Trabajo
Final de Grado

Facultad de Ingeniería en
Telecomunicaciones

Tema:

IMPLEMENTACIÓN DE UN SISTEMA AUTÓNOMO Y REDISEÑO DE LA RED DE DATOS

Matias E. Lauretta Saad

Tutor: Enrique G. Banchio
Ignacio Segura

Año 2018

Dedicatoria

A la paciencia y confianza en mí de mujer y mi familia.

Matias E. Lauretta Saad

Agradecimientos

A mi mujer por su confianza y apoyo.

A mi familia que siempre me alentó a terminar mis estudios.

A mis amigos y compañeros de la facultad que me acompañaron todos estos años.

Matias E. Lauretta Saad

Resumen

El auge de las tecnologías de internet conlleva a un incremento del uso de recursos de red lo que llevó al Instituto a tramitar la asignación de un número de sistema autónomo con el pool de direcciones IP asociadas. Siendo responsable de la propagación de estas direcciones y sus reglas de ruteo.

Por consiguiente, resulta necesario un replanteo de la red del instituto que satisfaga las necesidades de los usuarios actuales y se provea de herramientas de administración de red que facilite la gestión de tráfico para los servicios relevados y la seguridad de la red.

Con el presente trabajo se hace un relevamiento de las herramientas disponibles, servicios que brinda la red, necesidades futuras y se realiza un re-diseño lógico de la red y una propuesta de configuración para la herramienta de gestión y seguridad

Glosario

ISP: Proveedor de Servicios de Internet o ISP (del inglés Internet Service Provider)

AS: Sistema Autónomo o AS (del inglés Autonomous System)

ASN: Número de Sistema Autónomo o ASN (del inglés Autonomous System Number)

LACNIC: Registro de Direcciones de Internet para América Latina y Caribe (del inglés Latin American and Caribbean Internet Addresses Registry)

MAN: Metropolitan Área Network, Red de Área Metropolitana

WAN: Wide Area Network, Red de Área Amplia

LAN: Local Area Network. Red de área local

MAC: Media Access Control

IP: Internet Protocol

VLAN: Virtual Local Area Network

WAN: Wide Area Network

BGP: Border Gateway Protocol

TCP: Transmission Control Protocol

EIGRP: Enhanced Interior Routing Protocol

SNMP: Simple Network Management Protocol

SSH: Secure Shell

SO: Sistema Operativo

Índice de contenidos

Índice de Tablas

Índice de figuras e imágenes

Capítulo 1- Introducción

Los protocolos de propagación de rutas permiten que un router intercambie rutas de propagación con otro. Sin embargo este funcionamiento no puede escalarse a todo Internet ya que la cantidad de routers que la componen llevaría a que cada uno de ellos intente intercambiar rutas con todos los demás routers provocando tanto tráfico de red que colapsaría Internet. Para limitar este tipo de tráfico es que se divide Internet en grupos. Cada grupo comparte información de rutas y luego al menos un router dentro de cada grupo recolecta esta información y la envía a los demás grupos. Estos grupos se definen como Sistemas Autónomos.

Un Sistema Autónomo es un grupo de redes y equipos de red que son gestionados por uno o más administradores de red que poseen una clara y única política de ruteo. Cada Sistema Autónomo tiene un número asociado el cual es usado como un identificador para el SA en el intercambio de información de rutas.

El Registro Regional de Direcciones de Internet para América Latina y Caribe asignó al Instituto un número de Sistema Autónomo junto con un rango de direcciones IP. En este proceso, dicho Ente informa en su Manual de políticas en el apartado 2.3.2.9. “Encaminamiento no garantizado” que las direcciones otorgadas no están garantizadas de ser globalmente ruteables y esto deberá ser solucionados entre el Instituto y sus proveedores de conectividad, así el Instituto se responsabiliza de la propagación de estas direcciones y de las reglas de ruteo asociadas para que el SA esté accesible a través de Internet. Existen tres tipos de sistemas autónomos, aquellos que se conectan a un único SA, aquellos que se conectan a más de un SA y permiten el tránsito de paquetes entre ellos y aquellos sistemas autónomos que se conectan a más de un SA pero no permiten el tránsito de paquetes, siendo Stub, De Tránsito o multihomed respectivamente

La propagación de las rutas se hace mediante un protocolo de puerta de enlace de borde o BGP por sus siglas en inglés, los cuales pueden agruparse dentro de dos categorías, protocolos interiores y protocolos exteriores a la puerta de enlace, IBGP y EBGP respectivamente por sus siglas en Inglés. En la siguiente imagen los routers 1, 2 y 3 pertenecen a un SA y los routers 4, 5 y 6 pertenecen a otro SA. Para el intercambio de rutas dentro del SA se utilizan protocolos interiores de puerta de enlace y para el intercambio de información entre sistemas autónomos utilizan protocolos exteriores de puerta de enlace.

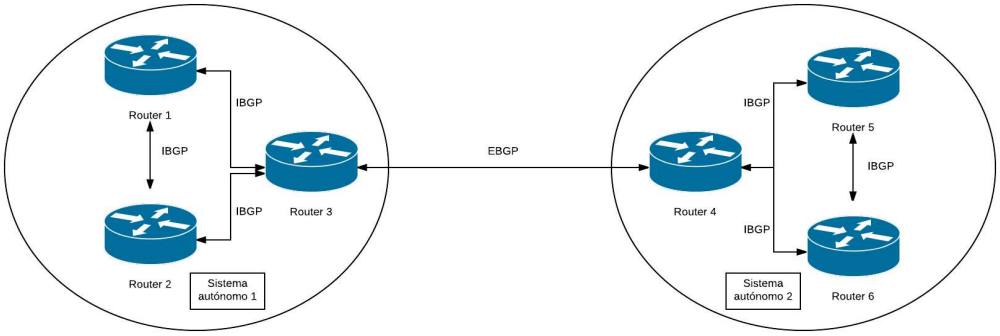


Figura 1: Ilustración de protocolos de propagación usando IBGP en cada uno de sistema autónomo y EBGP entre ambos sistemas autónomos.

Intuitivamente se puede asumir que los protocolos deberían permitir encaminar el tráfico de red por un trayecto óptimo de entre todos los posibles, pero no existe un consenso de requisitos que engloben todas las diferentes aplicaciones dentro de una métrica de optimización. Para algunas aplicaciones el mejor camino es el de mayor ancho de banda mientras que para otras puede ser el de menor retardo. La mayoría de los protocolos IBGP utiliza como métrica de ruta para determinar cuál camino utilizar la combinación de costos administrativos y cantidad de saltos. Cantidad de saltos se define como la cantidad de routers, o redes intermedias, que debe atravesar y el costo administrativo es un valor que ingresa manualmente el administrador de red para controlar la ruta a seleccionar. Mayormente el costo administrativo se utiliza para cumplimentar con políticas de encaminamiento de la red. Como contraparte, para los EBGP no es posible la utilización de métricas por la falta de estandarización entre SA sobre qué característica del trayecto utilizar como métrica, es por eso que solamente pueden informar de la existencia de un camino u otro.

Dentro de los EBGP se destaca el protocolo de borde de puerta de enlace, BGP por sus siglas en Inglés, por su amplia utilización como protocolo EBGP de Internet, actualmente se encuentra en la versión 4 del mismo y entre sus características principales se destaca que las rutas están determinadas por los SA que se debe atravesar sin tener que informar sobre los routers dentro de cada uno de los SA, permite la implementación de políticas que habilitan al administrador a determinar qué rutas son comunicadas fueras del SA y cuáles no.

El protocolo de información de encaminamiento, RIP por sus siglas en inglés, y el protocolo del primer camino más corto, OSPF por sus siglas en inglés, son los protocolos predominantes dentro de un sistema autónomo. RIP es uno de los primeros protocolos IBGP usados en Internet, utiliza como

métrica la cantidad de redes intermedias entre el origen y el destino, contabilizando como 1 salto si son redes adyacentes, siendo su principal ventaja la simplicidad, el administrador de red con iniciarlo en cada router y habilitar el tráfico de mensajes de difusión entre routers permite que todos los routers del SA en poco tiempo obtengan rutas a todos los destinos. Su principal desventaja es que cada paquete RIP puede llegar a consumir muchos ciclos del procesador por lo cual en sistemas autónomos de gran tamaño los cambios dentro de la red se propagan con cierta lentitud, es en la necesidad de un protocolo que permita administrar rutas en grandes organizaciones que se basa OSPF siendo esta su ventaja. Las principales características del OSPF son la posibilidad de que un router incluya rutas obtenidas por medio de otros protocolos y la posibilidad de agregar un costo administrativo a cada ruta.

Una red de datos es la interconexión de dispositivos de red con el objetivo de compartir los recursos, información y servicios. Esta interconexión se puede llevar a cabo física, lógica o inalámbricamente. Tradicionalmente las redes se clasifican por su localización o área que abarcan, dividiéndose en Redes de Área Amplia o WAN, Redes de Área Metropolitana o MAN y Redes de Área Local o LAN, pero también se pueden clasificar mediante su topología, relación entre los dispositivos conectados a la misma o el método de interconexión de los dispositivos de la red. La conexión de red puede ser física o virtual, mediante el empleo de un software se puede llevar a cabo una conexión de red lógica permitiendo que computadoras pertenezcan a la misma red conectándose a recursos de redes diferentes entre sí para el establecimiento de la conexión lógica. Estas se pueden categorizar en Redes Privadas Virtuales o VPN y Red Virtual de Área Local o VLAN.

Las Redes Virtuales de Área Local son un grupo de dispositivos que comparten un objetivo o función común sin considerar otras características como la ubicación física o geográfica, configurados para comunicarse como si estuvieran conectados a la misma red física. Esta agrupación lógica de capa 2 del modelo OSI no permite que dos VLAN diferentes se comuniquen entre sí sin que sean ruteadas por un dispositivo de capa 3. Son posibles mediante software instalado en los switch de la red que permiten la conexión mediante el insertado de una etiqueta de VLAN en los paquetes pertenecientes a cada red virtual cuando múltiples switch pertenecen a la misma red.

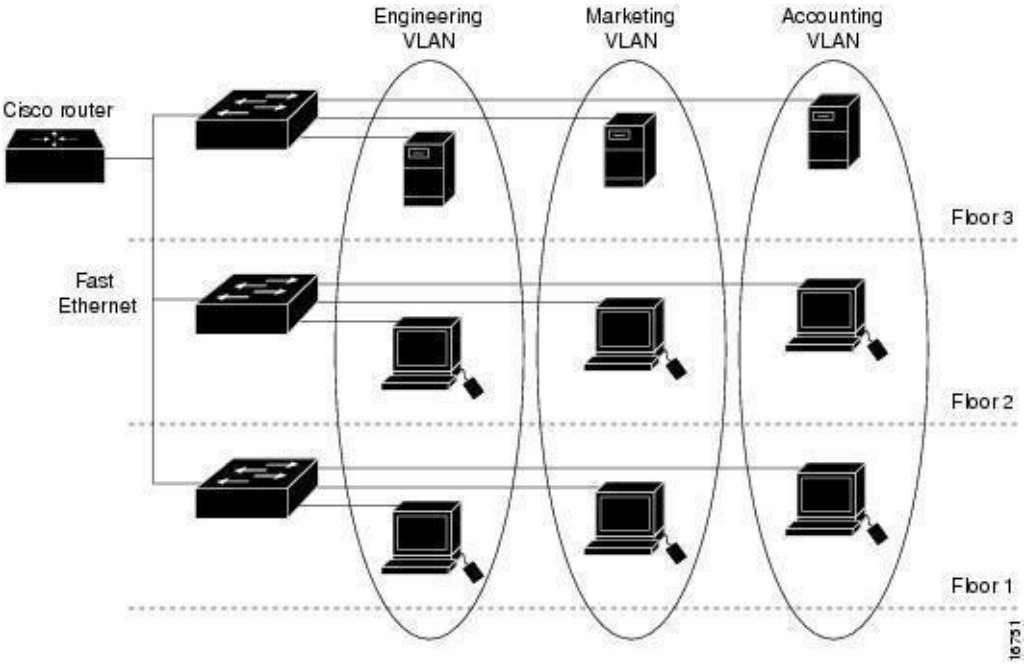


Figura 2 - Diagrama esquemático de 3 VLANs - Fuente: http://www.cisco.com/c/dam/en/us/td/i/000001-100000/15001-20000/16501-17000/16751.ps/_jcr_content/renditions/16751.jpg

La seguridad en las redes es un aspecto que cobra cada vez mayor importancia donde los firewalls juegan un rol clave siendo el primer nivel de seguridad de toda red. Un firewall es un sistema que permite la separación lógica e implementa una política de seguridad entre una red confiable y una red no confiable, Internet. Existen varios tipos de firewalls y configuraciones de estos sistemas, que podemos agruparlos en cuatro categorías: firewall de filtrado de paquetes, Dual-Homed Host, Screened Host y Screened Subnet. Un firewall de filtrado de paquetes es la implementación más simple y se logra al implementar un dispositivo que filtra paquetes desde el exterior a la red interna, la red interna tiene acceso al exterior directo. Un nivel extra de seguridad es la configuración Dual-Homed Host donde una máquina con dos o más placas de red hace de intermediaria entre la red externa e interna, en esta configuración es importante que el sistema no enrute los paquetes para mantener separadas las redes internas y externas. Avanzando en cuanto a la complejidad de la arquitectura y los niveles de seguridad se encuentra la configuración Screened Host donde se combinan un router implementando un simple filtrado de paquetes y listas de acceso junto con un sistema que ejecuta los servicios de proxy y un filtrado de paquetes, servicios y conexiones más complejo.

Screened Subnet es la configuración más segura y ampliamente

utilizada en la actualidad, consta de dos router, uno interno y otro externo, una máquina bastión entre ambos routers donde se implementan otro nivel de seguridad y una subred accesible desde el exterior conocida como zona desmilitarizada o DMZ por sus siglas en inglés. El router externo hace un primer filtrado de paquetes desde afuera hacia adentro, luego la máquina bastión implementa otro nivel de filtrado y por último el router interno para todos aquellos paquetes provenientes de afuera de la red, mientras que el proceso para paquetes salientes de la red interna hacia la externa se hace en orden inverso. El nivel extra de seguridad lo ofrece la subred accesible desde el exterior, ya que para poder entrar a la red interna los atacantes deberán acceder a esta, luego a la máquina bastión y luego a la red interna, mientras que en las otras configuraciones, logrando acceder a la máquina bastión obtienen acceso completo a la red interna.

El filtrado de paquetes es el proceso de inspección de la cabecera de los paquetes que atraviesan el firewall y tomando una decisión de aceptar, denegar o rechazar el tráfico basado en reglas definidas por el administrador de red. Estos firewall trabajan a nivel de capa 3 y 4 del modelo TCP/IP donde se trabaja con las cabeceras de los paquetes sin llegar a analizar los datos del mismo. La traducción de direcciones de red y de puertos, NAT y NAPT por sus siglas en Inglés respectivamente, es otra de las funciones que se implementan en el firewall y lo que se realiza es la alteración de la cabecera del datagrama modificando la dirección origen o destino del mismo debiendo realizarse el proceso inverso en los paquetes de respuesta. Esta alteración tiene por objetivo el redireccionamiento de los paquetes a una máquina concreta en función del servicio utilizado o requerido, también se utiliza para que múltiples equipos de una red privada accedan a internet a través de una única dirección pública.

Problemática

El instituto cuenta con tres métodos de enseñanza y dictado de clases, presencial, semipresencial y a distancia, de los cuales los últimos dos se basan en servicios web ofrecidos en formato de tutorías y clases virtuales que complementan los materiales físicos y tutorías presenciales para lo cual necesita garantizar la accesibilidad y disponibilidad de los recursos de red que proveen estos servicios.

El instituto en la actualidad tiene el control sobre la asignación de las direcciones IPv4 públicas de su propiedad pero son publicadas por el ISP siendo este el responsable de propagación de las rutas de encaminamiento desde y hacia el Instituto. A su vez una nueva conexión a otro ISP está en proceso de completarse con lo que es imperativo obtener el control de esas reglas de ruteo internas.

La red de datos del Instituto y las necesidades por las cuales se fundamentó la topología y configuración presenta desventajas para su uso actual por lo que el primer inconveniente se presenta en la división de subred y la asignación de cada host a su respectiva subred.

