

ARQUITECTURA Y SISTEMAS OPERATIVOS

SEGURIDAD EN SISTEMAS OPERATIVOS:

LA SEGURIDAD EN LOS SISTEMAS OPERATIVOS ES CRUCIAL PARA PROTEGER LA INFORMACIÓN Y GARANTIZAR LA ESTABILIDAD DE LOS SISTEMAS.

CADA DISPOSITIVO EN RED REPRESENTA UN PUNTO DE ENTRADA QUE DEBE PROTEGERSE.

POR ELLO, ES FUNDAMENTAL IDENTIFICAR RIESGOS CLAVE, ANALIZAR LAS PRINCIPALES VULNERABILIDADES E IMPLEMENTAR ESTRATEGIAS EFECTIVAS DE MITIGACIÓN.

PRINCIPALES VULNERABILIDADES EN SISTEMAS OPERATIVOS

MALWARE Y VIRUS

PROGRAMAS MALICIOSOS DISEÑADOS PARA DAÑAR O ACCEDER A UN SISTEMA SIN PERMISO.

EXPLOTACIÓN DE VULNERABILIDADES

USO DE FALLOS EN EL SOFTWARE PARA OBTENER ACCESO NO AUTORIZADO.

ATAQUES DE FUERZA BRUTA

INTENTOS REPETIDOS DE ADIVINACIÓN DE CONTRASEÑAS.

INGENIERÍA SOCIAL

MANIPULACIÓN PSICOLÓGICA PARA OBTENER INFORMACIÓN CONFIDENCIAL.

ESTRATEGIAS DE MITIGACIÓN

ACTUALIZACIONES Y
PARCHES DE SEGURIDAD

MANTENER EL SISTEMA ACTUALIZADO
REDUCE LAS VULNERABILIDADES EXPLOTABLES.

ANTIVIRUS Y SOFTWARE DE
SEGURIDAD

HERRAMIENTAS ESENCIALES PARA LA
DETECCIÓN Y ELIMINACIÓN DE AMENAZAS.

CONFIGURACIÓN SEGURA

DESHABILITAR SERVICIOS INNECESARIOS
Y RESTRINGIR ACCESOS.

AUTENTICACIÓN Y
CONTROL DE ACCESO

IMPLEMENTACIÓN DE MÚLTIPLES FACTORES
DE AUTENTICACIÓN Y POLÍTICAS DE ACCESO
RESTRICTIVAS.

