UNIVERSIDAD CENTRAL "MARTA ABREU" DE LAS VILLAS

FACULTAD DE INGENIERIA ELECTRICA,

DEPARTAMENTO DE ELECTRONICA Y TELECOMUNICACIONES

"Propuesta de solución de seguridad para la División de Desoft Villa Clara"

Tesis presentada en opción al Título Académico de Master en Telemática

Maestría de Telemática

Autor: Ing. Alberto Rodríguez Carvajal.

Tutor: Dr. C. Vitalio Alfonso Reguera

Consultante: MSc. Manuel Castro Artiles

Santa Clara, Cuba, 2016

UNIVERSIDAD CENTRAL "MARTA ABREU" DE LAS VILLAS

FACULTAD DE INGENIERIA ELECTRICA,

DEPARTAMENTO DE ELECTRONICA Y TELECOMUNICACIONES

"Propuesta de solución de seguridad para la División de Desoft Villa Clara"

Tesis presentada en opción al Título Académico de Master en Telemática

Maestría de Telemática

Autor: Ing. Alberto Rodríguez Carvajal.

Tutor: Dr. C. Vitalio Alfonso Reguera

Consultante: MSc. Manuel Castro Artiles

Santa Clara, Cuba, 2016

PENSAMIENTO

“Si conoces a los demás y te conoces a ti mismo, ni en cien batallas correrás peligro; si no conoces a los demás, pero te conoces a ti mismo, perderás una batalla y ganarás otra; si no conoces a los demás ni te conoces a ti mismo, correrás peligro en cada batalla.”

Sun Tzu, El Arte de la Guerra

AGRADECIMIENTOS

Deseo expresar mi más profundo agradecimiento a todas las personas que hicieron posible que llegara hasta aquí, en especial:

A mi papá que tanto me apoyo en vida cuando se encontraba entre nosotros.

A mi mamá que sin su apoyo incondicional hubiera sido imposible llegar hasta aquí.

A mi amigo y profesor Manuel Castro Artiles por sus tantas enseñanzas y apoyo.

Al profesor Vitalio Alfonso Reguera por sus enseñanzas a lo largo de mi carrera, la maestría y apoyo como tutor de la tesis.

A todos mis profesores del Departamento de Telecomunicaciones y Electrónica de la facultad de Eléctrica por sus enseñanzas y confianza en mí.

A todos mis hermanos de las Artes Marciales de Cuba y un poco más allá.

A mis compañeros de trabajo a Hiram, a Rebeca, a Mirtha, a Concha y a todos por su apoyo.

A todos muchas gracias

DEDICATORIA

Dedicado a la memoria de mi papá

RESUMEN

El presente trabajo tiene como objetivo principal una propuesta de sistema de seguridad aplicada al entorno de la infraestructura de red y servicios informáticos de la División de Desoft Villa Clara. Para ello se parte de los principios determinantes de la seguridad informática aplicada a redes de datos y entornos de virtualización. Además se realiza un estudio de la multiplataforma de virtualización Proxmox.

Se realiza una caracterización de los entornos de red de la División, haciéndose énfasis en las debilidades infraestructurales. Se identifican posibilidades para un mejoramiento del sistema de seguridad y se hace un levantamiento de la tecnología de hardware existente en la División recomendándose la más idónea para el entorno de virtualización; proyectándose hacia un crecimiento futuro, teniendo en cuenta las prestaciones que brinda esta Empresa a sus usuarios y clientes.

Por último se formaliza una propuesta de mejoras para el sistema de seguridad con que cuenta la Empresa actualmente, la arquitectura de red, los servicios (desde las perspectivas del lado del usuario); proponiéndose un grupo de herramientas para la, supervisión de servicios, detección de intrusos, análisis de tráfico y gestión que garantizan la confidencialidad de los paquetes que viajan a través de la red de datos. Se tendrán en cuenta las herramientas de software libre, previamente probadas sobre máquinas virtuales.

Palabras clave: Supervisión de los servicios, herramientas, vulnerabilidades, seguridad, virtualización

ABSTRACT

This paper's main objective is to propose a security system applied to the environment of network infrastructure and IT services at the Division Desoft Villa Clara. The starting point gegins with the determining principles of information security applied to data networks and virtualization environments. In addition, a study of cross-platform virtualization Proxmox is performed with a characterization of network environments, doing emphasis on infrastructural weaknesses and. identifying possibilities for improvement of the security system and a survey of existing hardware technology to recommend the most suitable for virtualization environment projecting toward future growth, taking into account the benefits provided by this company to its users and customers.
Finally a proposal for improvement security system currently available to the company, network architecture, services (from the perspective of the user side) is formalized; proposing a set of tools, service monitoring, intrusion detection, traffic analysis and management to ensure confidentiality of packets traveling through the data network. Free software tools, pretested on virtual machines are taken into account.

Keywords: Monitoring services, tools, vulnerabilities, security, virtualization

ÍNDICE

INTRODUCCIÓN

Desde la invención de los ordenadores y de las tecnologías de la información hasta el presente, su desarrollo tecnológico ha ido creciendo cada año de forma vertiginosa. Nunca antes en la historia una tecnología completamente nueva se ha propagado por todo el mundo con tal velocidad y con tan gran penetración de prácticamente todas las actividades humanas. [1]

En 1969 nació ARPANET con el objetivo de desarrollar protocolos de comunicación que permitiesen la comunicación entre las redes conectadas (a través de radio y satélite, y por otros medios). El ARPANET era una red experimental diseñada para apoyar la investigación militar; en concreto, investigaban sobre cómo construir redes que pudiesen resistir desastres parciales (como ataques de mísiles) y seguir funcionando. La ARPANET original se convirtió en Internet. Internet se basó en la idea de que habría múltiples redes independientes con un diseño bastante arbitrario, empezando por ARPANET como red pionera de conmutación de paquetes, pero que pronto incluiría redes satelitales, redes terrestres de radio y otras redes. Internet tal y como la conocemos hoy en día plasma una idea técnica subyacente fundamental, que es la de red de arquitectura abierta. [2]

El desarrollo de Internet ha creado un ambiente en el que millones de ordenadores en todo el mundo están todos conectados el uno al otro. Entre 2000 y 2015 la penetración de Internet se ha multiplicado casi por siete, pasando de 6,5 al 43 por ciento de la población mundial. La proporción de hogares con acceso a Internet aumentó del 18 por ciento en 2005 al 46 por ciento en 2015[3]. Además, el acceso a esta red es bastante ubicuo y barato, lo que permite a cualquier ladrón en el mundo atacar su equipo, independientemente de su ubicación física. Las computadoras personales son ahora también baratas. Los atacantes pueden fácilmente adquirir y establecer en varios ordenadores diferentes sistemas operativos y realizar la búsqueda de vulnerabilidades explotables. La búsqueda de vulnerabilidades en los sistemas que controlan permite a los atacantes refinar su código de explotación antes de usarlo en los sistemas actuales. Después de encontrar una nueva vulnerabilidad y desarrollar un exploit, pueden atacar sistemas similares en todo el mundo. [4]

En la medida que las empresas, universidades, centros de investigación y demás instituciones se han hecho más dependientes de las TIC, surge la necesidad de verificar que estas funcionan correctamente. La seguridad de las redes de transmisión de datos constituye una de las direcciones de investigación más importantes en la actualidad a nivel mundial. El creciente desarrollo dado por los avances en diversas ramas como la electrónica, la automática, las telecomunicaciones y la informática, así también como la utilización de nuevos medios de transmisión han diversificado el uso de estas y convertido en medio esencial para el correcto funcionamiento y avance de las sociedad moderna.[5]

Los sistemas de gestión de la seguridad que existen en las diferentes organizaciones no son lo suficientemente potentes y eficientes. Muchas veces debido a la falta de planificación por parte de la dirección de estas organizaciones para el desarrollo e implementación de programas estratégicos de seguridad que protejan y garanticen las TIC frente a las amenazas del entorno actual. [5]

Existen diferentes herramientas para diagnosticar el estado de un entorno informático para así sugerir soluciones integrales. Estas soluciones proporcionarían métodos seguros a acciones cotidianas inseguras, es decir, implicarían a los protagonistas del proceso informático que son, desde los usuarios normales hasta los usuarios más avanzados con niveles de responsabilidad alto y privilegios dentro del entorno de red, en buenas prácticas de manejo y gestión de las TIC.[6]

En los últimos años la seguridad informática en Cuba ha ido tomando un rol protagónico de gran importancia, dado que nuestro país no queda exento de todas las amenazas existentes en el mundo enfocadas a dañar las TIC. Cuba es un país en desarrollo económico que se encuentra afectado por el bloqueo económico impuesto por EEUU. Actualmente existen muchos sitios en Internet que no pueden ser accedidos desde nuestro país, debido a que se encuentran bloqueados solo para Cuba y en ocasiones la obtención de software, libros, manuales, etc. se hace muy dificultosa. Es por eso, que en ocasiones se hace necesaria la utilización de vías alternativas para la obtención de este tipo de materiales, por lo que cualquier pérdida de esta información que pueda ser provocada debido a un delito informático, contaminación por virus o desastre natural provocaría pérdidas considerables a la economía de cualquier empresa en particular y por lo tanto al país.

La División Desoft Villa Clara tiene como misión fundamental la informatización de la sociedad cubana. En esta empresa se maneja información de carácter sensible respecto a los proyectos de carácter nacionales que se desarrollan en la entidad. La ocurrencia de cualquier desastre natural o de otra naturaleza, por ejemplo contaminación por virus, provocaría la interrupción del flujo de trabajo en el sistema informático. Es por eso proponer un sistema de seguridad que proponga un grupo de soluciones de seguridad para mejorar la seguridad existente en la División de Desoft Villa Clara, utilizando las grandes ventas que brindan las herramientas del software libre, es la es la finalidad de este trabajo.

Planteamiento del problema

En la actualidad la confidencialidad, la integridad y la disponibilidad de la información existente en la empresa de Desoft Villa Clara son aspectos de vital importancia para esta empresa, debido a que en ella se maneja información sensible y confidencial perteneciente a proyectos de carácter nacional, y la ocurrencia de un desastre podría ocasionar perdidas de información así como la interrupción temporal de la continuidad de los procesos informáticos que allí se desarrollan.

Objetivo general

Diseñar un sistema que mejore la seguridad actual en la empresa Desoft Villa Clara, proponiendo para ello tecnologías de hardware y empleando herramientas de software libre para la gestión, control y supervisión de la seguridad en la red de computadoras.

Objetivos específicos

- Definir los requisitos necesarios que debe cumplir un sistema de seguridad para el ámbito empresarial.
- Seleccionar una tecnología de hardware que cumpla con las necesidades de la empresa.
- Seleccionar herramientas de software libre que permitan mejorar la gestión, el funcionamiento y seguridad de la red de área local.
- Realizar el diseño y modelado del Sistema de Seguridad.
- Implementar con máquinas virtuales herramientas de gestión y supervisión que permitan comprobar la seguridad de los paquetes que viajan a través de la red.

Estructura de la Tesis

Este trabajo está estructurado en tres capítulos:

Capítulo I: Se definirán los requisitos necesarios que debe cumplir un sistema de seguridad para el ámbito empresarial. Además, se desarrollará un estudio de las principales tecnologías de hardware y herramientas de software libre que permitan mejorar la gestión, el funcionamiento y seguridad de la red de área local.

Capítulo II: Consiste en seleccionar la tecnología de hardware que cumpla con las necesidades de la empresa y las herramientas de software libre que se ajusten al entorno de red de la misma. Además se realizará el diseño y modelado del Sistema de Seguridad empleando, para ello las herramientas seleccionadas.

Capítulo III: Se desarrollará la implementación con máquinas virtuales de herramientas de gestión y supervisión que permitan comprobar la seguridad de los paquetes que viajan a través de la red.

CAPITULO I: FUNDAMENTOS TEÓRICOS SOBRE LA SEGURIDAD EN REDES PARA AMBITO EMPRESARIAL

En este capítulo se hace un análisis sobre los principios fundamentales de la seguridad en redes de datos, así como el uso de las herramientas de software libre más utilizadas en auditorias informáticas y las tecnologías de hardware para la virtualización.

1.1 Definición de seguridad de la información

El diseño de un sistema de seguridad depende de cómo se lleven a la práctica los servicios fundamentales de identificación, autorización, autenticación, no repudio, integridad y confidencialidad.

Existen muchos autores que han definido el gran valor e importancia que tienen para las organizaciones el resguardo de la información; en [Nist95] se define:

"La protección de un sistema de información automático consiste en aplicar en orden los objetivos de preservar la integridad, accesibilidad, y confidencialidad de los recursos del sistema de información (estos pueden incluir hardware, software, firmware, información, datos, y telecomunicaciones)."

Esta definición introduce tres objetivos principales, que son los conceptos más importantes de seguridad en los sistemas de computación.

Confidencialidad: este término abarca dos conceptos:
Confidencialidad de los datos: asume que la información privada o confidencial no debe ser hecha accesible o visible de forma individual a personas no autorizadas.
Privacidad: Asume que el control individual o influencia que la información puede brindar es adquirida o almacenada por quien o quienes estén autorizados a revelar esa información.
Integridad: Este término asume dos conceptos relacionados:
Integridad de los datos: Asume que la información o programas son cambiados solamente de una manera específica y autorizada.
Integridad del sistema: Asume que un sistema representado en su funcionamiento de un modo determinado, debe ser libre de manipulación deliberada e inadvertida y no autorizada del sistema.
Accesibilidad: Asume que el sistema trabaja correctamente y los servicios no son denegados a los usuarios autorizados.

Estas tres formas de conceptos son frecuentemente referidas como *CIA Triad (Anexo 1).* Estos tres conceptos envuelven los objetivos fundamentales de la seguridad para ambos, datos e información y servicios del sistema de computación. Por ejemplo, el *NIST Standards for Security Categorization of Federal Information and Information Systems (FIPS 199)* lista la confidencialidad, la integridad y la accesibilidad como los tres objetivos principales de seguridad para la información o sistemas de información. La FIPS 199 brinda una útil caracterización de estos tres objetivos en términos de requerimientos y la definición de una perdida de seguridad en cada categoría.

Confidencialidad: Preserva restricciones autorizadas en el acceso y revelación de la información, incluye medios de protección personal a la privacidad y al propietario de la

información. Una pérdida de confidencialidad es la no autorizada accesibilidad a la información.
Integridad: Resguarda de modificaciones inapropiadas o destrucción de la información, incluye aseguramiento de la información, no repudio y autenticidad. Una pérdida de integridad es la no autorizada modificación o destrucción de la información.
Accesibilidad: Asegura a tiempo un fiable acceso y uso de la información. Una pérdida de accesibilidad es el rompimiento del acceso o el uso de la información, o un sistema de información.

Aunque el uso de la *CIA Triad* define estos objetivos de seguridad como mejores establecido, en algunos campos de seguridad son necesarios otros conceptos adicionales, que no vienen representados en la figura anterior. Dos de los conceptos más comúnmente mencionados son:

Autenticidad: La propiedad de inicio genuina debe verificarse que sea fiable, y confidencial en la validez de transmisión, de un mensaje, o mensaje originado. Esto significa verificar que los usuarios son quienes dicen ser ellos que son y que cada acceso al sistema es un acceso de la fuente real.

Identificación: Es la meta de la seguridad que genera los requerimiento para las acciones de una entidad al ser trazadas singularmente en esta entidad. Este apoya al no repudio, defecto de aislamiento, detección de instrucción y prevención, para posteriormente apoyar las acciones de recuperación y acción legal. Porque realmente un sistema seguro no es una meta factible, se debe ser capaz de trazar una brecha de seguridad al responsable del identificador. El sistema debe contar con un registro de sus actividades para permitir posteriormente un análisis forense de las trazas y determinar las brecha de seguridad en el sistema. [14]

La autenticación, autorización y no repudio son elementos dados al diseñar un sistema de seguridad, pueden usarse al mantener la seguridad del sistema con respecto a la confidencialidad, integridad y accesibilidad. Conociendo cada uno de estos seis conceptos y como se relacionan cada uno con otro ayudan a implementar un diseño profesional y seguro en un determinado sistema de seguridad. Cada componente es crítico en la seguridad total, con el fallo de uno de estos componentes puede resultar comprometida la seguridad de cualquier sistema. [15]

1.1.1 Normas y estándares relacionados con la seguridad en redes de datos

Las WLAN están basadas en el estándar IEEE *(Institute of Electrical and Electronics Engineers)* 802.11x, que fue el primero desarrollado por IEEE en el año 1997. Describe las normas a seguir por cualquier fabricante de dispositivos Wireless para que puedan ser compatibles entre sí. Fue diseñado para soportar un alcance medio, aplicaciones con tasas de datos más altas, tales como redes Ethernet, y para direccional estaciones móviles y portátiles. El estándar abarca los protocolos 802.1x, TKIP (*Protocolo de Claves Integra – Seguras – Temporales*) y AES. Se implementa en WPA (*Wi-Fi Protected Access*).

La ISO/IEC 27000, es un conjunto de estándares desarrollados -o en fase de desarrollo por ISO *(International Organization for Standardization)* e IEC *(Internacional Electrotechnical Commission)*, que proporcionan un marco de gestión de la seguridad de la información utilizable por cualquier tipo de organización, pública o privada, grande o pequeña.

El profesor Edward Humphreys, coordinador del grupo de trabajo responsable de los sistemas de gestión de seguridad de la información (SGSI) de ISO, subraya: *"Para garantizar la seguridad en el panorama digital actual, todas las organizaciones, sea cual sea su tamaño, deben establecer un marco de trabajo de gestión como punto de partida para hacer frente a los riesgos cibernéticos. La norma ISO/IEC 27001 se diseñó para ayudar a las organizaciones a hacerlo. La norma es el 'lenguaje común' del mundo cuando se trata de evaluar, tratar y gestionar los riesgos relacionados con la información".*

A continuación, se presentan las últimas revisiones y adiciones de la serie ISO/IEC 27000 —todas publicadas en 2015—, que forman parte de la "caja de herramientas de riesgos cibernéticos" ISO/IEC 27001, para ayudar a mantener estos riesgos bajo control.

