Caracterización de *firewall* de alta disponibilidad con filtro de contenido en un ambiente virtualizado.

Reporte Técnico de Investigación presentado por:
Alejandra Juana Torres Pérez 98709
Lucero Martínez Castrillo 98750

Requisito para la obtención del título de

**INGENIERO EN SISTEMAS COMPUTACIONALES**

Profesor Responsable: Mtro. Eduardo Castillo Luna

Mayo de 2015
Autorización de Impresión

Los abajo firmantes, miembros del comité evaluador autorizamos la impresión del proyecto de titulación

Caracterización de *firewall* de alta disponibilidad con filtro de contenido en un ambiente virtualizado.

Elaborado por las alumnas:
Alejandra Juana Torres Pérez 98709
Lucero Martínez Castrillo 98750

Mtro. Fernando Estrada Saldaña
Profesor de la Materia

Mtro. Eduardo Castillo Luna
Asesor Técnico
Declaración de Originalidad

Nosotras, Alejandra Juana Torres Pérez y Lucero Martínez Castrillo, declaramos que el material contenido en esta publicación fue generado con la revisión de los documentos que se mencionan en la sección de Referencias y que el prototipo desarrollado es original y no ha sido copiado de ninguna otra fuente, ni ha sido usado para obtener otro título o reconocimiento en otra Institución de Educación Superior.

Alejandra Juana Torres Pérez

Lucero Martínez Castrillo
Índice de contenidos

Autorización de Impresión ............................................................................................................. ¡Error! Marcador no definido.
Declaración de Originalidad ........................................................................................................ ¡Error! Marcador no definido.

Lista de Figuras .......................................................................................................................... viii
Lista de Tablas .............................................................................................................................. ix

Introducción ............................................................................................................................... 1

Capítulo 1. Planteamiento del problema ......................................................................................... 2
  1.1 Antecedentes ......................................................................................................................... 2
  1.2 Definición del problema ......................................................................................................... 4
  1.3 Objetivos de la investigación ................................................................................................. 4
  1.4 Preguntas de investigación ................................................................................................... 5
  1.5 Justificación de la investigación .......................................................................................... 5
  1.6 Limitaciones y delimitaciones de la investigación ................................................................. 6

Capítulo 2. Marco Teórico .............................................................................................................. 7
  2.1 Seguridad informática ........................................................................................................... 7
    2.1.1 Tipos de Seguridad ........................................................................................................... 8
    2.1.2 Seguridad en redes de comunicación .............................................................................. 9
  2.2 Firewall ................................................................................................................................. 10
    2.2.1 Ventajas de un firewall .................................................................................................... 10
    2.2.2 Limitaciones de un firewall ............................................................................................ 11
    2.2.3 Políticas de un firewall .................................................................................................. 12
    2.2.4 Tipos de firewall ............................................................................................................. 12
  2.3 Virtualización ......................................................................................................................... 15
    2.3.1 Recursos .......................................................................................................................... 17
      2.3.1.1 Hipervisores .............................................................................................................. 17
      2.3.1.2 Máquinas Virtuales ................................................................................................ 19
  2.4 Red de Campus Institucionales .............................................................................................. 20
    2.4.1 Concurrencia de Usuarios ............................................................................................. 22
    2.4.2 Alta Disponibilidad ......................................................................................................... 22
    2.4.3 Packet Filter ................................................................................................................... 25
2.4.3.1 Funcionamiento de Packet Filter .............................................. 26
2.4.4 CARP ......................................................................................... 29

2.5 Integración de Tecnologías ........................................................... 30
2.5.1 Herramientas de licencia libre ..................................................... 31
2.5.1.1 DansGuardian ....................................................................... 32
2.5.1.2 OpenBSD ............................................................................. 32
2.5.1.3 Kali Linux ........................................................................... 35

Capítulo 3. Materiales y Métodos ......................................................... 36
3.1 Descripción del área de estudio ....................................................... 37
3.2 Materiales ..................................................................................... 37
3.2.1 Hardware ................................................................................ 37
3.2.2 Software .................................................................................. 38
3.3 Métodos ....................................................................................... 40
3.3.1 Técnicas de investigación .......................................................... 40
3.3.2 Metodología ............................................................................ 40
3.3.3 Procedimiento ......................................................................... 41
3.3.4 Packet Filter ........................................................................... 49
3.3.4.1 Network Address Translation ............................................. 49
3.3.4.2 Reglas de Filtrado de contenido ........................................... 51
3.3.5 DansGuardian (Filtro de contenido web) .................................. 56
3.3.6 Pruebas de penetración ............................................................. 60
3.3.7 Contratiempos durante la integración ........................................ 61

Capítulo 4. Resultados de la investigación ........................................... 62
4.1 Presentación de resultados ............................................................ 62
4.1.1 Resultados de pruebas realizadas ............................................. 62
4.2 Análisis e interpretación de resultados .......................................... 67

Capítulo 5. Discusiones, conclusiones y recomendaciones .................. 68
5.1 Con respecto a las preguntas de investigación ................................ 68
5.2 Con respecto al objetivo de la investigación ................................... 70
5.3 Recomendaciones para futuras investigaciones ............................. 71

Referencias ......................................................................................... 72
Apéndices…………………………………………………………………………………………..74
Anexo I. Características entre Hipervisores..................................................................74
Anexo II. Protocolo corto............................................................................................79
Índice de Figuras

Figura 1. Esquema básico de PF habilitado en una tarjeta de red..........................28
Figura 2. Esquema avanzado de PF aplicado a dos interfaces...............................28
Figura 3. Metodología del proyecto.................................................................42
Figura 4. Esquema Firewall .................................................................43
Figura 5. Red Externa......................................................................................44
Figura 6. Diagrama de red externa.................................................................44
Figura 7. Red CARP.......................................................................................44
Figura 8. Diagrama de red CARP.................................................................44
Figura 9. Red Interna......................................................................................45
Figura 10. Diagrama de red interna.................................................................45
Figura 11. Propiedades de la Máquina virtual Firewall1.....................................46
Figura 12. Diagrama de los firewall..................................................................47
Figura 13. Configuraciones pfsync.................................................................48
Figura 14. Configuraciones CARP.................................................................49
Figura 15. Comprobación de estado de NAT..................................................50
Figura 16. Autorización de tráfico.................................................................53
Figura 17. Ping hacia firewall 1 desde red interna. ...........................................55
Figura 18. Ping de firewall 1 hacia la red interna. .............................................55
Figura 19. Prueba de alta disponibilidad. .......................................................53
Figura 20. Nping al firewall1 por puerto 110...................................................63
Figura 21. Nping al firewall2 por puerto 80.....................................................64
Figura 22. Nping al firewall2 por puerto 80 continuación. ...............................64
Figura 23. Resultado Nmap firewall1 con Zenmap.......................................65
Figura 24. Nmap escaneo general firewall1....................................................65
Figura 25. Detalles utilizando Zenmap a firewall1.........................................66
Figura 26. Topología del resultado al escanear firewall1 y firewall2..................66
Figura 27. Nmap firewall1..............................................................................66
Figura 28. Nmap firewall2..............................................................................67
**Lista de Tablas**

<table>
<thead>
<tr>
<th>Tabla</th>
<th>Descripción</th>
<th>Página</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Porcentajes de inactividad en Alta disponibilidad</td>
<td>24</td>
</tr>
<tr>
<td>2</td>
<td>Representación de lista de reglas</td>
<td>27</td>
</tr>
<tr>
<td>3</td>
<td>Plataformas soportadas por OpenBSD Firewall1 IPs</td>
<td>34</td>
</tr>
<tr>
<td>4</td>
<td>Firewall1 IPs</td>
<td>47</td>
</tr>
<tr>
<td>5</td>
<td>Firewall2 IPs</td>
<td>47</td>
</tr>
<tr>
<td>6</td>
<td>Opciones de Configuración DansGuardian</td>
<td>57</td>
</tr>
</tbody>
</table>
Introducción

Este proyecto surgió por la necesidad que se presenta dentro de las redes públicas en las cuales se vio la falta de sistemas de seguridad que ofrecieran a los administradores otra opción poco convencional, pero de igual manera tenga las funciones de uno convencional y a menor costo.

La realización de este proyecto constó en la creación de un sistema de seguridad que incorpora varias herramientas, las cuales al ser de software libre pero no por ello menos seguras otorgan beneficios para las redes institucionales, estos son alta disponibilidad, filtrado de contenido así como de paquetes, protección de puertos, efectividad y viabilidad.

Se aprovechó la innovación tecnológica, en este caso la virtualización, para generar las condiciones y el ambiente requerido para lograr la caracterización del firewall, consiguiendo integrar con ayuda del sistema operativo OpenBSD herramientas como Packet Filter, DansGuardian así como el uso del protocolo CARP para generar alta disponibilidad. Tales herramientas se eligieron por su compatibilidad y sus características únicas para configurar un firewall.

Para constatar el nivel de seguridad del firewall se realizaron diversas pruebas de penetración para conocer el desempeño del mismo. De igual forma se realizaron estas pruebas a un firewall comercial para comprobar la efectividad del firewall virtualizado.
Capítulo 1. Planteamiento del problema

1.1 Antecedentes

Durante años la humanidad se ha visto forzada a protegerse y esto no ha sido solamente en el ámbito físico como persona sino también en lo referente a su propiedad, es decir, el ser humano se preocupa por asegurar sus bienes. Así es como estas actitudes de defensa dieron inicio a lo que hoy en día llamamos seguridad.

Garantizar la seguridad es una tarea que poco a poco ha sido la misión de todas las organizaciones sin excepción alguna. La seguridad que puede ser otorgada puede ser física, lógica, territorial, social e informática, entre muchas otras más.

Uno de los bienes que la humanidad ha visto la necesidad de asegurar han sido los activos los cuales son información, para las organizaciones e instituciones son más importantes incluso que el dinero mismo. Con el fin de asegurar detectaron la necesidad de defender y tomar ciertas medidas que garantizaran la seguridad de lo que ellos consideraban importante.

La seguridad en cómputo es un área crítica en las redes de datos, se encarga de asegurar la integridad y privacidad de la información de un sistema informático y sus usuarios. Normalmente se deben establecer controles de seguridad física y lógica para mitigar amenazas y/o ataques a los activos de las organizaciones, para así crear barreras de seguridad que no son más que técnicas, aplicaciones y dispositivos de seguridad que utilizadas para la protección de la información otorgan muchos beneficios para el usuario y para la organización.

Algunos controles de seguridad incluyen sistemas antivirus, sistemas de prevención contra intrusos, filtros antispam y sistemas corta fuegos o mejor conocidos como firewall. Estos controles son la base que es necesaria para cada organización, la cual desea tener una protección para sus usuarios y para sí misma. Al realizar la integración de estos sistemas o controles se debe de tener en cuenta un análisis de riesgos, este análisis es una parte fundamental ya que ayuda a determinar el tipo de control que es requerido y la inversión que está dispuesta a realizar.
Los firewall son dispositivos de hardware o software para controlar comunicaciones, permitiéndoles o prohibiéndoles accesos según haya sido especificado. SueLEN ubicarse entre el punto de conexión de red interna con la red exterior (Internet) o en la red interna, de este modo la red es protegida de accesos no autorizados, así como de la intrusión de paquetes que puedan afectar o tener algún tipo de información no deseada. Todo lo que se refiere a tráfico pasará a través del firewall, así también el firewall ayudará a reforzar las políticas de seguridad que tenga la organización.

La tecnología de los firewall surgió a finales de 1980, cuando Internet era una tecnología bastante nueva en cuanto a su uso global y la conectividad. Los predecesores de los firewall fueron los routers, quienes mantenían a las redes separadas unas de otras. En caso de que llegara a surgir algún problema en un segmento de la red solo se mantenía en esa sección, no se corría el riesgo de que se difundiera a través de toda la red.

La virtualización es una de las últimas tendencias en la industria, ya que las organizaciones tienen por objeto aumentar la utilidad, la flexibilidad y la rentabilidad de sus recursos informáticos. VMware, Citrix, Microsoft, IBM, Red Hat y muchos otros proveedores ofrecen soluciones de virtualización.

La virtualización se introdujo por primera vez en la década de 1960 por IBM para impulsar la utilización de grandes sistemas (mainframe caros) dividiéndolos en máquinas virtuales separadas lógicas que podían ejecutar múltiples aplicaciones y procesos al mismo tiempo. En los años 1980 y 1990, este modelo de mainframe centralizado y compartido dio paso a un modelo distribuido (la computación cliente-servidor) en el cual muchos servidores independientes x86 de bajo costo eran capaces de ejecutar aplicaciones específicas.

Además de utilizar la tecnología de virtualización para dividir una máquina en varias máquinas virtuales, también puede utilizar las soluciones de virtualización para combinar múltiples recursos físicos en un recurso virtual.

Al integrar un firewall en un ambiente virtualizado se obtienen beneficios de escalabilidad, disponibilidad y aprovechamiento máximo de recursos. Lo que conlleva a lograr una tendencia para servicios integrados en una sola caja o centros de datos auto-contenidos.
A través de los años la tecnología ha ido integrando los servicios en un ambiente virtual, dichos servicios incluyen aquellos con infraestructura de red como switch, router y firewall, aunque estos últimos no se tienen en una modalidad amplia y a bajo costo.

1.2 Definición del problema

Existe una necesidad de tener opciones efectivas virtualizadas para reforzar la seguridad al realizar conexiones dentro de la red interna de una institución pública hacia Internet.

Para este tipo de instituciones en donde se tienen miles de usuarios concurrentes accediendo a Internet, se requiere una solución muy robusta cuyo gasto de inversión y gasto operativo puede ser muy alto, y donde inversiones en tecnologías ya realizados (hardware), se pueden aprovechar en ambientes virtualizados.

1.3 Objetivos de la investigación

Objetivo General

Integrar un prototipo de firewall con alta disponibilidad y filtro de contenido en un ambiente virtualizado que integre tecnología de seguridad perimetral a través de software libre con el fin de determinar la efectividad para bloquear firmas de ataques y filtrar contenido al nivel que lo hace un equipo de seguridad comercial.

Objetivos específicos

- Identificar los posibles factores que se presentan en las instituciones públicas, en cuanto a problemas con seguridad y red, para así realizar un prototipo que contraataque las complejidades habitualmente presentadas; de tal manera que sea un prototipo que se adecue a las necesidades de los usuarios.