RBAC

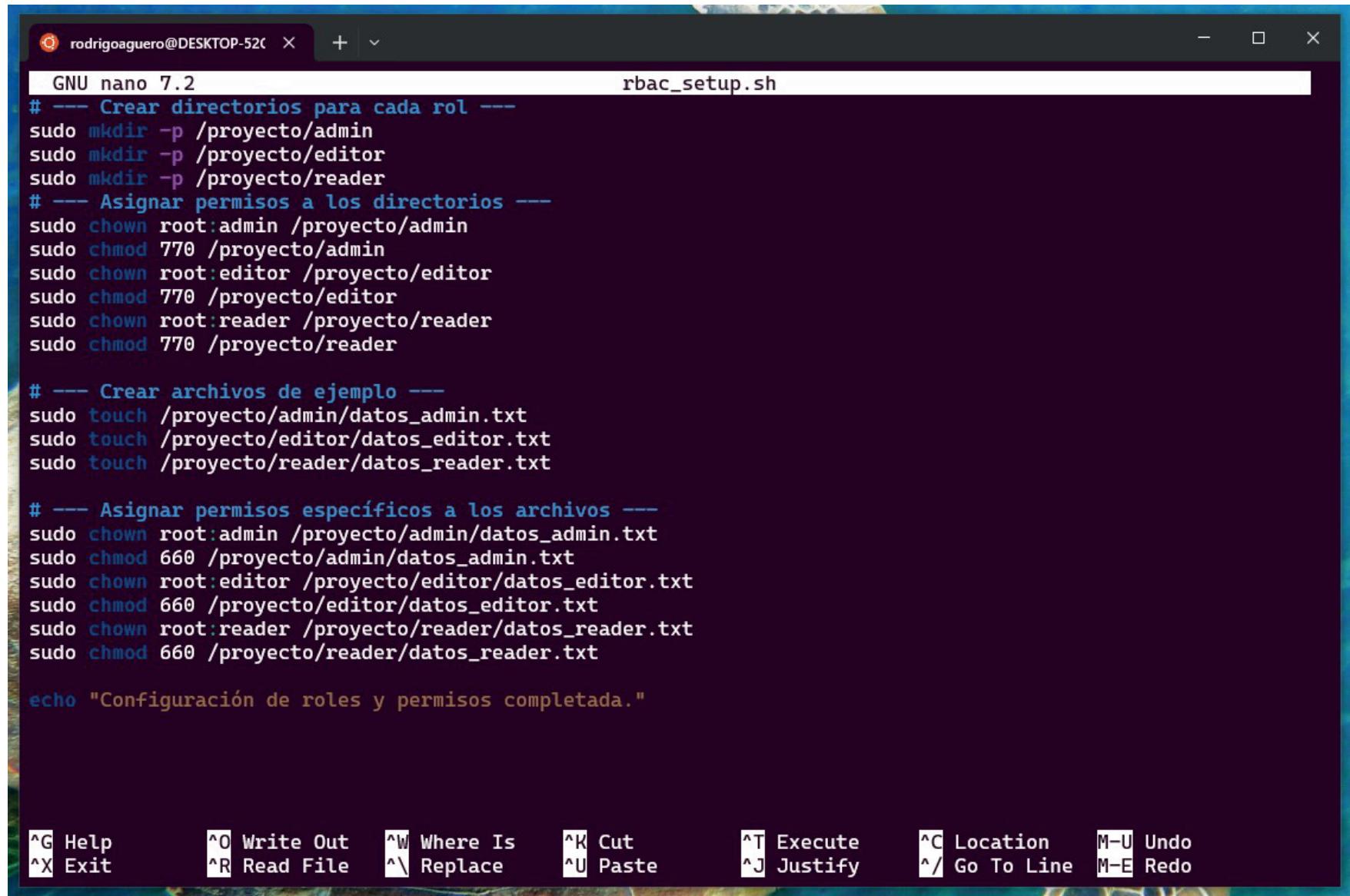
EL RBAC ES UNA TÉCNICA PARA GESTIONAR PERMISOS DE MANERA EFICIENTE, ASIGNANDO ROLES A LOS USUARIOS EN LUGAR DE PERMISOS INDIVIDUALES. ESTO SIMPLIFICA LA ADMINISTRACIÓN Y MEJORA LA SEGURIDAD DEL SISTEMA AL AGRUPAR PERMISOS EN ROLES ESPECÍFICOS SEGÚN LAS FUNCIONES DE LOS USUARIOS

CASO PRÁCTICO

LOS PASOS PARA LA IMPLEMENTACIÓN SON:

- CREAR GRUPOS QUE REPRESENTEN LOS ROLES.
- CREAR USUARIOS Y ASIGNARLOS A LOS GRUPOS.
- CONFIGURAR PERMISOS EN ARCHIVOS Y DIRECTORIOS.
- VERIFICAR EL CONTROL DE ACCESO.

ARCHIVO NANO PARA IMPLEMENTAR ROLES



The screenshot shows a terminal window titled "GNU nano 7.2" with the file name "rbac_setup.sh" in the title bar. The terminal is running on a Linux system, as indicated by the prompt "rodrigoaguero@DESKTOP-52C". The script content is as follows:

```
GNU nano 7.2                                     rbac_setup.sh
# --- Crear directorios para cada rol ---
sudo mkdir -p / proyecto/admin
sudo mkdir -p / proyecto/editor
sudo mkdir -p / proyecto/reader
# --- Asignar permisos a los directorios ---
sudo chown root:admin / proyecto/admin
sudo chmod 770 / proyecto/admin
sudo chown root:editor / proyecto/editor
sudo chmod 770 / proyecto/editor
sudo chown root:reader / proyecto/reader
sudo chmod 770 / proyecto/reader

# --- Crear archivos de ejemplo ---
sudo touch / proyecto/admin/datos_admin.txt
sudo touch / proyecto/editor/datos_editor.txt
sudo touch / proyecto/reader/datos_reader.txt

# --- Asignar permisos específicos a los archivos ---
sudo chown root:admin / proyecto/admin/datos_admin.txt
sudo chmod 660 / proyecto/admin/datos_admin.txt
sudo chown root:editor / proyecto/editor/datos_editor.txt
sudo chmod 660 / proyecto/editor/datos_editor.txt
sudo chown root:reader / proyecto/reader/datos_reader.txt
sudo chmod 660 / proyecto/reader/datos_reader.txt

echo "Configuración de roles y permisos completada."
```

The terminal window has a dark theme and includes a standard nano keybinding menu at the bottom.