Protección de la información en la nube (ISO/IEC 27017)

Se acaba de publicar un nuevo código de prácticas para los controles de seguridad de la información de los servicios en la nube, la norma ISO/IEC 27017. La nube es una de las innovaciones más utilizadas en el frenético mundo actual del comercio y los negocios. A medida que el servicio gana adeptos, los usuarios están exigiendo garantías de que los datos almacenados y procesados en la nube están seguros. Por su propia naturaleza, el mercado de los servicios en la nube es global, con proveedores repartidos por amplias zonas geográficas, y los datos se transfieren de forma rutinaria a través de fronteras nacionales. Por ello, contar con unas directrices internacionales es esencial.

Según Satoru Yamasaki, uno de los revisores que trabajaron en la norma, "ISO/IEC 27017 ayudará a los proveedores de servicios a llegar a un entendimiento común con sus clientes con respecto a los controles de seguridad adecuados y la forma de implementarlos. Esta norma internacional para los controles de seguridad en la nube facilitará el desarrollo y la expansión de unos sistemas de computación en la nube seguros".

Las nuevas directrices son el resultado de una iniciativa conjunta de los principales desarrolladores del mundo de las normas internacionales —IEC, ISO e ITU— para garantizar la máxima difusión.

Soluciones integradas para servicios (ISO/IEC 27013)

Cada vez hay más organizaciones que están optando por combinar un sistema de gestión de la seguridad de la información (ISO/IEC 27001) con un sistema de gestión de servicios (ISO/IEC 20000-1). Un sistema integrado implica que una organización puede gestionar de manera eficiente la calidad de sus servicios, procesar los comentarios de los clientes y resolver problemas garantizando a la vez la seguridad de la información.

ISO/IEC 27013 ofrece una estrategia sistemática para facilitar la integración de un sistema de gestión de la seguridad de la información con un sistema de gestión de servicios, lo que se traduce en menores costos de implementación y evita la duplicación de esfuerzos, ya que solo se necesita una auditoría, en lugar de dos, para obtener la certificación.

Comunicaciones intersectoriales e interorganizacionales (ISO/IEC 27010)

Cuando una organización comparte información con otra, ¿cómo pueden ambas tener la certeza de que sus datos estarán seguros? ISO/IEC 27010 es una adición sectorial a la caja de herramientas de ISO/IEC 27000, que ofrece directrices para la iniciación, la implementación, el mantenimiento y la mejora de la seguridad de la información en las comunicaciones interorganizacionales e intersectoriales. Incluye principios generales sobre la manera de cumplir con estos requisitos utilizando la mensajería establecida y otros métodos técnicos. Se espera que la norma fomente el crecimiento de las comunidades globales de intercambio de información.

Como el Dr. Mike Nash, uno de los revisores de la norma ISO/IEC 27010, señala, "la norma ISO/IEC 27010 básicamente personaliza y aplica las normas ISO/IEC 27001 e ISO/IEC 27002 a la comunicación entre organizaciones. Contar con esta norma proporciona a una organización la tranquilidad de que la información que ha compartido con otra no será revelada por accidente". Esta norma es particularmente relevante para la protección de la infraestructura nacional crítica, en la que el intercambio seguro de información confidencial es esencial. Su uso está también extendido entre los equipos de respuesta a incidentes de seguridad.

Detectar y prevenir ciberataques (ISO/IEC 27039)

¿Cómo pueden las organizaciones detectar y prevenir las intrusiones cibernéticas en sus redes, sistemas y aplicaciones? Las prácticas recomendadas muestran que tienen que ser capaces de saber cuándo y cómo se produce una intrusión en su red, sistema o aplicación. También deben estar preparadas para identificar la vulnerabilidad que se ha aprovechado y los controles que se deben implementar para prevenir intrusiones similares en el futuro. Una forma de hacerlo es usar un sistema de detección y prevención de intrusiones (SDPI).

ISO/IEC 27039 proporciona directrices para preparar e implementar un SDPI y cubre aspectos tan cruciales como la selección, la implementación y el uso. La norma resulta especialmente útil en el mercado actual, donde hay muchos productos y servicios de SDPI de código abierto y comercialmente disponibles basados en diferentes tecnologías y estrategias. La norma ISO/IEC 27039 guía a las organizaciones en todo el proceso.

Auditoria y certificación (ISO/IEC 27006)

Cada vez más organizaciones están recurriendo a las auditorias de certificación de terceros para demostrar que cuentan con un sistema sólido de gestión de la seguridad de la información (SGSI) que se ajusta a los requisitos de la norma ISO/IEC 27001. La norma ISO/IEC 27006 establece los requisitos que los organismos de certificación y registro deben cumplir para ser acreditados y poder ofrecer servicios de certificación con la norma ISO/IEC 27001.

"ISO/IEC 27006 es una acreditación de referencia para los organismos de certificación que ofrecen servicios relacionados con la norma ISO/IEC 27001", explicó el profesor Humphreys, que añadió: "Esto es importante porque la acreditación de organismos de

certificación proporciona confianza adicional en el proceso de auditoría y credibilidad en el certificado que otorgan". [16]

La ISO la precede una historia de evolución a través de los años (ver Figura 1.1)

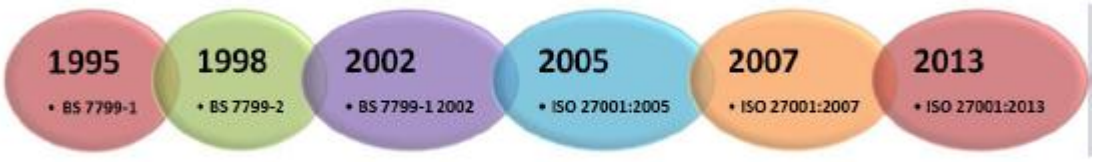


Figura 1.1: Historia y evolución de ISO 27001

Un *SGSI (Sistema de Gestión de Seguridad de la Información*) proporciona un modelo para establecer, implementar, operar, monitorear, revisar, mantener y mejorar la protección de los activos de información para lograr objetivos de negocio.

El análisis de los requisitos para la protección de los activos de información y la aplicación de controles adecuados para garantizar la protección de estos activos de información, contribuye a la exitosa implementación de un SGSI.

El *Sistema de Gestión de la Seguridad de la Información (SGSI)* en las empresas ayuda a establecer estas políticas, procedimientos y controles en relación a los objetivos de negocio de la organización.

La certificación de un SGSI es un proceso en el que una entidad de certificación externa, independiente y acreditada audita el sistema, determinando su conformidad con ISO/IEC 27001, su grado de implantación real y su eficacia y en caso positivo, emite el correspondiente certificado. El proceso de gestión de seguridad de la información puede ser observado en el Anexo 2:

El círculo de DEMING se constituye como una de las principales herramientas para lograr la mejora continua en las organizaciones o empresas que desean aplicar a la excelencia en sistemas de gestión.

El conocido Ciclo Deming o también se le denomina el ciclo PHVA que quiere decir según las iniciales (planear, hacer, verificar y actuar) o ingles PDCA (Plan, Do, Check, Act). [Ver Anexo 2]

1.1.2 Vulnerabilidad en redes de datos

Una vulnerabilidad de seguridad es un defecto o debilidad en el diseño, implementación o funcionamiento de un sistema que podría ser utilizado para violar su seguridad. Una vulnerabilidad de seguridad no es un riesgo, amenaza o ataque.

También puede ser definida una Vulnerabilidad como una debilidad de un activo o control que puede ser explotada por una o más amenazas. [ISO/IEC 27000].

Una vulnerabilidad de la seguridad es una imperfección o debilidad que va ser explotada o violada en un sistema, o de la información que contiene este último. Si la vulnerabilidad existe, entonces es posible para una amenaza realizarse satisfactoriamente a menos que se tomada una media.

Las recomendaciones de la UIT-T reconocen cuatro tipos de vulnerabilidades:

1. Modelo amenazas de vulnerabilidades, él es el resultado del fallo al prever posibles futuras amenazas.
2. El diseño y especificación de vulnerabilidades, el cual es resultado de errores o descuidos en el diseño de un sistema o protocolo y se hace inherentemente vulnerable.
3. Implementación de vulnerabilidades, el cual es introducido por errores o descuidos durante la implementación del sistema o protocolo.
4. Operación y configuración de vulnerabilidades, el cual se origina del inapropiado uso de las opciones en la implementación o despliegue débil de políticas y prácticas (semejantes a fallos o uso de encriptación en una red inalámbrica).[18]

1.2 La arquitectura de seguridad OSI

El valorar efectivamente la seguridad necesaria de una organización y al evaluar varios productos y políticas de seguridad, el responsable de la administración para la seguridad necesita de algunas vías sistemáticas de definir los requerimientos para la seguridad y caracterizar el aprovechamiento al satisfacer estos requerimientos.

La *ITU-T3 Recommendation X.800, Security Architecture for OSI*, define semejante sistema de aprovechamiento. El OSI arquitectura segura es útil al administrar como una vía de organización las tareas de proveer la seguridad. Además, esta arquitectura fue desarrollada como un estándar internacional, los vendedores de computadoras y comunicaciones tienen que desarrollar los requerimientos de seguridad para sus productos y servicios que son relacionadas a la definición estructural de servicios y mecanismos.

La seguridad de la arquitectura OSI, es enfocado en los ataques, mecanismos y servicios de seguridad. Estos pueden ser definidos:

- *Ataques de seguridad*: Cualquier acción que comprometa la seguridad de la información.
- *Mecanismos de seguridad:* Un mecanismo que es diseñado para detectar, prevenir o recobrarse de los ataques de seguridad.
- *Servicios de seguridad:* Un servicio que mejora la seguridad de los sistemas de procesamiento de datos y de la transferencia de la información. Tiene en cuenta los ataques de seguridad y hace uso de uno o varios mecanismos de seguridad.

1.3 Amenazas de seguridad

Una amenaza es definida como la causa potencial de un incidente no deseado, que puede provocar daños a un sistema o a la organización. [ISO/IEC 27000]

Una amenaza de la seguridad es definida como una potencial violación de la seguridad. Ejemplos de estas amenazas incluyen:

1. No autorizada revelación de la información.
2. No autorizada destrucción o modificación de los datos, equipos u otros recursos.
3. Robo, desaparición o pérdida de la información u otros recursos.

4. Interrupción o denegación de servicios.
5. Impersonal, o enmascarado como una entidad autorizada.

Estas amenazas pueden ser accidentales (algunas muchas veces llamadas inadvertidas) o intencional y pueden ser activas o pasivas. Una amenaza accidental es una con intención no premeditada, semejante como un sistema o software con mal funcionamiento o un fallo físico. Una amenaza intencional es la realizada por alguien comprometido a realizar un acto deliberado. Las amenazas intencionales pueden alcanzar un rango desde la casual exploración, usando herramientas accesibles y de fácil monitoreó, o un avanzado ataque para un sistema especialmente conocido. Cuando una amenaza intencional es realizada es llamada un ataque. Una amenaza activa es una que el resultado algunas veces cambia el estado u operación de un sistema, semejante como alteración de datos o destrucción del equipamiento físico. Una amenaza pasiva no abarca los cambios de estados. Escuchar e interceptar las líneas de teléfono son ejemplos de amenazas pasivas. [18]

1.4 Principales ataques de seguridad realizada a las redes de datos

Un sistema informático está compuesto por tres partes: el hardware, el software y los datos. A cualquiera de estos tres componentes pueden ser dirigidos los ataques, y los mismos pueden ser separados según su taxonomía, en: interrupción, intercepción, modificación y fabricación. [19]

1- Interrupción: Existe cuando un recurso de un sistema es destruido, o se hace indisponible o inusable. Es un tipo de ataque a la disponibilidad del recurso.

2- Intercepción: Ocurre cuando una parte no autorizada logra capturar el objeto siendo éste, un receptor no autorizado. Esta parte podría ser una persona, un programa o una computadora. Ejemplos incluyen conectar dispositivos a una red de datos privada para capturar paquetes de transmisión, o el copiado ilegal de archivos y programas.

3- Modificación: Esto se logra cuando además de ganar el acceso al recurso, se modifica y reenvía al destino. Este ataque es a la integridad de los datos, donde por ejemplo se podrían cambiar valores en un archivo o una base de datos, alterar un programa para que funcione distinto, o modificar los mensajes transmitidos en una red.

4- Fabricación: Es cuando una parte no autorizada genera objetos en el sistema. Se considera un ataque a la autenticidad, y como ejemplos se puede enumerar el agregado de filas o registros a una base de datos, la inserción de paquetes a una red, o en mayor escala, el envío de mensajes desde un servidor de correo electrónico controlado, mediante el reemplazo de la identidad de origen del remitente. [Ver Anexo 3]

Todos estos ataques se pueden realizar aprovechando distintas vulnerabilidades de los sistemas operativos. Los más comunes se pueden clasificar en: [19]:

- *Bugs:* El bug es un error de software generado durante el proceso de creación del mismo cuando no se contemplan todos los posibles estados que el sistema puede tomar en tiempo de ejecución. Los errores más comunes pueden ser: división por

cero, un ciclo infinito, desbordamiento de buffer (buffer overflow y underflow), utilización de variables no inicializadas, acceso a un área de memoria restringida.

- *Backdoors:* También llamado "puerta trasera", es una secuencia especial en el código, generada por el programador, para modificar el normal flujo del sistema. Estos métodos de modificación de accesos también pueden ser instalados a partir de programas maliciosos como el conocido puerta trasera o el Sony/BMGrootkit.
- *Acceso físico:* Otra forma de acceder a los datos es de forma directa a la terminal, con herramientas como OphCrack o John the Ripper con los que se puede fácilmente obtener el nombre de cuenta de administrador y sus contraseñas, y así lograr tener el control completo de la computadora.
- *Bomba Lógica:* Es uno de los mecanismos más viejos de amenaza computacional. Es una rutina que hace que un programa que funciona adecuadamente, en una fecha, momento determinado o condición específica, "explote", o generalmente deje de funcionar adecuadamente, pudiendo de manera optativa dañar información.
- *Caballo de Troya:* Es un programa útil o aparentemente útil que tiene comandos o procedimientos ocultos y que cuando es invocado, realiza funciones no queridas o dañinas, generalmente para tomar el control de la máquina.
- *Bacterias:* Son programas que no corrompen explícitamente archivos sino que su función es replicarse a sí mismas, realizando como principal daño, que el sistema se sature (disminuya su potencia computacional, espacio libre de memoria RAM, espacio libre en disco, etc.).
- *Virus:* Tal y como ocurre con los homónimos biológicos, son programas ocultos que se replican y liberan en determinadas fechas, o momentos, una carga letal, que generalmente involucra la destrucción de información en el sistema infectado, desde algún archivo de manera aleatoria, hasta todos los discos del sistema infectado.
- *Spyware:* Los programas espías son aplicaciones que recopilan información sobre una persona u organización sin su conocimiento. La función más común que tienen estos programas es la de recopilar datos sobre el usuario y distribuirlo a empresas publicitarias. Dado que el spyware usa normalmente la conexión de una computadora a Internet para transmitir información, consume ancho de banda, y por lo tanto puede verse afectada la velocidad de transferencia de datos de dicha computadora.
- *Rootkits:* Es una herramienta, o grupo de ellas que tiene como finalidad esconderse sí misma y esconder a otros programas, procesos, archivos, directorios, llaves registro y puertos, que permiten al intruso mantener el acceso a un sistema remotamente comandar acciones o extraer información sensible. Existen para una amplia variedad de sistemas operativos, como Linux, Solaris o Microsoft Windows. En los sistemas Unix/Linux, existen varias aplicaciones para rootkits, entre los más populares se encuentran el chkrootkit y el rkhunter.
- *Worms:* Los gusanos son semejantes a los virus en el sentido de que se esparcen, pero están orientados a otro tipo de contagio, intentando detectar debilidades en determinados programas que atienden servicios, como pueden ser servidores Web, Ftp, mails, entre otros. Una vez que encuentran una vulnerabilidad, envían un mensaje que produce un mal funcionamiento en el servidor, donde queda una copia del worm, el cual comienza a escanear desde ese servidor otras redes. Generalmente terminan por agotar los recursos computacionales de estos servidores, de toda la red, y a veces de toda Internet.

- *Ingeniería Social:* Proviene del estudio sistemático que realizan los atacantes, de las debilidades humanas. Para ello utilizan técnicas y mensajes que hacen uso de diferentes mecanismos, incluso del subconsciente, y normalmente de sentimientos de avaricia, solidaridad, compasión, ira, actualidad, desinformación, etc. Un clásico ejemplo es el envío de un email con la URL modificada, de un aviso de cambio de contraseñas del homebanking, obviamente redirigida a un sitio malicioso dedicado a recolectar usuarios y contraseñas de usuarios bancarios.
- *Spam:* Esta técnica se basa en la recolección de grandes bases de datos de correo electrónico por parte de robots que recorren la web en busca de emails. Es fuente común de posteriores infecciones con virus y otras plagas masivas.
- *Scam:* Es una técnica específica de Spam que normalmente sirve para obtener información privada y claves, para posteriormente acceder a sistemas del usuario y realizar hurto de información o bancaria, basado generalmente en los miedos del atacado.

