### **Implementación del Ciclo PDCA en un Sistema Operativo Linux con Servidor Web y Base de Datos**

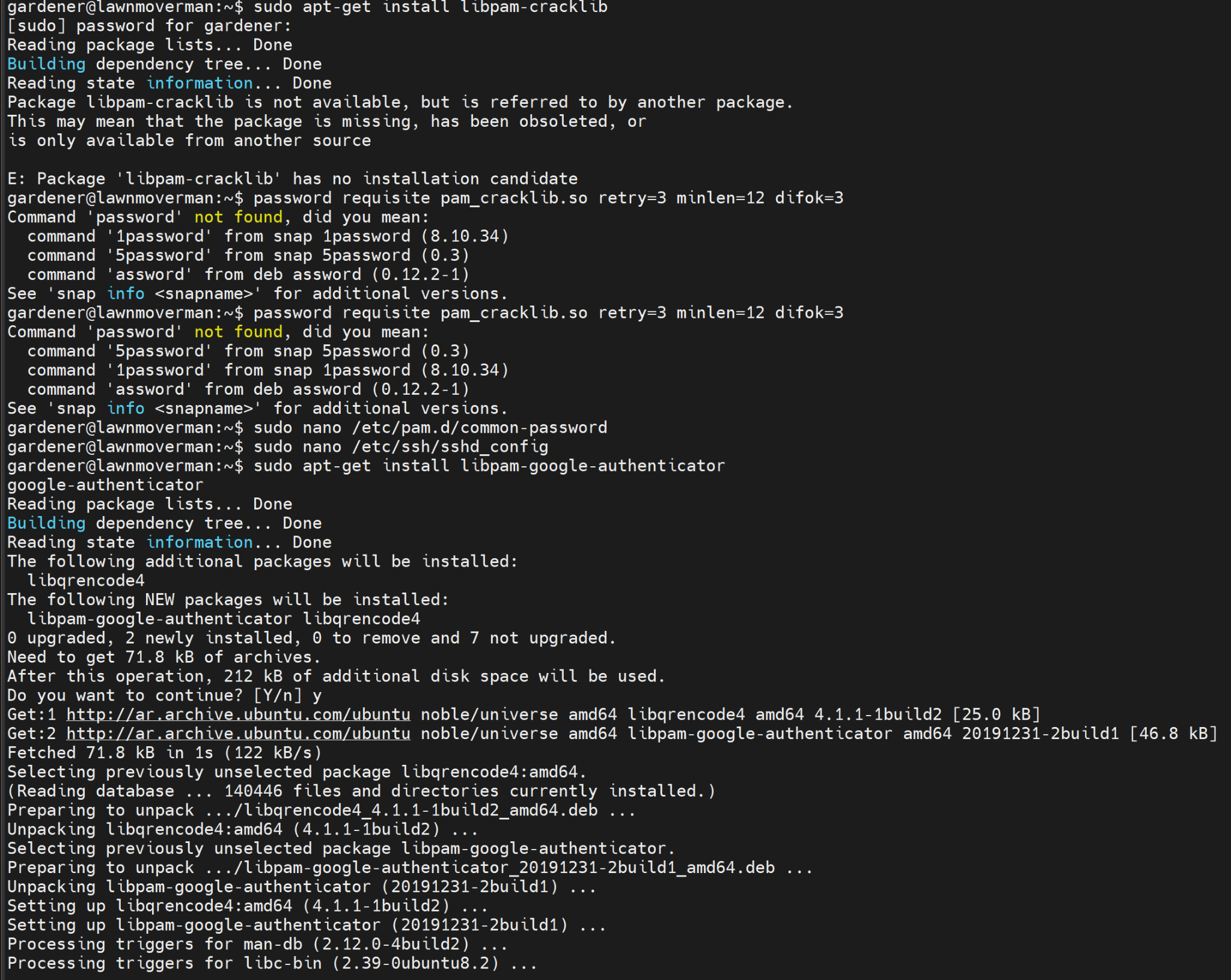
