Práctica 11. Hardening con *SELinux* y *AppArmor*

Seguridad de Sistemas Informáticos

Carlos Pérez Fino, Cheuk Kelly Ng Pante & Cristopher Manuel Afonso Mora

18 de diciembre de 2023

Índice general

1. Instalación de las máquinas virtuales

Para esta práctica se va a instalar dos máquinas virtuales, una *Ubuntu Server 16.04* y una *CentOS 7* para estudiar las configuraciones de hardening del sistema operativo basados en un control de acceso MAC implementadas en estos sistemas con *AppArmor* y *SELinux* respectivamente.

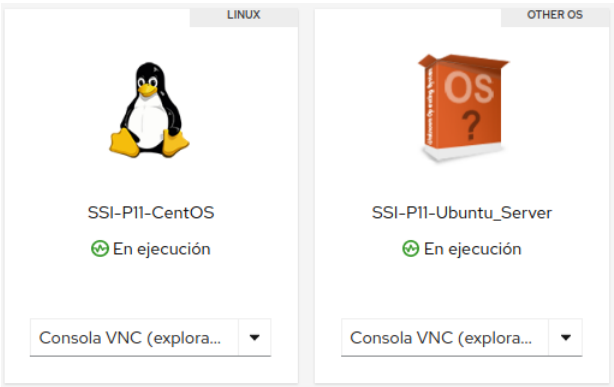

Figura 1.1: Instalación de las máquinas virtuales

2. Instalación y configuración de apache para la aplicación vulnerable *DVWA*

2.1. Instalación en Ubuntu Server 16.04

El primer paso es, antes de instalar Damn Vulnerable Web Application (DVWA) en la máquina virtual de Ubuntu Server 16.04, es instalar el servidor web Apache, el gestor de base de datos MariaDB y PHP. Para ello, ejecutamos los siguientes comandos:

```
$ sudo apt-get update
$ sudo apt-get install -y apache2 libapache2-mod-php
$ sudo apt-get install -y mariadb-server mariadb-client
$ sudo apt-get install -y php php-mysqli php-gd
```

Una vez instalado los paquetes, vamos a inicializar el servicio de Apache con el siguiente comando: `sudo service apache2 start`

y a ejecutar el script de configuración de MariaDB con el siguiente comando: `sudo mysql_secure_installation`

Y ahora vamos a instalar DVWA descargandolo desde el repositorio oficial de DVWA en GitHub. Para ello, primero instalamos git y clonamos el repositorio de DVWA en el directorio */var/www/html*. Lo pasos de Instalación son los siguientes:

```
$ sudo apt-get install git
$ git clone https://github.com/digininja/DVWA.git
$ sudo mv DVWA/ /var/www/html/
```

2.2. Instalación en CentOS 7

Para instalar DVWA en CentOS 7, primero instalamos el servidor web Apache, el gestor de base de datos MariaDB y PHP. Para ello, ejecutamos los siguientes comandos:

```
$ sudo yum update
$ sudo yum install -y httpd mariadb-server mariadb php php-mysql php-gd
```

Una vez instalado los paquetes, vamos a inicializar el servicio de Apache con el siguiente comando: `sudo systemctl start httpd.service`

y a ejecutar el script de configuración de MariaDB con el siguiente comando: `sudo mysql_secure_installation`

Y ahora vamos a instalar DVWA descargandolo desde el repositorio oficial de DVWA en GitHub. Para ello, primero instalamos git y clonamos el repositorio de DVWA en el directorio */var/www/html*. Lo pasos de Instalación son los siguientes:

```
$ sudo yum install git
$ git clone https://github.com/digininja/DVWA.git
$ sudo mv DVWA/ /var/www/html/
```

2.3. Configuración de DVWA

Para empezar la configuración de DVWA se va a acceder al directorio */var/www/html/DVWA* y copiamos el archivo *config.inc.php.dist* a *config.inc.php* con el siguiente comando:

```
$ sudo cp config/config.inc.php.dist config/config.inc.php
```