1.5 Servicios de seguridad

X.800 define un servicio de seguridad como un servicio que es provisto por una capa de protocolo de comunicación de sistemas abiertos y que asegura una adecuada seguridad de los sistemas o de los datos transferidos. Esta es definida en RFC 4949, la cual proporciona la siguiente definición: *es un proceso o servicio de comunicación que es provisto por un sistema al dar una específica y aceptable protección a los recursos de un sistema; la implementación de los servicios de seguridad, políticas de seguridad son implementados por mecanismos de seguridad. X.800 dividido estos servicios en cinco categorías y catorce servicios específico*s. (Ver Anexo 4). [20]

1.6 Mecanismos de seguridad

El Anexo 5 lista los mecanismos de seguridad definidos en X.800, y en el anexo 6, basado en un X.800, puede ser observada la relación entre los servicios de seguridad y los mecanismos de seguridad**.** [21]

Las políticas de seguridad deben estar sostenidas sobre tres pilares fundamentales:

- Prevención: Utilizando métodos de autenticación, identificación, control de acceso, transmisión segura, plataformas heterogéneas, sistemas honeypot, etc.
- Detección: A través de programas de auditoría como Tripwire o Nagios, encargados de realizar chequeos de integridad, proveer y presentar información al instante sobre el estado actual del sistema.
- Respuesta: Implementando métodos de resguardo (backup) y software de análisis forense (para detectar que hizo el intruso y que vulnerabilidad explotó).

1.6.1 Mecanismos de prevención y detección más usados en los sistemas Unix

- *Unicidad:* Consiste en incluir en los datos un número de secuencia, fecha/hora, número aleatorio, o alguna combinación de los anteriores para verificar la integridad de los mismos.
- *Control de enrutamiento:* Permite enviar información por zonas clasificadas y a su vez permite solicitar y establecer rutas alternativas en caso de violaciones de seguridad.
- *Tráfico de relleno:* Consiste en enviar tráfico falso junto con los datos válidos para que el atacante no pueda diferenciar los reales de los espurios. Vinculado directamente con la esteganografía, es una forma de ocultar información en objetos que puedan llegar a pasar desapercibidos (como ocultar un árbol en un bosque).
- *Cifrado:* Fundamental para garantizar la seguridad de la información. Consiste en aplicar un proceso de transformación a un texto claro mediante un cálculo matemático y así obtener un texto cifrado, incomprensible para entidades no autorizadas.
- *Gestión de claves:* Abarca la generación, distribución, almacenamiento, tiempo de vida, destrucción y aplicación de las claves de acuerdo a la política de seguridad aplicada.
- *Cortafuegos:* Son conjuntos de aplicaciones o equipos ubicados entre dos redes que establecen la política de acceso entre las partes.
- *Filtrado de paquetes:* En este caso se realiza una lectura pormenorizada de la cabecera de cada paquete y en función a una serie de reglas se permite el paso de los mismos o dicha trama es descartada. Los mismos pueden ser analizados verificando el protocolo utilizado las direcciones de origen/destino, o los puertos de origen/destino.
- *Proxy de aplicación:* Es un software encargado de eliminar las conexiones a servicios tales como FTP o Telnet, y permiten únicamente la utilización de servicios en donde se encuentra un proxy. Por ejemplo, si la puerta de enlace posee un Proxy solo para las conexiones HTTP y FTP, el resto de los servicios no estarán disponibles.
- *Monitoreo de la Actividad:* Es indispensable para garantizar la seguridad del sistema, mantener monitoreada la actividad de las aplicaciones de seguridad, y así poder detectar a tiempo los ataques a los que puede estar siendo sometido.

1.7 Modelos de seguridad

Un modelo general, que es referenciado por muchos autores para la seguridad en redes puede observarse en el Anexo 7. Cuando un mensaje va a ser transferido desde un punto a otro utilizando cualquier servicio de Internet. Dos partes, quienes son las principales en la transacción, pueden cooperar para que tome lugar el intercambio. Un canal de información lógica es establecido para definir una ruta a través de Internet desde la fuente al destino y por el cooperativo uso de los dos protocolos principales de comunicación (TCP/IP).

1.7.1 Seguridad por oscuridad

La seguridad por oscuridad, a veces llamada seguridad por ocultación, es un método que utiliza el secreto de diseño o implementación para asegurar que básicamente, por desconocimiento, no se encontrarán los puntos débiles de dicho sistema.

1.7.2 Seguridad del perímetro

En La seguridad basada en la defensa perimetral apunta a reforzar los puntos de acceso o conexión de nuestra red privada con la red externa. Para ello se tiene que evaluar y planear qué tipos de acceso requiere el sistema, implementar sistemas de seguridad para bloquear el resto del tráfico (por ejemplo firewall o proxy), proteger esos únicos puntos vulnerables, y ahí mismo ubicar sistemas de monitoreo y detección de intrusos para que den aviso al administrador del sistema y así poder ejecutar acciones defensivas a tiempo.

1.7.3 Seguridad en profundidad

La seguridad en profundidad asume que cada una de las medidas tomadas pueden ser rotas por algún atacante. Sin embargo, a medida que se agreguen capas en el sistema de seguridad, la probabilidad de que el atacante pueda esquivar todas y cada una de ellas sin ser descubierto disminuye proporcionalmente.

1.8 Herramientas de seguridad sobre software libre en redes de datos

La supervisión de la seguridad integra conocimientos sobre Auditorias de Seguridad, Análisis de Datos, Detección de Vulnerabilidades, Seguridad Perimetral, Implementación de Políticas, Detección de Intrusos y Supervisión del Tráfico entre otros.

Garantizar un adecuado nivel de seguridad en una red implica entre otras cosas que es necesario supervisar de manera continua y automática los servicios que se ofrecen en la misma para detectar posibles fallos o alteraciones del sistema así como a posibles intrusos. Es de vital importancia la selección de herramientas de supervisión adecuadas a cada finalidad que estén acordes con las condiciones reales en cuanto a hardware y software de la empresa en que serán utilizadas.

1.8.1 Metodología de una prueba de penetración

Una Prueba de Penetración es el proceso utilizado para realizar una evaluación o auditoria de seguridad de alto nivel. Una metodología define un conjunto de reglas, prácticas, procedimientos y métodos a seguir e implementar durante la realización de cualquier programa de auditoría en seguridad de la información. Una metodología de pruebas de penetración define una hoja de ruta con ideas útiles y prácticas comprobadas, las cuales deben ser manejadas cuidadosamente para poder evaluar correctamente los sistemas de seguridad.

1.8.2 Tipos de Pruebas de Penetración:

Existen diferentes tipos de Pruebas de Penetración, las más comunes y aceptadas son las Pruebas de Penetración de Caja Negra (Black-Box), las Pruebas de Penetración de Caja Blanca (White-Box) y las Pruebas de Penetración de Caja Gris (Grey-Box). [22]

1.8.2.1 Prueba de Caja Negra

No se tienen ningún tipo de conocimiento anticipado sobre la red de la organización. Un ejemplo de este escenario es cuando se realiza una prueba externa a nivel web, y está es realizada solo con el detalle de una URL o dirección IP proporcionado al equipo de pruebas. Este escenario simula el rol de intentar irrumpir en el sitio web o red de la organización. Así mismo simula un ataque externo realizado por un atacante malicioso. [22]

1.8.2.2 Prueba de Caja Blanca

El equipo de pruebas cuenta con acceso para evaluar las redes y ha sido dotado de diagramas de la red y detalles sobre el hardware, sistemas operativos, aplicaciones, entre otra información antes de realizar las pruebas. Esto no iguala a una prueba sin conocimiento, pero puede acelerar el proceso en gran magnitud con el propósito de obtener resultados más precisos. La cantidad de conocimiento previo conduce a realizar las pruebas contra sistemas operativos específicos, aplicaciones y dispositivos de red que residen en la red, en lugar de invertir tiempo enumerando lo que podría posiblemente estar en la red. Este tipo de prueba equipara una situación donde el atacante puede tener conocimiento completo de la red interna. [22]

1.8.2.3 Prueba de Caja Gris

El equipo de pruebas simula un ataque realizado por un miembro de la organización inconforme o descontento. El equipo de pruebas debe ser dotado con los privilegios adecuados a nivel de usuario y una cuenta de usuario, además de permitirle acceso a la red interna. [22]

1.8.3 Evaluación de Vulnerabilidades y Prueba de Penetración

Una evaluación de vulnerabilidades es el proceso de evaluar los controles de seguridad interna y externa para identificar las amenazas que planteen una seria exposición para los activos de la organización.

La principal diferencia entre una evaluación de vulnerabilidades y una prueba de penetración, radica en que las pruebas de penetración van más allá del nivel de únicamente identificar vulnerabilidades, y van hacia el proceso de su explotación, escalar privilegios, y mantener el acceso en el sistema objetivo. Mientras que la evaluación de vulnerabilidades proporciona una amplia visión de las fallas existentes en los sistemas, pero sin medir el impacto real de estas para los sistemas en consideración. [22]

1.8.4 Metodologías de Pruebas de Seguridad

Existen diversas metodologías O*pen Source* que tratan de conducir o guiar los requerimientos de las evaluaciones en seguridad. La idea principal de utilizar una metodología durante la evaluación, es ejecutar diferentes tipos de pruebas paso a paso para poder juzgar con mucha precisión la seguridad de un sistema. Entre estas metodologías se enumeran las siguientes: [22]

- Open Source Security Testing Methodology Manual (OSSTMM) http://www.isecom.org/research/
- The Penetration Testing Execution Standard (PTES) http://www.pentest-standard.org/
- Penetration Testing Framework http://www.vulnerabilityassessment.co.uk/Penetration%20Test.html
- OWASP Testing Guide https://www.owasp.org/index.php/Category:OWASP_Testing_Project
- Technical Guide to Information Security Testing and Assessment (SP 800-115) http://csrc.nist.gov/publications/PubsSPs.html
- Information Systems Security Assessment Framework (ISSAF) http://www.oissg.org/issaf

1.8.5 Introducción a Kali Linux

Para realizar las auditorias de seguridad, pruebas de penetración y vulnerabilidades en la red de datos se utilizara a Kali. El sistema operativo Kali es una plataforma basada en GNU/Linux Debian y es una reconstrucción completa de BackTrack, la cual contiene una gran cantidad de herramientas para capturar información, identificar vulnerabilidades, explotarlas, escalar privilegios y cubrir las huellas, entre estas se encuentran las populares Hydra, Maltego, Ettercap o Zaproxy. Como la mayoría de distribuciones Linux es de código abierto y gratuito así como la mayoría de sus herramientas. Las aplicaciones se encuentran divididas por secciones, dependiendo de qué ramo de seguridad abarquen. (Ver Anexo 8).

1.8.6 Características de kali Linux

Como fue mencionado con anterioridad Kali Linux es una completa reconstrucción de BackTrack Linux, y se adhiere completamente a los estándares de desarrollo de Debian. Se ha puesto en funcionamiento toda una nueva infraestructura, todas las herramientas han sido revisadas y empaquetadas, y se utiliza ahora Git para el VCS. Seguidamente se pueden observar algunas de sus características más importantes:

- Más de 300 herramientas de Pruebas de Penetración
- Es Libre y siempre lo será
- Git (Árbol de código abierto)
- Cumple con FHS (Filesystem Hierarchy Standart)
- Amplio soporte para dispositivos inalámbricos
- Parches al Kernel para inyección.
- Entorno de desarrollo seguro
- Paquetes y repositorios firmados con GPG y repos.
- Varios lenguajes

- Completamente personalizable
- Soporte ARMEL y ARMHF

1.8.7 Sistemas de supervisión

1.8.7.1 Técnicas de Detección de Sniffers

La detección de sniffers en la red de datos es una de las principales tareas que tiene un administrador para garantizar que la seguridad de red no se violada por ningún usuario o intruso ya sea dentro o fuera de la misma.

1.8.7.2 Técnicas locales de detección

Esta es la forma más trivial de localizar un sniffer. Si se dispone de acceso local a la máquina, en los sistemas operativos de la familia Unix se dispone de una utilidad comúnmente usada pero pocas veces con atención. El comando ifconfig informa, además del estado de las interfaces, si la misma se encuentra en modo promiscuo o no. Una buena práctica es verificar el estado de las interfaces con este comando periódicamente, y una manera de automatizarlo sería agregando dicha tarea en el crontab para que, en caso de descubrir que ha cambiado a modo promiscuo, envíe un alerta al administrador de la red (también podría deshabilitar la misma inmediatamente). Es evidente que la no detección temprana de una posible violación de seguridad puede hacer que el atacante disponga de tiempo como para sustituir el ejecutable ifconfig o cmp compilando uno de similares características pero que no informe el cambio de funcionamiento de la interfaz de red; o de la misma manera, deshabilitar la tarea programada en el crontab. Igualmente la integridad de los ficheros ejecutables es fácilmente comprobable utilizando funciones resumen como MD5 o SHA-1.

1.8.7.3 Técnicas de detección remota

Existen numerosas técnicas de detección de sniffers para el segmento de red, aunque hay que tener en claro que las mismas tienen ciertas limitaciones porque en general hay posibilidades de que igualmente existan sniffers en la red sin ser detectados, o también, máquinas completamente inocentes que sean detectadas como falsos positivos. Estas técnicas se pueden clasificar en las que son dependientes del sistema operativo, y las que no.

1.8.7.3.1 Técnicas dependientes del sistema operativo

Como su nombre lo indica, estas técnicas aprovechan algún fallo del sistema operativo (o de su subsistema encargado del TCP/IP) para reconocer que una placa de red se encuentra en modo promiscuo. Con un muy buen rendimiento cuando se exploran máquinas donde coincide exactamente la versión del sistema operativo utilizado, la gran desventaja es la cantidad de falsos negativos que genera.

1.8.7.3.2 Técnicas no dependientes del sistema operativo

En general, estas técnicas son menos fiables ya que suelen basarse en suposiciones sobre el comportamiento de determinados sniffers. A continuación se detallan diversas técnicas de esta metodología.

1.8.7.3.3 Pruebas de DNS

Los test de DNS se basan en la suposición de que los sniffer transforman las direcciones IP en sus correspondientes nombres para facilitar la lectura del tráfico en su presentación. Si este es el caso, se podría tratar de "engañar" al sniffer con paquetes falsos que contengan una IP nueva. Todas las máquinas, al ver que esta IP no les pertenece, descartarán dichos paquetes. Pero el sniffer, queriendo convertir la IP en un nombre, y al no disponer el mismo en su cache local, hará una consulta de DNS y de esta forma revelará su situación.

1.8.7.3.4 Pruebas de latencia

En este caso, el método se aprovecha de la condición de que en modo promiscuo, los paquetes se rechazan o aceptan a nivel de aplicación (ya que todos pasan al kernel como propios). En forma normal esto ocurriría a nivel de hardware lo que implicaría tiempos menores de respuesta si se analizan las estadísticas de tráfico. Es por eso que lo único que hay que hacer es analizar dichos tiempos. Aunque lamentablemente es posible evitar ser detectado como sniffer si se monitorea en busca de una drástica inundación de tráfico en la red, de manera que en este caso se pudiera cambiar la placa de red de modo promiscuo a modo normal; técnica de evasión ya implementada en varios de los sniffers actuales.

1.8.7.3.5 Pruebas ARP

Se puede enviar una petición ARP al objetivo con toda la información necesaria pero con la MAC ADDRESS errónea. Una máquina que no esté en modo promiscuo nunca verá ese paquete, puesto que no era el destino original, por lo tanto no contestará. Si en cambio, la placa de red se encuentra en modo promiscuo, la petición ARP será considerada y el kernel la procesará contestando y quedando evidenciada.

1.8.7.4 Sistemas de detección de intrusos

Un Sistema de Detección de Intrusos o IDS es un programa utilizado para detectar accesos no autorizados a una red, técnica relativamente nueva que se agrega a los métodos ya conocidos de defensa, básicamente para recolectar información utilizando plugins como sensores que buscan patrones o firmas de ataques conocidos analizando el tráfico de una red. Estos sistemas de detección pueden almacenar dichas señales y dar avisos de alerta al administrador de la red, o actuar en forma reactiva por ejemplo, reprogramando el firewall de la red o intercambiando información con otro componente de la seguridad del sistema.

1.8.7.4.1 NIDS

Sistema de detección de intrusos en una red. Busca detectar anomalías que inicien un riesgo potencial, tales como ataques de denegación de servicio, detectores de puertos abiertos o intentos de entrar en un ordenador analizando el tráfico en la red en tiempo real. Para ello, analiza todos los paquetes, buscando en ellos patrones sospechosos. Los NIDS no sólo vigilan el tráfico entrante, sino también el saliente o el tráfico local, ya que algunos ataques podrían ser iniciados desde el propio sistema protegido. A pesar de la vigilancia, su influencia en el tráfico es casi nula. Para que los NIDS sean efectivos, han de ser actualizados periódicamente. En caso de detectar un ataque contra el sistema, puede tomar medidas protectoras. Un aspecto negativo de los NIDS actuales es su complicación a la hora de obtener las opciones de configuración óptimas para su ejecución. De otro modo, obtendremos demasiados falsos positivos (falsas alarmas, con gran cantidad de información que luego un administrador tendrá que procesar) o pasará sin advertir ciertos ataques.

1.8.7.4.2 HIDS

Sistema de detección de intrusos en un host. Busca detectar anomalías que indican un riesgo potencial, revisando las actividades en la máquina (host). Puede tomar medidas protectoras. Las funciones de este tipo de software son muy similares a las de los IDS. Poseen funcionalidades que permiten configurar varios HIDS repartidos dentro de la red para enviar sus resultados a un servidor centralizado que los analizará en busca de los riegos y emitirá alertas.

1.8.7.4.3 Psad

El servicio es un detector de intrusos y un analizador de logs de iptables. Consta de tres servicios ligeros que analizan los logs de Iptables, para detectar los escaneos de puertos y otro tráfico sospechoso. Además, psad se nutre de las firmas del detector de intrusos Snort, para detectar accesos de programas del tipo “backdoor” (EvilFTP, GirlFriend) herramientas para ataques DOS (mstream, shaft) y otras más avanzadas de escaneo de puertos.

