2017

Hardsoftsecurity.es

David

[HARDENING DE NUESTRO CENTRO DE DATOS]

En este documento voy a explicar métodos y servicios para hacer más segura nuestra red. Voy a enseñar como instalar e implementar IDS, monitor de red y de equipos, VPN y servicios de copia de seguridad automatizados.

Contenido

1. INTRODUCCIÓN

El objetivo de este proyecto se basa en implementar servicios de monitoreo y anti-intrusiones en nuestro sistema, tanto para nuestra red como para nuestros servidores Linux y Windows. Todo el sistema que se va a implementar es de software libre, esto reduce los costes.

Cualquier empresa que tenga un centro de procesamiento de datos o una red corporativa que desea monitorizar su red y hacerla lo más segura posible, podrá implementar este conjunto de herramientas sin coste de licencias ni nada.

En este documento se explicarán los conceptos y pasos que se deberán dar para poder tener una red monitorizada y con alta seguridad, esto que quiere decir, que se abarcará desde las contraseñas de los equipos, la cual deberá tener una política que se tiene que cumplir a la hora de introducirlas, hasta el tráfico que se genera de entrada contra nuestra red, este último paso es muy importante ya que se identifica de donde y a donde se dirige todo el tráfico de nuestra red, también se desplegará un IDS para identificar este tráfico a nivel de paquete, así diferenciar todos los paquetes que entran en nuestra red e identificando posibles ataques, escaneo de puertos e incluso ataques de denegación de servicio.

Respecto a los servidores los cuales vamos a proteger, vamos abarcar una gran cantidad de S.O, como pueden ser sistemas operativos de Windows y también sistemas operativos basados en Linux. Esto que quiere decir, que vamos a implementar un endurecimiento de la red que abarque cualquier estructura o sistema operativo que tengamos desplegado en nuestra red profesional.

Hoy día un centro de datos se compone de un sistema de virtualización desplegado en una serie de servidores los cuales forman un cluster con los mismos, esto quiere decir que en estos sistemas de virtualización podemos encontrar cualquier sistema operativo, lo cual es muy importante tener monitorizado y protegido con las respectivas políticas, ya sean S.O basados en Windows o Linux.

Si se desea implementar los métodos y herramientas que se enseñaran en este documento es muy importante llegar crear fases de despliegue, ya sea desde una red montada desde cero o implementándola a una red ya en producción.

Una vez montado todo el sistema de endurecimiento y monitorización de todos los sistemas lo que se intentará conseguir es tenerlo todo monitorizado, es decir todo con alertas por sms, correo electrónico o la que creamos conveniente y todo esto mostrado en una pantalla a tiempo real.

2. JUSTIFICACIÓN

Gracias a mi periodo de prácticas, he llegado a comprender lo importante que es desplegar un sistema de seguridad en nuestras redes corporativas, siendo aún más importante si somos una empresa que brinda servicios a través de internet, ya que este tipo de empresas suelen estar más expuestas a los "ciberdelincuentes".

Gracias al Instituto Nacional de Ciberseguridad (Incibe), podemos llegar a encontrar datos alarmantes respecto a "ciberataques" realizados a empresas en una envergadura increíblemente grande, la cual estos ataques le han costado millones de euros a estas empresas.

Según el Instituto Nacional de Ciberseguridad, en el año 2016 se registró un 130% más de ataques a empresas y particulares, respecto al año 2015, donde las cifras de 2015 eran de 50.000 ataques registrados y 115.000 los ataques registrados en el año 2016. Revisando estos datos, también tenemos que tener en cuenta todos aquellos ataques que no han sido registrados y que sigues siendo anónimos. El principal problema de que estos ataques sigan siendo anónimos es debido a que las empresas que han sufrido dicho ataque deciden no denunciarlo, debido al desprestigio de su marca.

La mayoría de ataques se realizan desde servidores alojados en países extranjeros, principalmente procedentes de países como China, Rusia y Ucrania, la motivación des estos "ciberdelincuentes" es completamente económica, es decir hoy día solo hay algo mucho más caro que el mismísimo oro, y es la información.

Hoy día nos encontramos en la era de la información y estamos en un tiempo donde la información de una empresa es muy valiosa, esta información siempre es buscada por los "ciberdelincuentes", los cuales si consiguen entrar en nuestra red corporativa, llegando a conseguir datos de una base de datos o robando las credenciales de un administrador, podemos presuponer lo peor.

Normalmente si un "ciberdelincuente" llegase a obtener nuestros datos corporativos, podría hacer lo que quisiese con ellos, desde venderlos en el mercado negro e incluso venderlos a la competencia. Otro caso que se podría dar es que se utilizase nuestra red corporativa para realizar otro ataque remoto hacia otra red e incluso montar un nodo tor en nuestros servidores.

Este punto lo dedico a la justificación de porque se debe de implementar un sistema de seguridad en cualquier red, una vez leído esto, creo que es suficiente justificación para implementar las últimas tecnologías en seguridad informática.

3. ¿QUÉ ES EL HARDENING?

Hardening (palabra en inglés que significa endurecimiento) en seguridad informática es el proceso de asegurar un sistema mediante la reducción de vulnerabilidades en el mismo, esto se logra eliminando software, servicios, usuarios, etc; innecesarios en el sistema; así como cerrando puertos que tampoco estén en uso además de muchos otros métodos y técnicas.

Su propósito, entorpecer la labor del atacante y ganar tiempo para poder minimizar las consecuencias de un inminente incidente de seguridad e incluso, en algunos casos, evitar que éste se concrete en su totalidad. Una de las primeras cosas que hay que dejar en claro del Hardening es que no necesariamente logrará forjar equipos "invulnerables". Es importante recordar que, según el modelo de defensa en profundidad, se conseguirá una red más segura o no.

Como conclusión, el Hardening es una ayuda indispensable para ahorrarse bastantes dolores de cabeza por parte de los administradores de sistemas. Entre sus ventajas, se puede contar la disminución por incidentes de seguridad, mejoras en el rendimiento al disminuir niveles de carga inútil en el sistema, una administración más simple y mayor rapidez en la identificación de problemas, ya que muchas de las posibles causas de ellos quedarán descartadas en virtud de las medidas tomadas, y finalmente la posibilidad de poder hacer un seguimiento de los incidentes y en algunos casos identificar el origen de los mismos.

Una vez leído y entendido que es el hardening, lo que vamos a conseguir implementado todos los métodos que explicaremos en los siguientes apartados es endurecer la red, los sistemas operativos y los nodos de transmisión de nuestra red como podrían ser router o switchs.

Es muy importante entender que un buen hardening en nuestra red podría evitar o disminuir el daño que podría hacer un atacante dentro de nuestra red, es decir no es lo mismo tener los sistemas "abajo" que solo tener un único servicio aislado caído, se sufrirán perdidas económicas, pero no serán tan brutales como perder todo nuestro centro de procesamiento de datos.

Como se dice más arriba no intentamos tener una red completamente impenetrable, ya que esto es imposible, ya que las redes están creadas, gestionadas por humanos y nosotros no somos perfectos, pero siempre podemos intentar hacer más difícil que un "ciberdelincuente" le sea más difícil llegar a penetrar nuestros sistemas o tener acceso a nuestro centro de procesamiento de datos.

4. MONITORIZACIÓN DE LA RED

Como hemos comentado anteriormente uno de los aspectos más importantes dentro de la seguridad de una red, es la monitorización de la misma, para poder reconocer ataques de denegación de servicio u otro tipo de ataques, como podrían ser conexiones remotas que consuman más ancho de banda de lo normal y por eso en este apartado vamos a explicar porque se debe implementar un sistema de monitorización de la red.

Esto es muy importante en un entorno empresarial, debido a que si se da un servicio hacia internet es muy bueno tener constancia del tráfico general e individual que está generándose en nuestra red. Es decir si tenemos un sistema de monitorización de red general como los que tienen incorporados los Mikrotik, estos nos podrán mostrar el tráfico generado en tiempo real por cada una de sus interfaces en intervalos diarios, semanales, mensuales y anuales.

¿Qué vamos a poder identificar con este método?

Con este método lo que vamos a conseguir con un simple vistazo es visualizar la banda ancha que se está consumiendo por nuestra red e interfaces, así dándonos datos reales en tiempo real, con esto podemos controlar y limitar la banda ancha de cada interfaz a nuestro criterio. También podemos identificar el uso de nuestros servicios, es decir en caso de tener un FTP al cual se tiene acceso desde internet, podremos ver el tráfico general que se estaría generando en nuestro router.

Después tenemos otro método de monitorización el cual se monitoriza la red pero individualmente, es decir se monitoriza el tráfico generado por los equipos que tenemos montados en la red, este método se consigue implementando un conjunto de protocolos y uso de paquetes que se consigue implementando el servicio Netflow y en mi caso contra un Mikrotik.

¿Qué vamos a conseguir identificar con este método?

Con este método en concreto es muy recomendable, ya que uno de los principales problemas en un centro de procesamiento de datos, donde se virtualizan cientos de máquinas es casi imposible a simple vista identificar que máquina está colapsando la red enviando ciertos paquetes de red.

Con este método podemos identificar cualquier máquina que genere tráfico en la red que sea anómalo. Cabe decir que también es una buena práctica tener ambos métodos implementados en la misma red, ya que en el momento que en el tráfico general se detecte más ancho de banda de lo normal podremos dirigirnos a ver el tráfico individual de cada máquina y así localizar la máquina que esté dando problemas e identificar y solucionar el problema que lo provoca.

5. AUDITORÍAS DE SEGURIDAD

Uniendo las técnicas de auditorías de seguridad y de monitorización de redes, llegaremos a prever posibles vectores de ataque que podrían tomar los ciberdelincuentes y en estos casos encontrar patrones de ataque en la monitorización de la red.

¿En qué consiste una auditoría de seguridad?

Una auditoria de seguridad o penetración de sistema, se basa en seguir una metodología para buscar puntos débiles en nuestra red y sistemas que se encuentran alojados en nuestro centro de procesamiento de datos.

A día de hoy la principal metodología que se utiliza para intentos de penetración de sistemas se basa en recogida de información de internet, información física la cual nos proporciona escaneado los servicios accesibles desde internet de nuestro centro de procesamiento de datos, interpretación de dicha información en busca de posibles vulnerabilidades e incluso servicios configurados erróneamente, una vez localizados estos posibles puntos de entrada se procede a la explotación de estas posibles vulnerabilidades, en caso de conseguir acceso mediante alguna vulnerabilidad, se llega a la post-explotación, esto quiere decir que se intentará ver hasta dónde podemos llegar explotando esta vulnerabilidad, una vez realizado y descartado toda explotación de vulnerabilidad se genera un documento el cual se le da al cliente, donde se especifica metodología, software, herramientas utilizadas y un reporte con las vulnerabilidades encontradas con la solución que deberá implementar el administrador de la red en cuestión.

Una vez entendido en que consiste una auditoría de seguridad, en mí caso el administrador de la red, el de sistemas y el auditor de seguridad es la misma persona, la gran ventaja de esto es que se podrán realizar auditorías de seguridad regularmente, así siempre estando a la última respecto a la seguridad de la red, esto consiste en aplicar parches y actualizaciones a todo software vulnerable que se encuentre.

Muy importante para un administrador de red nunca descuidar ningún objeto de la red ya que hasta el fallo más minúsculo podría hacer que la red fuese un quebradero de cabeza para solucionar una intrusión de un atacante.

¿Qué se consigue con esto?

Con la auditorias de seguridad te cercioras de que tienes todo el software, servicios y la red lo más actualizada posible y con los parches de seguridad más recientes, haciendo más difícil el acceso de un atacante a la red de nuestro centro de procesamiento de datos.

6. ¿QUÉ ES UN SISTEMA DE DETECCIÓN DE INTRUSIONES?

Estas herramientas son muy buenas cuando se implementan con un sistema de monitorización de red, ya que si detectamos una subida o bajada anormal podremos comprobar el tráfico más específico con este sistema, por eso explico cómo funciona este sistema.

Un sistema de detección de intrusiones (o IDS de sus siglas en inglés Intrusion Detection System) es un programa de detección de accesos no autorizados a un computador o a una red.

El IDS suele tener sensores virtuales (por ejemplo, un sniffer de red) con los que el núcleo del IDS puede obtener datos externos (generalmente sobre el tráfico de red). El IDS detecta, gracias a dichos sensores, las anomalías que pueden ser indicio de la presencia de ataques y falsas alarmas.

El funcionamiento de estas herramientas se basa en el análisis pormenorizado del tráfico de red, el cual al entrar al analizador es comparado con firmas de ataques conocidos, o comportamientos sospechosos, como puede ser el escaneo de puertos, paquetes malformados, etc. El IDS no sólo analiza qué tipo de tráfico es, sino que también revisa el contenido y su comportamiento.

Normalmente esta herramienta se integra con un firewall. El detector de intrusos es incapaz de detener los ataques por sí solo, excepto los que trabajan conjuntamente en un dispositivo de puerta de enlace con funcionalidad de firewall, convirtiéndose en una herramienta muy poderosa ya que se une la inteligencia del IDS y el poder de bloqueo del firewall, al ser el punto donde forzosamente deben pasar los paquetes y pueden ser bloqueados antes de penetrar en la red.

Los IDS suelen disponer de una base de datos de “firmas” de ataques conocidos.

Dichas firmas permiten al IDS distinguir entre el uso normal del PC y el uso fraudulento, y/o entre el tráfico normal de la red y el tráfico que puede ser resultado de un ataque o intento del mismo.

Existen dos tipos de sistemas de detección de intrusos:

HIDS (HostIDS): el principio de funcionamiento de un HIDS, depende del éxito de los intrusos, que generalmente dejaran rastros de sus actividades en el equipo atacado, cuando intentan adueñarse del mismo, con propósito de llevar a cabo otras actividades. El HIDS intenta detectar tales modificaciones en el equipo afectado, y hacer un reporte de sus conclusiones.

NIDS (NetworkIDS): un IDS basado en red, detectando ataques a todo el segmento de la red. Su interfaz debe funcionar en modo promiscuo capturando así todo el tráfico de la red.

7. ¿QUÉ ES UNA VPN?

Como hemos comentado anteriormente, para tener conexiones remotas más seguras se ha pensado implementar VPN's, ya que con esto descartaremos que los atacantes vean nuestro tráfico de red.

Una red privada virtual (RPV), en inglés: Virtual Private Network (VPN), es una tecnología de red de computadoras que permite una extensión segura de la red de área local (LAN) sobre una red pública o no controlada como Internet. Permite que la computadora en la red envíe y reciba datos sobre redes compartidas o públicas como si fuera una red privada con toda la funcionalidad, seguridad y políticas de gestión de una red privada. Esto se realiza estableciendo una conexión virtual punto a punto mediante el uso de conexiones dedicadas, cifrado o la combinación de ambos métodos.

Ejemplos comunes son la posibilidad de conectar dos o más sucursales de una empresa utilizando como vínculo Internet, permitir a los miembros del equipo de soporte técnico la conexión desde su casa al centro de cómputo, o que un usuario pueda acceder a su equipo doméstico desde un sitio remoto, como por ejemplo un hotel. Todo ello utilizando la infraestructura de Internet.

La conexión VPN a través de Internet es técnicamente una unión wide area network (WAN) entre los sitios pero al usuario le parece como si fuera un enlace privado— de allí la designación "virtual private network".

Tipos de VPN:

VPN de acceso remoto:

Es quizás el modelo más usado actualmente, y consiste en usuarios o proveedores que se conectan con la empresa desde sitios remotos (oficinas comerciales, domicilios, hoteles, aviones preparados, etcétera) utilizando Internet como vínculo de acceso. Una vez autenticados tienen un nivel de acceso muy similar al que tienen en la red local de la empresa. Muchas empresas han reemplazado con esta tecnología su infraestructura dial-up (módems y líneas telefónicas).

VPN punto a punto.

Este esquema se utiliza para conectar oficinas remotas con la sede central de la organización. El servidor VPN, que posee un vínculo permanente a Internet, acepta las conexiones vía Internet provenientes de los sitios y establece el túnel VPN. Los servidores de las sucursales se conectan a Internet utilizando los servicios de su proveedor local de Internet, típicamente mediante conexiones de banda ancha. Esto permite eliminar los costosos vínculos punto a punto tradicionales (realizados comúnmente mediante conexiones de cable físicas entre los nodos), sobre todo en las comunicaciones internacionales. Es más común el siguiente punto, también llamado tecnología de túnel o tunneling).

8. POSIBLES ALTERNATIVAS PARA EL HARDENING

Para el monitoreo de la red se han estudiado las siguientes opciones:

-GlassWire (Gratuito)

-NetFlow sobre Mikrotik (Gratuito).

- Graph de Mikrotik (Gratuito).

Para el monitoreo de las máquinas físicas:

-PandoraFMS (Gratuito).

-Nagios (Gratuito).

Opciones elegidas para IDS:

-Snort (Gratuito).

Opciones para VPN:

-VPN L2TP\IPSec Road Warrior sobre Mikrotik (Gratuito).

-VPN L2TP\IPSec Site to Site sobre Mikrotik (Gratuito).

Opciones para el control de acceso:

-Apache guacamole (gratuito).

Opciones copias de seguridad:

-rsync (Gratuito).

-Fbackup (Gratuito).

Endurecimiento de servicios.

El sistema se compondrá de un conjunto de servicios, donde cada servicio estará instalado en un servidor, para hacer más fácil el seguimiento de nuestra red. Este sistema se compone de Snort para el monitoreo de la red, PandoraFMS para monitorizar los equipos, VPN road warrior para el acceso remoto, VPN punto a punto para unir dos redes entre sí, copias de seguridad automáticas, actualizaciones automáticas, monitoreo de la red WAN mediante graph de Mikrotik, Mikrotik con Netflow para monitorizar tráfico de los equipos, redundancia con VRRP, multiples scripts y políticas para la seguridad de los equipos.

Con este conjunto de herramientas lo que vamos a conseguir es un control de la red casi completo y en conjunto tendremos un centro de procesamiento de datos asegurado decentemente.

