

# Facilitando la Administración de Servidores Linux: Exploración y Aplicación de Cockpit como Interfaz de Gestión Web

Autor: Omar Elhani

Curso : 2º ASIR

## Índice

Índice.....	2
1. Objetivos .....	3
1.1 Objetivos iniciales .....	3
1.2 Objetivos conseguidos .....	3
2. Escenario necesario para la realización del proyecto .....	4
3. Fundamentos teóricos y conceptos .....	7
4. Descripción detallada de lo realizado .....	10
5. Conclusiones y propuestas para seguir trabajando sobre el tema .....	21
6. Bibliografía .....	23

## 1. Objetivos

### 1.1 Objetivos iniciales

- Implementar Cockpit en un entorno Linux.
- Explorar y documentar las funcionalidades de Cockpit.
- Evaluar la facilidad de uso y eficiencia de Cockpit en la administración de sistemas.
- Comparar Cockpit con otras herramientas de administración de sistemas.
- Proporcionar una guía detallada de instalación y configuración.
- Realizar pruebas prácticas para demostrar las capacidades de Cockpit.
- Documentar los resultados y las observaciones de las pruebas realizadas.

### 1.2 Objetivos conseguidos

- **Implementación exitosa de Cockpit:** Se logró instalar y configurar Cockpit en un servidor Linux.
- **Documentación de funcionalidades:** Se detallaron las funcionalidades clave de Cockpit, incluyendo configuración de red, gestión de usuarios, monitoreo del sistema y administración de almacenamiento.
- **Evaluación de facilidad de uso:** Se evaluó la interfaz de Cockpit en términos de facilidad de uso y eficiencia, destacando su accesibilidad y simplicidad.
- **Comparación con otras herramientas:** Se comparó Cockpit con otras herramientas de administración de sistemas, resaltando sus ventajas y desventajas.
- **Guía de instalación y configuración:** Se creó una guía detallada paso a paso para la instalación y configuración de Cockpit.
- **Pruebas prácticas:** Se realizaron pruebas prácticas que demuestran las capacidades de Cockpit, incluyendo la configuración de firewall, monitorización del rendimiento y gestión de actualizaciones del sistema.
- **Documentación de resultados:** Se documentaron los resultados de las pruebas, proporcionando una visión clara de las capacidades y limitaciones de Cockpit.

## 2. Escenario necesario para la realización del proyecto

Para llevar a cabo este proyecto, he decidido utilizar el software de virtualización VirtualBox para crear y gestionar las máquinas virtuales necesarias. El escenario constará de una máquina servidor de Cockpit y dos máquinas clientes. Estas máquinas clientes proporcionarán servicios específicos que serán monitorizados por el servidor Cockpit. A continuación, se describe detalladamente el escenario:

### Máquina Servidor de Cockpit

- **Descripción:** Esta máquina actuará como el servidor central para la administración y monitorización de las máquinas clientes. Utilizará Cockpit, una interfaz web de administración para servidores Linux.
- **Sistema Operativo:** Debian 12 (BookWarm)
- **Recursos Asignados:**
  - CPU: 3 núcleos
  - RAM: 4 GB
  - Almacenamiento: 50 GB
- **Software Instalado:**
  - Cockpit
- **Funcionalidad:**
  - Proporcionará una interfaz web accesible a través del puerto 9090 para la administración y monitorización de las máquinas clientes.
  - Monitorizará el estado del sistema, los recursos utilizados, los servicios en ejecución y las actualizaciones necesarias en las máquinas clientes.

### Máquina Cliente 1

- **Descripción:** Esta máquina proporcionará un servicio web que será monitorizado por el servidor Cockpit.
- **Sistema Operativo:** Ubuntu 20.04 LTS
- **Recursos Asignados:**
  - CPU: 1 núcleo
  - RAM: 2 GB
  - Almacenamiento: 15 GB
- **Software Instalado:**
  - Apache HTTP Server (para servir contenido web)
  - OpenSSH Server

- **Funcionalidad:**
  - Servirá una página web básica en el puerto 80.
  - Será monitorizada por Cockpit en términos de estado del servicio web, uso de recursos y disponibilidad.

### Máquina Cliente 2

- **Descripción:** Esta máquina proporcionará un servicio de base de datos que será monitorizado por el servidor Cockpit.
- **Sistema Operativo:** CentOS 8
- **Recursos Asignados:**
  - CPU: 1 núcleo
  - RAM: 2 GB
  - Almacenamiento: 15 GB
- **Software Instalado:**
  - MySQL Server (para la gestión de bases de datos)
  - OpenSSH Server
- **Funcionalidad:**
  - Gestionará una base de datos accesible en el puerto 3306.
  - Será monitorizada por Cockpit para asegurar la correcta ejecución del servicio de base de datos, el uso de recursos y la integridad de los datos.

### Configuración de la Red

- **Tipo de Red:** Red interna de VirtualBox (Host-Only Network) para asegurar la comunicación entre las máquinas sin exposición directa a Internet.
- **Direcciones IP:**
  - Servidor Cockpit:
  - Máquina Cliente 1:
  - Máquina Cliente 2:

### Procedimiento de Implementación

1. **Instalación de VirtualBox:** Instalar VirtualBox en la máquina anfitriona y crear una red interna.
2. **Creación de Máquinas Virtuales:** Crear las tres máquinas virtuales con las especificaciones mencionadas.

