PORTADA

TÍTULO DEL PROYECTO

https://github.com/wheresmyicecream/Trabajo-ISO-/branches

ABSTRACT

Nombre del alumno o de la alumna:Javier Granados (Scrum Master), Daniel Alaez, Marco García, Antonio Medina

Curso académico:1 ASIR

Tutora/Tutor del proyecto: Carmelo

ÍNDICE PAGINADO

1. JUSTIFICACIÓN DEL PROYECTO

2. INTRODUCCIÓN

3. OBJETIVOS

A. OBJETIVO GENERAL

B. OBJETIVOS ESPECÍFICOS

4. DESARROLLO

1. FUNDAMENTACIÓN TEÓRICA: lo que vamos a hacer, procedimientos, resolución de la hipótesis o situaciones planteadas, tareas a realizar
2. Materiales y métodos: estrategias de búsqueda, metodología y técnicas utilizadas
3. Resultados y análisis

(Sesión 1: Daniel Análisis comparativo, Antonio; Seguridad, Javier: Github y esquema)

(Sesión 2: Daniel instalacion linux , Antonio: análisis de necesidades e introducción (red team vs blue team), Javier: KanBan, raíces github)

○ Introducción y contexto

Este documento presenta una propuesta para implantar una infraestructura de sistemas operativos mixta en una pequeña empresa en crecimiento, con el fin de garantizar una plataforma estable, segura y rentable, adaptada a los distintos perfiles de usuario existentes en la organización.

Sector de la empresa: Servicios informáticos y soporte técnico

Tamaño: 12 empleados

Modalidad de trabajo: Presencial con opción de teletrabajo parcial

Sistema operativo: Linux (Ubuntu)

Necesidades y perfiles de usuario:

Perfil	Nº Usuarios	Tareas principales
Dirección	1	Gestión, informes, comunicación externa.
Administración	2	Facturación, nóminas, ofimática, correo.
Técnicos de soporte	3	Diagnóstico remoto, configuración de redes, asistencia a clientes.
Desarrolladores	6	Programación, testing, servidores locales, Git.

Ventajas del uso de linux:

Aspecto	Ventajas en Linux
Coste	Sin licencias de sistema operativo ni ofimática.
Seguridad	Menor exposición a malware, actualizaciones constantes y rápidas.
Flexibilidad	Alta personalización del entorno según necesidades del usuario.
Rendimiento	Requiere menos recursos que Windows; ideal para hardware modesto.
Automatización	Fácil integración con scripts, cron, backups automáticos.
Compatibilidad	Compatible con la mayoría de herramientas web, correo y edición.

Implementar exclusivamente Linux (Ubuntu) en esta pequeña empresa es una solución eficiente, segura y económicamente sostenible. Especialmente adecuada para empresas con cierto perfil técnico, Linux permitirá crecer sin depender de licencias, ofreciendo rendimiento y control total sobre los sistemas.

○ Análisis comparativo (Windows vs Linux

Analizar distintas versiones de Windows y distribuciones Linux (Ubuntu, Debian, Rocky Linux, Windows 11

Windows

Windows 10 es una de las versiones más populares y ampliamente utilizadas. Ofrece una interfaz moderna, soporte para una gran variedad de hardware y software, y es ampliamente adoptado tanto en entornos domésticos como empresariales. Recibe actualizaciones de seguridad periódicas y tiene soporte para aplicaciones legacy y modernas (UWP).

Windows 11

Windows 11 es la versión más reciente (lanzada en 2021). Introduce una interfaz más pulida, mejoras en la gestión de ventanas (Snap Layouts), integración con Microsoft Teams y requisitos de hardware más estrictos (como TPM 2.0 y Secure Boot). Está orientado a mejorar la productividad y la seguridad, pero puede no ser compatible con hardware más antiguo.

Ventajas de Windows (en general):
Compatibilidad con la mayoría del software comercial, facilidad de uso, soporte técnico amplio, integración con servicios de Microsoft.

Desventajas:
Licencia de pago, menos personalizable, más susceptible a malware si no se toman precauciones.

Linux

Ubuntu

Ubuntu es una de las distribuciones más populares y amigables para el usuario. Basada en Debian, se centra en la facilidad de uso, actualizaciones regulares y una gran comunidad. Es ideal tanto para principiantes como para usuarios avanzados. Se utiliza mucho en escritorios, servidores y entornos de desarrollo.

Debian

Debian es conocida por su estabilidad y robustez. Es la base de muchas otras distribuciones (incluyendo Ubuntu). Suele tener paquetes más antiguos pero muy probados, lo que la hace ideal para servidores y sistemas donde la estabilidad es prioritaria sobre la novedad.

Rocky Linux

Rocky Linux es una distribución empresarial, creada como reemplazo de CentOS tras el cambio de enfoque de Red Hat. Es compatible a nivel binario con Red Hat Enterprise Linux (RHEL), lo que la hace ideal para entornos empresariales que requieren estabilidad, soporte a largo plazo y compatibilidad con software de servidor.

Ventajas de Linux (en general):
Gratuito y de código abierto, altamente personalizable, menos susceptible a virus, ideal para servidores y desarrollo, gran variedad de distribuciones para diferentes necesidades.

Desventajas:

Curva de aprendizaje para usuarios nuevos, menor compatibilidad con software comercial (especialmente juegos y aplicaciones profesionales), soporte técnico más comunitario que comercial (excepto en distribuciones empresariales).

Característica	Windows 10/11	Ubuntu	Debian	Rocky Linux
Licencia	Comercial	Libre	Libre	Libre
Facilidad de uso	Muy alta	Alta	Media	Media
Actualizaciones	Automáticas	Regulares	Menos frecuentes	Regulares
Estabilidad	Alta	Alta	Muy alta	Muy alta
Soporte software	Muy amplio	Amplio (open source)	Amplio (open source)	Empresarial
Hardware antiguo	Mejor en Win10	Bueno	Excelente	Bueno
Orientación	Hogar/empresa	Escritorio/servidor	Servidor/infraestructura	Servidor/empresa

---

Resumen

- **Windows 10/11:** Mejor para usuarios que necesitan compatibilidad con software comercial, juegos o buscan una experiencia lista para usar.
- **Ubuntu**: Ideal para quienes quieren empezar en Linux, desarrolladores o buscan un sistema de escritorio moderno y fácil de usar.
- Debian: Perfecto para servidores o usuarios que priorizan la estabilidad y la robustez.
- Rocky Linux: Recomendado para empresas que buscan una alternativa gratuita y estable a RHEL/CentOS.

○ Escenarios de uso recomendados

○ Instalación paso a paso ○ Configuración básica y avanzada

○ Seguridad (firewall, políticas, actualizaciones, antivirus, logs)

Firewall

El firewall es la primera línea de defensa para bloquear accesos no autorizados a la red y a los sistemas.