ARCHIVO NANO PARA MONITOR_PROYECTOS.SH

The screenshot shows a terminal window titled "monitor_proyectos.sh" with the command "GNU nano 7.2". The file contains a bash script for monitoring project directories. The script sets up a log file at "/var/log/accesos_proyectos.log" and monitors three target directories: "/ proyecto/admin", "/ proyecto/editor", and "/ proyecto/reader". It logs the start of monitoring and then runs an inotifywait command to monitor access events in these directories. The script also provides instructions to stop the process by killing the inotifywait process.

```
GNU nano 7.2                               monitor_proyectos.sh

#!/bin/bash

LOG_FILE="/var/log/accesos_proyectos.log"
TARGET_DIRS="/ proyecto/admin / proyecto/editor / proyecto/reader"

echo "$(date) - Iniciando monitoreo de proyectos..." >> "$LOG_FILE"

# Monitorea eventos de 'acceso' (-e access) en los directorios, recursivamente (-r)
# Con --format puedes personalizar la salida
# --timefmt para el formato de la fecha
# --exclude para excluir archivos o directorios específicos si no quieres monitorear todo
inotifywait -m -r -e access --timefmt '%Y-%m-%d %H:%M:%S' --format '%T %w %f %e' $TARGET_DIRS >> "$LOG_FILE"

echo "Monitoreo iniciado en segundo plano. Los logs se guardan en $LOG_FILE"
echo "Para detenerlo, busca el proceso 'inotifywait' y mátalo."
echo "ps aux | grep inotifywait"

[ Read 16 lines ]
^G Help          ^O Write Out      ^W Where Is      ^K Cut           ^T Execute       ^C Location      M-U Undo
^X Exit          ^R Read File      ^\ Replace       ^U Paste         ^J Justify       ^/ Go To Line    M-E Redo
```

VERIFICAR EL CONTROL DE ACCESO.

```
rodrigoaguero@DESKTOP-52CLUP9:~$ cat /home/editor_user/intentos_fallidos.log
cat: /home/editor_user/intentos_fallidos.log: Permission denied
rodrigoaguero@DESKTOP-52CLUP9:~$ su - reader_user
Password:
$ cd / proyecto/admin
-sh: 1: cd: can't cd to / proyecto/admin
$ exit
rodrigoaguero@DESKTOP-52CLUP9:~$ cat /home/editor_user/intentos_fallidos.log
cat: /home/editor_user/intentos_fallidos.log: Permission denied
rodrigoaguero@DESKTOP-52CLUP9:~$ chmod +x monitor_proyectos.sh
rodrigoaguero@DESKTOP-52CLUP9:~$ sudo ./monitor_proyectos.sh
Monitoreo iniciado en segundo plano. Los logs se guardan en /var/log/accesos_proyectos.log
Para detenerlo, busca el proceso 'inotifywait' y mátalo.
ps aux | grep inotifywait
rodrigoaguero@DESKTOP-52CLUP9:~$ sudo cat /var/log/accesos_proyectos.log
Mon Jun 16 19:05:21 UTC 2025 - Iniciando monitoreo de proyectos...
Mon Jun 16 19:21:46 UTC 2025 - Iniciando monitoreo de proyectos...
2025-06-16 19:21:46 / proyecto/admin/ ACCESS,ISDIR
2025-06-16 19:21:46 / proyecto/editor/ ACCESS,ISDIR
2025-06-16 19:21:46 / proyecto/editor/ ACCESS,ISDIR
2025-06-16 19:21:46 / proyecto/reader/ ACCESS,ISDIR
rodrigoaguero@DESKTOP-52CLUP9:~$ nano rbac_setup.sh
rodrigoaguero@DESKTOP-52CLUP9:~$
```

CONCLUSIÓN:

**LA SEGURIDAD EN LOS SISTEMAS OPERATIVOS
ES UNA RESPONSABILIDAD FUNDAMENTAL.
AL ENTENDER LOS RIESGOS E IMPLEMENTAR
ESTRATEGIAS EFECTIVAS, SE LOGRA UN ENTORNO
OPERATIVO SEGURO Y CONFIABLE.**