A su vez, la gestión de tráfico y seguridad resulta una tarea compleja que requiere de mucho tiempo al administrador de la red el poder implementar un cambio en la configuración.

Principales Objetivos

Este trabajo tendrá como principales objetivos relevar la red actual y analizar los requerimientos de la red futura e investigar los diferentes protocolos de ruteo y propagación de rutas para la confección de una propuesta de implementación que abarque un re-diseño optimizado de la red interna dentro de un Sistema Autónomo que abarque las nuevas exigencias y necesidades relevadas junto con una herramienta de gestión de tráfico y seguridad.

Capítulo 2- Relevamiento de la red actual

Durante el relevamiento se recopiló información sobre la estructura y funcionamiento de la red pero también se relevaron las necesidades, problemas y limitaciones actuales como también los diferentes servicios y necesidades posibles en un futuro cercano que debían tenerse en cuenta para la realización de este trabajo final de grado.

Del estado actual de la red y su estructura.

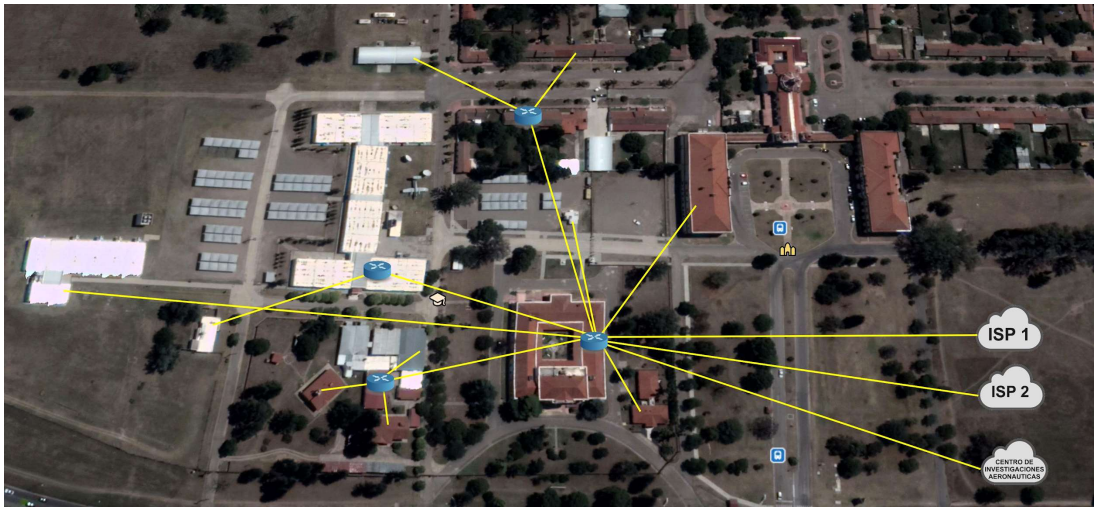
El relevamiento de la red de datos actual del Instituto arroja que fue definida hace más de 15 años y su arquitectura está basada en la topología del campus del Instituto de ese momento, se divide en **19** VLANs, cada una de ellas está asignada a un edificio o grupo de edificios. Desde el edificio central donde llega la única conexión a Internet y donde se encuentra el backbone, sala de servidores y demás infraestructura del núcleo de la red, se extiende a cada uno de los edificios por enlaces de fibra óptica en las conexiones Router-Switch y para los vínculos switch-usuario por medio de cable UTP cat 5/6 o inalámbricas. A su vez la red está dividida en subredes, las cuales están nuevamente asignadas basándose en la topología de red salvo algunas excepciones donde se hizo una distinción entre los distintos departamentos dentro de una misma subred, lo que lleva a que diferentes áreas deban compartir servicios y restricciones genéricas y no específicas para el desarrollo de sus tareas debido a las configuraciones que se realizaron en los switches por pertenecer a la misma VLAN, en otras palabras si dos departamentos están en la misma subred y uno debe tener acceso a un servidor interno el otro departamento también obtiene acceso a ese servidor por más que no sea requerido salvo que se configure una restricción de acceso específica para quienes puedan o no accederlo, que en muchas situaciones no está contemplado. En esta parte del relevamiento se identifica un inconveniente en cuanto a el dominio de broadcast de cada subred, que es más grande del necesario afectando la performance de la red con tráfico no necesario en algunas ocasiones y a su vez la necesidad de realizar una nueva subdivisión de la red, que permita aplicar reglas de filtrado de paquetes específicas por subred y no por usuario, contemplando las excepciones que se puedan generar en usuarios puntuales.

Otro aspecto relevado es la utilización de recursos de red que son distribuidos equitativamente entre todos los usuarios sin distinción de los servicios, salvo algunas limitaciones en el ancho de banda disponible para aquellas VLAN que pertenecen a la red inalámbrica pública y algunos bloqueos a sitios específicos para todos los usuarios de la red, no basándose en las necesidades de cada usuario o grupo de ellos sino en aquellos sitios

que, por los servicios ofrecidos, consumen grandes porciones del ancho de banda disponible como son aquellos que permiten almacenamiento en la nube. Se detecta la necesidad de una mejor distribución de los recursos que garanticen el correcto funcionamiento de aquellas funciones prioritarias del Instituto como a su vez la previsión de servicios del tipo de tiempo real y una configuración de firewall que fije limitaciones en el uso de estos servicios a quienes no lo necesitan para sus tareas.

El sistema de firewall actual se encuadra en la configuración screened subnet donde el vínculo a la red externa llega a un switch de capa 2/3 HPE 5500 EI, Switch que cumple la función de router externo y se conecta con un switch de capa 2/3 3COM que ofrece la división lógica del mísmo configurado como si fueran dos switches independientes, la máquina bastión corriendo OpenSUSE en su versión 12.3 con un procesador Intel Xeon y 32GB de memoria RAM que a su vez ejecuta el servidor Proxy cuenta con dos interfaces de red y es el vínculo físico entre la parte pública del switch de capa 2/3 3 COM y la parte privada. La zona desmilitarizada cuenta con **detalle de los servidores de la DMZ**.

Figura 3: Imagen satelital del instituto con la topología de red superpuesta



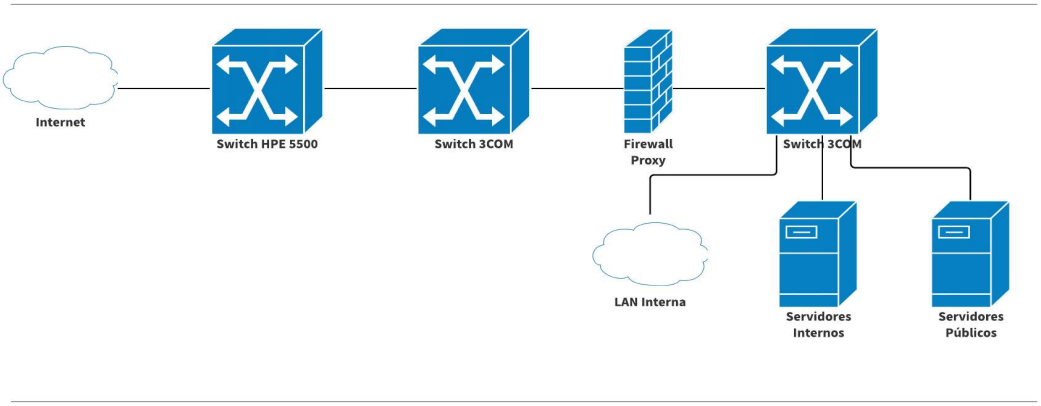


Figura 4: Topología del backbone y firewall del instituto

En este esquema la seguridad de la red se encuentra distribuida a lo largo de los diferentes switches y el firewall ejecutándose en la máquina bastión junto con el servidor proxy, la gestión de la seguridad es llevada de forma manual y distribuida en etapas, Switch externo, Switch interno y Firewall, cada uno con sus reglas particulares que muchas veces llevan a un conflicto entre ellas por no estar centralizada su gestión y administración. Junto con estas reglas, también se aplican listas de control de acceso, ACL por sus siglas en Inglés en los switches, un squid proxy web en configuración transparente que filtra el acceso a determinados sitios web que utilizan el protocolo de transferencia de hipertexto, HTTP por sus siglas en Inglés, como otro punto de control y filtrado de paquetes. Todas las reglas que se ejecutan en el bastión se configuran y mantienen editando directamente sobre los archivos de texto en forma de tabla que permite el módulo IPTables

De las necesidades y requerimientos actuales.

Uno de los servicios fundamentales del instituto es el de gestión académica, actualmente llevado a cabo con la herramienta SIU-Guaraní, que brinda la administración de las tareas académicas tanto para docentes y alumnos como para el personal administrativo y las autoridades. Este servicio, alojado en los servidores del instituto que se encuentran en la DMZ, siendo accedido localmente para los usuarios que están conectados directamente a la red del Instituto y desde el exterior, a través del portal dispuesto para tal fin. Debe estar disponible para su acceso en todo momento. Otros de los servicios de gran importancia son las tutorías y aulas virtuales, donde la mayor parte del alumnado se desarrolla académicamente vinculandose con sus profesores y son parte básica y fundamental para las modalidades de dictado de clases a distancia y semipresencial.

Las jornadas de puertas abiertas, ferias de empresas, actividades de extensión y seminarios entre otras actividades que congregan en los diferentes edificios destinados para tales fines a un gran número de personas que requieren de acceso a servicios de red internos y conexión a internet muchas veces congestionan la red. En este aspecto debe tenerse en cuenta una propuesta que permita la optimización de los recursos de forma flexible de rápida administración, evitando el colapso de la red pero aun así permitiendo las conexiones y el tráfico.

Se detectó la necesidad actual de una de las facultades de poder tener acceso a diferentes redes sociales, facebook messenger y google hangouts principalmente, para llegar a los alumnos para la realización de actividades que alumnos de otras carreras o de la Facultad de Ingeniería no requieren. Este requerimiento está alineado con un posible requerimiento de un área de comunicaciones institucionales, comunicaciones públicas o de marketing que gestiones las redes sociales del instituto. Los servicios de llamadas y video llamada de Whatsapp, otra red social, son utilizados por las autoridades y docentes para realizar colaboraciones y consultas con otros Institutos, Universidades y Facultades de Argentina y el mundo. De la consideración de esta necesidad se desprende otra de las necesidades actuales, la de tener la capacidad de realizar videoconferencias sin la necesidad de grandes configuraciones previas debido a las diferentes configuraciones y requerimientos de que suelen tener para poder realizarlas debido a los puertos utilizados, los redireccionamientos de los mismos y los conflictos habituales que se suceden con los servidores proxy y el firewall.

Por último, la necesidad de implementar un nuevo enlace con un ISP diferentes es de suma importancia y urgencia. La importancia radica en la posibilidad de contar con un camino alternativo de salida de paquetes a Internet, que permite un balanceo del tráfico saliente de red y la continuidad de los servicios en caso de que alguno de las dos conexiones falle sin mencionar que se incrementa el ancho de banda disponible.

De las necesidades y requerimientos futuros.

Las principales necesidades que se presentarán en un futuro cercano, al entender del autor de este TFG, están relacionadas con la mejora del hardware e infraestructura del Instituto, entre las que considero principales están el sistema de firewall y el reemplazo de los switches externo e internos por routers específicos para estas funciones y reemplazar la central telefónica actual por una del tipo IP.

Estas dos necesidades responden a la necesidad principal de que si se mantiene la tendencia en aumento del uso de recursos de red, interno y externos, sumado a el objetivo del Instituto de incrementar el volumen que

compone el alumnado el hardware que soportará la infraestructura de red se verá obsoleto frente a los requerimientos y la velocidad de procesamiento de las solicitudes afectará negativamente sobre el funcionamiento de la red.

De las herramientas de monitoreo, gestión y control.

Actualmente el Instituto no cuenta con una herramienta de gestión de tráfico y administración del firewall, el cual se realiza a través de la edición directa de los archivos de configuración, necesidad imperante dentro del departamento de redes. Salvo esta falta de una herramienta que unifique todas las políticas de control de tráfico de red, la división de redes cuenta con varias herramientas de monitoreo y control de la red, todas basadas en el protocolo de simple administración de red, o SNMP por sus siglas en inglés, que opera a nivel de capa de aplicación y permite el intercambio de información entre los dispositivos de red que lo soportan, como es el caso de los dispositivos principales del núcleo de la red relevada.

Las dos herramientas que más destacan son Cacti y Nagios. Si bien ambas soluciones presentan similares funciones, Cacti está orientada en la creación de gráficas y el seguimiento de las funciones de red basadas en gráficos mientras que Nagios está concebida como una herramienta que categoriza los diferentes eventos de la red en estados. De esta forma al ver una anomalía en un gráfico o un estado determinado, cacti o nagios respectivamente, se toman acciones correctivas, preventivas o de la naturaleza que requiera el problema identificado.

Capítulo 3 - Selección de la herramienta de gestión

Del relevamiento surgió una de las necesidades principales del área División de Redes la de poder contar con una herramienta que les permita de forma unificada la gestión de las políticas de tráfico y seguridad de red. Este capítulo describe el procedimiento realizado para la selección de dicha herramienta que se incluye en la propuesta. Para ello se utiliza el concepto de firewall como un sistema que permite la separación lógica e implementa una política de seguridad entre una red confiable y una red no confiable, descrito en el capítulo 1. A su vez, en la misma sección se describen las diferentes configuraciones que este sistema puede tener, y al Instituto poseer una topología Screened Subnet, se desarrollará sobre esta, la cual consta de un Bastión que es responsable de la ejecución de las políticas de seguridad y gestión de tráfico de la red por medio de un software firewall.

El bastión

El Bastión es un equipo que se encuentra accesible desde el exterior de la red, se encuentra expuesto, y permite el acceso seguro a la red interna a aquellas conexiones que cumplan con las políticas de seguridad ejecutadas por el software firewall corriendo en él. Estos software firewall son programas ejecutándose en forma de procesos o demonios que implementan el filtrado de paquetes, aceptan o rechazan las conexiones entrantes y salientes.

Este tipo de firewall es el que se utiliza con mayor amplitud en el mercado para redes de similar tamaño y utilización de recursos a la del Instituto. Dentro de este tipo de firewall el sistema operativo y el hardware del equipo bastión inciden en el rendimiento y performance del sistema, tenemos por ende dos sub-categorías aquellos que son sistemas operativos orientados a funcionar como Firewalls únicamente y aquellos que son un proceso o demonio que se ejecuta dentro de un sistema operativo que no es programado exclusivamente para tal función.

Las opciones para la configuración del bastión son:

- Una máquina con Microsoft Windows utilizando el Firewall por defecto de Windows.
- Una máquina con Microsoft Windows utilizando un Firewall externo.
- Un equipo de firewall físico dedicado.
- Una máquina con una distribución Linux orientada a firewalls
- Una máquina con una distribución Linux regular con el módulo Netfilter y el paquete IPTables

La primera alternativa, una máquina con Microsoft Windows utilizando el Firewall propietario, requiere para la implementación del mismo la creación y configuración de aspectos no estrictamente relativos a las conexiones que se intentan bloquear o permitir, por cómo está escrito el sistema operativo y su interacción con el firewall que se deben realizar, como la creación de un dominio Active Directory, creación de servidores de control de dominio que infiere en mayores recursos, que no son necesarios a tener en cuenta para las otras alternativas. La modificación o actualización del conjunto de reglas es tediosa al no poder aplicar reglas por grupo de políticas y no tiene una interfaz gráfica de configuración y administración amigable, con lo cual la curva de aprendizaje para la correcta configuración es otro motivo que se suma a los costos monetarios por licencias, motivo por el cual no se analizará esta opción

Otro aspecto de porque el Firewall que viene instalado por defecto con Microsoft Windows no se propondrá es su vulnerabilidad ya que el mismo sistema operativo es vulnerable de por sí debido a la forma en que está concebido, Microsoft Windows tiene en un mismo nivel, el Kernel, el espacio de aplicación y el del Usuario haciéndolo más vulnerable que las alternativas ya que el firewall se ejecuta en el nivel de aplicación.

La segunda opción tampoco se considerará por los motivos expuestos ut supra, si bien varios de los problemas encontrados en el firewall predeterminado de Microsoft Windows no se encuentran en uno que se desarrolle externamente al sistema operativo, estos suelen ser costosos y si se contemplan las licencias de uso del sistema operativo no se cumplen los objetivos del proyecto que es el realizar una propuesta que no conlleve una erogación de dinero.