Windows

Incorpora Windows Defender Firewall, que permite crear reglas de entrada y salida.
Soporta integración con políticas de grupo (GPO) en entornos empresariales.
Interfaz gráfica fácil de gestionar, incluso por personal no experto.

Linux (Ubuntu, Debian, Rocky)

Usa herramientas como UFW (Uncomplicated Firewall) en Ubuntu y firewalld o iptables en Debian y Rocky Linux.
Aunque más técnicas, ofrecen un alto grado de control y personalización.
Muy recomendables en servidores para restringir accesos por puertos y protocolos.

Recomendación

Implementar firewalls en todos los equipos y servidores, con reglas específicas según el rol del sistema (por ejemplo, permitir solo http/https en servidores web).

Políticas de seguridad

Las políticas de seguridad definen cómo deben comportarse los usuarios y el sistema ante diversos escenarios.

Windows

Uso de GPO (Group Policy Objects) en entornos con Active Directory para controlar contraseñas, bloqueo de dispositivos, ejecución de programas, etc.
Políticas locales disponibles para equipos independientes.

Linux

Control de permisos de archivos mediante el sistema de usuarios y grupos.
Uso de herramientas como sudo, SELinux (Rocky Linux) y AppArmor (Ubuntu) para controlar el acceso y limitar privilegios.

Recomendación

Aplicar el principio de mínimos privilegios en ambos entornos. En Linux, evitar el uso del

usuario root en tareas diarias. En Windows, usar cuentas estándar para usuarios normales y cuentas con privilegios limitados para tareas administrativas.

Actualizaciones

Las actualizaciones corrigen vulnerabilidades que pueden ser explotadas por atacantes.

Windows

Sistema de actualizaciones automáticas a través de Windows Update.

En empresas, se puede usar WSUS para centralizar y controlar la distribución de parches.

Linux

Actualizaciones mediante gestores de paquetes (apt, dnf, yum).

Posibilidad de automatizar actualizaciones de seguridad mediante cron jobs o herramientas como unattended-upgrades.

Recomendación

Mantener todos los sistemas actualizados automáticamente, especialmente aquellos expuestos a internet (como servidores web o bases de datos). Comprobar periódicamente que no haya fallos en los procesos de actualización.

Antivirus y antimalware

La protección contra malware es especialmente crítica en sistemas Windows, pero también puede ser útil en Linux.

Windows

Windows 11 incluye Microsoft Defender Antivirus, que ofrece protección en tiempo real, análisis programados y detección de amenazas basada en la nube.

Puede complementarse con soluciones empresariales como ESET, Bitdefender o Kaspersky.

Linux

Aunque menos afectado por malware tradicional, es recomendable usar herramientas como ClamAV en servidores de correo o archivos.

Los sistemas Linux deben protegerse especialmente contra ataques como rootkits o backdoors.

Recomendación

Implementar antivirus al menos en los equipos Windows y escáneres de malware en servidores Linux que gestionen contenido compartido (correos, ftp, etc.).

Registros de actividad (logs)

Los logs permiten detectar anomalías, hacer auditorías y cumplir con normativas.

Windows

Usa el visor de eventos para registrar accesos, errores del sistema, cambios en la configuración y más.

Se puede integrar con soluciones SIEM para análisis centralizado.

Linux

Usa archivos de log como /var/log/syslog, /var/log/auth.log, /var/log/secure.

Herramientas como rsyslog, journald y Logwatch permiten centralizar, monitorear y resumir eventos.

Recomendación

Centralizar los logs en un servidor dedicado, tanto en Windows como en Linux.

Implementar herramientas de monitoreo y alertas para detectar intentos de acceso fallidos, cambios sospechosos, etc.

Debemos combinar correctamente las herramientas y buenas prácticas de ambos entornos operativos. La combinación de firewalls activos, políticas bien definidas, actualizaciones constantes, protección antivirus adecuada y una buena gestión de logs es esencial para garantizar la seguridad a medida que la empresa crece.

Seguridad de nuestra empresa:

Implementación recomendada:

-Firewall: UFW activo en todos los equipos.

-**Actualizaciones automáticas**

-**Antivirus básico**

-**Cuentas de usuario limitadas:** Sin permisos root directo; uso de `sudo`.

-**Acceso remoto seguro:** VPN y SSH con autenticación por clave.

-**Centralización de logs:** syslog y revisión con Logwatch o scripts periódicos.

Posibles vulnerabilidades:

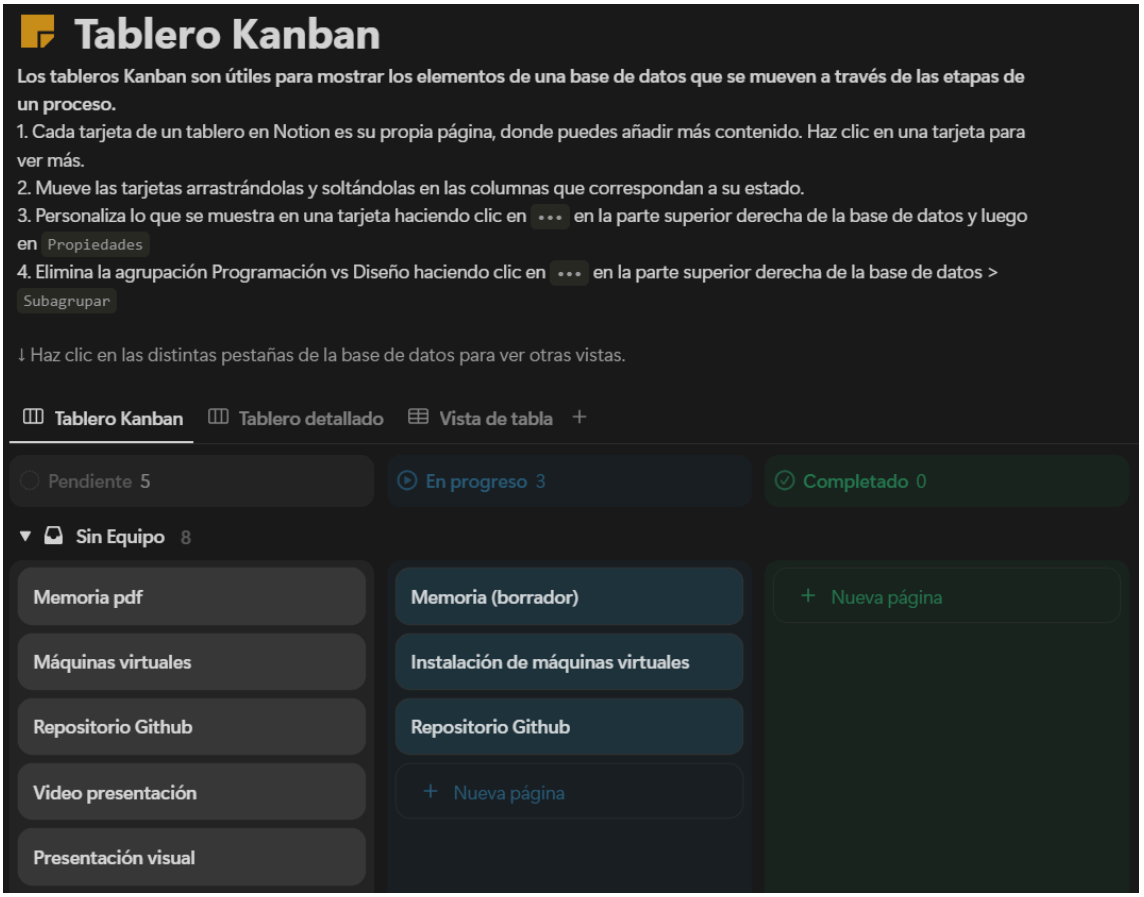
Vulnerabilidad	Riesgo Principal	Medidas de Corrección Clave
Cuentas mal configuradas	Acceso no autorizado	Crear usuarios sin permisos; controlar uso de `sudo`; activar 2FA
Sistema desactualizado	Exposición a exploits conocidos	Activar `unattended-upgrades`; aplicar parches críticos
Puertos abiertos innecesarios	Ataques remotos por servicios no usados	Configurar `ufw`; escaneo con `nmap`; cerrar puertos no requeridos
Acceso remoto inseguro	Intercepción o acceso no autorizado	Usar SSH con claves; desactivar contraseñas; VPN segura con WireGuard
Software de origen no confiable	Malware o puertas traseras	Usar repos oficiales; verificar firmas de paquetes; auditar software
Permisos incorrectos en archivos	Acceso a datos sensibles	Revisar y ajustar permisos con `chmod`, `chown`, uso de ACLs si es necesario
Logs no supervisados	Actividades maliciosas pasan desapercibidas	Centralizar logs; revisar `/var/log`; configurar alertas
Malware (aunque poco común)	Daños al sistema o fuga de datos	Usar ClamAV; evitar scripts externos; escaneo con `rkhunter`, `chkrootkit`

Red Team VS Blue Team:

Rol / Aspecto	🔴 Red Team (Ofensivo)	🔵 Blue Team (Defensivo)
Objetivo principal	Simular ataques reales para detectar debilidades	Defender la infraestructura y mitigar amenazas
Mentalidad	Atacante (piensa como un hacker)	Defensor (piensa como un guardián)
Actividades clave	- Pentesting - Ingeniería social - Explotación de fallos	- Monitorización - Gestión de incidentes - Aplicación de parches
Herramientas comunes	Metasploit, Nmap, Kali Linux, Burp Suite	SIEM, UFW, fail2ban, ClamAV, rkhunter, Suricata
Conocimientos requeridos	Exploits, redes, evasión de defensas	Análisis de logs, hardening, respuesta ante incidentes
Resultados esperados	Informe de vulnerabilidades y vectores de ataque	Infraestructura robusta, planes de contingencia, alertas
Tipo de pruebas	Intrusiones simuladas, phishing, escaneo	Análisis de tráfico, integridad de sistemas, backups
Interacción interna	Simulan ser un enemigo	Actúan como equipo interno de TI o seguridad
Relación entre ellos	Retan la seguridad existente	Aprenden del ataque y mejoran la protección

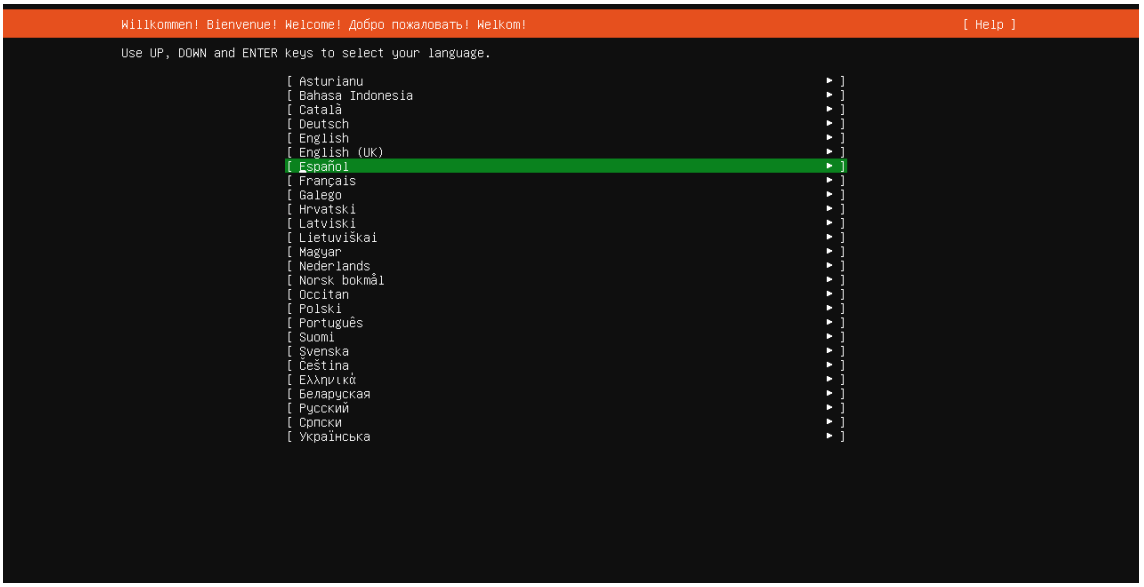
- **Automatización y scripts de mantenimiento**
- **Gestión de usuarios y permisos**
- **Documentación técnica**
- **Conclusiones y propuesta final**
- **Anexos: capturas, comandos usados, logs, configuración**

Tablero KanBan (también subido a imágenes github)

