Proyecto Fin de Carrera

TRANSFORMACIÓN DE UNA INFRAESTRUCTURA TECNOLÓGICA PARA UN CRECIMIENTO SOSTENIBLE

(Transforming IT Infrastructure to enable sustainable growth)

Para acceder al Título de

INGENIERO EN INFORMÁTICA

Autor: Antonio Albendea Herrera

Septiembre - 2012

Facultad de Ciencias

INGENIERÍA EN INFORMÁTICA

CALIFICACIÓN DEL PROYECTO FIN DE CARRERA

Realizado por: Antonio Albendea Herrera
Director del PFC: Jose Luis Bosque Orero
Título: "Transformación de una infraestructura tecnológica para un crecimiento sostenible"
Title: "Transforming IT Infrastructure to enable sustainable growth "

Presentado a examen el día:

para acceder al Título de

INGENIERO EN INFORMÁTICA

Composición del Tribunal:
Presidente (Apellidos, Nombre): Michael González Harbour
Secretario (Apellidos, Nombre): Mª del Carmen Martínez Fernández
Vocal (Apellidos, Nombre): Rafael Menéndez de Llano Rozas
Vocal (Apellidos, Nombre): Pablo Sánchez Barreiro
Vocal (Apellidos, Nombre): Roberto Sanz Gil

Este Tribunal ha resuelto otorgar la calificación de: .....

Fdo.: El Presidente

Fdo.: El Secretario

Fdo.: Vocal

Fdo.: Vocal

Fdo.: Vocal

Fdo.: El Director del PFC

Agradecimientos

A lo largo de estas palabras quisiera agradecer, esperando no olvidarme de nadie, a todas aquellas personas que me han apoyado durante estos años en los que he peleado por convertirme en Ingeniero Informático, siendo quizás por edad, la última oportunidad que podía tener.

En primer lugar quisiera mencionar a mi familia, especialmente a mi madre, hermana y abuelos por haber sido mi principal fuente de ánimos y un apoyo fundamental sin el cual, seguramente, no habría alcanzado este sueño. También a mi padre, por no haberme presionado nunca y haberme prestado su ayuda en los momentos que lo he necesitado. Y claro está, a mi pareja Beatriz, lo eres todo para mí y este sueño sería incompleto y carecería de todo sentido sino estuvieras en mi vida. También mencionar a mis tíos: César, Lucía, Antonio, Merche… gracias por estar ahí.

En segundo lugar, quisiera agradecer además a mi tía Mamen, por haberme acogido en tu casa durante estos años para tantas horas de estudio y trabajo y haberme tratado como a un hijo, y a mi tío Miguel, porque fuiste de todos los miembros de mi familia, quien mostró la mayor ilusión y alegría por ir a la universidad sin dudar nunca de mí. Espero que estés orgulloso de mí allá donde estés.

Gracias, también, a mi director de proyecto, José Luis Bosque, por haberte ofrecido en todo momento a coordinar mi proyecto sabiendo que era un riesgo al desarrollarlo en el extranjero, así como por todo tu apoyo y cariño durante estos años en los que has abierto las puertas de tu despacho hacia mi persona para cualquier duda o apoyo.

Tampoco puedo olvidarme de la empresa alemana que me abrió sus puertas, Celonis GmbH, que me ha permitido no sólo desarrollar este proyecto fin de carrera sino también haberme formado como persona y profesional dentro del apasionante mundo de la informática. No podré olvidaros, Bastian, Martin y Alexander por confiar en mí, sabiendo que no hablaba alemán y que mi nivel de inglés era muy bajo. Gracias a vuestra oportunidad, ahora sí que puedo decir que hablo inglés. Por supuesto al resto del equipo, me he sentido uno más en la empresa.

Gracias a mis amigos por escucharme y animarme siempre.

Por último a mis compañeros de la carrera y por supuesto amigos, por los momentos compartidos y la ayuda prestada.

INDICE GENERAL

CAPÍTULO 1
INTRODUCCIÓN

Este capítulo sirve como presentación general del trabajo a desarrollar durante el proyecto, y por tanto, en él se describe la situación tecnológica, informática y de red actual de la empresa así como sus necesidades de presente y futuro que marcarán los objetivos del mismo.

1.1 Introducción

Celonis GmbH es una empresa alemana, con sede en Múnich, desarrolladora de software a medida para la gestión de organizaciones. Mediante este software denominado *ORCHESTRA* se pretende mejorar la gestión activa y el control de procesos empresariales permitiendo una reducción de costes así como una mejora en la calidad productiva.

El gran objetivo de este proyecto consistirá en transformar la infraestructura informática y de red existente a una nueva. Para ello se aprovecharán los recursos disponibles en la empresa, así como el despliegue de un conjunto de servicios, de índole interna y externa, con los que la compañía pueda desarrollar su trabajo de la manera más satisfactoria posible. Esto permitirá garantizar a sus clientes la calidad del servicio ofertado puesto que se mejorará el rendimiento a la hora de procesar un gran volumen de datos a través de un software propietario.

Para poder desarrollar este objetivo uno de los conceptos claves que se van a aplicar será el de *virtualización* [1]. Su función principal es abstraer los recursos hardware de un computador para poder constituir grandes entornos de ejecución. En palabras más simples, virtualizar es permitir, dentro de un computador físico, la ejecución concurrente de una o más computadoras lógicas, con sus propios sistemas operativos y aplicaciones, así como con una asignación de recursos establecida.

La táctica de virtualización presenta grandes ventajas (sobre todo en ahorro de costes) pero requiere de un estudio previo con diferentes técnicas de virtualización y con diferentes paquetes software que permitan obtener el mayor rendimiento posible de los sistemas virtualizados. Sin embargo, este análisis comparativo de diversas técnicas de virtualización para el caso concreto de este proyecto no ha sido realizado, ya que el énfasis del mismo se ha puesto en la minimización del coste. Es por ello, que se ha desarrollado bajo la plataforma máquina virtual del kernel (*KVM, Kernel Virtual Machine*) [2], por tratarse de un software libre, de código abierto (*opensource*), y por tanto, sin coste. Aun así, este software opensource permitirá cubrir todas las necesidades de la empresa. Éstas consisten fundamentalmente en la implantación, configuración y despliegue de servicios informáticos virtuales a nivel interno (para uso por parte del personal propio) y externo, es decir, para los clientes. Los servicios requeridos a implementar en una infraestructura virtual bajo KVM incluirán controladores de dominio, servidores de correo, servidores de impresión y archivos, servicios de gestión de tickets (*OTRS, Open Technology Real Services*), servidores de bases de datos (*SQL, Structured Query Language*), ORCHESTA, así como otros servicios adicionales requeridos bajo demanda.

El despliegue de servicios obligará a implementar políticas de seguridad en la infraestructura virtual y de red. Parte de los servicios serán accedidos, como se mencionaba anteriormente, tanto por los miembros de Celonis GmbH de forma remota como por los clientes. Esto se logrará mediante la configuración de un mecanismo de acceso denominado red privada virtual (*VPN, Virtual Private Network*), el cual implica a su vez, la configuración de mecanismos de seguridad para el acceso, mediante filtrado y redirección de paquetes y servicios (*NETFILTER/IPTABLES*), así como conexiones entrantes y salientes seguras (*SSL, Secure Socket Layer)*.

Será fundamental a su vez, desplegar una política de respaldo de los servicios desarrollados, sobre todo en aquéllos en producción (servicios de pago contratados y accedidos por clientes). Se crearán políticas de copias de seguridad y respaldo mediante mecanismos de *backups*.

1.2 Objetivos

Teniendo en cuenta el escenario descrito anteriormente, los objetivos a cumplir en el desarrollo de este proyecto son los siguientes:

1º.- Reestructuración de la red informática, preparándola para un crecimiento sostenible, tanto de servicios como de maquinas físicas y virtuales. Esta reestructuración incluirá la división de la red en diferentes subredes, la habilitación de puntos de acceso inalámbricos y la eliminación de todo cableado superfluo.

2º.- Despliegue y gestión de un entorno de virtualización bajo KVM como centro de testeo y producción. El entorno se constituirá mediante un conjunto de servidores en modo clúster puesto que facilita la portabilidad de servicios entre los servidores.

3º.- Despliegue de dominios virtuales y gestión del directorio activo. Los empleados accederán a los recursos de la empresa validándose en el controlador de dominio, lo cual proporciona un primer mecanismo de seguridad de acceso.

4º.- Despliegue de servicios virtuales tales como servidores de impresión como política de acceso a las impresoras de la empresa, servidores de correo para el envió y recepción de correo electrónico, servidores OTRS para la recepción de incidencias, servidores SQL para el desarrollo de software corporativo, etc.

5º.- Despliegue y configuración de VPN (compatible tanto con sistemas operativos Linux como Windows) para el acceso validado desde el exterior a los recursos internos de la empresa.

7º.- Seguridad y automatización de backups de servidores virtuales como política de respaldo a los servicios en producción así como para el cumplimiento de la ley orgánica de protección de datos en Alemania.

8º.- Desarrollo de una infraestructura de software como servicio (*SaaS, Software as a Service*) para que los clientes de la empresa pueden acceder de forma segura a los servicios contratados.

1.3 Descripción de este documento

La memoria de este trabajo fin de carrera está estructurada en diferentes capítulos, tal y como se pasa a describir en las siguientes líneas.

En el segundo capítulo se describe en primer lugar la situación de la red informática perteneciente a Celonis GmbH previa a la adaptación al futuro entorno virtual. Además, se recogen las necesidades a cumplir planteadas por la empresa así como el material disponible para el desarrollo del proyecto. Se especifican, a su vez, los cambios realizados así como una explicación del presente y futuro de la red informática.

La implementación sobre una máquina física de: el cortafuegos *(firewall)* de la empresa, los servicios de resolución de nombres, el protocolo de configuración dinámica del host (*DHCP,* Dynamic Host Configuration Protocol) y el servicio VPN se describen en el tercer capítulo. Dicha máquina se encargará por tanto de proteger el entorno virtual a implementar y las diferentes subredes que conforman la red de la empresa, así como de facilitar la resolución de nombres de equipos y la asignación dinámica de las direcciones de red. Esta descripción incluirá una introducción sobre los requisitos a cumplir impuestos por la empresa, los servicios requeridos, el software a utilizar junto con su configuración básica así como el motivo de uso. Asimismo se describirá el concepto de *VPN* para accesos externos a la red empresarial mediante un mecanismo de seguridad denominado SSL.

En el cuarto capítulo, se explica detalladamente el concepto de virtualización así como sus ventajas y desventajas. Además, se hace un análisis teórico de las diferentes técnicas de virtualización posibles puesto que, como se mencionaba en la Introducción, KVM será la plataforma seleccionada por motivos económicos. Según esto, se derivará a un análisis del software virtualizador capaz de alcanzar el objetivo deseado así como la configuración a

establecer para ello. Por último, se describirán los servicios desplegados en la plataforma virtual así como del sistema de backup empleado.

Debido al surgimiento a lo largo del desarrollo del proyecto de un nuevo software opensource, compatible con KVM, denominado Proxmox el cual incluye funcionalidad de backup en una interfaz gráfica mejorada, el capítulo quinto discute las mejoras que dicho software proporcionaría a la empresa en comparación con lo desarrollado en los capítulos anteriores del proyecto. Además se considera la posibilidad de portar servicios virtuales a la nube (*cloud*) tomando como referencia la nube de la compañía Amazon.

En el sexto capítulo se recogen las conclusiones finales del proyecto y se analiza el alcance en el cumplimiento de los objetivos inicialmente planteados.

Finalmente, se recoge la bibliografía usada para el desarrollo del proyecto de forma íntegra.

CAPÍTULO 2
DISEÑO Y DESPLIEGUE DE LA RED INFORMÁTICA

En este capítulo se describe la situación inicial de la red de la empresa, esto es previa al inicio del desarrollo de este proyecto, así como sus características físicas y de configuración. A continuación, se especificarán los cambios introducidos así como las motivaciones de los mismos.

2.1 Antecedentes de la red informática

La red informática de la compañía Celonis GmbH tiene contratada una línea dedicada de acceso a internet. El proveedor de servicios, Cable Deutschland, provee de acceso a internet a los miembros de la compañía, alcanzándose los siguientes requisitos técnicos contratados:

- **Velocidad de descarga: 100 Mbps**
- **Velocidad de subida: 6 Mbps**

Celonis GmbH se ubica en una cuarta planta donde posee en régimen de alquiler 5 oficinas para el desarrollo de su actividad empresarial. Cada oficina dispone de al menos 6 tomas de corriente y 4 conectores de red/teléfono con su correspondiente numeración situados en el suelo.

- **Oficina 4.01: conectores de red/teléfono B19 – B24**
- **Oficina 4.02: conectores de red/teléfono C1-C4**
- **Oficina 4.03: conectores de red/teléfono C5-C16**
- **Oficina 4.50: conectores de red/teléfono B1-B12**
- **Oficina 4.51: conectores de red/teléfono B13-B18**

Asimismo, se dispone de un cuarto de comunicaciones común al edificio situado en la tercera planta, donde se facilita el acceso a cada proveedor ISP contratado por las diferentes compañías presentes en el mismo.

El cable módem conecta directamente al puerto del patchpannel con numeración B20 mediante un cable de red RJ-45. Este puerto tiene su correspondiente conexión mediante otro cable de red RJ-45 hacia un router inalámbrico con servicio DHCP encargado de asignar dirección IP a los equipos de la empresa, mediante conexión tanto cableada como inalámbrica.

En la Figura 1 se esquematiza la situación inicial de la red informática de la empresa. El router se ubica en la oficina 4.01 dando acceso inalámbrico al resto de las oficinas. Es importante mencionar el caso especial de la oficina 4.50, que dispone de un switch que le conecta de forma cableada con el router presente en la habitación 4.01. Esta configuración deficiente se debe a que la red inalámbrica, al parecer, perdía conectividad frecuentemente y los usuarios ubicación en la habitación 4.50 requerían de conexión permanente para el desempeño de sus labores. Además, también se puede observar cómo, a excepción de la habitación 4.01, ninguna habitación usaba para el acceso a internet los cajetines de red ubicados en el suelo de cada oficina. Esto fue motivado básicamente por decisión de los directores de Celonis GmbH, una decisión que no tuvo en cuenta las constantes caídas de la red inalámbrica. Se debe puntualizar, que la empresa es de reciente creación por lo que el objetivo a

corto plazo en este aspecto fue el de reducir tiempo de implementación y fundamentalmente el coste.

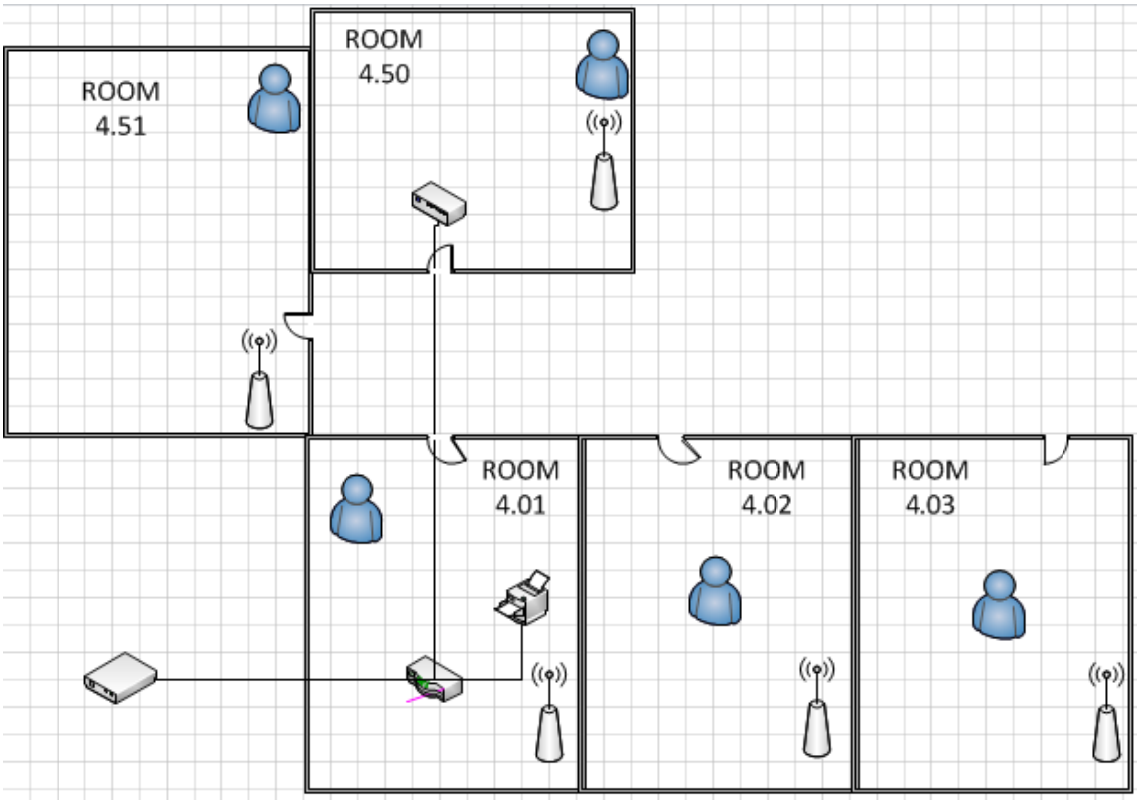

Figura 1. Esquema de la red original de la empresa Celonis GmbH.

2.2 Requisitos de la empresa a desarrollar en el proyecto

La empresa Celonis GmbH ha planificado una reconfiguración de la red actual para adaptarse a una nueva situación de cambios, ya que durante los meses posteriores a su creación, ha experimentado un crecimiento notable tanto en personal como de volumen de negocios. Por lo tanto, el primer objetivo de este Proyecto Fin de Carrera será la reconfiguración de la red descrita según una serie de requisitos de la empresa que se describen a continuación.

Debido a las necesidades ya comentadas, Celonis GmbH deseaba que se configurase un sistema de comunicaciones caracterizado por tres segmentos de red diferenciados, además de el segmento virtual que define la red privada, tal y como se muestra en la Tabla 1.

Tabla 1. Sub-redes de Celonis GmbH

Nombre de Red	Red / Máscara	Nombre DNS	Objetivo
Internal	192.168.1.0 / 24	Internal.celonis.de	Uso interno – Cable
Celonis-int	192.168.10.0 / 24	Wifiinternal.celonis.de	Uso interno – WIFI
Celonis-ext	192.168.20.0 / 24	Wifiexternal.celonis.de	Clientes – Wifi
VPN	10.0.0.0 / 24		Uso interno – VPN

Otro requisito impuesto por la empresa, es habilitar, al menos, una toma de red de los cajetines en cada habitación, con el objetivo de conectar un switch por cada una de ellas para facilitar conexión tanto cableada como inalámbrica.

Por último, la empresa desea utilizar su viejo firewall, con sistema operativo Ubuntu Server 10.04, el cual se encontraba hasta ese momento fuera de servicio, para establecer un primer mecanismo de seguridad, donde implementar las siguientes medidas:

- Se debe permitir el tráfico entre la red Internal y Celonis-int.
- Se debe denegar el tráfico entre la red Celonis-ext e Internal.
- Se debe denegar el tráfico entre la red Celonis-ext y Celonis-int.
- Se debe permitir el paso desde el exterior a usuarios autorizados para conectarse a los servicios internos de la empresa.

Para el despliegue de la nueva infraestructura de red, se dispone del siguiente material:

- Cable módem.
- Router inalámbrico modelo D-Link 4 puertos.
- Router inalámbrico modelo LinkSys – Cisco System 4 puertos.
- Punto de acceso inalámbrico TP-Link.
- Switch model TP-Link 7 puertos.
- Cables de red.
- Firewall con 4 interfaces de red: ETH0, ETH1, ETH2 y ETH3.