#### 

#### **Planificar**

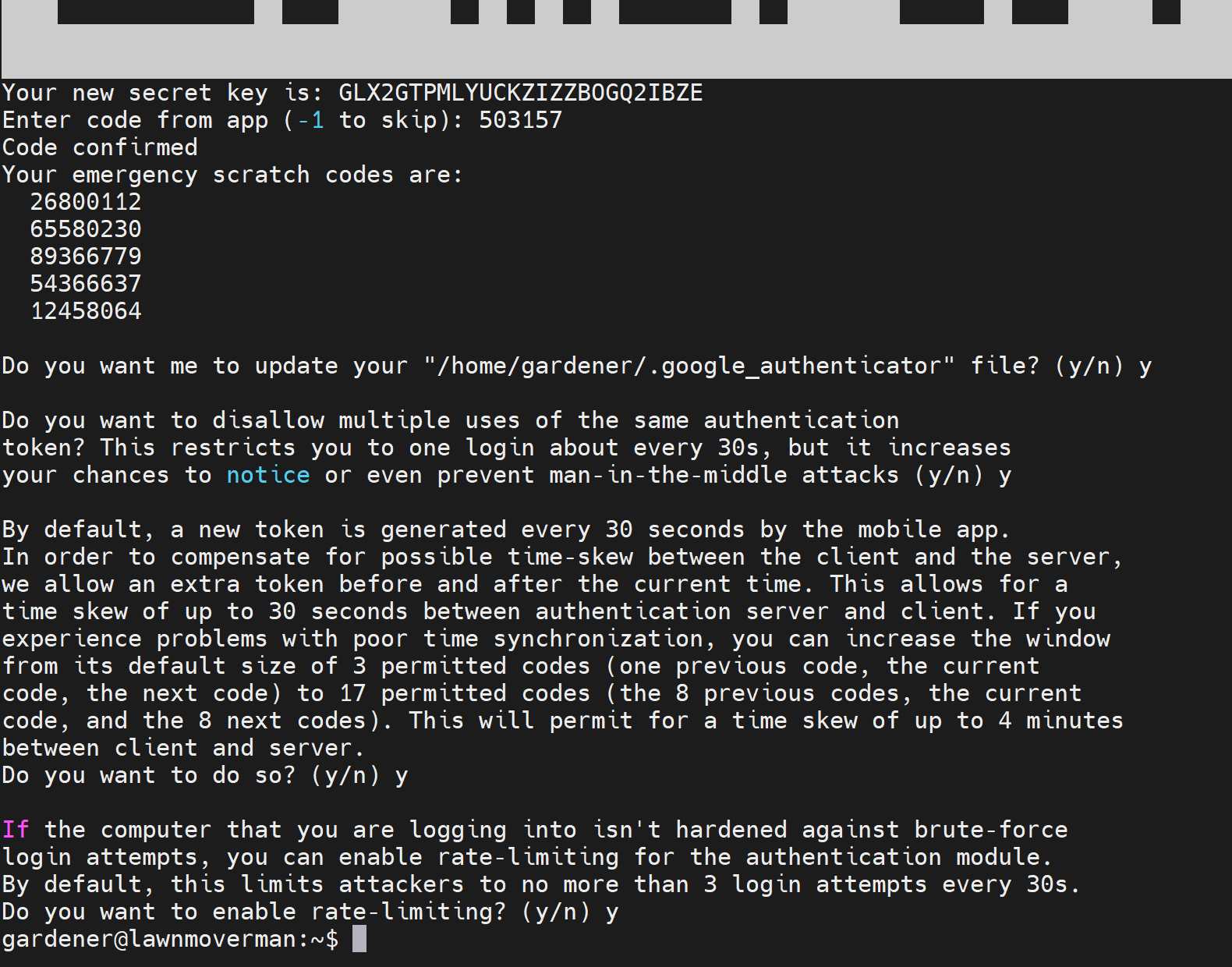
Identificación y Autentificación:

1. Política de Contraseña:
   * Problema: Contraseñas débiles que pueden ser fácilmente comprometidas.
   * Objetivo: Implementar una política de contraseñas robusta utilizando PAM (Pluggable Authentication Module) y el módulo pam\_cracklib para asegurar que las contraseñas cumplan con ciertos criterios de complejidad



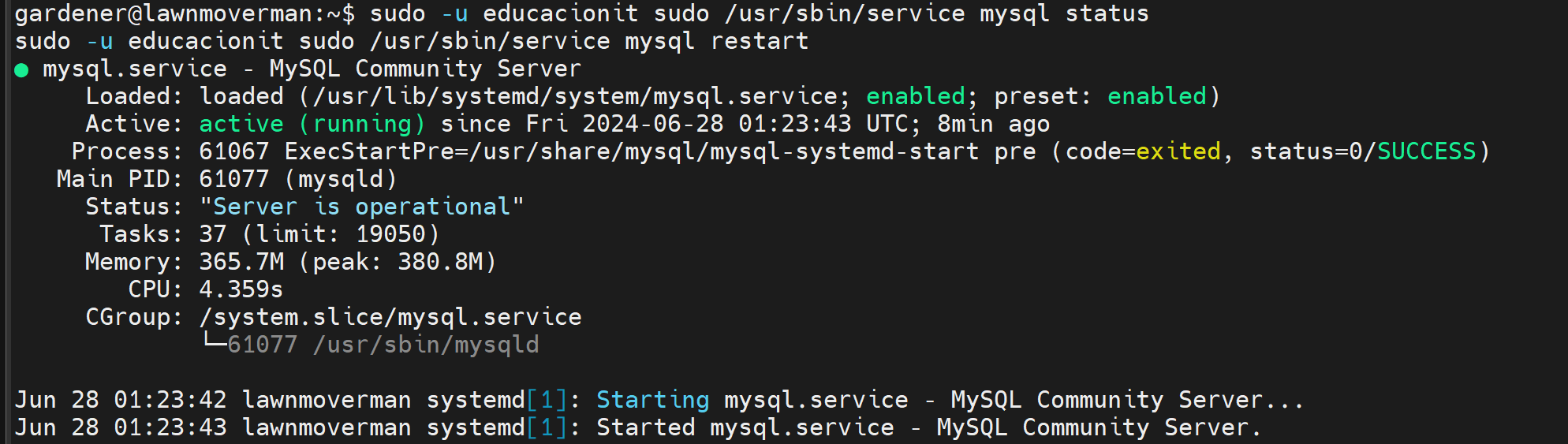


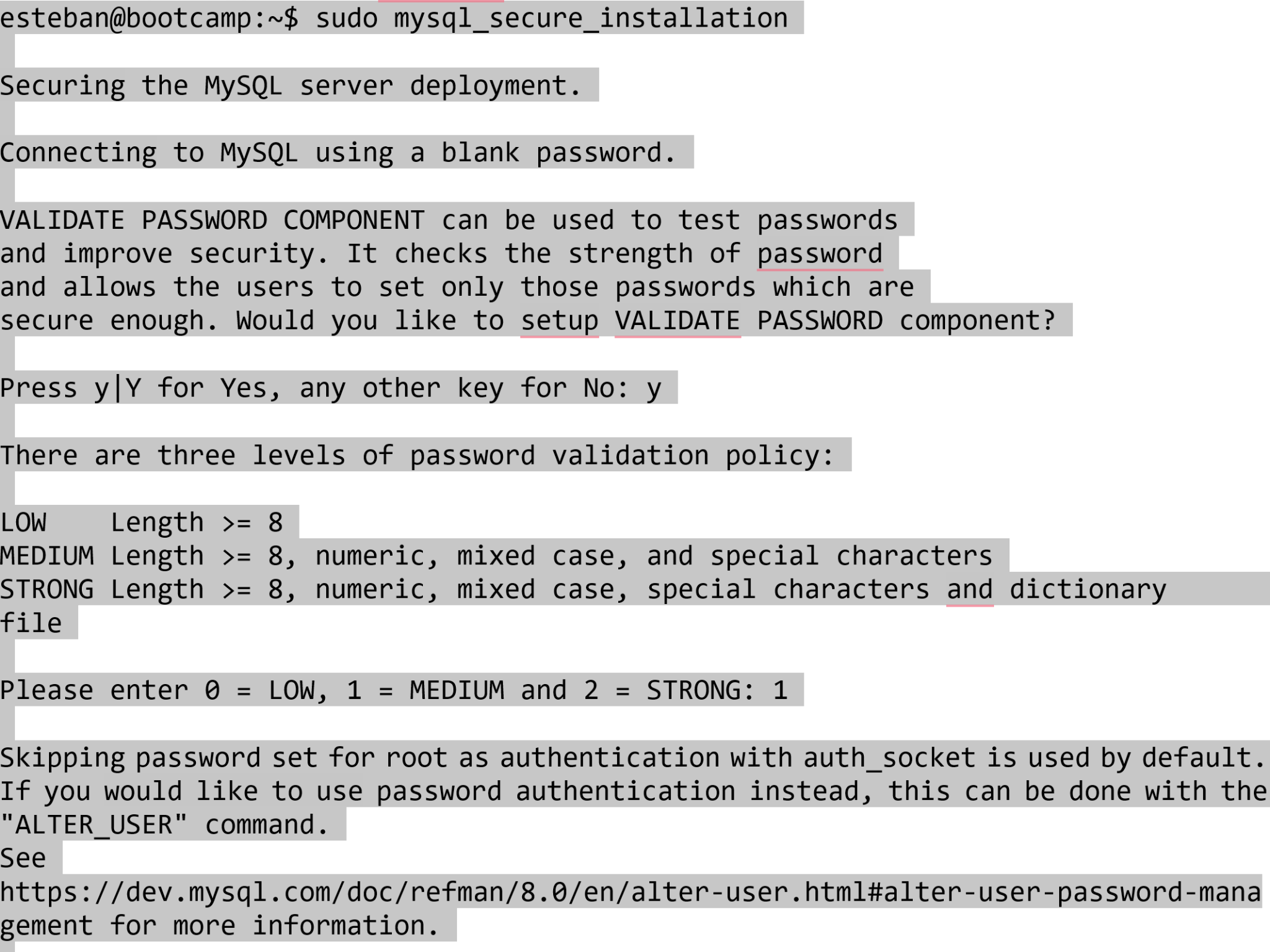
1. Servicios SSH y Autenticación Multifactor:
   * Problema: Acceso no autorizado a través de SSH.
   * Objetivo: Configurar SSH para usar autenticación con llaves y habilitar Google Authenticator para autenticación multifactor.

