SAD - U3 Seguridad Lógica

Descargar estas actividades

Índice

▼ Actividades Seguridad Lógica
- Actividad 1. Ataques a contraseñas
- Actividad 2. Contraseñas de Sistemas Operativos
- Actividad 3. Defensa ibérica
- Actividad 4. Contraseñas de Windows
- Actividad 5. Contraseñas de Linux
- Actividad 6. ACLs en Linux

Actividades Seguridad Lógica

Actividad 1. Ataques a contraseñas

Una buena opción para acortar los tiempos de los ataques a las contraseñas es tener algún indicio del formato de las mismas.

Partiendo del archivo **hashes_practicas.txt** que contiene los hashes de las contraseñas de los usuarios `user1.x`, `user2.x` y `user3.x`, intenta obtener sus contraseñas.

Realiza un ataque a las contraseñas, teniendo en cuenta que las contraseñas tienes las siguientes propiedades:

- **user1.x** se obtiene con un ataque de diccionario
- **user2.x** se obtiene con un ataque por fuerza bruta con máscaras, sabiendo que están formadas por 3 minúsculas seguidas de 3 dígitos, por ejemplo "abc123"
- **user3.x** se obtiene con un ataque por diccionario con máscaras, sabiendo que las contraseñas están formadas por una palabra del diccionario (500_passwords.txt) seguidas de un año comprendido entre 2000 y 2019.

Documéntalo todo con capturas de pantalla en las que se vea el comando y los resultados producidos.

Actividad 2. Contraseñas de Sistemas Operativos

Hemos conseguido sacar esta información del archivo SAM de Windows

```
100:AE4D4025B89026B533A46849C79CEE3D:7FFB9A84B18B17F66DA382F2C2FEC342:::
```

y esta contraseña de diccionario sacada de un sistema Linux

```
$6$00.zqUv8$nAOCHqjtXJ8QjPIFlXdZes604kCXGlPqypNh5ON/McDRxHn7Mip3dx3gaSLIaE9ieRJaPvjUpq9KD5bmUkRue/
```

¿De qué formas podemos romper estas contraseñas?

Para obtener las contraseñas de Windows puedes usar Ophcrack o John the Ripper. Para las contraseñas de Linux, puedes usar John the Ripper o Hashcat. **Es importante que te fijes en el tipo de hash que se está utilizando para cada contraseña**.

Con Ophcrack, puedes usar las tablas de Rainbow para obtener la contraseña de Windows.
Tablas rainbow

Actividad 3. Defensa ibérica

Tomando el hash que se encuentra en el archivo defensa_iberica.txt, intenta obtener la contraseña del usuario con alguna de las herramientas vistas en clase.

Actividad 4. Contraseñas de Windows

Configura la política de contraseñas de Windows para que cumpla con los siguientes requisitos:

- Longitud mínima de 8 caracteres
- Contraseñas complejas
- Historial de contraseñas para que no se puedan repetir las últimas 5 contraseñas
- Caducidad de la contraseña cada 90 días
- Bloqueo de la cuenta tras 3 intentos fallidos
- Duración de bloqueo de 10 minutos
- Restablecimiento el contador de intentos tras 1 minuto

Realiza varias pruebas y capturas de pantalla para comprobar que se cumplen las políticas establecidas.

a) Prueba a crear un usuario y a cambiar la contraseña para comprobar que se cumplen las políticas establecidas.

> Que se vea como se cumplen tanto las políticas de complejidad, como la caducidad de la contraseña y el historial de contraseñas.

b) Prueba a bloquear la cuenta con 3 intentos fallidos, con una diferencia de más 1 minuto entre intentos.
c) Prueba a bloquear la cuenta con 3 intentos fallidos, intentando repetidas veces en menos de 1 minuto.