2.3 Adaptación de la red a las necesidades de la empresa

Uno de los primeros pasos que se debían precisar, era establecer la forma en que se podía habilitar internet en los cajetines de red presentes en cada oficina.

El servicio DNS de nuestro ISP se representa a través del router D-link (facilitado por el proveedor de servicio) ubicado físicamente en la oficina 4.03. Este servicio permitirá resolver aquellas direcciones externas a los servidores de la compañía.

Asimismo, este dispositivo implementa el servicio DHCP, que asigna direcciones IP a aquellos computadores que se conecten físicamente por cable.

Una configuración básica, aceptada como solución debido a las pocas necesidades actuales en la compañía y por su bajo coste, será conectar el cable módem (Cable Deutschland) al puerto B20 del armario de comunicaciones (*patchpannel)* en un extremo, mientras que el otro extremo, situado en la

habitación 4.01 mostrada en la Fig. 1, conectaba al router D-Link. Uno de los puertos habilitados del router, conectará con la primer interfaz de red del firewall ETH0.

Desde el resto de los interfaces de red del firewall, se establecerán las siguientes conexiones:

- ETH1: se constituirá la red cableada llamada Internal y se habilitará conectividad en los cajetines de red de cada oficina.
- ETH2: se constituirá la red inalámbrica llamada Celonis-int.
- ETH3: se constituirá la red inalámbrica llamada Celonis-ext.

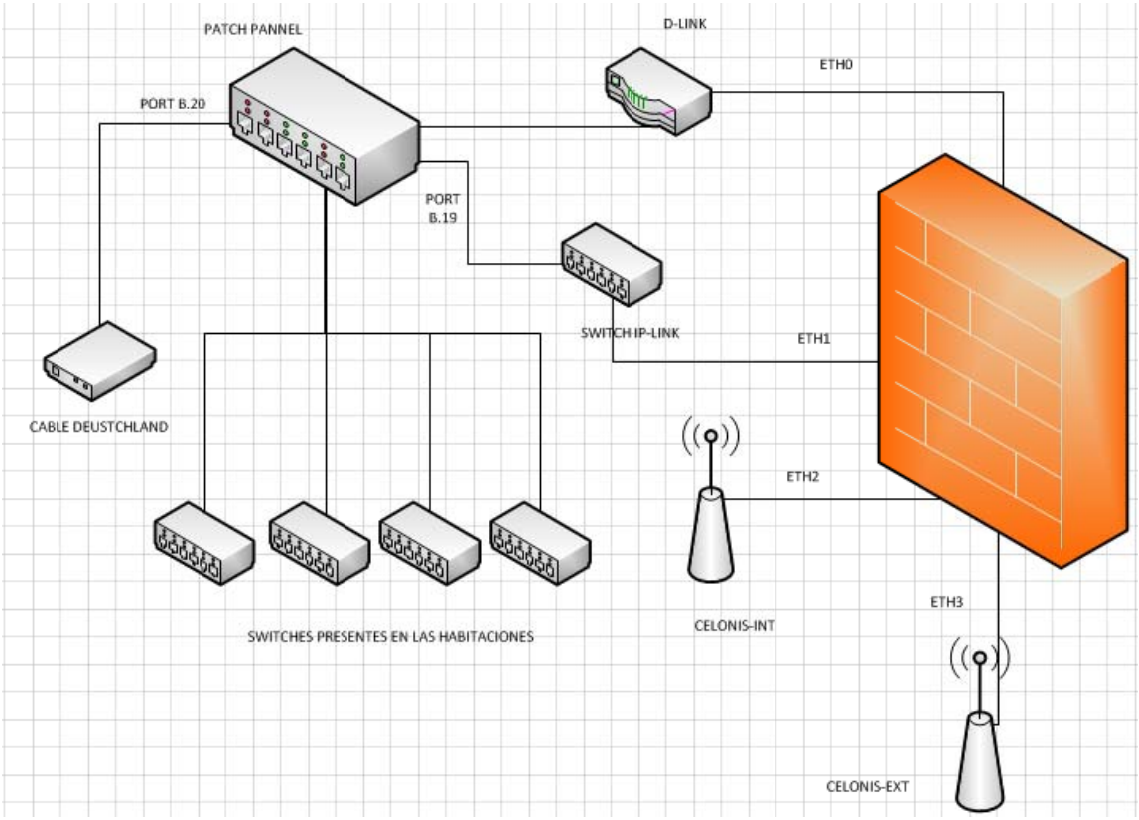

Figura 2. Red Modificada.

En la Figura 2 se puede comprobar el estado final de la red Celonis GmbH. Se puede observar que la interfaz **eth0** es la responsable de comunicar la red interna de la empresa (tanto cableada como inalámbrica) con el mundo exterior y viceversa. Para ello, **eth0** recibirá el tráfico exterior a través del router D-Link y dirigirá los diferentes paquetes a las correspondientes segmentos de red. Por otra parte, esta misma interfaz recibirá los paquetes de las diferentes subredes de la empresa para redirigirlos hacia internet.

A través de la interfaz **eth1**, se construirá la red Internal, con dirección de red 192.168.1.0/24, accesible mediante cable en cada oficina a través de los

diferentes switches habilitados en cada una de ellas. Mediante la interfaz **eth2** se constituirá la red inalámbrica Celonis-int, cuya dirección de red en este caso es 192.168.10.0/24, para uso interno de los empleados. Y, por último, mediante la interfaz **eth3** se constituirá la red inalámbrica Celonis-ext, con dirección de red 192.168.20.0/24 para uso de clientes y personal externo de la compañía.

Una vez establecido el sistema físico de comunicaciones hay que configurar el software del sistema operativo para que éste funcione correctamente. Será necesario, por tanto, configurar los interfaces de red para obtener conectividad entre las diferentes subredes de la empresa. Para ello, se ha llevado a cabo la edición del fichero **interfaces**, y así especificar la configuración de cada interfaz del firewall. Cada una de ellas es estática (lo cual se especifica mediante el parámetro *inet static*), está definida por una dirección de red (*address*) y una mascará de red (*netmask*). La interfaz eth0 define además la dirección IP del router (*gateway*), ya que será por éste por donde pase todo el tráfico generado por las subredes hacia el exterior y viceversa. La Figura 3 muestra el fichero de configuración construido para satisfacer estos requerimientos.

```
# The loopback network interface
auto lo
iface lo inet loopback

# The external interface, will get an IP from the cable modem.
auto eth0
iface eth0 inet static
        address 192.168.0.254
        netmask 255.255.255.0
        gateway 192.168.0.1

# The internal network
auto eth1
iface eth1 inet static
        address 192.168.1.1
        netmask 255.255.255.0

# The wifi internal network
auto eth2
iface eth2 inet static
        address 192.168.10.1
        netmask 255.255.255.0

# The wifi external network
auto eth3
iface eth3 inet static
        address 192.168.20.1
        netmask 255.255.255.0
```

Figura 3. Configuración de los interfaces de red.

2.4 Presente y futuro de la red informática

Actualmente, se ha configurado la red informática para un crecimiento escalonado, orientada a alojar un número determinado de maquinas virtuales y equipos informáticos, estableciéndose los siguientes parámetros:

Tabla 2. Diseño IP sub-redes.

Red	IP´s Dinámicas	IP´s reservadas	IP´s en uso	% Uso
Internal	100	154	12	4%
Celonis-int	100	154	8	3%
Celonis-ext	100	154	0	0%

Se puede establecer, por lo tanto, que la configuración de la red establecida es válida siempre y cuando el crecimiento se produzca de forma progresiva.

No obstante, será necesario, en el futuro, adaptarse completamente a la infraestructura situada en la planta tercera que, como se mencionaba anteriormente, es donde se encuentra ubicado el armario de comunicaciones. Para ello, se recomienda adaptar el armario de comunicaciones a los estándares ISO establecidos. Adicionalmente, se podrían tener en cuenta los siguientes aspectos:

1. Contratar una centralita RDSI/RTB para habilitar directamente todas las conexiones de red/teléfono de los cajetines de red.
2. Adquirir un switch homologado para incluirlo en el armario de comunicaciones.
3. Contratar un router de funcionalidad extendida para habilitarlo en el armario de comunicaciones.
4. Sistemas de alimentación ininterrumpidos (*SA*) para proteger los servidores de la empresa.
5. Trasladar el firewall al armario de comunicaciones.

CAPÍTULO 3
CONFIGURACIÓN Y DESPLIEGUE DE SERVICIOS EN UN FIREWALL

A lo largo de este capítulo se describe el proceso de configuración de un firewall. Con este último se pretende dotar a la empresa de un mecanismo de seguridad frente a accesos no deseados. Asimismo, se detallarán los servicios adicionales que han sido puestos en marcha en dicho dispositivo.

3.1 Definición de servicios en el firewall

Un firewall es un software o hardware que comprueba la información procedente de internet hacia una red, permitiéndose el paso o bloqueándose el acceso de conexiones, en función de su configuración [3].

Los firewalls ayudan a impedir que piratas informáticos (*hackers*) o software malintencionado obtengan acceso a los dispositivos que conforman la red, evitando un gran agujero de seguridad donde los datos puedes ser expuestos a accesos externos no autorizados.

Los firewalls tienen la capacidad de evitar conexiones a determinados servicios o puertos de los equipos que conforman la red.

3.1.1. Requisitos

Se dispone de un computador VENTO modelo A8 que posee 4 tarjetas de red y el sistema operativo Ubuntu Server de 64 bits. El nombre del equipo será **Wall**. El tipo de firewall, por sus características, es de categoría software (no es un dispositivo específico), lo que permite el despliegue de servicios adicionales en el propio firewall. Se deben implementar los siguientes servicios:

- **Sistema de Nombres de Dominio o "DNS".**

La motivación para desplegar el servicio DNS como servicio del firewall tiene como objetivo poder facilitar la resolución de nombres entre dispositivos de las diferentes subredes que conforman la empresa, así como facilitar las resoluciones externas a la red.

Los miembros de la empresa se conectarán a los servidores a través de sesiones de escritorio remoto mediante el nombre de equipo. Parte de los miembros se conectarán en la red Internal, mientras que el resto se conectarán en la red Celonis-int, por lo que se necesita este servicio para resolver las conexiones por nombre y no por dirección IP.

De la misma forma, miembros que accedan a la red mediante una conexión virtual (usando como canal la VPN de la empresa) se deberán conectar mediante escritorio remoto a los servicios de la empresa.

Al tratarse de un servicio a desplegar en un kernel Linux, tal y como se mencionaba en la descripción del material disponible para el desarrollo del proyecto, se instalará y configurara el servicio **bind9** facilitado por el propio kernel [4,5].

- **Protocolo de Configuración Dinámica de Host o "DHCP".**

La motivación para desplegar el DHCP como servicio reside en facilitar la configuración de red de los equipos que se conecten a las diferentes subredes de la empresa. Se configurará en cada subred un intervalo de 100 direcciones dinámicas, reservando el resto para la red, la dirección de multidifusión (*broadcast*), la dirección del router y para uso interno (servidores con dirección estática).

Al tratarse de un servicio a desplegar en un kernel Linux, se instalará y configurara el servicio **dhcp3-server** [6].

- **Firewall o sistema de enrutador entre subredes.**

La motivación para desplegar el servicio de firewall reside en facilitar el control del tráfico entrante y saliente, así como el tráfico entre las diferentes subredes de la empresa.

Se deben controlar conexiones a puertos y servicios, para evitar conexiones no deseadas y posibles agujeros de seguridad.

Al tratarse de una configuración a desplegar en un kernel Linux, instalara y configurara el servicio **iptables** [7-11].

3.1.2. Servicio de resolución de nombres

En esta sección se explicará con detalle algunos conceptos esenciales relativos al servidor de nombres, que serán de utilidad para entender correctamente el procedimiento de configuración realizado como parte de este PFC.

El servicio de resolución de nombres es una base de datos distribuida y jerárquica que almacena la información necesaria para los nombres de dominio [11-15]. Sus usos principales son:

- Asignación de nombres de dominio a direcciones IP.
- Localización de servidores de correo.

La resolución de nombres permite el acceso a servidores disponibles a través de un nombre, algo más sencillo que acceder mediante dirección IP.

El servicio DNS se compone de tres elementos fundamentales:

- Clientes DNS.

- Servidores DNS.
- Zonas de autoridad

Los clientes DNS son los programas de usuario que generan peticiones de consulta para resolver nombres. Se pregunta por la dirección IP asignada a un nombre. Sirva de ejemplo la resolución de direcciones web mediante los explorares (Internet Explorer, Mozilla Firefox, Google Chrome, etc.).

Los servidores DNS responden las consultas generadas por los clientes DNS, existiendo dos categorías diferenciadas de servidores [16,17]:

- **Servidor maestro:** también denominado **primario**, que obtiene los datos a partir de un fichero alojado en el propio servidor.
- **Servidor esclavo:** o **secundario**, que obtiene los datos a través de un servidor maestro realizando un proceso denominado transferencia de zona.

En este proyecto fin de carrera se configurarán dos servidores DNS como servicio a la empresa y al propio entorno virtual a desplegar. Se configurará un servidor maestro en el firewall y un servidor esclavo en un controlador de dominio virtual.

Las zonas de autoridad permiten al servidor maestro o primario cargar la información de una zona. Cada zona de autoridad abarca al menos un dominio y posiblemente sus sub-dominios, si estos últimos no son delegados a otras zonas de autoridad..

La información de cada zona de autoridad es almacenada de forma local en un fichero en el servidor DNS. Este fichero puede incluir varios tipos de registros [18] mostrándose los principales en la Tabla 3.

Tabla 3. Principales registros de una zona de autoridad.

Tipo de Registro	Descripción
A	Resuelve un nombre a dirección IP v4 de 32 bits.
AAAA	Resuelve un nombre a dirección IP de 64 bits.
CNAME	Registro de nombre canónico para construir alias
MX	Registro de servidor de correo de dominio
PTR	Resuelve una dirección IP v4 de 32 bits a nombre
NS	Registro del servidor de nombres con autoridad en el dominio
SOA	Registro de inicio de autoridad que especifica el servidor DNS primario o maestro que proporcionará la información con autoridad acerca de un dominio de internet, dirección de correo electrónico del administrador, numero de serie del dominio y parámetros para el tipo de zona.

Las zonas que se pueden resolver son:

- **Zonas de Reenvío** devuelven direcciones IP para búsquedas hechas por nombre de dominio completamente cualificado (*FQDN, Fully Qualified Domain Name*).
- **Zonas de Resolución Inversa** devuelven nombres FQDN para búsquedas hechas por dirección IP.

3.1.2.1 Configuración DNS

Para la instalación y configuración del servidor DNS primario de la compañía en Linux server, se necesitará la instalación de los paquetes **bind9 y dnsutils**. Para realizarlo utilizaremos en este caso el comando **apt-get install** en modo consola, ya que no se dispone de interfaz gráfica que requeriría la instalación de paquetes adicionales.

>> apt-get install bind9 dnsutils

El software **bind9** (el cual proviene del acrónimo *Berkeley Internet Name Domain***)** es una implementación del protocolo DNS y provee una implementación libre de los principales componentes del sistema de resolución de nombres de dominio, incluyendo:

- Un servidor de sistema de nombres de dominio.
- Una biblioteca resolutoria de sistemas de nombre de dominio.
- Herramientas para verificar la operación adecuada del servidor DNS.

Una vez realizada la instalación de estos paquetes es necesario configurarlos. Para ello se tienen que modificar una serie de ficheros incluyendo los datos de configuración adecuados. En las siguientes secciones se presentan estos ficheros y se explican las configuraciones que ha sido necesario realizar.

a) Fichero named.conf

Al fichero **named.conf** se le considera el principal archivo de configuración en el que se especifican los principales ficheros que contienen parámetros de configuración del servicio, que deben ser leídos y cargados para el funcionamiento del mismo. En la Figura 4 se muestra la configuración establecida en el fichero **named.conf** para este caso concreto. En dicha figura se puede ver que las zonas a configurar de la empresa se ubicarán en el fichero **named.conf.celonis-zones**. Las zonas por defecto instaladas por el servicio se ubicarán en el archivo de configuración **named.conf.default-zones** mientras que los parámetros de configuración se especificarán en el fichero **named.conf.options.**

```
root@wall:/home/antonio# cat /etc/bind/named.conf
// This is the primary configuration file for the BIND DNS server named.
//
// Please read /usr/share/doc/bind9/README.Debian.gz for information on the
// structure of BIND configuration files in Debian, *BEFORE* you customize
// this configuration file.
//
// If you are just adding zones, please do that in /etc/bind/named.conf.local

include "/etc/bind/named.conf.options";
include "/etc/bind/named.conf.celonis-zones";
include "/etc/bind/named.conf.default-zones";
```

Figura 4. Fichero named.conf configurado.

El fichero **named.conf.options** tiene como objetivo definir dos funcionalidades. La primera de ellas es identificar los routers a los que reenviar resoluciones DNS en caso de que el propio servicio DNS no sea capaz de resolverlas. Estos intermediarios denominados *forwarder,* son las direcciones IP de los routers de la empresa. La segunda de la definición del listado de las direcciones IP de las tarjetas de red del firewall sobre las que el servicio DNS debe escuchar (*listen-on*) solicitudes. En la Figura 5 se puede observar la configuración del fichero **named.conf.options** en el cual se especifica como forwarder la dirección IP privada del router D-Link, ya que toda petición externa a la red debe ser resuelta por nuestro ISP. Idénticamente se especifican los interfaces de red para las escuchas, esto es el análisis de los paquetes que entran y salen por cada interfaz de red.

```
root@wall:/home/antonio# cat /etc/bind/named.conf.options
options {
        directory "/var/cache/bind";

        // If there is a firewall between you and nameservers you want
        // to talk to, you may need to fix the firewall to allow multiple
        // ports to talk.  See http://www.kb.cert.org/vuls/id/800113

        // If your ISP provided one or more IP addresses for stable
        // nameservers, you probably want to use them as forwarders.
        // Uncomment the following block, and insert the addresses replacing
        // the all-0's placeholder.

        forwarders {
                192.168.0.1;
        };

        auth-nxdomain no;    # conform to RFC1035
        #listen-on-v6 { any; };
        listen-on {
                192.168.1.1;
                192.168.10.1;
                192.168.20.1;
                127.0.0.1;
                10.0.0.1;
        };
};
```

Figura 5. Fichero named.conf.options configurado.

De forma más concreta, los interfaces de red sobre los que se escucharán estos paquetes entrantes y salientes son los siguientes:

- ETH0 asociada a la dirección 10.0.0.1 correspondiente a la VPN.
- ETH1 asociada a la dirección 192.168.1.1 correspondiente a Internal.
- ETH2 asociada a la dirección 192.168.10.1 correspondiente a Celonis-int.
- ETH3 asociada a la dirección 192.168.20.1 correspondiente a Celonis-ext.
- 127.0.0.1 define al local host o dirección loopback.

