INSTITUTO NEBRIJA | Formación Profesional

TRABAJO FINAL MODULAR PARA SISTEMAS OPERATIVOS

PROYECTO IMPLANTACIÓN DE SISTEMAS OPERATIVOS

ENLACE REPOSITORIO GITHUB

https://github.com/VictoriaFedericaPatinPiti/iso_proyecto_final

Nombres alumnos:

Álvaro Sánchez Aida Torres Iván Ruiz

Scrum Master (S1) : Javier Muñoz

Curso académico: 1º ASIR

Tutor del proyecto: Carmelo

ÍNDICE PAGINADO

1. JUSTIFICACIÓN DEL PROYECTO

2. INTRODUCCIÓN Y CONTEXTO DEL ESCENARIO

3. ANÁLISIS COMPARATIVO ENTRE WINDOWS Y LINUX

    3.1 INSTALACIÓN

    3.2 LICENCIAS

    3.3 INTERFAZ GRÁFICA Y USABILIDAD

    3.4 HERRAMIENTAS ADMINISTRATIVAS

    3.5 SEGURIDAD

    3.6 MANTENIMIENTO

4. ESCENARIOS DE USO RECOMENDADOS

    4.1 USUARIO ESTÁNDAR

    4.2 USUARIO DESARROLLADOR

    4.3 ADMINISTRADOR DE SISTEMAS

    4.4 PERSONAL TÉCNICO

5. DISEÑO DE LA ARQUITECTURA PROPUESTA

6. INSTALACIÓN PASO A PASO

    6.1 WINDOWS

    6.2 LINUX (UBUNTU, DEBIAN O ROCKY LINUX)

7. CONFIGURACIÓN BÁSICA Y AVANZADA

    7.1 USUARIOS Y PERMISOS

    7.2 SERVICIOS Y REDES

    7.3 POLÍTICAS DE GRUPO Y SUDOERS

8. MEDIDAS DE SEGURIDAD APLICADAS

8.1 FIREWALL

8.2 ANTIVIRUS Y ACTUALIZACIONES

8.3 LOGS Y MONITOREO

8.4 CIFRADO BÁSICO

9. AUTOMATIZACIÓN Y SCRIPTS DE MANTENIMIENTO

9.1 CRON, TAREAS PROGRAMADAS

9.2 POLÍTICAS DE GRUPO

9.3 SCRIPTS DE LIMPIEZA Y BACKUP

10. METODOLOGÍA DE TRABAJO ÁGIL

10.1 ROLES DEL EQUIPO (SCRUM MASTER, DOCUMENTADOR ...)

10.2 HERRAMIENTAS DE GESTIÓN (KANBAN / TRELLO)

10.3 REUNIONES Y SEGUIMIENTO

11. USO DE REPOSITORIO GITHUB

11.1 ESTRUCTURA DEL PROYECTO

11.2 EVIDENCIAS DE TRABAJO EN EQUIPO (COMMITS, ISSUES)

12. DOCUMENTACIÓN TÉCNICA DEL PROYECTO

12.1 CONFIGURACIONES REALIZADAS

12.2 COMANDOS UTILIZADOS

12.3 LOGS Y EVIDENCIAS

13. CONCLUSIONES Y PROPUESTA FINAL

14. ANEXOS

14.1 CAPTURAS DE PANTALLA

14.2 SCRIPTS Y CONFIGURACIONES COMPLETAS

14.3 DATOS COMPLEMENTARIOS

1. JUSTIFICACIÓN DEL PROYECTO

UNA PEQUEÑA EMPRESA EN CRECIMIENTO NECESITA DESPLEGAR UNA INFRAESTRUCTURA DE SISTEMAS OPERATIVOS QUE SEA ESTABLE, SEGURA Y ADECUADA PARA DISTINTOS PERFILES DE USUARIO. EL EQUIPO TÉCNICO DEBE ESTUDIAR Y DOCUMENTAR UNA PROPUESTA DE IMPLANTACIÓN COMBINANDO SISTEMAS WINDOWS Y LINUX, JUSTIFICANDO SU ELECCIÓN EN FUNCIÓN DE LOS DISTINTOS PERFILES, TAREAS, LICENCIAS, SEGURIDAD Y COSTES.

EL PROYECTO SE DESARROLLARÁ EN UN ENTORNO VIRTUALIZADO, QUE SIMULARÁ LAS CONDICIONES REALES DE TRABAJO DENTRO DE UNA ORGANIZACIÓN. SE USARÁN HERRAMIENTAS DE GESTIÓN DE PROYECTOS ÁGILES PARA COORDINAR AL EQUIPO Y MANTENER UN FLUJO DE TRABAJO CONTINUO Y DOCUMENTADO.

