PR_01.3

Bloque 1: Comandos de Información y Preparación

1. **Identidad del Usuario:** Abre una terminal y ejecuta un comando para saber qué usuario eres y a qué grupos perteneces.
2. **Usuarios Conectados:** Muestra quién está conectado actualmente al sistema. Luego, ejecuta otro comando que te dé información más detallada, como el tiempo que llevan conectados y qué están ejecutando.
3. **Historial de Conexiones:** Lista los últimos inicios de sesión en el sistema.
4. **Crear Entorno de Trabajo:** En tu directorio personal (`/home/tu_usuario`), crea una carpeta principal para todos los ejercicios llamada `practicas_linux`.
5. **Estructura de Directorios:** Dentro de `practicas_linux`, crea la siguiente estructura de directorios: `proyectos`, `documentos` y `scripts`.

Bloque 2: Gestión de Usuarios y Grupos

1. **Crear Grupos:** Crea tres nuevos grupos en el sistema: `desarrolladores`, `analistas` y `becarios`.
2. **Verificar Grupos:** Confirma que los grupos se han creado correctamente buscando sus nombres en el archivo `/etc/group`.
3. **Crear un Usuario Básico:** Crea un nuevo usuario llamado `juan`.
4. **Crear Usuario con Grupo Primario:** Crea una usuaria llamada `ana` y asígnala directamente al grupo primario `desarrolladores`.
5. **Crear Usuario Completo:** Crea un usuario `david` asignándolo al grupo primario `analistas` y, a la vez, como miembro de los grupos secundarios `desarrolladores` y `becarios`.
6. **Establecer Contraseñas:** Asigna una contraseña a los usuarios `juan`, `ana` y `david`.
7. **Verificar Usuarios:** Comprueba que los tres nuevos usuarios existen en el sistema, inspeccionando el final del archivo `/etc/passwd`.
8. **Cambiar de Usuario:** Conviértete en el usuario `juan` usando el comando `su`. Una vez dentro de su sesión, comprueba quién eres y en qué directorio te

encuentras. Vuelve a tu sesión de usuario original.

9. **Modificar Grupos de un Usuario:** Modifica al usuario `juan` para que su grupo primario sea `becarios` y añádelo también al grupo secundario `analistas`.
10. **Verificar Modificación:** Comprueba que los cambios del usuario `juan` se han aplicado correctamente.
11. **Bloquear una Cuenta:** Bloquea la cuenta del usuario `juan` para que no pueda iniciar sesión.
12. **Intentar Cambiar a Usuario Bloqueado:** Intenta convertirte en el usuario `juan` de nuevo. Debería fallar.
13. **Desbloquear una Cuenta:** Desbloquea la cuenta del usuario `juan`.
14. **Eliminar un Grupo:** Elimina el grupo `becarios`. ¿Qué ocurre? (Nota: Fallará si algún usuario lo tiene como grupo primario).
15. **Eliminar Usuario y su Directorio:** Elimina al usuario `juan` y asegúrate de que su directorio personal (`/home/juan`) también se borre.

Bloque 3: Permisos y Propiedad de Archivos

Realiza los siguientes ejercicios dentro de la carpeta `practicas_linux`. de tu directorio `home`

1. **Crear Archivos de Prueba:** Dentro de la carpeta `proyectos`, crea un archivo vacío llamado `informe.txt`. Dentro de `scripts`, crea otro archivo vacío llamado `lanzar_app.sh`.
2. **Ver Permisos:** Muestra los permisos por defecto de los archivos y directorios que has creado. Anota quién es el propietario y el grupo.
3. **Cambiar Propietario:** Cambia el propietario del archivo `informe.txt` para que pertenezca a la usuaria `ana`.
4. **Cambiar Grupo:** Cambia el grupo del directorio `proyectos` para que pertenezca al grupo `desarrolladores`.
5. **Cambiar Propietario y Grupo:** Cambia el propietario y el grupo del archivo `lanzar_app.sh` para que pertenezcan al usuario `david` y al grupo `analistas`, respectivamente, con un solo comando.
6. **Permisos con Notación Octal (Archivo):** Usa la notación numérica (octal) para asignar los siguientes permisos a `informe.txt`: el propietario (`ana`) puede

leer y escribir; el grupo (`desarrolladores`) solo puede leer; y los otros no tienen ningún permiso.