En el fichero **named.conf.celonis-zones** define las diferentes zonas de resolución de la empresa, de forma que cada zona se corresponde con un segmento de red. Cada segmento tendrá definido una zona de resolución directa (de nombre a IP) y otra zona de resolución indirecta (de IP a nombre). En la Figura 6 se puede muestra la configuración del fichero **named.conf.celonis-zones** correspondiente a la resolución indirecta. Se puede deducir, que cada zona corresponde a una interfaz de red o subred operativa en el entorno de la empresa. Al configurarse un servidor maestro, se especificará en cada zona mediante **type máster** para que se resuelva buscando en la base de datos local. El campo **notify no** especifica que no se envían notificaciones al servidor DNS esclavo. Por otra parte el parámetro **file** determina el fichero que almacena los registros de zona, y, por último, la activación del parámetro **allow-update** especifica que el servicio DHCP puede modificar el valor de los registros de zona.

```
zone "1.168.192.in-addr.arpa" {
        type master;
        notify no;
        file "/var/cache/bind/db.192.168.1";
        allow-update { key DHCP_UPDATER;};
};

zone "10.168.192.in-addr.arpa" {
        type master;
        notify no;
        file "/var/cache/bind/db.192.168.10";
        allow-update { key DHCP_UPDATER;};
};

zone "20.168.192.in-addr.arpa" {
        type master;
        notify no;
        file "/var/cache/bind/db.192.168.20";
        allow-update { key DHCP_UPDATER;};
};
```

Figura 6. Fichero named.conf.celonis-zones configurado.

En la Figura 7 se puede observar la otra parte de la configuración del fichero **named.conf.celonis-zones** correspondiente a la resolución directa. A

cada zona se le ha asignado un nombre relacionado a una interfaz de red o subred operativa en el entorno de la empresa.

```
zone "corp.celonis.de" {
        type forward;
        forwarders { 192.168.1.40; };
};

zone "internal.celonis.de" {
        type master;
        file "/var/cache/bind/db.internal.celonis.de";
        allow-update { key DHCP_UPDATER;};
};

zone "wifiinternal.celonis.de" {
        type master;
        file "/var/cache/bind/db.wifiinternal.celonis.de";
        allow-update { key DHCP_UPDATER;};
};

zone "wifiexternal.celonis.de" {
        type master;
        file "/var/cache/bind/db.wifiexternal.celonis.de";
        allow-update { key DHCP_UPDATER;};
};
```

Figura 7. Fichero named.conf.celonis-zones configurado.

b) Fichero de zonas de autoridad

Como se ha especificado anteriormente, cada zona tiene asociado un fichero que contiene la información local en forma de registros. En la Figura 8 se muestra el contenido del fichero de zona **db.192.168.1** cuya función es resolver las direcciones IP de la red 192.168.1 a nombres de red.

```
root@wall:/home/antonio# cat /var/cache/bind/db.192.168.1
$ORIGIN .
$TTL 604800     ; 1 week
1.168.192.in-addr.arpa  IN SOA  localhost. root.localhost. (
                                1187       ; serial
                                604800     ; refresh (1 week)
                                86400      ; retry (1 day)
                                2419200    ; expire (4 weeks)
                                604800     ; minimum (1 week)
                                )
                        NS      ns.
$ORIGIN 1.168.192.in-addr.arpa.
1                       PTR     ns.internal.celonis.de.
$TTL 300        ; 5 minutes
102                     PTR     simhost.internal.celonis.de.
104                     PTR     cel-wrk-02.internal.celonis.de.
107                     PTR     cel-wrk-01.internal.celonis.de.
```

Figura 8. Fichero de zona db.192.168.1 que almacena fundamentalmente registros PTR.

En la Figura 9 se puede observar el contenido del fichero de zona db.internal.celonis.de cuya función es resolver nombres de red a direcciones IP. Estos ficheros establecen una relación entre la zona llamada **internal.celonis.de** y la red **192.168.1.**

```
root@wall:/home/antonio# cat /var/cache/bind/db.internal.celonis.de
$ORIGIN .
$TTL 604800     ; 1 week
internal.celonis.de     IN SOA  localhost. root.localhost. (
                                2869       ; serial
                                604800     ; refresh (1 week)
                                86400      ; retry (1 day)
                                2419200    ; expire (4 weeks)
                                604800     ; minimum (1 week)
                                )
                        NS      ns.internal.celonis.de.
$ORIGIN internal.celonis.de.
$TTL 300        ; 5 minutes
Bettis-PC               A       192.168.10.186
                        TXT     "318afffda668fa5c5b175ccc7f62f335c0"
C610A-IP                A       192.168.1.118
                        TXT     "0009529efa912535057057d3cf5ef18201"
cel-wrk-01              A       192.168.1.107
                        TXT     "0051a0eeda13c7761377639abffef13951"
cel-wrk-02              A       192.168.1.104
                        TXT     "00b36a29116d0355e4c667e19a11b6eadc"
```

Figura 9. Fichero de zona db.192.168.1 que almacena fundamentalmente registros A.

Mediante la configuración que se acaba de describir se pueden resolver nombres y direcciones IP entre los diferentes segmentos de red de la empresa. Sirva de ejemplo, que esta configuración permitirá, entre otras cosas, sesiones de escritorio remoto a maquinas de la red, conectando a través de nombre, de forma sencilla y directa.

3.1.3 Servicio DHCP

Para la instalación y configuración del servidor DHCP de la compañía en Linux server, se necesitará la instalación del paquete **dhcp3-server**. Este servicio bajo Linux permite [19,20]:

- Asignar direcciones y máscaras de red dinámicamente.
- Asignar direcciones DNS y WINS.
- Nombre de equipo y de dominio.
- Puerta de enlace por defecto.
- Servidores de impresión y de tiempo.

El servicio DHCP puede proporcionar configuraciones a los dispositivos mediante dos métodos:

- **MAC address** donde el servicio DHCP identifica una dirección MAC única proporcionándole la misma configuración cada vez que el servicio se lo solicite.
- **Pool de direcciones** donde se define un rango de direcciones IP denominado pool desde donde cada cliente DHCP recoge del servidor una configuración temporal.

Tal como se especificó en los requisitos, se constituirá un pool de 100 direcciones para cada subred. Para ello será necesario configurar el fichero **dhcpd.conf**. En la Figura 10 se puede observar la configuración del fichero **dhcpd.conf** para cada una de las subredes diseñadas anteriormente. A cada subred se le asocia una interfaz de red (parámetro *interface*), un rango IP (parámetro *range*) para la construcción del pool, un nombre, una dirección IP asociada a un servidor de nombres, la dirección IP del router en la red así como la dirección broadcast de la subred.

```
################## Celonis internal work network ##########
subnet 192.168.1.0 netmask 255.255.255.0 {
  interface eth1;
  range 192.168.1.100 192.168.1.200;
  option domain-name "internal.celonis.de";
  option domain-name-servers 192.168.1.1;
  option netbios-name-servers 192.168.1.40;
  option routers 192.168.1.1;
  option broadcast-address 192.168.1.255;
  authoritative;
}

############## Celonis wifi internal network ##########
subnet 192.168.10.0 netmask 255.255.255.0 {
  interface eth2;
  option domain-name "wifiinternal.celonis.de";
  range 192.168.10.100 192.168.10.200;
  option domain-name-servers 192.168.10.1;
  option routers 192.168.10.1;
  option broadcast-address 192.168.10.255;
  authoritative;
}

################ Celonis wifi external network ###########
subnet 192.168.20.0 netmask 255.255.255.0 {
  interface eth3;
  range 192.168.20.100 192.168.20.200;
  option domain-name "wifiexternal.celonis.de";
  option domain-name-servers 192.168.20.1;
  option routers 192.168.20.1;
  option broadcast-address 192.168.20.255;
  authoritative;
}
```

Figura 10. Fichero dhcpd.conf.

Además para cada zona DNS de las presentadas en la sección anterior, se configurará, una entrada en el fichero **dhcpd.conf** para permitir al servicio DHCP actualizar los registros presentes en las zonas de autoridad del servidor. El objetivo es mantener los registros DNS actualizados con las direcciones IP de los equipos en las diferentes subredes. El servicio DHCP puede asignar diferentes direcciones IP a los equipos a lo largo del tiempo, siendo necesario actualizar los registros cada vez que se produzca un cambio de IP. En la Figura 11 se muestra la configuración de una zona de autoridad del servicio DNS actualizada por el servicio DHCP.

```
zone internal.celonis.de. {
  primary 127.0.0.1;
  key DHCP_UPDATER;
}

zone 1.168.192.in-addr.arpa. {
  primary 127.0.0.1;
  key DHCP_UPDATER;
}
```

Figura 11. Actualización de una zona de autoridad.

3.1.4 Introducción a Netfilter/Iptables

Netfilter es la última generación de las utilidades de filtrado de paquetes que implementa un marco (*framework*) de última generación incluido en el kernel de Linux para el filtrado de paquetes denominado **iptables** [21].

Desarrollar un filtrado de paquetes para proteger a la red corporativa de Celonis GmbH permitirá:

- Control del tráfico de red.
- Seguridad de servicios de red.
- Observación del tráfico de red.

No es necesario instalar ningún paquete adicional puesto que ya que viene por defecto en el kernel de Linux Ubuntu Server.

Desplegar la seguridad mediante **iptables** permite al kernel de Linux manejar todos los paquetes del protocolo de control de transmisión de la capa de transporte (*TCP, Transmission Control Protocol*) así como del protocolo de datagrama de usuario (*UDP, User Datagram Protocol*) que circulen a través de las interfaces de red del firewall. Mediante **iptables** se le indicará al kernel qué hacer con cada paquete en base a unas características particulares (origen, destino, puerto, etc.) o mediante su contenido. El objetivo es identificar el paquete y decidir su destino.

3.1.4.1 Iptables: tablas, cadenas y reglas

La **regla** es la orden específica que se le indica al kernel qué hacer cuando localice un paquete con las características indicadas en la propia regla. Una **regla** está contenida dentro de una **cadena** y las cadenas están dentro de las **tablas** [22,23].

El funcionamiento de iptables es sencillo, basta con entender qué ocurre con un paquete que entra por un interfaz de red del firewall y el kernel lo procesa. El paquete atravesará todas las tablas de iptables (pero no todas las cadenas) y se verifica con cada regla encontrada dentro de las cadenas que tengan sentido. Si se localiza un paquete que cumpla las características definidas en una regla se procederá a ejecutar una acción.

Para establecer la seguridad, se necesita diferenciar entre **tablas, cadenas** y **reglas** para decidir **cuándo** (en qué momento) **y dónde** (tabla + cadena) el kernel revisa cada paquete.

Las **tablas** en **iptables** se dividen en varios tipos que se describen a continuación [24,25].

- **Tabla de filtrado de paquetes** (*filter*) donde se encuentran todas las cadenas que puedan contener reglas de filtrado de paquetes (acepten, rechacen o denieguen paquetes). Contiene las siguientes cadenas predefinidas y cualquier paquete se analizará contra ellas.

    - **Cadena de entrada (*INPUT*):** que como su propio nombre indica será atravesada por todos los paquetes destinados al sistema.
    - **Cadena de salida (*OUTPUT*):** denominada así porque es atravesada por todos los paquetes creados por el sistema.
    - **Cadena de redirección (*FORWARD*):** que será atravesada por los paquetes que pasan por el sistema durante el encaminamiento a su destino.

- **Tabla de traducciones de red** (NAT, Network Address Translation), responsable de configurar reglas de reescritura de direcciones o de puertos de los paquetes. El primer paquete de cualquier conexión pasará siempre por esta tabla para determinar si será necesario reescribir todos los paquetes de la conexión. Esta tabla a su vez consta de las siguientes cadenas:

    - **Cadena de preencaminamiento (*PREROUTING*):** a través de la cual pasan los paquetes entrantes antes de consultarse la tabla de encaminamiento local, principalmente para la traducción de las

direcciones de red de destino (*DNAT, Destination Network Address Translation*).

- **Cadena de postencaminamiento (*POSTROUTING*):** permite que los paquetes salientes pasen a través de esta cadena después de haberse tomado la decisión de encaminamiento, principalmente de la traducción de direcciones de red de origen (*SNAT, Source Network Address Translation*).
- **Cadena de salida (*OUTPUT*):** que permite hacer un DNAT limitado en paquetes generados localmente.

- **Tabla de destrozo** (*Mangle*) responsable de ajustar opciones de paquetes como por ejemplo la calidad de servicio. Todos los paquetes se analizarán en esta tabla. Se incluyen todas las opciones de las tablas filter y NAT.

3.1.4.2 Script programado para el firewall mediante Iptables

Para el correcto funcionamiento del firewall será necesaria la programación de un *shellscript*. La función de este código será habilitar las reglas que deben ejecutarse en la red para validar o invalidar paquetes que entren, salgan o circulen entre los diferentes segmentos de red de la empresa.

El primer paso a la hora de programar el script será asociar a cada interfaz de red del firewall un identificador. Estos identificadores, tal y como se muestra en la Figura 12, son cadenas de texto que identificarán a cada segmento de red. A la interfaz de red eth0 se le asocia la cadena de texto EXT, en referencia a que por dicha interfaz entra el tráfico del exterior a la red y viceversa. Asimismo, a cada segmento de la red definido por las interfaces eth1, eth2 y eth3 se le asigna una cadena de texto identificativa con la subred a la que se conectará. A la intefaz de red eth1 se le asignará la cadena de texto INTERNAL asociada al segmento de red internal. Por otro lado al interfaz de red eth2, se le asocia la cadena de texto WIFIINTERNAL, correspondiente al segmento de red Celonis-int. Por último, el interfaz de red eth3 queda asociado a la cadena de texto WIFIEXTERNAL cuyo segmento de red es Celonis-ext3.

Un caso especial que debe mencionarse es el de la cadena de texto VPN, cuya interfaz de red es inexistente (por tratarse de una red virtual). Sin embargo, le asociamos la dirección de red sobre la que va a operar la futura VPN, cuya configuración se explicará con más detalle en el siguiente apartado.

```
# Definition of the interfaces
EXT="eth0"
INTERNAL="eth1"
WIFIINTERNAL="eth2"
WIFIEXTERNAL="eth3"
VPN=10.0.0.0/24
```

Figura 12. Definición de interfaces.

La primera regla a configurar se centra en el servicio Shell segura (*SSH, Secure SHell*), que permite conexiones a equipos remotos a través del puerto número 22. Este servicio permite conectarse remotamente a computadores de la red, mediante usuario y contraseña, para iniciar una sesión de usuario. Se debe definir en el firewall que cualquier conexión al puerto 22, tanto de entrada como de salida, desde o hacia cualquier segmento de red de la compañía, debe ser permitida, a excepción del segmento de red Celonis-ext, donde el servicio SSH debe ser prohibido. La Figura 13 muestra la programación de las regla para cumplir tales objetivos.

```
# Allow SSH from INTERNAL
iptables -A INPUT -p tcp  -i $INTERNAL --dport 22 -m state --state NEW,ESTABLISHED -j ACCEPT
iptables -A OUTPUT -p tcp -o $INTERNAL --sport 22 -m state --state ESTABLISHED -j ACCEPT
# Allow SSH from  WIFIINTERNAL
iptables -A INPUT -p tcp  -i $WIFIINTERNAL --dport 22 -m state --state NEW,ESTABLISHED -j ACCEPT
iptables -A OUTPUT -p tcp -o $WIFIINTERNAL --sport 22 -m state --state ESTABLISHED -j ACCEPT
# Allow SSH from  VPN
iptables -A INPUT -p tcp  -s 10.0.0.0/24 --dport 22 -m state --state NEW,ESTABLISHED -j ACCEPT
iptables -A OUTPUT -p tcp -s 10.0.0.0/24 --sport 22 -m state --state ESTABLISHED -j ACCEPT
# Not Allow SSH from WIFIEXTERNAL
iptables -A INPUT -p tcp  -i $WIFIEXTERNAL --dport 22 -m state --state NEW,ESTABLISHED -j DROP
iptables -A OUTPUT -p tcp -o $WIFIEXTERNAL --sport 22 -m state --state ESTABLISHED -j DROP
```

Figura 13. Definición del servicio de conexión remota.

Un segundo servicio sobre el que se establecerá un mecanismo de seguridad son las resoluciones DNS en la red. En la Figura 14, se definen las reglas que permiten este servicio en todos los segmentos de red de la empresa a excepción del segmento Celonis-ext. Se puede apreciar que las resoluciones DNS dirigidas al puerto 53, puerto por defecto del servicio DNS, son controladas tanto con el protocolo TCP como por el protocolo UDP.

```
# Allow DNS for INTERNAL
iptables -A INPUT -i $INTERNAL -p tcp --dport 53 -j ACCEPT
iptables -A OUTPUT -o $INTERNAL -p tcp --sport 53 -j ACCEPT
iptables -A INPUT -i $INTERNAL -p udp --dport 53 -j ACCEPT
iptables -A OUTPUT -o $INTERNAL -p udp --sport 53 -j ACCEPT
# Allow DNS for WIFIINTERNAL
iptables -A INPUT -i $WIFIINTERNAL -p tcp --dport 53 -j ACCEPT
iptables -A OUTPUT -o $WIFIINTERNAL -p tcp --sport 53 -j ACCEPT
iptables -A INPUT -i $WIFIINTERNAL -p udp --dport 53 -j ACCEPT
iptables -A OUTPUT -o $WIFIINTERNAL -p udp --sport 53 -j ACCEPT
# Allow DNS for VPN
iptables -A INPUT -s 10.0.0.0/24 -p tcp --dport 53 -j ACCEPT
iptables -A OUTPUT -s 10.0.0.0/24 -p tcp --sport 53 -j ACCEPT
iptables -A INPUT -s 10.0.0.0/24 -p udp --dport 53 -j ACCEPT
iptables -A OUTPUT -s 10.0.0.0/24 -p udp --sport 53 -j ACCEPT
# Dont Allow DNS for WIFIEXTERNAL
iptables -A INPUT -i $WIFIEXTERNAL -p tcp --dport 53 -j DROP
iptables -A OUTPUT -o $WIFIEXTERNAL -p tcp --sport 53 -j DROP
iptables -A INPUT -i $WIFIEXTERNAL -p udp --dport 53 -j DROP
iptables -A OUTPUT -o $WIFIEXTERNAL -p udp --sport 53 -j DROP

# Allow DNS requests on external interface - eth0
iptables -A OUTPUT -o $EXT -p udp --dport 53 -m state --state NEW,ESTABLISHED -j ACCEPT
iptables -A INPUT -i $EXT -p udp --sport 53 -m state --state ESTABLISHED -j ACCEPT
```

Figura 14. Definición del servicio DNS.

El tercer servicio a controlar es el generado por el protocolo DHCP que asigna direcciones IP a los equipos de la red. Este servicio opera en los puertos 67 (servidor DHCP) y 68 (clientes DHCP). En la Figura 15 se muestran las reglas generadas para permitir la creación y establecimiento de conexiones del tipo UDP en todos los segmentos de red de la empresa. De esto modo los equipos presentes en cada segmento puedan recibir la configuración dinámica de red.

```
# Allow DHCP requests from all internal networks
iptables -I INPUT -i  $INTERNAL -p udp --sport 68 --dport 67 -m state --state NEW,ESTABLISHED -j ACCEPT
iptables -I OUTPUT -o $INTERNAL -p udp --sport 67 --dport 68 -m state --state ESTABLISHED -j ACCEPT
iptables -I INPUT -i  $WIFIINTERNAL -p udp --sport 68 --dport 67 -m state --state NEW,ESTABLISHED -j ACCEPT
iptables -I OUTPUT -o $WIFIINTERNAL -p udp --sport 67 --dport 68 -m state --state ESTABLISHED -j ACCEPT
iptables -I INPUT -i  $WIFIEXTERNAL -p udp --sport 68 --dport 67 -m state --state NEW,ESTABLISHED -j ACCEPT
iptables -I OUTPUT -o $WIFIEXTERNAL -p udp --sport 67 --dport 68 -m state --state ESTABLISHED -j ACCEPT
iptables -I INPUT -s 10.0.0.0/24 -p udp --sport 68 --dport 67 -m state --state NEW,ESTABLISHED -j ACCEPT
iptables -I OUTPUT -s 10.0.0.0/24 -p udp --sport 67 --dport 68 -m state --state ESTABLISHED -j ACCEPT
```

Figura 15. Definición del servicio DHCP.