3. **Instalación de Sistemas Operativos:** Instalar los sistemas operativos correspondientes en cada máquina virtual.
4. **Configuración de Servicios:**
  - En la máquina servidor, instalar y configurar Cockpit.
  - En la máquina cliente 1, instalar y configurar Apache HTTP Server.
  - En la máquina cliente 2, instalar y configurar MySQL Server.
5. **Configuración de Red:** Asignar direcciones IP estáticas a cada máquina y asegurar la conectividad entre ellas.
6. **Monitorización y Pruebas:** Utilizar Cockpit para monitorizar los servicios y recursos de las máquinas clientes, verificando que todo funcione correctamente.

### 3. Fundamentos teóricos y conceptos

#### Introducción a Cockpit

Cockpit es una interfaz web interactiva diseñada para la administración y monitorización de servidores Linux. Proporciona una manera sencilla y gráfica de gestionar diversas tareas administrativas, permitiendo a los administradores de sistemas controlar y supervisar el estado de sus servidores de manera eficiente.

#### Arquitectura de Cockpit

La arquitectura de Cockpit se basa en una estructura modular, lo que permite su extensión y personalización según las necesidades del usuario. Los principales componentes de Cockpit incluyen:

- **Frontend Web:** La interfaz gráfica de Cockpit accesible a través de un navegador web. Esta interfaz se comunica con el backend para mostrar información y permitir la administración del servidor.
- **Backend:** Un conjunto de scripts y servicios que interactúan directamente con el sistema operativo y los componentes del servidor. Estos scripts realizan tareas administrativas y envían la información al frontend.
- **WebSocket:** Cockpit utiliza WebSocket para mantener una comunicación bidireccional en tiempo real entre el frontend y el backend. Esto permite la actualización instantánea de la información mostrada en la interfaz web.
- **Sistema de Plugins:** Cockpit es extensible mediante plugins que añaden funcionalidades adicionales, como la administración de contenedores, máquinas virtuales, almacenamiento, entre otros.

#### Interfaz de Cockpit

La interfaz de Cockpit está diseñada para ser intuitiva y fácil de usar. Los principales elementos de la interfaz incluyen:

- **Dashboard:** Una vista general del estado del sistema, mostrando métricas clave como el uso de CPU, memoria, almacenamiento y la actividad de red.
- **Journal Logs:** Acceso a los logs del sistema, permitiendo la visualización y búsqueda de eventos importantes.

- **Servicios:** Gestión de los servicios del sistema, permitiendo iniciar, detener y reiniciar servicios fácilmente.
- **Actualizaciones:** Administración de actualizaciones del sistema, facilitando la instalación de parches y nuevas versiones del software.
- **Terminal:** Acceso a una terminal integrada para realizar tareas avanzadas directamente desde el navegador.

### Administración con Cockpit

Cockpit facilita varias tareas administrativas que son fundamentales para la gestión de servidores. Algunas de las funciones principales incluyen:

- **Monitorización del Sistema:** Proporciona una visión en tiempo real del rendimiento del sistema, permitiendo a los administradores identificar y solucionar problemas rápidamente.
- **Gestión de Usuarios:** Permite la creación, modificación y eliminación de cuentas de usuario, así como la gestión de permisos y roles.
- **Gestión de Redes:** Configuración y monitorización de interfaces de red, incluyendo la gestión de conexiones y la configuración de parámetros de red.
- **Administración de Almacenamiento:** Facilita la gestión de discos y volúmenes, permitiendo la creación y modificación de particiones, así como la monitorización del uso del almacenamiento.
- **Administración de Contenedores y Máquinas Virtuales:** Con plugins adicionales, Cockpit puede gestionar contenedores Docker y máquinas virtuales, proporcionando una interfaz gráfica para su administración.

### Seguridad en Cockpit

La seguridad es una parte integral de Cockpit. Utiliza los mecanismos de autenticación del sistema operativo, garantizando que solo los usuarios autorizados puedan acceder a la interfaz de administración. Además, la comunicación entre el navegador y el servidor Cockpit está protegida mediante HTTPS, asegurando la confidencialidad e integridad de los datos transmitidos.

### Conclusión

Cockpit es una herramienta poderosa y flexible para la administración de servidores Linux. Su interfaz web intuitiva y sus capacidades de monitorización y gestión en tiempo real la convierten en una solución ideal para administradores de sistemas que buscan simplificar y centralizar sus tareas administrativas. Con su arquitectura



modular y extensible, Cockpit puede adaptarse a una amplia variedad de entornos y necesidades, proporcionando una plataforma robusta para la administración de infraestructura TI.

## 4. Descripción detallada de lo realizado

En este apartado explicaremos como se han realizado los pasos de instalación y configuración de la herramienta. Veremos también las funciones exploradas y las pruebas que se han hecho.