- Estimar las herramientas a utilizar y se realizará un análisis para hacer uso de las herramientas más factibles en cuanto a viabilidad, compatibilidad y economía, esto último para que logre ser accesible para las instituciones públicas, tanto en costo material como operativo.
• Desarrollar un prototipo el cual sea flexible, con alta disponibilidad y efectivo contra posibles ataques, así como soporte de tráfico durante altas concurrencias.

1.4 Preguntas de investigación

¿Qué características debe contemplar la solución propuesta a fin de que ofrezca la seguridad requerida en un ambiente virtualizado?

¿Qué consideraciones de infraestructura se deben contemplar para un servicio de firewall en un ambiente virtualizado?

¿Cuáles son las reglas mínimas que debe tener un firewall en un ambiente educativo?

¿Qué tráfico se prioriza en el firewall contemplando aplicaciones que comúnmente se usan en escuelas?

¿Cómo se integra OpenBSD con Packet Filter y DansGuardian en alta disponibilidad?

¿Qué pruebas de penetración se van a realizar al firewall para probar su efectividad en el ambiente virtual?

¿Cómo se caracteriza el tráfico en una red de datos?

¿Cómo se mide el desempeño real de un firewall virtualizado contra uno no virtualizado?

1.5 Justificación de la investigación

Últimamente se ha venido aprovechando la consolidación de infraestructura en ambientes virtualizados, y el hacer uso de esta misma para integrar un servicio de seguridad puede representar un ahorro significativo para alguna organización, sin la necesidad de elegir entre calidad, eficiencia, costo, seguridad e integridad de la información.

Cada servicio requerido para aumentar la seguridad y la alta disponibilidad en la red que alguna institución ofrece suele tener un costo excesivo, sin mencionar que la complejidad y
a su vez, los costos, aumentan al tener estos controles de forma individual. Por esta razón se pretende utilizar varias herramientas que al integrarse tendrán el propósito de brindar un servicio eficiente y con menor costo.

Debido a los miles de usuarios que a diario hacen uso del servicio, es necesario contar con un control de contenido a nivel red. Esto puede aplicar principalmente en escuelas de nivel básico y medio superior, que se pretenden conectar a través del programa México Conectado, del gobierno federal.

Los beneficios que se obtendrán a partir del prototipo serán: firewall de estados, control de contenido, control de tráfico de datos, control de accesos no autorizados y garantizar una alta disponibilidad en el servicio utilizando recursos de virtualización.

1.6 Limitaciones y delimitaciones de la investigación

Limitaciones

Una limitante que puede presentarse durante el desarrollo del prototipo es que las herramientas que se utilizaran como las aplicaciones ya existentes no sean compatibles o que se necesiten licencias para su integración.

También pueden presentarse problemas con el proveedor virtual que será utilizado.

En caso de ser necesario se replantearán las herramientas que se utilizaran para el desarrollo del firewall.

Delimitaciones

El proyecto será un prototipo de firewall virtualizado, no se implementará.

Se realizarán pruebas contra un firewall comercial para comprobar su eficiencia y seguridad, así como su alta disponibilidad.

El ambiente en el que se desarrollara el firewall será virtual, se utilizaran maquinas virtuales para su realización.

Las herramientas que se utilizaran serán de licenciamiento libre.
Capítulo 2. Marco Teórico

2.1 Seguridad informática

Es la disciplina que se ocupa de diseñar normas, procedimientos, métodos y técnicas destinados a conseguir un sistema de información seguro y confiable. No obstante, las medidas de seguridad que se le aplican no dejan de tener siempre un margen de riesgo.

Para afrontar el establecimiento de un sistema de seguridad es necesario conocer:

- Cuáles son los elementos que componen un sistema. Está información se obtiene mediante entrevistas con los responsables o directivos de la organización para la que se hace el estudio de riesgos y mediante apreciación directa.
- Cuáles son los peligros que afectan al sistema, accidentalmente o provocados. Se deducen tanto de los datos aportados por la organización como por el estudio directo del sistema mediante la realización de pruebas y muestreos sobre el mismo.
- Cuáles son las medidas que deberían adoptarse para conocer, prevenir, impedir, reducir o controlar los riesgos potenciales. Se trata de decidir cuáles serán los servicios y mecanismos de seguridad que reducirían los riesgos al máximo posible.

Protección de activos

Son los recursos que pertenecen al propio sistema de información o que están relacionados con este. La presencia de los activos facilita el funcionamiento de la empresa u organización y la consecución de sus objetivos. Al hacer un estudio de los activos existentes hay que tener en cuenta la relación que guardan entre ellos y la influencia que se ejercen: cómo afectaría en uno de ellos un daño ocurrido a otro.

Amenazas

La presencia de uno o más factores de diversa índole (personas, máquinas o procesos) que –de tener la oportunidad- atacaran al sistema produciéndole daños aprovechándose de su nivel de vulnerabilidad. Hay diferentes tipos de amenazas de las que hay que proteger el sistema, desde las físicas como corta corrientes, fallos del hardware o riesgos ambientales
hasta los errores intencionales o no de los usuarios, la entrada de software malicioso (virus, troyanos, gusanos) o el robo, destrucción o modificación de la información.

**Vulnerabilidades**

Probabilidades que existen de que una amenaza se materialice contra un activo. No todos los activos son vulnerables a las mismas amenazas. Por tal razón es indispensable el realizar un análisis de riesgos y así tener en cuenta la vulnerabilidad de cada activo.

**Ataques**

Se dice que es producido un ataque accidental o deliberado contra el sistema cuando se ha materializado una amenaza.

### 2.1.1 Tipos de seguridad

**Seguridad lógica**

Los mecanismos y herramientas de seguridad lógica tienen como objetivo proteger digitalmente la información de manera directa.

- Control de accesos. Mediante nombres de usuarios y contraseñas.
- Cifrado de datos (encriptación). Los datos se enmascaran con una clave específica creada mediante un algoritmo de encriptación.
- Antivirus. Detectan e impiden la entrada de virus y otro software malicioso., en caso de infección es capaz de eliminarlos y corregir los daños ocasionados al sistema.
- Cortafuegos (*firewall*). Se trata de uno o más dispositivos de software, de hardware o mixtos que permiten, deniegan o restringen el acceso al sistema. Protege la integridad de la información.

**Seguridad Física**

Son tareas y mecanismos físicos cuyo objetivo es proteger al sistema (y, por lo tanto indirectamente la información) de peligros físicos y lógicos.
- Respaldo de datos. guardar copias de seguridad de la información del sistema en lugar seguro. Disponibilidad.
- Dispositivos físicos de protección. Pararrayos, detectores de humo, extintores, *firewall* por hardware, alarmas, sistemas de alimentación interrumpida y otros.[1]

### 2.1.2 Seguridad en redes de comunicación

Las razones por las que debemos de asegurar computadoras y redes contra ataques son muy similares a las razones por las que se debe de asegurar una persona o propiedad en la vida real. Los probables sospechosos, los problemas que ellos causan y los mecanismos de protección que se usan para defender la integridad de una persona son frecuentemente iguales, en este caso sin importar que estemos lidiando con unos y ceros. En un mundo ideal no habría necesidad de cercado, puertas o candados, porque el lado bueno de la naturaleza humana y las leyes de la sociedad serían suficientes para proteger a las personas, su privacidad y su propiedad. Desafortunadamente existe un mundo así y las personas rompen leyes, roban, traspasan propiedades ajenas e invaden la privacidad, razón por la que existe la necesidad de proteger.

Las cosas no son muy diferentes en el mundo de las redes, así como en el mundo real, Internet da a las personas con intenciones maliciosas muchas oportunidades para realizar las actividades que desean. Ellos pueden espiar las comunicaciones, irrumpir dentro de computadores y redes, bloquear conexiones entre maquinas, destruir datos, falsificar archivos y destruir un sistema entero.

La actual tecnología en redes da a los atacantes muchas maneras de amplificar el poder de sus acciones usando equipos con herramientas para quebrantar redes e ingresar a cientos de hosts; una sola persona puede potencialmente causar daños a una escala considerable.
Algunos prefieren irrumpir en una red y escuchar comunicaciones, copiar archivos confidenciales o cambiar registros esenciales.

La mejor forma de pelear contra los atacantes es la prevención. Para evitar problemas y mantener a las personas maliciosas lejos, muchas organizaciones invierten grandes cantidades de dinero en software de seguridad, hardware y auditorias [2]

2.2 Firewall

Un firewall es una parte de un sistema o una red que está diseñada para bloquear el acceso no autorizado, permitiendo al mismo tiempo comunicaciones autorizadas. Se trata de un dispositivo o conjunto de dispositivos configurados para permitir, limitar, cifrar, descifrar, el tráfico entre los diferentes ámbitos sobre la base de un conjunto de normas y otros criterios.

Pueden ser implementados en hardware o software, o una combinación de ambos. Los firewall se utilizan con frecuencia para evitar que los usuarios de Internet no autorizados tengan acceso a redes privadas conectadas a Internet, especialmente intranets. Todos los mensajes que entren o salgan de la intranet pasan a través del firewall, que examina cada mensaje y bloquea aquellos que no cumplen los criterios de seguridad especificados. [3]

Las principales tareas de un firewall son:

- Control de acceso basado en la dirección del emisor, del receptor o de los servicios direccionados (por ejemplo, protocolo de nivel de aplicación).
- Control de comportamiento, por ejemplo comprobar si existen virus en los ficheros entrantes.
- Control del usuario, por ejemplo autenticación basada en el origen del tráfico.
- Ocultar la red interna, por ejemplo su topología, direcciones, etc.
- Registrar en un log el tráfico que pasa.
• Hacer cumplir una política de seguridad, por ejemplo de control de acceso.

2.2.1 Ventajas de un firewall

Protege de intrusiones.
El acceso a ciertos segmentos de la red de una organización sólo se permite desde máquinas autorizadas de otros segmentos de la organización o de Internet.

Protección de información privada.
Permite definir distintos niveles de acceso a la información, de manera que en una organización cada grupo de usuarios definido tendrá acceso sólo a los servicios y la información que le son estrictamente necesarios.

Optimización de acceso.
Identifica los elementos de la red internos y optimiza que la comunicación entre ellos sea más directa. Esto ayuda a reconfigurar los parámetros de seguridad. [4]

2.2.2 Limitaciones de un firewall

Un firewall no puede proteger contra aquellos ataques cuyo tráfico no pase a través de él. Ni puede proteger de las amenazas a las que está sometido por ataques internos o usuarios negligentes. El firewall no puede prohibir a espías corporativos copiar datos sensibles en medios físicos de almacenamiento (cd, memorias, etc.) y sustraerlas del edificio, y mucho menos puede proteger contra los ataques de ingeniería social.

El firewall no puede proteger contra los ataques posibles a la red interna por virus informáticos a través de archivos y software. La solución real está en que la organización
debe ser consciente en instalar software antivirus en cada máquina para protegerse de los virus que llegan por cualquier medio de almacenamiento u otra fuente. No protege de los fallos de seguridad de los servicios y protocolos cuyo tráfico esté permitido. Hay que configurar correctamente y cuidar la seguridad de los servicios que se publiquen a Internet.

2.2.3 Políticas de un firewall

Hay dos políticas básicas en la configuración de un firewall que cambian radicalmente la filosofía fundamental de la seguridad en la organización:

Política restrictiva: Se deniega todo el tráfico excepto el que está explícitamente permitido. El firewall obstruye todo el tráfico y hay que habilitar expresamente el tráfico de los servicios que se necesiten.

Política permisiva: Se permite todo el tráfico excepto el que esté explícitamente denegado. Cada servicio potencialmente peligroso necesitará ser aislado básicamente caso por caso, mientras que el resto del tráfico no será filtrado.

La política restrictiva es la más segura, ya que es más difícil permitir por error tráfico potencialmente peligroso, mientras que en la política permisiva es posible que no se haya contemplado algún caso de tráfico peligroso y sea permitido por omisión.

2.2.4 Tipos de firewall

Firewall de capa de red o de filtrado de paquetes
Las reglas acerca del filtrado de paquetes a través de un router para rehusar/permitir el tráfico, está basado en un servicio en específico desde entonces muchos servicios vierten su información en numerosos puertos TCP/UDP conocidos.

Por ejemplo, un servidor Telnet está a la espera para conexiones remotas en el puerto 23 TCP y un servidor SMTP espera las conexiones de entrada en el puerto 25 TCP. Para bloquear todas las entradas de conexión Telnet, el router simplemente descarta todos los paquetes que contengan el valor del puerto destino TCP igual a 23. Para restringir las conexiones Telnet a un limitado número de servidores internos, el router podrá rehusar el paso a todos aquellos paquetes que contengan el puerto destino TCP igual a 23 y que no contengan la dirección destino IP de uno de los servidores permitidos.

Algunas características típicas de filtrado que un administrador de redes podría solicitar en un router filtra-paquetes para perfeccionar su funcionamiento serían:

- Permitir la entrada de sesiones Telnet únicamente a una lista específica de servidores internos.
- Permitir la entrada de sesiones FTP únicamente a los servidores internos especificados.
- Permitir todas las salidas para sesiones Telnet.
- Permitir todas las salidas para sesiones FTP.
- Rehusar todo el tráfico UDP.

_Firewall de capa de aplicación_

Trabaja en la capa de aplicación (nivel 7 OSI) de manera que los filtrados se pueden adaptar a características propias de los protocolos de este nivel. Por ejemplo, si se trata de tráfico HTTP, se pueden realizar filtrados según la URL a la que se está intentando acceder.

Un firewall a nivel 7 de tráfico HTTP suele denominarse proxy, y permite que los host de una organización entren a internet de una forma controlada.
Un firewall personal es un caso particular que se instala como software en un host, filtrando las comunicaciones entre dicho host y el resto de la red y viceversa. [5]

**Firewall Híbridos**

Combinan los tipos de firewall anteriores y los implementan en serie en vez de en paralelo. Si se conectan en serie, se mejora la seguridad total. Si se conectan en paralelo, entonces el perímetro de seguridad de red sólo ser tan seguro como el menos seguro de los métodos utilizados. En entornos de medio a elevado riesgo un firewall híbrido puede ser la elección ideal.

**Firewall para intranets**

Aunque los firewall normalmente se colocan entre una red corporativa y la red no segura del exterior (ó Internet), en grandes organizaciones, los firewall se utilizan a menudo para crear subredes diferentes dentro de la red interna (denominada también Intranet).