3.1.5 Servicio de enrutamiento y direccionamiento.

El servicio de encaminamiento y direccionamiento permite a dispositivos con múltiples interfaces de red enviar a otros dispositivos información por el interfaz adecuado. En nuestro caso, disponemos de un firewall que posee 4 interfaces de red donde cada interfaz se corresponde con una subred de la empresa. Será necesario por tanto, configurar el firewall para que el tráfico generado en la red llegue a su destino.

La configuración de este servicio permitirá al firewall, previa validación de los paquetes que entren y salgan por sus interfaces, entregar de forma correcta la información al destinatario correspondiente. En otras palabras, una vez que el firewall ha procesado un paquete, el dispositivo debe ser capaz de localizar que interfaz de red debe enviarlo.

Los objetivos principales del servicio son [26,27]:

- Interconectar los diferentes segmentos de red de la empresa.
- Tomar decisiones lógicas de encaminamiento basadas en direcciones IP.
- Determinar la mejor ruta para el flujo de datos de la red.

El primer paso para configurar este servicio es indicarle al núcleo de Linux del firewall que él será el dispositivo encargado de encaminar los paquetes de red que lo atraviesen. Para llevar esto a cabo se programarán en el shellscript previamente definido para la configuración de **iptables**, las dos funciones mostradas en la Figura 16. La primera de ellas, *enable_routing()*, permite modificar el fichero del sistema **ipv4** poniendo el parámetro *ip_forward* con valor 1. Esta modificación permite que el sistema pueda encaminar paquetes entre sus diferentes interfaces de red. La segunda función que es *disable_routing()* tiene el efecto contrario, poniendo el parámetro *ip_forward* con valor 0, para impedir al sistema encaminar paquetes entre las interfaces de red.

```
function enable_routing() {
        echo "Enabling Routing"
        echo 1 > /proc/sys/net/ipv4/ip_forward
}

function disable_routing() {
        echo "Disabling Kernel Routing"
        echo 0 > /proc/sys/net/ipv4/ip_forward
}
```

Figura 16. IP-forwarding.

Otro paso fundamental en el servicio de encaminamiento es asegurar, que cualquier paquete generado por un equipo en cualquier segmento de red, con destino a un equipo externo a la red, llegue a su destino final. Para ello, se debe permitir al firewall manipular estos paquetes reasignándoles una dirección IP pública, correspondiente en este caso a la dirección pública asignada por nuestro proveedor de servicios de internet. La Figura 17 muestra el código que debe programarse para permitir dicha función al firewall. Se enmascara la dirección IP privada de los equipos de las diferentes subredes con la dirección IP del firewall. Esto nos permite tener conectividad con el exterior ocultando las IP privadas. Esta regla tiene exclusivamente sentido en el interfaz de salida hacia internet, es decir, en el interfaz definido por la cadena EXT como se ha mencionado anteriormente.

```
# Enable NAT
iptables -t nat -A POSTROUTING -o $EXT -j MASQUERADE
```

Figura 17. Enmascaramiento de salida.

El tercer paso en la configuración del servicio de enrutamiento, es permitir, dentro de la configuración del firewall, el paso de paquetes entre los diferentes segmentos de red de la empresa. En la Figura 18, se muestra la configuración de las reglas de paso (*FORWARD*) que permiten el paso de paquetes entre los segmentos de red internal, Celonis-int, Celonis-ext y del interfaz de salida a internet EXT.

```
# Enable NAT between external and internal subnet
#iptables -A FORWARD -i $INTERNAL -o $EXT -j ACCEPT
#iptables -A FORWARD -i $EXT -o $INTERNAL -m state --state RELATED,ESTABLISHED -j ACCEPT
# Enable NAT between external and wifiinternal subnet
#iptables -A FORWARD -i $WIFIINTERNAL -o $EXT -j ACCEPT
#iptables -A FORWARD -i $EXT -o $WIFIINTERNAL -m state --state RELATED,ESTABLISHED -j ACCEPT
# Enable NAT between external and wifiexternal subnet
#iptables -A FORWARD -i $WIFIEXTERNAL -o $EXT -j ACCEPT
#iptables -A FORWARD -i $EXT -o $WIFIEXTERNAL -m state --state RELATED,ESTABLISHED -j ACCEPT
```

Figura 18. Forwarding entre las subredes y el interfaz de acceso a Internet.

Por último, para finalizar la configuración del servicio de enrutamiento, se debe construir adicionalmente, la tabla de enrutamiento en el firewall mediante el comando **route**. Esto permitirá al firewall tener configurado de forma estática las diferentes rutas existentes hacia los segmentos de red, decidiendo en cada caso por qué interfaz saldrá un paquete hacia su destino. Esta configuración de rutas estáticas en el firewall para el enrutamiento y dirección del tráfico de las interfaces de red se muestra en la Figura 19. Un caso especial es el de la interfaz tap0, creado por la VPN de la empresa de forma dinámica, que debe ser considerado de forma específica para permitir el encaminamiento de los paquetes generados en dicha red virtual.

```
antonio@wall:~$ sudo route -n
Kernel IP routing table
Destination     Gateway         Genmask         Flags Metric Ref    Use Iface
192.168.20.0    0.0.0.0         255.255.255.0   U     0      0        0 eth3
10.0.0.0        0.0.0.0         255.255.255.0   U     0      0        0 tap0
192.168.10.0    0.0.0.0         255.255.255.0   U     0      0        0 eth2
192.168.1.0     0.0.0.0         255.255.255.0   U     0      0        0 eth1
192.168.0.0     0.0.0.0         255.255.255.0   U     0      0        0 eth0
0.0.0.0         192.168.0.1     0.0.0.0         UG    100    0        0 eth0
```

Figura 19. Tabla de rutas del firewall.

3.2 Red Privada Virtual (VPN)

3.2.1 Introducción a las redes virtuales

Una red privada virtual (*VPN*) es una red lógica o virtual creada sobre una infraestructura compartida, pero que proporciona los mecanismos de seguridad necesarios para una comunicación segura [28,29]. Para ello se combinan dos tecnologías:

- Las tecnologías de seguridad, que permiten definir una red privada como medio de comunicación confidencial cuyo tráfico no puede ser interceptado por usuarios ajenos a la red.
- Las tecnologías de encapsulamiento de protocolos, que permiten que se pueda utilizar una infraestructura de red pública, como internet, para definir por encima de ella una red virtual, en vez de una conexión física dedicada para la red privada.

Las redes virtuales presentan una serie de ventajas entre las que podemos destacar [30]:

- **Ahorro:** permite conectar redes físicamente separadas sin necesidad de usar una red dedicada.
- **Transparencia:** interconectar redes diferentes es totalmente transparente para el usuario final.
- **Seguridad:** se puede asegurar múltiples servicios a través de un único mecanismo.
- **Movilidad:** se puede asegurar conexiones entre usuarios móviles y la red fija.
- **Simplicidad:** facilita la administración de la conexión de servidores y aplicaciones entre diferentes dominios.

No obstante, el uso de redes privadas virtuales presenta también una serie de desventajas que deben tenerse en cuenta:

- **Fiabilidad:** fallos en la red pueden dejar incomunicados los recursos de la red.
- **Confianza entre sedes:** si la seguridad de una subred involucrada en la red virtual se ve comprometida, puede afectar a la seguridad de todos los componentes de la VPN.
- **Interoperabilidad:** posibles incompatibilidades con las configuraciones VPN.

3.2.2 Requisitos

Se ha especificado por parte de la empresa Celonis GmbH la necesidad de configurar accesos externos a la red, para facilitar el acceso a los recursos internos al personal y clientes de la misma. Los usuarios designados por la empresa deben poder tener acceso a los servidores y servicios virtuales desde cualquier punto externo a la red. Este acceso se debe configurar sobre un canal seguro, de comunicación cifrada. Se debe poder configurar tanto con sistemas operativos Windows como con sistemas operativos Linux y como última condición, será indispensable y necesario configurar a cada usuario una contraseña con la que poder validar la conexión.

3.2.3 Sotware OpenVPN

Según los requisitos especificados con anterioridad, se ha decidido configurar una red privada virtual a través del software **OpenVPN.** Este software es una solución de conectividad basada en SSL de forma que se facilita una conectividad punto a punto con validación jerárquica de usuarios y hosts conectados remotamente.

La conexión a la VPN desde cualquier punto de la red será cifrada mediante una conexión SSL y seguridad TSL. Ambos protocolos hacen uso de certificados digitales para establecer una comunicación segura a través de internet.

3.2.4 Conexiones SSL/TTL mediante certificados digitales

Para entender el funcionamiento de esta tecnología es necesario entender los conceptos sobre los que se asienta [31-33].

- **Cifrado, no encriptado**: El cifrado es el proceso que transforma la información de manera que otros usuarios no puedan entenderla. Se realiza con base a un elemento único conocido como llave, así nadie, excepto el poseedor puede leerla. El procedimiento inverso al cifrado es el descifrado.
- **Llave pública y llave privada**: Son un par de "llaves" digitales asociadas a una persona o entidad y generadas mediante métodos criptográficos. La llave pública es usada para cifrar la información, haciendo una analogía, es como la llave utilizada para cerrar una puerta y mantener fuera a cualquier persona. La llave privada, por el contrario, se usa para descifrar, es decir, la llave que abre la puerta y sólo la posee la persona autorizada; por lo tanto ésta debe mantenerse en secreto.

- **Firma digital:** elemento que identifica y distingue a cada persona y que al firmar se adquieren derechos y obligaciones. La firma digital se genera con base a la llave privada de quien firma y por lo tanto es única.
- **Autoridad Certificadora:** Una autoridad certificadora (*CA*, *Certificate Authority*) es una entidad confiable que se encarga de garantizar que el poseedor de un certificado digital sea quien dice ser, bridando confianza a ambas partes de una comunicación segura SSL/TLS.
- **Certificado digital SSL / TLS**: Es un documento digital único que garantiza la vinculación entre una persona o entidad y su llave pública. Contiene información de su propietario como nombre, dirección, correo electrónico, organización a la que pertenece y su llave pública, así como información propia del certificado por mencionar: periodo de validez, número de serie único, nombre de la AC que lo emitió, firma digital de la AC cifrada con su llave privada y otros datos más que indican cómo puede usarse ese certificado [34].

3.2.5 Instalación del Servidor OpenVPN.

Al disponer de un firewall con sistema operativo Linux, instalaremos el software OpenVPN [35-38] sobre el propio sistema operativo que controla el firewall, para facilitar su configuración y aprovechar así la configuración de iptables descrita anteriormente.

Los paquetes Linux que se necesitan para la instalación son **openvpn** y **bridge-utils.** Para instalarlos bastaría con ejecutar desde un terminal:

>> sudo apt-get install openvpn bridge-utils

Como se describió a lo largo del Capítulo 2, la red virtual se fijará en la dirección de red 10.0.0.0/24, por lo que deberemos configurar un interfaz de red que comunique con los dispositivos conectados a esta subred. Podemos definir dos tipos de dispositivos:

- **Dispositivo punto a punto** (*TUN*): El controlador TUN emula un dispositivo punto a punto y es utilizado para crear túneles virtuales operando con el protocolo de internet (*IP*). De esta forma se pueden encapsular todos los paquetes que se transporten a través de él como datagramas TCP o UDP. Las máquinas que queden detrás de cada uno de los extremos del enlace pertenecerán a subredes diferentes.

- **Dispositivo Ethernet virtual** (*TAP*): Simula una interfaz de red Ethernet, más comúnmente conocido como modo **puente o bridge.** Estos túneles TAP virtuales **encapsulan directamente paquetes Ethernet**, lo que permite empaquetar entramados diferentes al IP. Las máquinas situadas

detrás de cada uno de los extremos del enlace pueden operar como parte de la misma subred (si se utiliza el protocolo IP). El modo de funcionamiento puente es particularmente útil para enlazar usuarios remotos, ya que éstos pueden conectarse a un mismo servidor y virtualmente formar parte de la red principal.

Para permitir que varios usuarios se puedan conectar simultáneamente a la VPN, configuraremos una interfaz TAP que permitirá la conexión de usuarios y clientes de la empresa a un mismo servidor, formando parte de la red. Para ello será necesario configurar este dispositivo en el fichero **interfaces**. En la Figura 20 se muestra la configuración del interfaz TAP asignándole una dirección de red propia dentro de la red virtual definida en los requisitos del proyecto (segmento de red VPN con dirección 10.0.0.0/24).

```
# The open VPN network adapter
auto tap0
iface tap0 inet static
        address 10.0.0.1
        netmask 255.255.255.0
        pre-up openvpn --dev tap0 --mktun
```

Figura 20. Configuración de un interfaz TAP.

3.2.5.1 Certificado digital del servidor

El siguiente paso consiste en generar los certificados. Será necesario construir un certificado de autoridad propio a través del script facilitado por el software. La Figura 21 muestra la configuración de parámetros necesaria para crear la Autoridad Certificadora (en este caso la empresa Celonis GmbH). La ejecución del script solicitará el país de la autoridad (*Alemania, Deuschland-DE*), provincia (*Bayer, BY*), la ciudad (*Munich*), la organización a la que pertenece (*Celonis GmbH*) y el mail de contacto.

```
export KEY_COUNTRY="DE"
export KEY_PROVINCE="BY"
export KEY_CITY="Munich"
export KEY_ORG="Celonis GmbH"
export KEY_EMAIL="info@celonis.de"
```

Figura 21. Parámetros de la Autoridad Certificadora.

Seguidamente se procede a crear el certificado y las llaves a través de la ejecución como root de la siguiente serie de comandos en el terminal:

>> cd /etc/openvpn/easy-rsa/

>>sudo chown -R root:admin

>>sudo chmod g+w

```
>>source ./vars

>> ./clean-all

>> ./build-dh

>> ./pkitool --initca ## CERTIFICADO AC Y LLAVE

>> ./pkitool --server server ## CERTIFICADO SERVIDOR Y LLAVE

>> cd keys

>> openvpn --genkey --secret ta.key  ## CONSTRUCCION LLAVE TSL

>> sudo cp server.crt server.key ca.crt dh1024.pem ta.key ../../
```

Por último es necesario editar el fichero de configuración del servidor OpenVPN denominado **server.conf** ubicado en */etc/openvpn/*. La Figura 22 muestra dicho fichero de configuración. El parámetro **dev** establece que tipo de interfaz virtual manejará el servidor OpenVPN, siendo en este caso un dispositivo TAP tal como se comentaba anteriormente, minetras que el parámetro **proto** indica que el protocolo a usar será TCP. Mediante **port** se especifica que el puerto en el que correrá el servicio será el 1194. Los parámetros **ca, cert, key** y **dh** especifican en que directorio se localizan los ficheros de configuración creados en este proceso, siendo en este caso la carpeta raíz donde se instalaron los ficheros del servicio. Por otro lado, los parámetros **user** y **group** especifican que usuario y grupo son los propietarios del proceso, siendo por seguridad, en este caso, ningún usuario y ningún grupo. El parámetro **server** constituye la red y máscara sobre la que se construirá la red virtual privada. El parámetro **ifconfig-pool-persist** establece un fichero que almacenará las IP en uso dentro de la VPN, por parte de usuarios conectados a la misma. El parámetro **status** permite crear un fichero **log** con el estado del servicio. El resto de parámetros no son de relevancia para el objetivo perseguido ya que proporcionan configuraciones adicionales como maximizar el redireccionamiento (**push**) y de compresión de la información (**comp-lzo**), por mencionar algunos ejemplos.

```
antonio@wall:~$ sudo cat /etc/openvpn/server.conf
dev tap0
proto tcp
port 1194
ca ca.crt
cert server.crt
key server.key
dh dh1024.pem

user nobody
group nogroup

server 10.0.0.0 255.255.255.0

ifconfig-pool-persist /etc/openvpn/clients.txt
status /etc/openvpn/status.txt

persist-key
#persist-tun

push "redirect-gateway def1"
#push "route 0.0.0.0 0.0.0.0 10.0.0.1"
push "dhcp-option DNS 10.0.0.1"

keepalive 10 120
verb 3
comp-lzo
max-clients 3
```

*Figura 22. Fichero de configuración **server.conf**.*

3.2.5.2 Certificado digital del cliente

Para generar un certificado que permita a un usuario acceder desde el exterior a la red Celonis GmbH a través de la VPN, se utilizará como **root** la siguiente secuencia de comandos en consola:

```
>> cd /etc/openvpn/easy-rsa/

>> source ./vars

>> ./pkitool a.albendea       ## crear un certificado con nombre a.albendea
```

Con ella se generan los archivos **a.albendea.crt** y **a.albendea.key.** El primero de ellos es una firma digital, es decir, establece quién es un usuario. El fichero con extensión key es la llave que nos permitirá establecer la conexión con la red virtual de la empresa y poder acceder de forma remota desde el exterior.

3.2.5.3 Encriptación de certificados digitales del cliente

Para mejorar la seguridad de los certificados digitales de los clientes, y evitar que, mediante un robo de los ficheros, un usuario no autorizado pueda acceder a la red Celonis GmbH a través de la VPN, se encriptarán los certificados de los clientes bajo el sistema de encriptamiento avanzado (*AES256,* Advanced Encryption Standard**)**, lo cual obligará al usuario a introducir una

contraseña para habilitar su uso. En caso de que un usuario no autorizado consiga robar los certificados digitales y trate de conectarse a la red de Celonis GmbH sin el conocimiento de esta contraseña, jamás podrá acceder a los recursos y servicios internos de la empresa.

Para establecer la criptografía en un certificado basta con ejecutar desde el terminal donde se ubique el servidor OpenVPN, el comando openssl [38]:

>> Openssl rsa –aes256 –in a.albendea.key –out a.albendeac.key

El objetivo de este comando es dotar de una clave o contraseña de uso al certificado .key .Cada vez que el fichero a.albendeac.key sea usado en lectura o escritura, será necesaria la contraseña para poder acceder a su contenido.

3.2.6 Instalación del cliente OpenVPN bajo clientes Microsoft

Para la instalación de un cliente OpenVPN será necesario descargarse el cliente desde la página web oficial (**http://openvpn.net/index.php/open-source/downloads.html**) e instalarlo con las opciones por defecto:

Finalizada la instalación será necesario copiar los ficheros **ca.crt**, **dh1024.pem** y los ficheros del cliente generados anteriormente (tanto el fichero **.crt** como el fichero **.key**) en la carpeta **config** perteneciente al cliente **OpenVPN**.

Será necesario, además, crear el archivo principal de configuración a leer por el cliente OpenVPN cada vez que ejecuta el servicio. Dicho fichero tiene extensión **.ovpn** y permite por tanto añadir parámetros de configuración del cliente, entre los cuáles se encuentran los siguientes:

Client # NOMBRE DEL FICHERO .ovpn
dev tap #TIPO DE DISPOSITIVO
proto tcp # PROTOCOLO
remote **IP PUBLICA:PUERTO #** DIRECCIÓN:PUERTO A CONECTAR
resolv-retry infinite
nobind
persist-key
persist-tun
CERTIFICADO Y LLAVE DEL CLIENTE
ca ca.crt
cert a.albendea.crt
key a.albendeac.key
comp-lzo
verb 3

Una vez finalizada la configuración de los parámetros del cliente, éste podrá conectarse a la red desde cualquier punto externo a la misma, siempre que disponga de conectividad. La captura de pantalla representada por la Figura 23 muestra un listado de los ficheros de configuración necesarios para el funcionamiento del cliente.