7. **Permisos con Notación Octal (Directorio):** Asigna permisos de lectura, escritura y ejecución para el propietario y solo de lectura y ejecución para los miembros del grupo al directorio `documentos`.
8. **Verificar Permisos:** Lista el contenido de `practicas_linux` para verificar que todos los cambios de propietario y permisos se han aplicado correctamente.
9. **Permisos con Notación Simbólica (Añadir):** Usa la notación simbólica para añadir el permiso de ejecución al propietario del script `lanzar_app.sh`.
10. **Permisos con Notación Simbólica (Quitar):** Quita el permiso de lectura al "resto del mundo" (otros) en el directorio `proyectos`.
11. **Permisos Recursivos:** Dentro de `proyectos`, crea una nueva carpeta `version2` con un archivo `notas.txt` dentro. Luego, cambia el propietario de la carpeta `proyectos` y todo su contenido para que pertenezca a `david` con un solo comando recursivo.
12. **Permiso Especial SGID en Directorio:** Establece el permiso especial SGID en el directorio `documentos`. Después, cambia a ser el usuario `david` (`su david`) y crea un nuevo archivo dentro de `documentos`. Verifica a qué grupo pertenece el nuevo archivo (debería heredar el del directorio `documentos`). Vuelve a tu usuario.
13. **Permiso Especial SUID:** Establece el permiso SUID en el script `lanzar_app.sh`. (Nota: Explica a tus alumnos qué implicaría esto si fuera un programa compilado).
14. **Comprobar `umask`:** Muestra el valor `umask` actual de tu sesión.
15. **Efecto de `umask`:** Cambia temporalmente tu `umask` a `077`. Crea un nuevo archivo llamado `privado.txt`. Comprueba sus permisos por defecto. Luego, restaura el `umask` a su valor original.

Bloque 4: Gestión de Servicios con `systemctl`

Nota: Para estos ejercicios, es seguro usar un servicio como `cups` (impresión) o `cron`/`crond` (tareas programadas). Evita usar servicios críticos como `sshd` si no estás seguro.

1. **Estado Detallado de un Servicio:** Comprueba el estado completo del servicio `cups`. Analiza la salida: ¿está activo (`active`), cargado (`loaded`) y habilitado (`enabled`)? Anota las últimas líneas de su registro (log) que aparecen.
2. **Comprobación Rápida:** Utiliza un comando más directo para verificar si el servicio `cups` está actualmente en ejecución (activo). La salida de este comando debería ser simplemente `active` o `inactive`.
3. **Ver Archivo de Unidad:** Muestra el contenido del archivo de unidad del servicio `cups` (`cups.service`). Esto te permitirá ver cómo está definido el servicio.
4. **Detener un Servicio:** Detén la ejecución del servicio `cups`. Comprueba su estado de nuevo para confirmar que está `inactive (dead)`.
5. **Iniciar un Servicio:** Vuelve a iniciar el servicio `cups`. Verifica una vez más que ha vuelto al estado `active (running)`.
6. **Reiniciar un Servicio:** El comando `restart` es muy común tras un cambio de configuración. Ejecútalo para el servicio `cups`.
7. **Habilitar para el Arranque:** Asegúrate de que el servicio `cups` esté configurado para iniciarse automáticamente cada vez que el sistema arranque.
8. **Verificar si está Habilitado:** Usa un comando específico para preguntar si `cups` está habilitado. La salida debería ser `enabled` o `disabled`.
9. **Deshabilitar para el Arranque:** Ahora, desactiva el servicio `cups` para que no se inicie automáticamente.
10. **Enmascarar un Servicio:** El enmascaramiento es una forma más contundente de deshabilitar, ya que impide cualquier tipo de inicio (manual o automático). Enmascara el servicio `cups`. Intenta iniciarlo después. Debería fallar. No olvides desenmascararlo (`unmask`) al terminar el ejercicio.

Bloque 4: Gestión de `ufw`

1. **Comprobar Estado y Activar UFW:**
   * Primero, ejecuta un comando para verificar el estado actual del firewall. Probablemente estará inactivo.
   * A continuación, activa UFW. Presta atención al mensaje de advertencia, especialmente si estás conectado por SSH.

2. **Permitir un Servicio Web (HTTP):**
   * Imagina que tu servidor necesita alojar una página web. Añade una regla para permitir todas las conexiones entrantes para el servicio `http`.
   * Verifica el estado del firewall de nuevo para confirmar que la regla (y el puerto 80) se ha añadido correctamente.
3. **Abrir un Puerto Específico:**
   * Imagina que estás ejecutando un servidor de aplicaciones web en el puerto 8080. Añade una regla para permitir las conexiones entrantes `TCP` a ese puerto.
4. **Permitir un Rango de Puertos:**
   * Supón que una aplicación FTP necesita un rango de puertos pasivos. Añade una regla para permitir las conexiones `TCP` en el rango de puertos desde el 3000 al 3100.
5. **Bloquear una Dirección IP:**
   * Por seguridad, has detectado actividad sospechosa desde la IP `192.168.100.50`. Añade una regla para denegar todas las conexiones provenientes de esa dirección IP.
6. **Listar Reglas para Borrar:**
   * Muestra todas las reglas activas del firewall, pero esta vez de forma numerada, para prepararte para eliminar una de ellas.
7. **Eliminar una Regla:**
   * Basándote en la lista del ejercicio anterior, elimina la regla que creaste para el puerto `8080`.
   * Vuelve a listar las reglas (de forma normal o numerada) para confirmar que la regla ha sido eliminada correctamente.