### 4.1 Instalación Cockpit en Debian 12

Antes de comenzar, nos aseguraremos de que nuestro sistema esté actualizado. Abrimos una terminal y ejecuta los siguientes comandos:

```
sudo apt update
```

```
sudo apt upgrade -y
```

Procederemos a instalar Cockpit en nuestra máquina:

```
sudo apt install cockpit -y
```

Iniciaremos y habilitaremos el servicio:

```
sudo systemctl start cockpit
```

```
sudo systemctl enable cockpit
```

Si tenemos un firewall configurado, necesitas abrir el puerto 9090 para permitir el acceso a la interfaz web de Cockpit. Si estás utilizando UFW (Uncomplicated Firewall), ejecuta los siguientes comandos:

```
sudo ufw allow 9090/tcp
```

```
sudo ufw reload
```

Abrimos un navegador web y accedemos a Cockpit utilizando la dirección IP de nuestro servidor seguido del puerto 9090. Por ejemplo:

<http://<tu-dirección-ip>:9090>

## 4.2 Configuración en las Máquinas Clientes

### Máquina Cliente 1: Servidor Web Apache

Primer paso instalamos Apache en nuestro servidor:

```
sudo apt install apache2 -y
```

Configuraremos nuestro firewall para permitir tráfico HTTP:

```
sudo ufw allow 'Apache'
```

```
sudo ufw reload
```

Instalaremos y habilitaremos Cockpit en nuestro servidor.

```
sudo apt install cockpit -y
```

```
sudo systemctl start cockpit
```

```
sudo systemctl enable cockpit
```

### Máquina Cliente 2: Servidor de Bases de Datos MySQL

Instalaremos Mysql Server en nuestro máquina:

```
sudo apt install mysql-server -y
```

Configuramos Mysql:

```
sudo mysql_secure_installation
```

Configuramos el firewall para permitir tráfico MySQL:

```
sudo ufw allow 3306/tcp
```

```
sudo ufw reload
```

Instalaremos y habilitaremos Cockpit

```
sudo apt install cockpit -y
```

```
sudo systemctl start cockpit
```

```
sudo systemctl enable cockpit
```

### 4.3 Configurar la monitorización desde nuestro servidor central

Accedemos a la interfaz de Cockpit desde nuestro servidor central:

```
http://<ip-servidor-cockpit>:9090
```

Para agregar las máquinas clientes seguiremos los siguientes pasos:

- Vamos a la sección Dashboard

- Hacemos Click en “Add new Host”
- Ingresaremos la IP de nuestra máquina y proporcionaremos las credenciales de la máquina.

Todos estos pasos los veremos en la demostración práctica añadiendo una máquina adicional

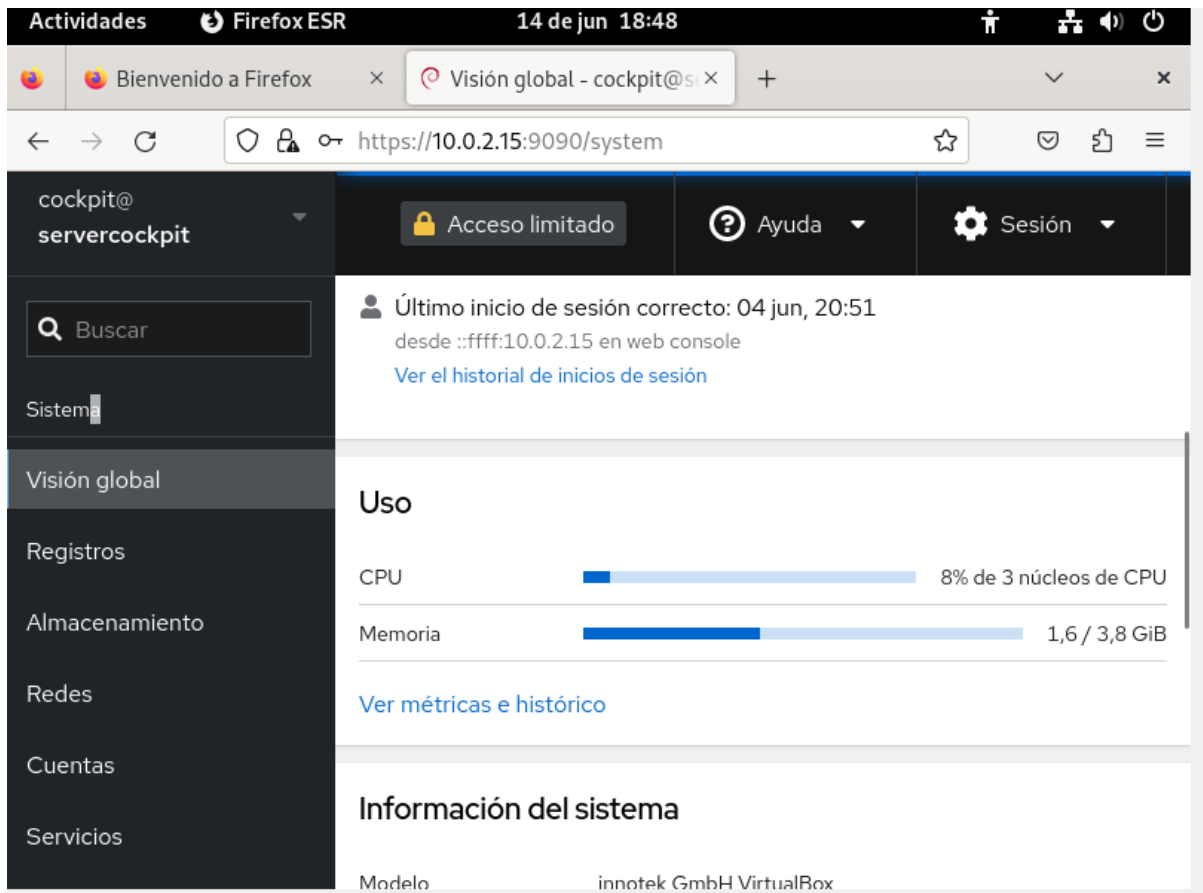
Con esta configuración, el servidor Cockpit podrá monitorizar y administrar las máquinas clientes de manera centralizada. Esto nos permitirá una gestión más eficiente y una supervisión integral del estado y rendimiento de todos los sistemas desde una única interfaz.