Los "firewall para Intranets" se utilizan para aislarse una subred particular de la red corporativa total. La razón del aislamiento de un segmento de red puede ser que ciertos empleados sólo pueden acceder a subredes guardadas por estos firewall sólo en base a una necesidad concreta. Un ejemplo puede ser un firewall para el departamento de nóminas ó contabilidad de una organización. La decisión de utilizar un firewall Intranet se basa generalmente en la necesidad de hacer cierta información disponible para algunos pero no para todos los usuarios internos ó para proporcionar un alto grado de responsabilidad para el acceso y utilización de información sensible ó confidencial.

Para cualquier sistema que guarde aplicaciones críticas de la organización ó que proporcione acceso a información sensible ó confidencial, deberían utilizarse firewall internos ó router de filtrado de paquetes para proporcionar control de acceso fuerte y soportar auditoría y registro. Estos controles deberían utilizarse para dividir la red corporativa interna a la hora de soportar políticas de acceso desarrolladas por los propietarios de información designados.
Firewall con capacidad VPN

Las redes privadas virtuales ó VPN (Virtual Private Networks) permiten a las redes seguras comunicarse con otras redes seguras utilizando redes no seguras como Internet. Puesto que algunos firewall proporcionan la "capacidad VPN", es necesario definir una política de seguridad para establecer VPN. Cualquier conexión entre firewall sobre redes públicas utilizan VPN cifradas para asegurar la privacidad e integridad de los datos que se pasan a través de la red pública. Todas las conexiones VPN deben ser aprobadas y gestionadas por el administrador de servicios de red. Deben establecerse los medios apropiados para distribuir y mantener claves de cifrado antes del uso operacional de los VPNs.

Firewall de inspección de estados

Es un método de filtrado de paquetes que trabaja a nivel de flujo o conexión, con ocasionales intervenciones a nivel de la aplicación. Este controla las configuraciones de cada paquete aprobado y luego pasa o bloquea el tráfico basado en dichas características. Mantienen una tabla de estado que hace seguimiento de las sesiones que atraviesan el firewall [6] y en función de ella hace inspección de cada paquete que atraviesa el dispositivo.

El mecanismo asume que si se permite el inicio de la conexión, cualquier conexión adicional que requiera esa aplicación será permitida.

Es un mecanismo confiable para filtrar tráfico de red entre dominios de seguridad.

2.3 Virtualización

En términos generales, la virtualización es un proceso y resultado al mismo tiempo del tratamiento y de la comunicación mediante computadora de datos, información y conocimientos. Más específicamente, la virtualización consiste en representar electrónicamente y en forma numérica digital, objetos y procesos que encontramos en el mundo real. En el contexto de la educación superior, la virtualización puede comprender la
representación de procesos y objetos asociados a actividades de enseñanza y aprendizaje, de investigación y gestión, así como objetos cuya manipulación permite al usuario, realizar diversas operaciones a través de Internet, tales como aprender mediante la interacción con cursos electrónicos, inscribirse en un curso, consultar documentos en una biblioteca electrónica, comunicarse con estudiantes y profesores y otros (Quéau, 1993).

Las posibilidades de la virtualización no sólo se limitan a la creación de máquinas aisladas; mediante la adecuada interconexión de dichas máquinas por medio de redes virtuales es posible crear escenarios de red completos. Asimismo, estos escenarios virtuales pueden conectarse con equipos externos, abriendo la posibilidad a la creación de escenarios mixtos en los que sistemas reales y virtuales interaccionen. [7]

La virtualización es una capa abstracta que desacopla el hardware físico del sistema operativo para brindar mayor flexibilidad y utilización de los recursos de las tecnologías de la información. Al separar la operación lógica del hardware físico, un entorno virtualizado proporciona mayor flexibilidad operativa y agiliza los cambios del sistema, ofreciendo una plataforma que refuerza la continuidad del negocio y escala con rapidez para satisfacer las demandas empresariales.

De esta forma, la virtualización permite que múltiples máquinas virtuales con sistemas operativos heterogéneos puedan ejecutarse sobre la misma máquina física. Cada máquina virtual tiene su propio hardware virtual, a través del cual opera el sistema operativo y aplicaciones. Permite implementar recursos informáticos aislando unas capas del sistema de otras: hardware, sistema operativo, aplicaciones, datos, redes, entre otras.

Se pueden obtener las siguientes características de la virtualización.

- **Particionamiento**: Ejecuta múltiples máquinas virtuales en un mismo host.
- **Aislamiento**: Cada máquina virtual está aislada del resto de máquinas virtuales en el mismo host.
- **Encapsulación**: Encapsulan el sistema (configuración de hardware, sistema operativo y aplicaciones) en ficheros.
• Independencia del hardware: Una máquina virtual puede funcionar en cualquier servidor, sin modificación.

Actualmente, existen múltiples sistemas de virtualización (Xen, VMware, KVM, User Mode Linux, etc.) Además, hoy en día la virtualización no sólo se circunscribe a la ejecución de sistemas operativos de propósito general como GNU/Linux o Windows, sino que permite también la emulación (limitada) de equipos de comunicaciones como routers CISCO mediante el software Dynamips/Dynagen.

2.3.1 Recursos

2.3.1.1 Hipervisores

El hipervisor, también llamado monitor de máquina virtual (VMM), es el núcleo central de algunas de las tecnologías de virtualización de hardware más populares y eficaces, entre las cuales se encuentran: VMware ESXi Free, VMware ESX, Xen, Citrix XenServer, Microsoft Hyper-V Server, Oracle VM.

Los hipervisores son aplicaciones que presentan a los sistemas operativos virtualizados (sistemas invitados) una plataforma operativa virtual (hardware virtual), a la vez que ocultan a dicho sistema operativo virtualizado las características físicas reales del equipo sobre el que operan. También son los encargados de monitorizar la ejecución de los sistemas operativos invitados.

Con el uso de hipervisores es posible conseguir que múltiples sistemas operativos compitan por el acceso simultáneo a los recursos hardware de una máquina virtual de manera eficaz y sin conflictos.

Existen tres tipos principales de hipervisores en el mercado:

*Hipervisores de tipo 1*
También llamados nativos, *unhosted* o *bare-metal*, en ellos el hipervisor se ejecuta directamente sobre el hardware físico; el hipervisor se carga antes que ninguno de los sistemas operativos invitados, y todos los accesos directos a hardware son controlados por él.

Aunque esta es la aproximación clásica y más antigua de la virtualización por hardware, actualmente las soluciones más potentes de la mayoría de fabricantes usa este enfoque. Es el caso de Microsoft Hyper-V, Citrix XEN Server y VMWare ESX-Server.

*Hipervisores de tipo 2*

También llamados *hosted*, en ellos el hipervisor se ejecuta en el contexto de un sistema operativo completo, que se carga antes que el hipervisor. Las máquinas virtuales se ejecutan en un tercer nivel, por encima del hipervisor.

Son típicos de escenarios de virtualización orientada a la ejecución multiplataforma de software, como en el caso de CLR de .NET o de las máquinas virtuales de Java.

*Hipervisores híbridos*

En este modelo tanto el sistema operativo anfitrión como el hipervisor interactúan directamente con el hardware físico.

Las máquinas virtuales se ejecutan en un tercer nivel con respecto al hardware, por encima del hipervisor, pero también interactúan directamente con el sistema operativo anfitrión.

Las características más importantes del VMWare son sin lugar a dudas los 3 grandes grupos y de cada una de ellas se desprenden como realizar cada una de las características importantes [8]:

- Optimizar el desarrollo y las pruebas de software
- Crear múltiples entornos de desarrollo y pruebas en un único sistema
- Aceleración de los ciclos de desarrollo y disminución del tiempo de salida al mercado.
- Creará aplicaciones de misión crítica basadas en Windows y/o Linux
• Disminución de los costos de hardware entre un 50 y 60%.
• Archivar entornos de prueba en *File Server* (Servidores de Archivos) y restaurarlos rápidamente, según sea necesario.
• Disminución del costo o tiempo de configuración entre un 25 y 55% dejando tiempo para realizar las importantes tareas de desarrollo y prueba.
• Probar nuevas actualizaciones de aplicaciones, correcciones y *service packs* de sistemas operativos en un solo computador.
• Mejora de la calidad de los proyectos mediante pruebas más rigurosas.
• Eliminación de los costosos problemas de implementación y mantenimiento.

La tabla que se encuentra en el Anexo I compara VMware ESX / ESXi 3.5 con Microsoft Hyper-V y el hipervisor muestra cómo la tecnología Hyper-V se presentan importantes limitaciones en todos los aspectos críticos de un hipervisor [9].

2.3.1.2 Máquinas virtuales

Una máquina virtual es un software que emula a un ordenador y puede ejecutar programas como si fuese una computadora real. Este software también es definido como un duplicado eficiente y aislado de una máquina física.

**Características de las máquinas virtuales**

• Aislamiento: las máquinas virtuales son totalmente independientes, entre sí y el hipervisor. Por tanto un fallo en una aplicación o en una máquina virtual afectará únicamente a esa máquina virtual. El resto de máquinas virtuales y el hipervisor seguirán funcionando normalmente.
• Seguridad: cada máquina tiene un acceso privilegiado (root o administrador) independiente. Por tanto, un ataque de seguridad en una máquina virtual sólo afectará a esa máquina.
• Flexibilidad: podemos crear las máquinas virtuales con las características de CPU, memoria, disco y red que necesitemos, sin necesidad de “comprar” un ordenador
con esas características. También podemos tener máquinas virtuales con distintos sistemas operativos, ejecutándose dentro de una misma máquina física.

- **Agilidad**: la creación de una máquina virtual es un proceso muy rápido, básicamente la ejecución de un comando. Por tanto, si necesitamos un nuevo servidor lo podremos tener casi al instante, sin pasar por el proceso de compra, configuración, etc.

- **Portabilidad**: toda la configuración de una máquina virtual reside en uno o varios ficheros. Esto hace que sea muy fácil clonar o transportar la máquina virtual a otro servidor físico, simplemente copiando y moviendo dichos ficheros que encapsulan la máquina virtual.

### 2.4 Red de campus institucionales

Los servicios ofrecidos a través de Internet son de distinta naturaleza y cada uno de ellos tiene sus propios requerimientos de seguridad. Por ejemplo la seguridad requerida para un sistema de control escolar es diferente a la seguridad requerida por una aplicación de videoconferencia. [10]

Internet es un sistema sumamente complejo debido a la gran cantidad de conceptos y tecnologías involucradas, y a la presencia de los imponderables errores humanos que suelen existir en el hardware y sobre todo en el software de las computadoras.

El uso creciente de Internet para actividades que van más allá de la comunicación personal y el entretenimiento han generado una creciente demanda de seguridad en la red. El comercio electrónico en sus diferentes modalidades, las transacciones bancarias, las certificaciones de componentes descargables de la red, los servicios de extranet, las redes privadas virtuales y los portales Web en general, son algunas de las aplicaciones de Internet que requieren mayores niveles de seguridad para garantizar su correcto funcionamiento.
La seguridad en redes involucra diferentes aspectos tales como:

- **Privacidad**: protección de la información contra accesos no autorizados.
- **Integridad**: prevención de la información en contra de modificaciones no autorizadas.
- **No rechazo de servicio**: evitar que a un usuario legítimo le sea denegado el acceso a un servicio disponible.
- **No repudio de autoridad**: evitar que un usuario niegue ser el autor de la información que él mismo generó.
- **Disponibilidad de los servicios**: evitar que un servicio sea inhabilitado ilegalmente.

Según sea la necesidad de cada uno de estos aspectos se determinan las características de la seguridad que es requerida por la institución, y se ve la manera de darle más importancia a ese aspecto.

A través de los años, la comunidad de Internet ha desarrollado diversos mecanismos de seguridad para aplicaciones específicas incluyendo correo electrónico, aplicaciones cliente servidor y acceso web, basados en tres elementos fundamentales: encriptación, autenticación e integridad.

Algunos mecanismos de seguridad serían: SSL, IPsec, algoritmos criptográficos y firewalls, entre muchos otros mecanismos de seguridad.

**Los firewalls** deben cumplir con las siguientes características generales:

- Todo el tráfico de adentro hacia fuera y viceversa, debe pasar a través del firewall.
- Solamente el tráfico que sea autorizado, según la política que se haya definido, podrá pasar a través del firewall.
- El firewall en sí mismo debe ser inmune a cualquier tipo de penetración. [11]

Garantizar la seguridad de la información y de los servicios disponibles a través de Internet es una tarea sumamente complicada pero necesaria para soportar la evolución de los servicios existentes y la aparición de nuevos servicios.
2.4.1 Concurrencia de usuarios

Las universidades y otras instituciones de educación superior, especialmente las de países en vías de desarrollo, están enfrentando la problemática de servir a una población cada vez mayor de estudiantes, más diversificada social y culturalmente, en un nuevo ambiente social, más dinámico y turbulento.

La otra problemática consiste en proveer a la población trabajadora de una educación permanente, como respuesta a los requerimientos dinámicos de una sociedad del conocimiento en proceso de gestación, en la cual predomina un alto índice de innovación y la transitoriedad y diversidad de relaciones entre los seres humanos, objetos, lugares y sobre todo, conocimientos (Toffler, 1970, 1980 y 1990).

2.4.2 Alta disponibilidad

La capacidad de trabajar continuamente durante un periodo de tiempo dado es a lo que se le llama alta disponibilidad, este dota a los equipos de redundancia ante fallos, es decir que los servicios que son para a comunidad de usuario se encuentren disponibles en todo momento así como la información consolidada en caso de pérdida de datos o problemas de servicios, en caso de que alguna de estas situaciones ocurra otra de las máquinas tomará el rol de la principal, de esta manera se tendrá un servicio continuo de datos sin afectar a la organización o institución. Además el cluster puede configurarse en modo activo-activo haciendo balanceo de carga del tráfico o en modo activo/pasivo en la que un único equipo procesa el tráfico de la red y es monitorizado por los demás para sustituirle en caso de caída [12].

La alta disponibilidad se divide en 2 tipos:

- Alta disponibilidad de hardware
La redundancia de hardware se da cuando se produce un fallo de hardware en alguna de las máquinas del clúster debido a alguna falla en fuente de poder, punto de red, cable o tarjeta de red, controladora de discos, switch, etc., así mismo si se tiene la configuración computacional adecuada en el equipo, se podrá cambiar el hardware rápidamente (HotSwap) sin tener que bajar los servicios que se tienen.