9. SOLUCIONES ELEGIDAS

Para satisfacer las necesidades de nuestra red, respecto a la seguridad de la misma, es decir cubrir la monitorización de la red, sistema IDS, monitorización de sistemas, conexiones remotas seguras y el endurecimiento de servicios se ha optado por las siguientes herramientas, ya que implementando todas estas entre sí, se llega a tener una red muy sólida:

a. **GRAPH MIKROTIK:** Herramienta incorporada en RouterOS el sistema operativo incorporado con los routers Mikrotik, esta herramienta lo que hace es registrar todo el tráfico que entra y sale de todas las interfaces de red del nuestro router mikrotik, posteriormente el propio router Mikrotik, habilitando una interfaz web en la propia configuración del Mikrotik, este nos mostrará todo el tráfico generado mediante gráficas muy detalladas de todo el tráfico que pasen por las interfaces.
b. **NETFLOW Y MIKROTIK:** NetFlow es un protocolo de red desarrollado por Cisco Systems para recolectar información sobre tráfico IP. Netflow se ha convertido en un estándar de la industria para monitorización de tráfico de red, y actualmente está soportado para varias plataformas e implementando este protocolo con un router mikrotik conseguimos una monitorización completa e individual de todos los equipos.
c. **PANDORAFMS:** Pandora FMS es un software de monitorización para gestión de infraestructura TI. Esto incluye equipamiento de red, servidores Windows y Unix, infraestructura virtualizada y todo tipo de aplicaciones. Pandora FMS tiene multitud de funcionalidades, lo cual lo convierte en un software de nueva generación que cubre todos los aspectos de monitorización necesarios en una red corporativa.
d. **SNORT:** Snort es un sniffer de paquetes y un detector de intrusos basado en red (se monitoriza todo un dominio de colisión). Es un software muy flexible que ofrece capacidades de almacenamiento de sus bitácoras tanto en archivos de texto como en bases de datos abiertas como lo es MySQL. Implementa un motor de detección de ataques y escaneo de puertos que permite registrar, alertar y responder ante cualquier anomalía previamente definida. Así mismo existen herramientas de terceros para mostrar informes en tiempo real (ACID) o para convertirlo en un Sistema Detector y Preventor de Intrusos (IDS).
e. **VPN MIKROTIK:** Se ha utilizado las opciones de creación de VPN que nos brinda RouterOS, tanto para VPN Road Warrior como para VPN site to site.
f. **APACHE GUACAMOLE:** Apache guacamole es un servicio que nos permite gestionar cualquier conexión remota como ssh, RDP, VNC o cualquier otro tipo de protocolo de conexión remota mediante un navegador y cuentas de usuario, así gestionando mucho mejor las conexiones y controlando de mejor manera quien se conecta y cuánto tiempo se mantienen en la sesión, llevando un control de acceso muy alto y restringido.

g. **RSYNC:** rsync es una aplicación libre para sistemas de tipo Unix y Microsoft Windows que ofrece transmisión eficiente de datos incrementales, que opera también con datos comprimidos y cifrados. Mediante una técnica de delta encoding, permite sincronizar archivos y directorios entre dos máquinas de una red o entre dos ubicaciones en una misma máquina, minimizando el volumen de datos transferidos. Una característica importante de rsync no encontrada en la mayoría de programas o protocolos es que la copia toma lugar con sólo una transmisión en cada dirección. Rsync puede copiar o mostrar directorios contenidos y copia de archivos, opcionalmente usando compresión y recursión.

h. **FBACKUP:** Fbackup es un software gratuito que nos permite realizar copias de seguridad de los archivos de sistemas operativos basados en Windows, esta herramienta tiene un potencial bastante bueno y es de uso gratuito como comercial y personal.

i. **HARDENING DE SISTEMAS:** Lo que se procederá en este punto es realizar el endurecimiento de los servicios, es decir por ejemplo los servicios más utilizados que serían mysql, apache, php, etc. En este punto consistirá en endurecer estos servicios para hacer la tarea de los ciberdelincuentes más difícil a la hora de encontrar una vulnerabilidad o una mala configuración de los servicios.

j. **FAIL2BAN:** Fail2ban es una herramienta que observa los intentos de login de variados servicios, tales como SSH, FTP, SMTP, HTTP, entre otros; y si encuentra intentos de login fallidos una y otra vez desde una misma IP, fail2ban rechazará estos intentos de login bloqueando con reglas de iptables a esas IPs que estaban intentando.

k. **DDoSDEFLATE:** Es un script diseñado en Shell script que nos permite mitigar ataques de denegación de servicio, lo que hace este script es comprobar las conexiones existentes, para posteriormente comprobar si existen irregularidades en las conexiones, es decir si alguna conexión excede el límite de conexiones establecido en el script, en caso de exceder este límite DDoSDeflate se encargara de bloquear la dirección IP de la que proviene el ataque mediante iptables.

Gracias a estas herramientas conseguiremos una red más segura donde podremos tener un monitor con todas las herramientas de monitorización, es decir graphs, pandorafms, pnrg de netflow y snort, todas estas herramientas funcionan a tiempo real, también fail2ban y ddosdeflate, que bloquearan intentos de login y ataques de denegación de servicio.

Y gracias a apache guacamole podremos tener conexión de acceso remoto seguro, también hay que decir que gracias a el endurecimiento de los servicios tendremos menos problemas respecto a malas configuraciones respecto a ellos y por último pero no menos importante tenemos las copias de seguridad de nuestras máquinas y datos que se realizaran con rsync y fbackup.

10. VENTAJAS Y DESVENTAJAS

a. GRAPHS MIKROTIK

VENTAJAS	DESVENTAJAS
Fácil configuración	Información genérica
Información del tráfico rápida	Para entender la información hace falta tener experiencia
Información fiable	A veces los gráficos generados se pierden si el router se reinicia.o se apaga
Información dividida en tiempo	Los gráficos suelen ser liosos.
Información en tiempo real	

b. NETFLOW Y MIKROTIK

VENTAJAS	DESVENTAJAS
Información fiable	Complicado de implementar por primera vez
Información individual de cada equipo	Es una aplicación cliente servidor
Auto descubrimiento de la red	Hay que activar configuraciones en mikrotik
Interfaz web clara y legible	Si no se tiene ningún conocimiento no se podría implementar.
Según la velocidad de la red, se representa en una unidad de velocidad	Las gráficas de entrada y salida de información tendrían que separarlas.
Información dividida en tiempo	Si no se tiene experiencia no se entedará

c. PANDORAFMS

VENTAJAS	DESVENTAJAS
Fácil de implementar.	Hay que tener un conocimiento previo
Sistema operativo óptimo para pandora	No es fácil de configurar las alertas
Monitorización de cualquier hardware	No es fácil de configurar la monitorización de algunos equipos
Monitorización de cualquier S.O	El intervalo de monitorización óptimo es de 5 minutos
Personalizar módulos de monitorización	Interfaz web no tiene por defecto el refresco del panel.

d. FAIL2BAN

VENTAJAS	DESVENTAJAS
Fácil de implementar	No es multiplataforma
Bloqueo efectivo	A veces bloquea ips amigables

e. DDOSDEFLATE

VENTAJAS	DESVENTAJAS
Fácil de implementar	No es multiplataforma
Bloqueo efectivo	

f. SNORT

VENTAJAS	DESVENTAJAS
Fácil combinación con una interfaz web.	Complejo de desplegar
Interfaz web snorby muy agradable	A veces una vez desplegado no funciona bien
Información muy fiable	Hay que tenerlo debidamente configurado para su óptimo funcionamiento
Configuración de reglas de control de tráfico	
Información a tiempo real	
Información fácil de interpretar	

g. VPN MIKROTIK

VENTAJAS	DESVENTAJAS
Fácil de implementar	Sin previo conocimiento es difícil de implementar
Fácil de usar	Hace falta tener conocimientos de redes para el enrutamiento
Log de mikrotik para llevar un orden de conexiones	
Creación de cuentas para llevar el control	
Se puede combinar con certificados para su conexión	
Desde cualquier sistema operativo se puede conectar	

h. APACHE GUACAMOLE

VENTAJAS	DESVENTAJAS
Fácil de instalar	
Fácil de gestionar	
Web muy amigable	
Fácil de gestionar conexiones remotas	

i. RSYNC

VENTAJAS	DESVENTAJAS
Fácil de usar	No es multiplataforma
Facilidad de hacer copias de seguridad	Solo para linux
Uso gratuito	Para automatizarlo hay que hacer una tarea

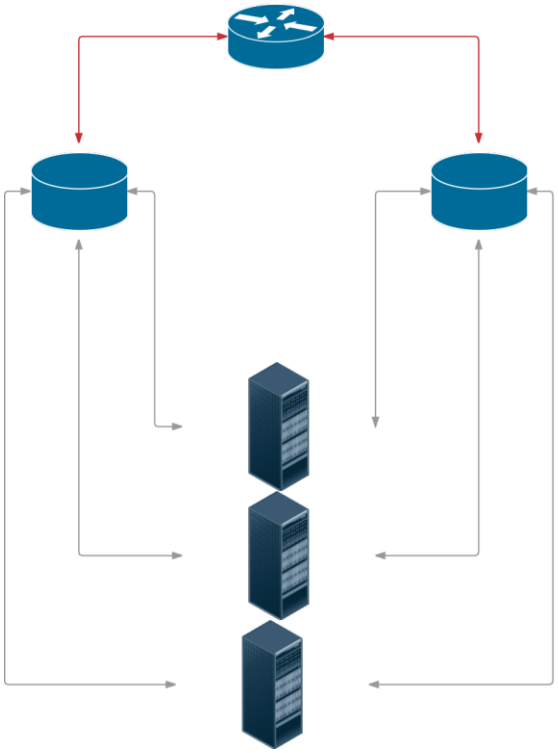
j. FBACKUP

VENTAJAS	DESVENTAJAS
Fácil de usar	No es multiplataforma
Interfaz agradable	Solo para windows
Copias de seguridad faciles	

11. DIAGRAMA DE RED

A continuación se muestra un diagrama de red definiendo como se encuentra actualmente la red:

DIAGRAMA DE RED



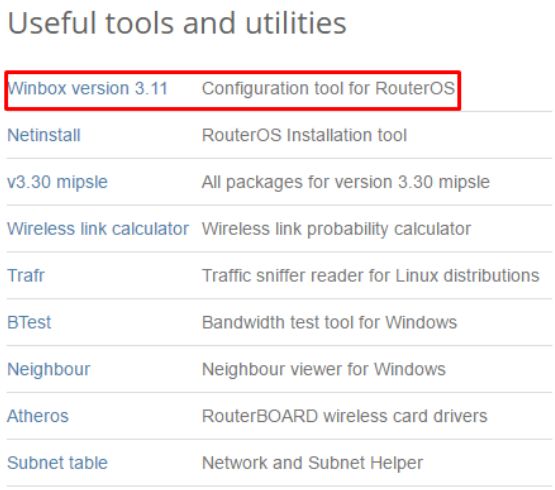
Como podemos ver tenemos 1 router, que se conecta a dos switches y de estos se conectan a los servidores, así creando redundancia en la conectividad de los servidores. Como podemos identificar tenemos un punto de fallo, este punto se hablará en las mejoras de la red más adelante. La conexión de estos equipos es muy simple, se debe de etiquetar todo el cableado como en la siguiente imagen y dejar la red física lo más organizada posible:

12. IMPLEMENTACIÓN DE VPN

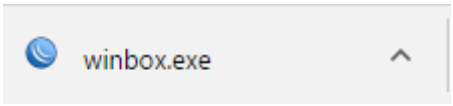
Los primeros pasos que vamos a realizar son las configuraciones básicas de nuestro mikrotik.

a. CONFIGURACIÓN BÁSICA MIKROTIK

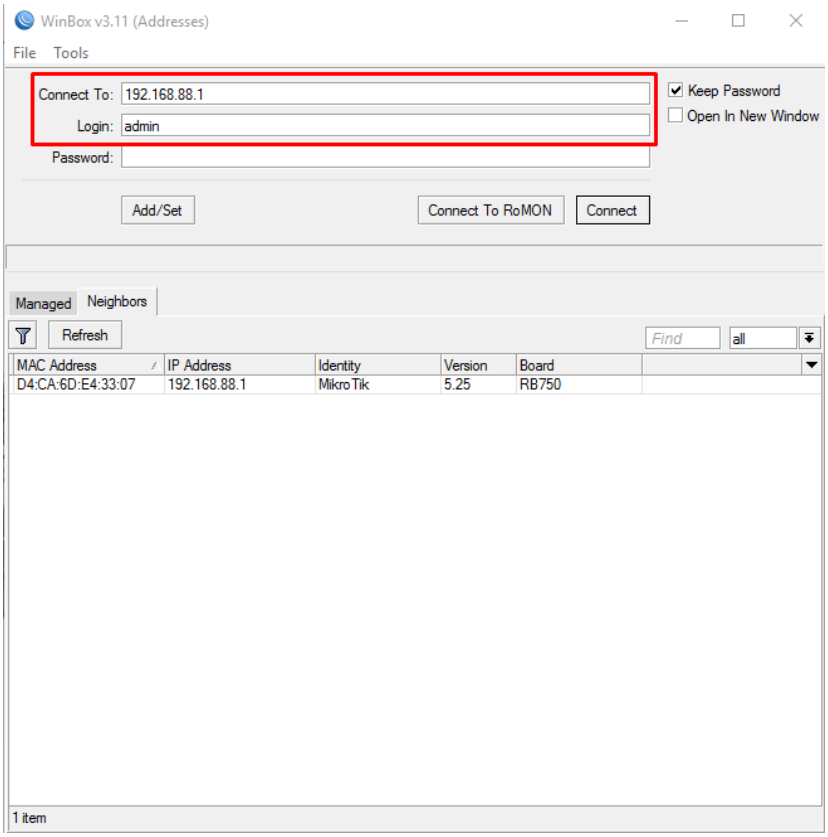
Para la configuración básica de los routers mikrotik vamos a utilizar la herramienta de configuración gráfica que nos ofrece mikrotik llamada "Winbox", donde se puede descargar en: https://download2.mikrotik.com/routeros/winbox/3.11/winbox.exe



Una vez descargada la aplicación la ejecutamos para poder conectarnos a nuestro router:



Al ejecutar la aplicación se nos abrirá un interfaz gráfica donde tendremos que poner la dirección ip de nuestro router mikrotik, por defecto vienen sin contraseña y con la dirección 192.168.88.1 por defecto:

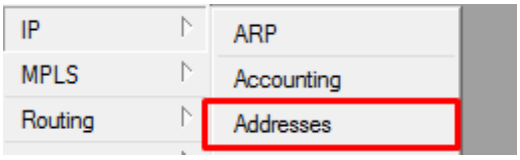


Tras conectarnos a el router mikrotik se nos presentará una interfaz gráfica donde tendremos todas las opciones necesarias para poder configurar todo lo necesario respecto a nuestra red.

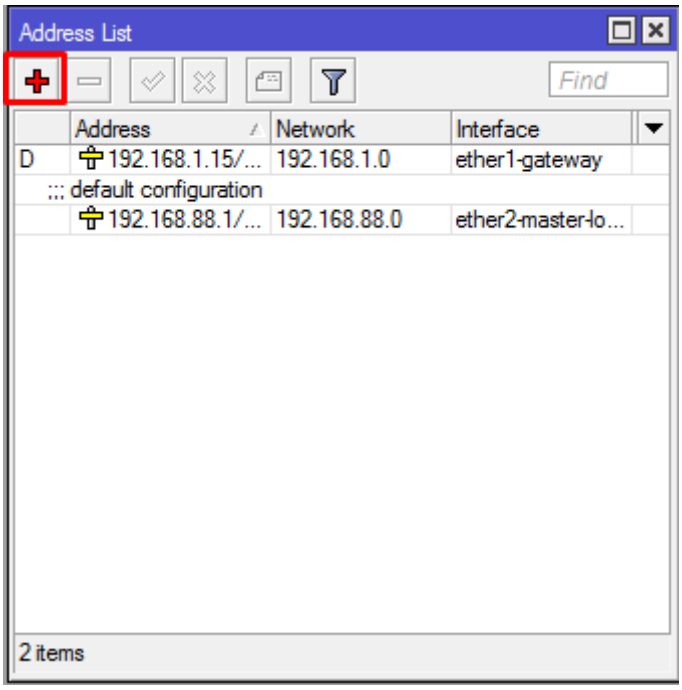
b. CONFIGURACIÓN DE RED LOCAL CPD

Lo primero que vamos a configurar en el ROUTER 1 del CPD que será uno de los routerboard de la empresa, será la dirección IP y el DHCP hacia la red local, comenzamos con la dirección de la red interna del ROUTER 1:

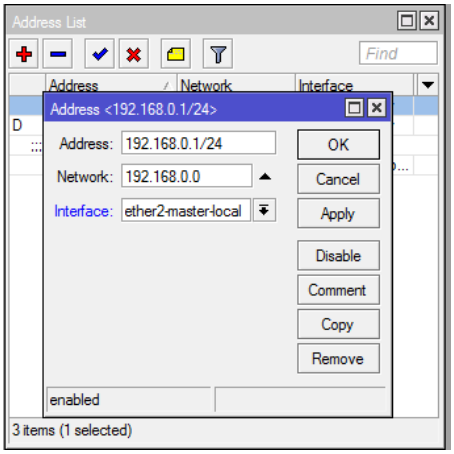
Nos dirigimos a IP → Addresses:



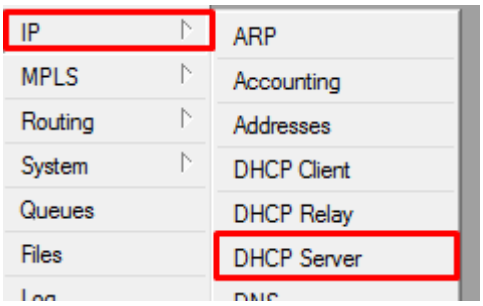
Nos aparecerá una ventana donde podremos añadir la dirección IP de nuestra interfaz:



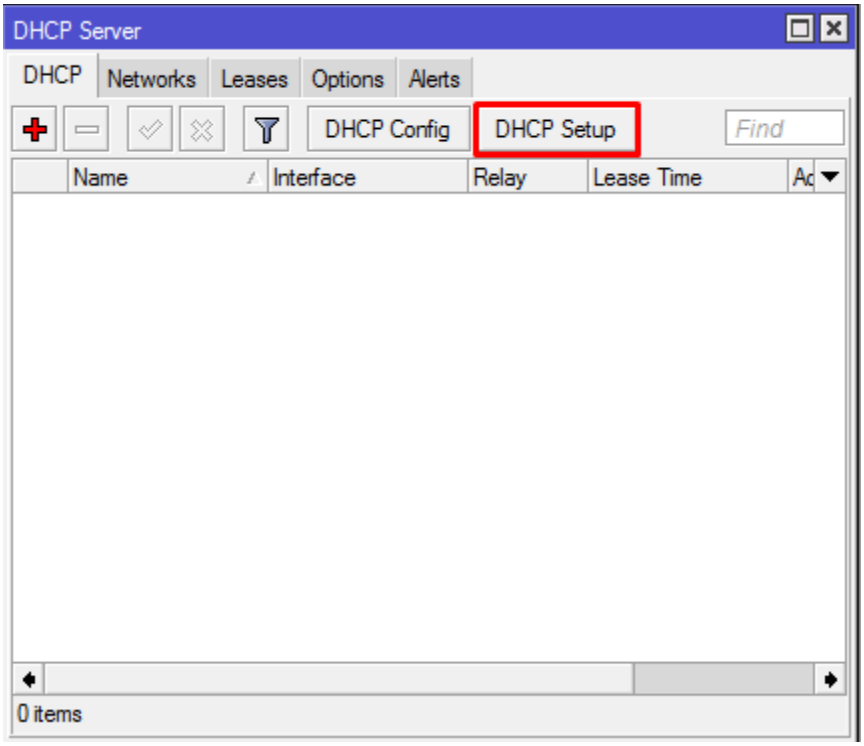
Tras darle en la interfaz a añadir, nos saldrá una ventana donde podremos configurar la interfaz:



El siguiente paso es establecer el servicio de DHCP, para esto nos dirigiremos a:

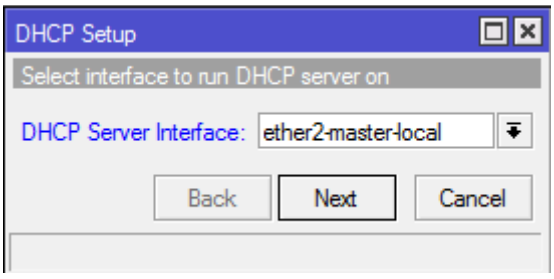


Una vez dentro de la interfaz de configuración pulsaremos en DHCP Setup:

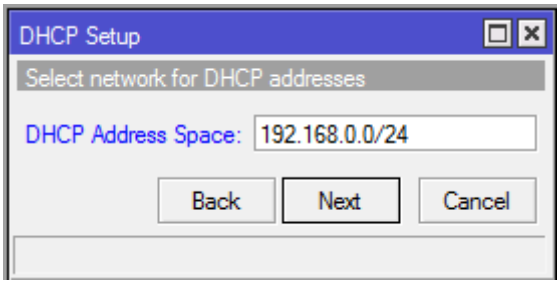


Una vez pulsado en DHCP Setup, se nos abrirá un asistente de configuración para configurar nuestro DHCP:

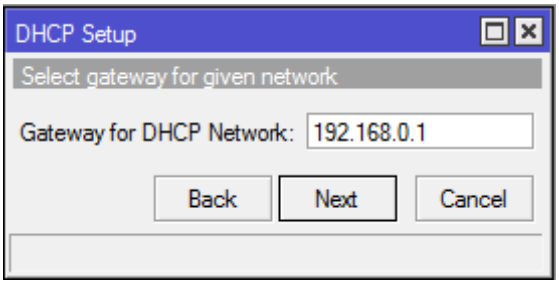
La primera opción se nos pide donde estará a la escucha nuestro servidor DHCP:



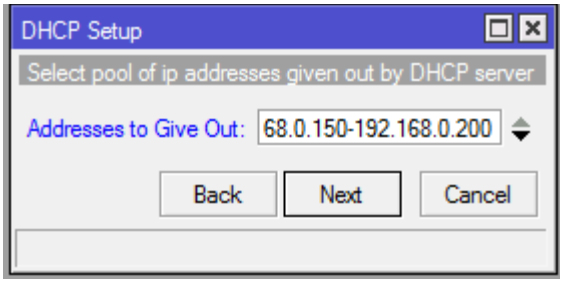
El siguiente paso se nos pide la dirección de IP de la red:



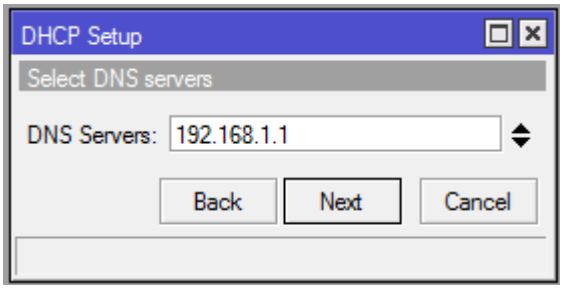
Lo siguiente es determinar la puerta de enlace que irá por defecto en las peticiones DHCP:



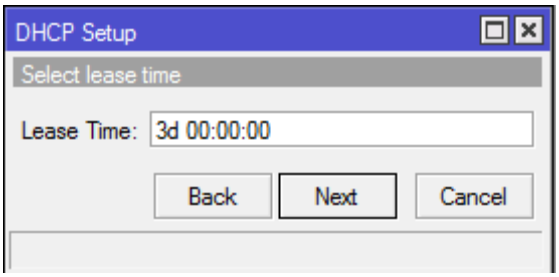
El siguiente paso determinamos el rango de direcciones IP DHCP:



Seleccionamos los DNS por defecto:



Finalmente el tiempo de cesión de las direcciones:

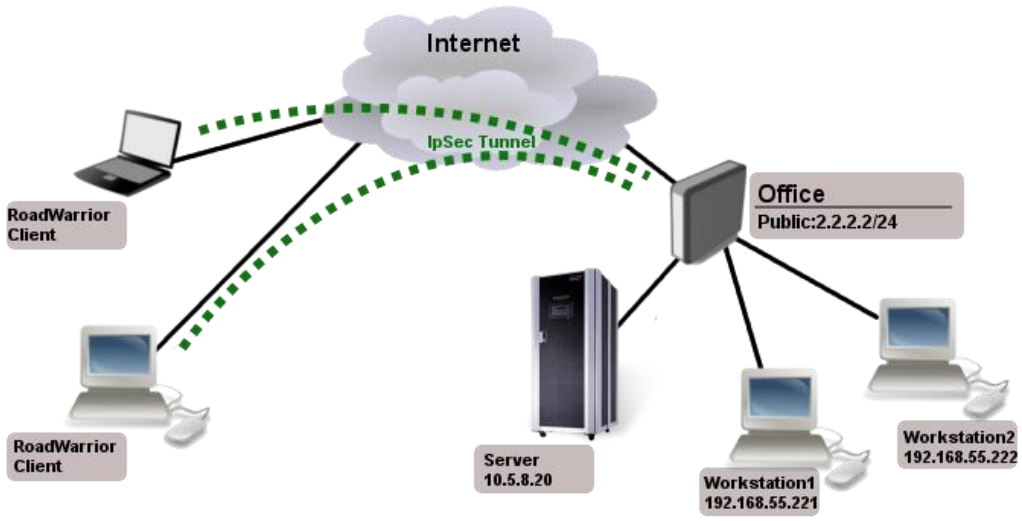


Aquí vemos como ha asignado la dirección IP del DHCP:

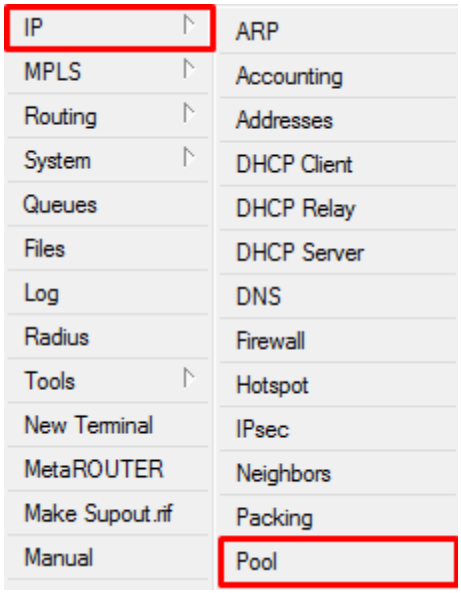
```
Adaptador de Ethernet Ethernet 2:

   Sufijo DNS específico para la conexión. . :
   Vínculo: dirección IPv6 local. . . : fe80::2842:4a4e:4c40:6f82%9
   Dirección IPv4. . . . . . . . . . . . . . : 192.168.0.200
   Máscara de subred . . . . . . . . . . . . : 255.255.255.0
   Puerta de enlace predeterminada . . . . . : 192.168.0.1
```

c. CONFIGURAR VPN ROAD WARRIOR L2TP/IPSEC

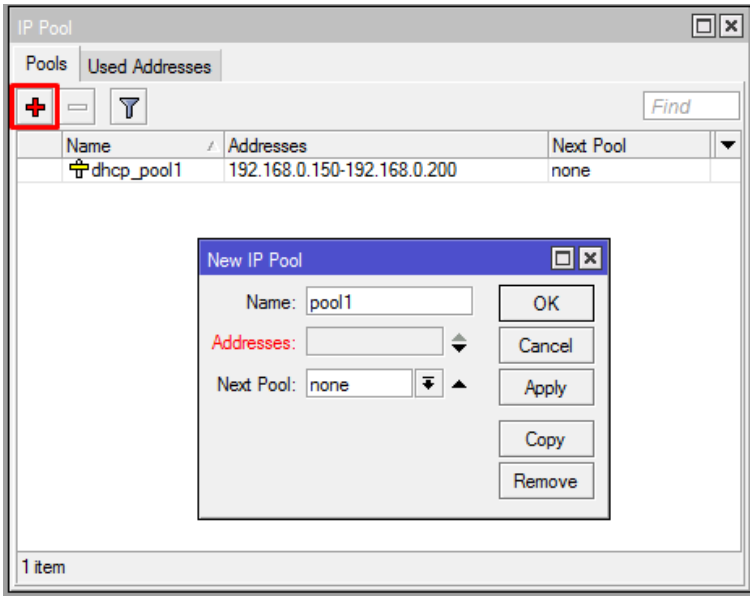
He decidido optar por esta VPN debido a que en múltiples ocasiones hemos necesitado tener acceso remoto a la red del CPD estando fuera de nuestras instalaciones, es debido a esto que hemos elegido este método para poder conectarnos y hacer un mantenimiento o incluso llegar hacer la configuración de algún servicio sin tener que estar en las instalaciones, sería suficiente con tener un dispositivo capaz de conectarse a través de una VPN. A continuación se muestra como se crearía esta VPN sobre un router Mikrotik.



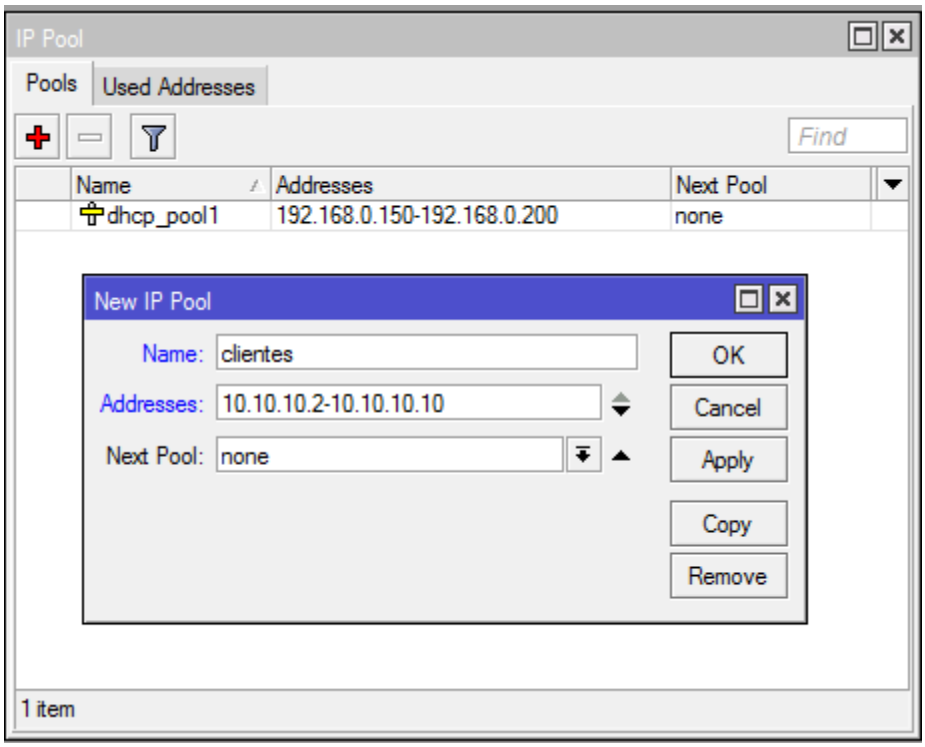
El primer paso para poder configurar una VPN road warrior es crear un pool de direcciones para las conexiones entrantes, para esto nos iremos a:



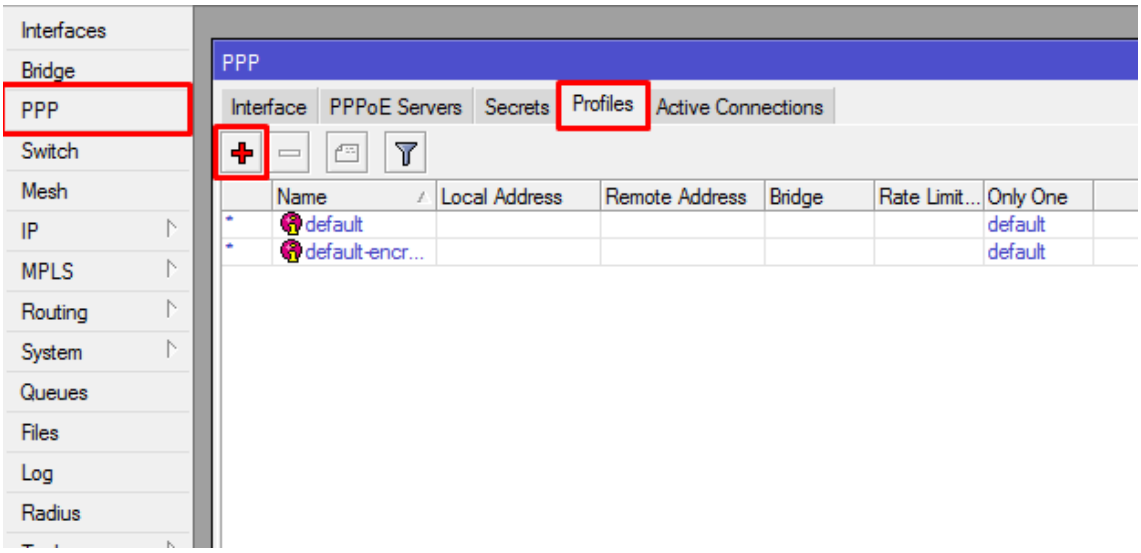
Se nos abrirá una interfaz de configuración donde añadiremos las direcciones:



Una vez configurado quedará de la siguiente manera, se ha configurado para que reciba 9 conexiones simultáneas:

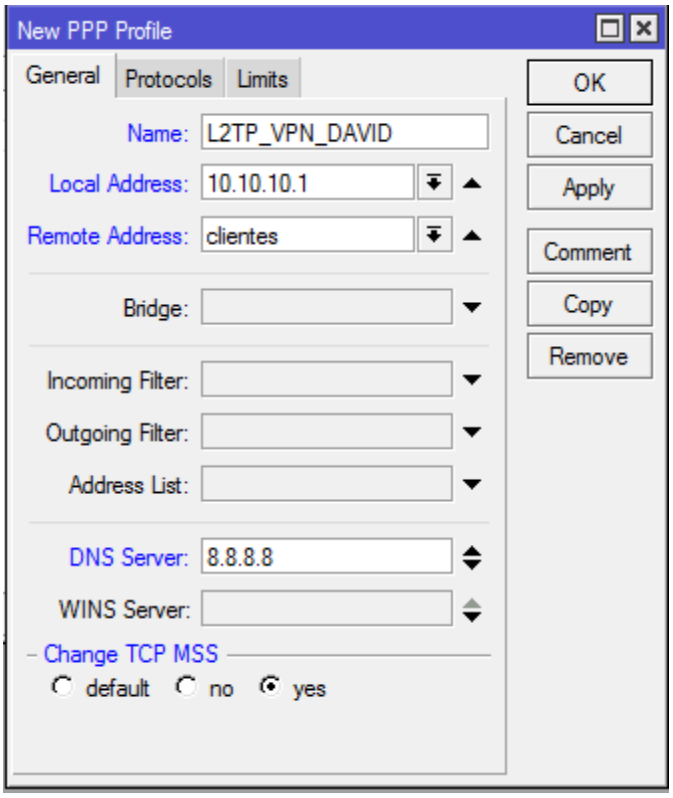


El siguiente paso es entrar en el menú PPP, y entramos en la pestaña Profiles:

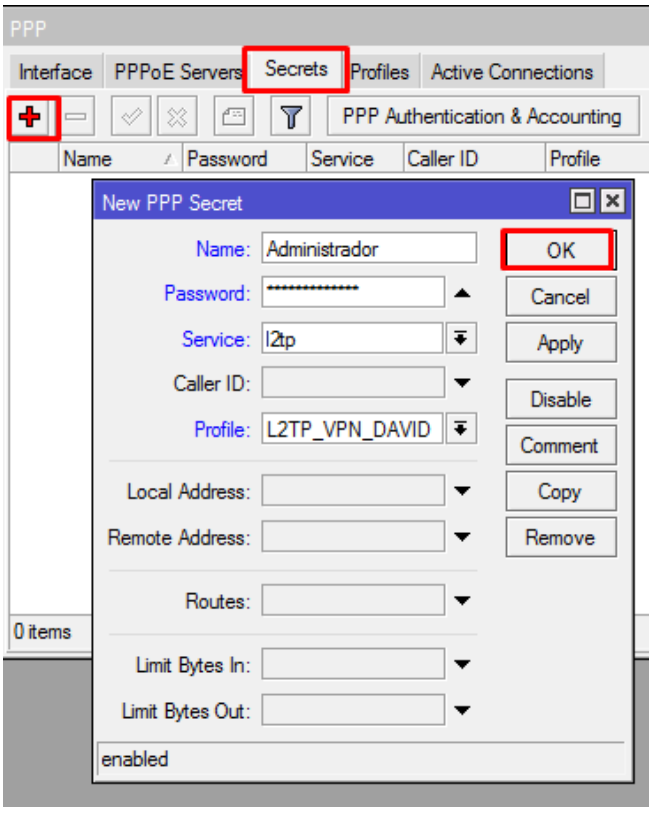


Una vez dentro de la interfaz, le damos a añadir para poder configurar la conexión PPP.

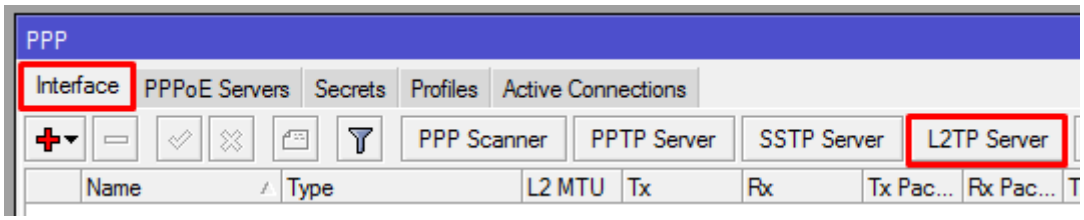
Una vez dentro configuramos las opciones de la siguiente manera:



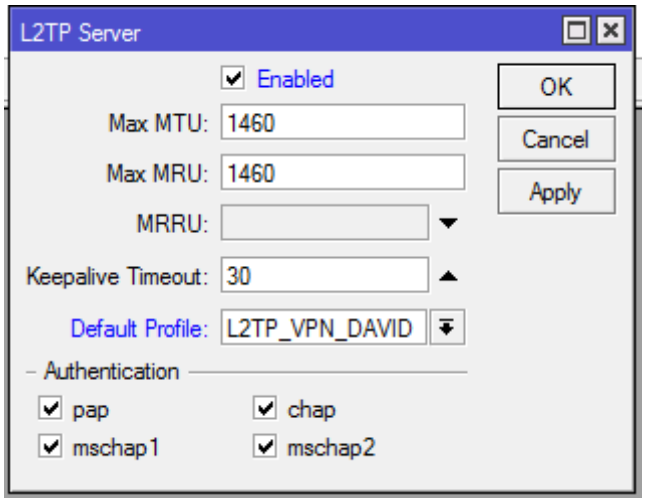
Hecho esto nos dirigiremos a la pestaña Secrets, aquí definimos el usuario y contraseña para la conexión VPN:



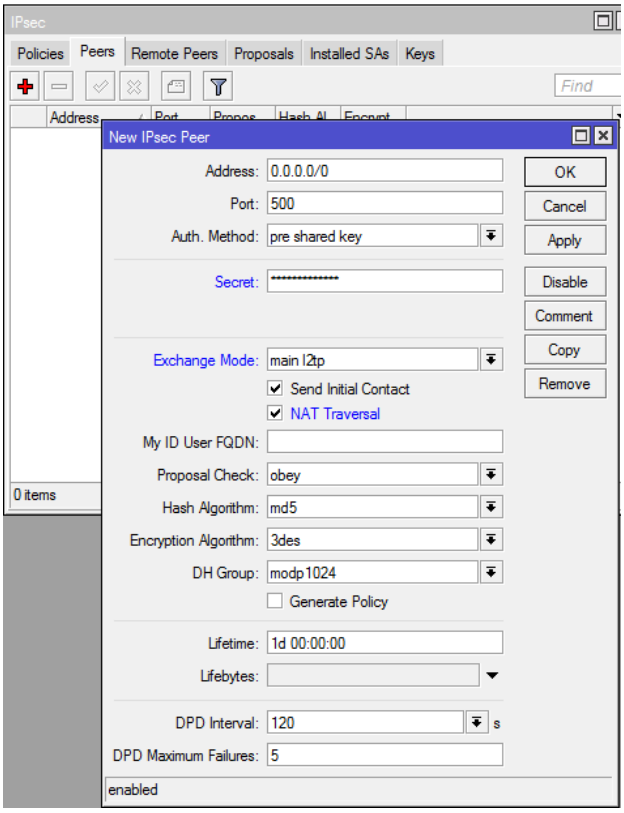
En la pestaña “Interface” debemos pulsar sobre “L2TP Server”, habilitarlo:



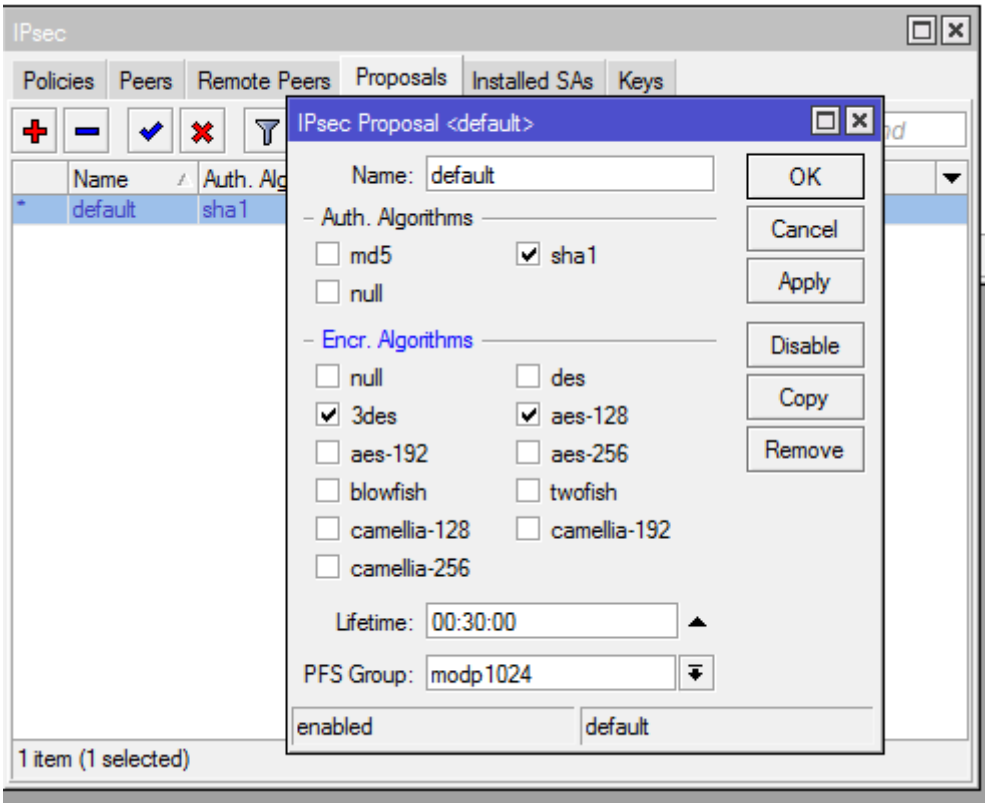
Deberá quedar de la siguiente manera:



Para proteger la información que transporta la conexión vamos hacer un encapsulamiento sobre IPsec, para eso nos vamos a IP → IPsec y rellenamos los datos de la siguiente manera:

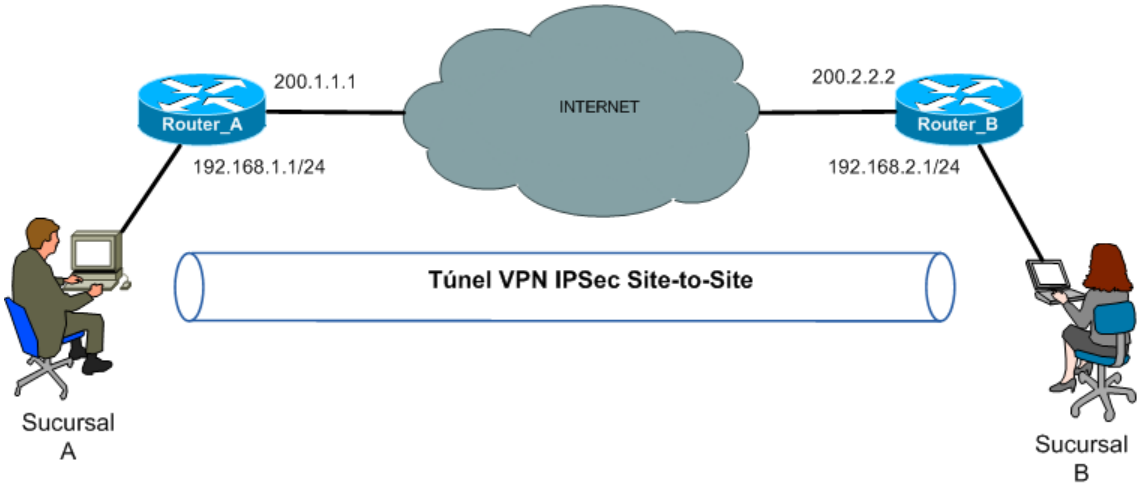


El siguiente paso es ir a la pestaña “Proposals” y dejarlo de la siguiente manera:

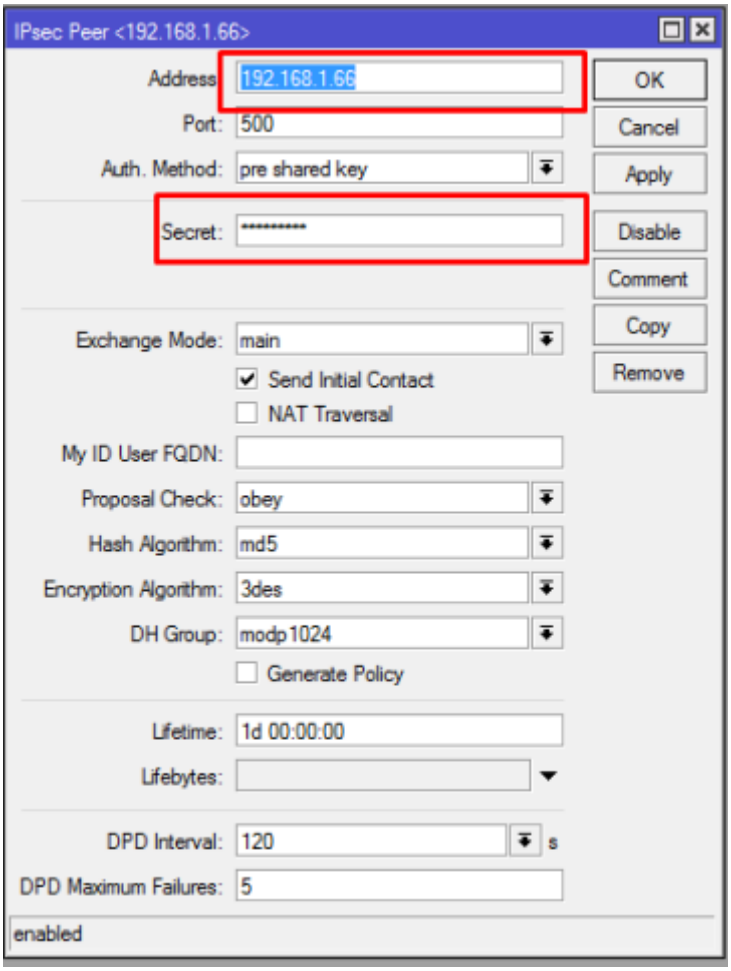


d. VPN SITE TO SITE MIKROTIK

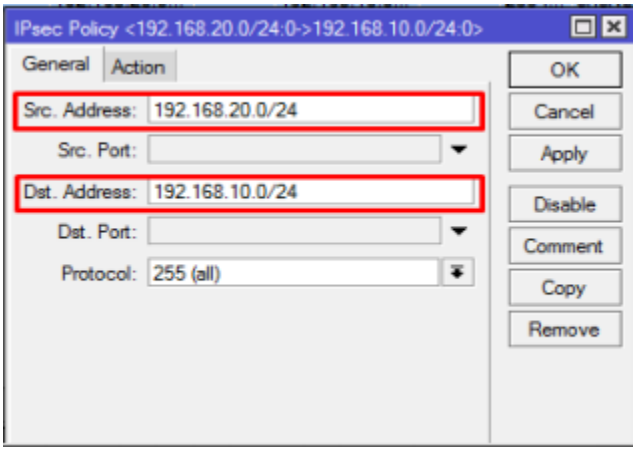
Esta tecnología de VPN nos permite conectar 2 redes diferentes y separadas geográficamente. Gracias a esta característica, he elegido implementar esta VPN, ya que nos venía muy bien tener tanto la red de la oficina como la red del CPD conectadas. Esta tecnología la vamos a implementar en un router Mikrotik, gracias a su facilidad de implementación esta tecnología se hace a través de una interfaz gráfica es decir a través de winbox. Esta implementación se basa en 2 mikrotik, que cumplen las configuraciones para establecer conexión entre ellos.

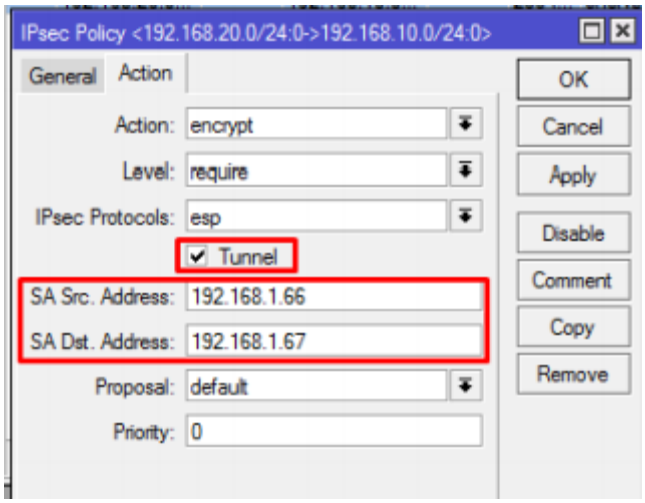


Entramos en Ip → IPSec → Peers, cuando estemos en esta pestaña lo que vamos hacer es crear un punto:

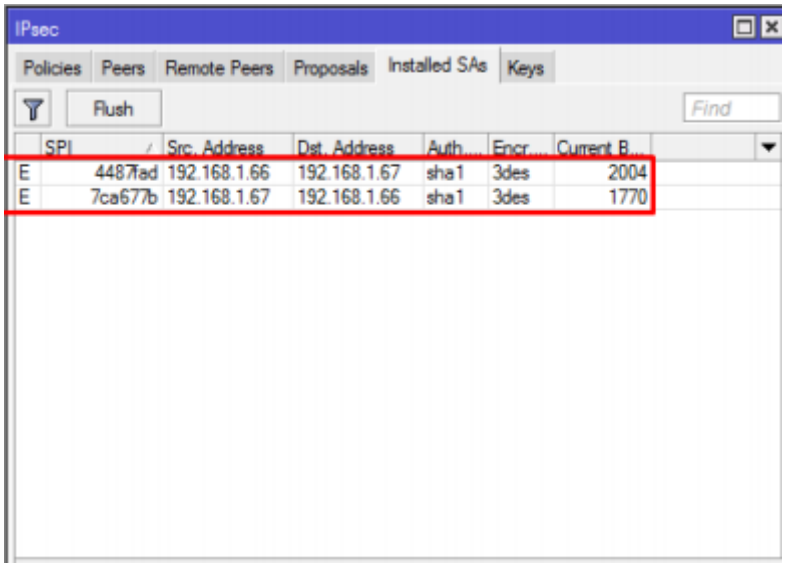


Posteriormente tenemos que crear las políticas para la VPN:

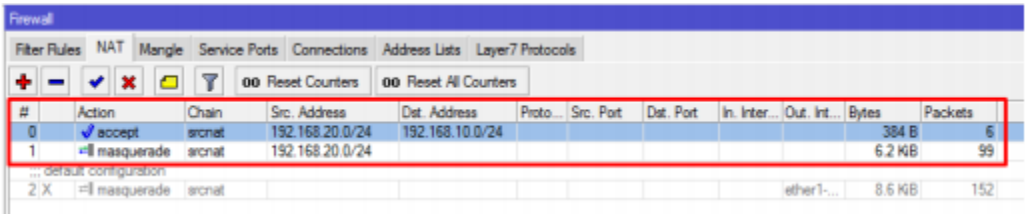




Tras crear las políticas se crean las conexiones automáticamente:



Después de comprobar todo esto lo que deberemos hacer es introducir las reglas del firewall correspondientes:



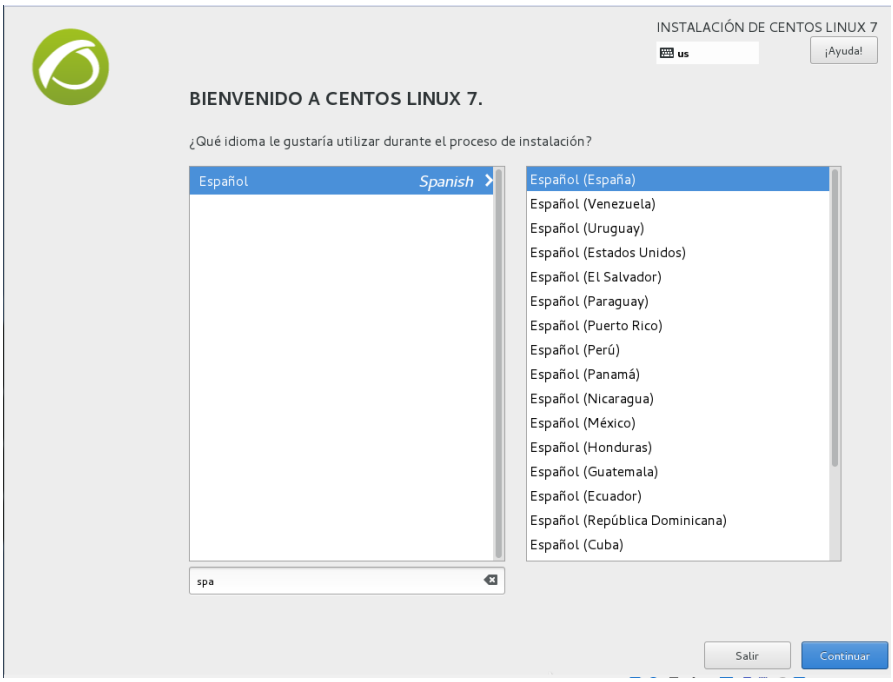
Para el router 2 se debe hacer el mismo proceso pero a la inversa.

13. INSTALACIÓN PANDORAFMS

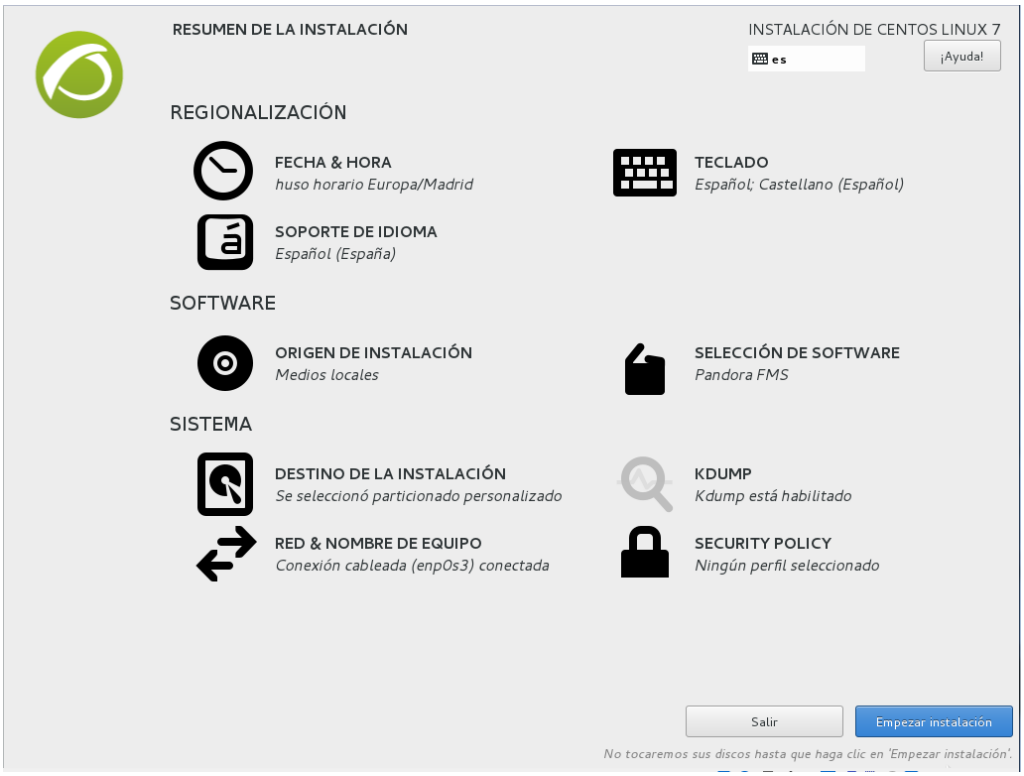
Como pandoraFMS lo instalaremos desde su sistema operativo, es decir es un sistema operativo en el que ya viene instalado pandoraFMS nos resultará más sencillo, nos dirigiremos a la página oficial de PandoraFMS y descargaremos la imagen iso, este lo grabaremos en un dvd o bootearemos un USB para la instalación, aquí el proceso de instalación del sistema operativo:

1.- Cuando se inicia la instalación del sistema operativo, pulsaremos en la primera opción:

2.- Elegimos español como idioma:

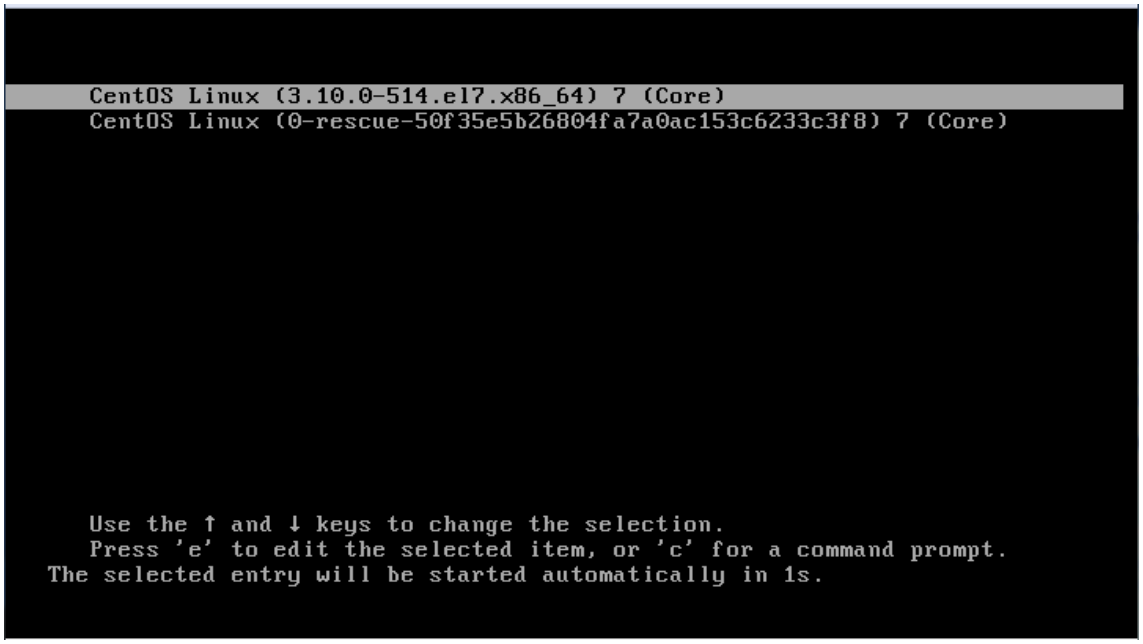


3.- Configuramos las diversas opciones de la instalación:

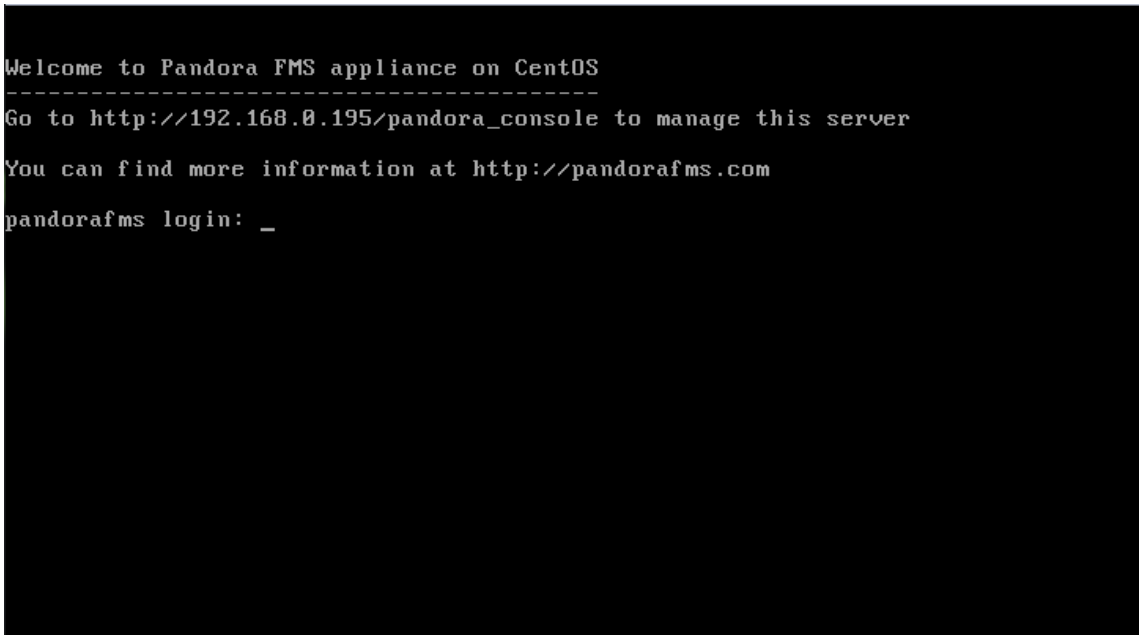


4.- Creamos una contraseña para el usuario root y un usuario local:

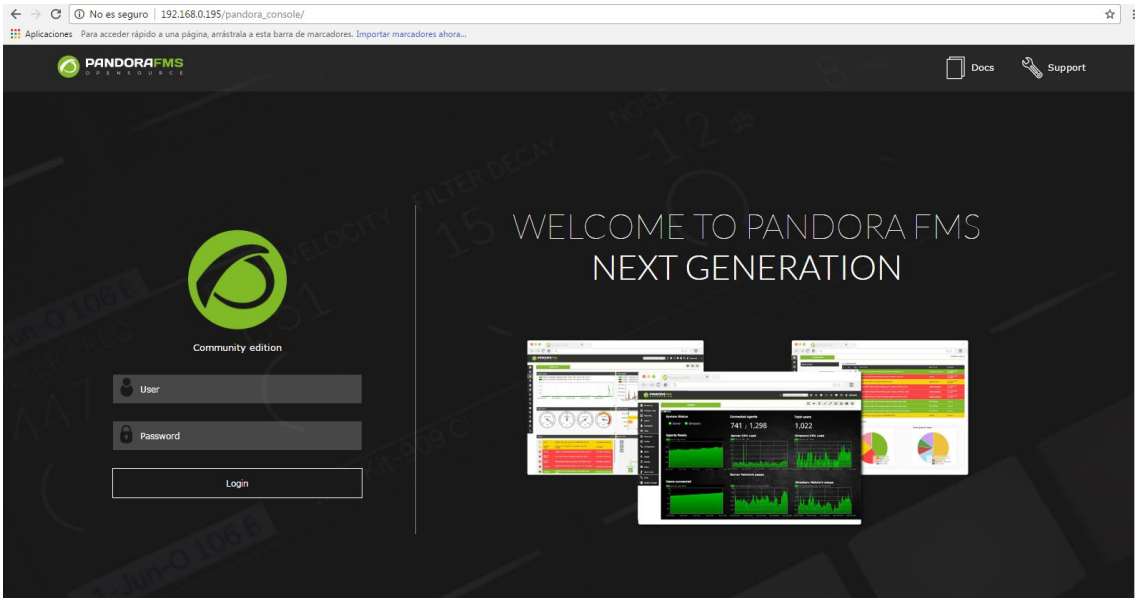
5.- Una vez finalizado el proceso de instalación reiniciamos el sistema y accedemos al S.O seleccionando la primera opción:



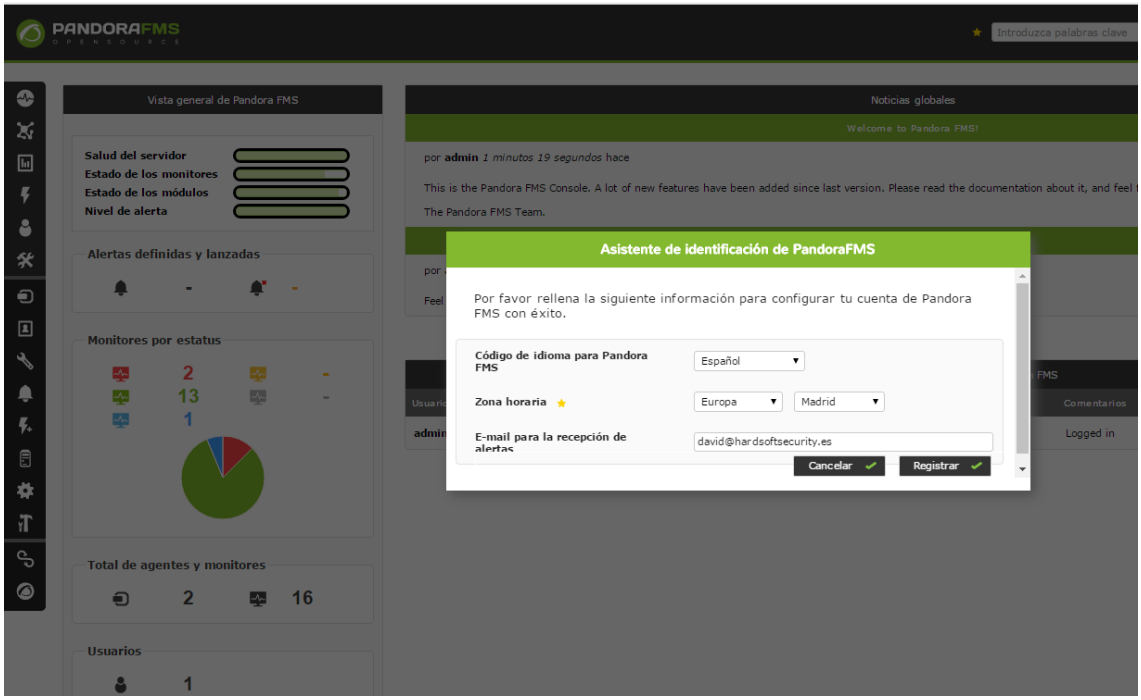
6.- Cuando inicia el sistema operativo nos indica la dirección URL que debemos ingresar en el navegador para acceder al login de PandoraFMS:



7.- Una vez ingresada la dirección en el navegador se nos presentará lo siguiente, donde ingresaremos con el usuario: admin y la contraseña proporcionada para el usuario root:



8.- Una vez dentro veremos la interfaz de PandoraFMS:



14. INSTALACIÓN NETFLOW SOBRE MIKROTIK

Lo que vamos a instalar el sistema operativo Debian en su versión 7.11, la cual instalaremos una serie de paquetes y reglas para poder visualizar todo el tráfico de la red monitorizada individualmente por equipos.

1.- Comenzamos por instalar el siguiente paquete:

```
root@monitor:/home/soporte# aptitude install pmacct
```

2.- Ya que el paquete anterior instala más de un paquete que no interesa tenerlo, procedemos a desinstalarlos:

```
root@monitor:/home/soporte# update-rc.d -f sfacctd remove
root@monitor:/home/soporte# update-rc.d -f uacctd remove

root@monitor:/home/soporte# update-rc.d -f pmacct remove
```

3.- Paramos el servicio de pmacct:

```
root@monitor:/etc/init.d# /etc/init.d/pmacct stop
```

4.- Renombramos el fichero nfacctd.conf para hacer una copia de seguridad del mismo:

```
root@monitor:/etc/init.d# mv /etc/pmacct/nfacctd.conf /etc/pmacct/nfacctd.conf.backup
```

5.- Abrimos un editor de texto para crear un nuevo fichero para remplazar al anterior:

```
root@monitor:/etc/init.d# nano /etc/pmacct/nfacctd.conf
```

6.- Tras crear el fichero, el siguiente tiene que quedar así:

```
 GNU nano 2.2.6              Fichero: /etc/pmacct/nfacctd.conf                    Modificado

!
! NFACCTD CONFIGURATION, ACEPTAR TRAFICO DESDE MIKROTIK
! TRAFFIC FLOW.
!
debug: false
daemonize: true
!
plugin_buffer_size: 80524
plugin_pipe_size: 18052324
!
networks_file_filter: true
networks_file: /etc/pmacct/hosts.def
!
! IMPORTANTE DEFINIR EL PUERTO QUE ESCUCHARA EL COLECTOR
! LUEGO EN EL MIKROTIK DEBEMOS INDICAR ESTE NUMERO DE PUERTO
!
nfacctd_port: 5055
!
!
! NOS INTERESA EL TRAFICO IN/OUT
!
plugins: memory[in], memory[out]
!
aggregate[in]: dst_host
aggregate[out]: src_host
!
! LA INFORMACION DE LOS FLOWS RECIBIDOS SE ALMACENARA
! EN LA MEMORIA, PARA LUEGO SER PROCESADA Y GRAFICADA
!
imt_path[in]: /tmp/pmacct_in.pipe
imt_path[out]: /tmp/pmacct_out.pipe
```

7.- Procedemos a editar el fichero hosts.def:

```
root@monitor:/etc/init.d# nano hosts.def
```

8.- Añadimos las redes que deseamos monitorizar:

```
 GNU nano 2.2.6                  Fichero: hosts.def                          Modificado

149.14.100.168/29
192.168.21.0/24
```

9.- Tras realizar todas estas modificaciones, paramos e iniciamos el servicio nfacctd:

```
root@monitor:/etc/pmacct# /etc/init.d/nfacctd stop
Stopping netflow accounting daemon:
root@monitor:/etc/pmacct# /etc/init.d/nfacctd start
Starting netflow accounting daemon: WARN ( /etc/pmacct/nfacctd.conf ): Unknown key: networks_file_fil
ter. Line 11 ignored.
nfacctd.
```

10.- Comprobamos que está funcionando:

```
root@monitor:/etc/pmacct# netstat -punlt|grep 5055
udp6       0      0 :::5055                 :::*                                3326/nfacctd: Core
```

11.- Lo siguiente que debemos hacer es dirigirnos al terminal de nuestro Mikrotik e insertar las siguientes líneas:

```
[admin@interxion1] > /ip traffic-flow set active-flow-timeout=30m cache-entries=1M
 enabled=yes inactive-flow-timeout=15s interfaces=all
```

```
[admin@interxion1] > /ip traffic-flow target add dst-address=192.168.21.114 port=5055 disabled=no v9-temp
late-refresh=20 v9-template-timeout=30m version=9
```

12.- Habilitamos la interfaz web de la siguiente manera, empezamos por crear la carpeta pnrq:

```
root@monitor:/home/soporte# mkdir /usr/local/pnrg
```

13.- Nos dirigimos a la carpeta creada:

```
root@monitor:/home/soporte# cd /usr/local/pnrg/
```

14.- Nos descargamos el paquete necesario para la visualización a tiempo real del tráfico:

```
root@monitor:/usr/local/pnrg# wget http://www.pmacct.net/pnrg/pnrg-0.1.tar.gz
```

15.- Lo descomprimimos:

```
root@monitor:/usr/local/pnrg# tar zxvf pnrg-0.1.tar.gz
```

16.- Lo movemos:

```
root@monitor:/usr/local/pnrg# mv pnrg-0.1/* .
```

17.- Instalamos una herramienta necesaria:

```
root@monitor:/usr/local/pnrg# apt-get install rrdtool
```

18.- Insertamos la tarea en crontab para que se refresque la información cada 5 minutos:

```
root@monitor:/usr/local/pnrg# echo "*/5 * * * * root ( cd /usr/local/pnrg/; ./pnrg-wrapper.sh )" > /etc/cron.d/pnrg
```

19.- Enlazamos el servicio pmacct para no tener que editar más ficheros:

```
root@monitor:/usr/local/pnrg# ln -s /usr/bin/pmacct /usr/local/bin/pmacct
```

20.- Creamos el directorio bin dentro de rrdtool:

```
root@monitor:/usr/local/pnrg# mkdir -p /usr/local/rrdtool/bin
```

21.- Creamos un enlace dentro de la carpeta creada anteriormente:

```
root@monitor:/usr/local/pnrg# ln -s /usr/bin/rrdtool /usr/local/rrdtool/bin/rrdtool
```

22.- Enlazamos rrdcgi dentro de la carpeta anteriormente creada:

```
root@monitor:/usr/local/pnrg# apt-get install apache2
```

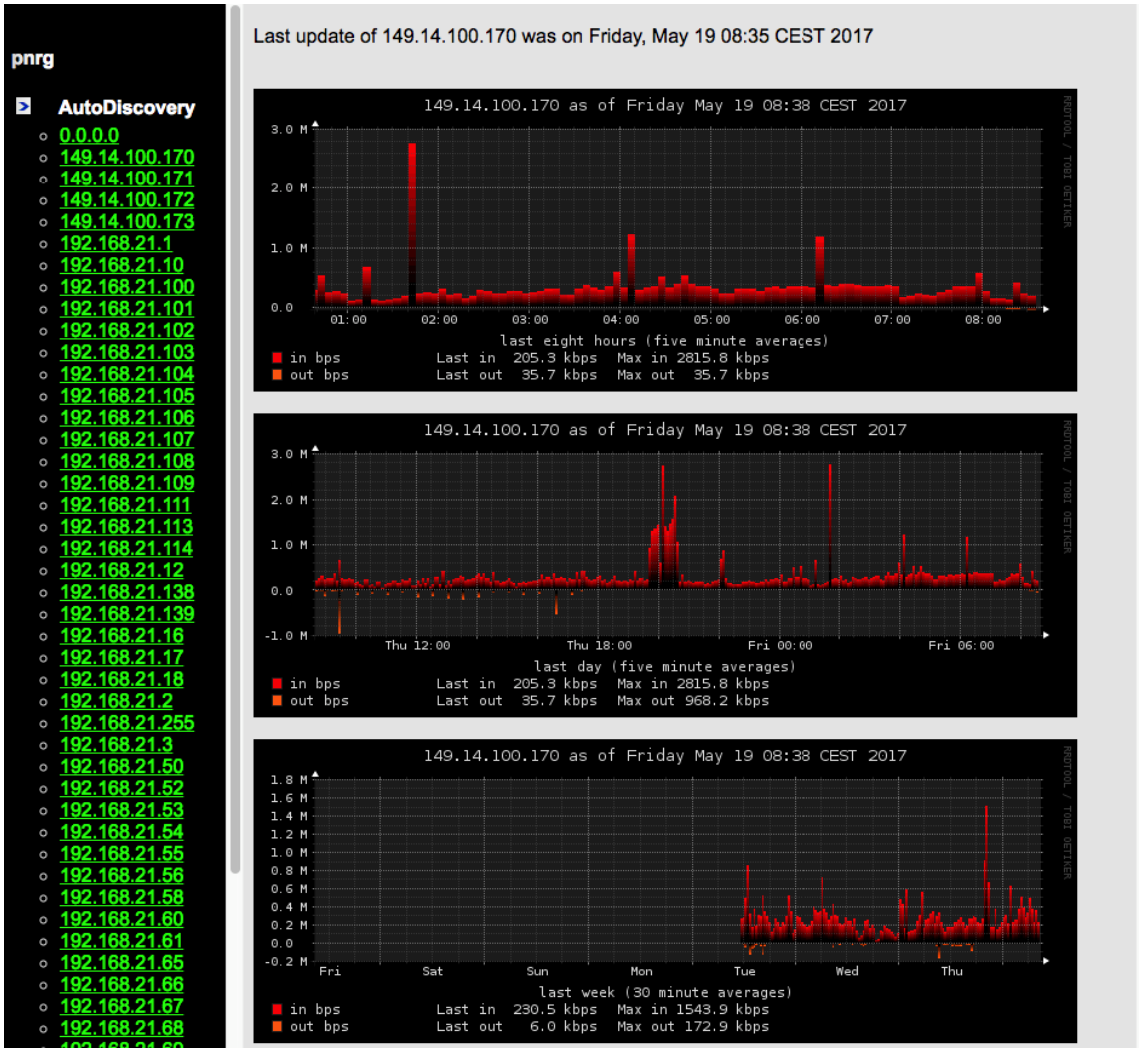
23.- Enlazamos la interfaz web dentro de la carpeta de apache:

```
root@monitor:/usr/local/pnrg# ln -s /usr/local/pnrg/spool /var/www/pnrg
```

24.- Y habilitamos el sitio web dentro de 000-default:

```
root@monitor:/usr/local/pnrg# nano /etc/apache2/sites-enabled/000-default
```

25.- Una vez realizado todo esto mediante un navegador web nos dirigiremos a la ip que tenga la máquina y se nos presentará la siguiente interfaz web:



15. INSTALACIÓN APACHE GUACAMOLE

Procedemos a instalar apache guacamole sobre Centos7, esta herramienta como se ha comentado anteriormente permite gestionar cualquier conexión remota a través de una interfaz web.