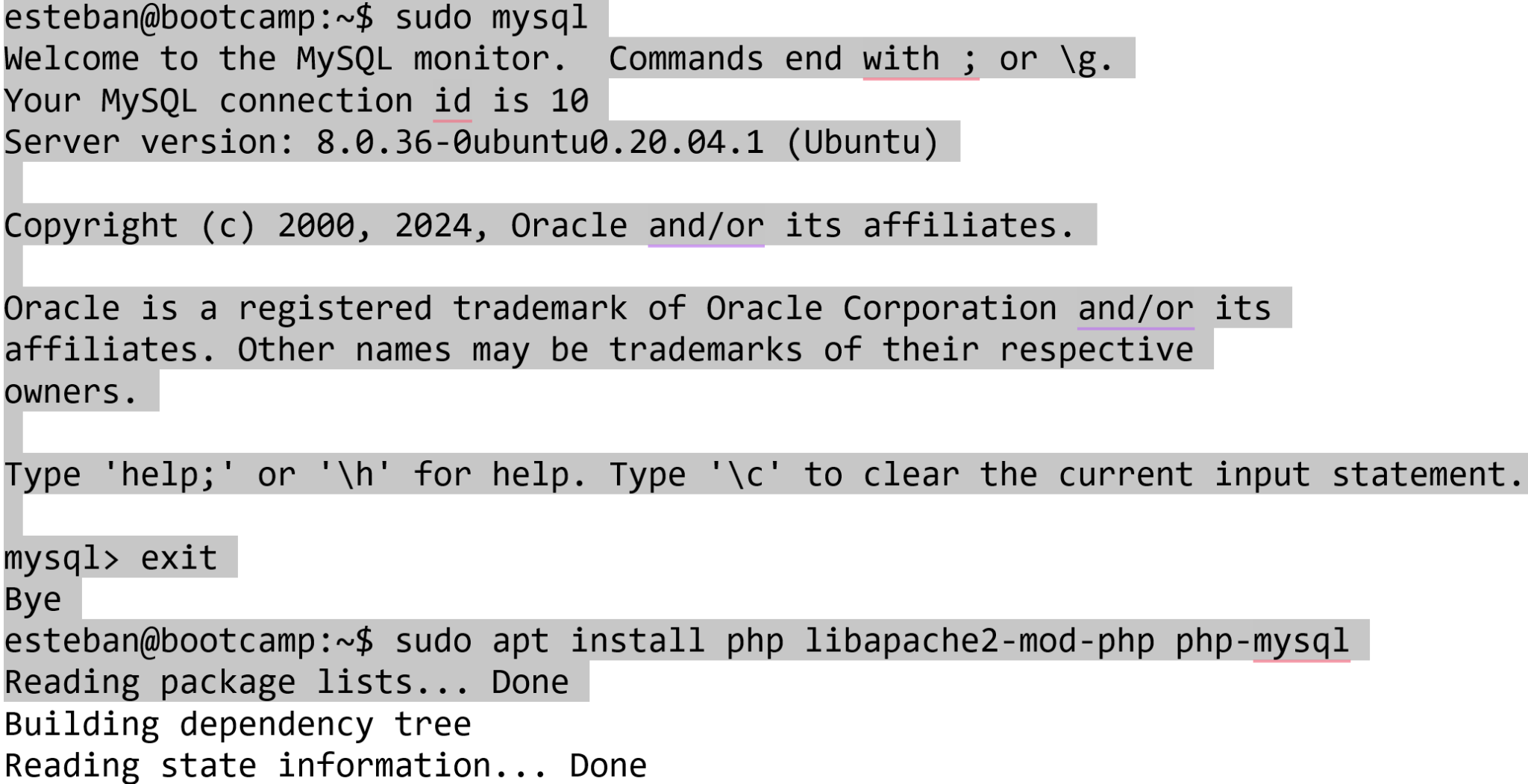


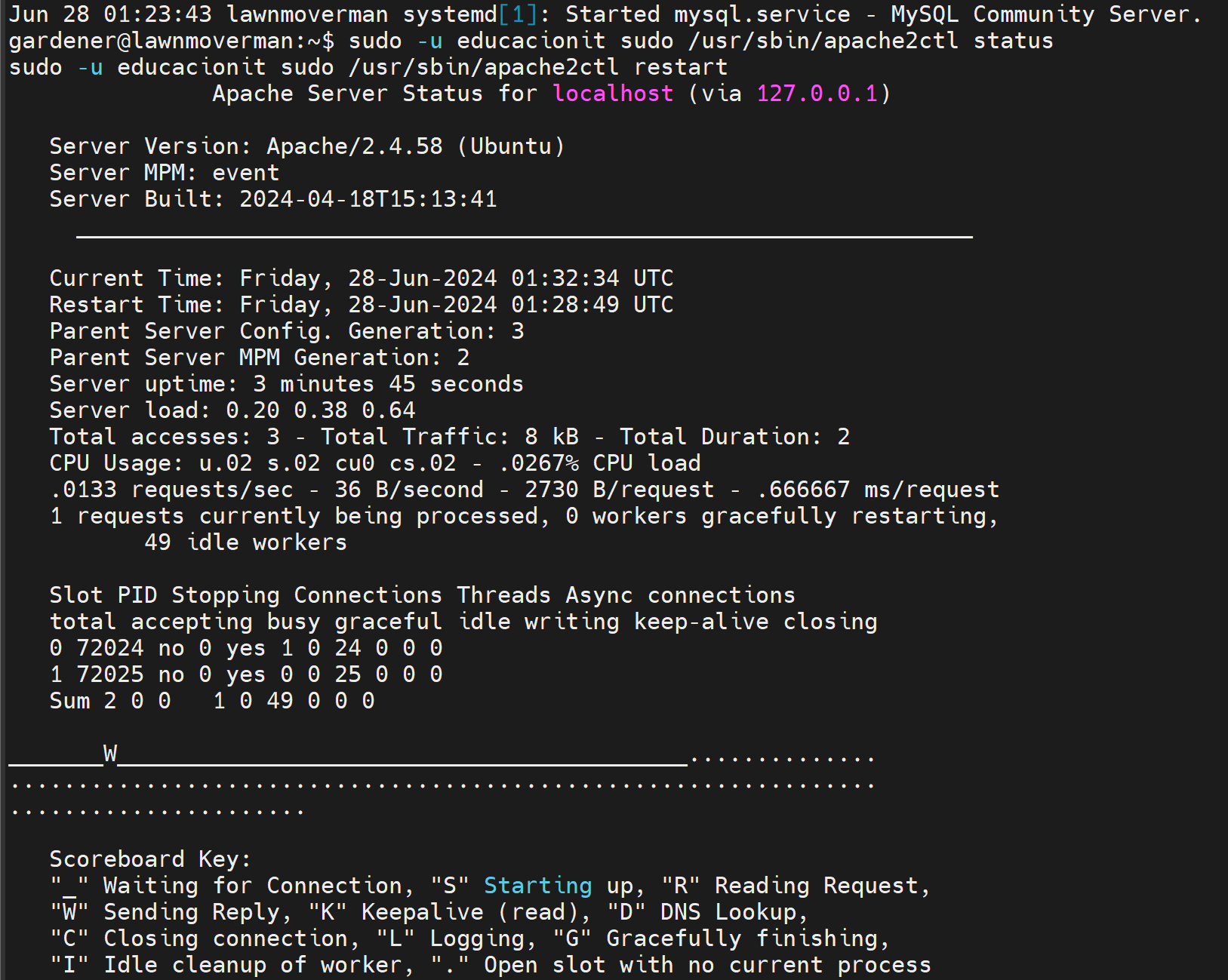
Autorización:

1. Uso de sudo y Permisos del Sistema de Archivos:
   * Problema: Usuarios con permisos excesivos.
   * Objetivo: Configurar sudo para que el usuario educacionit tenga permisos específicos para gestionar Apache2 y Mysql (restart, start, reload, status) y ajustar los permisos del sistema de archivos para limitar el acceso a archivos críticos.











1. Configuración de Apache2:
   * Problema: Acceso no autorizado a directorios web.
   * Objetivo: Configurar el archivo de índice de Apache2 y establecer permisos adecuados para los directorios web

#### **Do (Hacer)**

Implementación y Documentación:

1. Política de Contraseña:
   * Instalar PAM y configurar pam\_cracklib:
   * cpp

sudo apt-get install libpam-cracklib

sudo nano /etc/pam.d/common-password

* + Añadir la línea:
  + cpp

password requisite pam\_cracklib.so retry=3 minlen=12 difok=3

password [success=1 default=ignore] pam\_unix.so obscure use\_authtok try\_first\_pass sha512