1.8.7.4.4 Arpwatch

La herramienta Arpwatch utilizado para detección de anomalías en direcciones MAC, corre bajo Linux y está en los repositorios de las principales distribuciones, puede ser instalada con apt-get, yum, rpm, etc. Esta herramienta es muy útil para detectar ataques vía arp-spoofing.

1.8.7.5 Sistemas de Prevención de intrusos

Un Sistema de Prevención de Intrusos (IPS) es un dispositivo que ejerce el control de acceso en una red informática para proteger a los sistemas computacionales de ataques y abusos. La tecnología de Prevención de Intrusos es considerada por algunos como una extensión de los Sistemas de Detección de Intrusos (IDS), pero en realidad es otro tipo de control de acceso, más cercano a las tecnologías cortafuegos.

Los IPS fueron inventados de forma independiente por Jed Haile y Vern Paxon para resolver ambigüedades en el monitoreo pasivo de redes de computadoras, al situar

sistemas de detecciones en la vía del tráfico. Los IPS presentan una mejora importante sobre las tecnologías de cortafuegos tradicionales, al tomar decisiones de control de acceso basados en los contenidos del tráfico, en lugar de direcciones IP o puertos. Tiempo después, algunos IPS fueron comercializados por la empresa One Secure, la cual fue finalmente adquirida por NetScreen Technologies, que a su vez fue adquirida por Juniper Networks en 2004. Dado que los IPS fueron extensiones literales de los sistemas IDS, continúan en relación.

También es importante destacar que los IPS pueden actuar al nivel de equipo, para combatir actividades potencialmente maliciosas.

Los IPS se categorizan por la forma que detectan el tráfico malicioso:

- Detección Basada en Firmas
- Detección Basada en Políticas
- Detección Basada en Anomalías
- Detección Honey Pot (Jarra de Miel)

1.8.7.6 Supervisión de vulnerabilidades y disponibilidad de los servicios

Sistema que constantemente monitoriza una red de computadoras buscando problemas causados por servidores sobrecargados y/o caídos, conexiones de red, u otros dispositivos fallidos y luego notifica al administrador de esa red (vía email, teléfono celular u otras alarmas). Las de medición son tiempo de respuesta, disponibilidad y tiempo de funcionamiento, aunque las métricas de consistencia y fiabilidad están empezando a ganar popularidad.

1.8.7.7 SIEM

El acrónimo SIEM se atribuye a los analistas de Gartner Amrit Williams y Nicolett Marcos y se deriva de dos tecnologías independientes, pero complementarias: el Administrador de Eventos de Seguridad (SEM por sus siglas en inglés) y el Administrador de Información de Seguridad (SIM por sus siglas en inglés). Durante la última década, estas dos tecnologías han convergido en una única solución conjunta conocida hoy como SIEM. SEM fue una solución tecnológica que se centró en el seguimiento de eventos de seguridad en tiempo real, así como la correlación y el procesamiento. Estos eventos de seguridad eran típicamente alertas generadas por un dispositivo de seguridad de red, tales como un cortafuego o un Sistema de Detección de Intrusos (IDS por sus siglas en inglés). SIM, por otra parte, se centró en el análisis histórico de la información del archivo de registro para apoyar las investigaciones forenses y los informes. SIM a menudo analiza los mismos eventos que SEM, pero no lo hace en tiempo real. SIM centraliza el almacenamiento de registros y archivos, búsqueda y análisis de funciones y, sólidas capacidades de presentación de informes. Los sistemas SIEMs combinan las capacidades de cada una de estas tecnologías en una única solución, de hecho, las soluciones SIEM actuales con frecuencia incorporan una función de gestión de registros mucho más amplia. [30]

1.8.7.7.1 ARQUITECTURA BÁSICA DE LOS SISTEMAS SIEM

Los sistemas SIEM pueden ser comparados con una máquina compleja que posee un gran número de partes donde cada una realiza un trabajo específico e independiente. Todas estas partes deben colocarse a trabajar juntas adecuadamente o de lo contrario el sistema caerá en caos [30]. La figura 1.2 muestra la arquitectura básica de este tipo de sistemas.

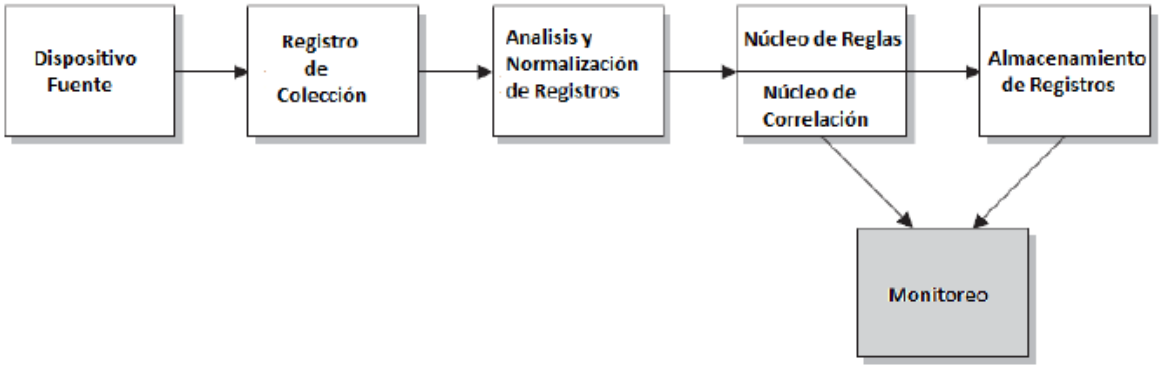


Figura 1.2. Arquitectura básica de un sistema SIEM [30]

A continuación se describen las partes o módulos que aparecen en la figura.

Dispositivo Fuente: La primera parte de un sistema SIEM es el dispositivo que captura la información. Un Dispositivo Fuente es el dispositivo, aplicación que recupera los registros que se almacenan y procesan en el SIEM. El dispositivo de origen puede ser un dispositivo físico en la red (como un router, un switch, o algún tipo de servidor), aunque también pueden ser los registros de una aplicación o cualquier otra información que puede adquirir1 como por ejemplo cortafuego (firewalls), servidores proxy, IDS, Sistemas de Prevención de Intrusiones (IPS por sus siglas en inglés), bases de datos, entre otros. Su comunicación con el resto del sistema puede ser mediante protocolos estándares o protocolos privativos, dependiendo del fabricante de sistema [30].

Registro de Colección: La siguiente parte del sistema es el dispositivo o la aplicación de flujo de registro, el cual obtiene de alguna manera todos los registros de los dispositivos fuentes para luego transportarlos al SIEM. Actualmente, la recolección de datos ocurre de diferentes maneras y a menudo depende del método implementado dentro del sistema final, pero en su forma más básica, los procesos de recopilación de registros se pueden dividir en dos métodos fundamentales de colección: o el Dispositivo Fuente envía sus registros al SIEM, lo que se llama el método de empuje, o el SIEM se extiende y recupera los registros del dispositivo de origen, lo cual se llama el método de extracción. Cada uno de estos métodos tiene sus aspectos positivos y negativos cuando se utilizan en un determinado entorno, pero ambos logran obtener los datos desde el dispositivo de origen en el SIEM. [30]
Análisis/Normalización de Registros: En este punto, los registros están todavía en su formato original en el repositorio centralizado y por tanto no resultan muy útiles para el sistema. Para que estos registros resulten útiles para el SIEM se les debe dar un formato estándar, lo cual se conoce como normalización. La normalización de los eventos no sólo

hace que sean fáciles de leer estos registros, sino que también facilita y permite un formato estándar para la generación de reglas del sistema, lo que significa que cada SIEM se encarga de las reglas de normalización de diferentes maneras. El resultado final es que todos los registros poseen el mismo aspecto dentro del sistema. Con frecuencia, antes de la normalización de los datos, se realizan copias de los registros, las cuales se almacenan en su formato original dentro del Log Storage. [30]

Núcleo de Reglas/Núcleo de Correlación: Este componente se encuentra dividido en 2 segmentos, el Núcleo de Reglas y el Núcleo de Correlación de Reglas. El Núcleo de Reglas amplía la normalización de los eventos con el fin de activar alertas en el SIEM debido a las condiciones específicas en estos registros. Estas reglas generalmente vienen predefinidas en el sistema, pero también se pueden definir reglas personalizadas. Por lo general, se pueden escribir estas reglas usando una forma de lógica booleana para determinar si se cumplen condiciones específicas y analizar patrones en los campos de datos [30], pero se debe tener precaución para evitar el establecimiento de reglas de correlación demasiado complejas o demasiadas reglas, ya que cada nueva norma aumentará exponencialmente los requisitos computacionales y, eventualmente, pueden hacer que el proceso de correlación resulte ineficaz [30]. La función del Núcleo de Correlación es comparar todos los eventos normalizados de diferentes fuentes con las reglas anteriormente creadas.

Almacenamiento de Registros: Este es usado para facilitar el trabajo en un único almacén de datos, facilitando la relación entre las diferentes funciones del SEM y las funciones forenses e informes del SIM. Su acoplamiento puede parecer sencillo, pero puede presentar una serie de retos y consideraciones. Este puede ser una base de datos, un archivo de texto plano o un archivo binario [30], ubicado de forma central o distribuida en dependencia al tamaño de la empresa, la cantidad de datos que son recogidos, y la infraestructura de TIC (Tecnologías de Información de Comunicación)

Monitoreo: Una vez que el SIEM tenga todos los registros y los acontecimientos que se han procesado, se necesita hacer algo útil con la información. Un SIEM tendrá una interfaz de consola y una interfaz que bien puede ser o basarse en una aplicación Web. Ambas interfaces le permiten visualizar y analizar todos los datos almacenados en el SIEM, facilitando de esta manera la gestión del sistema, pues brinda a los administradores una única visión de todo el entorno. También aquí se puede desarrollar el contenido y las reglas que se utilizan para extraer la información de los eventos que se están procesando [30].

1.8.7.8 OSSIM

La sigla OSSIM se deriva para *Open Source Security Information Management* (Herramienta de Código Abierto para la Gestión de Seguridad de la información), OSSIM no es una herramienta única, al decir OSSIM se entiende que es un conjunto de herramientas unidas en un solo programa que facilita el análisis, visualización y la gestión de manera centralizada de los eventos que ocurren en los diferentes componentes de la infraestructura IT de la empresa, obteniendo de esta forma mayor efectividad a la hora del monitoreo y de encontrar errores u vulnerabilidades en la seguridad de la red.

OSSIM es una herramienta que ayuda mucho en el monitoreo de la red, permitiendo controlar algo tan básico desde un log de la contraseña mal digitada hasta un posible

ataque que se esté dando a nuestra infraestructura. Esta herramienta trae incorporada cerca de 22 Funciones, todas estas son Open Source capaces de correlacionarse y así poder tener el control centralizado.

1.8.7.8.1 Características de OSSIM

Ossim trae un conjunto de características que permite al administrador de red gestionar de forma más eficiente la seguridad interna de los servidores, de la red de datos y voz, entre las cuales poden ser mencionadas los siguientes:

- Es gratuito.
- Monitoreo centralizado.
- Analiza el comportamiento de nuestra Red
- Presenta informes técnicos.
- Realiza un análisis de los posibles riesgos y anomalías en la red.
- Controla los posibles ataques/intruso en la red.
- Monitorea el excesivo tráfico que se pueda generar.
- Presenta una interfaz gráfica web amigable hacia al Administrador
- Permite recolectar logs de los servidores sin importar que distribución de Linux tenga instalado.
- El cliente recolector de logs que se instala en Windows es muy sencillo de configurar.
- Realiza pruebas de vulnerabilidad.
- Realiza notificaciones automáticas mediante alertas.

1.8.7.8.2 Componentes

Dentro de OSSIM Alienvault existe una gran variedad de las mejores herramientas Open Source, algunas de las más destacadas se enumeran a continuación:

Snort: es el más importante IDS Open Source disponible en la actualidad. OSSIM contiene una versión personalizada de esta herramienta y es quien alerta sobre intentos de ataques a la red.

OpenVAS: es la versión GPL (General Public License) de Nessus, una popular herramienta de escaneo de vulnerabilidades Open Source. Esta herramienta se utiliza para proporcionar búsqueda de vulnerabilidades de los recursos de red y añade esta valiosa información a la base de datos de OSSIM. Nessus también es incluido dentro de OSSIM y es soportado utilizando un plugin.

Ntop: es una popular herramienta Open Source para la monitorización del tráfico de la red. Esta herramienta proporciona información muy valiosa sobre el tráfico en la red, que puede ser utilizada para detectar de una manera proactiva el tráfico anormal o malicioso.

Nagios: es una popular herramienta Open Source de monitoreo de dispositivos de red. Es una de las herramientas más complejas, pero le permite al administrador tener una única visión del estado de los hosts de la red. A través del monitoreo de hosts, Nagios puede enviar alertas en caso de fallas y posee una interface web desde donde se puede observar el estado de la red.

PADS: El Sistema de Detección Pasiva de Activos (PADS por sus siglas en inglés) es una herramienta única. La herramienta supervisa silenciosamente el tráfico de red, los registros de los host y las actividades de servicio, con el objetivo de detectar anomalías sin generar tráfico de red, realizando un inventario de activos y revisando los servicios que cada cual ejecuta.

P0f: La herramienta P0f toma pasivamente las huellas dactilares del sistema operativo (el descubrimiento del tipo de sistema operativo y su versión). Esta herramienta escucha silenciosamente el tráfico de red e identifica los sistemas operativos que se comunican en la red. Esta información resulta útil en el proceso de correlación.

OCS-NG: La OCS-NG (Open Computer and Software Inventory Next Generation) ofrece la capacidad multiplataforma de gestión de recursos. Esta herramienta permite mantener un inventario actualizado en tiempo real de los dispositivos existentes en la red.

OSSEC: Sistema de Detección de Intrusiones de Host (HIDS por sus siglas en inglés) Open Source. Este se encarga de analizar los datos del host y detectar a través de ellos si un host está siendo víctima de algún ataque.7 OSSEC realiza esta tarea analizando logs, chequeando la integridad de archivos, monitoreando el registro de Windows, detectando rootkits, además de responder y alertar en tiempo real. Esta herramienta también ayuda a proteger al propio OSSIM.

OSVDB: La OSVDB (Open Source Vulnerability Database), es la base de datos que mantiene la información actualizada con respecto a las vulnerabilidades del sistema. Esta se ha utilizado por OSSIM durante el proceso de correlación y es quien proporciona un análisis cuando sea necesario.

NFSen/NFDump: Visor de flujos de red para la detección de anomalías en la red. Este además permite el procesamiento de Netflow v5, v7 y v9. NFSen proporciona una interfaz gráfica basada en web a NFDump. Ambos NFSen y NFDump se han integrado en OSSIM y han sido modificados para trabajar con las otras herramientas.

Inprotect: Interfaz basada en web para Nessus, OpenVAS y NMAP. Inprotect ofrece la posibilidad de definir perfiles de escaneo, programar sondeos, y exportar los resultados del análisis de distintos formatos.

OSSIM también tiene otras destacadas herramientas como Arpwatch, el cual es utilizado para detección de anomalías en el uso de direcciones MAC, MACSpade, el cual es un motor de detección de anomalías en paquetes utilizados para obtener conocimiento de ataques sin firma, Tcptrack, que es utilizado para conocer la información de las sesiones, con lo cual puede conceder información útil relativa a los ataques, Osiris, que es un HIDS, y Snare, quien colecciona los logs de sistemas Windows. [30]

1.8.7.8.3 Arquitectura

OSSIM tiene una arquitectura abierta, siendo OsssimServer el eje central de esta arquitectura, compartiendo con el Ossim-Framework y el Ossim-Agent. (Anexo 9)

Ossim-server: Como toda aplicación, Ossim funciona con un estándar cliente servidor y es obligatorio tener un solo servidor en toda nuestra red en el cual al instalar el perfil server (servidor) estamos configurando el ambiente que se encargue de procesar y recoger todos los logs que son generados por los diferentes dispositivos y servidores de nuestra red interna.

Ossim-framework: Esta componente sirve como intermediario para que la aplicación web del servidor no haga tareas en segundo plano como la lectura y escritura de la información que recibe, evitando así un innecesario uso de requerimiento como memoria y almacenaje y optimizar su funcionabilidad. Los propósitos primordiales de este componente son:

- Recolectar datos de los agentes y otros servidores
- Priorizar los eventos recibidos.
- Correlacionar los eventos recibidos de diferentes fuentes
- Realizar la evaluación de riesgos y disparar alarmas
- Almacenar eventos en la base de datos
- Reenviar eventos o alarmas a otros servidores

Ossim-agent: El nombre de Agent en la herramienta Ossim se les da a los plugins y aplicaciones que permite analizar todos los eventos específicos que se generan en la red de trabajo o en los diferentes servidores en la cual se está haciendo el monitoreo y seguimiento.

1.8.7.8.4 Escaneo de Vulnerabilidades

Osssim más que una herramienta de monitoreo de eventos “logs”, también es un SIEM (*Security Information and Event Management*) y trae incorporado diversas formas para gestionar la seguridad tales como:

- Antivirus que se encarga de detectar y eliminar software malicioso de los sistemas informáticos Windows.
- Detectores de intrusos basados en host (HIDS, *Host-based Intrusion Detection Systems*) encargado de monitorear procesos y archivos críticos del sistema bajo análisis.
- Detectores de intrusos basados en red (NIDS, *Network-based Intrusion Detection Systems*) responsables de la revisión de los datos que circulan por la red y avisan cuando observan tráfico que evidencia un ataque.
- Detectores de vulnerabilidades (Snort) que hacen un análisis detallado y arrojan como resultado las vulnerabilidades que existen en el sistema operativo y el software instalado.
- Detectores de disponibilidad que permiten verificar si el estado del equipo a monitorear se encuentra UP (activo) o DOWN (caído).