- Alta disponibilidad de aplicaciones

El software de alta disponibilidad es capaz de re arrancar automáticamente los servicios que han fallado en la máquina original en cualquier otra máquina del clúster. Y cuando la máquina que ha fallado se recupera, los servicios son nuevamente migrados a la máquina original. Esta capacidad de recuperación automática de servicios nos garantiza la integridad de la información, ya que no hay pérdida de datos. Evitando molestias a los usuarios.

Además, la Alta Disponibilidad nos permite poder tener nuestros sistemas críticos funcionando un 99,9% del tiempo, algo que por lo general es un gran desafío para los departamentos tecnológicos [12]. Y ya que si ponemos en la balanza poder tener un sistema crítico un uptime del 99,9 % V/S los costos de tener quedar con los sistemas detenidos podríamos tener clara la importancia de poder disponer de estas tecnologías que nos permiten un alto desempeño de nuestra información.

Principios de la ingeniería de alta disponibilidad:

- Elimina puntos de fallo. Es decir, gracias a la redundancia en el sistema en caso de que fallara un componente del sistema no significa el fallo total del sistema.
- Reliable crossover. En los sistemas multiproceso el punto de cruce se convierte en el único punto de fallo por lo cual se debe prever cruce fiable.
- Detección de fallos que se producen. Ya teniendo en claro los dos anteriores esto significa que el usuario no puede ver una falla en el sistema. El mantenimiento del mismo es necesaria. [13]

Ventajas de la Alta Disponibilidad
• Equipos pueden ser configurados en *cluster*, proporcionando escenarios de alta disponibilidad mediante la utilización de varios equipos redundantes entre sí, empleando un protocolo específico para la sincronización del *cluster*.
• El *cluster* puede estar formado hasta por 32 equipos.
• La funcionalidad de Alta Disponibilidad está soportada por varias las plataformas.
• Cada miembro del *cluster* debe ser del mismo modelo hardware así como tener instalada la misma versión del Sistema Operativo.
• La funcionalidad de Alta Disponibilidad está soportada tanto en modo *router* como en modo transparente.

En la tabla 2 se muestra el tiempo de inactividad que se permitirá por un porcentaje determinado de disponibilidad, esto dependiendo que tan necesario es que el sistema funcione de forma continua.

<table>
<thead>
<tr>
<th>Disponibilidad %</th>
<th>Inactividad por año</th>
<th>Inactividad por mes</th>
<th>Inactividad por semana</th>
</tr>
</thead>
<tbody>
<tr>
<td>90% (&quot;un nueve&quot;)</td>
<td>36.5 días</td>
<td>72 horas</td>
<td>16.8 horas</td>
</tr>
<tr>
<td>95%</td>
<td>18.25 días</td>
<td>36 horas</td>
<td>8.4 horas</td>
</tr>
<tr>
<td>97%</td>
<td>10.96 días</td>
<td>21.6 horas</td>
<td>5.04 horas</td>
</tr>
<tr>
<td>98%</td>
<td>7.30 días</td>
<td>14.4 horas</td>
<td>3.36 horas</td>
</tr>
<tr>
<td>99% (&quot;dos nueves&quot;)</td>
<td>3.65 días</td>
<td>7.20 horas</td>
<td>1.68 horas</td>
</tr>
<tr>
<td>99.5%</td>
<td>1.83 días</td>
<td>3.60 horas</td>
<td>50.4 minutos</td>
</tr>
</tbody>
</table>
99.8% 17.52 horas 86.23 minutos 20.16 minutos
99.9% ("tres nueves") 8.76 horas 43.8 minutos 10.1 minutos
99.95% 4.38 horas 21.56 minutos 5.04 minutos
99.99% ("cuatro nueves") 52.56 minutos 4.32 minutos 1.01 minutos
99.995% 26.28 minutos 2.16 minutos 30.24 segundos
99.999% ("cinco nueves") 5.26 minutos 25.9 segundos 6.05 segundos
99.9999% ("seis nueves") 31.5 segundos 2.59 segundos 604.8 milisegundos
99.99999% ("siete nueves") 3.15 segundos 262.97 milisegundos 60.48 milisegundos
99.999999% ("ocho nueves") 315.569 milisegundos 26.297 milisegundos 6.048 milisegundos
99.9999999% ("nueve nueves") 31.5569 milisegundos 2.6297 milisegundos 0.6048 milisegundos

TABLA 1. Porcentajes de inactividad en Alta disponibilidad.

2.4.3 Packet Filter

Un Filtro de Contenido (Packet Filter, PF) es un sistema de filtrado para OpenBSD el cual soporta TCP/IP y permite hacer NAT (Network Address Transaltios) [14]. PF además es capaz de normalizar, acondicionar tráfico y proveer un control de ancho de banda para las
comunicaciones. Es software libre, viene integrado en el Core del sistema operativo OpenBSD desde la versión 3.0.

Los filtros de contenido se utilizan para garantizar que el acceso a los recursos informáticos este controlado, así como observar el tráfico de red, clasificarlo y tomar acciones basadas en ciertas reglas o políticas definidas. Este valida la información que pasa a través de sí mismo verificando el contenido de los paquetes o sus encabezados, teniendo en cuenta las políticas que fueron definidas por un agente controlador.

Los criterios que se pueden utilizar para filtrar son:

- Direcciones IP
- Direcciones MAC
- Puertos origen y destino
- Protocolos (encabezados + payloads)
- Comportamientos, por ejemplo: la velocidad con la que entran los paquetes.

Las políticas que son dadas por el agente controlador son llamadas reglas de acceso, listas de control o reglas del firewall.

2.4.3.1 Funcionamiento de Packet Filter

Para saber cómo funciona Packet Filter se debe de tener en cuenta:

- Las reglas de acceso
- La interpretación de entrada (in) y salida (out) de las interfaces

Reglas de acceso

También conocidas como las políticas de acceso para el firewall se definen basadas en reglas de acceso que se hayan definido para los usuarios, y según los recursos que pueden ser puertos, protocolos, estaciones, entre otros. Un firewall en OpenBSD con PF será un archivo en texto plano que contiene una lista de reglas que se evaluaran según la lógica que se muestra en la tabla siguiente.
Se evalúa el paquete según las reglas que se definieron empezando por la REGLA-1 (las reglas se leen secuencialmente) hasta terminar con la tabla ya que para aceptar o denegar el paquete PF espera a que la evaluación esté terminada y lleve a cabo hasta la última regla. Es importante tener en cuenta cual será la acción final que se va a ejecutar ya que de esta depende como se comportara el firewall.

**Reglas de acceso “Quick”**

Esta regla se refiera a que la lógica de PF es modificada debido a la palabra *quick*, la cual al ser encontrada se ejecutará inmediatamente la acción que tenga configurada en ella, sin esperara a leer las siguientes reglas.

**Estructura (sintaxis) de las reglas de filtrado**

La estructura de las reglas como se puede ver es algo muy simple, solamente se selecciona la información opcional de los corchetes y se crean las reglas. Por ejemplo:

```
pass|block [in|out] [quick] [on interface] [proto protocolo] [from fuente] [to destino]
[port puerto]
```

**El asunto de las interfaces**

La sintaxis de PF es simple, lo complicado es entender cómo se comporta PF en las interfaces, según si tienen una única interfaz, son más de una, se manejan estados o no.

a. Una interfaz de red
En este caso es una máquina usando OpenBSD conectada a una red local o a Internet, en este caso se tiene la posibilidad de implantar políticas de acceso en la interfaz de red fxp0 que es la que se conecta a la red.
Las reglas de entrada se aplicarán para el paquete que entra a la interfaz y cuando este salga de la interfaz como se aplicarán las reglas de salida según se haya indicado. Así que el paquete tendrá que pasar dos pruebas antes de llegar a su origen ya sea que venga desde el mismo firewall o desde Internet.

FIGURA 1. Esquema básico de PF habilitado en una tarjeta de red.

b. Dos interfaces o más
Esto es lo más común cuando se quiere que el sistema OpenBSD se comporte como FW y gateway de salida a la vez. Se tendrán dos interfaces de red rl0, fxp0 desde las cuales se podrán aplicar políticas a los paquetes que pasan por allí.
Cuando hay dos interfaces de red, PF aplicará reglas en 4 puntos como se muestra en la figura.

FIGURA 2. Esquema avanzado de PF aplicado a dos interfaces.
Si un paquete entra por fxp0 y no está dirigido a la máquina local (el *firewall*) entonces pasará a una segunda interfaz rl0 donde tendrá que entrar y salir nuevamente.

Si un paquete entra por rl0 y no está dirigido a la máquina local (el *firewall*) entonces pasará a una segunda interfaz fxp0 donde tendrá que entrar y salir nuevamente.

Cuando se manejan estados solo se requiere que los paquetes se verifiquen en una sola dirección de la interfaz, entrando o saliendo.

### 2.4.4 CARP

CARP (*Common Address Redundancy Protocol*) es un protocolo que permite que múltiples usuarios en la misma red local compartan un conjunto de direcciones IP. Su propósito primario es proveer una redundancia en el *failover* (tolerancia a fallos, capacidad de un sistema de acceder a la información), este es usado especialmente en *firewall* y *router*.


Si se tiene una máquina la cual tiene un filtro de contenido y este se apaga lo que pasará es que las redes no se podrán comunicar entre sí o se comunicarán sin que haya un filtrado de paquetes. En cambio si el número de máquinas que corren el filtro de contenido y CARP, y una de ellas falla entonces la otra tomará el lugar de esta manteniendo los servicios del filtro de contenido arriba, las computadoras que estén a cada lado de este seguirán trabajando normalmente.

Para asegurarse que el nuevo filtro de contenido maestro se encuentra operando igual que el anterior se utiliza *pfsyncd* [16] el cual sincroniza los estados del filtro de contenido, es usado por PF.
Un grupo de host que utilizan CARP es llamado grupo de redundancia (group of redundancy), el cual aloja una dirección IP que es compartida o dividida por los miembros del grupo. Conjuntamente a este grupo, un host es designado como maestro (master) y los otros miembros son conocidos como esclavos (slaves).

Un uso común para CARP es la creación de un grupo de firewall con redundancia. Si el firewall principal está desconectado de una red o se encuentra fuera de uso, las direcciones IP serán tomadas por uno de los firewall esclavos y la disponibilidad del servicio no será interrumpida.

2.5 Integración de tecnologías

Según la Real Academia Española “RAE” la palabra integración viene del latín integrare que significa “dicho de las partes: constituir un todo”. Explicado de otra manera sería que por medio de varias partes se puede obtener un todo, este “todo” tendrá cada característica y propiedad que cada parte tiene por separado, es decir, en vez de tener cada una de las partes, lo cual en muchos casos es una inversión mucho mayor o incluso difícil para trabajar con ellas solo se tendrá una pieza (un todo) la cual será capaz de realizar cada una de las funciones de las partes, ahorrándose así la necesidad de comprar o de obtener cada pieza que necesite para un determinado fin.

La definición de tecnología según la RAE “ Conjunto de teorías y de técnicas que permiten el aprovechamiento práctico del conocimiento científico” la cual proviene del griego tecnos que es arte y logos que es tratado. Gracias a ello podemos decir que a través de la teoría y la técnica se llega a tener una manera muy práctica de aprovechar el conocimiento científico.

Dicho lo anterior se tiene que la integración de tecnologías es un conjunto de herramientas las cuales permiten tener un aprovechamiento práctico de ellas en un solo todo. Las
herramientas que son integradas tienen la función de responder a necesidades de las personas, así como de las instituciones y/u organizaciones.

Gracias a la integración de tecnologías es posible obtener varios beneficios en un solo producto, lo cual hace que sea más práctico, fácil el manejo y la administración de la tecnología que ha sido integrada.

2.5.1 Herramientas de licencia libre

El software que es libre o de licencia libre concierne a libertad no al precio que se tenga que pagar o no por tenerlo. Si se quiere entender este concepto es como cuando se hace referencia a la libertad de expresión, mas no a comida gratis. La libertad en cuanto a software libre es la libertad con que los usuarios manipulan, copian, estudian, ejecutan, cambian y a su vez hacen una mejora en el software. Para resumir o concretizar se refiere a la libertad que tienen los usuarios con el software. Estas libertades que tienen los usuarios son cuatro, a continuación se describirán cada una de ellas:

- Libertad de uso del programa sin importar el propósito que se le dé.
- Libertad de estudiar el funcionamiento del programa, siendo adaptado a las necesidades.
- Libertad que se tiene al distribuir copias, esto con el fin de ayudar a otros.
- Libertad de mejorar el programa y hacerlas públicas, haciendo que todos tengan un beneficio de esto.

GNU es un proyecto el cual inicio en 1984 teniendo el objetivo de crear un sistema operativo completo tipo Unix el cual sería Software libre. Ya que el núcleo de GNU no se encuentra finalizado se usa GNU con el núcleo Linux, dando por resultado el sistema operativo GNU/Linux.
2.5.1.1 DansGuardian

DansGuardian es un filtro de contenido el cual controla el acceso a sitios web, incluye un filtro de virus. Es usado en instituciones de gobierno, educación y empresas. Es flexible y se adapta a la implementación. Corre sobre Linux, FreeBSD, OpenBSD, NetBSD, Mac OS X, HP-UX, y Solaris [17].

Filtra el contenido de las páginas basado en varios métodos incluyendo comparación de caracteres, filtro de PICS y filtro por URL.

Este software se ofrece con una doble licencia: GPL v2 o comercial dependiendo del uso que se le dará.

2.5.1.2 OpenBSD

No solo el software libre se limita a GNU/Linux. Uno de los ejemplos más claros son los BSD, es una familia de sistemas muy parecidos y a su vez eficientes como éste. Fue desarrollado por la Universidad de Berkeley en California en 1969 en los laboratorios Bell.

El objetivo de FreeBSD es proporcionar un sistema operativo que sea sólido como una roca y eficiente sobre todo, que obtenga el rendimiento máximo de la maquina. También existe el proyecto NetBSD, éste busca la portabilidad sobre todo: no se limita a funcionar sobre los procesadores Intel o los PPC de los Apple, es capaz de comportarse exactamente igual sobre una vertiginosa lista de máquinas menos frecuentes. Por último, se encuentra OpenBSD, el cual se dice que es el sistema operativo más seguro del mundo de software libre, en especial porque es recomendable para el uso de firewalls. [18] Otros miembros de la familia son Mac OS X y DragonFlyBSD.

El sistema operativo OpenBSD es como ya se menciono antes, el más seguro del mundo, esto es debido a la seguridad que posee así como en la criptografía. Este sistema se
concentra en la portabilidad, en el cumplimiento de las normas y regulaciones, así como en la seguridad proactiva y en la criptografía integrada.