La tercera opción es descartada por el alto costo de la misma pero también por la dificultad asociada al cambio de tecnología, su curva de aprendizaje y los inconvenientes al momento de actualizar el dispositivo cuando su vida útil lo haga obsoleto o la escalabilidad de la red lleve a su máxima capacidad.

La cuarta opción no es la recomendada ni la más utilizada, ya que esta no evita que se deba instalar un software especializado, incluido o no con el sistema operativo, para la creación de las reglas del firewall y su posterior administración, estas distribuciones no están completamente documentadas o soportadas por sus desarrolladores y presentan fallos que las hacen inestables e inseguras, tampoco hay una comunidad de usuarios amplia en la que se puede presentar un problema para la solución comunitaria/contributiva y no son actualizadas con la periodicidad necesaria para un firewall cuando un fallo o problema es detectado en su código.

La quinta opción, una distribución Linux basada en la versión de Kernel 2.4.x o superior que incluye el framework de filtrado de paquetes NetFilter es para esta situación la mejor alternativa. Este módulo mediante la utilización del paquete IPTables define el conjunto de reglas que permiten la “manipulación” de paquetes, dichas reglas son aplicadas al software firewall provisto en el kernel de Linux, que a su vez por cómo está implementado es independiente del espacio de usuario presentando un nivel extra de seguridad en comparación con los sistemas basados en Microsoft Windows. Distribuciones Linux basadas en este kernel se encuentran disponibles en gran variedad entre las que se destacan Ubuntu, OpenSUSE, Debian, CentOS, Fedora, ArchLinux y Mint. Todas estas distribuciones pueden ser adquiridas bajo licencias GNU/GPL y actualmente están basadas en el kernel de Linux versiones 2.4.x o superiores. No vamos a cubrir en este trabajo la totalidad de las distribuciones disponibles ya que está fuera del objetivo de este trabajo final de grado, se expondrán los fundamentos por los cuales se elige OpenSUSE y las ventajas principales frente a las otras distribuciones.

La distribución OpenSUSE es un producto de la compañía SUSE que fue fundada un año después del anuncio de Linux, en 1992, siendo así una de las distribuciones basadas en el kernel GNU/Linux con mayor antigüedad y trayectoria en este sistema operativo, junto con Slackware y Debian. Desde el año 2015 el equipo de OpenSUSE adoptó como código fuente de su distribución el de SUSE Linux Enterprise (SLE por sus siglas) y compiló la versión LEAP beneficiándose de los paquetes y código fuente de un sistema estable en el que los paquetes son probados exhaustivamente antes de estar disponibles para su descarga aunque haya versiones más recientes pero en estado de prueba o desarrollo. Viene con la versión 4.1 de Kernel Linux que tiene un soporte extra en el tiempo. En resumen, esta distribución tiene los mejores aspectos de las demás combinadas, la estabilidad de Debian, la orientación a Servidores de CentOS y la facilidad de uso de Ubuntu e incluye el framework NetFilter. Para la edición, configuración y mantenimiento del conjunto de reglas en el archivo con formato de tabla iptables se presentan diversas herramientas que podemos agrupar en dos conjunto de acuerdo a la forma de acceder y editar este archivo que puede ser a través de línea de comando, CLI por sus siglas en inglés, o por medio de una interfaz gráfica de usuario, GUI por sus siglas en inglés. De entre estos dos conjuntos se opta para la configuración inicial y puesta en funcionamiento por aquellos softwares que incluyen una GUI los cuales proporcionan convenciones de uso que permiten obtener el dispositivo en funcionamiento rápidamente, permiten el manejo de un amplio rango de dispositivos por medio de la compatibilidad que proporcionan evitando la demora del aprendizaje de un nuevo lenguaje de configuración por CLI y su sintaxis por cada uno de los dispositivos de la red que necesiten configuración y mantenimiento.

Filtrado de paquetes en Linux

En linux el filtrado de paquetes está integrado en el núcleo del sistema operativo, conocido como Kernel, donde actualmente se encuentra el módulo Netfilter integrado. Esta interfaz o framework cumple la función de firewall de filtrado de paquetes y viene provista del módulo IPTables para establecer las reglas en archivos del tipo tablas y puede utilizarse para la manipulación de estas reglas una herramienta de gestión o edición directa sobre los archivos.

Netfilter realiza la gestión del filtrado de paquetes mediante los archivos de tablas que se organizan en cadenas, estas contienen las reglas que se evalúan secuencialmente hasta que se encuentre una que cumpla con la condición y se ejecute la acción asociada, caso contrario se ejecuta la acción por defecto.

El administrador de red puede crear cadenas y tablas propias, pero a los fines de entender el proceso de filtrado utilizaremos las que incluye el framework Netfilter por defecto. A saber el entorno contiene cuatro tablas: filter, mangle, nat y raw donde se encuentran cinco cadenas: INPUT, OUTPUT, FORWARD, PREROUTING y POSTROUTING. Es importante aclarar que no todas las cadenas están presentes en todas las tablas.

Descripción de las cadenas:

- INPUT: Incluye las reglas que se comparan con los paquetes a la entrada de la interfaz, filtrado de tráfico entrante, en base a estas reglas determina la acción a realizar sobre el paquete.
- OUTPUT: Incluye las reglas que se comparan con los paquetes a la salida de la interfaz, filtrado de tráfico saliente, en base a estas reglas determina la acción a realizar sobre el paquete.
- FORWARD: Contiene las reglas de reenvío de paquetes de una interfaz a otra.
- PREROUTING: Es la primer etapa en Netfilter. Determina la primera acción a realizar antes de que el paquete entre en el sistema basándose si el destino es local o no.
- POSTROUTING: Es la última etapa en el filtrado, determina la acción a realizar antes de enviar el paquete a la interfaz destino.

Descripción de las tablas:

- FILTER: Es la tabla predeterminada y se utiliza como su nombre lo indica para el filtrado de paquetes, está compuesta por las cadenas INPUT, OUTPUT Y FORWARD.
- MANGLE: Es la tabla que controla el proceso de modificar o no el contenido y las opciones de los paquetes. Están presentes todas las cadenas.
- NAT: Controla la traducción de direcciones y puertos, la componen las cadenas PREROUTING, POSTROUTING Y OUTPUT

- RAW: Contiene las excepciones en el seguimiento de paquetes. La acción más utilizada para esta tabla es NOTRACK. Las cadenas que se organizan en esta tabla son: PREROUTING y OUTPUT.

El entorno Netfilter permite el filtrado de paquetes, la traslación de direcciones y puertos (NAT/NAPT) y otras manipulaciones sobre el datagrama IP (packet mangling).

A modo de ejemplo las siguientes tablas indican el proceso de funcionamiento del entorno Netfilter.

Recorrido de un paquete entrante con destino local:

TABLA	CADENA	DESCRIPCIÓN
		Paquete entrante en la interfaz de red.
Mangle	Prerouting	Permite alterar algún parámetro en la cabecera
Nat	Prerouting	Permite cambiar la dirección de red destino, DNAT
		Decisión de encaminamiento: • Si el destino no es una dirección local se envía a la cadena Forward. Si no, continúa
Mangle	Input	Modificaciones del paquete antes de procesarlo
Filter	Input	Aplicado de reglas
		Envío a proceso local

Recorrido de un paquete saliente con origen local:

TABLA	CADENA	DESCRIPCIÓN
		Envío desde el proceso local
Mangle	Output	Permite alterar algún parámetro en la cabecera
Nat	Output	Permite cambiar la dirección de red destino, DNAT
Filter	Output	Aplicado de reglas de salida
Mangle	Postrouting	Modificaciones del paquete antes de procesarlo
Nat	Postrouting	Permite cambiar la dirección de red origen, SNAT
		Envío a la interfaz de red

Comparativa de Herramientas de gestión de Software firewalls

Las opciones de software GUI que permita la configuración inicial del sistema son diversas, se destacan Firewall Builder, Firestarter, Gufw Firewall, PeerGuardian Linux, FirewallD y Vuurmuur Firewall. A continuación se expresan las principales ventajas y desventajas de estas alternativas en una tabla comparativa.

Opción de Firewall	Principales Ventajas	Principales Desventajas
Firewall Builder - github.com/fwbuilder/fwbuilder	• Gran flexibilidad. • Permite administrar múltiples dispositivos. • Soporte para múltiples plataformas de firewalls. • Incorpora controles de seguridad automatizados, evitando errores en los comandos. • Documentación extensa. • Gran comunidad de usuarios • Posibilidad de mostrar la configuración de forma rápida • Posibilidad de cambios rápidos según requerimientos urgentes • Posibilidad de documentar los motivos del cambio. Ej: porque, quien solicitó, etc. • Soporte para IPv6. • Multiplataforma	• Discontinuado su mantenimiento y actualización en 2013 por los desarrolladores originales. Actualmente mantenido por la comunidad de usuarios y un equipo de colaboradores se hizo cargo del mantenimiento, actualización y desarrollo futuro por medio de la plataforma de desarrollo colaborativo Github.
PeerGuardian Linux - sourceforge.net/projects/peerguardian/	• Desarrollado y actualizado activamente. • Orientado a la privacidad y seguridad mientras el protocolo P2P puede ser utilizado.	• Bloquea conexiones basándose en lista de host controlada por los desarrolladores no en paquetes, protocolos o puertos. • Equipo de desarrollo chico. • Solo permite bloquear/permitir la conexión basándose en direcciones IP.
FirewallD www.firewalld.org	• Cambios se aplican sin reiniciar el firewall • Categorías de niveles de confianza a diferentes redes o interfaces de red • Soporta IPv4 e IPv6	• Basado en archivos de configuración XML que interactúan con iptables en vez de editar la configuración del sistema. • Toda la configuración, administración y mantenimiento se hace por CLI

Vuurmuur Firewall www.vuurmuur.org	• Administración remota segura vía SSH • Poderosas herramientas de monitoreo en tiempo real	• Utiliza una GUI personalizada basada en texto que se debe configurar y programar a través de ncurses • Discontinuado su mantenimiento y actualización en 2009.

Capítulo 4 – Protocolo de enrutamiento

En este capítulo se realizará una introducción a los protocolos interiores y exteriores a la puerta de enlace, o router de borde, su funcionamiento y relación con los sistemas autónomos.

Introducción a las interconexiones

En los primeros años de la década de 1960, Paul Baran, un investigador de la corporación RAND escribió una visión de una red militar de comunicaciones digitales que pudiera continuar funcional después de un importante daño producto de un ataque enemigo. Esta idea implicaba que la red se adaptara a fallos en la conexión, algo que la red telefónica o digital de conexiones de la época no estaba preparada para hacer debido a que cada conexión era configurada manualmente

En los comienzos de lo que hoy conocemos como Internet, cuando solo eran algunas Universidades y pocas organizaciones con el fin de compartir documentos e información relacionada a sus líneas de investigación las conexiones se establecían como canales o circuitos dedicados y ante la falla de alguna conexión o equipo intermedio toda la comunicación se veía interrumpida hasta que manualmente se configure otro camino o se revierta la falla que lo impedía. El crecimiento y auge de la conectividad hizo que no fuera una tarea que pudiera continuar realizándose de forma manual, por lo que quedaba la incógnita de cómo se iban a actualizar las tablas de enrutamiento para que refleje la forma en que está conectado todo en todo momento, la topología de “Internet”. La respuesta fueron los protocolos de enrutamiento dinámico, que permitieran la comunicación efectiva y rápida de los cambios en la red a todos los conectados.

Respuestas a las necesidades

La respuesta más simple a este inconveniente sería que los routers envíen sus rutas a sus pares de forma periódica y reciban de ellos rutas y así difundir la interconexión, este es el principio de funcionamiento del protocolo de información de enrutamiento, RIP por sus siglas en inglés, que a su vez ofrece un contador de saltos para cada ruta que le permite elegir el camino más adecuado. Luego se introdujo algoritmos que calculan el camino más “corto” a cada destino, que al caerse un enlace se continúa con el siguiente de menor costo, de esta forma se establece el protocolo del Primer Camino Corto Abierto, OSPF por sus siglas en inglés. Pero estos protocolos no pueden llevarse a gran escala donde la cantidad de redes, equipos de red y

equipos finales es tal que generaría tanto tráfico de red que sería imposible la comunicación, sumado al tiempo que llevaría a un router conocer todas las rutas disponibles, los requerimientos físicos de memoria en los mismos para el almacenamiento de dichas rutas entre otros inconvenientes. Esto llevó a generar una categorización de estos protocolos de enrutamientos como Interiores y Exteriores a la puerta de enlace, o en otras palabras, aquellos que son utilizados dentro o fuera del cada sistema autónomo, dejando los anteriormente mencionados dentro de los usados dentro de los Sistemas Autónomos.

El protocolo BGP

Dentro de la categoría de protocolos exteriores uno de los primeros fue el Protocolo de Puerta de Enlace Exterior, EGP por sus siglas en inglés, pero actualmente se utiliza el Protocolo de Borde de Puerta de Enlace, BGP por sus siglas en inglés, para compartir la información de ruteo entre sistemas autónomos. No es un protocolo de vector de distancia como es el caso de RIP ni uno de estado de la conexión como OSPF, sino que se considera un protocolo de camino-distancia debido a que no contempla las redes intermedias entre el origen y destino pero tampoco registra la topología completa de red, sino que obtiene la accesibilidad a cada sistema autónomo de los routers a los que está conectado y luego guarda en su tabla de enrutamiento el camino más corto a cada destino para luego anunciar estos caminos a su routers vecinos si la política de difusión así lo permite. De esta forma, las redes dentro de cada sistema autónomo tiene la misma ruta en BGP y puede existir más de una ruta para una misma red a través de múltiples sistemas autónomos pero solo utilizará la que el algoritmo de selección considere mejor. Si un enlace se interrumpe o un router se desconecta de la red, al momento de ser detectado este cambio de estado se re-direcciona todo el tráfico de red que antes se enviaba por ese camino hacia otras rutas, ahora mejores.

A su vez ofrece la posibilidad de configurar una política de distribución como se mencionó en el párrafo anterior, se pueden realizar listas de filtros y distribución para que no todas las rutas y enlaces tengan el mismo tratamiento, ya que es posible que en el caso de tener dos conexiones a ISP diferentes, uno no quiera que se publiquen sus rutas dentro de las redes del otro ISP. También pueden ser usadas para evitar el reenvío de caminos aprendidos por los vecinos que no sean generados por el router, entre otras funciones.

BGP utiliza el protocolo TCP como protocolo de transporte estableciendo una sesión entre dos routers vecinos o pares en el puerto 179 donde intercambian información de rutas. Por medio de esta sesión, que

permite que dos pares no estén directamente conectados, se publican rutas para llegar a un determinado destino y se incorporan en mi tabla de ruteo aquellas que son publicadas por los routers pares y/o vecinos si es que tienen un “mejor” costo. Junto con la información de encaminamiento se incluyen ciertos atributos que permiten a los administradores de red influir en las decisiones BGP,

Necesidad de implementación

Debido a que LACNIC asignó al Instituto un pool de direcciones IPv4 e IPv6 en conjunto con un ASN, la red privada que se conecta a un ISP el cual era responsable de garantizar su acceso y disponibilidad pasó a ser un sistema autónomo que debe cumplimentar una serie de requisitos y compromisos asumidos en ese acto. Uno de los apartados que se aceptó fue que el Instituto es responsable de garantizar la ruteabilidad global de las direcciones IP públicas propias y de la propagación de las mismas para que el SA esté accesible a través de Internet. Esto se interpreta como la implementación de un protocolo EGP en el router que realiza la conexión a los ISP que proveen de conectividad al instituto.

En la actualidad el protocolo recomendado por el Organismo regulador es BGP en su versión 4.

Recomendaciones de LACNIC

Toda configuración es particular para cada sistema en el que se está trabajando, pero siempre existen recomendaciones que por costumbres, buenas prácticas o inconvenientes detectados con anterioridad suelen ser de aplicación general. El organismo que regula y distribuye las direcciones IP ofrece las siguientes con el objetivo de mejorar la conectividad en la zona de aplicación.

En el caso de que no se puedan aprender o recibir la tabla completa de BGP (para el Marzo de 2017 constaba de ~610.000 prefijos IPv4) no filtrar para aprender redes “grandes” y permitir aprender redes más específicas.