1.8.7.9 Nagios

Nagios es un sistema de monitorización de redes de código abierto ampliamente utilizado, que vigila los equipos (hardware) y servicios (software) que se especifiquen, alertando cuando el comportamiento de los mismos no sea el deseado. Se trata de un software que proporciona una gran versatilidad para consultar prácticamente cualquier parámetro de interés de un sistema, y genera alertas, que pueden ser recibidas por los responsables correspondientes mediante correo electrónico y mensajes SMS, entre otros medios, cuando estos parámetros exceden de los márgenes definidos por el administrador de red.

1.8.7.10 Sistemas de supervisión de la navegación Web y correo

Los servicios más ampliamente utilizados son la navegación web y el uso del correo tanto de manera interna (la intranet de la empresa) como externa (internet). Estos sistemas nos permiten vigilar los accesos a nuestros servidores web, los usuarios que están usando el internet, que websites están visitando, así como el uso del correo electrónico.

1.8.7.11 Squid

Squid es un popular programa de software libre que implementa un servidor proxy y un dominio para caché de páginas web publicado bajo licencia GPL.

1.8.7.11.1 Sarg. Análisis de los reportes de navegación del squid

Es una herramienta que te permite ver con detalle la actividad de todos los equipos y/o usuarios dentro de la red de área local (LAN), provee mucha información acerca de las actividades de usuarios de Squid tiempo, bytes, sitios, etc.

1.8.7.11.2 Sqstat. Monitoreo en vivo de la navegación del squid

SqStat es un script creado en PHP que permite ver las conexiones de los usuarios a través de Squid en tiempo real. SqStat utiliza el protocolo cachemgr para obtener la información del proxy.

1.8.7.11.3 MRTG. Graficado de la navegación del squid

MRTG (Multi Router Traffic Grapher) es una herramienta, escrita en C y Perl, que se utiliza para supervisar la carga de tráfico de interfaces de red. MRTG genera un informe en formato HTML con gráficas que proveen una representación visual de la evolución del tráfico a lo largo del tiempo.

1.8.7.12 Apache

Es un servidor web HTTP de código abierto para plataformas Unix (BSD, GNU/Linux, etc.), Microsoft Windows, Macintosh y otras, que implementa el protocolo HTTP/1.1 y la noción de sitio virtual.

1.8.7.12.1 Visitors. Análisis de las estadísticas del servidor web apache

Visitors es un analizador de registros de servidores web diseñado para ejecutarse desde la línea de comandos con soporte para salida de texto o HTML, y la generación de estadísticas en tiempo real. Puede manejar la mayoría de los registros del servidor web Apache incluido el registro de acceso y es muy fácil de usar, no hay ningún archivo de configuración ni base de datos. También puede generar gráficos visuales de análisis de trazado.

1.8.7.13 Postfix

Postfix es un servidor de correo de software libre / código abierto, un programa informático para el enrutamiento y envío de correo electrónico, creado con la intención de que sea una alternativa más rápida, fácil de administrar y segura al ampliamente utilizado Sendmail Postfix es el agente de transporte por omisión en diversas distribuciones de Linux y en las últimas versiones del Mac OS X.

1.8.7.13.1 AWstats. Análisis de las estadísticas de correo

AWStats es una herramienta gratuita de gran alcance y muy completa que analiza los archivos log de servidores web, ftp o de correo. Este analizador de log trabaja desde línea de comandos y muestra toda la información posible en pocas páginas web gráficas. Utiliza un archivo de información intermedio para ser capaz de procesar archivos de registro grandes rápidamente.

1.8.8 Cortafuego o el filtrado de paquetes

Un cortafuego (firewall) es una puerta de enlace de la red con filtro y sólo es eficaz en aquellos paquetes que deben pasar a través de ella. Por lo tanto, sólo puede ser eficaz cuando la única ruta para estos paquetes es a través del cortafuego.

La falta de una configuración estándar (y el lema «proceso, no producto») explica la falta de una solución preconfigurada. Hay, sin embargo, herramientas que facilitan la configuración del cortafuego netfilter, con una representación gráfica de las reglas de filtrado. Fwbuilder es sin duda uno de los mejores de ellos.

El núcleo Linux 2.6 incorpora el cortafuego netfilter. Puede controlarlo desde el espacio de usuario con los programas iptables e ip6tables. La diferencia entre estos dos programas es que el primero actúa sobre la red IPv4, mientras que el segundo actúa sobre IPv6. Debido a que ambas pilas de protocolos de red probablemente continuarán con nosotros durante muchos años, ambas herramientas son necesarias y deberán ser utilizadas en paralelo. [52]

1.8.8.1 Shorewall

Shorewall es un lenguaje de alto nivel de propósito específico para manipular la infraestructura de control de paquetes del núcleo Linux, Netfilter.

Más específicamente, Shorewall es un script en lenguaje BASH (o en Perl en caso que deseemos utilizar shorewall-perl) que interpreta una serie de archivos de configuración a partir de los cuales hace sucesivas llamadas a iptables para definir el conjunto de reglas necesarias representadas por la configuración. Además de iptables, Shorewall también

utiliza otras herramientas para controlar otros módulos de red núcleo Linux como modprobe (para cargar los módulos de Netfilter), iproute (para la definición de reglas de ruteo) y tc (para el control de tráfico de paquetes).

1.8.9 Herramientas sobre software libre para la virtualización

Aun cuando el término virtualización ha sido acuñado en el contexto de los sistemas mainframe de IBM, introducidos en la década de los 60's, uno de los retos actuales es el aseguramiento de este tipo de entornos, que a diferencia de la infraestructura física, plantea nuevos desafíos. En contraste con los entornos físicos, los entornos virtuales basan su operación en infraestructura física unificada, es decir, un servidor físico puede contener uno o varios sistemas operativos hospedados en una misma plataforma. Aquí el tema de la seguridad de ambientes virtuales juega un papel importante.

La virtualización reduce los costes de espacio físico y de consumo eléctrico, aísla los fallos ya que, si un SO virtualizado da problemas no afectará al resto del sistema, ahorro en piezas de hardware, es posible migrar las máquinas virtuales en caliente ahorrando tiempo en la pérdida de servicio además de evitar servidores ociosos o congestionados.

La agilidad que otorga traspasar la administración de una plataforma tecnológica desde el mundo físico a uno lógico, sin duda representa una ventaja sustancial para las compañías. Sólo en términos de performance, la virtualización permite utilizar un servidor en un 70% y 80% de capacidad, a diferencia del 15% ó 20% de rendimiento que alcanzan las máquinas administradas en forma tradicional.

1.8.9.1 Proxmox

"Proxmox Virtual Environment, es un proyecto de código abierto, desarrollado y mantenido por Proxmox Server Solutions GmbH y el apoyo financiero de Internet Foundation Austria (IPA). Una completa plataforma de virtualización basada en sistemas de código abierto que permite la virtualización tanto sobre OpenVZ como KVM."[54]

Proxmox es una distribución bare-metal, es decir no necesita de un sistema operativo previo, el propio entorno proporciona su propio sistema operativo base. En realidad monta un Debian con los servicios básicos. De esta forma se obtiene un rendimiento mucho más elevado. Este cuenta con un amplio soporte para hardware, procesadores Intel y el chipset AMD. Además cuenta con soporte para Linux y Windows de 32 y 64 bits, y su licencia puede ser obtenida de forma gratuita. Una Web de administración con todas las características necesarias y facilidades para crear y gestionar infraestructuras virtuales., sin necesidad de utilizar ningún software de cliente como intermediario.

1.9 La criptografía y su basamento matemático para el diseño de un sistema de seguridad en el ámbito empresarial

La Real Academia Española define criptografía (del griego: oculto + escritura) como: "el arte de escribir con clave secreta o de modo enigmático".

Una definición más técnica de la criptografía podría ser definida como:

"Rama inicial de las Matemáticas y en la actualidad también de la Informática y la Telemática, que hace uso de métodos y técnicas con el objeto principal de cifrar, y por tanto proteger, un mensaje o archivo por medio de un algoritmo, usando una o más claves."

1.9.1 Fundamentos de la seguridad informática

Los pilares sobre los que descansa toda la teoría asociada a los criptosistemas son básicamente tres:

1. La teoría de la información: Estudio de la cantidad de información contenida en los mensajes y claves, así como su entropía.
2. La teoría de los números: Estudio de las matemáticas discretas y cuerpos finitos que permiten las operaciones de cifrado y descifrado.
3. La teoría de la complejidad de los algoritmos: Estudio de la clasificación de los problemas como computacionalmente tratables o intratables.

1.9.2 Herramientas de la criptografía clásica

Todas las herramientas de la criptografía clásica hacen uso de dos técnicas básicas orientadas a caracteres. Shannon propuso el uso de estas técnicas como herramientas para fortalecer la cifra:

1. Técnicas de sustitución: Los caracteres o letras del mensaje en claro se modifican o sustituyen por otros elementos o letras en la cifra. El criptograma tendrá entonces caracteres distintos a los que tenía el mensaje en claro.
2. Técnicas de transposición o permutación: los caracteres o letras del mensaje en claro se redistribuyen sin modificarlos y según unas reglas, dentro del criptograma. El criptograma tendrá entonces los mismos caracteres del mensaje en claro pero con una distribución o localización diferente.

1.9.3 Requisitos de seguridad de un sistema

Los requisitos necesarios para que un sistema de seguridad sea seguro pueden ser resumidos en los siguientes puntos:

1. El algoritmo de cifrado y descifrado deberá ser rápido y fiable.
2. Debe ser posible transmitir ficheros por una línea de datos.
3. La seguridad del sistema deberá residir solamente en el secreto de una clave y no en las funciones de cifra
4. La fortaleza del sistema se entenderá como la imposibilidad computacional (tiempo de cálculo en años que excede cualquier valor razonable) de romper la cifra o encontrar una clave secreta a partir de otros datos de carácter público.

1.9.4 Principios de encriptación convencional

Los requerimientos para el uso del encriptación convencional pueden ser resumidos en:

1- Algoritmo de encriptación fuerte. El oponente, conociendo el algoritmo de encriptación, debe ser incapaz de descifrar el texto cifrado o encontrar la clave

incluso si posee un conjunto de textos cifrados con su texto nativo correspondiente.

2- El emisor y el receptor deben obtener copias de clave secreta de forma segura y deben mantenerla en secreto. (ver Anexo 10) [59]

1.9.5 Criptografía

Un sistema criptográfico es caracterizado por tres dimensiones diferentes:

1. El tipo de operación usada para transformar el texto nativo en texto cifrado: sustitución, permutación o la combinación de ambas.
2. El número de claves usadas: simétrica o asimétrica.
3. La forma en que se procesa el texto nativo: cifrado de bloque o cifrado fluido. [59]

1.9.6 Criptoanálisis

Al proceso de intentar descubrir el texto nativo o la clave se le conoce como criptoanálisis. La estrategia usada por el criptoanalista depende de la naturaleza del esquema de encriptación y de la información accesible al criptoanalista.
En el Anexo 11 se resumen varios tipos de ataques crioptoanaliticos basados en la cantidad de información conocida por el criptoanalista.
Un esquema de encriptación es computacionalmente seguro si:

1. El costo de romper el cifrado excede el valor de la información cifrada.
2. El tiempo requerido para romper el cifrado excede el tiempo de vida útil de la información [59]

Tamaño de la clave (bits)	Número de alternativas de claves	Tiempo requerido (1/µs)	Tiempo requerido (1000000/µs)
32	$2^{32} = 4.3 \times 10^{9}$	35.8 minutos	2.15 milisegundos
56	$2^{56} = 7.2 \times 10^{16}$	1142 años	10 horas
128	$2^{128} = 3.4 \times 10^{38}$	5.4×10^{24} años	5.4×10^{18} años
168	$2^{168} = 3.7 \times 10^{50}$	5.9×10^{36} años	5.9×10^{30} años

Tabla 1.1 Tiempo medio de ruptura utilizando algoritmos de fuerza bruta

1.10 Protocolos de seguridad en redes de transmisión de datos

1.10.1 IP security (IPsec)

La IETF ha sabido por años que hay una falta de seguridad en Internet. Agregarla no era fácil pues surgió una controversia sobre dónde colocarla. La mayoría de los expertos en seguridad creían que para estar realmente seguro, el cifrado y las verificaciones de integridad tenían que llevarse a cabo de extremo a extremo (es decir, en la capa de aplicación). De tal manera, el proceso de origen encripta y/o protege la integridad de los datos y los envía al proceso de destino en donde se desencriptan y/o verifican. Por lo tanto, cualquier alteración hecha en medio de estos dos procesos, o en cualquier sistema

operativo, puede detectarse. El problema con este enfoque es que requiere cambiar todas las aplicaciones para que estén conscientes de la seguridad. Desde esta perspectiva, el siguiente mejor enfoque es colocar el cifrado en la capa de transporte o en una nueva capa entre la capa de aplicación y la de transporte, con lo que se conserva el enfoque de extremo a extremo pero no requiere que se cambien las aplicaciones.
La perspectiva opuesta es que los usuarios no entiendan la seguridad y no sean capaces de utilizarla correctamente, así como que nadie desee modificar los programas existentes de ninguna forma, por lo que la capa de red debe autentificar y/o cifrar paquetes sin que los usuarios estén involucrados. Después de años de batallas encarnizadas, esta perspectiva ganó soporte suficiente para que se definiera un estándar de seguridad de capa de red. El argumento fue en parte que tener cifrado de la capa de red no evitaba que los usuarios conscientes de la seguridad la aplicaran correctamente y que ayudara hasta cierto punto a los usuarios no conscientes de ella.
El resultado de esta guerra fue un diseño llamado IPsec (Seguridad IP), que se describe en los RFCs 2401, 2402 y 2406, entre otros. No todos los usuarios desean cifrado (pues éste es costoso computacionalmente). En lugar de hacerlo opcional, se decidió requerir cifrado todo el tiempo pero permitir el uso de un algoritmo nulo. Éste se describe y alaba por su simplicidad, facilidad de implementación y gran velocidad en el RFC 2410.

"IPsec no es un simple protocolo de seguridad. En cambio, IPsec provee un set de algoritmos de seguridad más un framework que autoriza a un par de entradas de comunicación al uso de cualquier algoritmo que provee una apropiada seguridad para las comunicaciones." [60]

IPsec define un mínimo set de algoritmos de seguridad que son obligatorios (y que toda implementación tiene que suministrar). En cada caso, el estándar define un uso específico. En la figura 1.3 lista los algoritmos requeridos de seguridad**".** [60]

Authentication

HMAC with MD5	RFC 2403
HMAC with SHA-1	RFC 2404

Encapsulating Security Payload

DES in CBC mode	RFC 2405
HMAC with MD5	RFC 2403
HMAC with SHA-1	RFC 2404
Null Authentication	
Null Encryption	

Figura 1.3: Algoritmos de seguridad que son obligatorios para IPsec

1.10.2 Tunel IPsec

La tecnología VPN usa encriptación a lo largo de un túnel IP-a-IP al tener una transferencia confidencial de un sitio al otro. IPsec es especificado al diseñar y acomodar un túnel de encriptación. En particular, la norma define la versiones de túnel de ambos el cabezal de autenticación y la seguridad de encapsulado. En la figura 1.4 se puede observar las capas de datagramas en el modo túnel.

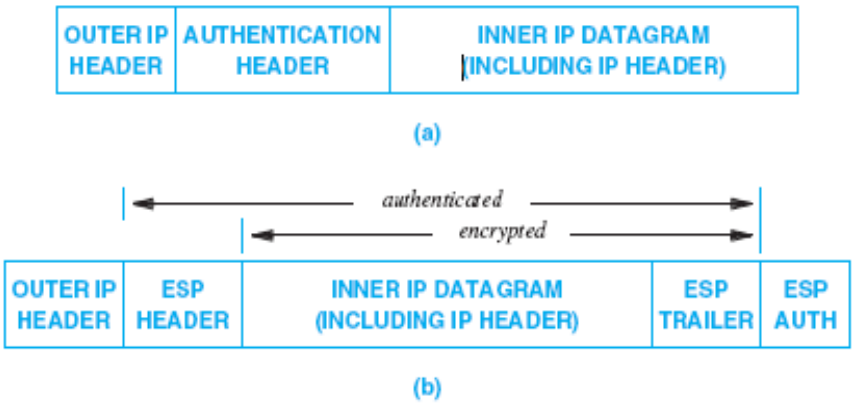


Figura 1.4: IPsec en modo túnel para (a) una autenticación de datagramas y (b) un encapsulado. Los datagramas enteros son protegidos

1.10.3 Secure Socket Layer (SSL and TLS)

Por los mediados de los años 90 cuando la seguridad de Internet era importante para realizar negocios y transacciones, varios grupos propusieron mecanismo de seguridad para ser utilizados con la utilización de las páginas Web. Aunque no estaban formalmente adoptados por el IETF, una de las propuestas que se realizo fue el factor estándar.

Conocido como Secure Sockets Layer (SSL), la tecnología fue originalmente desarrollada por Netscape, Inc. Como el nombre implica SSL reside en la misma capa del socket API. Cuando un cliente usa SSL al contactar un servidor, el protocolo SSL permite que cada lado autentique a sí mismo con el otro. Los dos lados entonces negocian al seleccionar un algoritmo de encriptación que ellos dos pueden soportar. Finalmente, SSL permite que los dos lados establezcan una conexión encriptada (una conexión que usa la selección del algoritmo de encriptación al garantizar la privacidad).