Se distribuye bajo la licencia BSD, la cual es aprobada por la OSI. Esta licencia constaba de 4 cláusulas (la original) las cuales abarcaban los siguientes puntos:

- Copyright al distribuir el código fuente en conjunto con la lista de condiciones y el párrafo de responsabilidad.
  
  "copyright (c) <year>, <copyright holder> all rights reserved."
  
  *this software is provided by <copyright holder> "as is" and any express or implied warranties, including, but not limited to, the implied warranties of merchantability and fitness for a particular purpose are disclaimed. in no event shall <copyright holder> or contributors be liable for any direct, indirect, incidental, special, exemplary, or consequential damages (including, but not limited to, procurement of substitute goods or services; loss of use, data, or profits; or business interruption) however caused and on any theory of liability, whether in contract, strict liability, or tort (including negligence or otherwise) arising in any way out of the use of this software, even if advised of the possibility of such damage.

- Copyright en las distribuciones en formato binario.

- Todo material publicitario que mencione debe de mostrar “This product includes software developed by the University of California, Berkeley and its contributors.”

- El nombre de la Universidad y los nombres de sus colaboradores no se deben de utilizar para promocionar o apoyar productos derivados de este software sin permiso previo y por escrito.

A través del tiempo las cláusulas fueron modificándose de acuerdo a las necesidades que se fueron presentando y por algunas contradicciones que se fueron presentando. Quedando solamente dos cláusulas:

- La distribución del código fuente debe contener el copyright así como la lista de condiciones y el párrafo de responsabilidad.

- Las distribuciones en forma binaria deben reproducir el copyright, así como la lista de condiciones y la renuncia en la documentación y/u otros materiales suministrados con la distribución.
OpenBSD funciona actualmente sobre 17 plataformas distintas de hardware, en la figura 1 se muestran la tabla de plataformas soportadas por OpenBSD. [19]

<table>
<thead>
<tr>
<th>Plataformas soportadas por OpenBSD</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Alpha</strong></td>
</tr>
<tr>
<td><strong>amd64</strong></td>
</tr>
<tr>
<td><strong>Armish</strong></td>
</tr>
<tr>
<td><strong>hp300</strong></td>
</tr>
<tr>
<td><strong>Hppa</strong></td>
</tr>
<tr>
<td><strong>i386</strong></td>
</tr>
<tr>
<td><strong>Landisk</strong></td>
</tr>
<tr>
<td><strong>loongson</strong></td>
</tr>
<tr>
<td><strong>Macppc</strong></td>
</tr>
<tr>
<td><strong>mvme68k</strong></td>
</tr>
<tr>
<td><strong>mvme88k</strong></td>
</tr>
<tr>
<td><strong>Sgi</strong></td>
</tr>
<tr>
<td><strong>So曹操</strong></td>
</tr>
<tr>
<td><strong>Sparc</strong></td>
</tr>
<tr>
<td><strong>sparc64</strong></td>
</tr>
<tr>
<td><strong>Vax</strong></td>
</tr>
<tr>
<td><strong>Zaurus</strong></td>
</tr>
</tbody>
</table>

TABLA 3. Plataformas soportadas por OpenBSD.

Durante todo este tiempo, las mejoras aportadas por las diferentes versiones de BSD, fueron incorporadas en el UNIX original, o adaptadas, algunos ejemplos son:
- Nuevo soporte de memoria virtual.
- Sockets.
- Incorporación de los protocolos de red (TCP/IP) utilizados por la Defense Advanced Research Projects Agency (DARPA)
- Llamadas al sistema o sycalls.
- La librería termcap.
- El verificador de código C link, entre otros.

2.5.1.3 Kali Linux

Kali Linux, es una distribución basada en Debian GNU/Linux diseñada a partir de la reescritura de BackTrack, la cual es una más avanzada y versátil prueba de penetración.

Permite la instalación sobre arquitecturas i386, amd64 y ARM (armel y armhf) [20]. Permite la instalación vía red y brinda imágenes para la descarga de máquinas virtuales prefabricadas con las herramientas instaladas de VMWare.

Es una plataforma de seguridad y auditoría, por lo cual para el uso de varias herramientas es necesario sean ejecutadas con privilegios de root.

Tiene instalados varios programas: Nmap, Wireshark, Jhon the Ripper y la suite Aircrack-ng, los cuales ayudan para el escaneo de redes así como en las auditorías.
Capítulo 3. Materiales y Métodos

Este proyecto consistió en el desarrollo de un prototipo de firewall virtualizado el cual fue integrado con varias herramientas como lo son la alta disponibilidad, filtro de contenido y DansGuardian todo esto utilizando un servidor virtual así como máquinas virtuales en las cuales se instaló el sistema operativo OpenBSD.

Se utilizó el protocolo CARP (Common Address Redundancy Protocol) que permite la conexión de múltiples host conectados a través de la misma red local y que se puedan utilizar un conjunto compartido de direcciones IP. Gracias a este protocolo se logró alta disponibilidad, si en el firewall principal (Firewall 1) llegaran a caer los servicios que son proporcionados entonces los servicios quedarían arriba sin sufrir daños gracias a que el firewall de respaldo (Firewall 2) tomaría su lugar.

Packet Filter es el bloqueo o paso selectivo de paquetes de datos mientras pasan a través de una interface de red. El criterio que realiza para la inspección de paquetes está basado en los encabezados de Capa 3 y Capa 4 los más utilizados son de origen y dirección de destino así como puerto de origen y de destino, y el protocolo.

Con el fin de comparar el prototipo de firewall con un firewall comercial y así probar su efectividad se crearon tres máquinas virtuales las cuales fueron utilizadas a la hora de realizar las pruebas de penetración, gracias a la herramienta Kali Linux.

Kali Linux es una herramienta la cual tiene programas como Nmap que es un escáner de puertos, Wireshark un sniffer utilizado para realizar análisis de protocolos, John the Ripper el cual es un crackeador de contraseñas, así mismo también contiene la suite Aircrack-ng el cual es un software de seguridad inalámbrica que analiza paquetes de redes e inyectar tráfico en la red, dicha herramienta se utilizó con el propósito de realizar las pruebas de penetración al prototipo de firewall.

Se realizó previamente una investigación de los componentes al igual que su funcionamiento, lo cual será explicado en este capítulo.
3.1 Descripción del área de estudio

El tipo de proyecto que se desarrolló fue de desarrollo tecnológico ya que fue necesaria la integración de varias herramientas tecnológicas. Además consta de una evaluación prospectiva, longitudinal, analítica y experimental que se llevó a cabo al realizar una comparación de un firewall comercial con el prototipo y el resultado obtenido.

Así mismo se realizó en primer lugar una investigación donde se llevaron a cabo mediciones y se realizaron pruebas para ver cómo se comportaba el prototipo ante ciertas situaciones.

3.2 Materiales

Las herramientas a utilizar para llevar a cabo el prototipo, se encuentran a continuación, separadas a nivel de hardware y software.

3.2.1 Hardware

Para poder realizar dicho proyecto se utilizaron dos computadoras físicas, una workstation que funge como servidor en el cual se desarrolló el laboratorio donde se realizó la caracterización de una red de educación media superior y las pruebas pertinentes.

Especificaciones de Servidor:

- Dell Precision Workstation T7500
- 4 procesadores 2.393 GHz, Intel® Xeon®
- Memoria: 30GB
3.2.2 Software

Después de una investigación se decidieron las herramientas indicadas tanto por su versatilidad como su compatibilidad para el desarrollo de este prototipo, mismas que se enlistan a continuación.

- VMware server ESXi 5.5 (Evaluation Mode)
  Hipervisor con una licencia temporal, sistema desde donde se crearon y administraron las máquinas virtuales utilizadas a lo largo del proyecto.

- OpenBSD 5.5
  Sistema operativo de licenciamiento libre, fue instalado en dos de las máquinas virtuales, en las cuales se configuró el firewall.

- Windows Server 2008 R2 Service Pack 2
  Se instaló de Windows Server 2008 R2 Service Pack 2 en tres de las máquinas.

- Kali Linux
  Se utilizó con el propósito de realizar las pruebas de penetración en el prototipo.

- DansGuardian 2.12
  Herramienta de código abierto para filtrar contenido web.

- Microsoft Exchange Server 2010
  Servidor de almacenamiento de correos electrónicos, desarrollado por Microsoft.

- Active Directory Computer and Users
  Consola de manejo (Microsoft Management Console MMC) para administrar información.

- IIS 7
  Servidor web el cual contiene diversos servicios para Microsoft Windows.
Fueron manejadas cinco máquinas virtuales, que fueron creadas y administradas por el hipervisor VMware, en dos de ellas se realizó la instalación del sistema operativo OpenBSD 5.5 el cual es un software de licenciamiento libre que permite y hace posible la configuración y creación de un firewall; de igual manera se logró la instalación de Windows Server 2008 R2 Service Pack 2 en tres de ellas, así mismo se realizó la instalación de servicios en estas máquinas virtuales.

En las máquinas con sistema operativo OpenBSD se realizó la configuración de un filtro de contenido, y se les configuró para tener alta disponibilidad con ayuda del protocolo CARP y pfsync, así como la integración de DansGuardian 2.12

Después de una investigación se decidieron las herramientas indicadas tanto por su versatilidad como su compatibilidad para el desarrollo de este prototipo.

**Servicios**

**VM 1**
Microsoft Windows Server 2008 64 bits
Memoria: 4GB
IIS 7.0
Microsoft Exchange Server 2010
ip: 172.16.68.61

**VM 2**
Microsoft Windows Server 2008 64 bits
Memoria: 4GB
Active Directory Computer and Users
ip: 172.16.68.62
VM 3
Microsoft Windows Server 2008 64 bits
Memoria: 4GB
DNS Server
ip: 172.16.68.63

3.3 Métodos

El prototipo se realizó en la Universidad Autónoma de Ciudad Juárez (UACJ) en el Instituto de Ingeniería y Tecnología durante el periodo febrero 2014 a enero 2015.

El universo que se tomó fue a partir de redes institucionales a nivel superior (estudio de las principales características con las que cuentan tales redes en particular). Se llevó a cabo la muestra con personal de Redes de la UACJ.

3.3.1 Técnicas de investigación

La técnica de investigación que se utilizó fue sistemática ya que se llevó a cabo una serie de pasos con el fin de concluir el prototipo de Firewall virtual.

3.3.2 Metodología

A lo largo del desarrollo del proyecto se utilizaron varios criterios, tales como: pruebas de estrés, penetración y revisión de los puertos.

Se llevó a cabo un análisis de red que comúnmente se visualizaría en instituciones de nivel media superior, así como las habituales fallas y sucesos de concurrencia de usuarios. La
información proporcionada acerca de tránsito de red de instituciones fue mediante entrevistas a personal de redes de la Universidad Autónoma de Ciudad Juárez.

De igual manera la seguridad es clave importante en el proyecto, por tal motivo se analizó profundamente las medidas, reglas y políticas necesarias para que el prototipo contara con un sistema seguro libre de ataques o caídas de servicios.

En una institución de nivel media superior, el contenido que se filtra por sus redes debe ser en cierta forma más estricta y cuidada. Se realizaron varias investigaciones sobre filtros de contenidos y sus funcionalidades con el fin de seleccionar el tráfico a filtrar mediante reglas de bloqueo y acceso.

Para mantener un sistema con alta disponibilidad, con las funcionalidades que se requerían en el proyecto, fue necesario hacer uso del protocolo CARP, el cual fue investigado y analizado, así como la herramienta pfsync, la cual hace comunicación entre los firewall para lograr la alta disponibilidad.

### 3.3.3 Procedimiento

Se llevaron a cabo investigaciones sobre los diversos temas a tratar. Con los datos adquiridos se realizó un estimado de las herramientas y tecnologías adecuadas para el desarrollo del prototipo, tales como:

- OpenBSD.
- Protocolo CARP (*Common Access Redundancy Protocol*).
- Packet Filter.
- Alta disponibilidad.
- DansGuardian.
Así como los conocimientos teóricos necesarios para el desarrollo de un firewall:

- Tipos de firewall.
- Políticas.
- Reglas.
- Limitaciones.

FIGURA 3. Metodología del proyecto.

Se diseñó una topología en la cual se visualizó la estructura que el firewall virtual tendrá, con el fin de analizar cada elemento del prototipo.
FIGURA 4. Esquema de Firewall

En la figura 4 se muestran el *Firewall* 1 (activo) y *firewall* 2 (pasivo) que están entre la red externa y la red interna, así mismo contienen una interfaz para red externa e interna; en ésta última red se alojan tres máquinas con servicios comunes en un centro de datos.

**Desarrollo**

Se procedió a recrear un laboratorio virtual, en el cual fue necesario el uso de un servidor Workstation Dell Precision T7500 Intel Xeon. A dicho servidor se le instaló como sistema operativo VMware EXSi 5.5 (*Evaluation Mode*) y así fungir como hipervisor en donde se alojaron las máquinas virtuales a utilizar.

Dentro del mismo, fueron creadas las tres redes necesarias para la comunicación entre la red externa, el *firewall* y la red interna como se muestra en las siguientes figuras.
FIGURA 5. Red Externa

Red Externa
148.210.68.0/24

Firewall 1
interfaz: em0
IP: 148.210.68.68

Firewall 2
interfaz: em0
IP: 148.210.68.69

FIGURA 6. Diagrama de red Externa

FIGURA 7. Red CARP

Red Carp
10.10.10.0/24

Firewall 1
interfaz: em1
IP: 10.10.10.1

Firewall 2
interfaz: em1
IP: 10.10.10.2

FIGURA 8. Diagrama de red CARP
Dos de las máquinas creadas se configuraron con el sistema operativo de licenciamiento libre OpenBSD 5.5 y las mismas características, las cuales fueron creadas para formar parte del firewall virtual. Se agregaron cuatro tarjetas de red, de las cuales se utilizaron tres para crear redes dentro del entorno virtual:

- Red interna.
- Red externa.
- Red CARP (comunicación entre los dos firewall).
FIGURA 11. Propiedades de la Máquina virtual Firewall (OpenBSD instalado).