Ahora, vamos a configurar las variables de configuración de DVWA en el archivo *config.inc.php*. Para ello, ejecutamos el siguiente comando:

```
$_DVWA[ 'db_server' ]   = getenv('DB_SERVER') ?: '127.0.0.1';
$_DVWA[ 'db_database' ] = 'dvwa';
$_DVWA[ 'db_user' ]     = 'dvwa';
$_DVWA[ 'db_password' ] = 'ssi12345';
$_DVWA[ 'db_port' ]     = '3306';
```

Y ahora vamos a crear la base de datos *dvwa* y el usuario *dvwa* con el siguiente comando:

```
$ sudo mysql -u root -p
mysql> create database dvwa;
Query OK, 1 row affected (0.00 sec)

mysql> create user dvwa@localhost identified by 'ssi12345';
Query OK, 0 rows affected (0.01 sec)

mysql> grant all on dvwa.* to dvwa@localhost;
Query OK, 0 rows affected (0.01 sec)

mysql> flush privileges;
Query OK, 0 rows affected (0.00 sec)
```

3. Realización de vulnerabilidades

- **Creación de una carpeta temporal en el directorio *`/var/tmp`***:

  `10.6.129.208; mkdir /var/tmp/prueba`

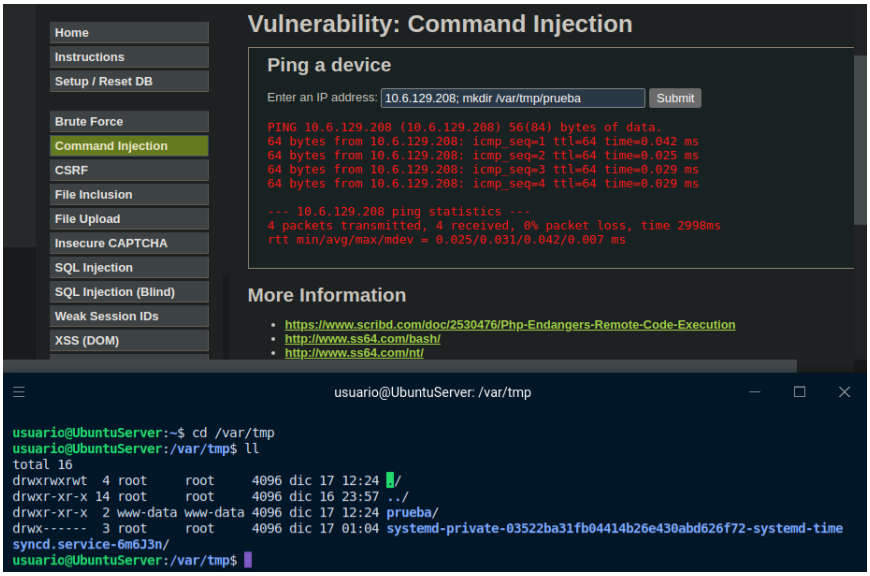

Figura 3.1: Creación de un fichero temporal en el directorio */var/tmp*

- **Creación de un archivo en la raíz web**:

  `10.6.129.208; echo "Hola, este es mi archivo" | sudo tee /var/www/html/miarchivo.html`

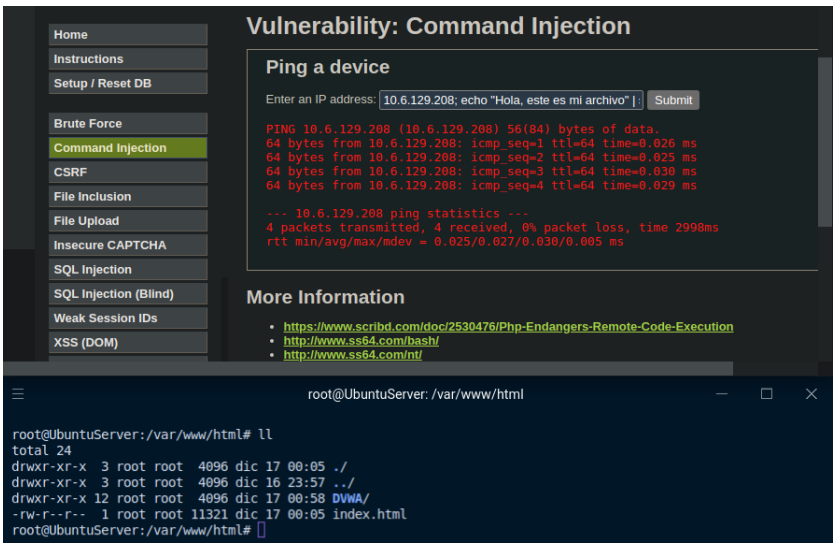

Figura 3.2: Creación de un archivo en la raíz web

- **Inyección de comandos (con éxito)**:

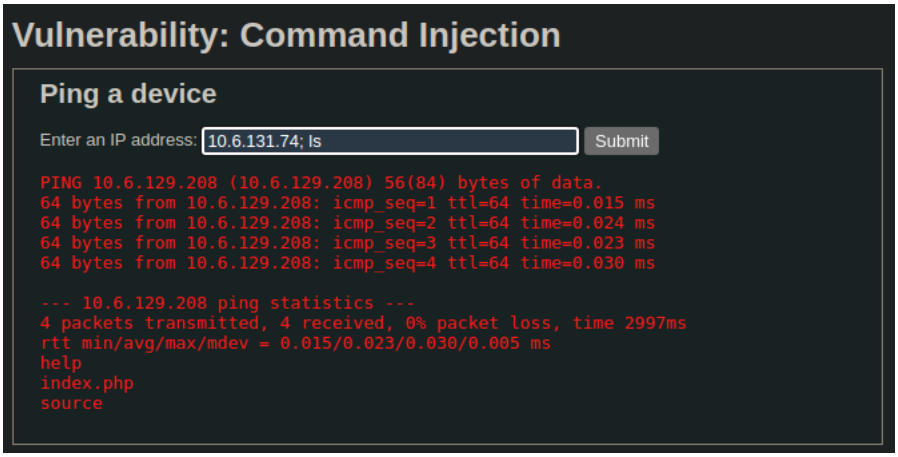

Figura 3.3: Inyección del comando *ls* con éxito

- **cat *`/etc/passwd`***:

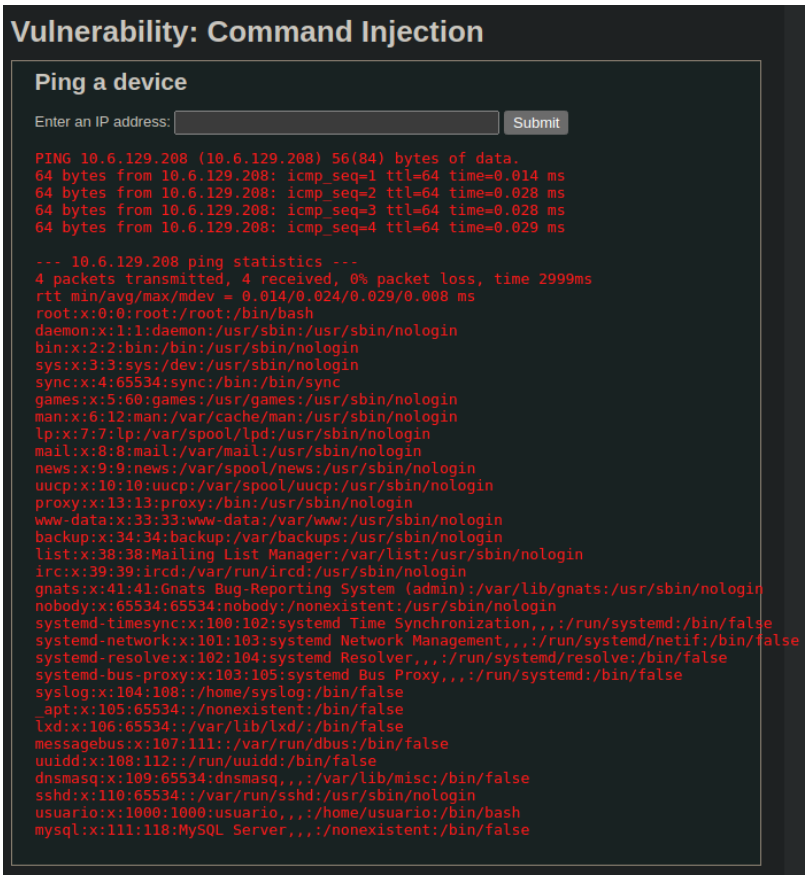

Figura 3.4: Visualización del contenido del archivo */etc/passwd*

- **Obtener un fichero por *wget***:

