobar que el nivel de logging de SSH es el correcto

```
# Test        : SSH-0003
# Description : Ensure SSH LogLevel is appropriate
# Reference   : https://www.tenable.com/...

if [ ${SSH_RUNNING} -eq 1 ]; then PREQS_MET="YES"; SKIPREASON=""; else PREQS_MET="NO"; SKIPREASON="SSH not installed, or not running"; fi
Register --test-no SSH-0003 --preqs-met ${PREQS_MET} --skip-reason "${SKIPREASON}" --weight L --network NO --category security --description "Ensure SSH LogLevel is appropriate"
if [ ${SKIPTEST} -eq 0 ]; then
    LOG_LEVEL=$(grep "^\s*LogLevel VERBOSE\s*$" /etc/ssh/sshd_config)
    if [ -z "$LOG_LEVEL" ];then
        Display --indent 2 --text "- LogLevel not set to VERBOSE" --result "${STATUS_ERROR}" --color RED
    else
        Display --indent 2 --text "- LogLevel set to VERBOSE" --result "${STATUS_OK}" --color GREEN
    fi
fi
```

SSH – ANSIBLE – 2

Objetivo

1. Configurar el parámetro *LogLevel* de ssh a *VERBOSE*
2. Reiniciar ssh

```yaml
- name: Ensure SSH LogLevel is appropriate
  # CODE

- name: Apply ssh changes
  # CODE
```

SSH – ANSIBLE – 2

Solución

Modificar el contenido del fichero */etc/ssh/sshd_config*, añadiendo la línea *LogLevel VERBOSE* y reiniciar el servicio *sshd*

```yaml
- name: Ensure SSH LogLevel is appropriate
  lineinfile:
    path: /etc/ssh/sshd_config
    state: present
    regexp: '^LogLevel'
    line: 'LogLevel VERBOSE'

- name: Apply ssh changes
  ansible.builtin.service:
    name: sshd
    state: restarted
```

APACHE HARDENING

/etc/apache2/apache2.conf

/etc/apache2/mods-enabled/ssl.conf

APACHE– LYNIS – EJEMPLO

Objetivo

1. Comprobar que el servicio *apache2* está activo *(se proporciona el test APACHE-000)*
2. Evitar que el servicio *apache2* corra como usuario root

```
# Test        : APACHE-0001
# Description : Ensure the Apache Web Server Runs As a Non-Root user and group → rootuser, rootuser
# Reference   : https://www.tenable.com/...

Register --test-no APACHE-0001 --weight L --network NO --category security --description "Checking Apache Web Server Runs As a Non-Root user and group"
if [ ${SKIPTEST} -eq 0 ]; then
    APACHE_USER=$(grep "^\s*User\s*www-data\s*$" /etc/apache2/apache2.conf)
    APACHE_GROUP=$(grep "^\s*Group\s*www-data\s*$" /etc/apache2/apache2.conf)
    if [ ! -z "$APACHE_USER" ] && [ ! -z "$APACHE_GROUP" ];then
        Display --indent 2 --text "- Apache runs as a Non-Root user and group" --result "${STATUS_OK}" --color GREEN
    else
        Display --indent 2 --text "- Apache does not run as a Non-Root user and group" --result "${STATUS_ERROR}" --color RED
    fi
fi
```

APACHE – ANSIBLE– EJEMPLO

Solución

Modificar el contenido del fichero /etc/apache/apache2.conf, cambiando los campos "User" and "Group" al usuario "www-data"

```yaml
- name: Ensure the Apache Web Server Runs As a Non-Root user
  lineinfile:
    path: /etc/apache2/apache2.conf
    state: present
    regexp: '^User'
    line: 'User www-data'
- name: Ensure the Apache Web Server Runs As a Non-Root group
  lineinfile:
    path: /etc/apache2/apache2.conf
    state: present
    regexp: '^Group'
    line: 'Group www-data'
```

APACHE- LYNIS - 1

Objetivo

Comprobar que Apache no usa protocolos inseguros como SSLv2, SSLv3 y TLSv1

```
# Test        : APACHE-0002
# Description : Ensure Weak SSL Protocols Are Disabled (SSLv2, SSLv3 and TLSv1) → TLSv1
# Reference   : https://www.tenable.com/...

Register --test-no APACHE-0002 --weight L --network NO --category security --description "Checking ..."
if [ ${SKIPTEST} -eq 0 ]; then
    # CODE
fi
```