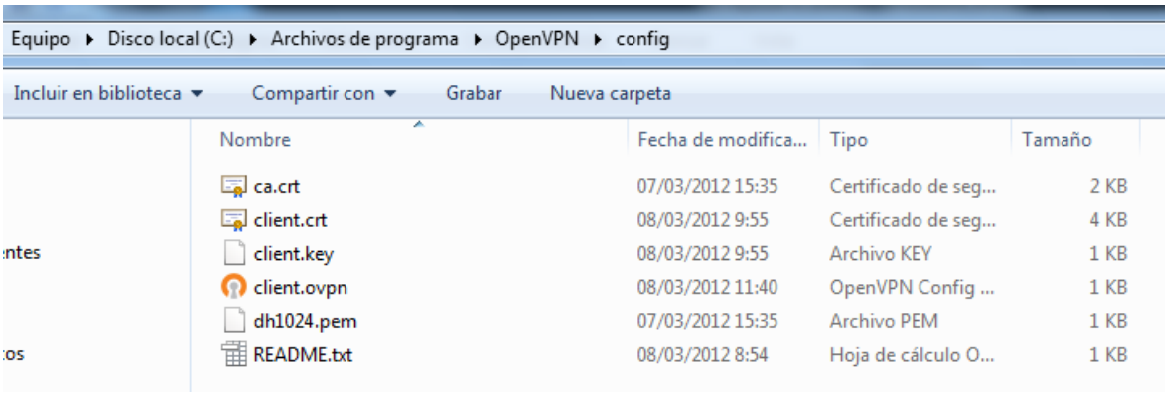

Figura 23. Archivos de configuración del cliente del usuario client.

3.2.7 Configuraciones adicionales

Para un correcto funcionamiento de la VPN aún será necesario realizar una configuración adicional. Cualquier conexión desde el exterior a la red de la empresa se realizará a través de la dirección IP pública, en este caso, la asignada al router. Es necesario, por tanto, configurar el router para que tenga la capacidad de redireccionar las conexiones hacia la red local. Esto requiere configurar el router D-Link modificando la opción port-fordwarding especificando que cualquier conexión OpenVPN validada debe ser redirigida hacia al interior de la red.

Un problema común al que se enfrentan las VPNs son los cambios en la dirección pública asignada al router. Este cambio afectaría a todos los miembros de la empresa que se conectasen mediante VPN, obligando a actualizar los ficheros de configuración. Si el número de usuarios que emplean la VPN es muy alto, el tiempo de modificación sería muy elevado. Este contratiempo se puede evitar mediante la utilización del servicio de redes dinámicas (*DYNDNS, Dynamic DNS Service*) [39]. Este servicio permitirá controlar los cambios que se realicen sobre nuestra IP pública. Los clientes se conectarán sobre este servicio y el mismo será quien les proporcione la dirección IP a la cual conectarse. Una vez contratado y configurado el servicio, basta con editar los ficheros **.ovpn** de los clientes OpenVPN y actualizar la entrada **remote** sustituyendo la IP pública por la dirección que resolverá la petición**.** Un ejemplo de configuración sería **remote celonis.dyndns-remote.com** que resuelve la dirección pública de la empresa Celonis GmbH de forma automatizada.

CAPÍTULO 4
DESARROLLO DE LA INFRAESTRUCTURA VIRTUAL

En este capítulo se hará hincapié en el diseño y puesta en marcha de una infraestructura virtual bajo el sistema KVM para la creación y configuración de máquinas virtuales con diferentes sistemas operativos (Windows Server, Linux Server, etc.). En definitiva, todos aquellos servicios que la empresa pueda necesitar u ofertar, pueden virtualizarse.

4.1 Virtualización de sistemas

Se puede definir la **virtualización** como una técnica de **abstracción del hardware** de un computador. Mediante un **hypervisor** se crea una capa de abstracción entre el hardware real de la computadora y el sistema operativo [40] de la máquina virtual, dividiéndose el hardware real en entornos de ejecución o **máquinas virtuales** [41-43].

4.1.1 Hypervisor

Un **hypervisor** es una capa software que gestiona los recursos principales del computador (procesador, memoria, disco y red) para distribuirlo entre los sistemas virtuales gestionados. Esto permitirá disponer de entornos de ejecución variados (Windows, Linux, Mac OS, etc.) en la misma máquina física que reparten o comparten los recursos, dando una visión de entornos privados de ejecución, donde cada entorno parece poseer los recursos en propiedad exclusiva [44].

El funcionamiento del hypervisor se muestra en la Figura 24, donde se observa como éste tiene el control de los recursos hardware (representados por memoria, disco, red y uso de CPU). Mediante el uso de un interfaz de programación de aplicaciones (*API, Application Programming Interface*) [45] se distribuyen los recursos hardware en instancias virtuales (memoria virtual o *virtual memory*, disco virtual o *virtual disk*,etc.) a asignar a diferentes instancias o máquinas virtuales (invitado virtual o *virtual guest* 1, *virtual guest* 2, etc.).

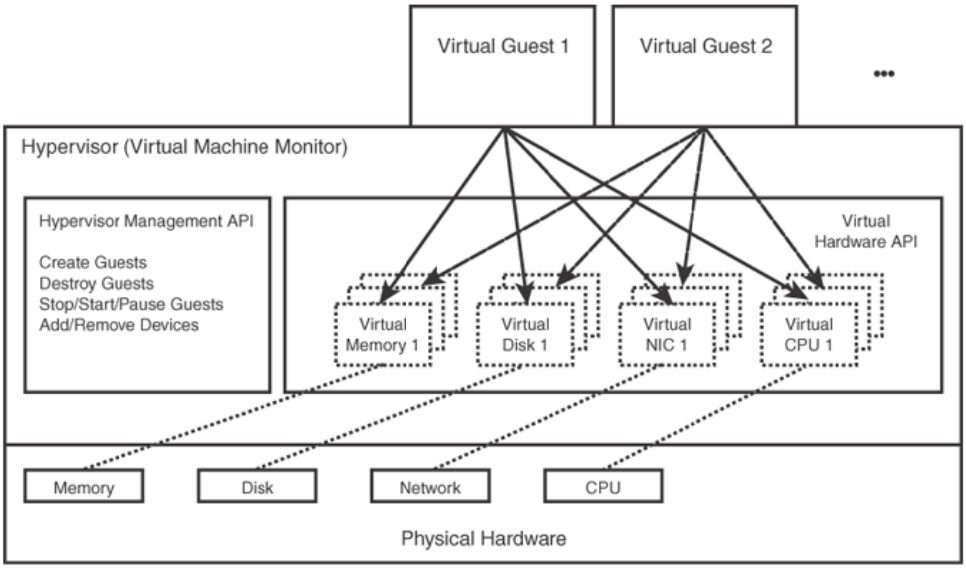

Figura 24. Hypervisor de un sistema.

La técnica de virtualización permite diferentes configuraciones:

- **Emulación**

La técnica de **emulación** persigue simular el hardware original de los sistemas virtualizados. Persigue ejecutar los clientes virtualizados imitando su arquitectura propietaria (este es el caso, por ejemplo, de los emuladores de videojuegos en PCs). La Figura 25 muestra un ejemplo gráfico de emulación. En ella, se imita, bajo un hardware específico, diferente hardware no nativo sobre el que corren sistemas virtuales no modificados. Estos sistemas no modificados, a su vez, ejecutan aplicaciones propietarias [46,47].

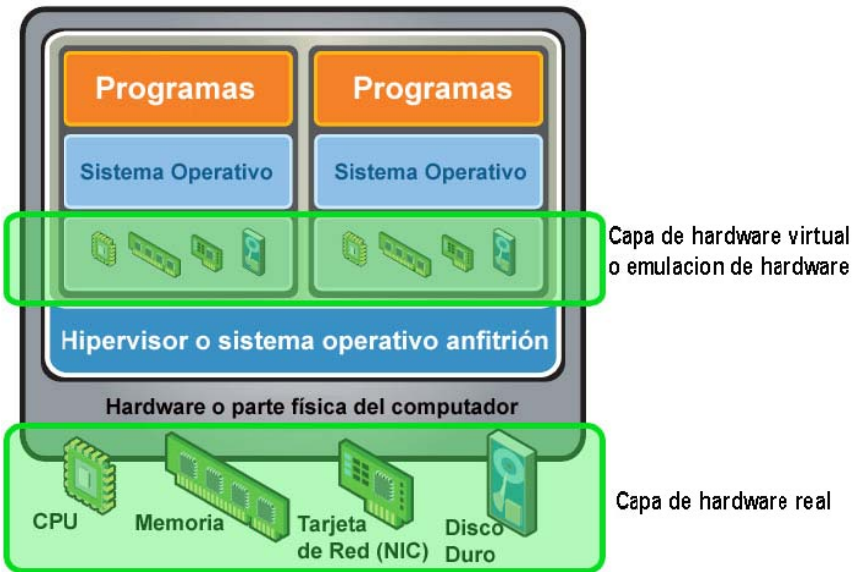

Figura 25. Emulación.

- **Virtualización completa (*Full virtualization*)**

La **virtualización completa** o **virtualización nativa** [48] es similar a la emulación. El anfitrión simula un hardware suficiente para permitir que se ejecuten sistemas virtuales sin modificar directamente bajo el hardware simulado, mediante un hypervisor. Una gran cantidad de instrucciones máquina provenientes de los sistemas virtuales son ejecutadas directamente. Se puede obtener virtualización completa mediante el uso de aplicaciones virtualizadoras tales como VMWare Workstation, VMWare Server, Windows Server 2008 R2 Hyper-V, VirtualBox, Parallels, Oracle VM, Xen Server, Virtual PC, etc.

Al igual que en la Fig. 25 se mostraba un ejemplo de emulación, el mostrado en la Figura 26 se corresponde con una virtualización completa. En ella, mediante el hypervisor correspondiente a este tipo de virtualización se ejecutan sistemas virtuales no modificados ni por el hypervisor ni por la arquitectura del anfitrión.

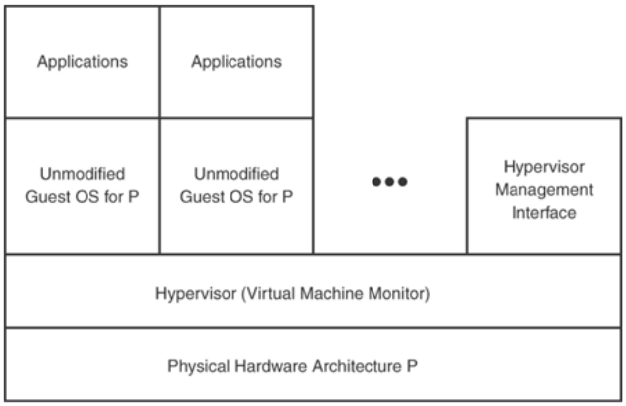

Figura 26. Full Virtualización.

- **Para-Virtualización**

La **para-virtualización** es la técnica más moderna de virtualización. Los procesadores más avanzados incluyen instrucciones virtualizadoras dentro de su repertorio. Esto permite mejorar la eficiencia de las máquinas virtuales y acercarlas a su rendimiento nativo. El hypervisor se ejecuta en el modo más privilegiado y se encarga de exportar versiones modificadas del hardware subyacente denominadas **instancias** vía software. Además, los sistemas virtuales son modificados para ejecutarse en las instancias. Se introducen pequeñas modificaciones y cambios en las instancias para que sean más rápidas y sencillas, de forma que permita trabajar con múltiples sistemas operativos invitados [49,50]. La Figura 27 representa un ejemplo de este tipo de virtualización.

Figura 27. Para-virtualización.

Con el objeto de clarificar las implicaciones de cada tipo de virtualización en la Tabla 4 se comparan específicamente las ventajas e inconvenientes de cada una de ellas.

Tabla 4. Ventajas y desventajas de las diferentes técnicas de virtualización.

Virtualización	Ventajas	Desventajas
Emulación	Simula un hardware físicamente no disponible.	Bajo rendimiento
Full Virtualización	Flexibilidad: correr diferentes versiones de diferentes S.O. de múltiples proveedores.	El S.O no sabe que está siendo virtualizado.
		En I/O agresivas se puede incurrir en caída de rendimiento.
Para-virtualización	Ligero y rápido, con velocidades similares a los sistemas nativos.	El S.O debería ser portable para evitar el uso de instrucciones sensibles.
	Permite al S.O interactuar con el hypervisor y mejora la programación de recursos.	Sistemas de código cerrado presentan problemas.
	Permite virtualizar S.O. no compatibles con la full virtualización.	

4.1.2 Kernel Virtual Machine (KVM)

Kernel Virtual Machine o KVM es una solución para implementar virtualización completa con Linux sobre hardware x86 – x64. Está formada por un módulo del núcleo (con el nombre kvm.ko) y herramientas en el espacio de usuario, siendo en su totalidad software libre. El componente KVM para el núcleo está incluido en Linux desde la versión 2.6.20.

El sistema KVM representa una ventaja, ya que se trata de un sistema de virtualización opensource, sin coste para la empresa. Sistemas operativos como Windows Server, clientes Windows o sistemas Linux puedes instalarse en maquinas virtuales bajo este sistema. Cada máquina virtual tiene su propio hardware virtualizado: una tarjeta de red, discos duros, tarjetas gráfica, etc.

4.1.2.1 Prerrequisitos KVM

El primer paso a dar es comprobar si el hardware subyacente permitirá la **full virtualización** bajo KVM. Para ello, es condición necesaria que el hardware

a virtualizar soporte **microinstrucciones** INTEL-VT (para arquitecturas Intel) o AMD-V (para arquitecturas AMD).

Para chequear este requisito, bastará con ejecutar el comando:

>> egrep '(vmx|svm)' /proc/cpuinfo

En la Figura 28 se muestra la salida generada por la ejecución del comando anterior sobre uno de los servidores de la empresa sobre el que instalaremos KVM. En caso de no visualizar nada, KVM no podría ejecutarse sobre este software en concreto. Sería necesaria otra alternativa.

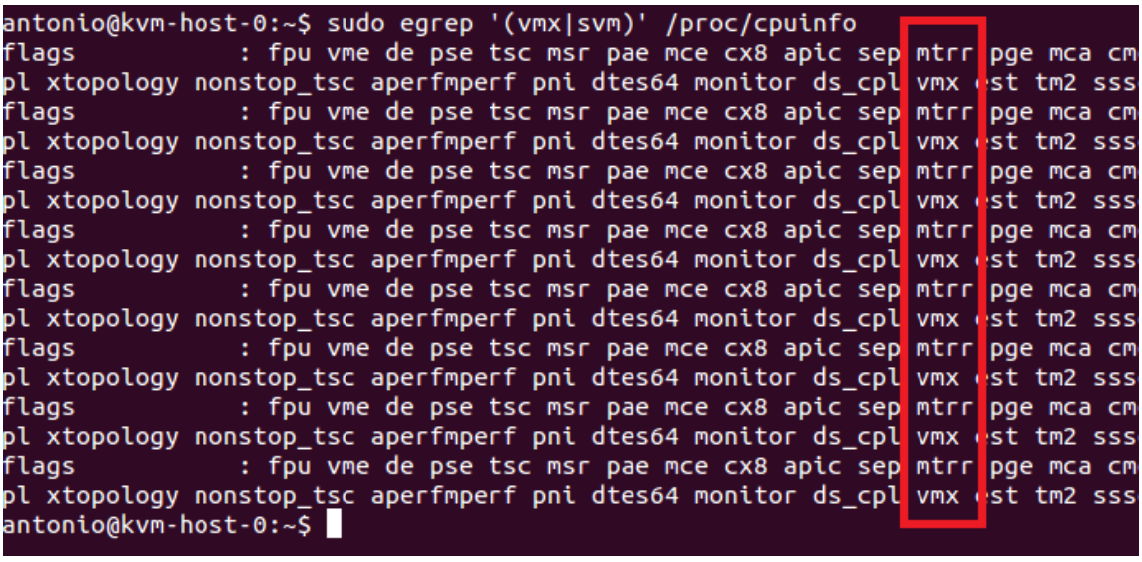

Figura 28. Microinstrucciones soportadas por cada procesador.

4.1.2.2 Instalación KVM

Para la instalación KVM bajo Linux [51] se requiere ejecutar en terminal el siguiente comando:

>> sudo apt-get install qemu-kvm libvirt-bin bridge-utils virt-manager

Los paquetes instalados tienen la siguiente función:

- qemu-kvm**:** hypervisor.
- libvirt-bin:binarios del demonio KVM que proporciona una herramienta provista de una línea de comandos.
- bridge-utils: utilería para proveer a la red de un puente o *bridge.*
- virt-manager: interfaz gráfica para administrar máquinas virtuales.

La instalación de los paquetes creará dos grupos administrativos denominados kvm y libvirt. Para añadir usuarios del sistema, que puedan administrar el entorno KVM, deben ser añadidos a estos grupos. Como root, se deben ejecutar los siguientes comandos en consola:

```
>> sudo adduser <usuario> libvirt
>> sudo adduser <usuario> kvm
```

Para comprobar la correcta instalación bastará con ejecutar desde consola:

```
>> sudo virsh –c qemu:/// system list
```

En la Figura 29 se puede analizar la conexión mediante consola al sistema para que visualice las máquinas virtuales y sus estados, aunque actualmente no se posee ninguna máquina virtual en ejecución.

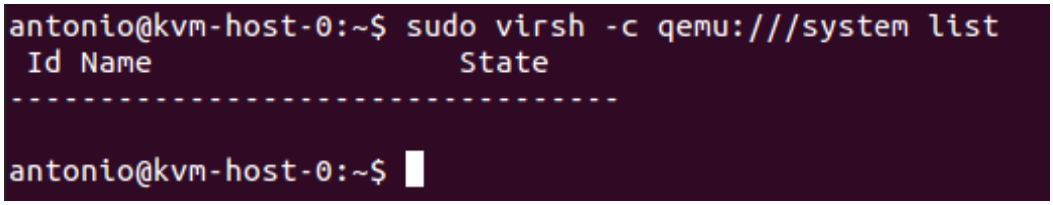

Figura 29. Instalación correcta de KVM.

4.1.2.3 Configuración de red en KVM

El paso posterior a la instalación KVM es configurar la red para permitir accesos externos a los servicios o servidores virtuales. El paquete instalado previamente **bridge-utils** se encargará de construir un puente para habilitar el acceso desde hosts externos de la red a los servicios o servidores virtuales hospedados en el servidor físico.

Actualmente, el servidor físico sobre el que se ha realizado la instalación KVM puede acceder a los servicios ofrecidos por equipos de la red o recursos de internet pero actualmente es invisible a los propios equipos.

Además, cualquier servicio o servidor desplegado en el servidor físico, por defecto, operara en la red virtual 10.0.2.0/24. Está configuración, por defecto, va en contra de la red desplegada en el capítulo 2, por lo que será necesario migrarla a la red 192.168.1.0/24.

La construcción del puente permite qué interfaces virtuales actúen como interfaces físicas, accesibles desde el resto de la red. Por cada servidor virtual desplegado, se dispondrá de al menos una interfaz virtual de red, por lo que es de vital importancia su acceso.

4.1.2.3.1 Red con puente o *bridge*

El dispositivo puente o *bridge* es una configuración específica de la tarjeta de red de un computador. Este tipo de configuración en modo puente permitirá el acceso en red a un número indeterminado de máquinas virtuales. El router o routers creerán que las máquinas virtuales son realmente máquinas físicas con su propia dirección IP. El computador es quién hace de puente entre los dispositivos de la red física y los sistemas virtuales.

La configuración bridge no se habilita por defecto en la instalación de los paquetes KVM, por lo que es necesario configurarla. El primer paso será configurar la capacidad CAP_NET_ADMIN o administración de la capa de red, para utilizar este tipo de red con puente. Ejecutaremos en terminal el siguiente comando:

```
>> sudo apt-get install libcap2-bin
```

Esta capacidad debe ser asignada únicamente a los administradores de la red, ya que un mal uso puede romper la red del sistema. El fichero **capability.conf** especificará qué usuarios tendrán asignada esta capacidad en el sistema.