Especificaciones de máquinas firewall:

- OpenBSD 5.5
- 4 procesadores
- Memoria: 2 GB
- 4 adaptadores de red

A las dos máquinas se les dio nombre, Firewall 1 y Firewall 2 respectivamente, el Firewall 1 tuvo como tarea ser el firewall maestro o activo, al Firewall 2 se le asignó ser el firewall de respaldo o pasivo.
FIGURA 12. Diagrama de los firewall

**Firewall 1**

<table>
<thead>
<tr>
<th>Interfaz</th>
<th>IP</th>
<th>Descripción</th>
</tr>
</thead>
<tbody>
<tr>
<td>Em0</td>
<td>148.210.68.68</td>
<td>ip firewall</td>
</tr>
<tr>
<td>Em1</td>
<td>10.10.10.1</td>
<td>Pfsync</td>
</tr>
<tr>
<td>Em2</td>
<td>172.16.68.2</td>
<td>red interna</td>
</tr>
<tr>
<td>Carp1</td>
<td>172.16.68.1</td>
<td>virtual interna</td>
</tr>
<tr>
<td>Carp2</td>
<td>148.210.68.64</td>
<td>virtual externa</td>
</tr>
</tbody>
</table>

**Firewall 2**

<table>
<thead>
<tr>
<th>Interfaz</th>
<th>IP</th>
<th>Descripción</th>
</tr>
</thead>
<tbody>
<tr>
<td>Em0</td>
<td>148.210.68.69</td>
<td>ip firewall</td>
</tr>
<tr>
<td>Em1</td>
<td>10.10.10.2</td>
<td>Pfsync</td>
</tr>
<tr>
<td>Em2</td>
<td>172.16.68.3</td>
<td>red interna</td>
</tr>
<tr>
<td>Carp1</td>
<td>172.16.68.3</td>
<td>virtual interna</td>
</tr>
<tr>
<td>Carp2</td>
<td>148.210.68.66</td>
<td>virtual externa</td>
</tr>
</tbody>
</table>

TABLA 4. *Firewall1* IPs

TABLA 5. *Firewall2* IPs

Primeramente en cada una de las máquinas, se configuró la red (ip, máscara, broadcast, gateway) en la interfaz em0.

La em1 fue asignada a la red CARP para la comunicación entre los dos firewall, esta configuración se llevó a cabo con pfsync, herramienta integrada a OpenBSD, que tiene
como objetivo sincronizar la tabla de estados entre los firewall. En caso de que se produzca una conmutación, el tráfico puede fluir sin interrupción a través del nuevo firewall maestro, también exponerlos cambios que se realicen en la tabla de estados mediante monitorización. Para la realización de esto se utilizó el siguiente comando:

```bash
Sintaxis:

# ifconfig pfsync0 syncdev em1
```

*pfsync0*: Nombre de la interfaz

*syncdev*: El nombre de la interfaz física usada para enviar actualizaciones a pfsync.

**FIGURA 13. Configuración pfsync**
Configuración Protocolo CARP (*Firewall* 1):

<table>
<thead>
<tr>
<th>! configuración de CARP en red interna</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td># ifconfig carp1 create</td>
<td></td>
</tr>
<tr>
<td># ifconfig carp1 vhid 1 carpdev em2 pass</td>
<td></td>
</tr>
<tr>
<td>fwv2015i 172.16.68.1 netmask 255.255.255.0</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>! configuración CARP en red externa</th>
</tr>
</thead>
<tbody>
<tr>
<td># ifconfig carp2 create</td>
</tr>
<tr>
<td># ifconfig carp2 vhid 2 carpdev em0 pass</td>
</tr>
<tr>
<td>fw2015e 148.210.68.64 netmask 255.255.255.0</td>
</tr>
</tbody>
</table>

**FIGURA 14. Configuración CARP**

### 3.3.4 Packet Filter

Después de las configuraciones anteriores de red y para que el *firewall* trabaje con alta disponibilidad, se trabajó con las herramientas para realizar el filtrado de paquetes. Dicho filtro se dividió en dos módulos: Traducción de direcciones de redes NAT (por sus siglas en inglés) y PF. Para lograr la integración de cada una de las herramientas (PF y Dansguardian) fue necesario colocar las reglas de acceso y denegación en el archivo `/etc/pf.conf`, tales reglas se mostrarán más adelante.

#### 3.3.4.1 Network Address Translation

Primeramente fue necesario habilitar de manera permanente el reenvío de IP (*IP forwarding*) en el archivo `/etc/sysctl.conf` para activar el reenvío de IP, se reinició la máquina en donde se configuró.

, con la siguiente línea de comando:

```
net.inet.ip.forwarding=1
```

Se inició configurando NAT, ya que es el encargado de realizar la traducción de los paquetes que se envían y se reciben. Dentro del archivo de configuración de PF (/etc/pf.conf), fueron declaradas macros para simplificar la comprensión de las configuraciones y posibles cambios en el futuro:
Se realizó un NAT estático de tipo bidireccional, en el cual los paquetes que son enviados desde la red interna (171.16.68.0/24), salen por la ip de la interfaz em0 (148.210.68.68), la cual se colocó como su dirección de origen debido al NAT. De igual manera, los paquetes entrantes por 148.210.68.68 se redireccionaron hacia la red interna.

#NAT

```
match out on $ext_if from $int_if:network to any nat-to $ext_if
```

Para comprobar su funcionamiento y el estado de NAT, así como la muestra de las sesiones actuales se aplicó el comando siguiente:

```
pfctl -s state
```

**FIGURA 15. Comprobación de estado de NAT**
3.3.4.2 Reglas de filtrado de contenido

Para empezar con la herramienta de PF fue necesario analizar el grado de seguridad que se quería en el prototipo, así como tener claramente definido lo que se quería bloquear o dejar pasar a través de la red externa e interna. Como anteriormente se había señalado, el prototipo va dirigido a un ambiente institucional y para un administrador de servicios, de este modo se enfocaron las reglas hacia servicios que habitualmente se tienen en un centro de datos. Los objetivos de seguridad fueron los siguientes:

- Permitir entrada y salida de paquetes en la red local.
- Habilitar puertos necesarios (80, 443, 53, 25, 22, 3306), así como priorizar algunos de esos puertos.
- Bloqueo de redes reservadas.
- Normalizar el tráfico.
- Verificar el tráfico entrante y saliente.
- Protección contra paquetes o direcciones de origen falsas.

Como se mencionó anteriormente, PF es el sistema de filtrado de OpenBSD, el cual está integrado con el sistema operativo. Este se habilitó por default, pero para mayor seguridad se colocó el comando a continuación para activarlo.

```
pf=YES
pflog_enable="YES"
```

Esta herramienta cuenta con un archivo especial, el cual cuenta con texto que es interpretado por PF donde son especificadas las reglas que se utilizaron para filtrar los paquetes.

```
/etc/pf.conf
```
Se crearon dos macros más, además de las mencionadas anteriormente, para señalar los puertos de los servicios a utilizar, tanto UDP como TCP.

udp_ports="{53}"  
tcp_ports="{22, 25, 53, 443, 80, 8080, auth}"  

Se tomó la práctica de denegación determinada, la cual consiste en denegar el paso a todo, donde se bloquea el tráfico en todas las interfaces en cualquier dirección, y desde cualquier origen hasta cualquier destino. Así se permitió el acceso de forma selectiva y personalizada, para realizar el bloqueo se colocaron las primeras dos reglas:

block in all
block out all

Los puertos señalados en las macros anteriores fueron los únicos que obtuvieron acceso, de igual forma se indicó un registro de estado.

#TRAFICO A PROTOCOLOS Y PUERTOS AUTORIZADOS

pass in log on $int_if inet proto tcp from $int_if:network to !$int_if port $tcp_ports keep state

pass out log on $ext_if inet proto tcp from $int_if:network to !$int_if port $tcp_ports keep state

pass in log on $int_if inet proto udp from $int_if:network to !$int_if port $udp_ports keep state

pass out log on $ext_if inet proto udp from $int_if:network to !$int_if port $udp_ports keep state
pass in log on $int_if inet proto icmp from $int_if:network to !$int_if
pass out log on $int_if inet proto icmp from $int_if:network to !$int_if

pass in on $int_if from $red_int
pass out on $int_if to $int_if:network

pass on $ext_if from $int_if:network to any
pass out on $ext_if proto { tcp udp icmp } all modulate state

FIGURA 16. Autorización de tráfico

Se autorizó tanto el protocolo CARP, así como pfsync para su correcto funcionamiento.

pass on {em0 em2} proto carp keep state
pass quick on em1 proto pfsync keep state

Se normalizaron los paquetes entrantes.
match in all scrub (no-df)

Las siguientes instrucciones fueron colocadas para hacer uso de respuesta predeterminada para las reglas de filtrado block, así mismo la segunda línea se usó para activar el registro de las estadísticas de la interfaz externa.
set block-policy return

Se colocó tanto para la interfaz interna como la externa, una instrucción para contrarrestar problemas de falsificación de direcciones.
antispoof for $ext_if
anttispoof for $int_if

Se creó una tabla en la que se bloquean las redes privadas y no autorizadas.

```
table <blocked_nets> { 0.0.0.0/8, 10.0.0.0/8, 127.0.0.1/8, 169.254.0.0/16, 172.16.0.0/12, 192.0.2.0/24, 192.88.99.0/24, 192.168.0.0/16, 198.18.0.0/15, 198.51.100.0/24, 203.0.113.0/24, 224.0.0.0/4, 240.0.0.0/4 }
```

block in log on $ext_if from <blocked_nets>

Para comprobar el envío de paquetes a través de la red interna y externa, se realizaron pruebas con el comando ping, de esta forma se comprobó que eran satisfactorias las respuestas hacia las distintas interfaces.
FIGURA 17. Ping hacia firewall 1 desde red interna.

FIGURA 18. Ping de firewall 1 hacia la red interna.
3.3.5 DansGuardian (filtro de contenido web)

Para comenzar a trabajar con DansGuardian fue necesario realizar la instalación de Squid, el cual es un complemento para el filtro de contenido web; y se configuró en el puerto 3128 ya que DansGuardian se conecta a él a través de este puerto, a su vez DansGuardian aceptará conexiones por el puerto 8080. Es necesario que estar conectado como root durante la instalación.

Primeramente, se descargó squid 3.3.11

```
#pkg_add -v ftp://ftp.openbsd.org/pub/OpenBSD/5.5/packages/amd64/squid-3.3.11.tgz
env FLAVOR=transparent make install clean
```

Una vez que fue instalado correctamente, se ingresó al archivo /etc/squid/squid.conf, ahí es donde se cambió la información de la red utilizada.

```
# vi /etc/squid/squid.conf
http_port 172.16.68.2:3128
acl our_networks src 172.16.68.0/24
http_access allow 172.16.68.0/24
httpd_accel_port 80
httpd_accel_host virtual
httpd_accel_with_proxy on
httpd_accel_uses_host_header on
```

Después se descargó DansGuardian por medio de wget y se instaló (DansGuardian usa el autoconf de gnu, auto-generando el archivo Makefile con el script "configure".)

```
cd /tmp
wget http://sourceforge.net/projects/dansguardian/files/dansguardian-2.12.0.3.tar.bz2/download
cd dansguardian-2.12.0.3
```
Para configurar (con la opción de ayuda *help* podrá ver las opciones de configuración) y compilar se realizó lo siguiente:

```
./configure --cgidir=/var/www/cgi-bin/ \  
--sysconfdir=/usr/local/etc/dansguardian/ \ 
--sysvdir=/usr/local/etc/rc.d/ \ 
--bindir=/usr/sbin/ \  
--mandir=/usr/share/man/mkdir /var/log/dansguardian
make
make install
make clean
```

La siguiente lista explica cada una de las opciones de configuración:

<table>
<thead>
<tr>
<th>Opción</th>
<th>Descripción</th>
<th>Valor predeterminado</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Bindir</strong></td>
<td>Donde se instalan los binarios o ejecutables</td>
<td>/usr/sbin/</td>
</tr>
<tr>
<td><strong>Sysconfdir</strong></td>
<td>Donde se instalan los archivos de configuración y datos</td>
<td>/etc/dansguardian/</td>
</tr>
<tr>
<td><strong>Sysvdir</strong></td>
<td>Donde se instala el <em>script</em> de inicio</td>
<td>/etc/rc.d/init.d/</td>
</tr>
<tr>
<td><strong>Cgidir</strong></td>
<td>Ubicación del directorio cgi-bin</td>
<td>/home/httpd/cgi-bin/</td>
</tr>
<tr>
<td><strong>Mandir</strong></td>
<td>Ubicación de los manuales</td>
<td>/usr/man/</td>
</tr>
<tr>
<td><strong>Logdir</strong></td>
<td>Donde se crearán los archivos históricos (<em>logs</em>)</td>
<td>/var/log/dansguardian/</td>
</tr>
<tr>
<td><strong>runas_usr</strong></td>
<td>El usuario de sistema corre como</td>
<td>Nobody</td>
</tr>
<tr>
<td><strong>runas_grp</strong></td>
<td>El grupo del usuario del sistema es</td>
<td>Nobody</td>
</tr>
<tr>
<td><strong>Piddir</strong></td>
<td>Ubicación del archivo de identificación del proceso (pid)</td>
<td>/var/run/</td>
</tr>
</tbody>
</table>

**TABLA 6. Opciones de configuración DansGuardian.**

Se configuró DansGuardian para que trabajara con Squid
# vi /usr/local/etc/dansguardian/dansguardian.conf
reportinglevel = 3
languagedir = '/usr/local/etc/dansguardian/languages'
language = 'usenglish'
loglevel = 1
filterip = 172.16.68.0/24
filterport = 8080
proxyip = 172.16.68.1
proxyport = 3128

Lo siguiente fue editar /etc/pf.conf para tener a PF trabajando con Squid y DansGuardian
# vi /etc/pf.conf

# Proxy transparente y DansGuardian
rdr on $int_if inet proto tcp from any to any port www -> 148.210.68.1 port 8080

# PASSES FOR TRANSPARENT PROXY
pass in on $int_if inet proto tcp from any to 148.210.68.1 port 8080 keep state
pass out on $ext_if proto tcp from any to any port www keep state
pass in all

Se procedió a iniciar Squid, DansGuardian, y reiniciar PF con las nuevas reglas.

Iniciar Squid
# /usr/local/sbin/squid -z
después se usó /usr/local/sbin/squid the -z para crear los directorios swap

Iniciar Dansguardian
# /usr/local/etc/rc.d/dansguardian.sh start

Reiniciar PF para que lea las nuevas reglas en pf.conf
# pfctl -f /etc/pf.conf
Al tener un inicio exitoso se sabe que se está corriendo OpenBSD como firewall con filtro de contenido y un proxy transparente.