APACHE- LYNIS - 1

Solución

Buscar la configuración del parámetro SSLProtocol en /etc/apache2/mods-enabled/ssl.conf

```
# Test        : APACHE-0002
# Description : Ensure Weak SSL Protocols Are Disabled (SSLv2, SSLv3 and TLSv1) → TLSv1
# Reference   : https://www.tenable.com/...

Register --test-no APACHE-0002 --weight L --network NO --category security --description "Checking ..."
if [ ${SKIPTEST} -eq 0 ]; then
    PROTOCOL=$(grep "^\s*SSLProtocol\s*-all\s*+TLSv1.2\s*$" /etc/apache2/mods-enabled/ssl.conf)
    if [ ! -z "$PROTOCOL" ];then
        Display --indent 2 --text "- Weak SSL Protocols Are Disabled" --result "${STATUS_OK}" --color GREEN
    else
        Display --indent 2 --text "- Weak SSL Protocols Are Enabled" --result "${STATUS_ERROR}" --color
RED fi
fi
```

APACHE – ANSIBLE– 1

Solución

Modificar el contenido del fichero
/etc/apache2/mods-enabled/ssl.conf

```
- name: Ensure Weak SSL Protocols Are Disabled (SSLv2, SSLv3 and TLSv1)
  # CODE
```

APACHE – ANSIBLE– 1

Solución

Modificar el contenido del fichero /etc/apache2/mods-enabled/ssl.conf, modificando la linea SSLProtocol a '-all +TLSv1.2'

```yaml
- name: Ensure Weak SSL Protocols Are Disabled (SSLv2, SSLv3 and TLSv1)
  lineinfile:
    path: /etc/apache2/mods-enabled/ssl.conf
    state: present
    regexp: '^\s*SSLProtocol'
    line: '        SSLProtocol  -all +TLSv1.2'
```

APACHE- LYNIS – 2

Objetivo

Evitar que el tiempo que se mantiene activa una conexión sea menos de 15 segundos y de esta forma evitar posibles denegaciones de servicio

```
# Test        : APACHE-0003
# Description : Ensure KeepAliveTimeout is Set to a Value of 15 or less
# Reference   : https://www.tenable.com/...

Register --test-no APACHE-0003 --weight L --network NO --category security --description "Checking ..."
if [ ${SKIPTEST} -eq 0 ]; then
    # CODE
fi
```

APACHE- LYNIS – 2

Solución

Buscar el parámetro *KeepAliveTimeout /etc/apache2/apache2.conf*

```
# Test        : APACHE-0003
# Description : Ensure KeepAliveTimeout is Set to a Value of 15 or less
# Reference   : https://www.tenable.com/...

Register --test-no APACHE-0003 --weight L --network NO --category security --description "Checking ..."
if [ ${SKIPTEST} -eq 0 ]; then
    KEEP_ALIVE=$(grep "^\s*KeepAliveTimeout" /etc/apache2/apache2.conf  | cut -d " " -f2 )
    if [ $KEEP_ALIVE -lt "16" ];then
        Display --indent 2 --text "- KeepAliveTimeout is set to a value of 15 or less" --result "${STATUS_OK}" --color GREEN
    else
        Display --indent 2 --text "- KeepAliveTimeout is not set to a value of 15 or less" --result "${STATUS_ERROR}" --color RED
    fi
fi
```

APACHE – ANSIBLE– 2

Objetivo

1. Cambiar el parámetro que define el máximo tiempo de inactividad de las conexiones
2. Reiniciar Apache para aplicar todos los cambios

```yaml
- name: Ensure KeepAliveTimeout is Set to a Value of 15 or less
  # CODE

- name: Apply apache changes
  # CODE
```

APACHE – ANSIBLE– 2

Solución

Modificar el contenido del fichero /etc/apache/apache2.conf, modificando la opción KeepAliveTimeout a 15 o menos

```yaml
- name: Ensure KeepAliveTimeout is Set to a Value of 15 or less
  lineinfile:
    path: /etc/apache2/apache2.conf
    state: present
    regexp: '^KeepAliveTimeout'
    line: 'KeepAliveTimeout 15'

- name: Apply apache changes
  ansible.builtin.service:
    name: apache2
    state: restarted

```

SOLUCIONES Y EJEMPLOS REALES

https://github.com/R00tedSec/HackOn-Workshop

https://github.com/Kaputt4/hardening_server

¡GRACIAS!

¿Preguntas, casos de uso, curiosidades...?