#### **4.4 Pruebas de funcionalidades en Cockpit.**

En este apartado veremos varias funcionalidades que tiene Cockpit que después serán demostradas en la demostración práctica.

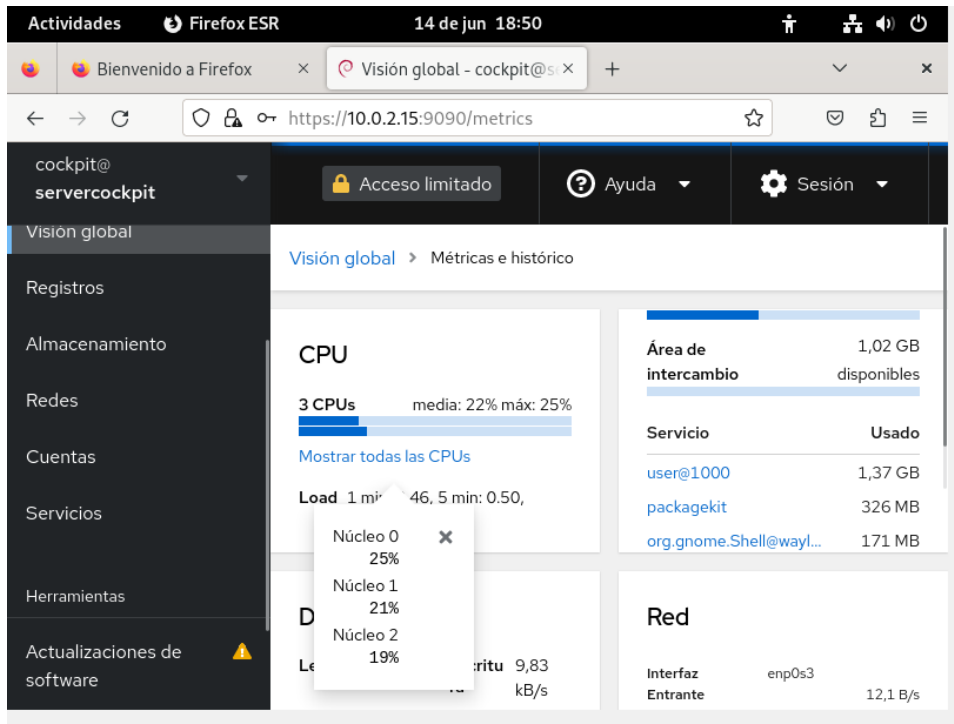
##### **Monitorización del sistema.**

En la interfaz principal de navegaremos a la sección “Dashboard”

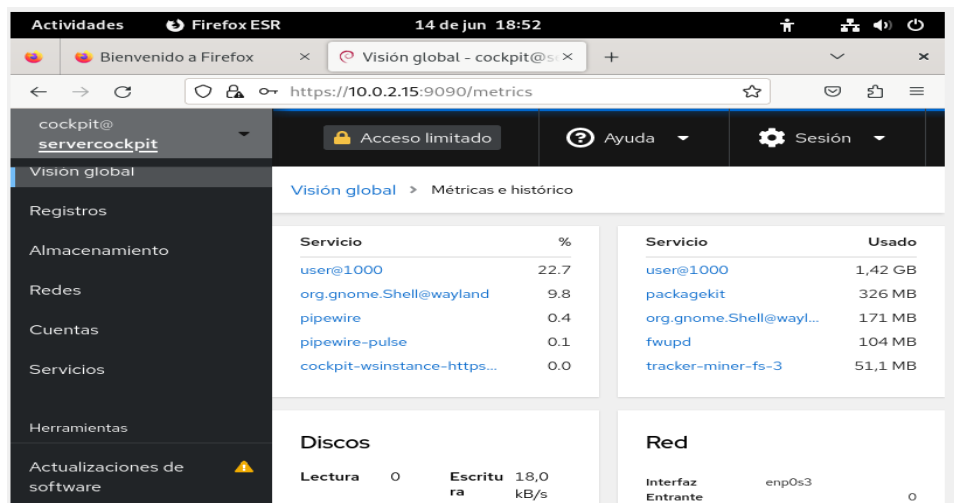


Desde ahí observamos las métricas clave del sistema, incluyendo el uso de CPU, memoria, almacenamiento y actividad de red.