La utilización de certificación de recursos (RPKI) para validar el derecho de uso de los recursos asignados al instituto para mejorar la seguridad en el enrutamiento de paquetes IP, la construcción de filtros para el anuncio rutas, construcción de reglas de rutas basadas en la validez de los prefijos, autenticación de routers en la red para el protocolo de descubrimiento de vecinos y la firma de información de los servicios de Whois.

Se recomienda nunca redistribuir prefijos BGP en un IGP o viceversa, no usar IGP para transportar los prefijos de los clientes u otras redes

externas, no recibir los prefijos definidos en el RFC 1918, ni aceptar mis propios prefijos o mayores de /24.

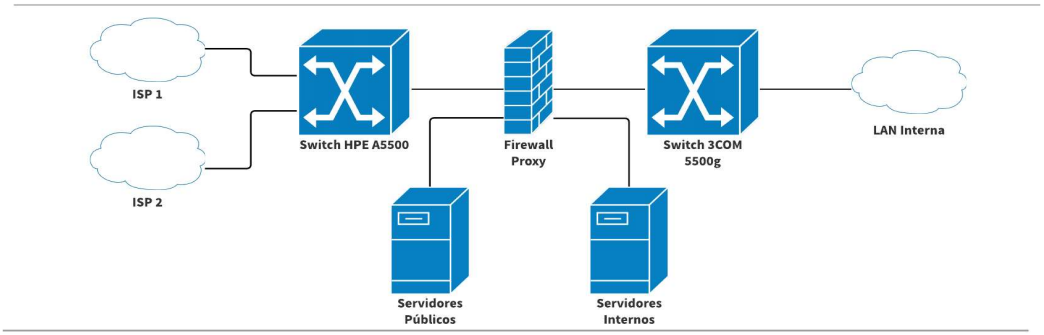
Capítulo 5 – Implementación

En este capítulo se explicarán los recursos disponibles y metodología utilizada en este proyecto.

Introducción - Brief de la propuesta

Se propone la implementación de un Sistema Autónomo que permitirá tener el control de las reglas de ruteo de la red mediante el protocolo BGP, zonas desmilitarizadas y el rediseño de la red interna del Instituto incluyendo una configuración inicial de la herramienta de gestión del Firewall y junto con las distintas VLans a crear, con su usuario objetivo asociado y recomendaciones sobre la configuración de los switches capa 2/3.

La configuración propuesta del firewall es con cuatro interfaces de red: red externa, red interna, DMZ-Interna y DMZ-Externa.



Esta propuesta de configuración es poco usual, debido a las dos zonas desmilitarizadas y responde a la búsqueda de optimización de los recursos de red disponibles teniendo en cuenta la seguridad de la red. Se crean la DMZ-Interna y la DMZ-Externa, ya que hay servicios comunes a toda la red interna que no deben ser accedidos desde el exterior de la red, siendo estos colocados dentro de los servidores en la DMZ-Interna se agrega un nivel de seguridad extra. Todos aquellos servicios que son accedidos desde el exterior de la red y desde el interior se colocaran dentro de los servidores alojados en la DMZ-Externa. Esta configuración si bien significa una nueva interfaz para controlar por el firewall no suma tiempo de procesamiento ya que los paquetes se procesarán igualmente ya sea para una o dos DMZ pero tiene la intención de impactar en la utilización de la red, balanceando la carga en dos enlaces en vez de uno, impactando en los servicios consultados al mejorar tiempo de respuesta por tener colas de procesamiento más

pequeñas.

En la DMZ-Externa se ubicaran todos los servicios que deben ser accedidos desde afuera del instituto como lo es su portal web, a estos servicios se le asignan direcciones IP públicas.

En la DMZ-Interna se encuentran los servicios que deben ser accedidos únicamente desde la red interna como aquellos servidores de simulación o uno de base de datos.

La red interna se propone hacer una división en subredes basadas en las necesidades de los usuarios y a su vez una categorización en niveles de acceso de cada subred, Nivel Básico, Medio y Libre. La propuesta contempla la creación de 29 subredes y se detallan en el apéndice Subneteo.

Equipos alcanzados por la propuesta

Como es un rediseño de la red y el objetivo especifica la no modificación de la infraestructura más si la adecuación de la misma es que no se deberá incurrir en nuevos dispositivos de red y todas las nuevas configuraciones serán realizadas con el equipamiento disponible en el instituto. La propuesta abarca la adecuación de la configuración de los siguientes dispositivos:

- Bastión Intel XEON corriendo OpenSUSE Leap
- Router HP A5500-24G EI
- Switch 3Com 5500G-EI 24-Port
- Máquina del Administrador de red.

Experimentación

Se propone la realización de 29 subredes, cada una correspondiente a una VLAN diferente, que responde a las diferentes áreas, departamentos de trabajo o grupo de usuarios del Instituto. Luego se categorizan estas VLANs en Acceso Básico, Acceso Medio y Acceso Libre basados en los servicios que requieren acceder dentro y fuera del instituto. Completada esta segmentación se identifican excepciones dentro de cada subred que deban tenerse en cuenta para la creación de reglas especiales de filtrado y por último se determina que recursos externos e internos estarán disponibles para cada uno de los tres niveles de acceso definidos.

Por qué FWBuilder

FWBuilder es una alternativa que se ajusta a las necesidades y requerimientos de la red, los motivos que llevaron a su elección son su interfaz gráfica de configuración y administración orientada a objetos, el

hecho que crea archivos de configuración para el paquete netfilter del kernel en vez de interactuar con él como otras alternativas, su curva de aprendizaje reducida por el conocimiento previo de la herramienta por los administradores de red del instituto y la posibilidad de gestión multiplataforma que ofrece, permitiendo si se decidiera cambiar el firewall por una cierta alternativa, solamente cambiar el hardware y enviarle los archivos de configuración sin tener que reescribir las reglas.

Sistema operativo

Dentro del abanico de sistemas operativos de código abierto disponibles hoy en día, la elección para el sistema operativo anfitrión donde se instalará el firewall se recomienda OpenSuse Leap 42.2. Esta versión cuenta con un kernel Linux en su versión 4.4 el cual incluye el paquete netfilter, y no genera un consumo de recursos innecesarios que pudieran afectar el funcionamiento general del bastión impactando en la disponibilidad de la red. A su vez, este kernel tiene un soporte extendido en el tiempo a comparación con las otras versiones disponibles.

La misma recomendación de sistema operativo anfitrión se realiza para los equipos que utilicen administradores de red.

Hardware

Una de las ventajas de la herramienta FWBuilder en sí, es que crea y edita los archivos con formato de tablas que configuran el módulo IPTables del framework Netfilter en un dispositivo para luego ser ejecutados en otro, previo su instalación en el equipo destino. Es por ello que en la descripción del hardware utilizado para este trabajo diferenciamos los equipos donde corren los sistemas de los que se utilizaron para la configuración de los mismos.

Para la creación de los archivos, se utilizó una notebook con las siguientes características técnicas:

- Procesador: Intel(R) Core(TM) i5-2450M CPU @ 2.50GHz, 2501 Mhz, 2 Núcleos, 4 Procesadores lógicos
- Memoria RAM: 8GB DDR3
- Disco Rigido: SSD SanDisk 120GB
- Sistema Operativo: Windows 10 Pro 64Bits

Para la gestión posterior de estos archivos se utilizarán las máquinas de los administradores de red, y los mismos serán ejecutados en el firewall propiamente dicho, el bastión.

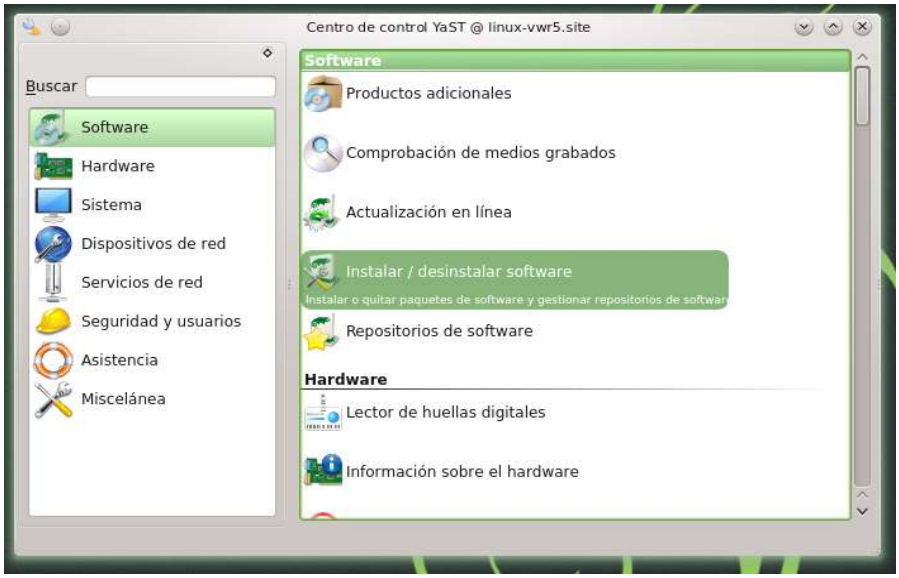
Instalación de FWBuilder en OpenSUSE

Es necesario aclarar que en el Bastión no se instalará esta herramienta, ya que no es necesaria para la ejecución del paquete Netfilter, al estar provisto por el S.O. y la instalación que se presenta en este apartado es sobre la máquina del Instituto asignada a uno de los administradores de la red y encargado de la administración del firewall. Ya que la herramienta

FWBuilder es para el manejo, configuración y administración de las reglas del firewall.

Para la instalación del software y sus dependencias utilizaremos el centro de control de OpenSUSE llamado YAST por sus siglas en Inglés, la cual nos permite desde un entorno gráfico acceder a todas las configuraciones del sistema, entre ellas la de instalar y desinstalar software.

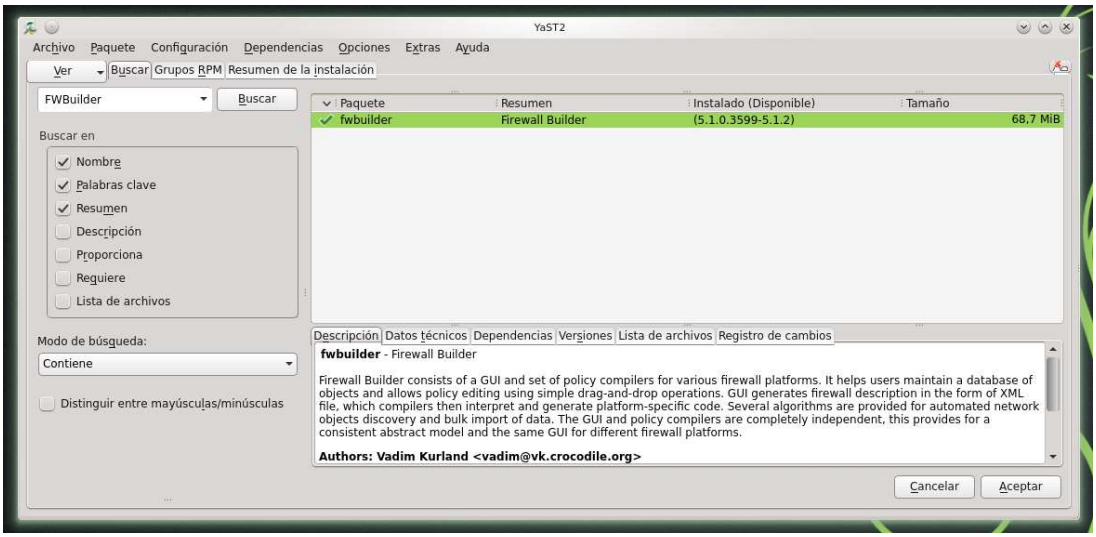
Para ello iniciamos la herramienta en su entorno gráfico, en el panel izquierdo buscamos la sección Software y dentro de esta sección utilizaremos la función “Instalar / Desinstalar Software” como se muestra en la imagen a continuación:



Esta herramienta nos permite buscar entre los repositorios de software el archivo binario, sus archivos de dependencias e instalarlo con solo seguir los siguientes pasos, que se listan a continuación.

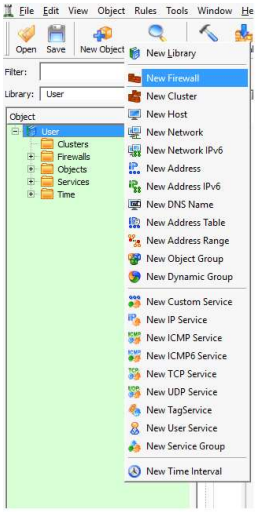
- Iniciamos el gestor de Software que luego de hacer una comprobación en los repositorios en busca de actualizaciones para el sistema, nos muestra la sección de búsqueda.
- En el cuadro de búsqueda escribimos el nombre del software a instalar, en este caso: FWBuilder y hacemos click en buscar.
- Seleccionamos el paquete a instalar por medio de un click en el lateral izquierdo, lo que se nos confirma con la tilde verde que se sitúa en ese espacio indicando que hemos seleccionado el paquete para su instalación.
- Hacemos click en la pestaña “Resumen de la Instalación” para corroborar que se instale el paquete seleccionado y las librerías relacionadas de ser necesarias sus descargas.

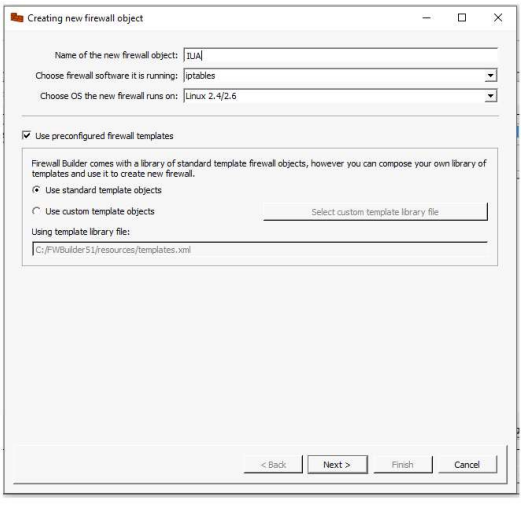
- Damos click en el botón Aceptar para confirmar y proceder a la instalación.



Configuración inicial del FWBuilder

Una vez instalado el software de gestión FWBuilder procedemos con la configuración inicial del mismo. En esta etapa del proceso se define el sistema operativo donde se ejecutará y el tipo de software firewall para el cual se crearán los archivos de configuración. Para ello en la ventana principal de la herramienta se hace clic en objeto nuevo y luego en nuevo firewall se abrirá una nueva ventana con el asistente de configuración donde se ingresa los parámetros del SO, versión del software de firewall y el nombre con el que se hará referencia a este firewall.





Se continúa con el asistente de configuración donde configuramos

manualmente las cinco interfaces y completamos la configuración básica.

Ya contamos con la herramienta que nos permitirá generar los archivos de configuración del firewall, la siguiente etapa es la de creación de los objetos necesarios para la creación de las reglas de filtrado de paquetes descritos en la primer parte de esta sección y detallados en los apéndices Objetos FWBuilder.

Creación de reglas

En esta sección se describirán los pasos para la creación de reglas de filtrado de paquetes con FWBuilder. Para ello se propone un ordenamiento de las mismas debido al funcionamiento lógico del módulo IPTables donde el orden de las reglas impacta en el funcionamiento en forma directa. Se propone primero incluir las reglas que prevén ataques externos, intromisiones y demás medidas de seguridad que protejan la red del Instituto respecto a accesos no deseados desde el exterior. Luego incluir las reglas que permiten la normal ejecución de los servicios brindados por los servidores en la DMZ Externa, seguido por las excepciones identificadas de equipos específicos para concluir con las reglas generales de los grupos de vlans. En otras palabras, el ordenamiento comenzaría con los servicios públicos hacia los privados y desde lo específico hacia lo genérico.