Para iniciar Squid cuando el firewall reinicie se necesitó agregar lo siguiente a /etc/rc.local

```
#START SQUID
if [-x /usr/local/sbin/squid ]; then
  echo -n ' squid'; /usr/local/sbin/squid
fi
```

También se puede iniciar DansGuardian manualmente haciendo lo siguiente

```
#START DANSGUARDIAN
if [-x /usr/local/sbin/dansguardian ]; then
  echo -n ' dansguardian'; /usr/local/sbin/dansguardian
fi
```

Cuando una página es denegada, DansGuardian re direcciona al usuario al script cgi perl en su servidor web, donde el usuario verá el reporte. De esta forma es fácil personalizar los mensajes que el usuario recibirá. Para configurar la dirección del servidor web que mostrara la página con el reporte de error, modifique la variable accessdeniedaddress dentro del archivo dansguardian.conf

Se configuro la rotación de los archivos históricos (logs) para que evitara que el crecimiento desmedido de los históricos llene el sistema de archivos. DansGuardian tiene cinco de estos archivos access.log, access.log1... así hasta el 4. Cada semana era eliminado uno de estos archivos (el más antiguo). Fue necesario entonces agendar la ejecucion de este proceso de la siguiente manera: Se ejecuto crontab -e (como root) y se agregó la siguiente línea:

```
59 23 * * sat /etc/dansguardian/logrotation
```

Con esto se agendó la rotación todos los sábados a las 23:59.
3.3.6 Pruebas de penetración

- Denegación de servicio
Se realizó la prueba de penetración con LOIC (*Low Orbit Ion Cannon*) el cual es una aplicación de código abierto, realiza un ataque de denegación de servicio del objetivo al enviar paquetes TCP, UDP o peticiones HTTP con el fin de determinar cuál es la cantidad de peticiones por segundo que puede resolver la red objetivo antes de colapsar.

- ICMP flood
Internet Control Message Protocol ICMP flood esta técnica fue utilizada para atacar el ancho de banda enviando de manera continua un número elevado de paquetes ICMP ping de un gran tamaño en la red objetivo (paquetes de solicitud de eco y de respuesta), se realizó con ayuda de nping.

```
# nping --tcp-connect --flags syn --dest-port 80 --rate 90000 --count 900000 148.210.68.68
```

- Ping Sweep
Es un método que establece un rango de direcciones IP que encuentra host activos. Esto se realizó con NMAP (Network Mapper) el cual es un descubridor de redes y servicios dentro de una red. Algunas de sus funciones son: escaneo de puertos, descubre puertos, detección de servicios dentro de una red remota para determinar el nombre de la aplicación así como la versión, también determina el sistema operativo y las características de hardware de los dispositivos de red, interacción con el objetivo por medio de script.

Se utilizó el siguiente comando para realizar la prueba
```
# nmap -sP 148.210.68.68
```

- Enumeración de servicios (escaneo)
Se llevó a cabo un escaneo al objetivo en general con el fin de saber qué servicios está proveyendo y la información de cada uno de ellos.

Esta prueba se realizó con NMAP de la siguiente manera:
```
# nmap -sVC -O -T4 148.210.68.68
```
3.3.7 Contratiempos durante la integración

Los contratiempos que se tuvieron durante la integración y realización de la caracterización del firewall virtualizado fueron:

- Incopatibilidad del hipervisor Hiper-V con OpenBSD
  No se lograba tener acceso a la tarjeta de red por lo cual se tuvo que hacer un cambio en lo referente al hipervisor.

De igual manera a la hora de instalar OpenBSD, como sistema operativo en las máquinas virtuales de Hiper-V fue complicada su instalación ya que el hipervisor no veía la imagen del sistema operativo.
Debido a estos problemas se optó por utilizar el hipervisor VMware.

- Perdida de información del servidor donde se aloja el proyecto.
  Hubo pérdida de información varias veces en el Servidor Host, ya que no se contaba con respaldos una de estas veces fue una pérdida total de la información con la que se contaba.
Capítulo 4. Resultados de la investigación

Este capítulo muestra los resultados del prototipo de firewall virtualizado:

- Realización de pruebas para comprobar funcionamiento, eficacia, disponibilidad y protección contra ataques.

El método de recolección de información es dado por medio de las pruebas realizadas al firewall, tales como bloqueos de firmas de ataque, pruebas estrés son algunas de ellas.

4.1 Presentación de resultados

4.1.1 Resultados de pruebas realizadas

CARP (failover)
Se realizaron pruebas para forzar un failover y comprobar que la alta disponibilidad funcionaba correctamente. Como fue mencionado anteriormente, el firewall1 fue configurado para desempeñar el papel de master y el firewall2 como backup, se bajó la interfaz de carp1 para checar el funcionamiento mediante el comando ifconfig carp1 down.

FIGURA 19. Prueba Alta disponibilidad
En la figura anterior, se puede observar que al momento de realizar dicho cambio, inmediatamente la interfaz de carp1 en el **firewall2** se convierte en **master** sin afectar nada dentro del sistema.

Otro método que se realizó para la alta disponibilidad, fue aumentar el **advskew** de la interfaz **master**, esto con el fin de bajar el nodo maestro y ver nuevamente la respuesta del **firewall2** (**backup**).

Ambas pruebas funcionaron correctamente sin ningún contratiempo, comprobando así que se cuenta con un **firewall** con alta disponibilidad.

Pruebas de penetración:
- Denegación de Servicio, realiza un ataque de denegación de servicio del objetivo al atacar el ancho de banda, enviando de manera continua con un número elevado de paquetes TCP, UDP o peticiones HTTP de un gran tamaño en la red (paquetes de solicitud de eco y de respuesta).

```
# nping --tcp-connect --flags syn --dest-port 80 --rate 90000 --count 900000 148.210.68.69
```

![FIGURA 20. Nping al firewall1 por puerto 110](attachment://figura20.png)
FIGURA 21. Nping al firewall2 por puerto 80

![Imagen de Nping al firewall2 por puerto 80]

FIGURA 22. Nping firewall2 por puerto 80 continuación

- Ping Sweep, establece un rango de direcciones IP que encuentra host activos.

nmap -sP 148.210.68.68
FIGURA 23. Resultado Nmap firewall con Zenmap

- Enumeración de servicios, escaneo general del objetivo con el fin de saber qué servicios está proveyendo y la información de los mismos.

#nmap -sVC -O -T4 148.210.68.68

FIGURA 24. Nmap escaneo general firewall
- Zenmap muestra topología del objetivo a vulnerar así como los detalles del mismo.

FIGURA 25. Detalles utilizando Zenmap a firewall1

FIGURA 26. Topología del resultado al escanear firewall1 y firewall2

FIGURA 27. Nmap firewall1
Análisis e interpretación de resultados

Los resultados obtenidos de las pruebas realizadas al firewall respecto a la alta disponibilidad como a la integración de Packet Filter con DansGuardian, se concluyeron de manera funcional y efectiva.

Los resultados obtenidos de las pruebas de penetración no fueron los deseados ya que el firewall utilizado por la red de la UACJ, en la cual se encuentra alojado nuestro prototipo bloqueo los ataques realizados y dio como resultado las capturas anteriores.

FIGURA 28. Nmap firewall2
Capítulo 5. Discusiones, conclusiones y recomendaciones

Al integrar diversas herramientas tecnológicas de código abierto (como son Packet Filter, DansGuardian y OpenBSD) para la Caracterización de firewall de alta disponibilidad con filtro de contenido en un ambiente virtualizado se abarcó en gran medida el objetivo. El cual muestra al contrarrestar ataques de penetración que el firewall institucional de la UACJ bloquea los ataques hechos con Kali Linux por lo cual los ataques a puertos, concurrencia de usuarios, detección de servicios y dispositivos en una red, entre otros fueron frustrados.

Además de las características antes mencionadas, este prototipo es accesible para instituciones públicas, tanto físicamente (hardware) como económicamente ya que el equipo necesario es mínimo así como las herramientas (software) utilizado son de licenciamiento libre.

En conclusión el prototipo logró su objetivo en cierta medida ya que se realizó la integración de las tecnologías mencionadas pero no se comprobó que el prototipo este al nivel de un firewall comercial como se tenía planteado en el objetivo general de este proyecto. Una de las razones por las cuales esto no fue posible fue porque no se tuvo al alcance la manipulación de un firewall comercial para realizar las pruebas necesarias.

5.1 Con respecto a las preguntas de investigación

Las preguntas que fueron planteadas son las siguientes:

- ¿Qué características debe contemplar la solución propuesta a fin de que ofrezca la seguridad requerida en un ambiente virtualizado?

Con la integración de las herramientas previamente descritas las cuales son capaces de dar alta disponibilidad, seguridad y al ser de licenciamiento libre logra ser económico y a su vez viable, apoyándose en el ambiente virtualizado para reducir el uso de hardware y que sea eficiente.
• ¿Qué consideraciones de infraestructura se deben contemplar para un servicio de firewall en un ambiente virtualizado?
Lo que se debe contemplar para un servicio de firewall virtualizado es: un servidor, red privada y red pública, instalación de hipervisor y creación de máquinas virtuales.

• ¿Cuáles son las reglas mínimas que debe tener un firewall en un ambiente educativo?
NAT, entrada y salida de paquetes, bloqueo de puertos, bloqueo de redes no seguras, permisión de paquetes que fueron previamente autorizados.

• ¿Qué tráfico se prioriza en el firewall contemplando aplicaciones que comúnmente se usan en escuelas?

Tráfico de datos

• ¿Cómo se integra OpenBSD con Packet Filter y DansGuardian en alta disponibilidad?
Gracias a que estos son herramientas compatibles entre sí se logró la integración a través de configuraciones específicas y con el protocolo CARP se obtuvo alta disponibilidad.

• ¿Qué pruebas de penetración se van a realizar al firewall para probar su efectividad en el ambiente virtual?
Las pruebas de penetración son: denegación de servicio, ping sweep, escaneo de red para la detección de servicios que están activos, NMAP que envía paquetes masivos atacando el ancho de banda así como ataques para descifrar contraseñas.

• ¿Cómo se caracteriza el tráfico en una red de datos?
Para caracterizar el tráfico en una red de datos se utilizó Kali Linux el cual ayuda en la seguridad informática así como en auditorías.

• ¿Cómo se mide el desempeño real de un firewall virtualizado contra uno no virtualizado?
Con la ayuda de herramientas como Kali Linux, NMAP, LOIC, entre otros para realizar pruebas de penetración y con los resultados que son arrojados se puede medir el desempeño de los firewall.
5.2 Con respecto al objetivo de la investigación

Objetivo General
Integrar un prototipo de firewall con alta disponibilidad y filtro de contenido en un ambiente virtualizado que integre tecnología de seguridad perimetral a través de software libre con el fin de determinar la efectividad para bloquear firmas de ataques y filtrar contenido al nivel que lo hace un equipo de seguridad comercial.

Objetivos específicos
- Identificar los posibles factores que se presentan en las instituciones públicas, en cuanto a problemas con seguridad y red, para así realizar un prototipo que contraataque las complejidades habitualmente presentadas; de tal manera que sea un prototipo que se adecue a las necesidades de los usuarios.
- Estimar las herramientas a utilizar y se realizará un análisis para hacer uso de las herramientas más factibles en cuanto a viabilidad, compatibilidad y economía, esto último para que logre ser accesible para las instituciones públicas, tanto en costo material como operativo.
- Desarrollar un prototipo el cual sea flexible, con alta disponibilidad y efectivo contra posibles ataques, así como soporte de tráfico durante altas concurrencias.

El objetivo de este proyecto se logró en parte, es decir, se realizó la integración de las tecnologías Packet Filter, Dansguardian, CARP en un ambiente virtualizado para crear un firewall virtualizado proporcionando alta disponibilidad y al analizar las herramientas a utilizar se tomaron las mejores para este proyecto. El material utilizado es el mínimo, en éste se realizó la caracterización del firewall por lo cual es bajo el costo que se tendrá al ser implementado y gracias a su práctica configuración de reglas se adecua a las necesidades requeridas.

Por otra parte este prototipo no cumplió satisfactoriamente el objetivo propuesto ya que no se llevo a cabo la comparación de firewall virtualizado con el firewall comercial, por lo
cual no se logro determinar si el prototipo de firewall virtualizado se encuentra al nivel de uno comercial.

5.3 Recomendaciones para futuras investigaciones

Para las futuras investigaciones

- Realizar la comparación entre firewall virtualizado y firewall comercial para así determinar si está o no a su nivel.

- Ampliar el rango de reglas de filtrado usadas en este proyecto, esto con el fin de abarcar más la seguridad del firewall en la red y que cuenten con mayor grado de explicites.

- Extender el alcance del proyecto que no esté solo dirigido a redes institucionales.

- Este proyecto solo es un prototipo se sugiere que sea implementado en un entorno real.