Actividad 5. Contraseñas de Linux

Configura la política de contraseñas de Windows para que cumpla con los siguientes requisitos:

- Longitud mínima de 8 caracteres
- Contraseñas complejas con al menos un caracter de cada de los siguientes elementos: mayúsculas, minúsculas, números y caracteres especiales
- Historial de contraseñas para que no se puedan repetir las últimas 5 contraseñas
- Caducidad de la contraseña cada 180 días con aviso previo de 14 días.

Realiza varias pruebas y capturas de pantalla para comprobar que se cumplen las políticas establecidas.

a) Prueba a crear un usuario y a cambiar la contraseña para comprobar que se cumplen las políticas establecidas.

> Que se vea como se cumplen tanto las políticas de complejidad, como la caducidad de la contraseña y el historial de contraseñas.

b) Para conseguir que también se cumplan las siguientes políticas:

- Bloqueo de la cuenta tras 3 intentos fallidos
- Duración de bloqueo de 10 minutos
- Restablecimiento el contador de intentos tras 1 minuto

Investiga como conseguir bloquear la cuenta con 3 intentos fallidos en menos de un minuto.

c) Para la configuración anterior, investiga como conseguir que la cuenta no se bloquee con 3 intentos fallidos, con una diferencia de más 1 minuto entre intentos.

Actividad 6. ACLs en Linux

Crea un pequeño script `/tmp/script1.sh` que muestre la fecha y hora actual.

```
#!/bin/bash
echo "La fecha y hora actual es: $(date)"
```

> ⚠ **Permisos especiales**
>
> Si no hacemos nada más, el script y los permisos que configuremos, sobre todo aquellos relacionados con la ejecución, no funcionaran como esperamos debido a los permisos especiales de Linux (setuid, setgid y sticky bit).
> En este caso, si miramos los permisos del directorio `/tmp`, veremos que tiene el sticky bit activado (tiene una `t` al final de los permisos), lo que impide que los usuarios que no son propietarios del archivo puedan ejecutarlo, aunque tengan permisos de ejecución.
> Por lo tanto, para que los permisos funcionen como esperamos, debemos crear el script en un directorio que no tenga el sticky bit activado, como por ejemplo `/home/usuario` o `/opt`.
>
> También podemos cambiar los permisos del directorio `/tmp`, pero no es recomendable por razones de seguridad. Si queres hacerlo, puedes usar el siguiente comando:
>
> ```
> chmod 1755 /tmp
> ```

Crea un usuario llamado `user1` y un grupo llamado `grupo1`. Haz al usuario y al grupo propietarios del archivo `/tmp/script1.sh` y asigna permisos de **lectura, escritura y ejecución al propietario, de lectura y escritura al grupo y de lectura al resto de usuarios**.

Crea un usuario llamado `user2` y añádelo al grupo `grupo1`. Comprueba que `user2` puede leer y escribir en el fichero `/tmp/script1.sh`, pero **no puede ejecutarlo**.

Crea un usuario llamado `user3` y añádelo al grupo `grupo3`. Comprueba que `user3` no puede modificar ni ejecutar el script `/tmp/script1.sh`, pero sí puede ver su contenido.

Ahora, usando acls extendidas haz las siguientes configuraciones:

- `user2`: asígnale permisos de lectura y ejecución sobre el sript `/tmp/script1.sh`
  ¿Qué permisos tiene ahora `user2` sobre el script?
  ¿Puede modificarlo?
  ¿Qué ha pasado con sus permisos por pertenecer a group1?
- `user3` añádele, a lo que ya tiene, permiso para modificar el script `/tmp/script1.sh` pero no ejecutarlo.
  ¿Qué permisos tiene ahora `user3` sobre el script?

- `group3` podrá leer y ejecutar el script `/tmp/script1.sh`
  ¿Qué permisos tiene ahora `user3` sobre el script?
  ¿Puede ejecutarlo? ¿Por qué?
- Cambia a `user2` de grupo (a `group2`) y comprueba si sigue pudiendo ejecutar el script.
  ¿Ha cambiado algo en sus permisos?

Tras cada paso, muestra capturas de pantalla que muestren el comando ejecutado y el resultado obtenido, así como la configuración de ACLs.