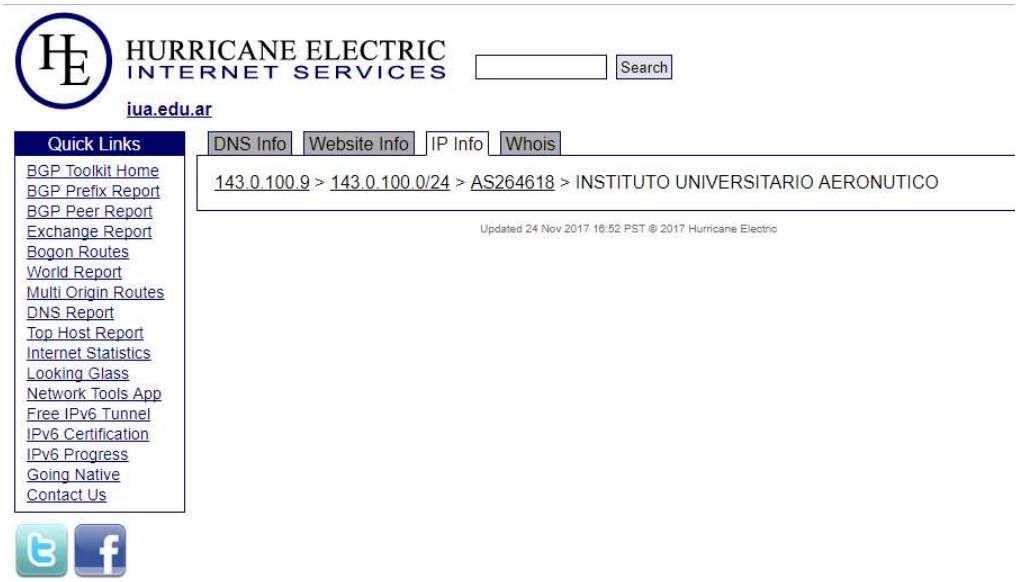
La política por defecto es la de denegación, por lo que si no se crea una regla que coincida con el tipo de conexión o paquete analizado la misma será rechazada. Basado en esto es que se realiza una propuesta de reglas básicas y toda conexión necesaria que no fuese contemplada a medida que sea requerida permitirá por medio de la función de añadir comentarios y observaciones que tiene FWBuilder registrar quien solicita el acceso, los motivos, quien aprobó el acceso y un número de seguimiento si se contara con tal. También se permite de esta forma, si el requerimiento de acceso es por un tiempo determinado la creación de un objeto con dicho intervalo de tiempo para que de forma automática se le conceda o rechace la conexión.

Los archivos generados se incluyen en el apéndice pero se describirán algunas reglas a modo de explicación del procedimiento propuesto para la generación de las mismas. Las primeras reglas que describe es la regla número 2 y 3 que permiten en conjunto el acceso remoto al firewall. Estas reglas, suponen un riesgo de seguridad por permitir una conexión desde el exterior de la red al bastión obteniendo control a nivel de usuario y la primer medida aplicada para la mitigación del riesgo es la división de las conexiones en dos reglas obligando a realizar un salto a una máquina intermedia para completar el acceso al bastión. El primer salto es desde el exterior a la máquina de administración del bastión donde se toma como medida de seguridad que solamente se permite la conexión desde una dirección MAC

conocida y a través del protocolo SSH únicamente dentro del horario en el que los administradores de la red no se encuentran realizando sus tareas dentro del Instituto, restricción generada a partir de la regla número tres. Una vez que se accede a una de las dos máquinas de administración del bastión el segundo salto también se debe realizar mediante el protocolo SSH y de esta forma se obtiene acceso al bastión donde se ejecuta el firewall.

En resumen, lo que las reglas 2 y 3 determinan es el acceso al bastión donde se ejecuta el software de firewall y se puede realizar únicamente desde dos equipos que están dentro de la red del instituto, físicamente ubicados dentro de uno de sus edificios, y en el imprevisto de un problema que ocurra fuera del horario laboral de los administradores de red motivo por el cual deban acceder a los servicios administrados, se permite la conexión desde el exterior a sus equipos desde una dirección MAC conocida, toda otra conexión se rechaza y registra en un log para su análisis.

Las otras reglas que se analizan son aquellas que restringen el acceso a servicios externos de HTTP y HTTPS. Para ello se propone la utilización de tablas de direcciones IP, versiones 4 y 6, que contengan todas las direcciones asociadas a un sistema autónomo debido a que no se pueden conocer todos los nombres que resuelven a una dirección IP pero si todos los bloques de direcciones IP asociados a un NSA y su propietario. De esta forma se bloquea el acceso a grandes proveedores de servicios, que resultan ser los de mayor consumo de ancho de banda, que no estén asociados a las tareas de cada área. Para ello se utiliza una herramienta de BGP como la que se encuentra en internet en bgp.he.net, realizar una búsqueda por ip o dirección web del dominio que se desea información y obtendremos información general como los nombre de los servidores de correo o los servidores de resolución de nombres y lo relativo al SA y los bloques de direcciones IP en el apartado “IP Info” como muestra la siguiente imagen



De esta forma podemos obtener el número de sistema autónomo, y mediante éste las direcciones IP asociadas al mismo, las cuales ingresamos en un archivo de texto plano con formato de tabla, una dirección debajo de la otra, y podremos así bloquear las conexiones a estas direcciones evitando los re direccionamientos que suelen realizar estos sitios para permitir que sus usuarios accedan a sus servicios aun cuando la política de una red no lo permita. Este esquema se implementa en las reglas 11 y 12 de la propuesta.

Configuraciones recomendadas para el Switch 3COM 5500g-EI 24-Port

Políticas de Calidad de Servicio - QoS

La primera configuración recomendada es la de Generic Traffic Shaping que permite limitar la utilización del ancho de banda de un puerto del switch. Esto permite que un puerto o grupo de puertos no ocupen todo o gran parte del ancho de banda de la red. Esta política deberá ser diferente por cada VLAN debido a los requerimientos de uso de cada área o departamento, la propuesta que incluye este trabajo final es la de aplicar tres límites máximos, 50% - 80% - 100% en concordancia con los niveles de acceso de cada VLAN.

La configuración de este parámetro se puede realizar por medio de la herramienta de administración web o a través de la consola de administración en la que se deberán seguir tres pasos:

Paso	*Comando*	*Observaciones*
Entrar en la vista de sistema	**system-view**	
Entrar a la vista de interfaz o de grupo de puertos	• Vista de interfaz: **interface** • Vista de grupo de puertos: **port-group-manual** nombre-grupo-de-puertos	Se puede utilizar cualquier comando, en la vista de interfaz configura ese único puerto y en la vista de grupo de puertos configura todos los puertos de ese grupo
Configurar GTS para un cola	**qos gts queue** número-de-cola **cir** committed-information-rate [**cbs** committed-burst-size]	

Otra política de calidad de servicio es la centralización de la implementación de listas de control de acceso, actualmente las ACL están en uso en la red pero distribuidas en cada switch de la red, lo que hace muchas veces que un cambio sea complejo de realizar debido a la cantidad listas a modificar o controlar. Centralizando las listas de acceso en un switch se facilita la administración y gestión de las mismas. Es por esto que se propone la configuración de las mismas en este nivel de la red utilizando “Ethernet frame
header ACLs” para las conexiones permitidas en los servidores principales y críticos, como los de base de datos, donde el criterio sea del tipo de encabezados de capa 2, como dirección MAC origen o destino, y para aquellos servicios menos restringidos utilizar las listas correspondientes a “Advanced ACLs” que permiten criterios de capa 2 y 3, tales como dirección IP origen o destino.

Se recomienda la implementación de un mapeo de prioridades del tipo local, las cuales son asignadas por el dispositivo únicamente con el propósito de asignación de tiempos de espera y procesado. Esta prioridad está compuesta por dos parámetros “Local Precedence” y “Drop Precedence”. El primer valor determina la cola de salida del switch, mientras mayor sea este valor mayor preferencia tendrá el paquete y se procesa antes que uno con menor valor en este campo. La prioridad “Drop Precedence” determina qué paquetes serán descartados si las colas se llenan, mientras mayor sea esta prioridad menos importantes son los paquetes y se desestimará primero. Se propone que los paquetes provenientes de las VLAN 23, 24, 19, 20, 11, 12, 17, 14, tienen prioridad Local alta y Drop bajo, en el orden propuesto.

Configuración del protocolo BGP, difusión de rutas y número de sistema autónomo

Para el switch de capa 3 HP A5500 EI proponemos la configuración del protocolo BGP4 como EGP y del protocolo RIP como IGP. El primero por ser el de mayor utilización en los enrutadores de Internet en la actualidad y estar en concordancia con las recomendaciones de LACNIC. El protocolo RIP es la alternativa más adecuada dentro de los protocolos IGP por el tamaño de la red y por la simplicidad de configuración y puesta en marcha, donde solo necesita ser habilitado en los routers y switches de red como así también su tráfico.

Para configurar el protocolo BGP se procede con la siguiente configuración sencilla de permitir establecer conexiones EBGP a router pares o grupo de ellos siguiendo los pasos requeridos para tal fin. Todas las configuraciones opcionales quedarán a criterio de los administradores de la red y los acuerdos que lleguen los ISP con el Instituto. Para poder proceder a esta configuración se requiere tener la siguiente información provista por los IPS de antemano: número de sistema autónomo y dirección IP pública de cada enlace. Los comandos a ejecutar en el Switch son:

[HP5500] **system-view**
[HP5500-bgp] **bgp** 264618
[HP5500-bgp] **router-id** {dirección ip de loopback}
[HP5500-bgp] **peer** { dirección ip del IPS1} **as-number** { asn }
[HP5500-bgp] **peer** { dirección ip del IPS2} **as-number** { asn }
[HP5500-bgp] **quit**

Como se mencionó anteriormente, el protocolo IBGP RIP solo requiere habilitarlo en cada switch o router de la red para completar la configuración básica del mismo, pero en el caso del switch 3Com 5500g se incluyen las direcciones de cada una de las subredes conectadas al mismo para mejorar la perfomance del mismo, se describen los comandos a continuación a modo de ejemplo sin incluir las direcciones de cada subred:

[3com5500] **rip**
[3com5500-rip-1] **network** {dirección IP subred 1}
[3com5500-rip-1] **network** {dirección IP subred 2}
[3com5500-rip-1] **network** {dirección IP subred 3}

Capítulo 6 - Conclusiones

La asignación de un Sistema Autónomo, los requisitos necesarios para ello junto con el relevamiento de los requerimientos actuales y futuras del Instituto expuso la necesidad de adecuar los sistemas y herramientas actuales de la división de redes y servidores del departamento de tecnología de la información del Instituto.

Se realizó un relevamiento de cada una de las áreas afectadas por este TFG, junto con este relevamiento se condujo una investigación teórica sobre los protocolos de ruteo, herramientas de administración de firewalls, el firewall propiamente dicho por nombrar los principales, que llevó en algunos casos a contactar directamente al equipo de desarrolladores de la herramienta de gestión del firewall que permitió confirmar cuestiones de continuidad del servicio y soporte técnico, un aspecto importantísimo para una herramienta de este tipo. Otro contacto se realizó con el área de soporte de la herramienta de proveida por HURRICANE ELECTRIC para la búsqueda de información del protocolo BGP, que obtiene datos del ruteo global incluyendo información de los sistema autónomo, direcciones IPv4 e IPv6 asociadas, pares del SA y otra información necesaria para la creación de reglas como la herramienta looking glass que permite obtener un traceroute y una ruta BGP para la identificación de host destinos a bloquear y sus SA, confirman que la búsqueda de información es gratuita hasta un máximo de 200 consultas diarias y para el caso de necesidad de grandes búsquedas se puede solicitar un acceso que permite hasta 20.000 consultas por día a un costo.

Estas consultas, con empresas y desarrolladores en Estados Unidos y Europa, en conjunto con la investigación teórica e histórica de las tecnologías resultó en una propuesta de configuración y readecuación de los servicios que satisfacerá las necesidades actuales, mejorará la prestación de servicios internos y externos que llevarán a una optimización de los recursos actuales y proveerá herramientas más completas para la administración de la red del instituto que ofrecerán soporte futuro y prolongarán la vida útil de la infraestructura del Instituto.

Un aspecto importante es que se permite la inclusión del protocolo IPv6, que hoy en día no está implementado dentro del Instituto pero se encuentra en trámite. Esta implementación será transparente para la propuesta de configuración aquí descrita que incorporará esta versión dle protocolo IP sin inconvenientes, al permitir al administrador del firewall crear los objetos dentro de la herramienta de administración con las direcciones configuradas en los equipos y bloquear, aceptar o redireccionar las conexiones. También se hace referencia que algunas de las reglas de bloqueo de conexiones incluidas en la propuesta rechazan conexiones a direcciones IPv6 de algunos sitios, confirmando la utilización de la configuración propuesta para ambas versiones del protocolo.

La propuesta de configuración de la máquina bastión, basada en el framework de NetFilter, permitirá un nivel extra de seguridad al estar integrada con el kernel del sistema operativo que ejecutará el filtrado de paquetes permitirá el control de los mismos, sumado a la configuración de

doble zona desmilitarizada agregará seguridad a la red a la vez que permitirá una optimización en los servidores.

De esta forma, en concordancia a los objetivos planteados inicialmente y los que resultaron del relevamiento, se presenta una propuesta que los abarca, sin inversión económica y con software libre y de código abierto.

Referencias web y bibliografía consultada

www.lacnic.net

https://labs.lacnic.net/BGP-Filtrar-tamano/

http://www.lacnic.net/web/lacnic/informacion-general-rpki

http://www.lacnic.net/innovaportal/file/2621/1/bgp-panama-lacnic29.pdf

ftp.hp.com/pub/networking/software/

ftp.hp.com/pub/networking/training/

http://www.lacnic.net/web/lacnic/servicios-asn

http://www.redescisco.net/sitio/2010/06/22/la-encapsulacion-de-datos-un-concepto-critico/

http://standards.ieee.org/about/get/802/802.html

http://www.cisco.com/c/en/us/td/docs/switches/lan/catalyst4500/12-2/25ew/configuration/guide/conf/vlans.html

https://www.bluecoat.com/sites/default/files/documents/files/VLAN_Tagging.1.pdf

http://www.cisco.com/c/en/us/products/security/firewalls/what-is-a-firewall.html

https://www.netfilter.org/projects/iptables/index.html

http://www.cisco.com/c/en/us/td/docs/solutions/Enterprise/Security/IE_DG.html

https://www.dte.us.es/docencia/etsii/gii-ti/tecnologias-avanzadas-de-la-informacion/Laboratorio-2-Netfilter.pdf

https://www.ibiblio.org/pub/linux/docs/LuCaS/Manuales-LuCAS/SEGUNIX/unixsec-2.1-html/node235.html

Comunicaciones y Redes de Computadores – William Stallings – 7° Edición

http://www.lacnic.net/web/lacnic/ipv4-isp

http://www.lacnic.net/web/lacnic/servicios-asn

http://www.redescisco.net/sitio/2010/06/22/la-encapsulacion-de-datos-un-concepto-critico/

http://standards.ieee.org/about/get/802/802.html

http://www.cisco.com/c/en/us/td/docs/switches/lan/catalyst4500/12-2/25ew/configuration/guide/conf/vlans.html

https://www.bluecoat.com/sites/default/files/documents/files/VLAN_Tagging.1.pdf

http://www.cisco.com/c/en/us/products/security/firewalls/what-is-a-firewall.html

https://www.netfilter.org/projects/iptables/index.html

http://www.cisco.com/c/en/us/td/docs/solutions/Enterprise/Security/IE_DG.html

http://www.cisco.com/c/dam/en/us/td/i/000001-100000/15001-20000/16501-17000/16751.ps/_jcr_content/renditions/16751.jpg

BGP - Building Reliable Networks with the Border Gateway Protocol - Iljitsch van Beijnum