2. INTRODUCCIÓN

SOBRE EL PROYECTO SE PLANTEA UNA SOLUCIÓN TECNOLÓGICA PARA UNA PEQUEÑA EMPRESA EN CRECIMIENTO QUE NECESITA UNA INFRAESTRUCTURA INFORMÁTICA FIABLE, SEGURA Y ADAPTADA A DIFERENTES PERFILES DE USUARIOS.

MUCHAS ORGANIZACIONES REQUIEREN ENTORNOS MIXTOS CON EL USO DE DIFERENTES SISTEMAS OPERATIVOS PROPIETARIOS. WINDOWS EN UN USO COMÚN O SISTEMAS DE CÓDIGO ABIERTO, COMO LINUX. ESTA COMBINACIÓN PERMITE APROVECHAR LO MEJOR DE AMBOS SISTEMAS: COMPATIBILIDAD Y FACILIDAD DE

USO DE WINDOWS JUNTO CON LA ESTABILIDAD, SEGURIDAD Y BAJO COSTE ENTRE LAS DIFERENTES OPCIONES DE DISTRIBUCIONES LINUX.

EL OBJETIVO PRINCIPAL DEL PROYECTO ES DISEÑAR, COMPARAR Y DESPLEGAR AMBOS ENTORNOS (WINDOWS Y LINUX) EN UN ENTORNO VIRTUALIZADO QUE UTILICE UNA EMPRESA REAL. A TRAVÉS DEL ANÁLISIS TÉCNICO, LA INSTALACIÓN, CONFIGURACIÓN, AUTOMATIZACIÓN Y DOCUMENTACIÓN DE TAREAS, SE BUSCA OFRECER UNA PROPUESTA DE INFRAESTRUCTURA EQUILIBRADA QUE RESPONDA A LAS NECESIDADES DE DISTINTOS PERFILES: USUARIOS ESTÁNDAR, DESARROLLADORES, Y PERSONAL ADMINISTRATIVO O TÉCNICO.

SE TRABAJARÁ UTILIZANDO METODOLOGÍAS ÁGILES COMO **SCRUM** Y HERRAMIENTAS DE GESTIÓN COMO **KANBAN**, FOMENTANDO LA ORGANIZACIÓN, LA COMUNICACIÓN Y LA COLABORACIÓN DENTRO DEL EQUIPO. TODOS LOS AVANCES SERÁN REGISTRADOS EN UN REPOSITORIO GITHUB, DONDE QUEDARÁN REFLEJADAS LAS

TAREAS, SCRIPTS, CONFIGURACIONES Y EVIDENCIAS DEL TRABAJO REALIZADO POR CADA UNO DE LOS MIEMBROS COMPONENTES DEL PROYECTO.

INTEGRAMOS DIFERENTES COMPETENCIAS DEL MÓDULO DE IMPLANTACIÓN DE SISTEMAS OPERATIVOS DEL CICLO FORMATIVO DE PRIMERO DE ASIR.

3. ANÁLISIS COMPARATIVO

3.1 INSTALACIÓN

WINDOWS	LINUX (UBUNTU, DEBIAN, ROCKY LINUX)
INSTALACIÓN GRÁFICA SENCILLA, REQUIERE LICENCIA. ES MUY AMIGABLE PARA USUARIOS CON POCA EXPERIENCIA	PUEDE SER GRÁFICA O POR TERMINAL, ALGUNAS DISTRIBUCIONES MÁS TÉCNICAS, NO REQUIERE LICENCIA Y ES GRATUITA
CONCLUSIÓN: LA INSTALACIÓN DE WINDOWS ES MÁS FÁCIL PARA PRINCIPIANTES, PERO IMPLICA COSTE POR LICENCIA. LINUX ES MÁS FLEXIBLE Y SIN COSTE, AUNQUE PUEDE SER MÁS COMPLEJA PARA USUARIOS NOVATOS	

3.2 LICENCIAS

WINDOWS	LINUX (UBUNTU, DEBIAN, ROCKY LINUX)
SISTEMA OPERATIVO PROPIETARIO, CON LICENCIA DE PAGO POR USUARIO O DISPOSITIVO	CÓDIGO ABIERTO, GRATUITO, SIN COSTES DE LICENCIA, AUNQUE EXISTEN VERSIONES EMPRESARIALES CON SOPORTE DE PAGO
CONCLUSIÓN: LINUX REDUCE COSTES POR NO NECESITAR LICENCIAS, IDEAL PARA EMPRESAS CON PRESUPUESTO LIMITADO. WINDOWS, OFRECE SOPORTE OFICIAL Y COMPATIBILIDAD GARANTIZADA CON MUCHAS APLICACIONES	