Una segunda configuración necesaria será detener los servicios de red del servidor físico. Esto permitirá modificar la configuración de red de una forma segura. Para ello, como root se ejecuta en terminal el siguiente comando:

```
>> sudo invoke-rc.d networking stop
```

Se debe editar el fichero **interfaces** para configurar manualmente los parámetros del puente. En la Figura 30 se puede observar la configuración del bridge. El puente se define por **br0** asignándole una dirección IP, una máscara de red, la dirección de red, la dirección de multidifusión o *broadcast* y la dirección de la puerta de enlace o *gateway*. Además se especifican los siguientes parámetros:

- bridge_ports: especifica qué interfaz físico hará de puente, siendo eth0 en nuestro caso.
- bridge_stp: deshabilita el protocolo de spanning tree. Por razones de seguridad se recomienda deshabilitarlo.
- bridge_fd: establece el retardo o *delay* en el reenvío por parte del bridge.
- bridge_maxwait: establece el tiempo que deben esperar los scripts del bridge hasta llegar al estado de reenvío. Se recomienda deshabilitarlo.

Asimismo, dentro de la configuración del fichero **interfaces** añadimos el comando post-up ip link set **br0** address para asignar una dirección MAC ficticia al puente.

```
auto br0
iface br0 inet static
        address 192.168.1.20
        network 192.168.1.0
        netmask 255.255.255.0
        broadcast 192.168.1.255
        gateway 192.168.1.1
        bridge_ports eth0
        bridge_stp off
        bridge_fd 0
        bridge_maxwait 0

post-up ip link set br0 address e0:69:95:d1:ce:cf
```

Figura 30. Configuración del bridge.

Para iniciar la nueva configuración, se ejecutará en el terminal:

```
>> sudo /etc/init.d/networking restart
```

4.2 Virtualización de servicios

4.2.1 Requisitos

La empresa Celonis GmbH requiere el despliegue de los siguientes servicios virtuales:

1. **Controlador de dominio virtual**. El objetivo del servicio será integrar los servidores virtuales bajo una gestión de directorio activo e implementar futuras políticas de seguridad. Además, los miembros de la empresa así como los equipos físicos de la red deberán también integrarse en el controlador.

   Se denominará al dominio virtual **corp.celonis.de.**

2. **Servidor de correo Microsoft Exchange**. El objetivo del servicio será facilitar a los miembros de la compañía la función de calendario compartido, para una mejora en la gestión del tiempo. Además, el servidor deberá trabajar conjuntamente con el servidor de correo externo de la compañía para facilitar el mismo servicio de correo en Exchange, debiéndose sincronizar ambos servidores.

3. **Servidor de impresión**. El objetivo del servicio será facilitar a los miembros de la compañía un servicio de impresión entre las diferentes impresoras disponibles.

4. **Servidor SQL**. El objetivo del servicio será facilitar a los miembros de la compañía de servidores Microsoft SQL 2008 para el desarrollo de su trabajo. El acceso a los servidores se realizará con funciones del directorio activo.

Para el desarrollo de los servicios se dispone de un computador modelo RECT caracterizado principalmente por 8 procesadores y 16 GB de memoria RAM. Se establecerá como nombre del servidor **kvm-host-0**. Además, se dispondrá de forma auxiliar de dos servidores de la misma familia caracterizados por disponer de 4 procesadores y 16 GB de memoria RAM. El nombre de los servidores auxiliares será **cel-wrk-01** y **cel-wrk-02** respectivamente.

Por último, cualquier servicio desplegado en la infraestructura virtual deberá integrase en alguna política de backup.

4.2.2 Introducción a Virtual Machine Manager

Virtual Machine Manager (*VMM*) o monitor del sistema, es una aplicación libre que provee de una interfaz gráfica para la gestión de máquinas virtuales [52]. Presenta, a modo visual, un resumen de las máquinas virtuales (también denominadas dominios virtuales) así como una estadística en el uso, por parte de las máquinas, de los recursos del sistema así como de su rendimiento. Esta opción, es la recomendada por la wiki de KVM como gestor gráfico del entorno virtual. La Figura 31 muestra un resumen y estadísticas de uso del sistema de máquinas virtuales en los servidores que conforman el cluster.

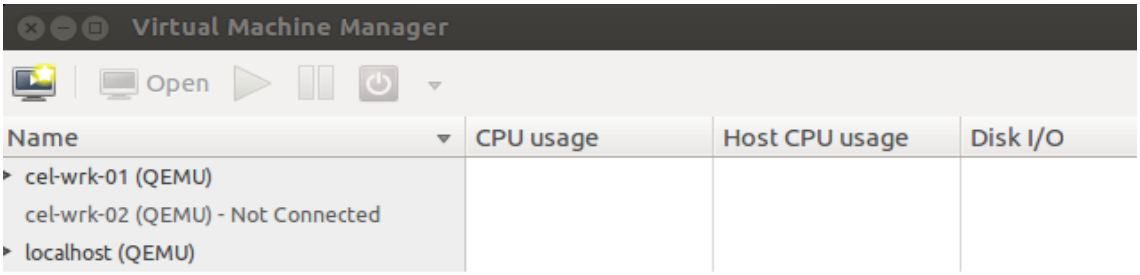

Figura 31. Resumen y Estadísticas de Maquinas Virtuales.

En la Figura 32, el interfaz VMM muestra las características del servidor **kvm-host-0**, servidor que forma parte del clúster de servidores de máquinas virtuales. El interfaz determina las características físicas del servidor así como una estadística en el uso de CPU y Memoria RAM en uso.

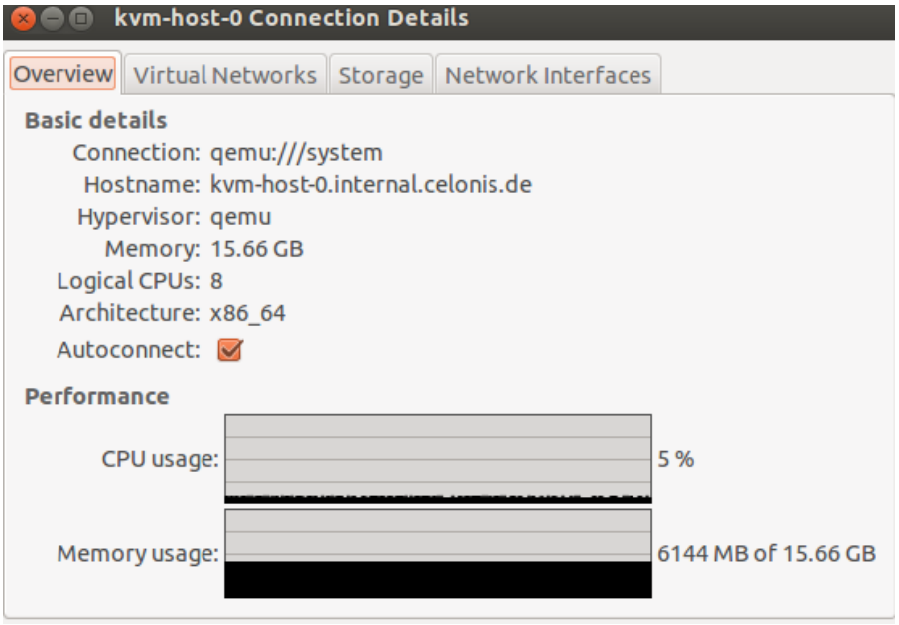

Figura 32. Resumen del uso de recursos sobre un servidor físico.

Esta aplicación tiene la capacidad, a su vez, de proporcionar un asistente para la creación de nuevas maquinas virtuales, de configurar y ajustar los recursos asignados a cada máquina y de su hardware virtual. La Figura 33 muestra el asistente de creación de una maquina virtual.

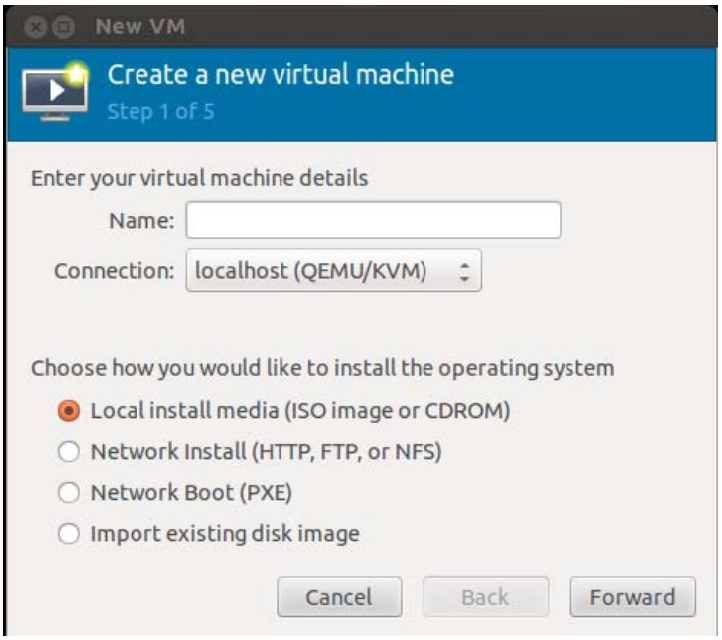

Figura 33. Asistente de creación de máquinas virtuales.

Se incluye, además, un sistema denominado computación virtual en red (*VNC, Virtual Networking Computing)* para obtener una completa consola gráfica de cada máquina virtual.

4.2.2.1 Herramientas de soporte

Una de las características más importantes de VMM es su compatibilidad con las herramientas provistas por la librería libvirt-bin, previamente instalada. Además dispone de un entorno gráfico de gestión y proporciona unas herramientas de soporte para la gestión de entornos virtuales:

- Herramienta Virt Install: esta herramienta provee de una línea de comandos para provisionar sistemas operativos en las máquinas virtuales. También provee de una interfaz de programación de aplicaciones (*API*) a VMM para el asistente gráfico de creación de máquinas virtuales.
- Herramienta Virt Clone: esta herramienta provee de una línea de comandos para clonar servidores virtuales. Su función básica es realizar una copia exacta de los discos virtuales de las máquinas virtuales reasignado a cada máquina clonada un nuevo nombre, un nuevo identificador universal único (*UUID*) y una nueva dirección MAC.
- Herramienta Virt Image: esta herramienta provee de una línea de comandos para instalar sistemas operativos en máquinas virtuales a partir de una imagen maestra predefinida o plantilla (*template*). La imagen maestra se construye sobre un disco preinstalado, con su sistema operativo y unos recursos mínimos asignados.
- Aplicación Virtual Machine Viewer: esta aplicación provee de una interfaz ligera para visualizar las máquinas virtuales.

4.2.2.2 Almacén de recursos

Un **almacén de recursos** o **storage** es una unidad organizativa que almacenará recursos disponibles, desde software hasta los propios discos máquinas virtuales o plantillas. El sistema VMM permite definir almacenes de recursos como medida organizativa del entorno virtual y gestionar mejor los recursos disponibles.

Para definir los almacenes, se disponen de los siguientes medios de almacenamiento:

- Disco duro primario 1 TB SATA donde reside el S.O. del servidor físico denominado **sistema**.
- Dos discos duros 1 TB SATA denominados **vms** y **backup** respectivamente.
- Disco externo USB – eSATA 2 TB denominado **external**.

En la Figura 34 se muestra los discos duros que permitirán definir los almacenes en el servidor **kvm-host-0.**

Figura 34. Gestor de discos de la máquina física.

Los almacenes se constituirán tal como se muestra en la Tabla 5.

Tabla 5. Definición de almacenes

Disco Duro	Nombre	Directorio	Objetivo
sda1	Sistema	/images	Almacenará las imágenes de instalación ISO y software general
sda1	Sistema	/templates	Almacenara las plantillas de instalación
sdb1	Vms	/vms	Almacenara los discos duros de las maquinas virtuales
sdc1	Backup	/backup	Almacenara las copias de seguridad de las máquinas virtuales
sdd1	External	/external	Almacenara las copias de seguridad de las máquinas virtuales

La Figura 34 muestra la constitución final de los almacenes configurados en el servidor **kvm-host-0.**

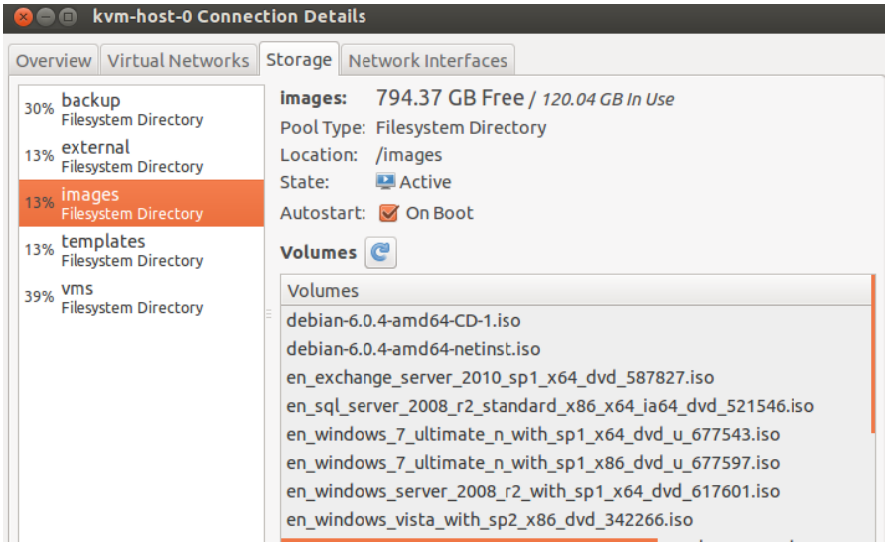

Figura 35. Definición del storage.

4.2.2.2.1 Cifrando el almacén.

Los almacenes, por su definición, no dejan de ser más que simples contenedores alojados en discos o directorios dentro de un disco. La información que se almacena en ellos puede llegar a ser crítica, teniendo en cuenta que las máquinas virtuales podrán contener información extremadamente sensible.

Por ello, un mecanismo de seguridad consiste en cifrar los discos duros que almacenarán la información [53,54]. Este mecanismo permitirá, proteger la información contenida en los discos duros en caso de robo. Cualquier conexión de los discos en otro computador sería inútil si el objetivo consiste en extraer la información. Los discos **sdb, sdc y sdd** deberán ser cifrados por ser susceptibles de poseer máquinas virtuales con información personal en su interior.

Para el cifrado de discos se utilizará el software denominado Linux Unified Key Setup (*LUKS*), un formato estándar de disco.

Aunque por defecto, en los kernel Linux server 2.6.32 o superiores ya se incluye el software, para instalar las librerías se debe ejecutar en consola:

```
>> sudo apt-get install cryptsetup
```

Posteriormente a la instalación del software de cifrado, se procederá a llenar los discos duros con datos aleatorios.

```
>> sudo dd if=/dev/urandom of=/dev/sdb1
>> sudo dd if=/dev/urandom of=/dev/sdc1
>> sudo dd if=/dev/urandom of=/dev/sdd1
```

Se procede a crear las particiones LUKS necesarias.

```
>> sudo cryptsetup –verify-passphrase –hash=sha256 –cipher=aes-cbc-essiv:sha256 --
   key-size=256 luksFormat /dev/sdb1
>> sudo cryptsetup –verify-passphrase –hash=sha256 –cipher=aes-cbc-essiv:sha256 --
   key-size=256 luksFormat /dev/sdc1
>> sudo cryptsetup –verify-passphrase –hash=sha256 –cipher=aes-cbc-essiv:sha256 --
   key-size=256 luksFormat /dev/sdd1
```

Es necesario, configurar el gestor de volúmenes (también conocido como mapeador de dispositivos) [55] para activar las particiones.

```
>> sudo cryptsetup luksOpen /dev/sdb1 VMS
>> sudo cryptsetup luksOpen /dev/sdc1 BACKUP
>> sudo cryptsetup luksOpen /dev/sdc1 EXTERNAL
```

Se debe comprobar que se ha configurado correctamente la activación anterior.

```
>> sudo cryptsetup status VMS
>> sudo cryptsetup status BACKUP
```

Una muestra de configuración correcta se puede apreciar en la Figura 36. En ella se muestras los volúmenes VMS y BACKUP como particiones cifradas bajo LUKS. Las particiones creadas tendrán formato **ext4** formateadas con el comando **mkfs.ext4**.

```
antonio@kvm-host-0:~$ sudo cryptsetup status VMS
/dev/mapper/VMS is active and is in use.
  type:    LUKS1
  cipher:  aes-cbc-essiv:sha256
  keysize: 256 bits
  device:  /dev/sdb1
  offset:  2056 sectors
  size:    1953100256 sectors
  mode:    read/write
antonio@kvm-host-0:~$ sudo cryptsetup status BACKUP
/dev/mapper/BACKUP is active and is in use.
  type:    LUKS1
  cipher:  aes-cbc-essiv:sha256
  keysize: 256 bits
  device:  /dev/sdc1
  offset:  2056 sectors
  size:    1953100256 sectors
  mode:    read/write
```

Figura 36. Particiones cifradas bajo LUKS.

Por último, se forzará a pedir una contraseña (*passphrase*) establecida en el proceso de configuración descrito anteriormente. Cada vez que el sistema sea reiniciado, se pedirá la contraseña para el montaje de los volúmenes. En caso de no introducirse, los volúmenes no serán accesibles, por lo que todo dato almacenado en ellos será inaccesible. Sera necesario configurar los ficheros de sistema **fstab** y **crypttab**. Las Figuras 37 y 38 muestran las configuraciones de los ficheros de sistema mencionados anteriormente para el cifrado de volúmenes.

```
antonio@kvm-host-0:~$ sudo cat /etc/crypttab
# <target name> <source device>         <key file>      <options>
cryptswap1 /dev/sda5 /dev/urandom swap,cipher=aes-cbc-essiv:sha256

VMS /dev/sdb1 none luks,cipher=aes-cbc-essiv:sha256
BACKUP /dev/sdc1 none luks,cipher=aes-cbc-essiv:sha256
EXTERNAL /deb/sdd1 none luks,cipher=aes-cbc-essiv:sha256
```

Figura 37. Configuración /etc/crypttab.

```
antonio@kvm-host-0:~$ sudo cat /etc/fstab
# /etc/fstab: static file system information.
#
# Use 'blkid -o value -s UUID' to print the universally unique identifier
# for a device; this may be used with UUID= as a more robust way to name
# devices that works even if disks are added and removed. See fstab(5).
#
# <file system> <mount point>   <type>  <options>       <dump>  <pass>
proc            /proc           proc    nodev,noexec,nosuid 0       0
# / was on /dev/sda1 during installation
UUID=60eaadf9-3e34-4253-886d-8dcff819538b /               ext4    errors=remount-ro 0       1
# swap was on /dev/sda5 during installation
#UUID=837e0784-435b-450c-8848-6a1c5c99d8cc none            swap    sw              0       0
/dev/mapper/cryptswap1 none swap sw 0 0
#/dev/mapper/raid /vms ext4 rw,errors=remount-ro 0       1
/dev/dm-2 /backup ext4 rw,errors=remount-ro 0 1
/dev/dm-1 /vms ext4 rw,errors=remount-ro 0 1
/dev/dm-3 /external ext4 rw,errors=remount-ro 0 1
```

Figura 38. Configuración /etc/fstab.

Para comprobar la correcta configuración, podemos acceder a la utilidad de discos del servidor y analizar la información proporcionada de cada disco, tal como se muestra en la Figura 39 y Figura 40, representantes del volumen VMS previamente cifrado. La Figura 39 muestra la **partición 1 del disco sdb** como un volumen encriptado. La figura 40 muestra el sistema de archivos montado sobre la partición del tipo **ext4**.