Apéndice subredes

vlan ID	# Red	Sured	Desde	Hasta	Cant.	Broad cast	Máscara de red	Dpto
1	1	192.168.0.0	192.168.0.1	192.168.0.254	256	192.168.0.255	255.255.255.0	
2	2	192.168.1.0	192.168.1.1	192.168.1.254	256	192.168.1.255	255.255.255.0	AIT
3	3	192.168.2.0	192.168.2.1	192.168.2.254	256	192.168.2.255	255.255.255.0	Biblioteca
4	4	192.168.3.0	192.168.3.1	192.168.3.254	256	192.168.3.255	255.255.255.0	CASA 44
5	5	192.168.4.0	192.168.4.1	192.168.4.254	256	192.168.4.255	255.255.255.0	CIA
6	6	192.168.5.0	192.168.5.1	192.168.5.254	256	192.168.5.255	255.255.255.0	DTI
7	7	192.168.6.0	192.168.6.1	192.168.6.254	256	192.168.6.255	255.255.255.0	Dto. Econ./Cont.
8	8	192.168.7.0	192.168.7.1	192.168.7.254	256	192.168.7.255	255.255.255.0	Egresados
9	9	192.168.8.0	192.168.8.1	192.168.8.254	256	192.168.8.255	255.255.255.0	ESFA
10	10	192.168.9.0	192.168.9.1	192.168.9.254	256	192.168.9.255	255.255.255.0	Externa
11	11	192.168.10.0	192.168.10.1	192.168.10.254	256	192.168.10.255	255.255.255.0	FCA
12	12	192.168.11.0	192.168.11.1	192.168.11.254	256	192.168.11.255	255.255.255.0	FI
13	13	192.168.12.0	192.168.12.1	192.168.12.254	256	192.168.12.255	255.255.255.0	Gestion
14	14	192.168.13.0	192.168.13.1	192.168.13.254	256	192.168.13.255	255.255.255.0	Guardia
15	15	192.168.14.0	192.168.14.1	192.168.14.254	256	192.168.14.255	255.255.255.0	I+D
16	16	192.168.15.0	192.168.15.1	192.168.15.254	256	192.168.15.255	255.255.255.0	Ingeniería
17	17	192.168.16.0	192.168.16.1	192.168.16.254	256	192.168.16.255	255.255.255.0	Laboratorios
18	18	192.168.17.0	192.168.17.1	192.168.17.254	256	192.168.17.255	255.255.255.0	Militar
19	19	192.168.18.0	192.168.18.1	192.168.18.254	256	192.168.18.255	255.255.255.0	Proxy
20	20	192.168.19.0	192.168.19.1	192.168.19.254	256	192.168.19.255	255.255.255.0	Pública

21	21	192.168.20.0	192.168.20.1	192.168.20.254	256	192.168.20.255	255.255.255.0	Rectorado
22	22	192.168.21.0	192.168.21.1	192.168.21.254	256	192.168.21.255	255.255.255.0	Seguridad
23	23	192.168.22.0	192.168.22.1	192.168.22.254	256	192.168.22.255	255.255.255.0	videoconferencias
24	24	192.168.23.0	192.168.23.1	192.168.23.254	256	192.168.23.255	255.255.255.0	Servidores Internos
25	25	192.168.24.0	192.168.24.1	192.168.27.254	1022	192.168.27.255	255.255.252.0	wifi
26	26	192.168.28.0	192.168.28.1	192.168.31.254	1022	192.168.31.255	255.255.252.0	wifi
27	27	192.168.32.0	192.168.32.1	192.168.35.254	1022	192.168.35.255	255.255.252.0	wifi
28	28	192.168.36.0	192.168.36.1	192.168.39.254	1022	192.168.39.255	255.255.252.0	wifi
29	29	192.168.40.0	192.168.40.1	192.168.43.254	1022	192.168.43.255	255.255.252.0	wifi
30	30	192.168.44.0	192.168.44.1	192.168.44.254	256	192.168.44.255	255.255.255.0	Servidores externos
31	31	192.168.45.0	192.168.45.1	192.168.45.254	256	192.168.45.255	255.255.255.0	reservada para uso futuro
...	...	...	...	...	...	...	...	...
240	240	192.168.254.0	192.168.254.1	192.168.254.254	256	192.168.254.255	255.255.255.0	reservada para uso futuro
241	241	192.168.255.0	192.168.255.1	192.168.255.254	256	192.168.255.255	255.255.255.0	reservada para uso futuro