3.3 INTERFAZ GRÁFICA Y USABILIDAD

WINDOWS	LINUX (UBUNTU, DEBIAN, ROCKY LINUX)
INTERFAZ GRÁFICA CONOCIDA Y FAMILIAR PARA LA MAYORÍA DE USUARIOS, FÁCIL EN USO DIARIO	INTERFAZ VARIABLE SEGÚN LA DISTRIBUCIÓN Y ENTORNO DE ESCRITORIO (GNOME, KDE, XFCE, ETC.); PUEDE REQUERIR ADAPTACIÓN
CONCLUSIÓN: WINDOWS ES MÁS ACCESIBLE PARA USUARIOS ESTÁNDAR POR SU INTERFAZ FAMILIAR, MIENTRAS QUE LINUX PUEDE NECESITAR UNA CURVA DE APRENDIZAJE, AUNQUE OFRECE OPCIONES DE PERSONALIZACIÓN	

3.4 HERRAMIENTAS ADMINISTRATIVAS

WINDOWS	LINUX (UBUNTU, DEBIAN, ROCKY LINUX)
PANEL DE CONTROL, POWERSHELL, EDITOR DE POLÍTICAS DE GRUPO, GUI CON GESTIÓN	GESTIÓN PRINCIPALMENTE POR TERMINAL, COMANDOS BASH, SYSTEMD, SCRIPTS; GRAN POTENCIAL PARA AUTOMATIZACIÓN
CONCLUSIÓN: WINDOWS FACILITA LA ADMINISTRACIÓN PARA USUARIOS CON CONOCIMIENTOS BÁSICOS, LINUX OFRECE MAYOR CONTROL Y POSIBILIDADES DE AUTOMATIZACIÓN PARA ADMINISTRADORES AVANZADOS	

3.5 SEGURIDAD

WINDOWS	LINUX (UBUNTU, DEBIAN, ROCKY LINUX)
MÁS VULNERABLE POR SU POPULARIDAD; REQUIERE ANTIVIRUS Y POLÍTICAS DE SEGURIDAD	SISTEMA MÁS SEGURO POR DISEÑO, CON PERMISOS Y MENOR RIESGO MALWARE
CONCLUSIÓN: LINUX ES MÁS SEGURO POR DEFECTO Y MENOS PROPENSO A ATAQUES, MIENTRAS QUE WINDOWS NECESITA MEDIDAS ADICIONALES PARA MANTENER LA SEGURIDAD	

3.6 MANTENIMIENTO

WINDOWS	LINUX (UBUNTU, DEBIAN, ROCKY LINUX)
ACTUALIZACIONES AUTOMÁTICAS CON REINICIOS FRECUENTES *PUEDE INTERRUMPIR EL TRABAJO	ACTUALIZACIONES FLEXIBLES, NORMALMENTE SIN NECESIDAD DE REINICIAR, MAYOR ESTABILIDAD
CONCLUSIÓN: LINUX PERMITE UN MANTENIMIENTO MÁS CONTINUO Y MENOS INTRUSIVO, MIENTRAS QUE EL SISTEMA OPERATIVO WINDOWS PUEDE GENERAR INTERRUPCIONES DEBIDO A REINICIOS OBLIGATORIOS	

CONCLUSIÓN FINAL

LA ELECCIÓN ENTRE WINDOWS Y LINUX DEBE BASARSE EN EL PERFIL Y NECESIDADES ESPECÍFICAS DE LOS USUARIOS Y LA EMPRESA. WINDOWS ES IDEAL PARA USUARIOS ESTÁNDAR QUE BUSCAN FACILIDAD Y

COMPATIBILIDAD, AUNQUE CON COSTES Y MAYORES REQUERIMIENTOS DE SEGURIDAD. LINUX ES MEJOR PARA ENTORNOS TÉCNICOS QUE VALORAN LA ESTABILIDAD, SEGURIDAD Y AHORRO EN LICENCIAS, AUNQUE PUEDE REQUERIR FORMACIÓN INICIAL.

UNA INFRAESTRUCTURA DUAL QUE COMBINE AMBOS SISTEMAS PERMITE APROVECHAR LAS VENTAJAS DE CADA UNO, UNA SOLUCIÓN EQUILIBRADA, SEGURA Y EFICIENTE PARA LA EMPRESA EN CRECIMIENTO.