TLS está en una capa de protocolo que corre en una capa de transporte fiable, típicamente TCP (Transmission Control Protocol). Otras aplicaciones de protocolos, semejantes como HTTP (Hypertext Transfer Protocol) e IMAP (Internet Message Access Protocol) puede correr encima de TLS. TLS es una aplicación independiente, y es usado al proveer seguridad a cualquier de las dos aplicaciones de comunicación que transmiten datos a través de una red, vía un protocolo de una aplicación. Este también puede ser usado al crear una VPN (Virtual Private Network) que conecta un sistema externo a un sistema interno de red, permitiendo al sistema acceder a múltiples servicios y recursos internos como si estuvieran en la propia red.[61]

1.11 Tecnologías de hardware para la virtualización

1.11.1 Un Cluster con Proxmox

Un clúster con Proxmox consiste de dos o más nodos de computadoras con Proxmox como sistema operativo y conectado en la misma red. Una máquina virtual puede ser migrar desde un nodo al otro en el mismo clúster, lo cual permite redundancia si falla un

nodo por alguna razón. En La figura 1.5 se puede observar un clúster básico con Promox en dos-nodos con FreeNAS almacenamiento compartido. [63]

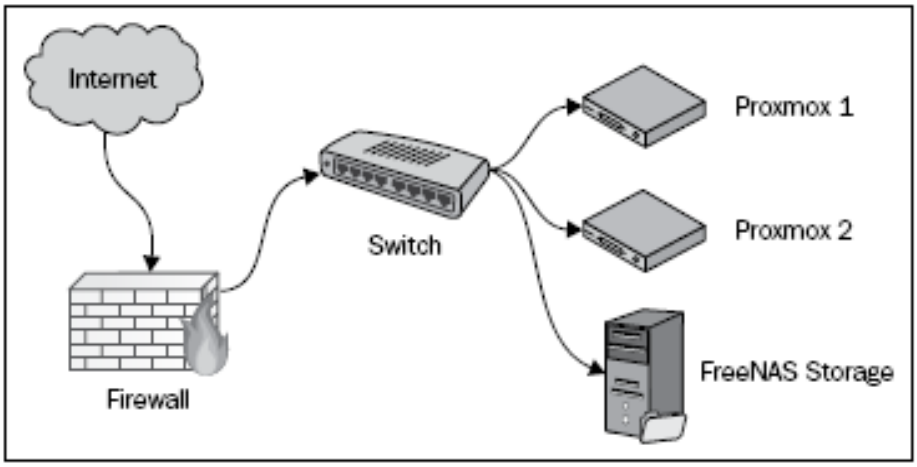


Figura 1.5: Básico clúster Proxmox de dos-nodos en Freenas Almacenamiento compartido

1.11.2 La lista de hardware

La siguiente es una lista de componentes del hardware (Ver Anexo 12), que son necesarias para colocar el primer clúster básico con Proxmox. Es importante siempre saber si el hardware con que se cuenta soporta la virtualización. No todas las plataformas de hardware soportan virtualización, especialmente si ellas son de tecnologías completamente obsoletas, para más detalles en como chequear los componentes se puede acceder a:
http://virt-tools.org/learning/check-hardware-virt/.

Una rápida vía al chequear es a través del BIOS y echar una mirada en las siguientes configuraciones de las opciones del BIOS. Algunas de estas deben ser habilitadas para que el servidor pueda utilizarse para la virtualización.

- Intel ® Virtualization Technology
- Virtualization Technology (VTx)
- Virtualization

Esta es una lista de hardware para construir un clúster básico con Proxmox para un determinado propósito, esta no es la infraestructura apropiada para una red de Clase-Empresarial. [64]

1.11.3 Configuración del hardware

El diagrama que se puede observar en el Anexo 13 es el diagrama de red para un clúster básico con Proxmox. Debe ser iniciado con dos nodos clústeres con un almacenamiento compartido (shared storage) configurado con cualquiera de los dos Ubuntu o FreeNAS. La configuración que se puede observar en el (Anexo 13) es una guía solamente. Dependiendo del nivel de experiencia, presupuesto, y accesibilidad de hardware a mano,

usted puede realizar la configuración por cualquier vía apropiada. Sin importar la vía que se utilice para la configuración, de encontrarse con los siguientes requerimientos mínimos: [65]

- Dos nodos con Proxmox con dos interfaces de red
- Un almacenamiento compartido (shared storage) con NFS y conectividad ISCSI
- Un cortafuego (firewall) físico
- Un switch con 8 puertos físicos
- Una máquina virtual con KVM
- Un OpenVZ/container machina

Conclusiones del Capítulo 1

Hasta este punto se definieron los requisitos necesarios con que debe contar un sistema de seguridad para el ámbito empresarial, además se seleccionaron un grupo de herramientas de software de libre que van a permitir mejorar la gestión, el funcionamiento y la seguridad de la red de datos, finalmente se realizó un estudio de las tecnología de hardware necesarias para dar inicio al proceso de virtualización en la División.

CAPÍTULO II DISEÑO DEL SISTEMA DE SEGURIDAD

En este capítulo se hará un completo análisis de la infraestructura actual de la red de la División Desoft Villa Clara, para identificar las diferentes aplicaciones, servicios y prestaciones en la red de datos. Esto permitirá realizar una valoración de la infraestructura existente en cuanto a revisión, análisis y reconocimiento del sistema existente, además del análisis de la actual topología, conectividad y arquitectura de la red LAN. La seguridad en redes es un tema bastante amplio, por esa razón las tecnologías utilizadas para ello son diversas, por lo cual este trabajo se centrará en un diseño seguro para la red LAN, las tecnologías de hardware para servidores y dispositivos de interconexión, plataforma de software libre para el proceso de virtualización, herramientas de software libre para la detección de vulnerabilidades en la red, mecanismos de seguridad y protocolos de seguridad.

2.1 Análisis de la infraestructura actual de la red

La División Desoft Villa Clara tiene actualmente más de 100 usuarios, las mayorías insertados directamente a la producción, desarrollando y desplegando aplicaciones informáticas, entre los que se encuentran especialistas y directivos.
La División cuenta con un total de 180 PC, de ellas 100 conectadas a los diferentes servicios de la red de la sede central y 80 que se encuentran en los municipios, desde estos solo cuatro municipios se conectaran a la división para utilizar el servicio de correo electrónico mediante un modem. La tecnología para servidores existente en la empresa son 11 PCs de escritorio que hacen esta función, estas últimas no cuentan con los requerimientos y prestaciones necesarios para comenzar el proceso de virtualización, que sería lo ideal para un menor consumo de energía, mayor resguardo y disponibilidad de la información.
Además debe tenerse en cuenta las particularidades del Departamento de Desarrollo, donde se utilizan diferentes aplicaciones que requieren de servidores con buenos requerimientos de hardware para el procesamiento, manejo y almacenamiento de grandes cantidades de datos. Por citar un ejemplo: los gestores de bases de datos más utilizados son: SQLServer, PostgreSQL, Oracle, MySQL, para el desarrollo en ambas plataformas Linux y Windows. Por otra parte hay que tener presente el futuro crecimiento de la empresa.
La utilización de tecnología profesional para los servidores no solo facilitaría el trabajo de gestión de los administradores de red, sino que permitiría que algunos servicios que brinda la división y que requieren de servidores con muy buenas prestaciones puedan ejecutarse sobre estos últimos sin ningún tipo de dificultad. Además permitiría implementar un sistema de Backup (Salva) mucho más eficiente del que existe actualmente, ya que sería incorporada la tecnología RAID. También es necesaria la adquisición de un Switch-L3, ya que la tecnología con que se cuenta son Switch-L2, con un Switch-L3 se podría segmentar la red creando VLANs y disminuir así el dominio de colisiones, esto ayudaría a mejorar la integridad, confidencialidad, disponibilidad para cada usuario conectado a la red de datos. Es por eso que se propone adquirir una tecnología para servidores y dispositivos de interconexión que satisfaga las necesidades y prestaciones con que debe contar una empresa profesional como Desoft.

2.2 Topología y servicios de la Intranet en la red de datos División Desoft Villa Clara

En la División, se cuenta con una red local (LAN) con topología en estrella con Switch en cascada, el protocolo de comunicación empleado es el ETHERNET + TCP/IP.
El acceso a la red corporativa de DESOFT, es mediante una línea arrendada de 512 Kbps y la navegación directa de Internet con 256 Kbps desde la División con el proveedor ENET. Cuenta con un MÓDEM-Router Huawei, y un MÓDEM-Router TP-Link y con un cortafuego para protegerse de la red externa e interna, los servidores se encuentran montados en una DMZ. Existe un cable de fibra óptica desde el local de Desarrollo hasta un Switch ubicado en el nodo. (Ver Anexo 14 y 15)

2.3 Metodología para el diseño de la red corporativa

Existen muchos tipos de redes: entre ellas pueden ser citadas redes de área local (LAN), redes de área amplia (WAN), redes de área metropolitana (MAN), redes de campus de área (CANs), las redes Ethernet, redes Token Ring, Redes Fiber Distributed Data Interface (FDDI), modo de transferencia asíncrono (ATM), las redes Frame Relay, redes T1, redes DS3, redes de puentes, redes enrutadas y redes punto a punto, por nombrar algunas. En este apartado se realizará la reestructuración de la red LAN de datos de la División, llevando a la práctica para ello la teoría del modelo jerárquico (Ver Anexo 16), la cual permitirá de forma escalonada un diseño más seguro y eficiente, a partir del análisis por capas, siempre teniendo en cuenta los servicios red que se brindan la empresa, la topología de esta última y los dispositivos de interconexión.

2.3.1 Ventajas de utilizar el modelo jerárquico

A continuación se resumen algunas ventajas del modelo jerárquico:

- Ahorro en los costos
- Reduce la carga en los dispositivos de red: evita que los dispositivos tengan que comunicarse con demasiados dispositivos similares (reduce las “adyacencias de CPU”).
- Limita los dominios de broadcast
- Aumenta la simplicidad y la comprensión
- Facilita los cambios de red
- Facilita el escalamiento a un tamaño mayor
- Mejora la aislamiento de fallas

2.3.2 Capa Central (Core)

Esta capa es el backbone con conmutación de alta velocidad, la cual es crucial para permitir comunicaciones corporaciones, para el que el diseño de esta capa sea el adecuado debe proporcionar las siguientes características:

- Es el backbone de conmutación de la red de alta velocidad.
- Alta confiabilidad.
- Redundancia.
- Tolerancia a fallos.
- Rápida adaptación a cambios.
- Ofrecer baja latencia y buen nivel de gestión.
- Permitir la manipulación mediante filtros.
- Poseer un diámetro limitado y consistente.

2.3.3 Capa de Distribución

La capa de distribución está ubicada entre la capa Central y la de Acceso. Para exista un desempeño en la capa de distribución debe permitir realizar las siguientes funciones:

- Políticas.
- Seguridad.
- Direccionamiento.
- Definición de dominios de broadcast y multicast.
- Enrutamientos entre VLANs.
- Traducción de medios (entre ethernet y token ring)

2.3.4 Capa de Acceso

- La capa de acceso provee acceso a los usuarios al segmento locales de la red.
- Ofrece ancho de banda compartido y conmutado. Micro segmentación utilizando Switches.
- Acceso a usuarios remotos a través de tecnologías WAN como ISDN, frame relay, RAS, WIFI y líneas dedicadas.

2.3.5 Metodología para el diseño de la red

Los pasos para la reestructuración de la red se exponen a continuación:

1. Obtener información para soportar los requerimientos técnicos y negocios.
2. Información sobre la red actual.
3. Deben ser consideradas las aplicaciones involucradas.
4. Diseñar la red LAN.
5. Diseñar para protocolos específicos.
6. Crear el documento de diseño y seleccionar las aplicaciones de administración.
7. Probar el diseño.(Ver Anexo 17)

Siguiendo la metodología antes expuesta y teniendo en cuenta la información con que se cuenta en cuanto a: estructura de la red actual de la División Desoft Villa Clara, dispositivos de interconexión, servicios que se brindan en la red, futuro crecimiento y centralización de las cuentas de usuarios en los servidores de la Oficina Central en la ciudad Habana, es que se concibió el diseño de red que puede ser observado en los Anexo 18 y 19. Las pruebas preliminares de las mejoras de seguridad en la red serán realizadas a través de la herramientas de sobre software libre muchas de ellas contenidas en Kali Linux, aprovechando así la potencialidad de estas para pruebas de test y penetración, y los servicios serán supervisados constantemente a través de herramientas tales como Nagios entre otras.

2.4 Selección de las tecnologías de hardware para la virtualización

Como se mencionó con anterioridad todos los servicios de la red de datos se encuentran corriendo sobre PCs de tecnología obsoleta que no permiten emprender el proceso de virtualización sobre ninguna de ellas, el mejor de estos servidores cuenta con las siguientes características técnica: INTEL SERVER BOARD S3000AH, DualCore Intel Xeon 3050, 2133 MHz (8 x 267), dos memorias de 2 GB DDR2-800 DDR2 CHC, dos discos duros de 500 GB.
Los requerimientos mínimos de hardware para cada uno de los servicios que se ofrecen en la división actualmente pueden observase en el (Anexo 20), esta relación se obtuvo a partir de la práctica diaria y de pruebas en tiempo real.
A partir de lo mencionado con anterioridad y de una valoración de precios de diferentes proveedores es que se formaliza una propuesta para comprar de tecnología necesaria para el proceso de virtualización en la empresa (ver Anexo 21) son expuestas las características técnicas del hardware y así como el costo aproximado. [6]

2.5 Principios de seguridad para el diseño de entornos virtuales seguro

2.5.1 Seguridad en entornos de virtualización

Las máquinas virtuales (virtual machines), a diferencia de un equipo físico, están reducidas a un simple archivo; que si bien representa flexibilidad para el administrador, también significa una vulnerabilidad que puede ser explotada para robar la máquina completa, incluyendo su contenido. Recordemos que en los entornos virtuales, varias máquinas virtuales pueden compartir una sola interfaz física (Ver Anexo 22), en consecuencia, dichos equipos pueden ser víctimas de diversos tipos de ataques entre una máquina virtual y otra residente en el mismo equipo físico.
Por otro lado, la seguridad virtual se extiende más allá de las máquinas virtuales, por ejemplo, los sistemas de almacenamiento en red se ve expuestos a amenazas y constituyen otra línea de acción para los atacantes. Una recomendación es mantener los sistemas de almacenamiento separados del resto de las máquinas virtuales.
En un esquema virtual, en donde se utilizan equipos para ejecutar las tareas de procesamiento de las máquinas virtuales (y su almacenamiento se encuentra en un almacenamiento de red SAN), es fácil ver cómo se ve comprometido todo el sistema de almacenamiento cuando no se contemplan este tipo de riesgos, sobre todo al momento

de la instrumentación de entornos virtuales basados en sistemas de almacenamiento separado.
En este tipo de esquemas de operación, existe un servidor denominado “Servidor de procesamiento” que puede contener una o varias máquinas virtuales y un “Sistema de almacenamiento” (por ejemplo, uno del tipo SAN). Este sistema es un equipo físico separado del servidor de procesamiento, cuya función es alojar los archivos de cada una de las máquinas virtuales a través de interfaces, ya sea de tipo iSCSI o Fiber Channel. Al interconectarse con el servidor de procesamiento, utiliza canales de comunicación que nuevamente quedan vulnerables ante cualquier posible ataque (Ver Anexo 23).

2.5.2 Aplicación de Seguridad en entornos virtuales

Sabnis, S., Verbruggen, M., Hickey, J. and McBride, A. J. (2012), hacen mención de algunos aspectos para la aplicación de seguridad en entornos virtuales al inicio de su diseño, por ejemplo: clasificación del tráfico e información real entre máquinas virtuales, mecanismos de autentificación y controles de acceso robustos, controles para el acceso y la operación, corrección de vulnerabilidades e instalación de actualizaciones de seguridad, así como configuración de auditoría y escaneo de vulnerabilidades.

Se debe considerar la utilización de VLANs para la separación del tráfico entre máquinas virtuales, lo que permitirá cierto nivel de aislamiento entre cada una de ellas. La utilización de firewalls personales en cada una de las máquinas también constituye una línea de defensa, puede administrar el tráfico de red permitido desde y hacia cada una de las máquinas. Otra opción es el empleo de switches virtuales, éstos pueden segmentar la red y controlar el tráfico, sobre todo cuando varias máquinas virtuales hacen uso de una sola interfaz física (Ver Anexo 24). Mantener actualizados los sistemas también representa un menor riesgo en ambientes virtuales.

Se puede hacer uso de herramientas para ayudar a resolver problemas de seguridad virtual, tanto en entornos virtuales puros como en mixtos. Para asegurar los entornos virtuales al igual que cualquier componente físico de TI, se debe comenzar con un plan de instrumentación de seguridad para los entornos virtuales. Un buen punto de inicio es consultar a los principales proveedores de soluciones, que son unos de los primeros involucrados en el tema debido a la relevancia que tiene la seguridad en ambientes virtuales. El análisis de vulnerabilidades en entornos virtuales, políticas, tecnologías y mejores prácticas, así como tomar en cuenta que este tipo de entornos no operan de igual forma que los físicos. Sin embargo, al igual que en éstos últimos, existen herramientas que ayudan a proteger y mantener la integridad de los entornos virtuales a través de una buena administración de la seguridad virtual.

2.6 Conocimientos necesarios para la instalación de Proxmox

2.6.1 Software necesarios para la instalación de un clúster

Para la instalación de Proxmox es necesario descargar los software que van ser utilizados en la virtualización, las URL de cada sitios pueden ser observadas en la en la tabla 2.

Software	Download link
Proxmox VE	http://proxmox.com/downloads
FreeNAS	http://www.freenas.org/download-releases.html
Ubuntu Server	http://www.ubuntu.com/download
clearOS community	http://www.clearfoundation.com/Software/downloads.html

Tabla 2.1 Software para la Instalación de un clúster

2.6.2 Pasos para la instalación de nodos con Proxmox

Utilizando el diagrama de red del Anexo 12 se realizaran los siguientes pasos personalizados para realizar un ejemplo de instalación en entorno de la División Desoft Villa Clara.