Apéndice Propuesta de configuración Firewall

Nota del autor: Por su extensión, las reglas 11 y 12 fueron editadas y solo se incluyen en este apéndice el comienzo y final de cada una.

```
#!/bin/sh
#
#  This is automatically generated file. DO NOT MODIFY !
#
#  Firewall Builder  fwb_ipt v5.1.0.3599
#
#  Generated Mon Sep 03 19:18:53 2018 Argentina Standard Time by Mato
#
```

```
# files: * IUA.fw /etc/IUA.fw
#
# Compiled for iptables (any version)
#
# This firewall has three interfaces. Eth0 faces outside and has a static routable address; eth1 faces inside; eth2 is connected to DMZ subnet.
# Policy includes basic rules to permit unrestricted outbound access and anti-spoofing rules. Access to the firewall is permitted only from internal network and only using SSH. The firewall uses one of the machines on internal network for DNS. Internal network is configured with address 192.168.1.0/255.255.255.0, DMZ is 192.168.2.0/255.255.255.0. Since DMZ used private IP address, it needs NAT. There is a mail relay host located on DMZ (object 'server on dmz'). Policy rules permit SMTP connections to it from the Internet and allow this server to connect to a host on internal network 'internal server'. All other access from DMZ to internal net is denied. To provide access to the mail relay its private address is mapped to firewall's outside interface address by NAT rule #1.


FWBDEBUG=""

PATH="/sbin:/usr/sbin:/bin:/usr/bin:${PATH}"
export PATH


LSMOD="lsmod"
MODPROBE="modprobe"
IPTABLES="iptables"
IP6TABLES="ip6tables"
IPTABLES_RESTORE="iptables-restore"
IP6TABLES_RESTORE="ip6tables-restore"
IP="ip"
IFCONFIG="ifconfig"
VCONFIG="vconfig"
BRCTL="brctl"
IFENSLAVE="ifenslave"
IPSET="ipset"
LOGGER="logger"

log() {
    echo "$1"
    which "$LOGGER" >/dev/null 2>&1 && $LOGGER -p info "$1"
}

getInterfaceVarName() {
    echo $1 | sed 's/\./_/'
}

getaddr_internal() {
    dev=$1
    name=$2
    af=$3
    L=$($IP $af addr show dev $dev |  sed -n '/inet/{s!.*inet6* !!;s!/.*!!p}' | sed 's/peer.*//')
    test -z "$L" && {
        eval "$name=""
```

```
        return
    }
    eval "${name}_list=\"$L\""
}

getnet_internal() {
    dev=$1
    name=$2
    af=$3
    L=$($IP route list proto kernel | grep $dev | grep -v default |  sed 's! .*$!!')
    test -z "$L" && {
        eval "$name=""
        return
    }
    eval "${name}_list=\"$L\""
}


getaddr() {
    getaddr_internal $1 $2 "-4"
}

getaddr6() {
    getaddr_internal $1 $2 "-6"
}

getnet() {
    getnet_internal $1 $2 "-4"
}

getnet6() {
    getnet_internal $1 $2 "-6"
}

# function getinterfaces is used to process wildcard interfaces
getinterfaces() {
    NAME=$1
    $IP link show | grep ": $NAME" | while read L; do
        OIFS=$IFS
        IFS=" :"
        set $L
        IFS=$OIFS
        echo $2
    done
}

diff_intf() {
    func=$1
    list1=$2
    list2=$3
    cmd=$4
    for intf in $list1
    do
```

```
        echo $list2 | grep -q $intf || {
        # $vlan is absent in list 2
            $func $intf $cmd
        }
    done
}

find_program() {
  PGM=$1
  which $PGM >/dev/null 2>&1 || {
    echo "\"$PGM\" not found"
    exit 1
  }
}
check_tools() {
  find_program which
  find_program $IPTABLES
  find_program $MODPROBE
  find_program $IP
}
reset_iptables_v4() {
  $IPTABLES -P OUTPUT  DROP
  $IPTABLES -P INPUT   DROP
  $IPTABLES -P FORWARD DROP

cat /proc/net/ip_tables_names | while read table; do
  $IPTABLES -t $table -L -n | while read c chain rest; do
      if test "X$c" = "XChain" ; then
        $IPTABLES -t $table -F $chain
      fi
  done
  $IPTABLES -t $table -X
done
}

reset_iptables_v6() {
  $IP6TABLES -P OUTPUT  DROP
  $IP6TABLES -P INPUT   DROP
  $IP6TABLES -P FORWARD DROP

cat /proc/net/ip6_tables_names | while read table; do
  $IP6TABLES -t $table -L -n | while read c chain rest; do
      if test "X$c" = "XChain" ; then
        $IP6TABLES -t $table -F $chain
      fi
  done
  $IP6TABLES -t $table -X
done
}


P2P_INTERFACE_WARNING=""
```

```
missing_address() {
    address=$1
    cmd=$2

    oldIFS=$IFS
    IFS="@"
    set $address
    addr=$1
    interface=$2
    IFS=$oldIFS


    $IP addr show dev $interface | grep -q POINTOPOINT && {
        test -z "$P2P_INTERFACE_WARNING" && echo "Warning: Can not update address of
interface $interface. fwbuilder can not manage addresses of point-to-point interfaces yet"
        P2P_INTERFACE_WARNING="yes"
        return
    }

    test "$cmd" = "add" && {
      echo "# Adding ip address: $interface $addr"
      echo $addr | grep -q ':' && {
          $FWBDEBUG $IP addr $cmd $addr dev $interface
      } || {
          $FWBDEBUG $IP addr $cmd $addr broadcast + dev $interface
      }
    }

    test "$cmd" = "del" && {
      echo "# Removing ip address: $interface $addr"
      $FWBDEBUG $IP addr $cmd $addr dev $interface || exit 1
    }

    $FWBDEBUG $IP link set $interface up
}

list_addresses_by_scope() {
    interface=$1
    scope=$2
    ignore_list=$3
    $IP addr ls dev $interface | \
      awk -v IGNORED="$ignore_list" -v SCOPE="$scope" \
        'BEGIN {
           split(IGNORED,ignored_arr);
           for (a in ignored_arr) {ignored_dict[ignored_arr[a]]=1;}
         }
         (/inet |inet6 / && $0 ~ SCOPE && !($2 in ignored_dict)) {print $2;}' | \
        while read addr; do
          echo "${addr}@$interface"
            done | sort
}
```

```
update_addresses_of_interface() {
    ignore_list=$2
    set $1
    interface=$1
    shift

    FWB_ADDRS=$(
      for addr in $*; do
        echo "${addr}@$interface"
      done | sort
    )

    CURRENT_ADDRS_ALL_SCOPES=""
    CURRENT_ADDRS_GLOBAL_SCOPE=""

    $IP link show dev $interface >/dev/null 2>&1 && {
      CURRENT_ADDRS_ALL_SCOPES=$(list_addresses_by_scope $interface 'scope .*'
"$ignore_list")
      CURRENT_ADDRS_GLOBAL_SCOPE=$(list_addresses_by_scope $interface 'scope
global' "$ignore_list")
    } || {
      echo "# Interface $interface does not exist"
      # Stop the script if we are not in test mode
      test -z "$FWBDEBUG" && exit 1
    }

    diff_intf missing_address "$FWB_ADDRS" "$CURRENT_ADDRS_ALL_SCOPES" add
    diff_intf missing_address "$CURRENT_ADDRS_GLOBAL_SCOPE" "$FWB_ADDRS" del
}

clear_addresses_except_known_interfaces() {
    $IP link show | sed 's/://g' | awk -v IGNORED="$*" \
        'BEGIN {
          split(IGNORED,ignored_arr);
          for (a in ignored_arr) {ignored_dict[ignored_arr[a]]=1;}
        }
        (/state/ && !($2 in ignored_dict)) {print $2;}' | \
        while read intf; do
          echo "# Removing addresses not configured in fwbuilder from interface $intf"
          $FWBDEBUG $IP addr flush dev $intf scope global
          $FWBDEBUG $IP link set $intf down
        done
}

check_file() {
    test -r "$2" || {
        echo "Can not find file $2 referenced by address table object $1"
        exit 1
    }
}

check_run_time_address_table_files() {
    :
```

```
}

load_modules() {
   :
   OPTS=$1
   MODULES_DIR="/lib/modules/`uname -r`/kernel/net/"
   MODULES=$(find $MODULES_DIR -name '*conntrack*' \! -name '*ipv6*'|sed  -e 's/^.*\///' -e
's/\([^\.]\)\..*/\1/')
   echo $OPTS | grep -q nat && {
      MODULES="$MODULES $(find $MODULES_DIR -name '*nat*'|sed  -e 's/^.*\///' -e 's/\([^
\.]\)\..*/\1/')"
   }
   echo $OPTS | grep -q ipv6 && {
      MODULES="$MODULES $(find $MODULES_DIR -name nf_conntrack_ipv6|sed  -e 's/^.*
\///' -e 's/\([^\.]\)\..*/\1/')"
   }
   for module in $MODULES; do
      if $LSMOD | grep ${module} >/dev/null; then continue; fi
      $MODPROBE ${module} ||  exit 1
   done
}

verify_interfaces() {
   :
   echo "Verifying interfaces: eth0 eth1 lo eth2 eth3"
   for i in eth0 eth1 lo eth2 eth3 ; do
      $IP link show "$i" > /dev/null 2>&1 || {
         log "Interface $i does not exist"
         exit 1
      }
   done
}

prolog_commands() {
   echo "Running prolog script"

}

epilog_commands() {
   echo "Running epilog script"

}

run_epilog_and_exit() {
   epilog_commands
   exit $1
}

configure_interfaces() {
   :
   # Configure interfaces
   update_addresses_of_interface "eth0 143.0.100.250/24" ""
```

```
    update_addresses_of_interface "eth1 192.168.1.1/24" ""
    update_addresses_of_interface "lo 127.0.0.1/8" ""
    update_addresses_of_interface "eth2 143.0.100.19/24" ""
    update_addresses_of_interface "eth3 192.168.1.13/24" ""
}

script_body() {
    # ================ IPv4


    # ================ Table 'filter', automatic rules
    # accept established sessions
    $IPTABLES -A INPUT   -m state --state ESTABLISHED,RELATED -j ACCEPT
    $IPTABLES -A OUTPUT  -m state --state ESTABLISHED,RELATED -j ACCEPT
    $IPTABLES -A FORWARD -m state --state ESTABLISHED,RELATED -j ACCEPT


    # ================ Table 'nat',  rule set NAT
    #
    # Rule 0 (NAT)
    #
    echo "Rule 0 (NAT)"
    #
    $IPTABLES -t nat -A POSTROUTING  -j ACCEPT
    $IPTABLES -t nat -A PREROUTING  -j ACCEPT
    #
    # Rule 1 (NAT)
    #
    echo "Rule 1 (NAT)"
    #
    # no need to translate
    # between DMZ and
    # internal net
    $IPTABLES -t nat -A POSTROUTING  -s 192.168.2.0/24  -d 192.168.1.0/24  -j ACCEPT
    $IPTABLES -t nat -A PREROUTING  -s 192.168.2.0/24  -d 192.168.1.0/24  -j ACCEPT
    #
    # Rule 2 (NAT)
    #
    echo "Rule 2 (NAT)"
    #
    # Translate source address
    # for outgoing connections
    $IPTABLES -t nat -A POSTROUTING -o eth0   -s 192.168.1.0/24  -j SNAT --to-source
143.0.100.250
    $IPTABLES -t nat -A POSTROUTING -o eth0   -s 192.168.2.0/24  -j SNAT --to-source
143.0.100.250
    #
    # Rule 3 (NAT)
    #
    echo "Rule 3 (NAT)"
    #
    $IPTABLES -t nat -A PREROUTING  -d 143.0.100.250  -j DNAT --to-destination
192.168.2.10
```

```
# ================ Table 'filter', rule set Policy
#
# Rule 0 (eth0)
#
echo "Rule 0 (eth0)"
#
# anti spoofing rule
$IPTABLES -N ln_RULE_0
$IPTABLES -A INPUT -i eth0   -s 143.0.100.19   -m state --state NEW  -j ln_RULE_0
$IPTABLES -A INPUT -i eth0   -s 143.0.100.250   -m state --state NEW  -j ln_RULE_0
$IPTABLES -A INPUT -i eth0   -s 192.168.1.1   -m state --state NEW  -j ln_RULE_0
$IPTABLES -A INPUT -i eth0   -s 192.168.1.13   -m state --state NEW  -j ln_RULE_0
$IPTABLES -A INPUT -i eth0   -s 192.168.2.0/24   -m state --state NEW  -j ln_RULE_0
$IPTABLES -A FORWARD -i eth0   -s 143.0.100.19   -m state --state NEW  -j ln_RULE_0
$IPTABLES -A FORWARD -i eth0   -s 143.0.100.250   -m state --state NEW  -j ln_RULE_0
$IPTABLES -A FORWARD -i eth0   -s 192.168.1.1   -m state --state NEW  -j ln_RULE_0
$IPTABLES -A FORWARD -i eth0   -s 192.168.1.13   -m state --state NEW  -j ln_RULE_0
$IPTABLES -A FORWARD -i eth0   -s 192.168.2.0/24   -m state --state NEW  -j ln_RULE_0
$IPTABLES -A ln_RULE_0  -j LOG  --log-level info --log-prefix "RULE 0 -- DENY "
$IPTABLES -A ln_RULE_0  -j DROP
#
# Rule 1 (lo)
#
echo "Rule 1 (lo)"
#
# No debería unicamente incluir las redes internas en el
# source?
# Por si me mandan un paquete al FW loopback desde afuera
# pero "marcado" con origen de adentro, lo que devolvería
# el paquete a una maquina interna y la "infectaría"
$IPTABLES -A INPUT -i lo   -m state --state NEW  -j ACCEPT
$IPTABLES -A OUTPUT -o lo   -m state --state NEW  -j ACCEPT
#
# Rule 2 (eth1)
#
echo "Rule 2 (eth1)"
#
# Acceso por SSH al firewall es permitido
# solo a los dos host que pueden tener
# acceso
$IPTABLES -N Cid5358X1621.0
$IPTABLES -A INPUT -i eth1  -p tcp -m tcp  -m multiport  --dports 21,22  -m state --state
NEW  -j Cid5358X1621.0
$IPTABLES -N ln_RULE_2
$IPTABLES -A Cid5358X1621.0  -s 192.168.33.33   -j ln_RULE_2
$IPTABLES -A Cid5358X1621.0  -s 192.168.33.34   -j ln_RULE_2
$IPTABLES -A ln_RULE_2  -j LOG  --log-level info --log-prefix "RULE 2 -- ACCEPT "
$IPTABLES -A ln_RULE_2  -j ACCEPT
#
# Rule 3 (global)
```

```
    #
    echo "Rule 3 (global)"
    #
    # Esta regla (controversial al menos)
    # es para que en caso de necesidad
    # urgente de control o configuración
    # se pueda entrar desde fuera del
    # instituto a las dos unicas PC que
    # tienen acceso directo al FW y se filtra
    # por medio de la dirección fisica de
    # la interface de red para proteger
    # el acceso desde afuera de la red
    # Fuente: https://es.wikipedia.org/wiki/VNC#Funcionamiento
    $IPTABLES -N Cid4056X1650.0
    $IPTABLES -A FORWARD -i +  -p tcp -m tcp  -m mac --mac-source 00:00:00:00:00:00  --
dport 22  -m state --state NEW  -j Cid4056X1650.0
    $IPTABLES -N Cid4056X1650.1
    $IPTABLES -A Cid4056X1650.0  -d 192.168.33.33   -j Cid4056X1650.1
    $IPTABLES -A Cid4056X1650.0  -d 192.168.33.34   -j Cid4056X1650.1
    $IPTABLES -N In_RULE_3
    $IPTABLES -A Cid4056X1650.1 -m time  --timestart 18:00  --timestop 23:59  -j In_RULE_3
    $IPTABLES -A Cid4056X1650.1 -m time  --timestart 00:00  --timestop 23:59  --days
Sat,Sun -j In_RULE_3
    $IPTABLES -A In_RULE_3  -j LOG  --log-level info --log-prefix "RULE 3 -- ACCEPT "
    $IPTABLES -A In_RULE_3  -j ACCEPT
    #
    # Rule 4 (global)
    #
    echo "Rule 4 (global)"
    #
    # Todos los demas intentos de conectarse
    # al FW son denegados y se registran en logs
    $IPTABLES -N RULE_4
    $IPTABLES -A OUTPUT  -d 143.0.100.19   -m state --state NEW  -j RULE_4
    $IPTABLES -A OUTPUT  -d 143.0.100.250   -m state --state NEW  -j RULE_4
    $IPTABLES -A OUTPUT  -d 192.168.1.1   -m state --state NEW  -j RULE_4
    $IPTABLES -A OUTPUT  -d 192.168.1.13   -m state --state NEW  -j RULE_4
    $IPTABLES -A INPUT  -m state --state NEW  -j RULE_4
    $IPTABLES -A RULE_4  -j LOG  --log-level info --log-prefix "RULE 4 -- DENY "
    $IPTABLES -A RULE_4  -j DROP
    #
    # Rule 5 (eth3)
    #
    echo "Rule 5 (eth3)"
    #
    # El FW usa uno de los servidores
    # en la red interna para DNS que
    # se alojan en la DMZ Interna.
    $IPTABLES -N Cid5386X1621.0
    $IPTABLES -A FORWARD -i eth3  -p tcp -m tcp  -d 192.168.1.10   --dport 53  -m state --
state NEW  -j Cid5386X1621.0
    $IPTABLES -A FORWARD -i eth3  -p udp -m udp  -d 192.168.1.10   --dport 53  -m state --
state NEW  -j Cid5386X1621.0
```

```
   $IPTABLES -A Cid5386X1621.0  -s 143.0.100.19   -j ACCEPT
   $IPTABLES -A Cid5386X1621.0  -s 143.0.100.250   -j ACCEPT
   $IPTABLES -A Cid5386X1621.0  -s 192.168.1.1   -j ACCEPT
   $IPTABLES -A Cid5386X1621.0  -s 192.168.1.13   -j ACCEPT
   $IPTABLES -A OUTPUT -o eth3  -p tcp -m tcp  -d 192.168.1.10   --dport 53  -m state --state
NEW  -j ACCEPT
   $IPTABLES -A OUTPUT -o eth3  -p udp -m udp  -d 192.168.1.10   --dport 53  -m state --
state NEW  -j ACCEPT
   #
   # Rule 6 (global)
   #
   echo "Rule 6 (global)"
   #
   # Port 113 used for Identification/Authorization service. When a client
   # program on your end contacts a remote server for services such as
   # POP, IMAP, SMTP, IRC, FTP, etc. that remote server sends back a
   # query to the IDENT port 113 asking for identification from your system...
   # Port 113 can be probed by attackers and it poses some security
   #  concerns, but the problem with filtering/stealthing port 113 is that
   # if legitimate requests get no response at all from port 113 queries,
   #  the connection to them (which initiated their query in the first place)
   #  will be delayed or perhaps even completely abandoned.
   # Fuente: http://www.speedguide.net/port.php?port=113
   $IPTABLES -A OUTPUT -p tcp -m tcp  --dport 113  -j REJECT
   $IPTABLES -A INPUT -p tcp -m tcp  --dport 113  -j REJECT
   $IPTABLES -A FORWARD -p tcp -m tcp  --dport 113  -j REJECT
   #
   # Rule 7 (eth0)
   #
   echo "Rule 7 (eth0)"
   #
   # Mail relay on DMZ can accept
   # connections from hosts on the
   # Internet
   $IPTABLES -A FORWARD -i eth0  -p tcp -m tcp  -d 192.168.2.10   --dport 25  -m state --
state NEW  -j ACCEPT
   $IPTABLES -A OUTPUT -o eth0  -p tcp -m tcp  -d 192.168.2.10   --dport 25  -m state --state
NEW  -j ACCEPT
   $IPTABLES -A FORWARD -o eth0  -p tcp -m tcp  -d 192.168.2.10   --dport 25  -m state --
state NEW  -j ACCEPT
   #
   # Rule 8 (global)
   #
   echo "Rule 8 (global)"
   #
   # this rule permits a mail relay
   # located on DMZ to connect
   # to internal mail server
   # IDEM REGLA DE ARRIBA
   $IPTABLES -A FORWARD -p tcp -m tcp  -s 192.168.2.10   -d 192.168.1.10   --dport 25  -m
state --state NEW  -j ACCEPT
   #
   # Rule 9 (global)
```

```
#
echo "Rule 9 (global)"
#
# Mail relay needs DNS and can
# connect to mail servers on the
# Internet
$IPTABLES -A INPUT -p tcp -m tcp  -m multiport  -s 192.168.2.10   -d ! 192.168.1.0/24   --
dports 53,25  -m state --state NEW  -j ACCEPT
$IPTABLES -A INPUT -p udp -m udp  -s 192.168.2.10   -d ! 192.168.1.0/24   --dport 53  -m
state --state NEW  -j ACCEPT
$IPTABLES -A FORWARD -p tcp -m tcp  -m multiport  -s 192.168.2.10   -d !
192.168.1.0/24   --dports 53,25  -m state --state NEW  -j ACCEPT