1.- El primer paso para instalarlo es descargar un script desarrollado en Shell script el cual nos instalará automáticamente la herramienta:

```
[root@centos david]# wget --no-check-certificate https://sourceforge.net/project
s/guacamoleinstallscript/files/CentOS/guacamole-install-script.sh_
```

2.- Una vez descargado el script, le daremos permisos de ejecución:

```
[root@centos david]# ls
download  guacamole-install-script.sh
[root@centos david]# chmod +x guacamole-install-script.sh
```

3.- Tras darle permisos de ejecución lo iniciamos:

```
[root@centos david]# ./guacamole-install-script.sh _
```

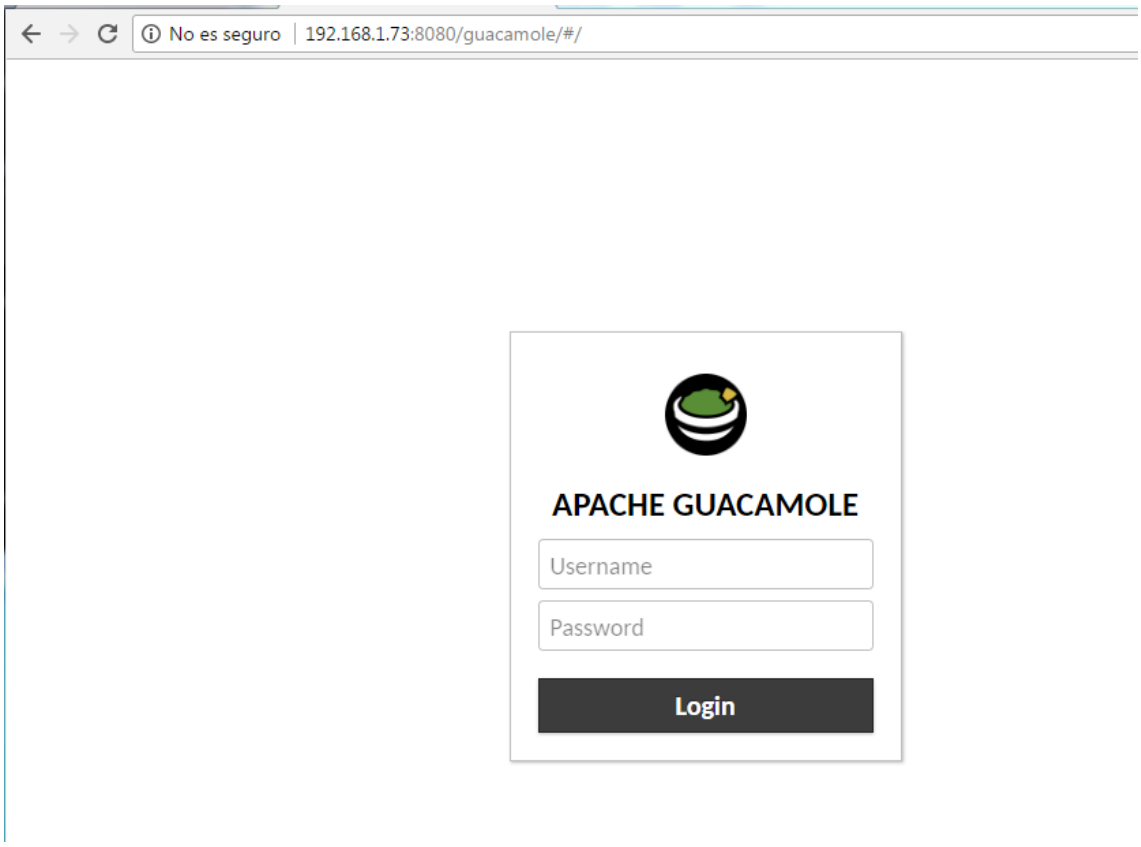
4.- Cuando se inicia el script nos pedirá que ingresemos usuario, contraseña para las bases de datos:

```
            /dMMNdyo/-.''''          ''''.-:+shmMNh:
              :yNMMMMMMNmdhhyyyyyyyyhhdmNNMMMMMMNy:
                ':sdNNMMMMMMMMMMMMMMMMMMMMNds:'
                    '-/+syhdmNNNNNNmdhyo/-'

                        Installation Menu
                Guacamole Remote Desktop Gateway 0.9.10-incubating

Enter the root password for MariaDB: David12345
Enter the Guacamole DB name: daviddb
Enter the Guacamole DB username: david
Enter the Guacamole DB password: David12345
Enter the Java KeyStore password (least 6 characters): David12345
Do you wish to Install the Proxy feature (Nginx)?: no
```

5.- Lo siguiente que nos aparecerá son unas preguntas para generar los certificados para tener una conexión segura en nuestras conexiones remotas:

```
Creating Guacamole Tables...


Setting Tomcat Server...


Please complete the Wizard for the Java KeyStore

¿Cuáles son su nombre y su apellido?
  [Unknown]:  David De Maya
¿Cuál es el nombre de su unidad de organización?
  [Unknown]:  hardsoftsecurity
¿Cuál es el nombre de su organización?
  [Unknown]:  hardsoftsecurity
¿Cuál es el nombre de su ciudad o localidad?
  [Unknown]:  Orihuela
¿Cuál es el nombre de su estado o provincia?
  [Unknown]:  Alicante
¿Cuál es el código de país de dos letras de la unidad?
  [Unknown]:  ES
¿Es correcto CN=David De Maya, OU=hardsoftsecurity, O=hardsoftsecurity, L=Orihue
la, ST=Alicante, C=ES?
  [no]:  yes_
```

6.- Una vez hecho esto nos dirigimos al navegador colocando la siguiente URL, 192.168.21.73:8080/guacamole/#/ y a continuación nos saldrá lo siguiente:



16. INSTALACIÓN DE SNORT Y SNORBY

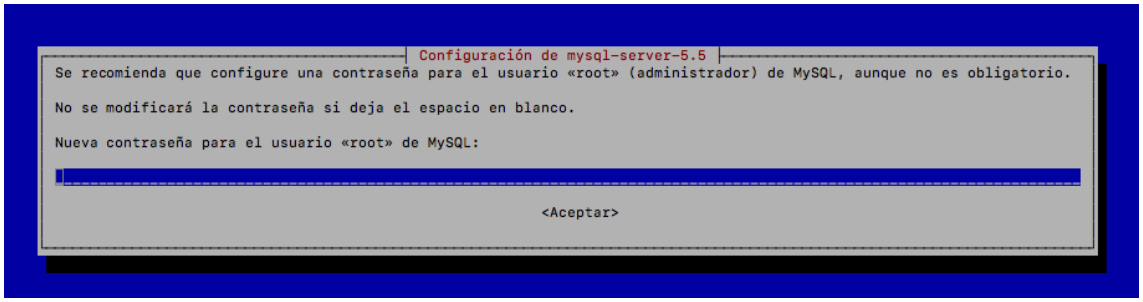
Vamos a implementar snort y snorby sobre debían 7.11.

1.- Comenzamos por instalar una serie de paquetes:

```
root@snort:~# apt-get install git ruby ruby-dev mysql-server libmysqlclient-dev libmysql++-dev imagemagick libmagickwand-dev wkhtmltop
df gcc g++ build-essential linux-headers-amd64 libssl-dev libreadline-gplv2-dev zlib1g-dev libsqlite3-dev libxslt1-dev libxml2-dev -y
```

apt-get install git ruby ruby-dev mysql-server libmysqlclient-dev libmysql++-dev imagemagick libmagickwand-dev wkhtmltopdf gcc g++ build-essential linux-headers-amd64 libssl-dev libreadline-gplv2-dev zlib1g-dev libsqlite3-dev libxslt1-dev libxml2-dev –y

2.- Nos pedirá que insertemos la contraseña de MYSQL:



3.- Para instalar snorby comenzaremos por descargarnos el código:

```
root@snort:~# git clone http://github.com/Snorby/snorby.git
```

4.- Ahora vamos a actualizar la versión de ruby, instalamos los siguientes paquetes:

```
root@snort:~/snorby# apt-get install build-essential bison openssl libreadline6 libreadline6-dev \
> libyaml-dev libxml2-dev libxslt-dev zlib1g zlib1g-dev libssl-dev autoconf \
> libc6-dev ncurses-dev libaprutil1-dev libffi-dev libcurl4-openssl-dev libapr1-dev
```

5.- Nos descargamos el código de ruby para instalarlo:

```
root@snort:~/snorby# wget http://cache.ruby-lang.org/pub/ruby/2.1/ruby-2.1.2.tar.gz
```

6.- Descomprimimos el paquete:

```
root@snort:~/snorby# tar -xvf ruby-2.1.2.tar.gz
```

7.- Preparamos el sistema para instalar ruby:

```
root@snort:~/snorby/ruby-2.1.2# ./configure
```

8.- Hacemos un MAKE:

```
root@snort:~/snorby/ruby-2.1.2# make
```

9.- Y make install:

```
root@snort:~/snorby/ruby-2.1.2# make install
```

10.- Comprobamos la versión:

```
root@snort:~/snorby# ruby -v
ruby 2.1.2p95 (2014-05-08 revision 45877) [x86_64-linux]
```

11.- Instalamos las bundle dentro de la carpeta de snorby:

```
root@snort:~/snorby# bundle install
```

12.- instalamos la gema bundler:

```
root@snort:~/snorby# gem install bundler
Fetching: bundler-1.15.0.gem (100%)
Successfully installed bundler-1.15.0
Parsing documentation for bundler-1.15.0
Installing ri documentation for bundler-1.15.0
Done installing documentation for bundler after 8 seconds
1 gem installed
root@snort:~/snorby# bundle install
```

13.- Instalamos las gemas:

```
root@snort:~/snorby# bundle install
```

Con el siguiente resultado:

```
Bundle complete! 69 Gemfile dependencies, 117 gems now installed.
Use `bundle info [gemname]` to see where a bundled gem is installed.
```

14.- Renombramos el fichero database.yml.example, para ingresar los datos para la base de datos:

```
root@snort:~/snorby# cp config/database.yml.example config/database.yml
```

15.- Editamos el fichero:

```
root@snort:~/snorby# nano config/database.yml
```

16.- Tiene que quedar de la siguiente manera:

```
  GNU nanc 2.2.6                          Fichero: config/database.yml

# Snorby Database Configuration
#
# Please set your database password/user below
# NOTE: Irdentation is important.
#
snorby: &snorby
  adapter: mysql
  username: root
  password: "Enter Password Here" # Example: password: "s3cr3tsauce"
  host: lccalhost

development:
  database: snorby
  <<: *sncrby

test:
  database: snorby
  <<: *sncrby

production:
  database: snorby
  <<: *sncrby
```

17.- copiamos el fichero snorby_config.yml.example:

```
root@snort:~/snorby# cp config/snorby_config.yml.example config/snorby_config.yml
```

18.- Procedemos a instalar snorby:

```
root@snort:~/snorby# bundle exec rake snorby:setup
```

19.- Comprobamos que ha creado la estructura de la base de datos:

```
[root@snort:~/snorby# mysql -u root -p -D snorby
```

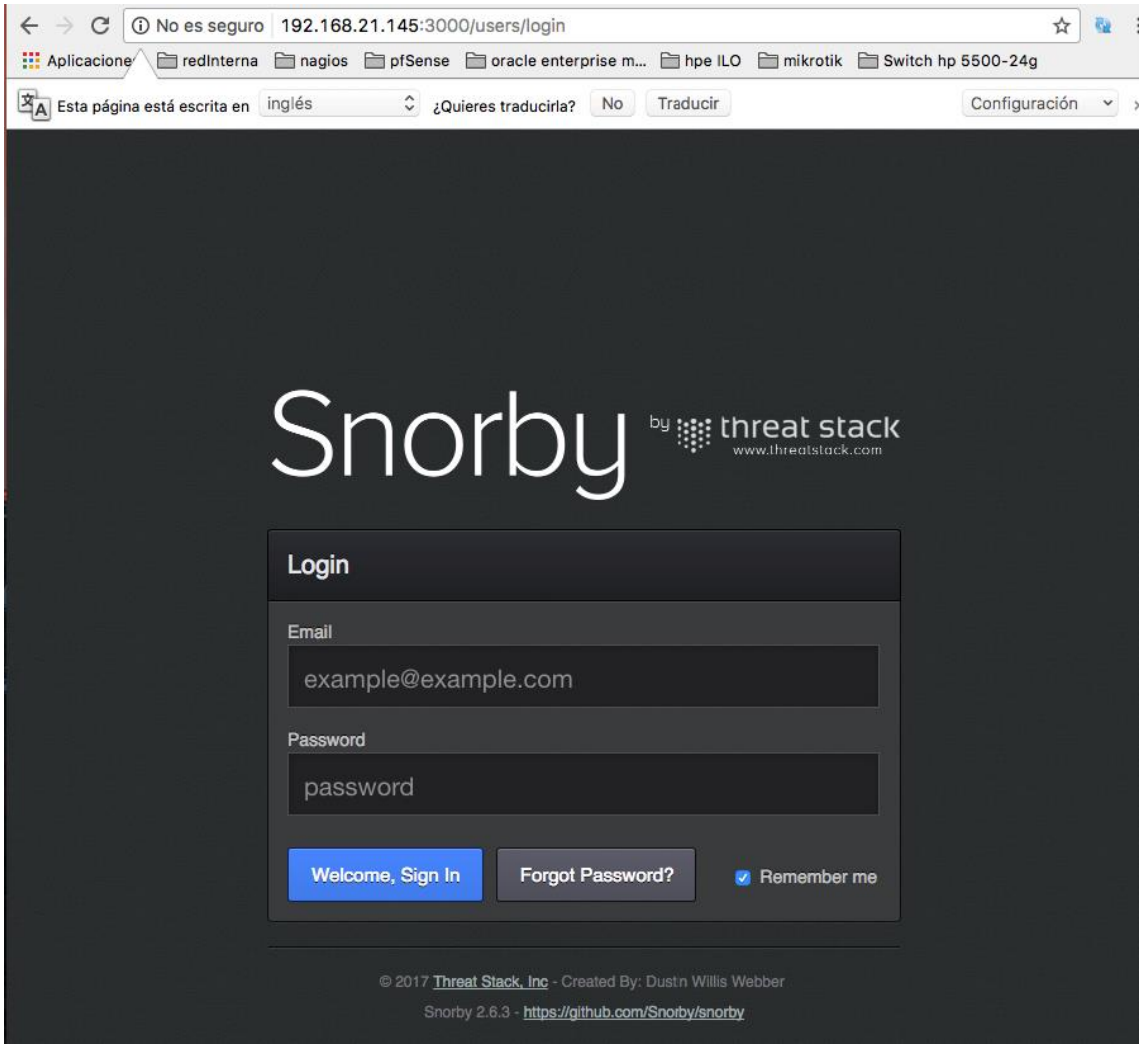
20.- La estructura que debe de presentar es la siguiente:

```
mysql> show tables;
+-------------------+
| Tables_in_snorby  |
+-------------------+
| agent_asset_names |
| aggregated_events |
| asset_names       |
| caches            |
| classifications   |
| data              |
| delayed_jobs      |
| detail            |
| encoding          |
| event             |
| events_with_join  |
| favorites         |
| icmphdr           |
| iphdr             |
| lookups           |
| notes             |
| notificaticns     |
| opt               |
| reference         |
| reference_system  |
| schema            |
| search            |
| sensor            |
| settings          |
| severities        |
| sig_class         |
| sig_refererce     |
| signature         |
| tcphdr            |
| udphdr            |
| users             |
+-------------------+
31 rows in set (0.00 sec)
```

21.- Ejecutamos snorby:

```
root@snort:~/snorby# bundle exec rails server -e production
```

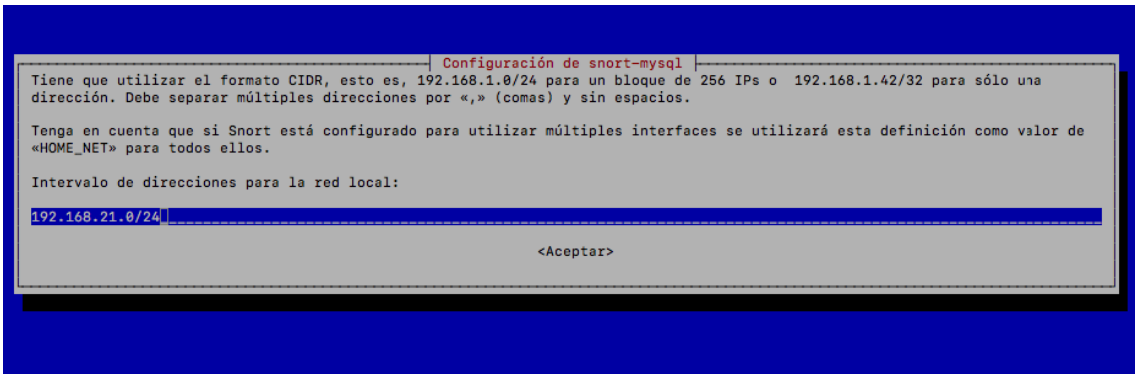
22.- Se nos presenta la interfaz de snorby:



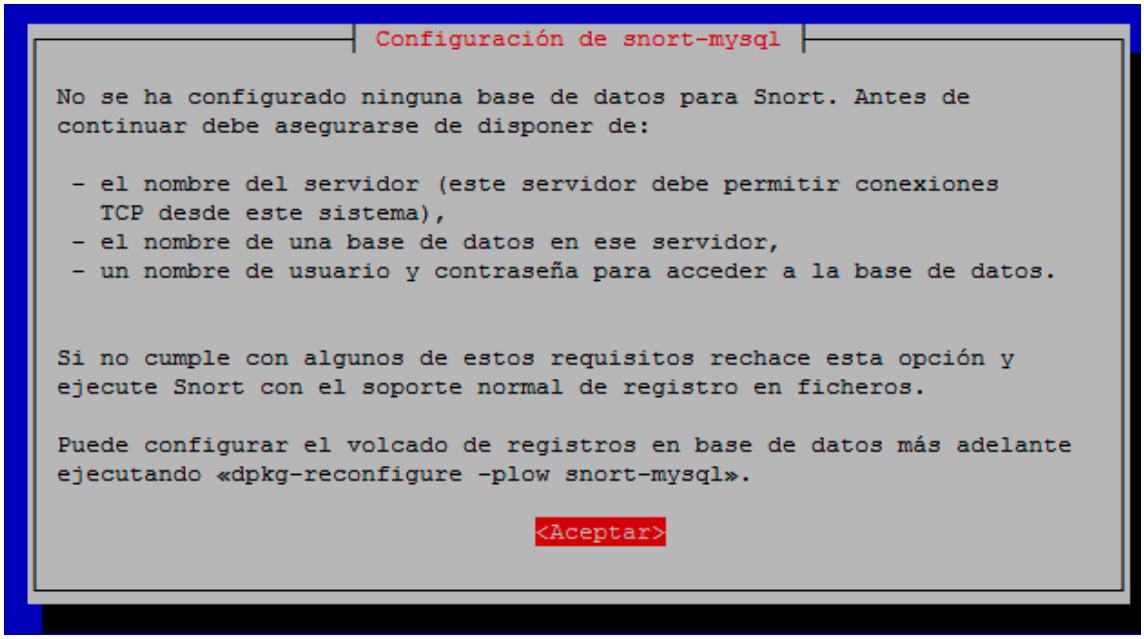
23.- Procedemos a instalar snort, comenzamos instalando unos paquetes:

```
root@snort:~/snorby# apt-get install snort-mysql snort-rules-default -y
```

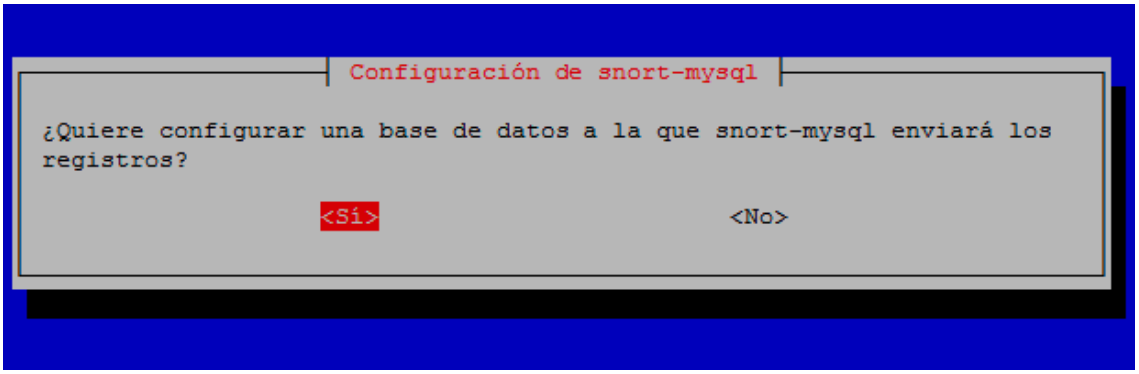
24.- Insertamos el rango de ip que deseamos sniffear:



25.- Aceptamos el siguiente paso:



26.- Aceptamos la configuración de la base de datos de snort:

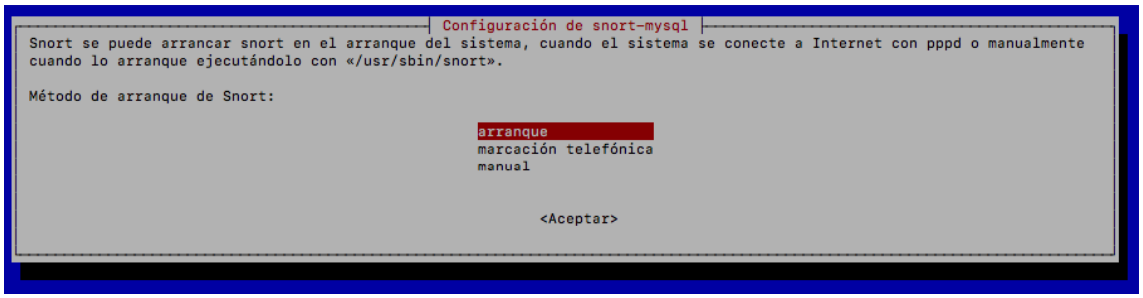


27.- Continuamos el proceso:

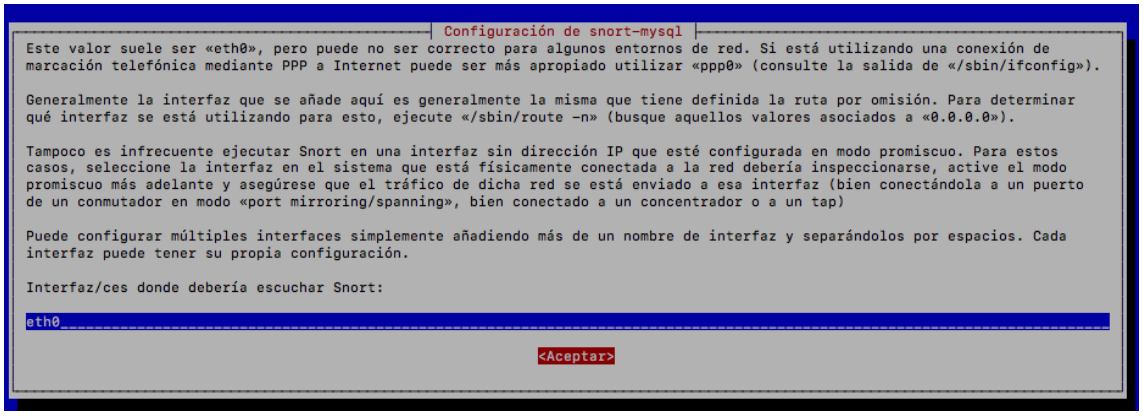
28.- Reconfiguramos el paquete de mysql:

```
root@snort:~/snorby# dpkg-reconfigure --force snort-mysql
```

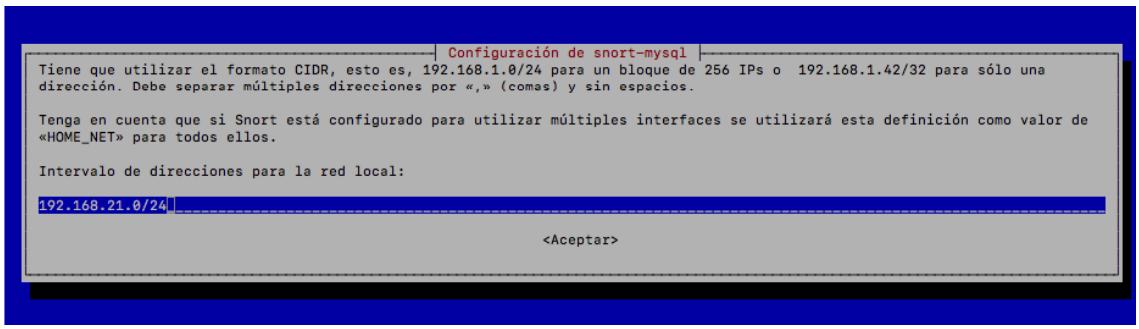
29.- Elegimos el método de arranque:



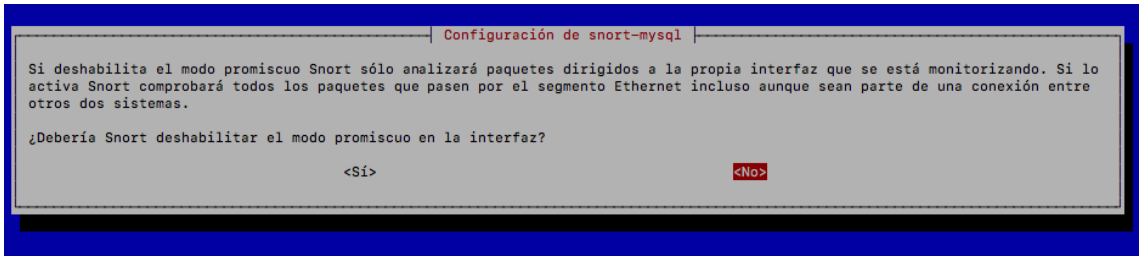
30.- Elegimos la interfaz que deseamos poner en modo monitor:



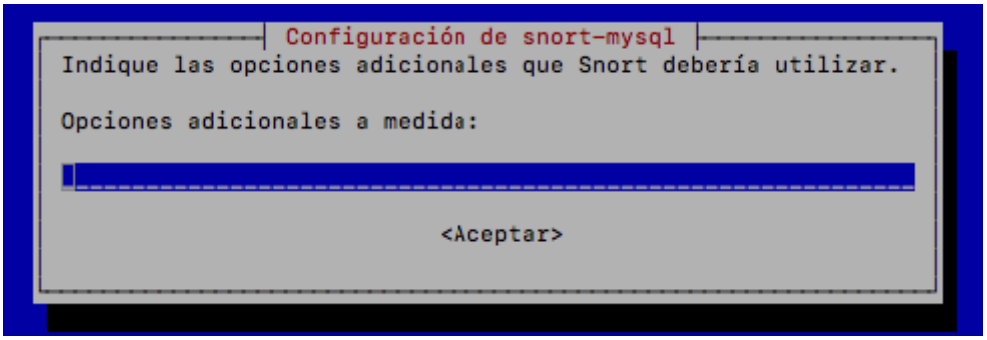
31.- Volvemos a introducir el rango de red:



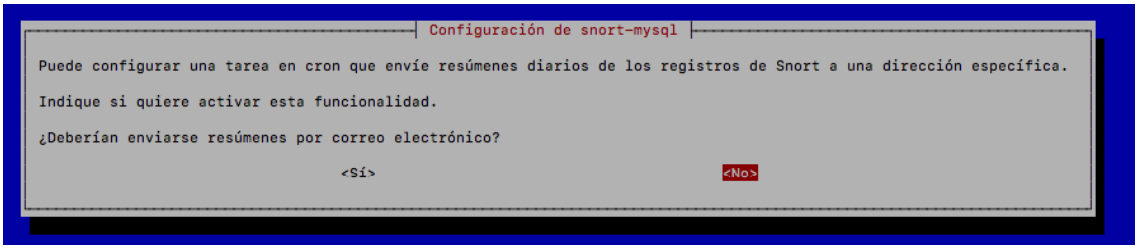
32.- Dejamos habilitado el modo promiscuo:



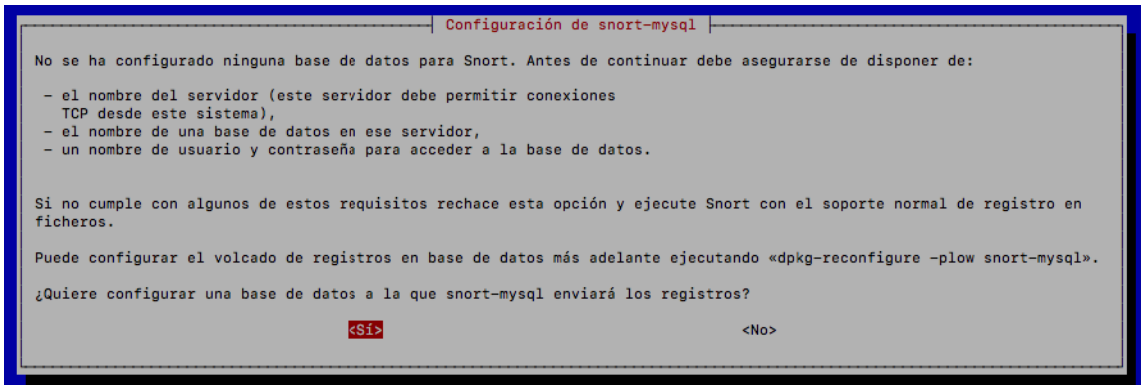
33.- Dejamos la siguiente opción en blanco:



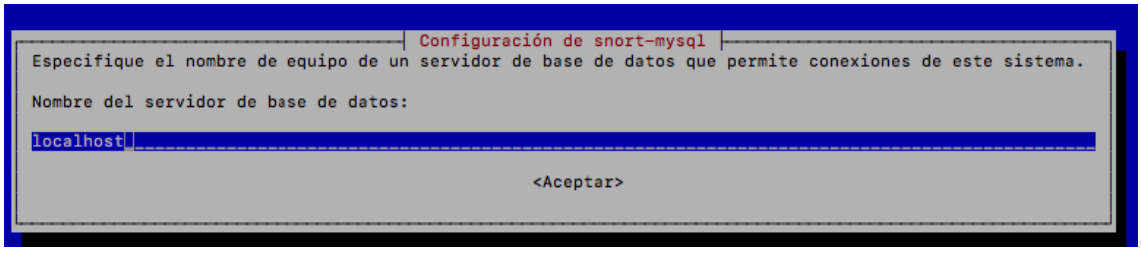
34.- Como vamos a ver las alertas en tiempo real no hace falta habilitar las alertas por correo:



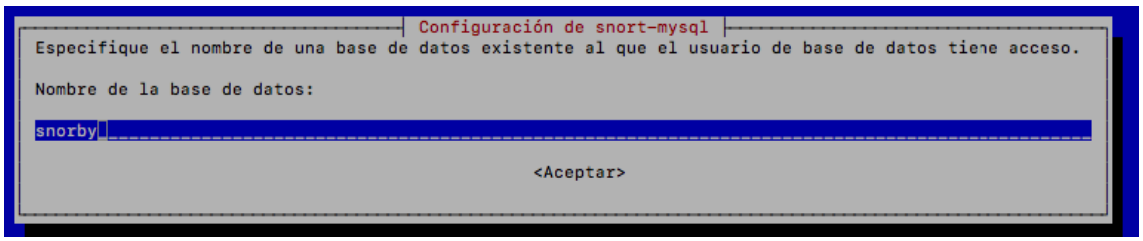
35.- El siguiente paso le decimos que sí aunque hemos configurado ya la base de datos:



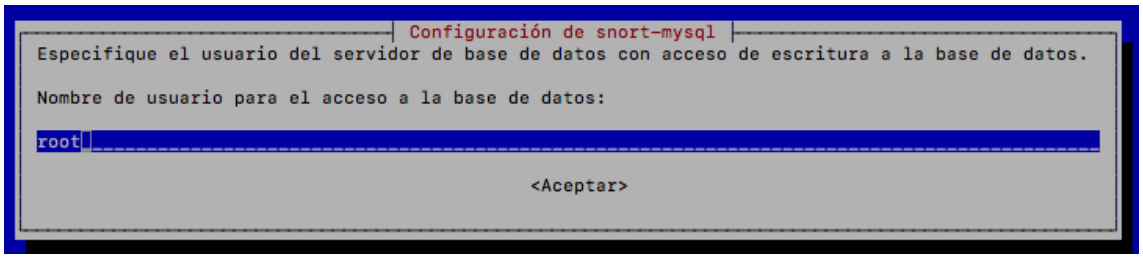
36.- Nombre del servidor de base de datos:



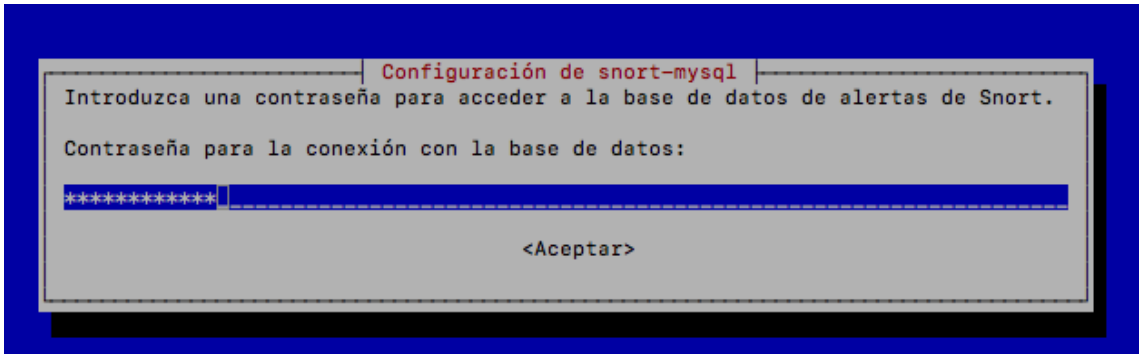
37.- Nombre de la tabla dentro de la base de datos:



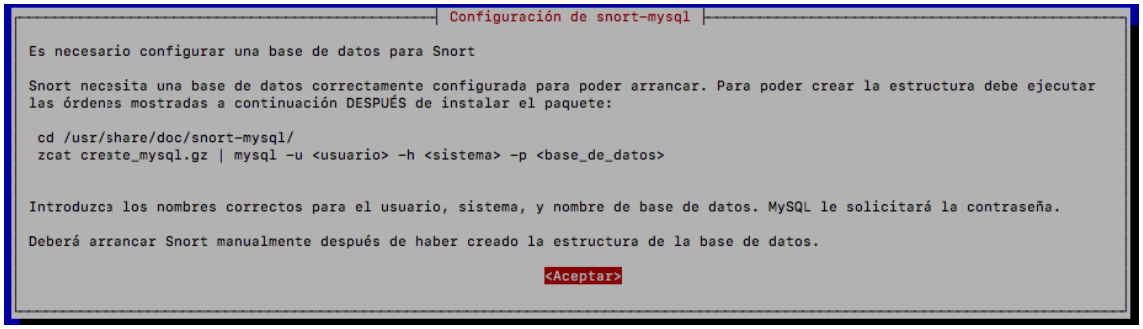
38.- Le decimos que usuario es el que tiene que usar:



39.- Nos pide la contraseña del usuario:



40.- Aceptamos el siguiente paso:



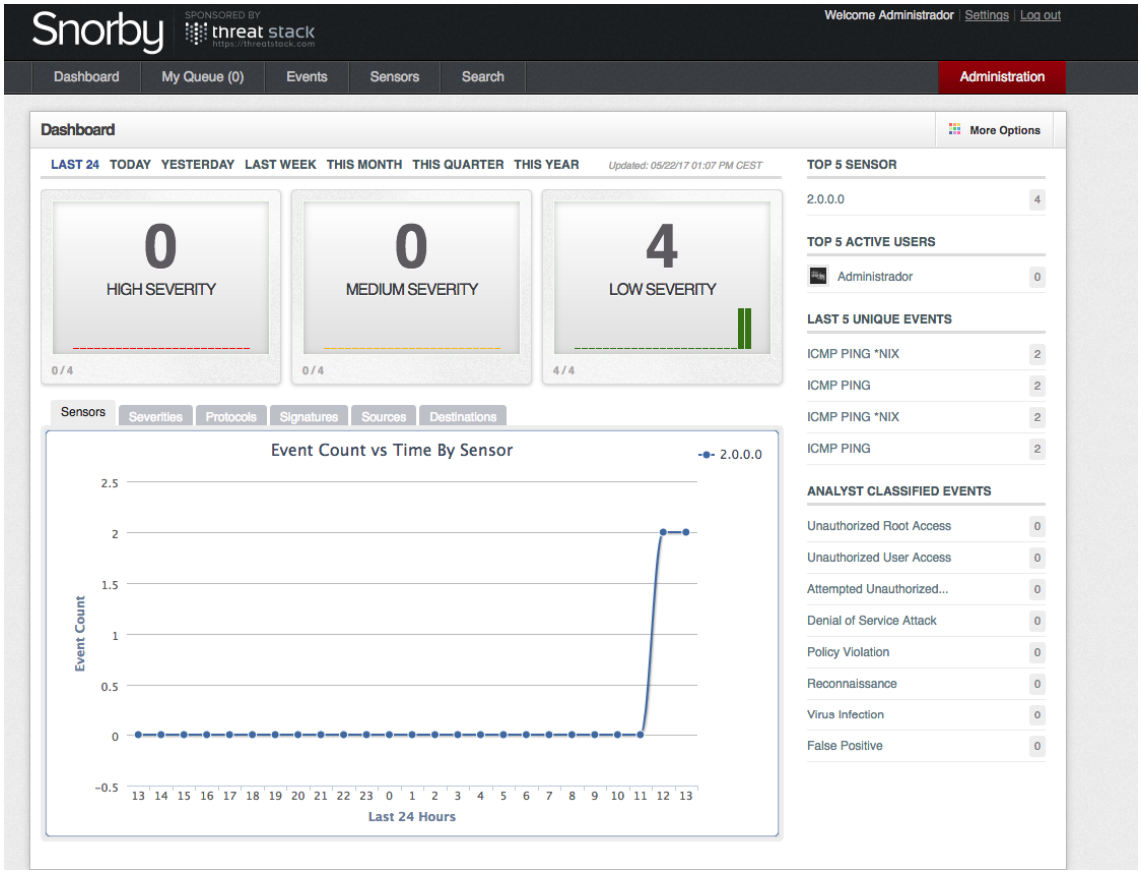
41.- Borramos el fichero db-pending-config, para que no nos vuelva a pedir la configuración de la BBDD:

```
root@snort:~/snorby# rm /etc/snort/db-pending-config
```

42.- Iniciamos snort y snorby:

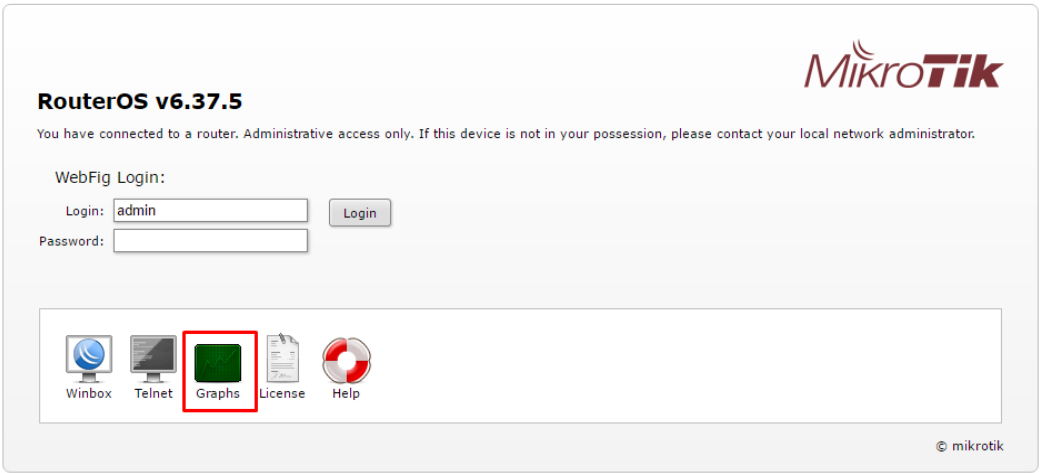
```
[root@snort:~/snorby# /etc/init.d/snort start
[ ok ] Starting Network Intrusion Detection System : snort (eth0 using /etc/snort/snort.conf ...done).
root@snort:~/snorby# bundle exec rails server -e production
```