Si hacemos clic en cada métrica podremos observar las métricas claves del sistema. Por ejemplo podemos ver la CPU:

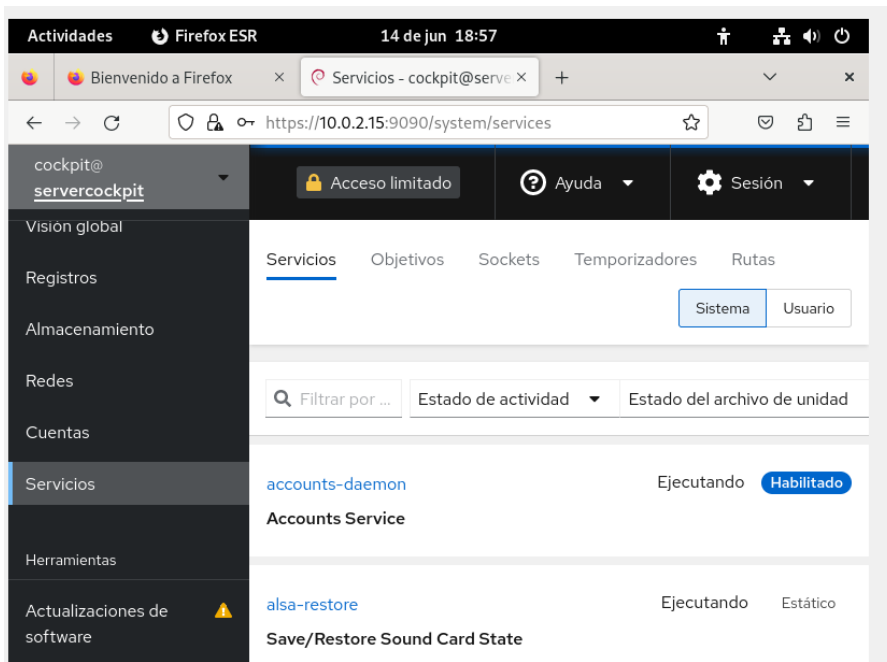


Y los servicios que se están utilizando:

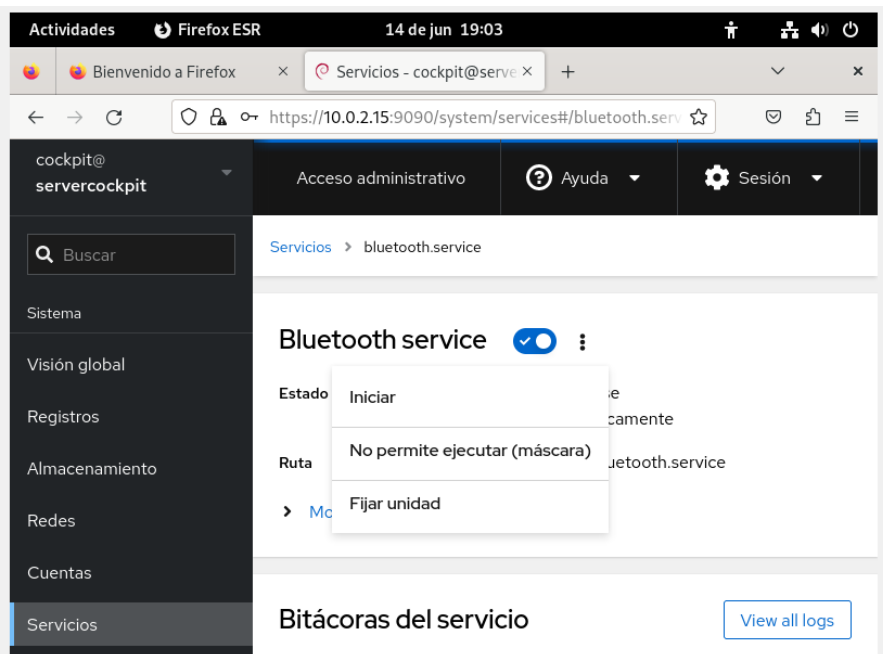


### Prueba de gestión de servicios

En la interfaz principal navegaremos a la sección “Services”:



Buscamos un servicio que queramos gestionar y clicamos en el:

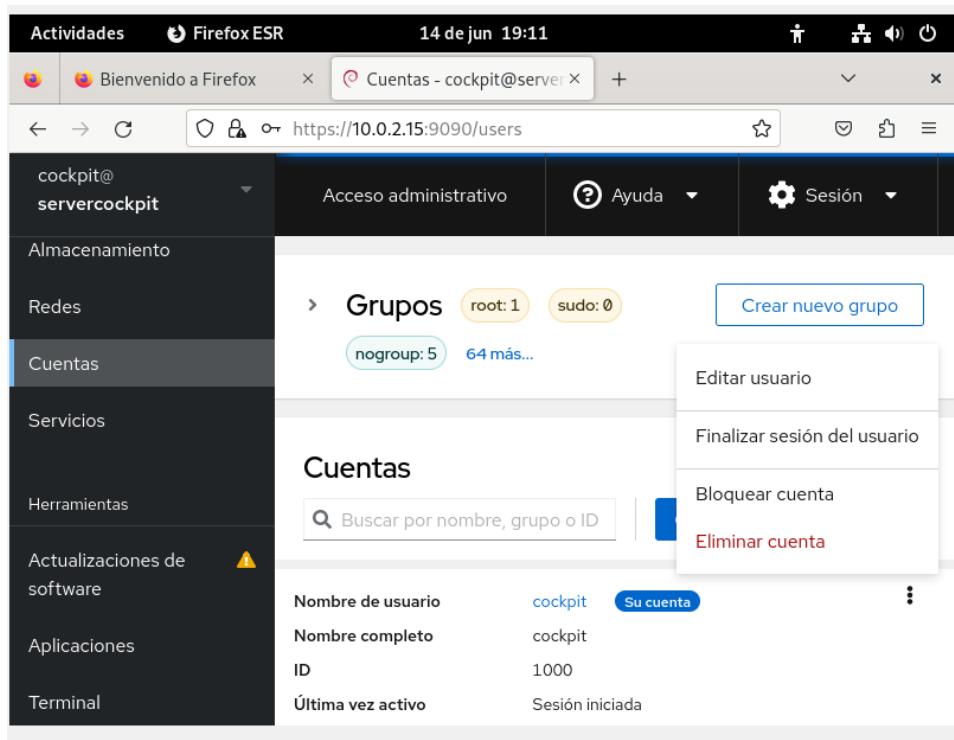


Podremos iniciar/detener/reiniciar el servicio desde nuestra interfaz.



## Prueba de gestión de usuarios

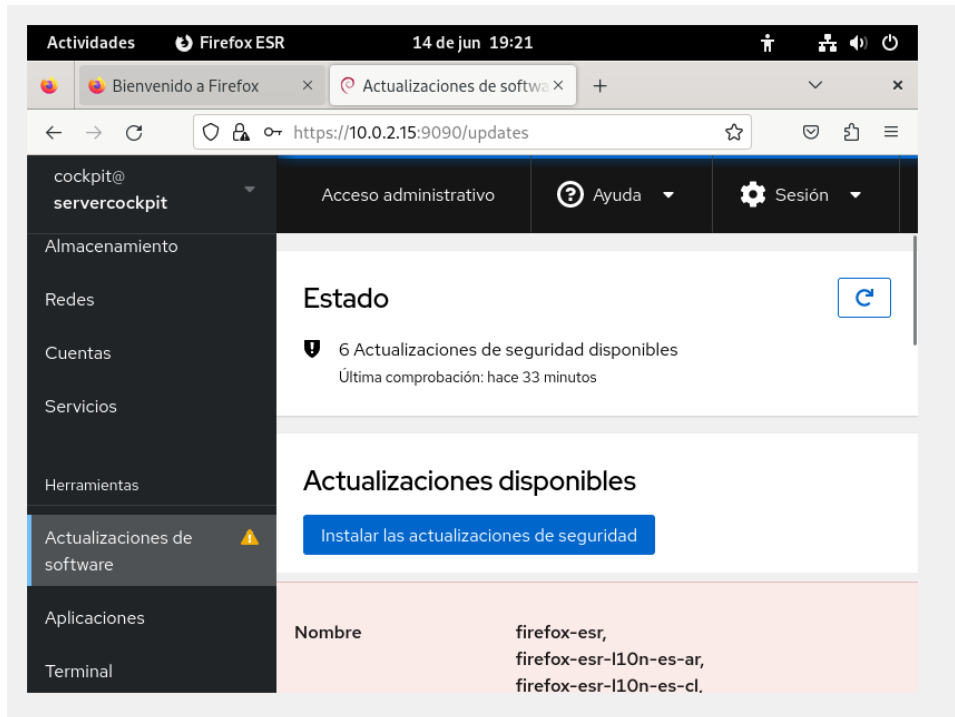
Navegamos a la sección de “Accounts”:



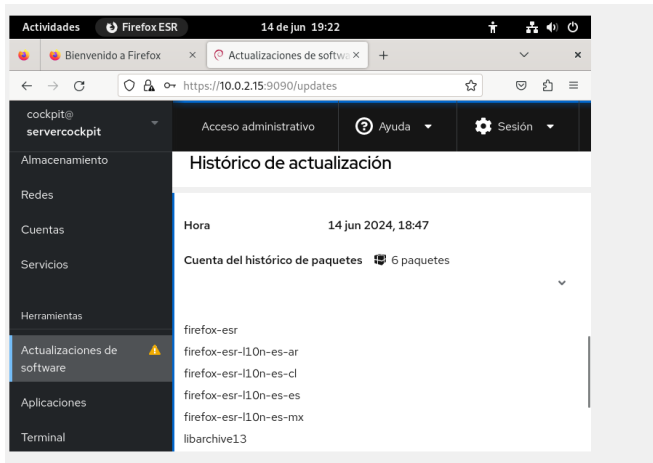
Desde esta sección podremos modificar los usuarios existentes y gestionar sus permisos