$IPTABLES -A FORWARD -p udp -m udp  -s 192.168.2.10   -d ! 192.168.1.0/24   --dport
53  -m state --state NEW  -j ACCEPT
#
# Rule 10 (global)
#
echo "Rule 10 (global)"
#
# Cualquier acceso desde
# la DMZ a cualquier red
# interna es denegado.
# Ahora bien, no debería
# ser solamente en una
# direccion?
$IPTABLES -N Cid5555X1621.0
$IPTABLES -A INPUT  -s 192.168.2.10   -j Cid5555X1621.0
$IPTABLES -N RULE_10
$IPTABLES -A Cid5555X1621.0  -d 192.168.1.0/24   -j RULE_10
$IPTABLES -A Cid5555X1621.0  -d 192.168.2.0/24   -j RULE_10
$IPTABLES -A Cid5555X1621.0  -d 192.168.3.0/22   -j RULE_10
$IPTABLES -A Cid5555X1621.0  -d 192.168.4.0/22   -j RULE_10
$IPTABLES -A Cid5555X1621.0  -d 192.168.7.0/22   -j RULE_10
$IPTABLES -A Cid5555X1621.0  -d 192.168.9.0/22   -j RULE_10
$IPTABLES -A Cid5555X1621.0  -d 192.168.10.0/22   -j RULE_10
$IPTABLES -A Cid5555X1621.0  -d 192.168.11.0/22   -j RULE_10
$IPTABLES -A Cid5555X1621.0  -d 192.168.13.0/22   -j RULE_10
$IPTABLES -A Cid5555X1621.0  -d 192.168.15.0/22   -j RULE_10
$IPTABLES -A Cid5555X1621.0  -d 192.168.32.0/22   -j RULE_10
$IPTABLES -A Cid5555X1621.0  -d 192.168.116.10   -j RULE_10
$IPTABLES -N Cid5555X1621.1
$IPTABLES -A FORWARD  -s 192.168.2.10   -j Cid5555X1621.1
$IPTABLES -A Cid5555X1621.1  -d 192.168.1.0/24   -j RULE_10
$IPTABLES -A Cid5555X1621.1  -d 192.168.2.0/24   -j RULE_10
$IPTABLES -A Cid5555X1621.1  -d 192.168.3.0/22   -j RULE_10
$IPTABLES -A Cid5555X1621.1  -d 192.168.4.0/22   -j RULE_10
$IPTABLES -A Cid5555X1621.1  -d 192.168.7.0/22   -j RULE_10
$IPTABLES -A Cid5555X1621.1  -d 192.168.9.0/22   -j RULE_10
$IPTABLES -A Cid5555X1621.1  -d 192.168.10.0/22   -j RULE_10
$IPTABLES -A Cid5555X1621.1  -d 192.168.11.0/22   -j RULE_10
$IPTABLES -A Cid5555X1621.1  -d 192.168.13.0/22   -j RULE_10
$IPTABLES -A Cid5555X1621.1  -d 192.168.15.0/22   -j RULE_10
$IPTABLES -A Cid5555X1621.1  -d 192.168.32.0/22   -j RULE_10
```

```
$IPTABLES -A Cid5555X1621.1  -d 192.168.116.10   -j RULE_10
$IPTABLES -A RULE_10  -j LOG  --log-level info --log-prefix "RULE 10 -- DENY "
$IPTABLES -A RULE_10  -j DROP
#
# Rule 11 (global)
#
echo "Rule 11 (global)"
#
# Se rechazan el acceso a redes de compraventa
$IPTABLES -N Cid3924X3855.0
$IPTABLES -A FORWARD -p tcp -m tcp  -m multiport  --dports 80,443  -j Cid3924X3855.0
$IPTABLES -N Cid3924X3855.1
$IPTABLES -A Cid3924X3855.0  -s 192.168.2.0/24   -j Cid3924X3855.1
$IPTABLES -A Cid3924X3855.0  -s 192.168.10.0/22   -j Cid3924X3855.1
$IPTABLES -A Cid3924X3855.0  -s 192.168.11.0/22   -j Cid3924X3855.1
$IPTABLES -A Cid3924X3855.0  -s 192.168.15.0/22   -j Cid3924X3855.1
$IPTABLES -A Cid3924X3855.0  -s 192.168.16.0/22   -j Cid3924X3855.1
$IPTABLES -A Cid3924X3855.0  -s 192.168.20.0/22   -j Cid3924X3855.1
$IPTABLES -N RULE_11
$IPTABLES -A Cid3924X3855.1  -d 8.18.145.0/24   -j RULE_11
$IPTABLES -A Cid3924X3855.1  -d 8.45.162.0/24   -j RULE_11
$IPTABLES -A Cid3924X3855.1  -d 13.32.1.0/24   -j RULE_11
$IPTABLES -A Cid3924X3855.1  -d 13.32.2.0/23   -j RULE_11
$IPTABLES -A Cid3924X3855.1  -d 13.32.4.0/22   -j RULE_11
$IPTABLES -A Cid3924X3855.1  -d 13.32.4.0/23   -j RULE_11
$IPTABLES -A Cid3924X3855.1  -d 13.32.6.0/23   -j RULE_11
$IPTABLES -A Cid3924X3855.1  -d 13.32.8.0/21   -j RULE_11
$IPTABLES -A Cid3924X3855.1  -d 13.32.8.0/22   -j RULE_11
$IPTABLES -A Cid3924X3855.1  -d 13.32.12.0/22   -j RULE_11
$IPTABLES -A Cid3924X3855.1  -d 13.32.16.0/20   -j RULE_11
$IPTABLES -A Cid3924X3855.1  -d 13.32.16.0/24   -j RULE_11

(…)

$IPTABLES -A Cid3924X3855.1  -d 207.171.184.0/21   -j RULE_11
$IPTABLES -A Cid3924X3855.1  -d 207.171.184.0/23   -j RULE_11
$IPTABLES -A Cid3924X3855.1  -d 207.248.68.0/23   -j RULE_11
$IPTABLES -A Cid3924X3855.1  -d 209.173.58.0/24   -j RULE_11
$IPTABLES -A Cid3924X3855.1  -d 209.225.49.0/24   -j RULE_11
$IPTABLES -A Cid3924X3855.1  -d 214.16.96.0/24   -j RULE_11
$IPTABLES -A Cid3924X3855.1  -d 216.33.196.0/24   -j RULE_11
$IPTABLES -A Cid3924X3855.1  -d 216.33.197.0/24   -j RULE_11
$IPTABLES -A Cid3924X3855.1  -d 216.113.172.0/23   -j RULE_11
$IPTABLES -A Cid3924X3855.1  -d 216.113.175.0/24   -j RULE_11
$IPTABLES -A Cid3924X3855.1  -d 216.113.176.0/24   -j RULE_11
$IPTABLES -A Cid3924X3855.1  -d 216.137.36.0/23   -j RULE_11
$IPTABLES -A Cid3924X3855.1  -d 216.137.39.0/24   -j RULE_11
$IPTABLES -A Cid3924X3855.1  -d 216.137.41.0/24   -j RULE_11
$IPTABLES -A Cid3924X3855.1  -d 216.137.42.0/24   -j RULE_11
$IPTABLES -A Cid3924X3855.1  -d 216.137.43.0/24   -j RULE_11
$IPTABLES -A Cid3924X3855.1  -d 216.137.44.0/24   -j RULE_11
$IPTABLES -A Cid3924X3855.1  -d 216.137.45.0/24   -j RULE_11
$IPTABLES -A Cid3924X3855.1  -d 216.137.48.0/23   -j RULE_11
```

```
   $IPTABLES -A Cid3924X3855.1  -d 216.137.50.0/23   -j RULE_11
   $IPTABLES -A Cid3924X3855.1  -d 216.137.52.0/23   -j RULE_11
   $IPTABLES -A Cid3924X3855.1  -d 216.137.52.0/24   -j RULE_11
   $IPTABLES -A Cid3924X3855.1  -d 216.137.53.0/24   -j RULE_11
   $IPTABLES -A Cid3924X3855.1  -d 216.137.56.0/24   -j RULE_11
   $IPTABLES -A Cid3924X3855.1  -d 216.137.57.0/24   -j RULE_11
   $IPTABLES -A Cid3924X3855.1  -d 216.137.58.0/24   -j RULE_11
   $IPTABLES -A Cid3924X3855.1  -d 216.137.59.0/24   -j RULE_11
   $IPTABLES -A Cid3924X3855.1  -d 216.137.60.0/23   -j RULE_11
   $IPTABLES -A Cid3924X3855.1  -d 216.137.61.0/24   -j RULE_11
   $IPTABLES -A Cid3924X3855.1  -d 216.137.62.0/24   -j RULE_11
   $IPTABLES -A Cid3924X3855.1  -d 216.137.63.0/24   -j RULE_11
   $IPTABLES -A Cid3924X3855.1  -d 216.182.224.0/21   -j RULE_11
   $IPTABLES -A Cid3924X3855.1  -d 216.182.232.0/21   -j RULE_11
   $IPTABLES -A RULE_11  -j LOG  --log-level info --log-prefix "RULE 11 -- REJECT "
   $IPTABLES -A RULE_11  -j REJECT  --reject-with icmp-net-prohibited
   #
   # Rule 12 (global)
   #
   echo "Rule 12 (global)"
   #
   # Se rechazan el acceso a redes de social media
   $IPTABLES -N Cid5611X1621.0
   $IPTABLES -A OUTPUT -p tcp -m tcp  -m multiport  --dports 80,443  -j Cid5611X1621.0
   $IPTABLES -N Cid5611X1621.1
   $IPTABLES -A Cid5611X1621.0  -s 192.168.1.0/24   -j Cid5611X1621.1
   $IPTABLES -A Cid5611X1621.0  -s 192.168.3.0/22   -j Cid5611X1621.1
   $IPTABLES -A Cid5611X1621.0  -s 192.168.4.0/22   -j Cid5611X1621.1
   $IPTABLES -A Cid5611X1621.0  -s 192.168.6.0/22   -j Cid5611X1621.1
   $IPTABLES -A Cid5611X1621.0  -s 192.168.7.0/22   -j Cid5611X1621.1
   $IPTABLES -A Cid5611X1621.0  -s 192.168.8.0/22   -j Cid5611X1621.1
   $IPTABLES -A Cid5611X1621.0  -s 192.168.9.0/22   -j Cid5611X1621.1
   $IPTABLES -A Cid5611X1621.0  -s 192.168.13.0/22   -j Cid5611X1621.1
   $IPTABLES -A Cid5611X1621.0  -s 192.168.17.0/22   -j Cid5611X1621.1
   $IPTABLES -A Cid5611X1621.0  -s 192.168.18.0/22   -j Cid5611X1621.1
   $IPTABLES -A Cid5611X1621.0  -s 192.168.21.0/22   -j Cid5611X1621.1
   $IPTABLES -A Cid5611X1621.1  -d 8.22.161.0/24   -j REJECT  --reject-with icmp-net-
prohibited
   $IPTABLES -A Cid5611X1621.1  -d 8.25.194.0/23   -j REJECT  --reject-with icmp-net-
prohibited
   $IPTABLES -A Cid5611X1621.1  -d 8.25.195.0/24   -j REJECT  --reject-with icmp-net-
prohibited
   $IPTABLES -A Cid5611X1621.1  -d 8.25.196.0/23   -j REJECT  --reject-with icmp-net-
prohibited
   $IPTABLES -A Cid5611X1621.1  -d 8.25.196.0/24   -j REJECT  --reject-with icmp-net-
prohibited
   $IPTABLES -A Cid5611X1621.1  -d 8.39.53.0/24   -j REJECT  --reject-with icmp-net-
prohibited
   $IPTABLES -A Cid5611X1621.1  -d 8.39.61.0/24   -j REJECT  --reject-with icmp-net-
prohibited
   $IPTABLES -A Cid5611X1621.1  -d 31.13.24.0/21   -j REJECT  --reject-with icmp-net-
prohibited
```

```
(…)

   $IPTABLES -A Cid5611X1621.1  -d 208.117.255.0/24   -j REJECT  --reject-with icmp-net-
prohibited
   $IPTABLES -A Cid5611X1621.1  -d 209.237.192.0/19   -j REJECT  --reject-with icmp-net-
prohibited
   $IPTABLES -A Cid5611X1621.1  -d 216.52.16.0/24   -j REJECT  --reject-with icmp-net-
prohibited
   $IPTABLES -A Cid5611X1621.1  -d 216.52.17.0/24   -j REJECT  --reject-with icmp-net-
prohibited
   $IPTABLES -A Cid5611X1621.1  -d 216.52.18.0/24   -j REJECT  --reject-with icmp-net-
prohibited
   $IPTABLES -A Cid5611X1621.1  -d 216.52.20.0/24   -j REJECT  --reject-with icmp-net-
prohibited
   $IPTABLES -A Cid5611X1621.1  -d 216.52.21.0/24   -j REJECT  --reject-with icmp-net-
prohibited
   $IPTABLES -A Cid5611X1621.1  -d 216.52.22.0/24   -j REJECT  --reject-with icmp-net-
prohibited
   $IPTABLES -A Cid5611X1621.1  -d 216.73.80.0/20   -j REJECT  --reject-with icmp-net-
prohibited
   $IPTABLES -N Cid5611X1621.2
   $IPTABLES -A FORWARD -p tcp -m tcp  -m multiport  --dports 80,443  -j Cid5611X1621.2
   $IPTABLES -N Cid5611X1621.3
   $IPTABLES -A Cid5611X1621.2  -s 192.168.1.0/24   -j Cid5611X1621.3
   $IPTABLES -A Cid5611X1621.2  -s 192.168.3.0/22   -j Cid5611X1621.3
   $IPTABLES -A Cid5611X1621.2  -s 192.168.4.0/22   -j Cid5611X1621.3
   $IPTABLES -A Cid5611X1621.2  -s 192.168.6.0/22   -j Cid5611X1621.3
   $IPTABLES -A Cid5611X1621.2  -s 192.168.7.0/22   -j Cid5611X1621.3
   $IPTABLES -A Cid5611X1621.2  -s 192.168.8.0/22   -j Cid5611X1621.3
   $IPTABLES -A Cid5611X1621.2  -s 192.168.9.0/22   -j Cid5611X1621.3
   $IPTABLES -A Cid5611X1621.2  -s 192.168.13.0/22   -j Cid5611X1621.3
   $IPTABLES -A Cid5611X1621.2  -s 192.168.17.0/22   -j Cid5611X1621.3
   $IPTABLES -A Cid5611X1621.2  -s 192.168.18.0/22   -j Cid5611X1621.3
   $IPTABLES -A Cid5611X1621.2  -s 192.168.21.0/22   -j Cid5611X1621.3
   $IPTABLES -A Cid5611X1621.3  -d 8.22.161.0/24   -j REJECT  --reject-with icmp-net-
prohibited
   $IPTABLES -A Cid5611X1621.3  -d 8.25.194.0/23   -j REJECT  --reject-with icmp-net-
prohibited
   $IPTABLES -A Cid5611X1621.3  -d 8.25.195.0/24   -j REJECT  --reject-with icmp-net-
prohibited
   $IPTABLES -A Cid5611X1621.3  -d 8.25.196.0/23   -j REJECT  --reject-with icmp-net-
prohibited
   $IPTABLES -A Cid5611X1621.3  -d 8.25.196.0/24   -j REJECT  --reject-with icmp-net-
prohibited
   $IPTABLES -A Cid5611X1621.3  -d 8.39.53.0/24   -j REJECT  --reject-with icmp-net-
prohibited
   $IPTABLES -A Cid5611X1621.3  -d 8.39.61.0/24   -j REJECT  --reject-with icmp-net-
prohibited

   (…)

   $IPTABLES -A Cid5611X1621.3  -d 216.52.16.0/24   -j REJECT  --reject-with icmp-net-
prohibited
```

```
   $IPTABLES -A Cid5611X1621.3  -d 216.52.17.0/24   -j REJECT  --reject-with icmp-net-
prohibited
   $IPTABLES -A Cid5611X1621.3  -d 216.52.18.0/24   -j REJECT  --reject-with icmp-net-
prohibited
   $IPTABLES -A Cid5611X1621.3  -d 216.52.20.0/24   -j REJECT  --reject-with icmp-net-
prohibited
   $IPTABLES -A Cid5611X1621.3  -d 216.52.21.0/24   -j REJECT  --reject-with icmp-net-
prohibited
   $IPTABLES -A Cid5611X1621.3  -d 216.52.22.0/24   -j REJECT  --reject-with icmp-net-
prohibited
   $IPTABLES -A Cid5611X1621.3  -d 216.73.80.0/20   -j REJECT  --reject-with icmp-net-
prohibited
   #
   # Rule 13 (global)
   #
   echo "Rule 13 (global)"
   #
   $IPTABLES -N Cid4116X1517.0
   $IPTABLES -A INPUT -p tcp -m tcp  --sport 20  --dport 1024:65535  -m state --state NEW  -
j Cid4116X1517.0
   $IPTABLES -A INPUT -p tcp -m tcp  -m multiport  --dports 21,20  -m state --state NEW  -j
Cid4116X1517.0
   $IPTABLES -N RULE_13
   $IPTABLES -A Cid4116X1517.0  -s 192.168.33.33   -j RULE_13
   $IPTABLES -A Cid4116X1517.0  -s 192.168.33.34   -j RULE_13
   $IPTABLES -N Cid4116X1517.1
   $IPTABLES -A FORWARD -p tcp -m tcp  --sport 20  --dport 1024:65535  -m state --state
NEW  -j Cid4116X1517.1
   $IPTABLES -A FORWARD -p tcp -m tcp  -m multiport  --dports 21,20  -m state --state
NEW  -j Cid4116X1517.1
   $IPTABLES -A Cid4116X1517.1  -s 192.168.33.33   -j RULE_13
   $IPTABLES -A Cid4116X1517.1  -s 192.168.33.34   -j RULE_13
   $IPTABLES -A RULE_13  -j LOG  --log-level info --log-prefix "RULE 13 -- ACCEPT "
   $IPTABLES -A RULE_13  -j ACCEPT
   #
   # Rule 14 (global)
   #
   echo "Rule 14 (global)"
   #
   $IPTABLES -N RULE_14
   $IPTABLES -A FORWARD -p tcp -m tcp  -s 192.168.14.0/22   --sport 20  -d
192.168.23.0/22   --dport 1024:65535  -m state --state NEW  -j RULE_14
   $IPTABLES -A FORWARD -p tcp -m tcp  -m multiport  -s 192.168.14.0/22   -d
192.168.23.0/22   --dports 21,20  -m state --state NEW  -j RULE_14
   $IPTABLES -A RULE_14  -j LOG  --log-level info --log-prefix "RULE 14 -- ACCEPT "
   $IPTABLES -A RULE_14  -j ACCEPT
   #
   # Rule 15 (global)
   #
   echo "Rule 15 (global)"
   #
   # Por defecto se rechaza toda otra conexion no explicitada anteriormente
   $IPTABLES -N RULE_15
```

```
$IPTABLES -A OUTPUT  -j RULE_15
$IPTABLES -A INPUT  -j RULE_15
$IPTABLES -A FORWARD  -j RULE_15
$IPTABLES -A RULE_15  -j LOG  --log-level info --log-prefix "RULE 15 -- REJECT "
$IPTABLES -A RULE_15  -j REJECT


# ============== ROUTING RULES ==============

HAVE_MKTEMP=$(which mktemp)

test -n "$HAVE_MKTEMP" && {
  TMPDIRNAME=$(mktemp -d)
  test -z "$TMPDIRNAME" && exit 1
}

test -z "$HAVE_MKTEMP" && {
  TMPDIRNAME="/tmp/.fwbuilder.tempdir.$$"
  (umask 077 && mkdir $TMPDIRNAME) || exit 1
}

TMPFILENAME="$TMPDIRNAME/.fwbuilder.out"
OLD_ROUTES="$TMPDIRNAME/.old_routes"

#
# This function stops stdout redirection
# and sends previously saved output to terminal
restore_script_output()
{
  exec 1>&3 2>&1
  cat $TMPFILENAME
  rm -rf $TMPDIRNAME
}

# if any routing rule fails we do our best to prevent freezing the firewall
route_command_error()
{
  echo "Error: Routing rule $1 couldn't be activated"
  echo "Recovering previous routing configuration..."
  # delete current routing rules
  $IP route show | while read route ; do $IP route del $route ; done
  # restore old routing rules
  sh $OLD_ROUTES
  echo "...done"
  restore_script_output
  epilog_commands
  exit 1
}

# redirect output to prevent ssh session from stalling
exec 3>&1
exec 1> $TMPFILENAME
exec 2>&1
```

```
    # store previous routing configuration (sort: 'via' GW has to be
    # inserted after device routes)

    $IP route show | sort -k 2 | awk '{printf "ip route add %s\n",$0;}' > $OLD_ROUTES

    echo "Deleting routing rules previously set by user space processes..."
    $IP route show | grep -v '\( proto kernel \)\|\(default via \)' | \
        while read route ; do $IP route del $route ; done

    echo "Activating non-ecmp routing rules..."
    #
    # Rule 0 (main)
    #
    echo "Routing rule 0 (main)"
    #
    #
    #
    $IP route add 192.168.24.0/22  metric 10 dev eth0 \
    || route_command_error "0 (main)"

    $IP route add 192.168.28.0/22  metric 10 dev eth0 \
    || route_command_error "0 (main)"

    $IP route add 192.168.32.0/22  metric 10 dev eth0 \
    || route_command_error "0 (main)"

    $IP route add 192.168.36.0/22  metric 10 dev eth0 \
    || route_command_error "0 (main)"

    $IP route add 192.168.40.0/22  metric 10 dev eth0 \
    || route_command_error "0 (main)"


    restore_script_output
    echo "...done."
}

ip_forward() {
    :
    echo 1 > /proc/sys/net/ipv4/ip_forward
}

reset_all() {
    :
    reset_iptables_v4
}

block_action() {
    reset_all
}

stop_action() {
```

```
    reset_all
    $IPTABLES -P OUTPUT  ACCEPT
    $IPTABLES -P INPUT   ACCEPT
    $IPTABLES -P FORWARD ACCEPT
}

check_iptables() {
    IP_TABLES="$1"
    [ ! -e $IP_TABLES ] && return 151
    NF_TABLES=$(cat $IP_TABLES 2>/dev/null)
    [ -z "$NF_TABLES" ] && return 152
    return 0
}
status_action() {
    check_iptables "/proc/net/ip_tables_names"
    ret_ipv4=$?
    check_iptables "/proc/net/ip6_tables_names"
    ret_ipv6=$?
    [ $ret_ipv4 -eq 0 -o $ret_ipv6 -eq 0 ] && return 0
    [ $ret_ipv4 -eq 151 -o $ret_ipv6 -eq 151 ] && {
        echo "iptables modules are not loaded"
    }
    [ $ret_ipv4 -eq 152 -o $ret_ipv6 -eq 152 ] && {
        echo "Firewall is not configured"
    }
    exit 3
}

# See how we were called.
# For backwards compatibility missing argument is equivalent to 'start'

cmd=$1
test -z "$cmd" && {
    cmd="start"
}

case "$cmd" in
    start)
        log "Activating firewall script generated Mon Sep 03 19:18:53 2018 by Mato"
        check_tools
         prolog_commands
        check_run_time_address_table_files

        load_modules "nat "
        configure_interfaces
        verify_interfaces

         reset_all

        script_body
        ip_forward
        epilog_commands
        RETVAL=$?
```

```
        ;;

    stop)
        stop_action
        RETVAL=$?
        ;;

    status)
        status_action
        RETVAL=$?
        ;;

    block)
        block_action
        RETVAL=$?
        ;;

    reload)
        $0 stop
        $0 start
        RETVAL=$?
        ;;

    interfaces)
        configure_interfaces
        RETVAL=$?
        ;;

    test_interfaces)
        FWBDEBUG="echo"
        configure_interfaces
        RETVAL=$?
        ;;



    *)
        echo "Usage $0 [start|stop|status|block|reload|interfaces|test_interfaces]"
        ;;

esac

exit $RETVAL
```