1. Se ensamblaran los tres nodos con sus debidos componentes, y se conectaran todos ellos con un Switch LAN.
2. Encender el primer nodo y acceder al BIOS para realizar los cambios necesarios, tal como habilitar la virtualización.
3. Se pondrá a iniciar el nodo desde el disco de instalación Proxmox.
4. El siguiente proceso transcurrirá a lo largo de instalación grafica de Proxmox. Se entrara la dirección 192.168.50.1 Se definirá el dominio pxserver01.vcl.desoft.cu, o cualquier otro hostname que sea seleccionado.
5. Ejecutar el paso 3 y 4 para el segundo nodo. Se utilizara la dirección IP 192.168.50.2, o cualquier otra que se vaya a utilizar. Aquí se utilizara el dominio pxserver02.vcl.desoft.cu, o cualquier otro hostname.

2.6.3 Creación de un Clúster

Linux utiliza el siguiente comando para una realizar el logue y conexión segura en el nodo con Proxmox:

ssh root@192.168.50.1

Para crear el clúster se utilizará el siguiente comando:

root@pxserver01:~# pvecm create pxcluster-nodo1

Con el siguiente comando se puede chequear el estado de creación del clúster:

root@pxserver01:~# pvecm status

Después de ser creado el clúster, el próximo paso es adicionar los nodos Proxmox con el comando que sigue:

root@pxpxserver02:~# pvecm add 192.168.50.1

Seguidamente se verificara que el nodo este trabajando ahora con el clúster con el siguiente comando:

root@pxpxserver02:~# pvecm nodes

El próximo paso sería iniciar sección a través de la Web Proxmox GUI y ver los clústeres y juntos al almacenamiento compartido. Se utilizara la siguiente URL en un explorador para acceder a la interfaz gráfica:

https://<ip_proxmox_node>:8006

2.6.4 La interfaz gráfica de Proxmox (GUI)

La GUI (Graphical User Interface,) para Proxmox o Proxmox GUI, permite a los usuarios interactuar con el clúster de Proxmox gráficamente usando menús y una representación visual de los estados del clúster. Pero eventualmente todas las acciones de administración se pueden realizar desde líneas de comando, Command-line Interface (CLI), esto puede ser una tarea difícil y engorrosa, esto tomaría más tiempo de lo que normalmente se necesita con una interfaz gráfica. Al utilizar correctamente clúster de Proxmox, es muy importante tener claramente conocimientos de Proxmox GUI. La GUI puede ser fácilmente accedida desde el explorador que usted desee, escribiendo una URL similar a: https://192.168.1.1:8006 Como se puede observar en el Anexo 25.

En el Anexo 25 fueron marcados varios campos, seguidamente se expondrá el significado de cada uno de ellos:

- (1) URL de acceso a la interfaz de Proxmox GUI a través de un explorador.
- (2) El botón de logout al salir la interfaz de Proxmox GUI.
- (3) El botón al abrir la máquina virtual y creación de la caja de dialogo.
- (4) El botón al abrir la OpenVZ container y creación de la caja de dialogo.
- (5) La barra de menú Proxmox tabbed.
- (6) El menú drop-down al cambia el periodo del estado gráfico.
- (7) Los estados de los bloques de información para los nodos de Proxmox, máquinas virtuales, o contenedores.
- (8) Los containers OpenVZ.
- (9) Plantillas de máquinas virtuales accesibles para clonar.
- (10) Máquina virtual de KVM.
- (11) Nodos de Proxmox.
- (12) Almacenamientos compartidos.
- (13) Recursos de pools.
- (14) La representación gráfica de varios estatutos.
- (15) Se pueden observar las tareas de log[78]

2.7 Herramientas sobre software libre para el modelado del sistema de seguridad

2.7.1 Instalación de Kali

El Kali Linux puede ser instalado en un disco duro como cualquier distribución GNU/Linux, también puede ser instalado y configurado para realizar un arranque dual con un Sistema Operativo Windows, de la misma manera puede ser instalado en una unidad USB, o instalado en un disco cifrado. [79]

Para información detallada sobre las diversas opciones de instalación para Kali Linux, en la siguiente página: http://docs.kali.org/category/installation/.

2.7.1.1 Requisitos previos de instalación

La instalación de Kali necesita:

Un mínimo de 8 GB de espacio en disco para la instalación de Kali Linux.

Para las arquitecturas i386 y amd64, un mínimo de 512 MB de RAM.

CD-DVD Drive / Soporte de arranque mediante USB

http://docs.kali.org/

Luego de contar con la instalación que puede ser descargada desde:

http://es.docs.kali.org/downoading-es/descarga-imagenes-oficiales-de-kali

Kali es instalado como cualquier otra distribución Linux.

2.7.1.2 Configuración de la red

Para indicarle una dirección IP estática a la máquina virtual con Kali, es necesario configura su interfaz de red, para ello se debe proceder de la siguiente forma:

- Entrar en un Terminal.
- Utilizando el editor favorito y editar el archivo interfaces que se encuentra dentro de /etc/network/.
- Seleccionar la interfaz de red eth0, se pudiera ver utilizado cualquiera con la que contara esta máquina.
- Luego de indicar a **eth0** que se va comportar como estática editar los parámetros:

Address

Netmask

Network

Broadcast

Gateway

En la figura 2.1 se pude observar un ejemplo de la configuración de red, de una dirección IP dinámica a una dirección IP estática.

Original

```
auto lo
iface lo inet loopback

auto eth0
iface eth0 inet dhcp

auto eth1
iface eth1 inet dhcp

auto eth2
iface eth2 inet dhcp

auto ath0
iface ath0 inet dhcp

auto wlan0
iface wlan0 inet dhcp
```

Edited

```
auto lo
iface lo inet loopback

# primary network interface
# edited to maintain persistent state
auto eth0
iface eth0 inet static
     address 192.168.204.128
     netmask 255.255.255.0
     network 192.168.204.0
     broadcast 192.168.204.255
     gateway 192.168.204.1

auto eth1
iface eth1 inet dhcp

auto eth2
iface eth2 inet dhcp

auto ath0
iface ath0 inet dhcp
```

Figura 2.1: Edición del archivo interfaces

- Finalmente debe ser activada y reiniciada la interfaz de red con los siguientes comando:

  root@kali~# update-rc.d networking defaults

  root@kali~# /etc/init.d/networking restart

2.7.1.3 Paquetes y repositorios

Por defecto, Kali usa solamente los repositorios oficiales. Es posible que un proceso incompleto de instalación no permita adicional las fuentes de los repositorios correctamente. [80]

El archivo source.list puede ser editado con el siguiente comando:

(echo debhttp://http.kali.org/kiali kali main contrib non-free >> /etc/apt/sources.list), o por el uso de un editor de texto.

Por defecto los paquetes en el repositorio deben mostrarse en: /etc/apt/sources.list

La lista se deben presentarse como sigue, en caso que no se encuentren en el archivo, será necesario editar el archivo sources.list y adicional:

Kali

deb http://http.kali.org/kali kali main contrib non-free

Kali-dev

deb http://http.kali.org/kali kali-dev main contrib non-free

Kali Security updates

deb http://security.kali.org/kali-security kali/updates main

contrib non-free

2.7.1.4 Interactuando con la interfaz gráfica de Kali

Cuando se accede a la interfaz gráfica de Kali, por defecto instala una interfaz gráfica con una barra y unos pocos iconos, seleccionado el menú Applications, y posteriormente Kali Linux, pueden ser observada serie de aplicaciones organizadas de acuerdo a la función que realizan: (Ver Anexo 26)

Entre las herramientas que serán utilizadas para la búsqueda de posibles problemas de seguridad pueden citarse: las destinadas al análisis vulnerabilidades, por ejemplo OpenVas, el cual tiene un servicio de actualizaciones diarias de los test de vulnerabilidad de red (Network Vulnerability Test, también llamados NVT o pluggins), para escaneo de puertos va a ser utilizado el: Nmap, o en su versión grafica el Zenmap, y como analizador de protocolos se hará uso del: Wireshark , entre otras más que trae Kali. La mayoría de estas herramientas puede ser instalada de forma independiente desde los repositorios de muchas de las distribuciones Linux, entre ellas pueden ser mencionadas Debian y Ubuntu.

2.7.1.5 Tcpdump

Esta herramienta de supervisión que funciona desde la línea de comandos se utiliza con mayor frecuencia para:

- Depurar aplicaciones que utilizan la red para comunicar.
- Depurar la red misma.
- Capturar y leer datos enviados por otros usuarios u ordenadores. Algunos protocolos como telnet y HTTP no cifran los datos que envían en la red. Un usuario que tiene el control de un router a través del cual circula tráfico no cifrado puede usar tcpdump para conseguir contraseñas u otras informaciones

2.7.1.5.1 Principales parámetros

- -A: Imprime cada paquete en código ASCII
- -D: Imprime la lista de interfaces disponibles
- -n: No convierte las direcciones de salida
- -p: No utiliza la interfaz especificada en modo promiscuo
- -t: No imprime la hora de captura de cada trama
- -X: Imprime cada paquete en hexadecimal y código ASCII
- -c count: Cierra el programa tras recibir 'count' paquetes
- -i interface: Escucha en la interfaz especificad

2.7.1.5.2 Como filtrar el tráfico en la red

- type [host|net|port]: Máquina en particular [host], red completa [net] o puerto concreto [port].

- dir [src|dst|src or dst|src and dst]: Especifica desde [src] o hacia dónde [dst] se dirige la información.
- proto [tcp|udp|ip|ether]: Protocolo que queremos capturar.

2.7.1.5.3 Como interpretar la salida

- src > dst: flags [dataseq ack window urgent options]
- 15:23:44.772291 IP 192.168.1.17.52798 > 85.Red-83-37-170.dynamicIP.rimatde.
- net.65000: ack 1791 win 7851 <nop, nop, timestamp 5520421 997821>
- 15:23:44772291: Indica hh:mm:fracciones
- src: Dirección y puerto origen.
- dst: Dirección y puerto destino.

2.7.1.5.4 Ejemplos de los principales usos

Capturar tráfico cuya dirección IP de origen sea 192.168.3.1

tcpdump src host 192.168.3.1

Capturar tráfico cuya dirección origen o destino sea 192.168.3.2

tcpdump host 192.168.3.2

Capturar tráfico con destino a la dirección MAC 50:43:A5: AE: 69:55

tcpdump ether dst 50:43:A5:AE:69:55

Capturar tráfico con red destino 192.168.3.0

tcpdump dst net 192.168.3.0

2.7.1.6 IPTraf

IPTraf es una utilidad de consola para Linux que proporciona estadísticas sobre el tráfico IP de las redes que se encuentren activas en el sistema. Es capaz de brindar mucha información como el número de paquetes y bytes en una conexión TCP, estadísticas de una interfaz e indicadores de actividad, caídas en el tráfico TCP/UDP y número de bytes y paquetes en una estación LAN.

2.7.1.6.1 Instalación y ejecución de IPTraf

Para comenzar a utilizar IPTraf, se lo debe instalar primero; lo que en un sistema basado en Debian o Ubuntu es tan simple como alguno de los siguientes comandos:

apt-get install iptraf

aptitude install iptraf

Una vez instalado el programa, uno puede ejecutarlo para supervisar la red.

Desafortunadamente, se necesitan de permisos de superusuario para hacer esto. En Ubuntu, significa agregar la palabra sudo antes de la orden

sudo iptraf

Una vez ejecutado, el usuario verá un menú donde puede elegir comenzar el monitoreo local o el tráfico a través de cualquiera de las tarjetas de red instaladas en la máquina.

2.7.1.7 Wireshark

Es una herramienta interactiva de monitoreo de tráfico de la red.

2.7.1.7.1 Principales características

- Disponible para Linux y Windows
- Captura de paquetes en vivo desde una interfaz de red
- Muestra los paquetes con información detallada de los mismos
- Abre y guarda paquetes capturados
- Importa y exporta paquetes en diferentes formatos
- Filtrado de información de paquetes
- Resaltado de paquetes dependiendo el filtro
- Creación de estadísticas

2.7.1.7.2 Principales componentes de la interfaz grafica

Barra de herramientas: Muestra todas las opciones a realizar sobre la captura datos.

- Barra de herramientas principal: Están las opciones más usadas en Wireshark.
- Barra de filtros: Área donde se aplican filtros a la captura actual de manera rápida
- Listado de paquetes: Muestra un resumen de cada paquete que es capturado por Wireshark
- Panel de detalles de paquetes: Una vez seleccionado un paquete en el listado de paquetes, muestra información detallada del mismo.
- Panel de bytes de paquetes: Muestra los bytes del paquete seleccionado, y resalta los bytes correspondientes al campo seleccionado en el panel de detalles de paquetes.
- Barra de estado: Breve información acerca del estado actual de Wireshark y la captura.

2.7.2 Instalación de IDS

2.7.2.7 Instalación y configuración de Psad

El servicio es un detector de intrusos y un analizador de logs de iptables.

2.7.2.7.1 Principal información que proporciona

Psad hace uso de los mensajes de log de Netfilter para la detección, alerta y (opcionalmente) scanning de puertos de tráfico sospechoso. El psad usa flags de TCP scans y analiza el TCP para determinar el tipo de scan (syn, fin, xmas u otros) y opciones de línea de comando correspondiente que pueden ser suministradas por el Nmap para generar un barrido. Además psad hace uso de las firmas que figuran en el sistema de detección de intrusos, por ejemplo Snort.

2.7.2.7.2 Instalación y Configuración

Para instalar el psad en Debian/Ubuntu o derivado de estos sistemas, se debe proceder de la siguiente de la siguiente forma:

 # apt-get install psad

Configurar el psad:

Abrir y editar el archivo */etc/syslog.conf* con el editor favorito:

vim /etc/syslog.conf

Posteriormente será adicionada la siguiente línea:

color=#000000]
Kern.info | /var/lib/psad/psadfifo
[color]

Alternativamente, puede se escribirá el siguiente comando para actualizar el syslog.conf:

 # echo -e "kern.info | /var/lib/psad/psadfifo" >> /etc/syslog.conf

El psad necesita ser configurado para grabar todos los mensajes kern.info en Syslog por un tubo en /var/lib/psad/psadfifo.

Ahora se procederá a cerrar y guardar el archivo. Posterior a la operación anterior se procederá a reiniciar el Syslog:

 # /etc/init.d/sysklogd restart
 # /etc/init.d/klogd

El archivo de configuración se encuentra en /etc/psad/psad.conf:

 # vim /etc/psad/psad.conf

Aquí es necesario definir un email para recibir las notificaciones cuando haya un scanning de puertos u otras configuraciones:

[color=#000000]
EMAIL ADDRESSES administrador@vcl.desoft.cu
[/color]

Definir hostname del host(FQDN):

[color=#000000]
HOME_NET NOT_USED;
[/color]

Para ajustar los niveles de seguridad en la configuración, se pueden definir un conjunto de puertos para ignorar, por ejemplo el psad puede ignorar los puertos UDP 53 y 5000. Entonces se utilizara la siguiente sintaxis:

[color=#000000]
IGNORE_PORTS udp/53, udp/5000;
[/color]

También se puede habilitar iptables para bloquear en tiempo real, definiendo dos variables:

[color=#000000]
ENABLE_AUTO_IDS Y;
IPTABLES_BLOCK_METHOD Y;
[/color]

El psad tiene muchas opciones. Para conocer más al respecto será necesario leer la documentación (man psad). Ahora se debe proceder a guardar y cerrar el archivo y posteriormente se reiniciará el servicio psad con el comando:

 # /etc/init.d/psad restart

2.7.2.8 Instalación y configuración de Arpwach

Es utilizado para detección de anomalías en direcciones MAC y corre bajo Linux.

2.7.2.8.1 Configuración

Una vez instalado el paquete Arpwatch del repositorio, debe ser editado el fichero de configuración, que está en /etc/arpwatch.conf, para que pueda supervisar la subred que se le indique y envíe las alertas. Para esto, deberá ser añadida en el fichero anterior, la siguiente línea: *eth0 -a -n 192.168.50.0/24*.
Obviamente, Arpwatch, o cualquier otro sistema similar, no puede supervisar más que la subred o subredes a la que pertenece el ordenador donde fue instalado, ya que los paquetes arp no saltan de VLAN en VLAN. Esto quiere decir que se necesita un Arpwatch por cada subred, así que se debe ser cuidadoso y tratar de minimizar el trabajo. Típicamente, debe ser instalado en las subredes más críticas, como puedan ser la de administración y sistemas.
Para que envíe las alertas a al administrador, existen muchas formas. Una de ellas es configurarlo para que envíe un mail al administrador, añadiendo al fichero /etc/arpwatch.conf una línea como la que siguiente:

eth0 -a -n 192.168.50.0/24 -m adminsitrador@vcl.desoft.cu

En el directorio /var/lib/arpwatch/ se va a encontrar la base de datos de este servicio anti-sniffers.

2.7.3 Instalación de IPS

2.7.3.7 Honeypots

Los “Honeypots”, literalmente “tarros de miel”, también conocidos como sistemas de decepción o engaño, constituyen una trampa -la otra acepción del término- para posibles atacantes. Un honeypot no es más que un sistema cuya única finalidad es la de ser probado, atacado o incluso comprometido.
Los Honeypots son una tecnología nueva con enorme potencial para la comunidad informática. Los primeros conceptos fueron introducidos por primera vez por varios íconos en la seguridad informática, especialmente aquellos definidos por Cliff Stoll y Bill Cheswick. Desde entonces, han estado en una continua evolución, desarrollándose de manera acelerada y convirtiéndose en una poderosa herramienta de seguridad hoy en día.
Un Honeypot es un sistema diseñado para analizar cómo los hackers emplean sus armas para intentar entrar en un sistema (analizan las vulnerabilidades) y alterar, copiar o destruir sus datos o la totalidad de éstos (por ejemplo borrando el disco duro del servidor). Por medio del aprendizaje de sus herramientas y métodos se puede, entonces, proteger mejor los sistemas. Pueden constar de diferentes aplicaciones, una de ellas sirve para capturar al intruso o aprender cómo actúan sin que ellos sepan que están siendo vigilados.
Los Honeypots son en su forma más básica son servidores de información falsos, posicionados estratégicamente en una red de prueba, los cuales son alimentados con información falsa que es disfrazada como archivos de naturaleza confidencial. A su vez, estos servidores son configurados inicialmente de manera que sea difícil mas no

imposible el hecho de ser penetrados por un atacante informático, exponiéndolos de manera deliberada y haciéndolos altamente atractivos para un “hacker” en busca de un blanco. Por último, el servidor es habilitado con herramientas de monitoreo y rastreo de información, de manera que cada paso y rastro de actividad de un “hacker” pueda ser registrado en una bitácora que indique esos movimientos de manera detallada.