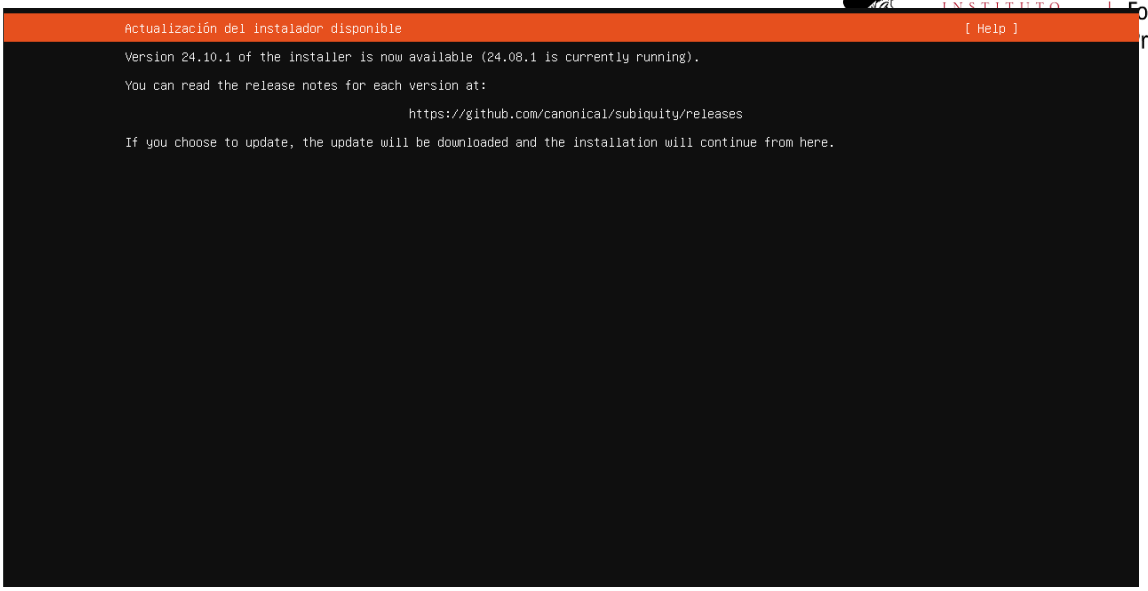


Al iniciarse la máquina nos saldrá para elegir estas 2 opciones, debemos esperar sin elegir nada.

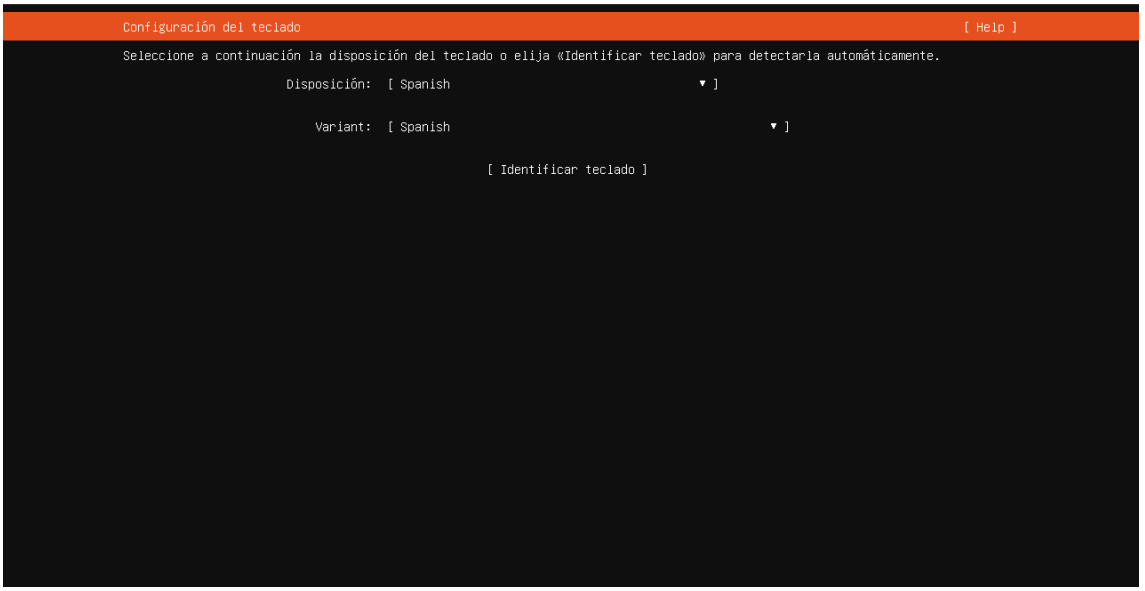
Después, seleccionaremos el idioma “Español” y le daremos a la tecla “enter”.



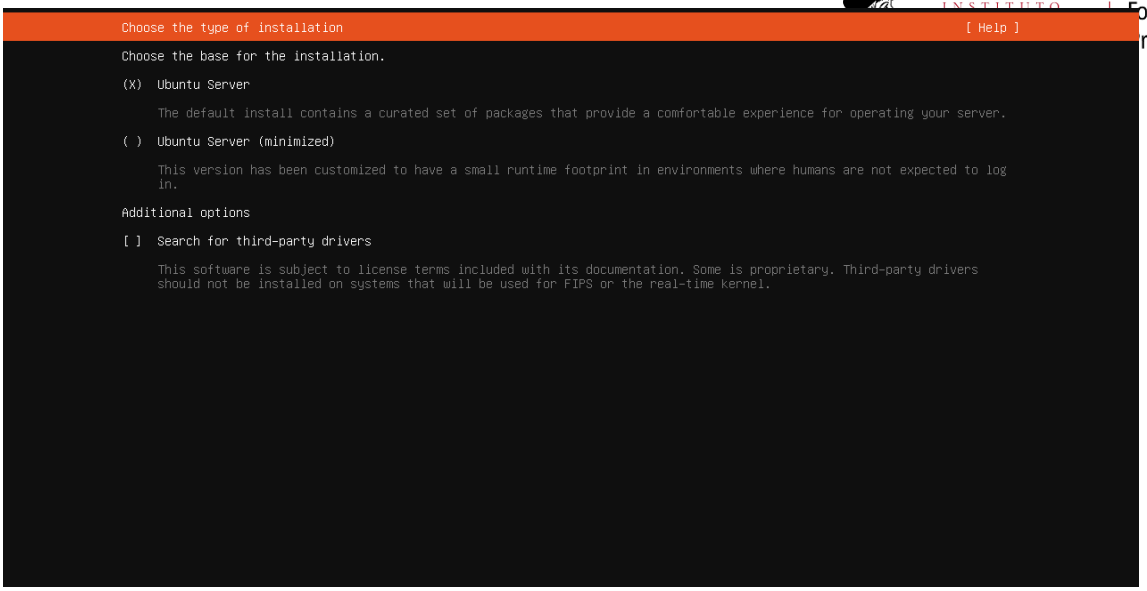
Ahora, le daremos a la tecla enter en la opción de “Continuar sin actualizar”.