4. DESARROLLO

4.1 FUNDAMENTACIÓN TEÓRICA: SEGURIDAD EN LA INFRAESTRUCTURA DUAL

LA SEGURIDAD ES UN PILAR FUNDAMENTAL EN LA IMPLANTACIÓN DE CUALQUIER INFRAESTRUCTURA DE SISTEMAS OPERATIVOS. EN ESTE PROYECTO, SE HAN APLICADO DISTINTAS MEDIDAS DE SEGURIDAD ADAPTADAS A CADA SISTEMA (WINDOWS Y LINUX), GARANTIZANDO LA PROTECCIÓN DE LOS EQUIPOS, USUARIOS, DATOS Y SERVICIOS, TANTO A NIVEL PREVENTIVO COMO REACTIVO. A CONTINUACIÓN, SE DETALLAN LAS ACCIONES Y CONFIGURACIONES REALIZADAS:

1. Firewall y control de tráfico

Windows:

- Configuración de Windows Defender Firewall para permitir únicamente tráfico entrante en puertos autorizados (por ejemplo, RDP para administración remota).
- Creación de reglas personalizadas para bloquear conexiones sospechosas.
- Habilitación del registro de eventos del firewall para auditoría.

Linux:

- Uso de ufw (Uncomplicated Firewall) como interfaz de configuración.
- Reglas aplicadas:

```
sudo ufw default deny incoming

sudo ufw allow ssh

sudo ufw allow from 192.168.1.0/24 to any port 22 proto tcp

sudo ufw enable
```

- Verificación:

```
sudo ufw status verbose
```

2. Gestión de permisos y privilegios

Windows:

- Creación de usuarios separados por roles: administrador, técnico, usuario estándar.
- Aplicación de directivas de seguridad mediante Políticas de Grupo (GPO):
    - Restricción del acceso a configuraciones del sistema.
    - Desactivación del panel de control y la línea de comandos para usuarios no autorizados.
    - Prohibición de uso de dispositivos USB no autorizados.

Linux:

- Control de accesos mediante grupos (`admin`, `sudo`, `users`).
- Edición del archivo `/etc/sudoers` con `visudo` para restringir comandos administrativos.
- Aplicación de permisos con `chmod` y `chown` para proteger archivos críticos:

```
sudo chmod 700 /root

sudo chown root:root /etc/shadow
```

3. Actualizaciones automáticas y mantenimiento

Windows:

- Activación de Windows Update con instalación automática.
- Configuración de política para reinicio automático fuera del horario laboral.

Linux:

- Configuración de actualizaciones automáticas con unattended-upgrades:

```
sudo apt install unattended-upgrades

sudo dpkg-reconfigure --priority=low unattended-upgrades
```

- Automatización mediante cron para notificaciones de parches pendientes.

4. Protección antimalware

Windows:

- Uso de Windows Defender Antivirus con protección en tiempo real.
- Configuración de análisis programados semanales y alertas en caso de infección.

Linux:

- Instalación de ClamAV para análisis bajo demanda de archivos descargados o dispositivos externos:

```
sudo apt install clamav

sudo freshclam

sudo clamscan -r /home
```

- Escaneo periódico mediante cron y envío de logs por correo interno del sistema.

5. Cifrado de datos y discos

Windows:

- Activación de BitLocker en unidades sensibles, con cifrado AES.
- Uso de TPM (Trusted Platform Module) para asegurar la integridad del arranque.

Linux:

- Cifrado de particiones de backup y datos sensibles con LUKS:

```
sudo cryptsetup luksFormat /dev/sdX

sudo cryptsetup luksOpen /dev/sdX secure_data
```

- Montaje automático seguro con clave cifrada o passphrase.

6. Registros y auditoría de seguridad

Windows:

- Activación del Auditor de Seguridad para registrar:
  - Intentos de inicio de sesión fallidos.
  - Cambios en políticas del sistema.
  - Instalación de software.
- Visualización mediante el Visor de Eventos (`eventvwr.msc`).

Linux:

- Monitoreo de archivos de log críticos:

  `/var/log/auth.log`, `/var/log/syslog`, `/var/log/fail2ban.log`

- Instalación de herramientas como logwatch o logrotate para reportes automáticos y gestión de tamaño de logs.
- Configuración de fail2ban para bloquear IPs tras múltiples intentos fallidos de acceso SSH.

7. Medidas adicionales

- Desactivación de servicios innecesarios para reducir superficie de ataque.
- Restricción de puertos abiertos solo a los mínimos necesarios.
- Pruebas básicas de escaneo de vulnerabilidades con herramientas como:
  - nmap
  - lynis

    Microsoft Baseline Security Analyzer (MBSA)

- Concienciación básica al usuario:
  - Políticas de contraseñas seguras.
  - Bloqueo automático de sesión tras inactividad.

Materiales y métodos: estrategias de búsqueda, metodología y técnicas utilizadas

Resultados y análisis