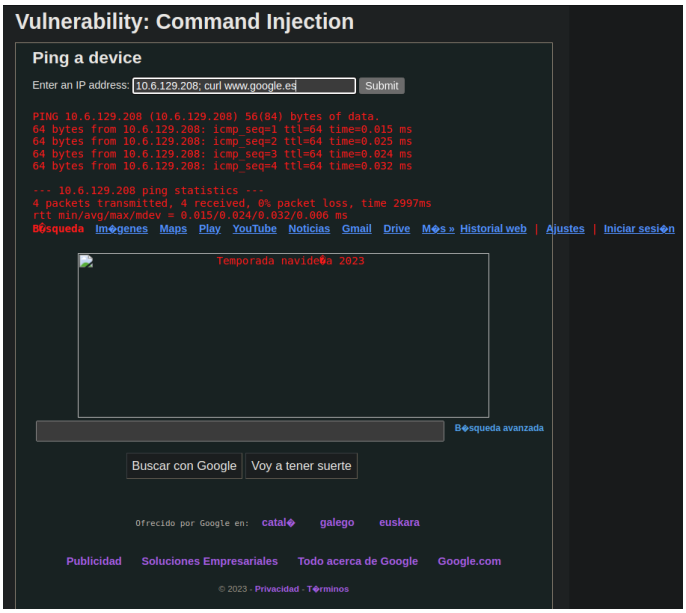

Figura 3.5: Obtención de un fichero por *wget*

- **Reverse Shell**: Para realizar el reverse shell, se ha descargado desde `https://github.com/php webshell/c99shell` el archivo *c99.php* y se ha subido al servidor web. Al inyectar c99shell en el servidor web, al intentar acceder a este lo que hará Apache es ejecutar el código PHP y devolverá el resultado de la ejecución del código PHP.

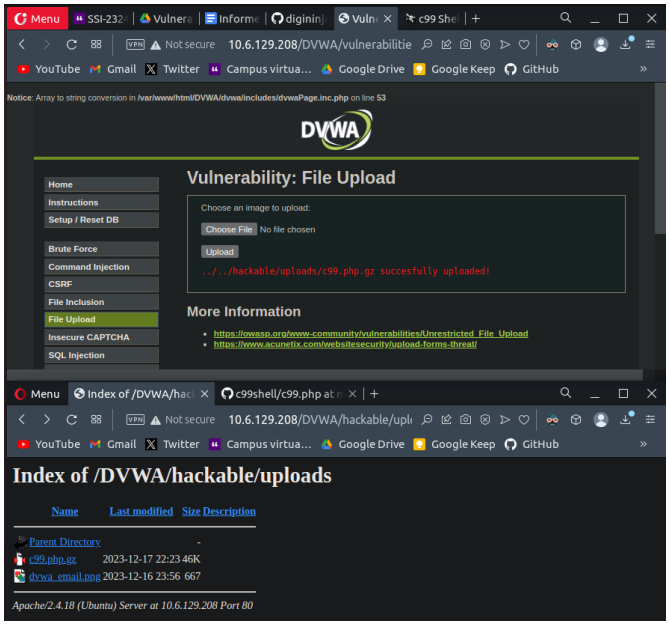

Figura 3.6: Inyección de c99shell en el servidor web

Como se ha subido descomprimido, se puede inyectar el código PHP descomprimiendo el archivo *c99shell.php.gz* por lo que en el "Command Injection" vamos a ejecutar el siguiente comando:

```
10.6.129.208; /bin/gunzip -v ../../hackable/uploads/c99.php
```