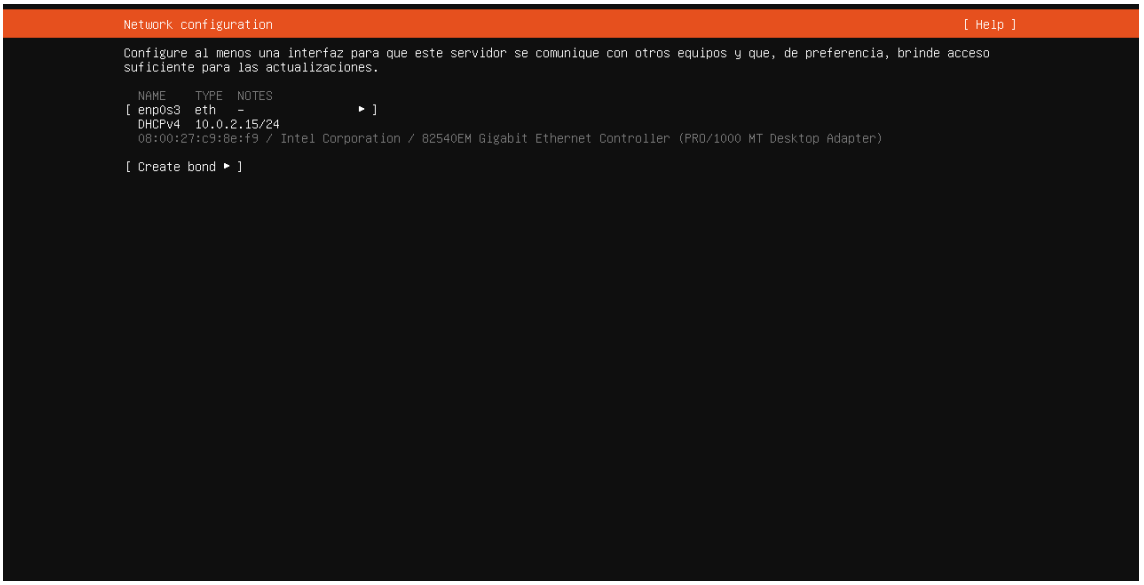
Ahora le daremos a la opción de “Hecho”



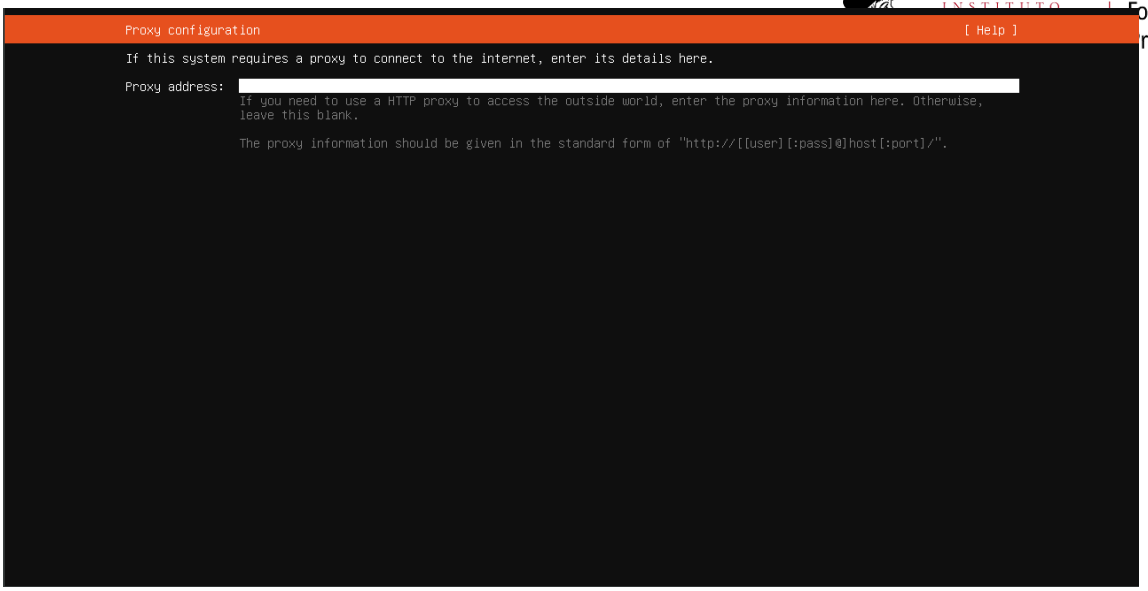
Ahora dejaremos la opción de “Ubuntu Server” y le daremos a “Hecho”.



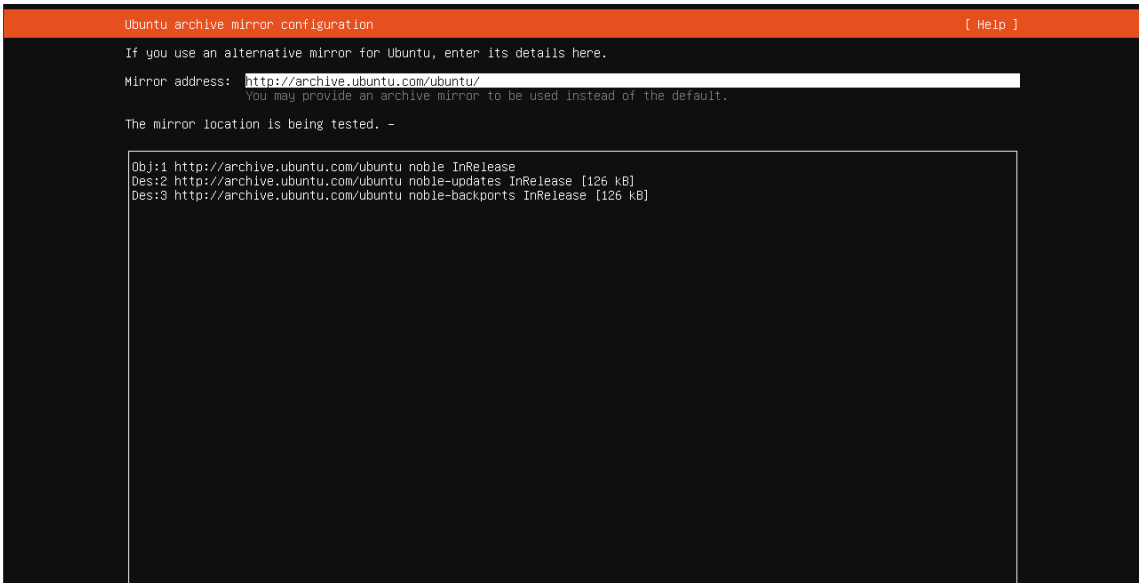
Ahora le volveremos a dar a “Hecho”.