- El proyecto puede ser llevado a nivel de hardware con la creación de un Network Appliance (dispositivo de red).
Referencias


[12] https://www.netlinux.cl/soluciones-it/alta-disponibilidad


http://www.openbsd.org/cgi-bin/man.cgi/OpenBSD-4.8/cat4/carp.0?query=carp&sec=4&manpath=OpenBSD-4%2e8

[16] OpenBSD, PFSYNC. Última modificación: April 29, 2010
http://www.openbsd.org/cgi-bin/man.cgi/OpenBSD-4.8/cat4/pfsync.0?query=pfsync&sec=4&manpath=OpenBSD-4%2e8


Anexo I. Características entre Hipervisores

<table>
<thead>
<tr>
<th>CARACTERÍSTICAS</th>
<th>VMWARE ESX / ESXI 3.5</th>
<th>MICROSOFT HYPER-V RTM</th>
<th>COMPARACIÓN HIPERVISORES</th>
</tr>
</thead>
<tbody>
<tr>
<td>Huella en el disco</td>
<td></td>
<td></td>
<td>Hyper-V es totalmente dependiente de un SO de propósito general (Windows 2008). Mejor de Microsoft huella en el disco caso es 2.6GB (Hyper-V + Server Core), que es más de 80x veces más grande que ESXi (32MB). Hyper-V + Windows Server 2008 es aún mayor con una huella de aproximadamente 10 GB. El gran tamaño de ambas opciones se crea un blanco más fácil para los ataques a la seguridad y la sobrecarga de rendimiento cada vez mayor. Además, los parches para Windows Server 2008 / Server Core afectarán a los clientes la tecnología Hyper-V. VMware ESXi en cambio es un hipervisor en pleno funcionamiento en la huella de un 32 MB de disco, lo que reduce el riesgo de tiempo de inactividad y aumenta la fiabilidad.</td>
</tr>
<tr>
<td>Pequeños</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
| Independencia del sistema operativo |                      |                      | Hyper-V depende de Windows 2008 y, en consecuencia, favorecer a los clientes de Windows cuando se trata de hacer pleno uso de los recursos de hardware disponibles:  
Hyper-V apoyo a las operativos invitados de Windows se limita sólo a Novell SUSE  
Hyper-V apoyo a la 4-VSMP formas está limitada a Windows Server 2008. Para Windows 2003, Hyper-V sólo admite un máximo de 2-vías VSMP  
VMware ESX / ESXi está optimizado para la virtualización, ofrece la apoyo más amplio OS (5 versiones de Windows, 9 versiones de Linux, NetWare y Solaris) y permite que todos los sistemas operativos de apoyo a los resultados de manera óptima utilización de los recursos disponibles (4-VSMP forma está disponible en todos los sistemas operativos de servidor compatible) |
| Conductores endurecido |                      |                      | Windows 2008 es un sistema operativo de propósito general que se basa en los controladores de dispositivos genéricos construidas por terceros que no están diseñados para la |
virtualización. Este hecho aumenta drásticamente la probabilidad de problemas de rendimiento, el tiempo de inactividad relacionados con parches de conductor, y los fallos del sistema, a pesar de que los conductores pasar por el proceso de certificación de Microsoft. Los controladores de Windows ya pasar por la certificación de Microsoft hoy en día y todavía existen problemas de controladores. Por otra parte, VMware colabora estrechamente con IHV para probar y optimizar sus drivers para su uso con ESX / ESXi para garantizar, la fiabilidad y el rendimiento general cuando se ejecuta simultáneamente las cargas de trabajo virtualizadas.

<table>
<thead>
<tr>
<th>Eficiencia de recursos</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>La gestión de memoria avanzada</strong></td>
<td>La gestión de memoria del sistema es en la mayoría de los casos el factor dominante que determina el número de máquinas virtuales simultáneas que se pueden ejecutar en una máquina física. Hyper-V no puede cometer más de la memoria de acogida porque no es compatible con las características tales como la transparencia que comparten la memoria de página y globos de memoria para recuperar la memoria física de inactividad. En consecuencia Hyper-V ofrece una menor densidad de la máquina virtual, y la proporción de bajas en comparación con la consolidación de servidores VMware ESX / ESXi. VMware ESX / ESXi apoya plenamente overcommit de memoria, lo que permite altas tasas de densidad de VM al tiempo que ofrece un rendimiento consistente. Los clientes de VMware utilizan regularmente overcommit de memoria en sus centros de datos de producción a un ritmo de 1,5 X a 2,5 veces sin ningún tipo de impacto notable a los usuarios finales.</td>
</tr>
<tr>
<td><strong>Gestión avanzada de almacenamiento</strong></td>
<td>Hyper-V utiliza el sistema de archivos NTFS para el almacenamiento. NTFS no es un sistema de archivos en clúster. Por lo tanto, Hyper-V no ofrece soporte nativo para discos virtuales compartidos y la migración en vivo. Hyper-V de la falta de un sistema de archivos en clúster significa que con la migración rápida de Microsoft (basados en la agrupación de acogida) todas las máquinas virtuales en un LUN se migrará al mismo tiempo. Para evitar este resultado no deseado, el usuario debe asignar una VM por LUN, que crea una pesadilla de almacenamiento. La única otra opción es la compra de un tercero en el sistema de archivos en cluster de concesión de licencias</td>
</tr>
</tbody>
</table>
adicionales y gastos de apoyo. VMware ESX / ESXi viene con VMFS, un sistema de archivos en clúster diseñado específicamente para la virtualización. VMFS apoya plenamente la migración en vivo, y permite a varias máquinas virtuales para compartir un único LUN y todavía emigran / conmutación por error de máquinas virtuales individuales.

**Gestión avanzada de redes**

Hyper-V no es compatible con NIC física nativo trabajo en equipo, en lugar que requiere de 3 controladores de hardware independientes (Intel, Broadcom). Esta limitación impide que Hyper-V de proporcionar fuera de la protección de la caja, independiente del hardware de un fallo de la NIC. Además, la tecnología Hyper-V para los conductores NIC apoyo no se virtualización consciente, no pueden ser controlados desde el hipervisor, y se deben cargar en la partición de los padres.VMware ESX / ESXi apoya plenamente NIC trabajo en equipo, resultando en una sencilla, completa de la solución de cuadro de conmutación por error de la NIC.

**Rendimiento**

**Escalabilidad I / O**

Hyper-V utiliza un “controlador indirecto” modelo que mantiene todos los controladores de dispositivo en una instancia de Windows 2008 almacenada en la partición de los padres de Hyper-V y se basa en el sistema operativo de propósito general para manejar la E / S de tráfico. Este diseño de los cuellos de botella de E / S, como todo lo que resultados peticiones E / S debe viajar a través de Windows Server 2008 (o Server Core), que no está optimizado para la virtualización. El resultado es problemas de escalabilidad cuando se ejecuta muchas máquinas virtuales bajo carga. Este problema se limita la escalabilidad ratios de consolidación.VMware ESX / ESXi emplea un modelo de controlador directo con un planificador construyó específicamente para manejar las demandas de múltiples cargas de trabajo de alto. Esto da lugar a la escalabilidad de las relaciones más altas, mientras que la densidad de VM con la entrega consistente y de alto rendimiento.

**Asignación de Recursos Distribuidos**

Hyper-V carece de muchas capacidades para distribuir de manera óptima los recursos de hardware., Con base en tiempo real las condiciones de carga de trabajo. A diferencia de ESX / ESXi, Hyper-V no es compatible con:
| Apoyo para las Tecnologías de Mejora del Rendimiento | Hyper-V carece de apoyo de varias tecnologías que mejoran el rendimiento, tales como:  
AMD RVI (tablas anidadas página), incluido en la última versión de la CPU AMD Barcelona  
Páginas de memoria grande, que permite mejorar el rendimiento cuando la virtualización de bases de datos  
VMware ESX / ESXi es compatible con las tablas de páginas y páginas anidadas grandes cantidades de memoria y por lo tanto capaz de un rendimiento superior, incluso en los entornos más exigentes. Aplicaciones de misión crítica e intensivos, como una base de datos de Oracle se desempeñan mejor en un entorno virtual de VMware. |
<p>| Soporte Linux virtualizado | Hyper-V es compatible con la “vieja manera” de hacer Linux para la virtualización, donde los administradores mantienen dos núcleos, uno para la física y otro para las máquinas virtuales. Este adicionales de mantenimiento mayor complejidad y los costes de gestión.Por otra parte, VMware ESX / ESXi apoya para Linux-a través de la virtualización paravirt_ops / VMI (Virtual Machine Interface) – la nueva industria aceptado forma de para-virtualizar Linux. Esta nueva forma no requiere de los administradores de TI para mantener y apoyar los núcleos separados para máquinas físicas y virtuales – es todo en un solo núcleo. En la actualidad, SLES 10, y Ubuntu 7 todo paravirt- |</p>
<table>
<thead>
<tr>
<th>Características Avanzadas</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>La tecnología de seguridad virtual de</strong></td>
<td><strong>Hyper-V no ofrece dedicada capacidades de los proveedores de seguridad para ampliar sus capacidades de seguridad para entornos virtualizados. VMsafe</strong>, Una característica de las próximas VMware ESX / ESXi, permitirá un rico ecosistema de soluciones de seguridad de terceras partes para los entornos virtualizados. VMsafe es una tecnología de seguridad virtual que proporciona una visibilidad de grano fino de los recursos de máquina virtual, que permite supervisar todos los aspectos de la ejecución del sistema y detener los virus antes indetectable, rootkits y malware antes de que se puede infectar un sistema.</td>
</tr>
</tbody>
</table>
Propuesta del tema para la materia de Seminario de Titulación I

| Nombre: Alejandra Torres / Lucero Martínez | Matrícula: 98709 / 98750 |
| Programa Académico: Ingeniería en Sistemas Computacionales |
| Departamento: Ingeniería Eléctrica y Computación |

Título: Caracterización de firewall de alta disponibilidad con filtro de contenido en un ambiente virtualizado.

Contextualización:

La seguridad en cómputo es un área crítica en las redes de datos, normalmente se deben establecer controles de seguridad física y lógica para mitigar amenazas y/o ataques a los activos de las organizaciones.

Algunos controles de seguridad incluyen sistemas antivirus, sistemas de prevención contra intrusos, filtros antispam y sistemas corta fuegos o mejor conocidos como firewalls. La integración de estos sistemas o controles depende de un análisis de riesgos que ayuda a determinar el tipo de control y la inversión que se está dispuesta a realizar.

Los firewall son dispositivos de hardware o software para controlar comunicaciones, permitiéndoles o prohibiéndoles accesos según haya sido especificado. Éste se ubica en el punto de conexión de red interna con la red exterior (Internet) de este modo la red se protege de accesos no autorizados.

La virtualización es una de las últimas tendencias en la industria, ya que las organizaciones tienen por objeto aumentar la utilidad, la flexibilidad y la rentabilidad de sus recursos informáticos.

Al integrar un firewall en un ambiente virtualizado se obtienen beneficios de escalabilidad, disponibilidad y aprovechamiento máximo de recursos. Lo que conlleva a lograr una tendencia para servicios integrados en una sola caja o centros de datos auto-contenidos.

A través de los años la tecnología ha ido integrando los servicios en un ambiente virtual, dichos servicios incluyen aquellos con infraestructura de red como switches, routers y firewalls, aunque estos últimos no se tienen en una modalidad amplia y a bajo costo.
Definición del Problema:

Existe una necesidad de tener opciones efectivas virtualizadas para reforzar la seguridad al realizar conexiones dentro de la red interna de una institución pública hacia Internet.

Para este tipo de instituciones en donde se tienen miles de usuarios concurrentes accediendo a Internet, se requiere una solución muy robusta cuyo gasto de inversión y gasto operativo puede ser muy alto, y donde inversiones en tecnologías ya realizados (hardware), se pueden aprovechar en ambientes virtualizados.

Objetivo:

Integrar un prototipo de firewall con alta disponibilidad y filtro de contenido en un ambiente virtualizado que integre tecnología de seguridad perimetral a través de software libre con el fin de determinar la efectividad para bloquear firmas de ataques y filtrar contenido al nivel que lo hace un equipo de seguridad comercial.

Preguntas de Investigación:

¿Qué características debe contemplar la solución propuesta a fin de que ofrezca la seguridad requerida en un ambiente virtualizado?
¿Qué consideraciones de infraestructura se deben contemplar para un servicio de firewall en un ambiente virtualizado?
¿Cuáles son las reglas mínimas que debe tener un firewall en un ambiente educativo?
¿Qué tráfico se prioriza en el firewall contemplando aplicaciones que comúnmente se usan en escuelas?
¿Cómo se integra open BSD con packet filter y DansGuardian en alta disponibilidad?
¿Qué pruebas de penetración se van a realizar al firewall para probar su efectividad en el ambiente virtual?
¿Cómo se caracteriza el tráfico en una red de datos?
¿Cómo se mide el desempeño real de un firewall virtualizado contra uno no virtualizado?

Justificación:

Últimamente se ha venido aprovechando la consolidación de infraestructura en ambientes virtualizados, y el hacer uso de esta misma para integrar un servicio de seguridad puede representar un ahorro significativo para alguna organización.

Cada servicio requerido para aumentar la seguridad y la alta disponibilidad en la red
que alguna institución ofrece suele tener un costo excesivo, sin mencionar que la complejidad y a su vez, los costos, aumentan al tener estos controles de forma individual. Por esta razón se pretende utilizar varias herramientas que al integrarse tendrán el propósito de brindar un servicio eficiente y con menor costo.

Debido a los miles de usuarios que a diario hacen uso del servicio, es necesario contar con un control de contenido a nivel red. Esto puede aplicar principalmente en escuelas de nivel básico y medio superior, que se pretenden conectar a través del programa México Conectado, del gobierno federal.

Los beneficios que se obtendrán a partir del prototipo serán: **firewall** de estados, control de contenido, control de tráfico de datos, control de accesos no autorizados y garantizar una alta disponibilidad en el servicio utilizando recursos de virtualización.

**Solución Propuesta:**

Se integrará un prototipo con software libre que soporte la funcionalidad de un **firewall** virtualizado, con el cual se realizará una caracterización del ambiente que simulará el tráfico habitual de datos en una institución pública. Dicho prototipo será capaz de normalizar, acondicionar tráfico y proveer un control de ancho de banda para las comunicaciones por medio de un filtro de contenido **Packet Filter** y **DansGuardian**, además contará con un servicio de alta disponibilidad por medio del protocolo **CARP** que se integra con **OpenBSD**.

**Metodología Propuesta:**

- Analizar los servicios de red comúnmente utilizados en las instituciones públicas, así como los problemas de tráfico que se presentan, e investigar acerca de las herramientas que se utilizarán (más viables, económicas y su compatibilidad).

- Realizar un análisis y determinar la estructura y diseño del prototipo a crear, y se procederá a reclutar los materiales necesarios para lograr el objetivo deseado, debiendo tener como mínimo dos máquinas virtuales.

- Integrar la infraestructura para el ambiente de virtualización.

- Integrar la funcionalidad del **firewall** el cual será capaz de satisfacer las necesidades antes mencionadas, dentro de un entorno que estará caracterizando el mismo servicio de red que ofrecen las instituciones públicas.

- Integrar la funcionalidad de alta disponibilidad y filtro de contenido.
- Determinar el conjunto de pruebas de desempeño y penetración que se van a realizar al sistema de *firewall*.

- Realizar las pruebas necesarias dentro de un ambiente controlado, tales como tráfico de datos, posibles ataques, alta concurrencia de acceso, etc., con el fin de simular el ambiente habitual de la red.

- Cotejar resultados con los que ofrece un equipo de *firewall* comercial.

<table>
<thead>
<tr>
<th>Observaciones del evaluador:</th>
</tr>
</thead>
</table>

<table>
<thead>
<tr>
<th>Aprobado</th>
<th>Aprobado con condición</th>
<th>Rechazado</th>
</tr>
</thead>
</table>

**Fecha de terminación del proyecto:** Mayo 2015

**Nombre del asesor responsable:**

Eduardo Castillo Luna

---

Alumno

Evaluador

Evaluador

Maestro de la materia

Asesor