Figura 39. Utilidad de discos: información de la partición VMS.

Figura 40. Utilidad de discos: información sistema de archivos en sdb1 donde se monta VMS.

Una forma de ver el rendimiento que se puede obtener en discos encriptados, se pueden utilizar una técnica denominada comparativa (*benchmark*) [56]. La idea es tener una estimación del tiempo de proceso en la ejecución de un programa. En este caso, serán los discos duros virtuales los que estarán en ejecución dentro de los volúmenes cifrados. La Figura 41 muestra los resultados de ejecutar un *benchmark* en la partición **sdb1** donde se miden velocidades de lectura (línea azul) y de escritura (línea roja) sobre discos duros almacenados en **sdb1**.

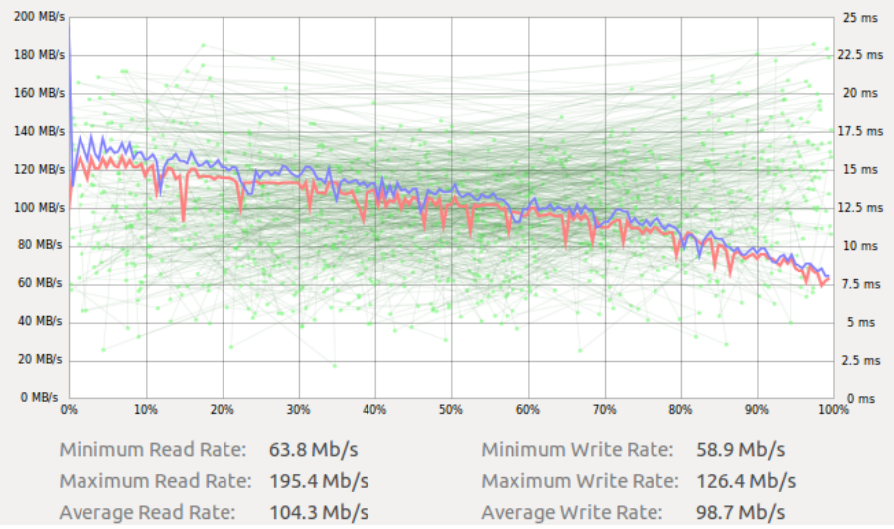

Figura 41. Benchmarking: información de rendimiento del disco sdb1.

4.2.2.3 Formato de discos

Para la instalación de sistemas virtuales se necesitan discos duros virtuales, que simulan discos duros físicos como medios de almacenamiento [57-59]. El sistema de virtualización KVM permite manejar diferentes formatos de discos duros, pero los más utilizados son.

- Formato en crudo (*RAW*): es una imagen binaria simple de la imagen de disco, y es muy portable. Las imágenes bajo este formato solo ocupan el espacio registrado por los datos que albergan.

- **For**mato copia en escritura (*QCOW2*): posee una serie de características especiales incluyendo la habilidad de tomar múltiples snapshots [60], encriptación AES, y compresión zlib [61] opcional entre otras.

Según la utilidad buscada, se seleccionará un formato u otro. Ambos formatos en según qué entorno (sistema operativo virtualizado, drivers de los discos duros, etc.) proporcionarán rendimientos diferentes tanto de lectura como escritura de disco. Aunque no existe una comparativa exacta del rendimiento, el formato QCOW2 tiene mayor penalización en comparación con RAW cuando es necesario hacer crecer la imagen del sistema. Para la creación de máquinas virtuales base o plantillas, se utilizará el formato QCOW2 mientras que para crear maquinas virtuales cuyo rendimiento sea importante (servidores SQL) se utilizara el formato RAW. La Figura 42 es un ejemplo de sistemas virtuales bajo formato QCOW. Se observan las diferentes plantillas para sistemas Microsoft. La Figura 43 muestra sistemas virtuales bajo formato RAW.

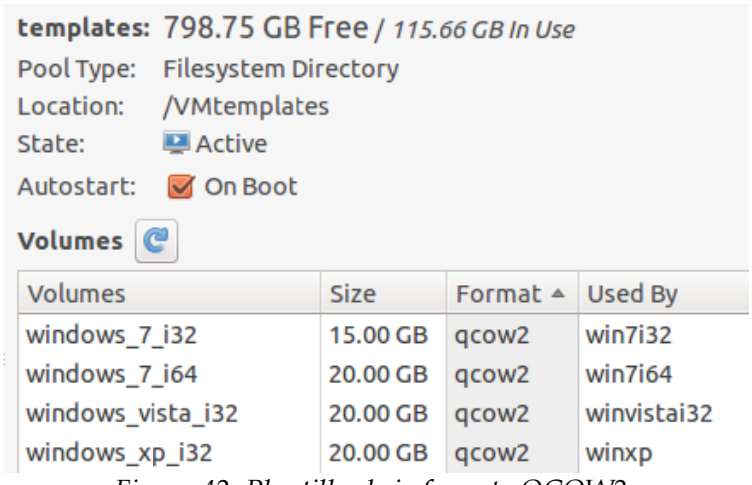

Figura 42. Plantillas bajo formato QCOW2.

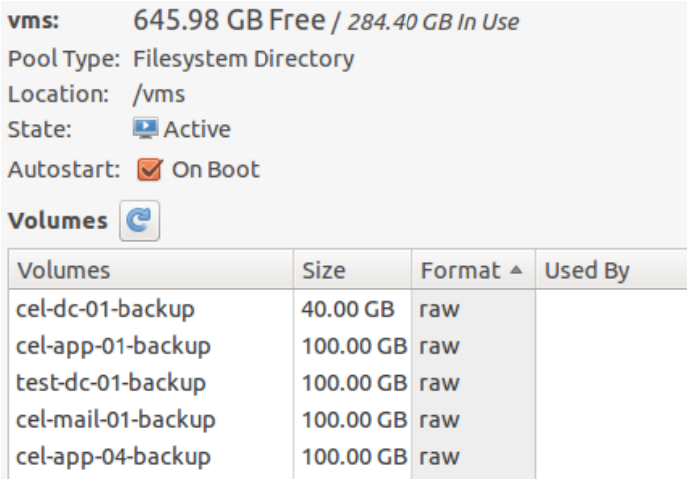

Figura 43. Sistemas en producción bajo formato RAW.

4.2.3 Configuración de servidores y servicios virtuales

Finalizada la configuración de la infraestructura virtual, se procede a instalar una serie de servidores y servicios virtuales sobre los que la empresa se apoyará para su constante crecimiento.

4.2.3.1 Controlador de dominio

Un controlador de dominio gestiona la autentificación de usuarios para permitir o denegar el acceso de estos a los recursos compartidos o maquinas de red, normalmente, a través del uso de contraseñas, entre otras funciones. El controlador de dominio mantiene una lista de usuarios y contraseñas creándose un repositorio centralizado de contraseñas evitando así almacenarlas localmente en los equipos de la red. Cuando un usuario desea acceder a un recurso compartido el controlador de dominio entra en acción validando al usuario. Si la autentificación es correcta, el controlador de dominio establecerá una conexión de sesión con los derechos de acceso correspondientes para el recurso y el usuario. En caso contrario, si la autentificación es incorrecta, se denegara el acceso al recurso.

Para la configuración de un controlador de dominio [62] bajo el sistema operativo **Windows Server 2008** será necesario configurar los servicios de resolución de nombres de dominio o **dns** y el servicio de directorio activo o **active directory.**

El primer pasó para la configuración de un controlador de dominio será la instalación y configuración de los servicios DNS y directorio activo. Es necesario, como paso previo a la instalación, configurar una dirección de red estática, la cual será a su vez dirección DNS al ser el servicio a instalar, tal como se puede observar en la Figura 44. La dirección 192.168.1.40 será la dirección IP

asignada al controlador de dominio que a su vez ejercerá de servidor DNS preferido. El firewall, cuya IP es 192.168.1.1, será el servidor DNS secundario, encargado de resolver aquellos nombres que el controlador de dominio no sea capaz. Recordar que el servicio DNS fue también previamente instalado en el propio firewall.

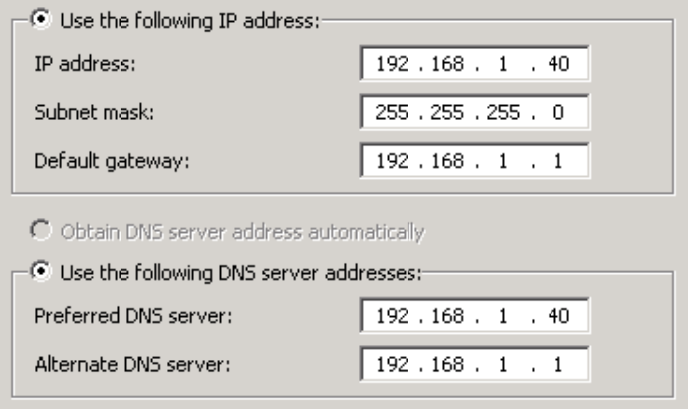

Figura 44. Configuración IP del controlador de dominio.

El siguiente paso en la configuración será promocionar el servidor a controlador de dominio ejecutando el comando **dcpromo.exe** el cual permitirá instalar el servidor DNS y el servicio de directorio activo. La Figura 45 muestra la forma de promocionar el controlador de dominio. La Figura 46 muestra la creación de una zona DNS a administrar, cuyo nombre será **corp.celonis.de.** La Figura 47 muestra la zona creada, finalizada la configuración en la propia promoción del servidor. Esta figura recoge el servicio DNS y el servicio de directorio activo ya en pleno funcionamiento.

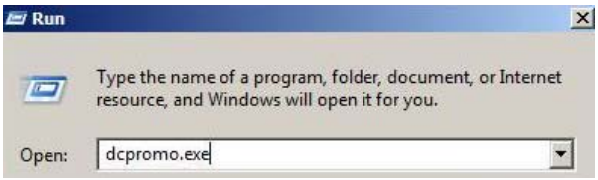

Figura 45. Promoción del controlador de dominio.

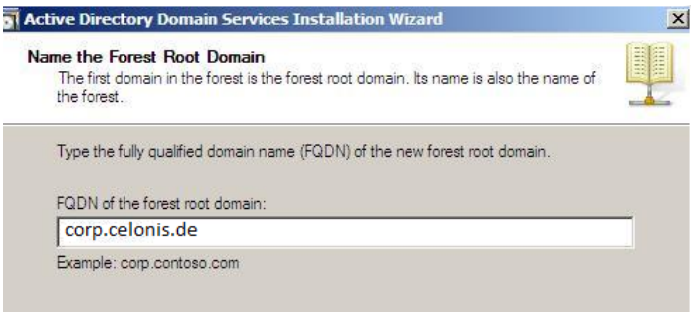

Figura 46. Dominio a gestionar.

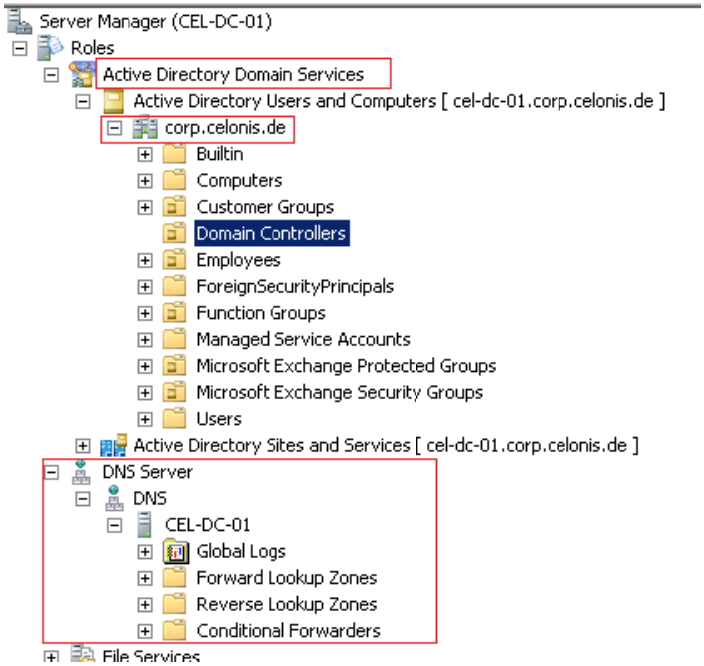

Figura 47. Controlador de Dominio.

La configuración de parámetros adicionales, en la promoción del controlador de dominio es muy intuitiva, omitiéndose en esta memoria el proceso de configuración. El servicio DNS configurado controlará la resolución de equipos y direcciones IP dentro del dominio **corp.celonis.de** a través de las zonas de resolución directa e indirecta configuradas. La Figura 48 muestra registros A de los equipos integrados ya en el controlador de dominio.

CEL-SQL-01	Host (A)	192.168.1.137
CEL-OTRS-01	Host (A)	192.168.1.152
CEL-LP-23	Host (A)	192.168.1.185
(same as parent folder)	Host (A)	192.168.1.40
cel-dc-01	Host (A)	192.168.1.40
CEL-LOTUS-TEST	Host (A)	192.168.1.45
cel-printsrv	Host (A)	192.168.1.50
CEL-APP-01	Host (A)	192.168.1.51
CEL-APP-02	Host (A)	192.168.1.52
CEP-APP-03	Host (A)	192.168.1.53
CEL-APP-04	Host (A)	192.168.1.54
CEL-LP-20	Host (A)	192.168.10.168
ANTONIO	Host (A)	192.168.10.187
cel-lp-21	Host (A)	192.168.10.191
cel-lp-12	Host (A)	192.168.10.197
cel-lp-22	Host (A)	192.168.10.197

Figura 48. Registros de la zona directa del controlador de dominio.

El servicio de directorio activo controlará el acceso de usuarios autentificados en el dominio a recursos y equipos de la red. Los usuarios

mostrados en la Figura 49 tendrán acceso al dominio **corp.celonis.de** así como a sus recursos.

Figura 49. Usuarios con acceso a recursos compartidos.

4.2.4 Backup de servidores virtuales

Para realizar backup de los servidores del entorno KVM se procederá a programar un script controlado por el demonio **cron** que se ejecute regularmente y realice una copia de seguridad de las máquinas en producción.

Se creará además un *log* de ejecución que controle la evolución de las copias de seguridad así como el envió al correo del *log* y de la evolución las copias de seguridad.

Se debe editar un fichero con la extensión **.sh** para su ejecución automatizada. Podemos crear este fichero con el editor de texto **nano** y configurarlo de la siguiente forma:

```
#!/bin/bash
### Scripts to backups VM enviroment
sendEmail -f a.albendea@celonis.de -t a.albendea@celonis.de -s
smtp.1und1.de -xu helpdesk@celonis.de -xp helpbyCel -u "VIRTUAL
MACHINES SUSPEND" -m "The virtual machines was suspended by
backup - DON'T REPLY THIS MAIL" -a /var/log/backupVM.log –v
```

La primera parte del script envía un mail al administrador de la red para avisarle de que se van a suspender las máquinas virtuales para proceder a su copa de seguridad. Se adjunta además el *log* denominado **backupVM.log** que controlará la evolución.

```
echo " Nightly VM Backup Started CEL-DC-01: $(date) " >>
/var/log/backupVM.log

virsh suspend cel-dc-01
```

```
virt-clone --force -o cel-dc-01 -n cel-dc-01-backup -f /backup/cel-dc-01-backup

echo " Nightly VM Backup End CEL-DC-01: $(date) " >> /var/log/backupVM.log
```

La segunda parte del script suspende la máquina virtual **cel-dc-01** (controlador de dominio) clonándola. Se adjunta en el *log* la hora de inicio y fin del backup.

```
echo " RESUME VM ENVIROMENT : $(date)" >> /var/log/backupVM.log

virsh resume cel-dc-01

echo " RESTORE COMPLETE - SCRIPT FINISH 100% : $(date)" >> /var/log/backupVM.log
```

La tercera parte del script restaura la máquina virtual **cel-dc-01** la cuál vuelve a estar operativa y accesible por los usuarios. Se adjunta en el log la hora de inicio y fin de la restauración.

```
cp -f /backup/cel-dc-01-backup /external/.

echo " CEL-DC-01-BACKUP COPIED INSIDE EXTERNAL HARD-DRIVE : $(date)" >> /var/log/backupVM.log
```

La cuarta parte del script realiza una copia del *backup* del controlador de dominio al disco duro externo de la empresa. Se adjunta en el log la hora de inicio y fin de la copia.

```
sendEmail -f a.albendea@celonis.de -t a.albendea@celonis.de -s smtp.1und1.de -xu helpdesk@celonis.de -xp helpbyCel -u "VIRTUAL MACHINES COPIEDIN EXTERNAL HD" -m "The virtual machines have been copied inside EXTERNAL - DON'T REPLY THIS MAIL" -a /var/log/backupVM.log -v
```

La última parte del script manda un mail al administrador de la red adjuntando el log completo del *backup* realizado anteriormente. El demonio **cron** nos permitirá ejecutar el *backup* de forma automatizada, especificándole cuando queremos que se realice la copia de seguridad. Configuraremos cron de tal forma que el backup se realice una vez por semana, cada viernes noche a las 23:55. Para habilitar esta configuración en cron ejecutaremos en siguiente comando en consola y editaremos el fichero de configuración tal como se muestra en la Figura 50.

```
>> sudo crontab –e
```

```
m h  dom mon dow   command
 59 23 * * 5 /home/antonio/scripts/backups_v2.sh
```

Figura 50. Configuración del backup en cron.

CAPÍTULO 5
PRESENTE Y FUTURO DE LA INFRAESTRUCTURA

En este capítulo se dará una visión general del estado actual de la infraestructura tras configurar y desplegar servicios anteriores. Asimismo, también se hace un desarrollo de futuro con posibles alternativas y nuevas ideas.

5.1 Alternativas para la gestión de máquinas virtuales

El sistema de gestión KVM bajo VMM no es único. Existen alternativas de software libre (*open source*) muy interesantes desde el punto de vista funcional y que deberán ser consideradas en el futuro al incluirse funcionalidades actualmente no disponibles. A continuación se describen posibles alternativas compatibles al 100%, con el proyecto desarrollado en los capítulos anteriores.

5.1.1 Proxmox

Proxmox es un software que permite configurar una completa infraestructura de servidores virtuales. Está optimizado para el rendimiento y es de fácil uso.

Se basa es una instalación basada en un hypervisor **baremetal** [63] (software que se ejecuta directamente sobre el hardware del computador). El sistema base se instala sobre una distribución Linux debían.

Proxmox permite diferentes tipos de virtualizaciones:

- **OpenVZ:** para virtualizar servidores Linux mediante contenedores seguros y aislados, tratando a cada servidor de forma independiente. Cada contenedor puede ser reiniciado independientemente y tener acceso con privilegios de root, sus propios usuarios, sus propios procesos, archivos, aplicaciones, configuración del sistema, etc.

- **KVM:** tanto en virtualización del tipo virtualización completa como del tipo paravirtualización.

La gestión del entorno virtual se realiza mediante una interfaz web, sin necesidad de instalar herramientas adicionales. Este entorno nos permite configurar:

- Almacenes de recursos
- Plantillas de sistemas con software preinstalado
- Sistema de backup incorporado
- Clustering con migración en caliente
- Accesso SSL encriptado (https)

Actualmente Proxmox se encuentra en versión 2.1 y dispone de soporte y comunidad oficial, así como de documentación.

5.1.2 Web Virtual Manager

Web Virtual Manager es una interfaz web basada en la librería libvirt para la gestión de máquinas virtuales. Permite crear y configurar máquinas virtuales configurando de forma precisa recursos como por ejemplo número de procesadores o memoria RAM a usar. Se dispone de un visor *VNC* a través de túnel *SSH* como consola gráfica de la máquina virtual. Únicamente trabaja con el hypervisor KVM.