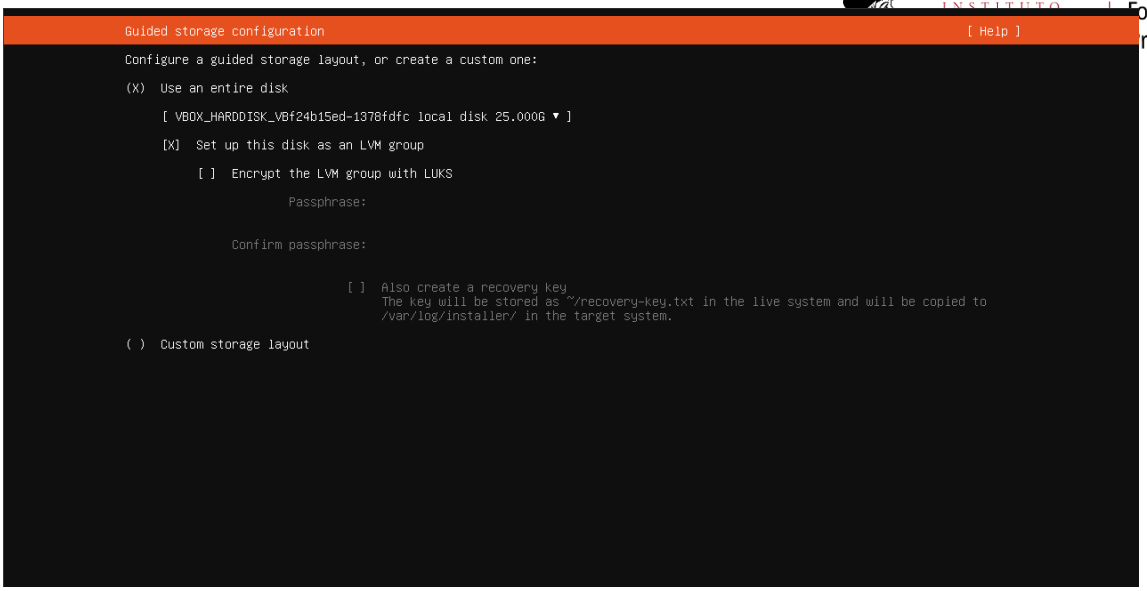
Le volveremos a dar a “Hecho” ya que no tenemos proxy.



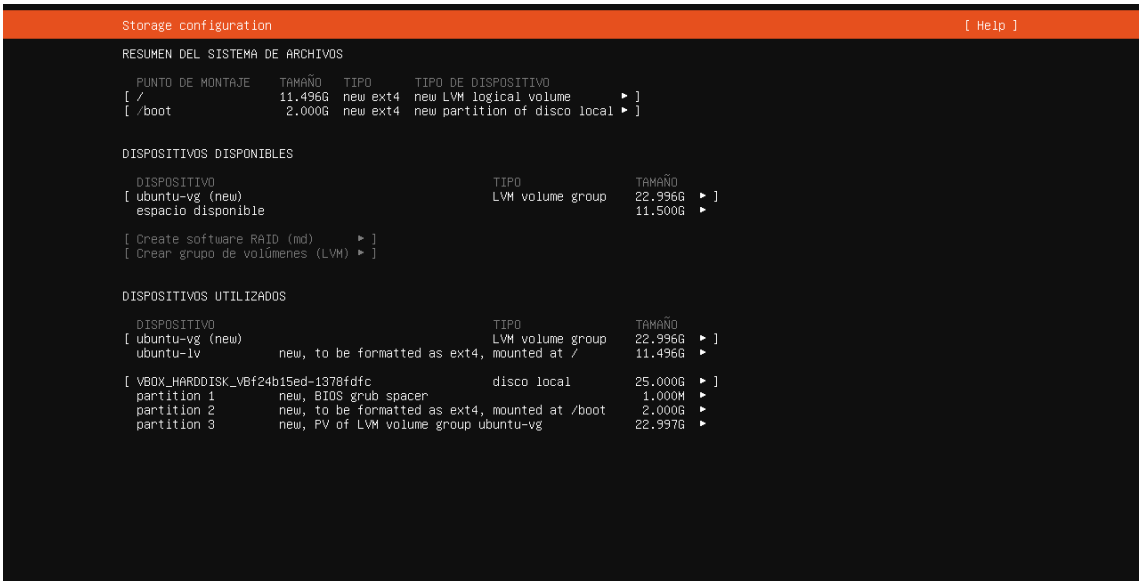
Ahora, le daremos a “Hecho” otra vez.



Dejaremos marcada la opción de “Use entire disk” y le daremos a “Hecho”.

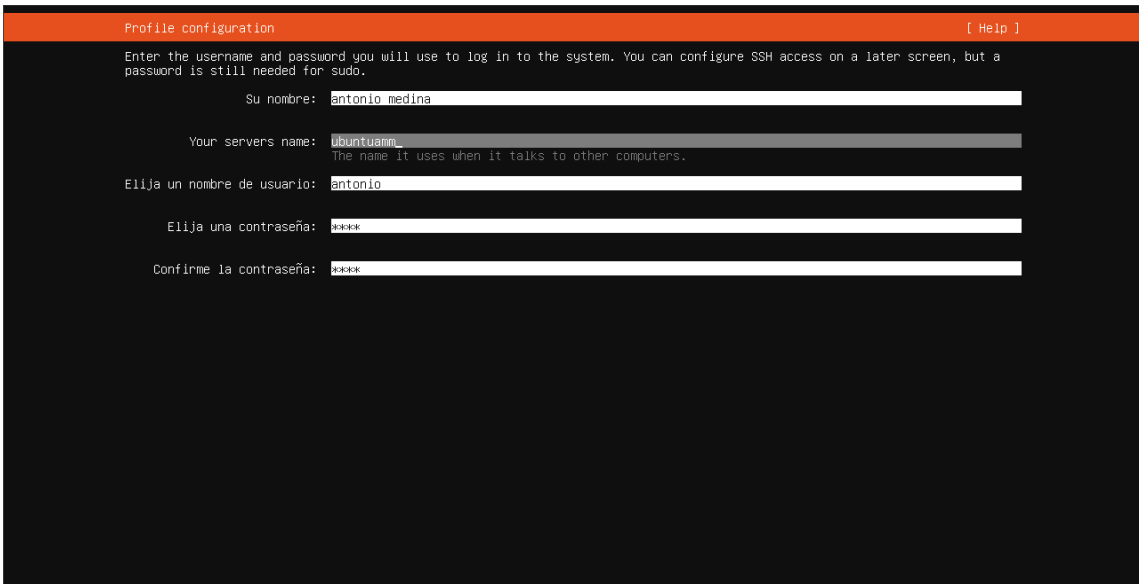


Le daremos a “Hecho”.