password requisite pam\_deny.so

password required pam\_permit.so

1. Configuración de SSH y Google Authenticator:
   * Configurar SSH para usar llaves:
   * cpp

sudo nano /etc/ssh/sshd\_config

* + Asegurarse de que las siguientes líneas estén presentes:
  + cpp

PubkeyAuthentication yes

PasswordAuthentication no

* + Instalar y configurar Google Authenticator:
  + cpp

sudo apt-get install libpam-google-authenticator

google-authenticator

sudo nano /etc/pam.d/sshd

* + Añadir la línea:
  + cpp

auth required pam\_google\_authenticator.so

1. Configuración de sudo y Permisos del Sistema de Archivos:
   * Editar el archivo de configuración de sudo:
   * cpp

sudo visudo

* + Añadir:
  + cpp

educacionit ALL=(ALL) NOPASSWD: /usr/sbin/apache2ctl, /usr/bin/systemctl restart mariadb, /usr/bin/systemctl start mariadb, /usr/bin/systemctl reload mariadb, /usr/bin/systemctl status mariadb

* + Ajustar permisos del sistema de archivos:
  + cpp

sudo chown -R root:root /var/www

sudo chmod -R 755 /var/www

1. Configuración de Apache2:
   * Editar el archivo de configuración de Apache2:
   * cpp

sudo nano /etc/apache2/apache2.conf

* + Asegurarse de que los permisos sean adecuados:
  + cpp

<Directory /var/www/>

Options Indexes FollowSymLinks

AllowOverride None

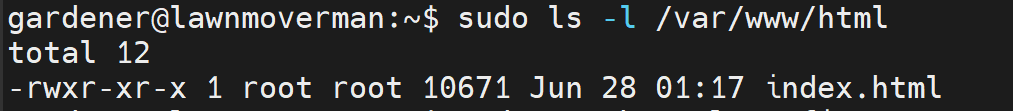
Require all granted

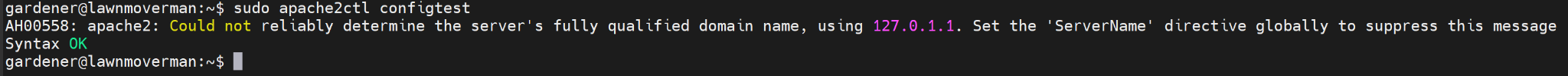
</Directory>

#### **Check (Verificar)**

Verificación de las Tareas Realizadas:

1. Política de Contraseña:
   * Verificar que las contraseñas cumplen con los requisitos establecidos.
   * Punto a mejorar: Implementar un sistema de notificación para contraseñas que están a punto de expirar.
2. SSH y Google Authenticator:
   * Verificar que el acceso SSH solo es posible con llaves y autenticación multifactor.
   * Punto a mejorar: Realizar pruebas de penetración para asegurar que no hay vulnerabilidades en la configuración de SSH.
3. Permisos y sudo:
   * Verificar que el usuario educacionit solo tiene los permisos especificados.
   * Punto a mejorar: Revisar periódicamente los permisos de sudo para asegurar que no se han realizado cambios no autorizados.
4. Apache2:
   * Verificar que los permisos de los directorios web son correctos y que no hay acceso no autorizado.
   * Punto a mejorar: Implementar un sistema de monitoreo para detectar accesos no autorizados.





#### **Actuar**

Checklist para Futuros Despliegues:

1. Política de Contraseña:
   * Configurar pam\_cracklib con los parámetros adecuados.
   * Verificar la complejidad de las contraseñas.
2. SSH y Google Authenticator:
   * Configurar SSH para usar llaves y deshabilitar la autenticación por contraseña.
   * Instalar y configurar Google Authenticator.
3. Permisos y sudo:
   * Configurar sudo para usuarios específicos.
   * Ajustar los permisos del sistema de archivos.
4. Apache2:
   * Configurar los permisos de los directorios web.
   * Implementar un sistema de monitoreo para detectar accesos no autorizados.

Este plan estandarizado ayudará a asegurar que futuros despliegues sean seguros y cumplan con los requisitos de seguridad establecidos.

<https://github.com/andrespedrospano/desafio17>