Se caracteriza por ofrecer:

- Host System
  - Uso de CPU y memoria general
  - Gestión de almacenes de recursos
  - Gestión de red
  - Gestión de máquinas virtuales
  - Clonación y snapshots de máquinas virtuales
  - Logs
- Virtual Machine
  - Uso de CPU y memoria local
  - Gestión de imágenes ISO
  - Acceso VNC
  - Reinicio, apagado y forzado de apagado de maquinas virtuales

Actualmente se encuentra en fase beta. Se dispone de soporte vía mail y apenas existe documentación.

5.2 Amazon Cloud

Una de las opciones más interesantes de futuro para Celonis GmbH, es poder aprovechar la capacidad de **cloud computing** para contratar recursos informáticos externos y permitir la transferencia de servidores y servicios virtuales desplegados en la infraestructura local.

Supóngase la contratación de una serie de servicios virtuales a la empresa, donde por capacidad no pueden ser alojados físicamente. Los servicios virtuales deben tener la capacidad de ser portables a un entorno cloud como el de Amazon y accesible a los clientes, dando la sensación a estos últimos de que acceden a los servicios ofertados por la empresa.

Un claro ejemplo son los servicios basados en Windows Server 2008 con servidores SQL en producción. Amazon permite importar máquinas virtuales con sistemas operativos Windows Server 2003 R2 y Windows Server 2008

(Datacenter,Enterprise y Standar). Este primer requisito se cumple utilizando una infraestructura virtual bajo KVM.

Un segundo requisito, es que Amazon permite únicamente trabajar con ciertos formatos de discos duros virtuales. Los formatos VMDK y RAW son aceptados por la plataforma. La plataforma de virtualización KVM permite trabajar con ambos formatos, siendo los sistemas en producción despleglados bajo el segundo formato.

En conclusión, el crecimiento de la plataforma está garantizado al disponer de recursos externos en caso de necesidad compatibles con la infraestructura local y que cumple los requisitos impuestos.

CAPÍTULO 6
VALORACIÓN PERSONAL, CONCLUSIONES Y TRABAJOS FUTUROS

Este capítulo recoge las valoraciones personales y conclusiones adquiridas a lo largo de este proyecto fin de carrera. Durante siete meses he desarrollado este trabajo fuera de mi país, en un nuevo entorno llevo de buenos momentos, y como no, de dificultades superadas a lo largo del camino.

6.1 Valoración Personal

Este proyecto ha sido desarrollado gracias a la beca de prácticas Erasmus en una empresa llamada Celonis GmbH con sede en la ciudad de Múnich, Alemania. Esta empresa, con menos de 2 años de nacimiento, proporciona servicios a empresas tan conocidas como Siemens, Bayer o la Universidad de Múnich TUM.

A través de esta beca, he conseguido adquirir experiencia en una compañía nacional alemana, que me ha permitido experimentar un intercambio cultural y de idiomas. Mediante una entrevista personal, anteriormente a las navidades, fui aceptado como becario en el departamento IT y de sistemas de la compañía.

Un primer obstáculo encontrado a lo largo del camino ha sido el idioma. Mi conocimiento de la lengua alemana es nulo aunque durante siete meses de estancia he conseguido, a nivel personal, entender ciertas conversaciones y aprender vocabulario a nivel personal. Puedo decir, que he adquirido de forma innata el nivel A1, dentro del marco europeo de idiomas. Mediante el inglés, he sido capaz no sólo de comunicarme sino de experimentar una mejoría notable en la soltura del idioma. Ahora soy capaz no sólo de hablar más fluidamente sino que he mejorado la escucha, quizás el aspecto más difícil dentro de los idiomas. A esto han ayudado mis compañeros de empresa, que no dudaban en corregirme el idioma para mejorar. Me habría gustado, quizás, un apoyo económico de la empresa para estudiar alemán o inglés, aunque lo hice, resultó un gasto costoso y la beca precisamente no es que este dotada de un gran sustento económico (280€ al mes, insuficiente en los tiempos actuales).

Otra gran dificultad ha sido adaptarme a la empresa. A diferencia de España, aquí se trabaja un promedio de 2 horas más al día, desmintiendo, según mi opinión personal, la teoría de que los alemanes son productivos en el trabajo. Se tiene por costumbre comer en la empresa, generalmente en la mesa de trabajo, para lo cual se proporcionan 45 minutos pero dónde lo normal es trabajar comiendo. Esto supone, que los empleados de las empresas trabajen de más. Pero reconozco que tiene como ventaja una flexibilidad laboral difícilmente localizable en empresas españolas. La gente acumula horas en vacaciones, donde si sumas los 30 días que por ley corresponden a un trabajador al año, se puede de más días libres aquí que en España.

Un gran aspecto que me ha sorprendido, es la segmentación del mercado laboral. Los titulados universitarios gozan de buena reputación en el mercado laboral alemán y cobran un salario digno, aproximadamente entre 40.000-60.000€ anuales brutos. Quizás un país que valore las titulaciones universitarias merece un gran respeto en mi opinión, puesto que España no es precisamente

un ejemplo a seguir en ese campo, sobretodo en aspecto salarial. Ojala, algún día, los ingenieros o titulados universitarios de España gozemos de ese status que nos hemos ganado con el sudor de nuestra frente, estudiando horas y horas por mejorar.

Mi gran conclusión personal, es un dicho muy popular actualmente en Munich. Se dice que "España forma ingenieros y Alemania los utiliza". Alemania ha encontrado en España su gran mercado laboral.

6.2 Conclusiones y trabajos futuros

A lo largo del desarrollo del proyecto se ha tratado de alcanzar todos los objetivos planteados a su comiendo y presentados en la Sección 1.2, pero como todo proyecto a largo plazo, los objetivos y necesidades pueden sufrir variaciones. En este capítulo se repasan los objetivos iniciales del proyecto para concretar los aspectos que se han desarrollado correctamente y comentar las posibles actuaciones para el futuro.

Un primer objetivo a cumplir fue una reestructuración de la red interna de la empresa. Se modificó la configuración inicial para habilitar las tomas de red (al menos una) del cableado estructurado del edificio para evitar cables entre habitaciones y evitar así posibles accidentes. Además, el desarrollo de este objetivo permitió una mejora en la fiabilidad de la red, caracterizada anteriormente por constantes caídas de red.

No obstante, una cuestión que está pendiente y se podría abordar en el futuro, habilitar en el futuro todas las tomas del cableado estructurado y eliminar la presencia de switches para evitar conexiones en cascada, así como suprimir el switch de la tercera planta que habilita internet al cableado estructurado. Se recomienda una centralita ADSL para este propósito.

El segundo objetivo consistió en adaptar la infraestructura para un crecimiento sostenible. Mediante un firewall se segmentó la red de la compañía en cuatro subredes gestionadas mediante *iptables*. Tanto la red interna como celonis-int permite a los empleados de la empresa trabajar en red y tener acceso a internet (acceso cableado o inalámbrico) y a los servidores virtuales. La red celonis-ext está configurada para personal externo a la empresa pero sin acceso a los recursos internos. El firewall recibió la aceptación de mis superiores y paso las pruebas pertinentes.

Además, se permiten accesos externos mediante certificado de seguridad y palabra clave (*passphrase)*. El firewall de la empresa proporciona una capa de seguridad muy importante, evitando posibles accesos exteriores no deseados. Este objetivo se ha desarrollado completamente y se ha finalizado de forma

satisfactoria. Desde mi casa, podía conectarme a los servidores de la oficina a través de openVPN:

Un tercer objetivo desarrollado fue el despliegue de la infraestructura virtual KVM como centro de testeo y producción. Bajo un servidor Linux Ubuntu, se configuró la infraestructura para el futuro despliegue de servidores de la familia Windows y Linux, tal como se demandaba en la empresa. La configuración desarrollada permitirá gestionar a las maquinas virtuales como si de servidores físicos se tratase.

Este punto se desarrolló de forma satisfactoria aunque en el futuro se sugiere una revisión. Tratar a las maquinas virtuales como equipos físicos de red permite que cualquier ataque a la red pueda repercutir sobre los servidores virtuales. Por ello, se recomienda configurar las máquinas virtuales en una red de área local virtual o Virtual Local Area Network (*VLAN*) con seguridad local basada en *iptables*, ya que se dispone de más de una tarjeta de red.

Otra alternativa más sencilla, es configurar la VLAN a través de un switch que permita este rol, aunque esta técnica incrementaría la inversión a realizar en la propia infraestructura.

El cuarto objetivo a desarrollar fue el despliegue de un controlador de dominio. Este objetivo se desarrolló de forma completa y satisfactoria, permitiendo la integración de todos los equipos de la empresa (tanto sistemas operativos Windows como Linux) en el dominio virtual **corp.celonis.de** y validarse para el acceso a recursos compartidos.

Una de las ventajas del controlador radicó en la creación de grupos de usuarios para la administración de servidores SQL, evitando así administraciones locales del servidor y delegando la seguridad en los accesos al controlador de dominio.

En el futuro se recomienda, si se produce un incremento en el número de usuarios del dominio, desplegar las políticas de seguridad que el controlador de dominio permite.

El quinto objetivo del proyecto fue el despliegue de un conjunto de servicios virtuales. Actualmente se ha desplegado en la infraestructura virtual un servidor de impresión para la gestión de impresoras. Además, se desplegó un servidor de correo electrónico bajo la plataforma Microsoft Exchange. Este servidor estuvo operativo aproximadamente dos semanas pero fue retirado ya que no cubría las necesidades de la empresa. Este servidor requería de un conector para interactuar con el servidor de correo externo contratado a la compañía **1UND1**, pero requería chequear por cada usuario del dominio su

correo personal para comprobar qué correos descargaron o no, lo cual provocaba un enorme retraso en cada comprobación.

Además se desplegó bajo un servidor virtual Linux el servicio virtual OTRS como gestor de tickets de incidencias. Este sistema, únicamente se ha testeado y no se ha puedo en producción todavía. Las pruebas fueron satisfactorias por lo que será el futuro gestor de incidencias de la empresa. Los servidores SQL también desplegados, funcionan correctamente.

El sexto objetivo perseguía el despliegue de una VPN para acceder a los servidores virtuales de forma externa y de manera segura. Este objetivo se ha cubierto correctamente y cualquier empleado de la empresa posee su propio certificado para acceder.

El séptimo objetivo, que cubre la seguridad y automatización de backups ha sido cubierto mediante la programación de un script que facilita las copias de seguridad de las maquinas virtuales. No obstante, se recomienda la migración a una nueva infraestructura (como puede ser proxmox) que permita las copias de seguridad en vivo y no mediante terminal de comando o scripts.

El último objetivo propuesto, la infraestructura software como servicio o SaaS no ha podido desarrollarse debido a que las actuales necesidades de negocio no han evolucionado lo suficiente aun. No obstante la infraestructura está preparada para ejercer de SaaS temporal e incluso se puede migrar al cloud de Amazon para ofrecer nuestro software como servicio.

Bibliografía

[1] Eugenio Velazquez, ¿Qué es la virtualización?, http://www.tecnologiapyme.com/software/que-es-la-virtualizacion 08/01/2009

[2] Red Hat, KVM Wiki, http://www.linux-kvm.org/page/Main_Page 04/02/2009

[3] Wikipedia, Cortafuegos, http://es.wikipedia.org/wiki/Cortafuegos_(inform%C3%A1tica) 27/08/2012

[4] Evi Nemeth, Garth Snyder, Trent R. Hein, Linux Administration Handbook SecondEdition, Prentice Hall, 30/10/2006

[5] Roderick W. Smith, LPIC-2 Linux Professional Institute Certification Study Guide, Sybex.

[6] Roderick W. Smith, LPIC-2 Linux Professional Institute Certification Study Guide, Sybex.

[7] Pello Xabier Altadil Izura, IPTables Manual Práctico

[8] Pablo Neyra Ayuso, Netfilter, http://www.netfilter.org, 1999

[9] Wikipedia, IPtables, http://es.wikipedia.org/wiki/Netfilter/iptables , 28/06/2012

[10] Massachusetts Institute of Technology, http://web.mit.edu, 2002

[11] Evi Nemeth, Garth Snyder, Trent R. Hein, Linux Administration Handbook SecondEdition, Prentice Hall, 30/10/2006

[12] Roderick W. Smith, LPIC-2 Linux Professional Institute Certification Study Guide, Sybex.

[13] Jeff Garzik, http://linux.yyz.us/nsupdate/ , 2008

[14] Factor Evolution, http://www.linuxparatodos.net/ , 2000

[15] Instituto Nacional de Tecnologías Educativas y de Formación del Profesorado, http://www.ite.educacion.es/

[16] Evi Nemeth, Garth Snyder, Trent R. Hein, Linux Administration Handbook SecondEdition, Prentice Hall, 30/10/2006

[17] Roderick W. Smith, LPIC-2 Linux Professional Institute Certification Study Guide, Sybex.

[18] Hosting Windows, http://www.herramientasdns.com/ , 2012

[19] Evi Nemeth, Garth Snyder, Trent R. Hein, Linux Administration Handbook SecondEdition, Prentice Hall, 30/10/2006

[20] Roderick W. Smith, LPIC-2 Linux Professional Institute Certification Study Guide, Sybex.

[21] Wikipedia, IPtables, http://es.wikipedia.org/wiki/Netfilter/iptables , 28/06/2012

[22] Daniel Colenti, Tablas Cadenas y Reglas, http://www.danielcoletti.com.ar/Documentos/Tech/Iptables/iptables/node7.html , 27/06/2003

[23] Mauricio Campiglia, IPTables para Novatos, http://www.cafeconf.org/2007/slides/mauricio_campiglia_iptables_para_novatos.pdf

[24] Daniel Colenti, Tablas Cadenas y Reglas, http://www.danielcoletti.com.ar/Documentos/Tech/Iptables/iptables/node7.html , 27/06/2003

[25] Mauricio Campiglia, IPTables para Novatos, http://www.cafeconf.org/2007/slides/mauricio_campiglia_iptables_para_novatos.pdf

[26] Evi Nemeth, Garth Snyder, Trent R. Hein, Linux Administration Handbook SecondEdition, Prentice Hall, 30/10/2006

[27] Roderick W. Smith, LPIC-2 Linux Professional Institute Certification Study Guide, Sybex.

[28] ProgramoWeb Powered, VPN, http://programoweb.com/72263/definicion-y-tipos-de-vpn/

[29] Universidad de Valencia, VPN, http://www.uv.es/siuv/cas/zxarxa/vpn.htm#quees

[30] Tomas Fernandez Pena, VPN, http://www.ac.usc.es/docencia/ASRII/Tema_4html/node19.html , 28/02/2008

[31] Dante Odín Ramirez Lopez, Cifrado SSL/TSL, http://revista.seguridad.unam.mx/numero-10/el-cifrado-web-ssltls , 03/05/2011

[32] OpenVPN Comunity, SSL/TLS, http://openvpn.net/ , 2002

[33] Juan José Tomás Cánovas, OpenVPN: VPN rápido y seguro, Editorial Académica Española, 2011

[34] Favshare, Certificados Digitales SSL/TSL, http://www.favshare.com/es/pdf/servidores-dedicados/certificados-digitales-TLS-y-SSL.pdf

[35] Luciano Lagassa, OpenVPN, http://ubunlog.com/instala-tu-propio-servidor-vpn-con-openvpn-en-ubuntu-10-04-server/ , 07/09/2012

[36] Yames Yonan, openVPN, http://laurel.datsi.fi.upm.es/~rpons/openvpn_como/ , 2002

[37] Sergio de Luz - Redes Zone, OpenVPN, http://www.redeszone.net/redes/openvpn-conectate-a-cualquier-red-de-forma-segura-mediante-openvpn-manual-para-gnulinux-y-windows-7-32bits-y-64bits-clienteservidor-ssltls/

[38] Andrés J. Diaz , gestión de certificados con openssl, http://stuff.gpul.org/2004_cripto/doc/chuleta_openssl.pdf , Julio 2004

[39] dyndns, http://dyn.com/dns/

[40] Christian Paredes Valdivia, Capa de abstracción del Hardware, http://atilathehun.blogspot.de/2010/01/la-capa-de-abstraccion-de-hardware-o.html , 12/01/2009

[41] Wikipedia, Hypervisor, http://es.wikipedia.org/wiki/Hipervisor , 02/06/2012

[42] VMlogia, Virtualizacion, http://www.vmlogia.com/tiposdev.aspx

[43] Xen, Virtualizacion, http://www.redes-linux.com/manuales/virtualizacion/AdminXen.pdf

[44] Datakeeper, Hypervisor, http://www.datakeeper.es/?p=716 ,

[45] Wikipedia, API, http://es.wikipedia.org/wiki/Interfaz_de_programaci%C3%B3n_de_aplicaciones , 23/08/2012

[46] VMlogia, Virtualizacion, http://www.vmlogia.com/tiposdev.aspx

[47] Xen, Virtualizacion, http://www.redes-linux.com/manuales/virtualizacion/AdminXen.pdf

[48] Xen, Virtualizacion, http://www.redes-linux.com/manuales/virtualizacion/AdminXen.pdf

[49] VMlogia, Virtualizacion, http://www.vmlogia.com/tiposdev.aspx

[50] Xen, Virtualizacion, http://www.redes-linux.com/manuales/virtualizacion/AdminXen.pdf

[51] Ubuntu Comunity, KVM Installation, https://help.ubuntu.com/comunity/KVM/Installation

[52] Red Hat, Virtual Machine Manager (VMM), http://virt-manager.org/

[53] Ubuntu Community, Cifrado, http://doc.ubuntu-es.org/Encriptar_particion

[54] Conocimiento Abierto, Cifrado, http://conocimientoabierto.es/cifrar-particion-disco-duro-externo-linux/197/

[55] Red Hat, Mapeador de dispositivos, https://access.redhat.com/knowledge/docs/es-ES/Red_Hat_Enterprise_Linux/6/html/Logical_Volume_Manager_Administration/device_mapper.html

[56] Wikipedia, Bechmarks, http://es.wikipedia.org/wiki/Benchmark , 05/09/2012

[57] Wikibooks, Formatos de discos duros virtuales, http://en.wikibooks.org/wiki/QEMU/Images , 25/08/2012

[58] Wikipedia, Formato RAW, http://es.wikipedia.org/wiki/.raw , 21/08/2012

[59] Wikipedia, Formato QCOW2, http://en.wikipedia.org/wiki/Qcow#qcow2 , 24/08/2012

[60] Wikipedia, Snapshot , http://es.wikipedia.org/wiki/Copia_instant%C3%A1nea_de_volumen , 23/01/2012

[61] Wikipedia, Compresión Zlib, http://es.wikipedia.org/wiki/Zlib, 27/10/2011

[62] Aprende Informática Conmigo, Controlador de Dominio, http://www.aprendeinformaticaconmigo.com

[63] Wikipedia, Baremetal, http://es.wikipedia.org/wiki/Hipervisor , 02/06/2012

Linux Administration HandBook: **ISBN 0-13-148004-9**
Running Xen: A Hands-On Guide to the Art of Virtualization

Publisher: Prentice Hall
Pub Date: April 10, 2008
Print ISBN-10: 0-13-234966-3
Print ISBN-13: 978-0-13-234966-6
eText ISBN-10: 0-13-207467-2
eText ISBN-13: 978-0-13-207467-4

Fedora 13 Manual de Virtualización

DESCARGA DE SOFTWARE

http://www.microsoft.com/bizspark/

FOROS

http://forums.meulie.net/

Grupo KVM Linkedin