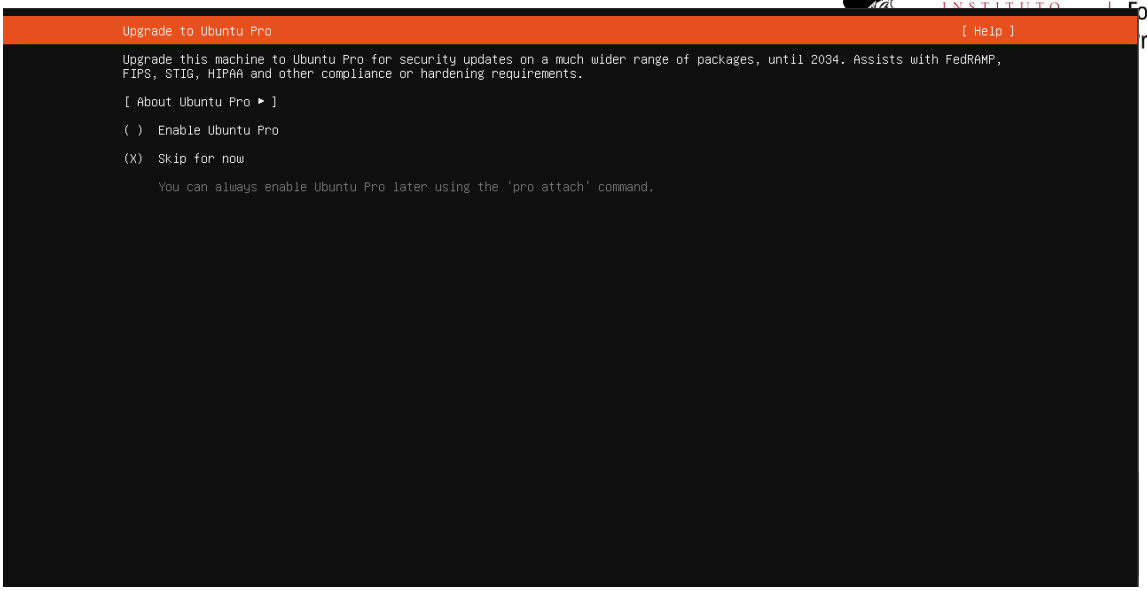


Ahora le daremos a “Continuar”.

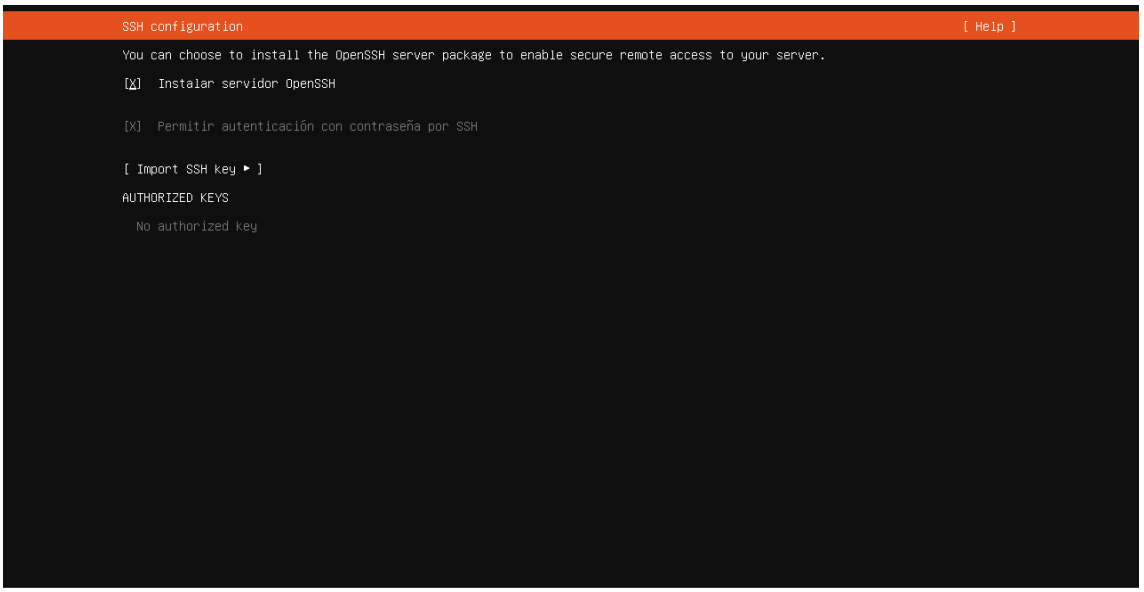
Ahora introduciremos nuestro nombre, el nombre del servidor, el nombre de usuario y la contraseña, después le daremos a “Hecho”.



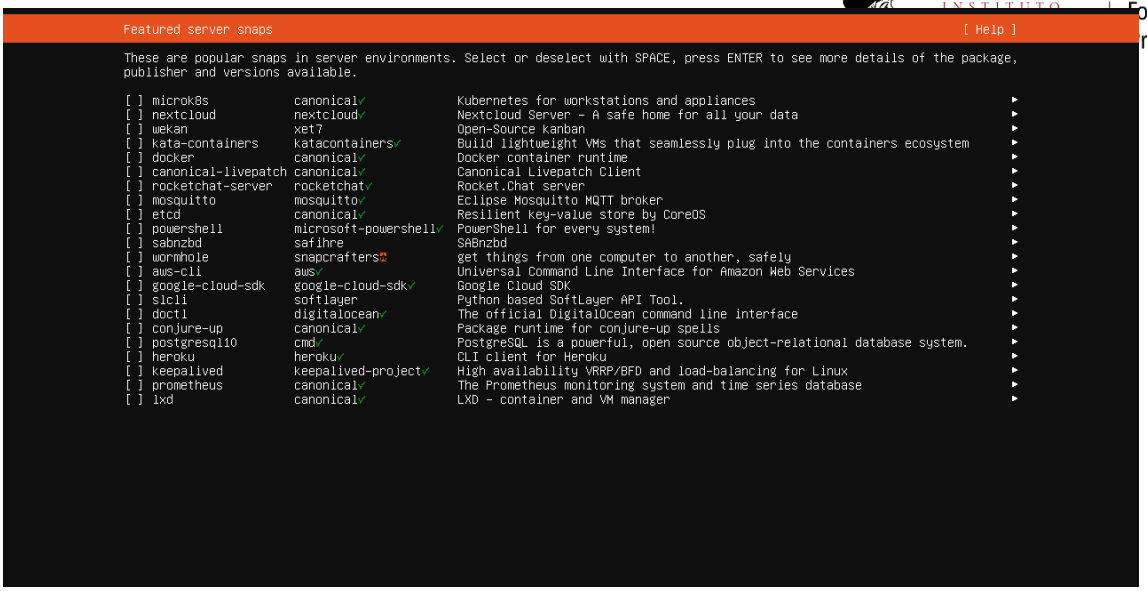
Ahora le daremos a “Continuar”.