## Prueba de actualización del sistema

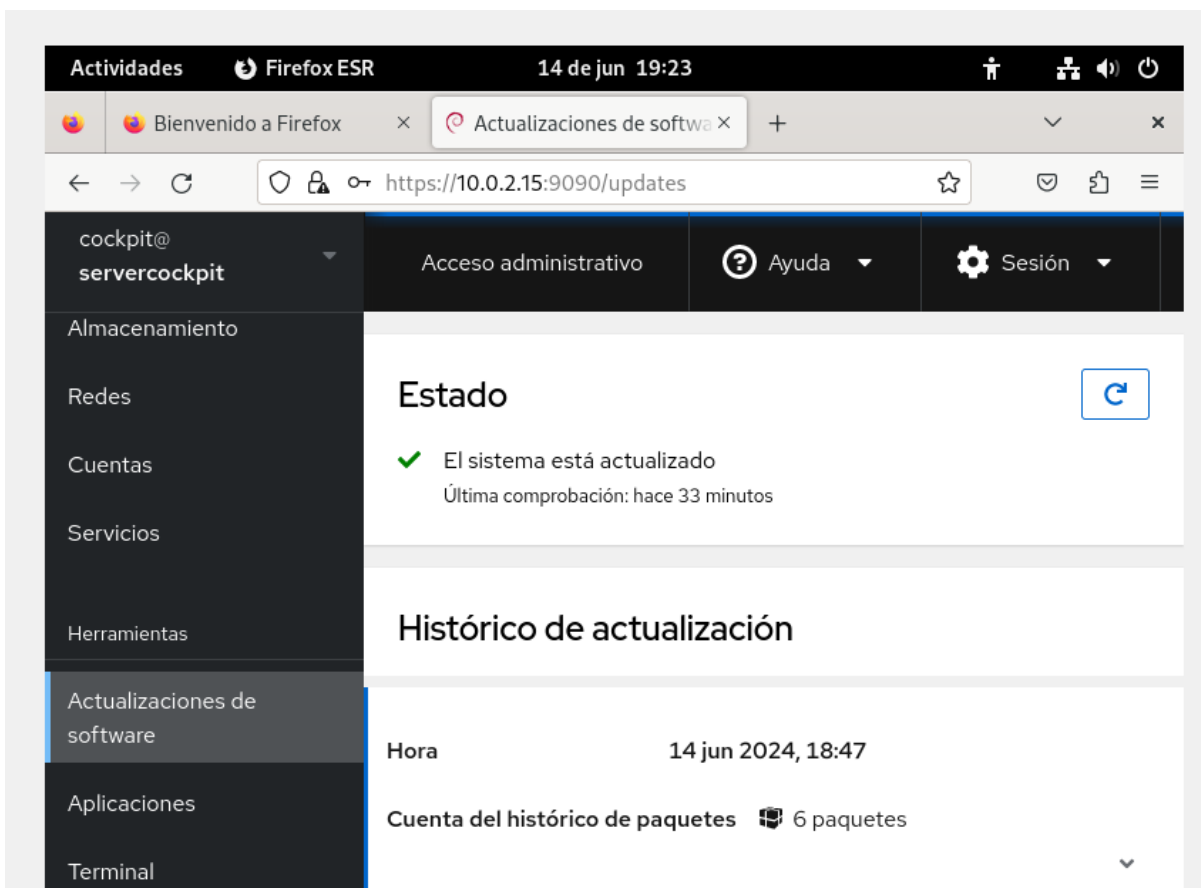
En la interfaz de Cockpit navegaremos a la sección “Software Updates”:



Si tenemos actualizaciones disponibles como en nuestro caso podremos realizarla desde nuestra interfaz. Además, podemos ver la lista de paquetes que se van actualizar:



Una vez realizado tendremos lo siguiente:



Estas pruebas demuestran la capacidad de Cockpit para monitorizar el estado del sistema, gestionar servicios, administrar usuarios y actualizaciones de manera efectiva. Estas funcionalidades son esenciales para la administración eficiente y centralizada de servidores.