43.- Una vez realizado todo nos dirigimos a un navegador y colocando la dirección ip de la máquina snort se nos presentará la interfaz de snort:



17. GRAPHS DE MIKROTIK

Este servicio normalmente viene habilitado por defecto y se puede acceder mediante la interfaz web de administración de nuestro mikrotik:



18. INSTALACIÓN FAIL2BAN

En esta instalación de FAIL2BAN se instalará en CentOS7:

1.- El primer paso es instalar el siguiente repositorio en nuestra máquina CentO 7:

```
[root@guac00 soporte]# yum install epel-release
```

2.- Tras instalar el repositorio y actualizarlos, procedemos a instalar Fail2ban:

```
[root@guac00 soporte]# yum install fail2ban
```

3.- Habilitamos el servicio de fail2ban:

```
[root@guac00 fail2ban]# systemctl enable fail2ban
```

4.- Nos dirigimos al directorio /etc/fail2ban para configurar el servicio:

```
[root@guac00 fail2ban]# cd /etc/fail2ban/
```

5.- Dejamos el fichero como en la siguiente imagen:

```
 GNU nano 2.3.1            Fichero: jail.local

[DEFAULT]
# Ban hosts for one hour:
bantime = 3600

# Override /etc/fail2ban/jail.d/00-firewalld.conf:
banaction = iptables-multiport

[sshd]
enabled = true
```

6.- Reiniciamos el servicio:

```
[root@guac00 fail2ban]# systemctl restart fail2ban
```

7.- Comprobamos que está monitorizando el servicio:

```
[root@guac00 fail2ban]# fail2ban-client status
Status
|- Number of jail:      1
`- Jail list:   sshd
[root@guac00 fail2ban]#
```

8.- Copiamos jail.conf y lo llamamos jail.local:

```
[[root@guac00 fail2ban]# cp jail.conf jail.local
```

9.- Editamos el fichero:

```
[[root@guac00 fail2ban]# nano jail.local
```

10.- Colocamos el rango de ips que deseamos ignorar:

```
[DEFAULT]

#
# MISCELLANEOUS OPTIONS
#

# "ignoreip" can be an IP address, a CIDR mask or a DNS host. Fail2ban will not
# ban a host which matches an address in this list. Several addresses can be
# defined using space (and/or comma) separator.
ignoreip = 127.0.0.1/8 192.168.21.0/24 192.168.0.0/24
```

11.- Creamos la jaula para ssh:

```
  GNU nano 2.3.1            Fichero: jail.local

# bantime = 3600
#
# [sshd]
# enabled = true
#
# See jail.conf(5) man page for more information
[ssh-iptables]
enabled = true
filter = sshd
action = iptables[name=SSH, port=ssh, protocol=tcp]
logpath = /var/log/secure
maxretry = 5

# Comments: use '#' for comment lines and ';' (following a space) for inline co$


[INCLUDES]
```

12.- Reiniciamos el servicio y comprobamos que este correcto:

```
[[root@guac00 fail2ban]# service fail2ban restart
 Redirecting to /bin/systemctl restart  fail2ban.service
[[root@guac00 fail2ban]# chkconfig fail2ban on
 Nota: Reenviando petición a 'systemctl enable fail2ban.service'.
```

13.- Con todo este proceso ya estará baneando direcciones ip.

19. DDOSDEFLATE

1.- El primer paso para instalar esta herramienta es descargarla en algún directorio:

```
soporte@webs00:~$ wget https://github.com/jgmdev/ddos-deflate/archive/master.zip
--2017-04-11 12:32:44--  https://github.com/jgmdev/ddos-deflate/archive/master.z
ip
Resolviendo github.com (github.com)... 192.30.253.112, 192.30.253.113
Conectando con github.com (github.com)[192.30.253.112]:443... conectado.
Petición HTTP enviada, esperando respuesta... 302 Found
Ubicación: https://codeload.github.com/jgmdev/ddos-deflate/zip/master [siguiente
]
--2017-04-11 12:32:45--  https://codeload.github.com/jgmdev/ddos-deflate/zip/mas
ter
Resolviendo codeload.github.com (codeload.github.com)... 192.30.253.120, 192.30.
253.121
Conectando con codeload.github.com (codeload.github.com)[192.30.253.120]:443...
conectado.
Petición HTTP enviada, esperando respuesta... 200 OK
Longitud: no especificado [application/zip]
Grabando a: "master.zip"

master.zip              [ <=>                ]  19,22K  --.-KB/s    in 0,1s

2017-04-11 12:32:45 (196 KB/s) - "master.zip" guardado [19682]
```

2.- Una vez descargado el zip de instalación de la herramienta procedemos a instalar el paquete unzip para poder descomprimirlo:

```
soporte@webs00:~$ sudo apt-get install unzip
[sudo] password for soporte:
Leyendo lista de paquetes... Hecho
Creando árbol de dependencias
Leyendo la información de estado... Hecho
Los paquetes indicados a continuación se instalaron de forma automática y ya no
son necesarios.
  linux-headers-4.4.0-62 linux-headers-4.4.0-62-generic linux-headers-4.4.0-64
  linux-headers-4.4.0-64-generic linux-headers-4.4.0-66
  linux-headers-4.4.0-66-generic linux-headers-4.4.0-70
  linux-headers-4.4.0-70-generic linux-image-4.4.0-62-generic
  linux-image-4.4.0-64-generic linux-image-4.4.0-66-generic
  linux-image-4.4.0-70-generic linux-image-extra-4.4.0-62-generic
  linux-image-extra-4.4.0-64-generic linux-image-extra-4.4.0-66-generic
  linux-image-extra-4.4.0-70-generic
Utilice «sudo apt autoremove» para eliminarlos.
Paquetes sugeridos:
  zip
Se instalarán los siguientes paquetes NUEVOS:
  unzip
0 actualizados, 1 nuevos se instalarán, 0 para eliminar y 17 no actualizados.
Se necesita descargar 158 kB de archivos.
Se utilizarán 530 kB de espacio de disco adicional después de esta operación.
Des:1 http://es.archive.ubuntu.com/ubuntu xenial/main amd64 unzip amd64 6.0-20ub
untu1 [158 kB]
Descargados 158 kB en 0s (640 kB/s)
Seleccionando el paquete unzip previamente no seleccionado.
(Leyendo la base de datos ... 252850 ficheros o directorios instalados actualmen
te.)
Preparando para desempaquetar .../unzip_6.0-20ubuntu1_amd64.deb ...
Desempaquetando unzip (6.0-20ubuntu1) ...
Procesando disparadores para mime-support (3.59ubuntu1) ...
Procesando disparadores para man-db (2.7.5-1) ...
Configurando unzip (6.0-20ubuntu1) ...
```

3.- Procedemos a descomprimir el fichero zip descargado:

```
soporte@webs00:~$ unzip master.zip
Archive:  master.zip
26add315ab8d890852cc092c33ec51621d087cc2
   creating: ddos-deflate-master/
  inflating: ddos-deflate-master/ChangeLog
  inflating: ddos-deflate-master/LICENSE
  inflating: ddos-deflate-master/Makefile
  inflating: ddos-deflate-master/README.md
   creating: ddos-deflate-master/config/
  inflating: ddos-deflate-master/config/ddos.conf
  inflating: ddos-deflate-master/config/dependencies.list
  inflating: ddos-deflate-master/config/ignore.host.list
 extracting: ddos-deflate-master/config/ignore.ip.list
  inflating: ddos-deflate-master/install.sh
   creating: ddos-deflate-master/man/
  inflating: ddos-deflate-master/man/ddos.1
   creating: ddos-deflate-master/src/
  inflating: ddos-deflate-master/src/ddos.initd
  inflating: ddos-deflate-master/src/ddos.logrotate
  inflating: ddos-deflate-master/src/ddos.newsyslog
  inflating: ddos-deflate-master/src/ddos.rcd
  inflating: ddos-deflate-master/src/ddos.service
  inflating: ddos-deflate-master/src/ddos.sh
  inflating: ddos-deflate-master/uninstall.sh
soporte@webs00:~$
```

4.- Tras descomprimir el fichero procedemos a entrar en el y ejecutamos el script de instalación:

```
soporte@webs00:~$ cd ddos-deflate-master/
soporte@webs00:~/ddos-deflate-master$ ./install.sh
```

5.- Una vez instalado, veremos algo por el estilo:

```
Installing DOS-Deflate 0.9

Adding: /etc/ddos/ddos.conf... (done)
Adding: /etc/ddos/ignore.ip.list... (done)
Adding: /etc/ddos/ignore.host.list... (done)
Adding: /usr/local/ddos/LICENSE... (done)
Adding: /usr/local/ddos/ddos.sh... (done)
Creating ddos script: /usr/local/sbin/ddos... (done)
Adding nan page... (done)
Adding logrotate configuration... (done)


Setting up init script... (done)
Activating ddos service... (done)

Installation has completed!
Config files are located at /etc/ddos/

Please send in your comments and/or suggestions to:
https://github.com/jgmdev/ddos-deflate/issues

soporte@webs00:~/ddos-deflate-master$
```

6.- A continuación nos dirigiremos al fichero de configuración de la herramienta:

```
soporte@webs00:~/ddos-deflate-master$ nano /etc/ddos/ddos.conf
```

7.- Una vez dentro modificamos lo que está señalado en la captura sustituyendo el correo por el vuestro y la ip por la ip que tenga asignada la máquina:

```
  GNU nano 2.5.3        Archivo: /etc/ddos/ddos.conf              Modificado

# Paths of the script and other files
PROGDIR="/usr/local/ddos"
SBINDIR="/usr/local/sbin"
PROG="$PROGDIR/ddos.sh"
IGNORE_IP_LIST="ignore.ip.list"
IGNORE_HOST_LIST="ignore.host.list"
CRON="/etc/cron.d/ddos"
# Make sure your APF version is atleast 0.96
APF="/usr/sbin/apf"
CSF="/usr/sbin/csf"
IPF="/sbin/ipfw"
IPT="/sbin/iptables"

# frequency in minutes for running the script as a cron job
# Caution: Every time this setting is changed, run the script with --cron
#          option so that the new frequency takes effect
FREQ=1

# frequency in seconds when running as a daemon
DAEMON_FREQ=5

# How many connections define a bad IP? Indicate that below.
NO_OF_CONNECTIONS=150

# The firewall to use for blocking/unblocking, valid values are:
# auto, apf, csf, ipfw, and iptables
FIREWALL="auto"

# An email is sent to the following address when an IP is banned.
# Blank would suppress sending of mails
EMAIL_TO="david@hardsoftsecurity.es"

# Number of seconds the banned ip should remain in blacklist.
BAN_PERIOD=600

# Connection states to block. See: man netstat
CONN_STATES="ESTABLISHED|SYN_SENT|SYN_RECV|FIN_WAIT1|FIN_WAIT2|TIME_WAIT|CLOSE_$

# Only check on the incoming connections to test which to ban (currently only f$
# on outgoing and incoming connections)
ONLY_INCOMING=false

# The external ipv4 address. Used to rewrite the 0.0.0.0 address to the HOST_IP$
# on a 0.0.0.0 socket is shown as connected to the external interface. If you h$
# enter your machine's ip here.
HOST_IP="192.168.21.92"
```

8.- Tras haber modificado este fichero nos dirigiremos al demonio de la herramienta para modificar lo siguiente marcado en la captura:

```
soporte@webs00:~/ddos-deflate-master$ sudo nano /usr/local/ddos/ddos.sh
```

9.- Una vez dentro del fichero modificamos lo marcado en la captura:

```
 GNU nano 2.5.3        Archivo: /usr/local/ddos/ddos.sh          Modificado

#!/bin/bash
################################################################################
# DDoS-Deflate version 0.9 Author: Zaf <zaf@vsnl.com>                          #
################################################################################
# Contributors:                                                                #
# Jefferson González <jgmdev@gmail.com>                                        #
# Marc S. Brooks <devel@mbrooks.info>                                          #
################################################################################
# This program is distributed under the "Artistic License" Agreement           #
#                                                                              #
# The LICENSE file is located in the same directory as this program. Please    #
# read the LICENSE file before you make copies or distribute this program      #
################################################################################
```

10.- Después de todo esto iniciamos el servicio de DDoSReflate:

```
soporte@webs00:~/ddos-deflate-master$ sudo /etc/init.d/ddos start
Starting ddos daemon...
soporte@webs00:~/ddos-deflate-master$
```

11.- Ahora vamos a realizar desde una máquina Kali Linux un ataque de denegación de servicio con la herramienta slow loris:

```
root@hs:~/Escritorio/slowloris.pl-master# ./slowloris.pl -dns www.
```

12.- Una vez lanzado el ataque veremos algo parecido a lo siguiente:

```
Welcome to Slowloris - the low bandwidth, yet greedy and poisonous HTTP client b
y Laera Loris
Defaulting to port 80.
Defaulting to a 5 second tcp connection timeout.
Defaulting to a 100 second re-try timeout.
Defaulting to 1000 connections.
Multithreading enabled.
Connecting to www.grupontt.com:80 every 100 seconds with 1000 sockets:
                Building sockets.
                Building sockets.
                Building sockets.
                Building sockets.
                Building sockets.
                Building sockets.
                Building sockets.
                Building sockets.
                Building sockets.
                Building sockets.
                Building sockets.
                Building sockets.
                Building sockets.
                Building sockets.
                Building sockets.
                Building sockets.
                Building sockets.
                Building sockets.
                Building sockets.
                Building sockets.
                Building sockets.
                Building sockets.
                Building sockets.
```

13.- Después vamos al servidor para revisar el log de baneos para ver el efecto que ha tenido este ataque sobre el servidor, la herramienta lo que hará es bloquear esta conexión durante 600 segundos poniendo la dirección ip en la lista negra denegando todas las conexión que vayan hacia nuestro servidor:

```
soporte@webs00:/var/www/html/prueba$ sudo cat /var/log/ddos.log
[2017-04-11 12:41:00] daemon started
[2017-04-11 12:53:30] added cron job
[2017-04-11 12:55:02] added cron job
[2017-04-11 12:56:01] daemon stopped
[2017-04-11 12:56:01] daemon started
[2017-04-11 12:57:17] banned [redacted] with 154 connections for ban period 6
00
```

20. FORTIFICACIÓN DE ENTORNO LAMP

Lo que vamos hacer en esta sección es endurecer los servicios de MYSQL, PHP y APACHE.

a. MYSQL

1.- El primer paso para asegurar este servicio es establecer la directiva "bind-address=127.0.0.1", esto hará que la base de datos solo escuche peticiones de la propia y no peticiones externas.

2.- Para evitar fuga de información del sistema mediante un ataque SQLi, procederemos a modificar el fichero "/etc/mysql/my.cnf", añadiendo los siguientes parámetros:

…

[mysqld]

…

Local-infile = 0

Secure-file-priv = /dev/null

…

3.- Renombrar el usuario root, para hacer la labor del atacante más difícil en identificar el usuario administrador:

Update mysql.user set user="David" where user="root";

Flush privileges;

4.- Comprobar la existencia de usuarios anónimos, en caso de existir eliminarlos:

Select user usuarios, host from mysql.user where user="";

5.- Comprobar los permisos de los usuario y solo darles permiso para lo que vayan hacer, nunca darles más de lo que van a usar, con el siguiente comando podréis comprobar los permisos de vuestros usuarios:

Show grants for 'usuario'@'localhost';

6.- Ejecutar siempre el siguiente script de instalación ya que elimina posibles configuraciones erróneas:

Mysql_secure_installation

b. PHP

1.- Evitar que PHP nos informe de su versión, nos dirigiremos al fichero "/etc/php/apache2/php.ini" y modificaremos lo siguiente:

Expose_php = off

2.- Para deshabilitar la salida de errores el parámetro "display_errors debe estar como a continuación:

Display_errors = off

3.- Limitar el rango de acción de PHP a un directorio:

Open_basedir = /var/www

4.- Evitar la ejecución de funciones, para que en caso de que un atacante subiese una Shell no pudiese ejecutarla:

Disable_functions = phpinfo, system, exec, Shell_exec, ini_set, dl, eval

5.- Deshabilitar RFI:

Allow_url_fopen = off

Allow_url_include = off

c. APACHE

1.- Controlar el usuario y grupo de apache:

User www-data

Group www-data

2.- Deshabilitar módulos innecesarios, esto se hará con la herramienta a2dismod

3.- Deshabilitar información ofrecida por el servidor, nos dirigimos a "/etc/apache2/conf.d/security":

dejamos ServerTokens ProductOnly y ServerSignature Off.

21. AUDITORÍAS DE SEGURIDAD

Para las auditorías del estado de la red se utilizarán las siguientes herramientas:

Para el footprinting:

- CentralOps (http://centralops.net/co/) : Esta herramienta nos permite realizar un footprint físico del servidor donde se aloja la web de la víctima.
- Extracción de metadatos con Evil Foca.
- DNSenum para sacar información del DNS.
- TheHardvester, para sacar posibles correos electrónicos.
- Nmap, para el escaneo de puertos y servicios.
- Shodan para la información pública del servidor.

Para el escaneo de vulnerabilidades:

- Nmap, para escanear vulnerabilidades.
- Nessus, para escanear vulnerabilidades.

Para la explotación de vulnerabilidades:

- Metasploit
- Exploit-db

Tras la recogida de información, escaneo de vulnerabilidades y explotación le sigue la documentación, esta documentación se tiene que explicar toda la metodología seguida, herramientas utilizadas y técnicas utilizadas. Por motivos de seguridad en este documento no se mostrará el documento resultante de la auditoría de seguridad realizada.

22. MEJORAS

Tras tener en producción todas estas herramientas, se han llegado a conclusiones para asegurar cada vez más la seguridad de la red, una de las primeras mejoras que se van a implementar son las VLAN, posteriormente también se montara un Security Center de Kaspersky para el control de antivirus de sistemas Windows, Linux y también para solucionar su vulnerabilidades.

También se llevará una mejora a largo plazo que la realización de redundancia de Gateway mediante el protocolo BGP, esta última todavía no podemos abarcarla debido a su nivel de conocimientos necesarios, aparte de los requisitos previos que son necesarios para poder obtener direcciones, es decir nuestro propio rango de direcciones IP dentro de “Internet”.

Tras debatir todas estas mejoras, se ha llegado a la conclusión que una vez que estén en producción todas las mejoras, solo será necesario mantenerlas y estar al día con las últimas tecnologías y nuevos tipos de ataque para poder implementar nuevas mejoras que puedan evitar estos ataques.

23. CONCLUSIÓN

Tras estar un tiempo trabajando con todas estas herramientas me he dado cuenta que todas se completan entre sí, es decir ninguna hace la competencia a la otra, ya que estas trabajar en armonía, endurecen y dan seguridad a la red.

También tengo que comentar que al principio era difícil implementar todos estos servicios bien, pero cuando está todo perfectamente colocado, todo trabaja como si fuese un reloj suizo.

Para mí esta experiencia de poder desplegar un sistema de seguridad tan amplio me ha dado a entender la realidad de la importancia de la seguridad dentro de las redes empresariales, ya que si en caso de que un "ciberdelincuente" consiguiese penetrar los sistemas las perdidas en información o en servicio serian increíbles, por eso cada día este sector de la informática está en auge.

24. FUENTES

https://joanesmarti.com/tu-propio-ids-con-snort-y-snorby-en-linux-debian-7/

https://www.youtube.com/watch?v=EeqDKmZdtkY

https://www.youtube.com/watch?v=LGtJK9xGq-4

https://www.youtube.com/watch?v=V05WfhmCW_M&t=2s

https://www.fail2ban.org/wiki/index.php/Main_Page

https://github.com/jgmdev/ddos-deflate

y muchas más que no he podido recuperar.