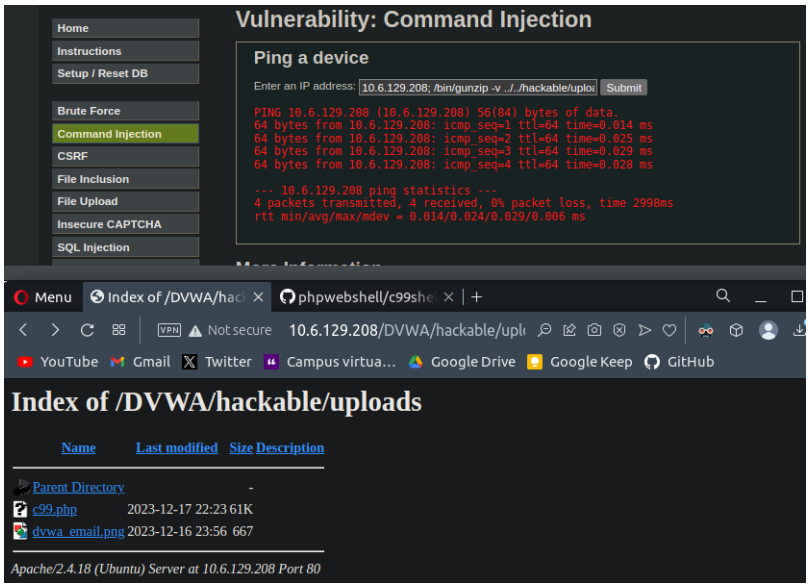

Figura 3.7: Descompresión de c99shell en el servidor web

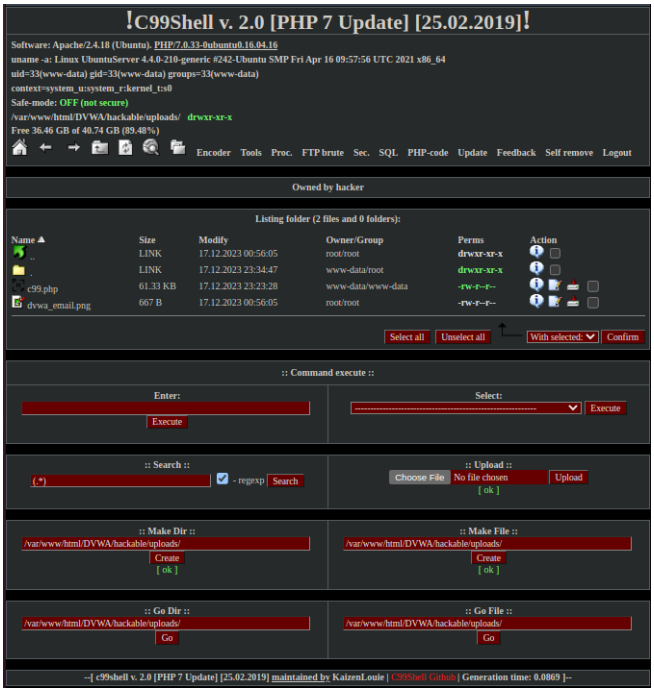

Figura 3.8: Ejecución de c99shell en el servidor web

4. Resultados de los análisis de vulnerabilidades

Vulnerabilidad	Ubuntu Server 16.04	CentOS 7
Creación carpeta temporal en */var/tmp*	✓	✓
Creación de un archivo en la raíz web	×	×
Inyección de comandos (con éxito)	✓	✓
cat */etc/passwd*	✓	✓
Obtener un fichero por *wget*	✓	✓
Reverse Shell	✓	✓

Cuadro 4.1: Tabla de vulnerabilidades con la configuración por defecto

Vulnerabilidad	Ubuntu App Armor	Ubuntu SELinux	CentOS 7 SELinux
Creación carpeta en */var/tmp*	✓	✓	×
Creación archivo en la raíz web	×	×	×
Inyección de comandos (con éxito)	✓	✓	✓
cat */etc/passwd*	✓	✓	✓
Obtener un fichero por *wget*	✓	✓	✓
Reverse Shell	✓	✓	✓

Cuadro 4.2: Tabla de vulnerabilidades con la configuración de AppArmor y SELinux

5. Bibliografía

1. digininja. (2023). DVWA. GitHub. Recuperado de `https://github.com/digininja/DVWA`
2. ComputerSecurityStudent. (2014). DVWA v1.0.7. Recuperado de `https://www.computersecuritystudent.com/cgi-bin/CSS/process_request_v3.pl?HID=688b0913be93a4d95daed400990c4745&TYPE=SUB`
3. phpwebshell. (2021). c99shell. GitHub. Recuperado de `https://github.com/phpwebshell/c99shell`