## 5. Conclusiones y propuestas para seguir trabajando sobre el tema

### Conclusiones

El proyecto de implementación y monitorización de máquinas virtuales utilizando Cockpit ha demostrado ser una solución efectiva y eficiente para la administración centralizada de sistemas. A lo largo del proceso, hemos podido comprobar varias ventajas significativas de esta herramienta:

- **Facilidad de Uso:** Cockpit ofrece una interfaz gráfica intuitiva que simplifica la gestión y monitorización de los servidores. La posibilidad de realizar tareas administrativas a través de un navegador web reduce la complejidad y el tiempo necesario para la administración del sistema.
- **Monitorización en Tiempo Real:** Cockpit permite la supervisión en tiempo real de los recursos del sistema, incluyendo el uso de CPU, memoria, almacenamiento y actividad de red. Esto facilita la identificación y resolución rápida de problemas, mejorando la disponibilidad y el rendimiento del sistema.
- **Gestión Centralizada de Servicios:** La capacidad de iniciar, detener y reiniciar servicios desde una única interfaz centralizada mejora la eficiencia operativa. Además, la integración con sistemas como Apache y MySQL demuestra la versatilidad de Cockpit para gestionar diversos servicios y aplicaciones.
- **Actualización Simplificada del Sistema:** La funcionalidad de gestión de actualizaciones de Cockpit permite mantener el sistema seguro y actualizado con facilidad. La instalación de actualizaciones y parches es crucial para prevenir vulnerabilidades de seguridad y asegurar el funcionamiento óptimo del sistema.

### Propuestas para seguir trabajando en el tema

Para maximizar los beneficios de Cockpit y continuar mejorando la administración y monitorización de sistemas, se proponen las siguientes líneas de trabajo:

- **Integración con Herramientas de Gestión y Automatización:**

Explorar la integración de Cockpit con herramientas de automatización como Ansible, Puppet o Chef. Esto permitiría una gestión de la configuración y automatización de tareas más avanzada, reduciendo el esfuerzo manual y minimizando errores.

- **Desarrollo de Scripts Personalizados:**

Crear y desplegar scripts personalizados que amplíen las funcionalidades de Cockpit. Por ejemplo, scripts para la gestión avanzada de usuarios, copias de seguridad automatizadas o monitoreo específico de aplicaciones críticas.

- **Implementación de Seguridad Avanzada:**

Investigar e implementar medidas de seguridad adicionales, como autenticación de dos factores (2FA) para acceder a Cockpit y cifrado avanzado para la comunicación entre el servidor y las máquinas clientes. Esto fortalecería la seguridad del entorno administrado.

- **Monitorización y Alertas Proactivas:**

Configurar alertas y notificaciones proactivas basadas en umbrales de uso de recursos. Integrar estas alertas con sistemas de mensajería y notificación (por ejemplo, correo electrónico, Slack) para garantizar que los administradores sean notificados inmediatamente de cualquier problema crítico.

- **Escalabilidad y Desempeño:**

Realizar pruebas de estrés y escalabilidad para evaluar el desempeño de Cockpit en entornos con un gran número de máquinas virtuales. Basado en los resultados, ajustar la configuración para optimizar el rendimiento y asegurar la capacidad de gestionar cargas de trabajo más grandes y complejas.

- **Formación y Capacitación:**

Desarrollar programas de capacitación para el equipo de administración de sistemas, centrados en el uso de Cockpit y las mejores prácticas para la monitorización y gestión de servidores. Asegurar que todo el equipo esté al tanto de las funcionalidades y ventajas de Cockpit maximiza su uso efectivo.

- **Documentación y Soporte:**

Crear documentación detallada y específica para el entorno y las necesidades de la organización. Esto incluye guías de usuario, procedimientos estándar operativos (SOP), y soluciones de problemas comunes. Una buena documentación asegura que el conocimiento se comparta y se mantenga dentro de la organización.

## 6. Bibliografía

### Documentación oficial de Cockpit:

Cockpit Project. (2023). Cockpit Documentation. Recuperado de <https://cockpit-project.org/guide/latest/>

### Instalación y configuración de Cockpit en Debian 12:

Debian Wiki. (2023). Cockpit - Debian Wiki. Recuperado de <https://wiki.debian.org/Cockpit>

DigitalOcean. (2023). How To Install and Use Cockpit on Debian 10. Recuperado de <https://www.digitalocean.com/community/tutorials/how-to-install-and-use-cockpit-on-debian-10>

### Gestión de servicio y monitorización:

Red Hat Customer Portal. (2023). Using Cockpit. Recuperado de [https://access.redhat.com/documentation/en-us/red\\_hat\\_enterprise\\_linux/7/html/using\\_cockpit/index](https://access.redhat.com/documentation/en-us/red_hat_enterprise_linux/7/html/using_cockpit/index)

Linux.com. (2023). Monitor and Administer Linux Servers with Cockpit. Recuperado de <https://www.linux.com/training-tutorials/monitor-and-administer-linux-servers-cockpit/>

TechRepublic. (2023). How to Use Cockpit to Monitor and Administer Your Data Center Servers. Recuperado de <https://www.techrepublic.com/article/how-to-use-cockpit-to-monitor-and-administer-your-data-center-servers/>

### Artículos y Tutoriales Relacionados

Linux Journal. (2023). Simplifying Server Management with Cockpit. Recuperado de <https://www.linuxjournal.com/content/simplifying-server-management-cockpit>

TechTarget. (2023). Cockpit for Linux: An Admin's Guide. Recuperado de <https://www.techtarget.com/searchdatacenter/tip/Cockpit-for-Linux-An-admins-guide>

DZone. (2023). Getting Started with Cockpit for Linux Server Management. Recuperado de <https://dzone.com/articles/getting-started-with-cockpit-for-linux-server-management>