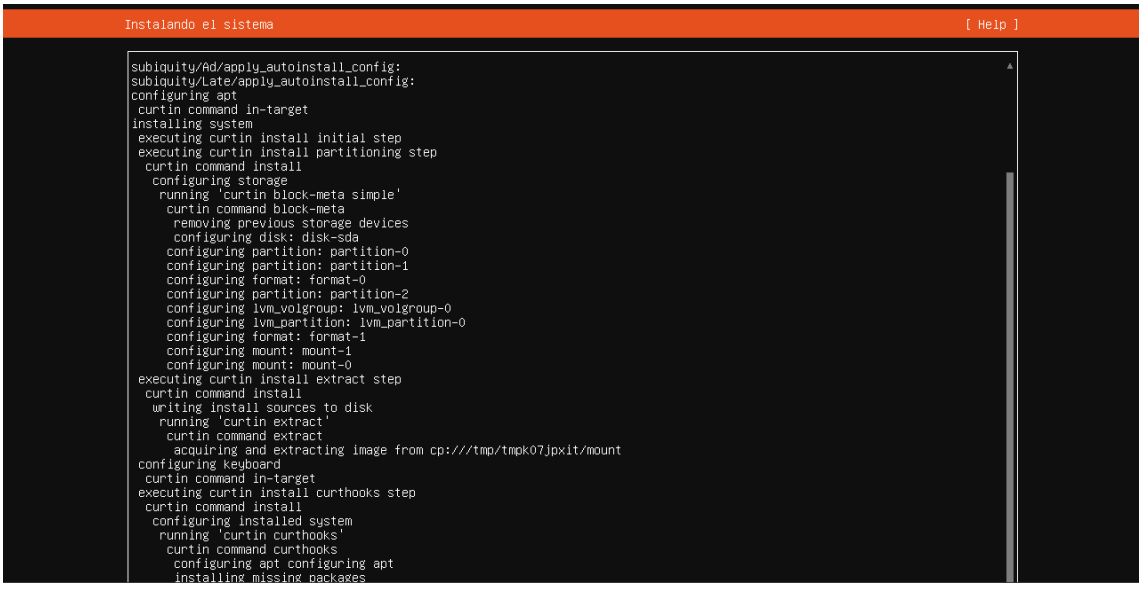
Ahora marcaremos la casilla de “Instalar servidor OpenSSH” y le daremos a “Hecho”.



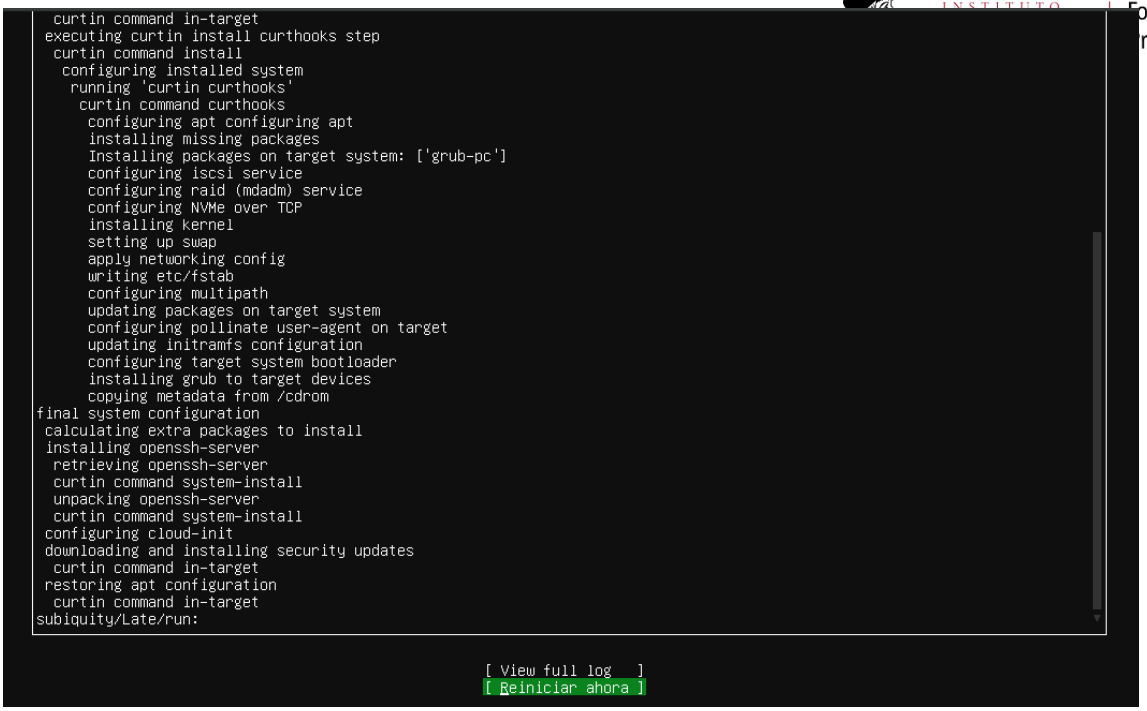
Ahora le daremos a “Hecho”.



Ahora tendremos que esperar y le daremos a



Una vez hayamos terminado le daremos a enter en la opción de “Reiniciar ahora” para reiniciar nuestra máquina.


curtin command in-target
executing curtin install curthooks step
curtin command install
configuring installed system
running 'curtin curthooks'
curtin command curthooks
configuring apt configuring apt
installing missing packages
Installing packages on target system: ['grub-pc']
configuring iscsi service
configuring raid (mdadm) service
configuring NVMe over TCP
installing kernel
setting up swap
apply networking config
writing etc/fstab
configuring multipath
updating packages on target system
configuring pollinate user-agent on target
updating initramfs configuration
configuring target system bootloader
installing grub to target devices
copying metadata from /cdrom
final system configuration
calculating extra packages to install
installing openssh-server
retrieving openssh-server
curtin command system-install
unpacking openssh-server
curtin command system-install
configuring cloud-init
downloading and installing security updates
curtin command in-target
restoring apt configuration
curtin command in-target
subiquity/Late/run:
[View full log]
[Reiniciar ahora]

5. CONCLUSIONES

6. LÍNEAS DE INVESTIGACIÓN FUTURAS

(No son obligatorios, pero pueden aparecer)

7. BIBLIOGRAFÍA

8. ANEXOS

9. OTROS PUNTOS

(No son obligatorios, pero pueden aparecer)

- Aportaciones personales
- Retos profesionales
- Restos personales
- Agradecimientos