Las funciones principales de un Honeypot son:

- Desviar la atención del atacante de la red real del sistema, de manera que no se comprometan los recursos principales de información.
- Capturar nuevos virus o gusanos para su estudio posterior.
- Formar perfiles de atacantes y sus métodos de ataque preferidos, de manera similar a la usada por una corporación policiaca para construir el archivo de un criminal basado en su modus operandi.
- Conocer nuevas vulnerabilidades y riesgos de los distintos sistemas operativos, entornos y programas las cuales aún no se encuentren debidamente documentadas.

En un contexto más avanzado, un conjunto de Honeypots forma una Honeynet, proporcionando así una herramienta que abarca un conjunto extendido de posibles amenazas y proporciona al administrador de sistemas mayor información para su estudio. Inclusive, hace más fascinante el ataque para intruso debido a que se incrementan las posibilidades, blancos y métodos de ataque.

2.7.3.8 Ubicación de los Honeypots

La ubicación de los Honeypots es esencial para maximizar su efectividad, ya que debido a su carácter intrínsecamente pasivo; una ubicación de difícil acceso eliminará gran parte de su atractivo para potenciales atacantes. Por otro lado, si su ubicación es demasiado artificial u obvia cualquier experimentado atacante la descubrirá y evitará todo contacto.

Se debe tener en cuenta que los Honeypots se debe integrar con el resto del sistema que se tiene implementado por ejemplo: servidores WWW, servidores de ficheros, DNS. De manera de asegurar que no interfiera con las otras medidas de seguridad que puedan ya existir en la red como Firewalls, IDS.

Los Honeypots pueden servir tanto para la detección de atacantes internos como externos, se debe tener siempre en cuenta la posibilidad de establecer Honeypots internos para la detección de atacantes o sistemas comprometidos en la red, por ejemplo sistemas infectados con gusanos o virus. Seguidamente serán resumidas algunas ubicaciones estratégicas de los Honeypots

2.7.3.8.1 Antes del cortafuego (Front of firewall)

Esta localización permitirá evitar el incremento del riesgo inherente a la instalación del Honeypot. Como este se encuentra fuera de la zona protegida por el firewall, puede ser atacado sin ningún tipo de peligro para el resto de la red (Ver Anexo 27)

Esta configuración evitara las alarmas de otros sistemas de seguridad de la red (IDS) al recibir ataques en el Honeypot. Sin embargo, existe el peligro de generar mucho tráfico debido precisamente a la facilidad que ofrece el Honeypot para ser atacado.

Cualquier atacante externo será lo primero que encuentra y esto generará un gran consumo de ancho de banda y espacio en los ficheros de log. Por otro lado, esta ubicación evita la detección de atacantes internos.

2.7.3.8.2 Detrás del cortafuego (Behind the firewall)

En esta posición, el Honeypot queda afectado por las reglas de filtrado del firewall. Por un lado se tiene que modificar las reglas para permitir algún tipo de acceso al Honeypot por posibles atacantes externos, y por el otro lado, al introducir un elemento potencialmente peligroso dentro de la red se puede permitir a un atacante que gane acceso al Honeypot y a la red. (Ver Anexo 28)

La ubicación tras el firewall permite la detección de atacantes internos así como firewalls mal configurados, máquinas infectadas por gusanos o virus e incluso atacantes externos.

Sin embargo las contrapartidas más destacables son la gran cantidad de alertas de seguridad que generarán otros sistemas de seguridad de la red (Firewalls, IDS). Al recibir ataques el Honeypot se ve la necesidad de asegurar el resto de nuestra red contra el Honeypot mediante el uso de firewalls extras o sistemas de bloqueo de acceso, ya que si un atacante logra comprometer el sistema tendrá vía libre en su ataque a toda la red.

Hay varias circunstancias que obligan a este tipo de arquitectura, como por ejemplo la detección de atacantes internos o la imposibilidad de utilizar una dirección IP externa para el Honeypot.

2.7.3.8.3 En la zona desmilitarizada (DMZ)

La ubicación en la zona desmilitarizada permite por un lado juntar en el mismo segmento a los servidores de producción con el Honeypot y por el otro controlar el peligro que añade su uso, ya que tiene un firewall que lo aísla de resto de la red local. (Ver Anexo 29)

Esta arquitectura nos permite tener la posibilidad de detectar ataques externos e internos con una simple reconfiguración del sistema de firewall puesto que se encuentra en la zona de acceso público.

Además se elimina las alarmas de los sistemas internos de seguridad y el peligro que supone para la red al no estar en contacto directo con esta.

La detección de atacantes internos se ve algo debilitada, puesto que al no compartir el mismo segmento de red que la LAN, un atacante local no accederá al Honeypot. Sin embargo, desde la red local si es posible acceder al Honeypot, con lo que un atacante interno que intente atacar a los servidores públicos u otros sistemas externos por ejemplo un gusano, muy probablemente acabe siendo detectado.

2.7.4 Instalación de herramientas de supervisión de servicios de red

2.7.4.7 Fase de Implementación de OSSIM

Para poder realizar la implementación de OSSIM es necesario tomar en cuentas los siguientes factores:

2.7.4.7.1 Requisitos técnicos

El requisito técnico más importantes es el hardware para instalar OSSIM AlienVault, este dependerá en gran medida del número de eventos que tenga que procesar el servidor, de la cantidad de datos que se pretenda almacenar en la base de datos de OSSIM, y de la cantidad de hosts disponibles en la red que pretendamos analizar:

2.7.4.7.2 Parámetros de configuración

Es muy importante configurar los parámetros adecuados en el servidor OSSIM para poder recolectar los eventos de forma eficiente, los parámetros personalizados más importantes son:

- Configuración de los parámetros de red.
- Configuración al panel de Administración vía Acceso Web.
- Puerto por donde escucha el servidor.

Posterior al realizar el proceso de instalación de OSSIM se podrá acceder a la interfaz Web de administración (ver Anexo 30) en la cual se pueden observar las siguientes opciones:

Dashboards (Cuadro de mandos)

Se puede observar una visión completa de todos los componentes del servidor OSSIM como nivel de gravedad de la amenaza, las vulnerabilidades de las PCs en la red, el estado de implementación, los mapas de riesgos y las estadísticas OTX (Threat exchanged program of AlienVault).

Analysis (Análisis)

El análisis es un componente muy importante en cualquier dispositivo SIEM. El servidor OSSIM analiza los host basados en sus registros. Este menú muestra las alarmas, SIEM (eventos de seguridad), entradas y registros.

Environments (Ambientes)

En este menú del servidor de OSSIM, la configuración está relacionada con los activos de la organización. Se puede observar el activo, el grupo, la configuración de red y las vulnerabilidades.

Reports (Reportes)

La notificación es un componente de cualquier servidor de registros. El servidor OSSIM también genera informes que son muy útiles para la investigación detallada de cualquier host específico.

Configuration (Configuración)

En el menú de configuración para instalar y configurar AlinetVault SIEM (OSSIM), el usuario puede cambiar las configuraciones del servidor de OSSIM, tales como cambiar la dirección IP de la interfaz de administración, adicionar más host para el monitoreo, adicionar y eliminar diferentes sensores/plugins.

2.7.4.8 Instalación de Nagios

Es uno de los sistemas más usados para el monitoreo del estatus de los servicios.

2.7.4.8.1 Principales características

- Supervisión de servicios de red (SMTP, POP3, HTTP, NTTP, ICMP, SNMP).
- Supervisión de los recursos de equipos hardware (carga del procesador, uso de los discos, log del sistema) en varios sistemas operativos.
- Supervisión remota, a través de túneles SSL cifrados o SSH.
- Chequeo de servicios paralizados.
- Posibilidad de definir la jerarquía de la red, permitiendo distinguir entre host caídos y host inaccesibles.
- Notificaciones a los contactos cuando ocurren problemas en servicios o hosts, así como cuando son resueltos (a través del correo electrónico, Jabber, SMS).
- Posibilidad de definir manejadores de eventos que ejecuten al ocurrir un evento de un servicio o host para resoluciones de problemas proactivas.
- Rotación automática del archivo de registro.
- Soporte para implementar hosts de monitores redundantes.
- Visualización del estado de la red en tiempo real a través de interfaz web, con la posibilidad de generar informes y gráficas de comportamiento de los sistemas monitorizados, y visualización del listado de notificaciones enviadas, historial de problemas, archivos de registros.

2.7.4.8.2 Configuración

Los archivos de configuración del Nagios se encuentran en el directorio /etc/nagios3, en el archivo htpasswd.users se encuentran los usuarios y sus contraseñas con acceso a la interfaz web, en el archivo nagios.cfg se especifican los directorios donde Nagios buscara el archivo de definiciones de sus comandos con cfg_file, y el directorio de los plugins y archivos de configuración de los recursos a monitorear con cfg_dir. Por defecto los plugins se encuentran en /etc/nagios-plugins/config y los archivos de configuración en /etc/nagios3/conf.d.

2.7.4.9 Sarg para el análisis de los reportes de navegación del squid

SARG es una herramienta de análisis de Log de Squid, tiene soporte para generar reportes en diferentes idiomas, mediante los reportes de uso Web.

2.7.4.9.1 Principal información que proporciona

- Top Ten de sitios más visitados
- Reportes diarios, semanales y mensuales
- Gráficas semanales y mensuales del consumo por usuario/host
- Detalles de todos los sitios a los que entró un usuario/host
- Descargas

Se puede configurar para generar reportes web de los accesos a Internet de forma periódica, además de poder ejecutarlo manualmente para generar reportes de fechas, usuarios o dominios en específico.

2.7.4.9.2 Ejemplos para la creación de reportes

Por fecha o rango de fecha

- sarg -d 14/04/2010
- sarg -d 14/04/2010-15/04/2010
- Por usuario o por dominio
- sarg -d 14/04/2010 -u jperez
- sarg -d 14/04/2010 -s vcl.desoft.cu

2.7.4.9.3 Configuración

Luego de instalado el Sarg desde la línea de comandos se debe verificar el archivo principal de configuración ubicado en /etc/squid/sarg.conf. Allí se cambia el idioma a español y se define la dirección donde se encuentran los archivos de acceso del Squid y la dirección donde se generaran las paginas HTML con los reportes, además se recomienda se cambie el título de los reportes.

language Spanish

access_log /var/log/squid3/access.log

output_dir /var/www/squid-reports

title "Reportes de Acceso Web por Usuarios"

2.7.4.10 Sqstat para la supervisión en vivo de la navegación del squid

Es un script para conocer los usuarios activos en el proxy mediante el protocolo cachemir que no es más que una interfaz del Squid que recibe toda la información de los componentes, se accede mediante http usando el protocolo especial cache_object.

2.7.4.10.1Configuración

El archivo principal de configuración está ubicado en el directorio /var/www/sqstat/config.inc.php donde se define el ip del proxy y el puerto por el cual escucha, además se debe colocar la contraseña para acceder a la cache del squid.

$squidhost[0]="192.168.0.1";

$squidport[0]=8080;

$cachemgr_passwd[0]="secreto";

También se deben realizar modificaciones en el archivo de configuración del squid en /etc/squid3/squid.conf para permitir que este sea interrogado por el SqStat mediante la adición de varias listas de control de acceso. Se coloca una contraseña a la cache mediante la directiva cachemgr_password. En el caso que el Squid esté configurado para pedir autentificación estas modificaciones deben ser adicionas antes de la primera línea que la especifique.

cachemgr_passwd secreta all

acl manager proto cache_object

acl manager-src src 127.0.0.1 192.168.0.1/255.255.255.255

http_access allow manager manager-src

2.7.4.11 MRTG para el graficado de la navegación del squid

MRTG se ejecuta como un demonio o invocado desde las tareas programadas del cron. Por defecto, cada cinco minutos recolecta la información de los dispositivos y ejecuta los scripts que se le indican en la configuración.

En un primer momento, MRTG consultaba la información, la procesaba y generaba el informe y las gráficas. En las últimas versiones, esta información es almacenada en una base de datos gestionada por RRDtool a partir de la cual, y de forma separada, se generan los informes y las gráficas.

2.7.4.11.1Configuración

Para lograr su funcionamiento se necesita activar el protocolo SNMP en el squid editando su archivo de configuración en /etc/squid3/squid.conf y adicionando las siguientes líneas:

acl snmppublic snmp_community public

snmp_port 3401

snmp_access allow snmppublic localhost

snmp_access deny all

En el fichero de configuración del MRTG en /etc/squid3/mrtg-squid.conf configuramos las rutas donde se crearán los gráficos y las paginas HTML, así como la ubicación del archivo mib en el que se especificarán los objetos a monitorizar dentro del Squid. Luego se edita el archivo /etc/crontab para ejecutar el arte cada 5 min o el tiempo que se desee.

ImageDir: /var/www/mrtg-squid

Workdir: /var/www/mrtg-squid

LoadMIBS: /etc/squid3/mib.txt

**/5 * * * * root LANG=C /usr/bin/mrtg /etc/squid3/mrtg-squid.conf --debug="cfg*

2.7.4.12 Visitors para Análisis de las estadísticas del servidor Web apache

Es una herramienta para generar reportes usando los archivos de reportes de los servidores Web.

2.7.4.12.1Principales características

- Puede procesar más de 15,000 líneas de reportes en computadoras promedio.
- Se ejecuta desde la línea de comandos y genera reportes en formato HTML y texto. Los reportes en formato de texto pueden ser usados para verificar el servidor Web mediante ssh.
- Genera las estadísticas en tiempo real.
- Analiza archivos de reportes para la mayoría de los servidores Web.
- Es portable, el código C puede ser compilado en cualquier sistema operativo

2.7.4.12.2Principal información que muestra

- Peticiones de páginas.
- Peticiones de imágenes.
- Vista de páginas por visitas.
- Páginas accedidas y la fecha de los últimos accesos de cada página accedida.
- Navegadores, sistemas operativos y dominios que accedieron.
- Distribución de los accesos por tiempo y fecha.
- Errores de páginas no encontradas.

2.7.4.12.3 Configuración

Para generar los reportes en formato HTML se debe colocar en el cron la línea siguiente para ser ejecutada cada cierto tiempo. */30 * * * * root visitors -A

/var/log/apache2/access.log -o html > /var/www/visitors/index.html

2.7.4.13 AWstats para análisis de las estadísticas de correo

Es una herramienta para el control de los reportes generados por los servidores de correo.

2.7.4.13.1 Información que ofrece

- Distribución de los correos por hora, fecha, y organizados por días, semanas y meses, así como el tamaño de los mismos.
- Cantidad de correos por usuario.
- Remitente y destinatario de cada correo.
- Dirección IP del cliente que envió el correo.
- Lista de las direcciones IP desconocidas.
- Lista de errores del servidor SMTP

2.7.4.13.2 Configuración

Como el AWstats por defecto realiza análisis para servidores web se necesitan realizar varios cambios para lograr el funcionamiento adecuado. Primeramente se copia hacia el directorio del AWstats el script maillogconvert.pl para que se encargue de darle el formato adecuado a los archivos de reportes que produce el correo, el mismo se encuentra en el directorio /usr/share/doc/awstats/examples/. Luego se debe editar el fichero de configuración comenzando por definir la dirección donde se encuentran los log del correo estos deben ser formateados antes de ser analizados:

LogFile="/usr/share/awstats/maillogconvert.pl standard < /var/log/mail.log |"

También se especifica qué tipo de log se utilizará y como estará formateado LogType=M, LogFormat="%time2 %email %email_r %host %host_r %method %url %code %bytesd". Para generar los reportes se usa en comando /usr/lib/cgi-bin/awstats.pl -config=postfix – update y para generarlos mediante html /usr/lib/cgi-bin/awstats.pl -config=postfix -output - staticlink > /var/www/awstats/index.html. Se pueden adicionar al cron para que se generen de manera automática cada cierto tiempo.

2.7.4.14 SquidAnalyzer

Es otro analizador de log de Squid hecho en Perl, con mayor funcionalidad que la anterior y con una interfaz más elegante y cuidada. Una de las características más interesantes de este analizador es que es capaz de agrupar las peticiones por segmento de red, con lo que se obtendrá un punto más granularidad a la hora de hacer estadísticas de uso.

2.7.4.14.1 Instalación

SquidAnalyzer depende para su instalación de la versión de Perl con que cuente el sistema instalado, para ello se ejecutara el siguiente comando:

perl –v

Para la instalación de este paquete, se ejecutara el siguiente comando:

apt-get install perl

Ahora se descompactará el paquete con el siguiente comando:

tar zxcf squidanalyzer-5.2.tar.gz

Seguidamente se realiza la instalación:

#/home/squid-analyzer-5.2/install_all.sh

Ahora se editara el archivo http.conf en el servidor apache:

Alias /squidreport/ /var/www/squidanalyzer

Seguidamente se reiniciará el servicio apache:

Service httpd restart

Finalmente se ejecutara en un explorador la siguiente URL:

http://localhost/squidreports

2.8 Redundancia de servidores e información

2.8.1 Raid

Con el modelo de redundancia de servidores e información se pretende conseguir: una duplicidad de servidores, arreglos de disco RAID (Redundant Array of Independet Disks) y además realizar el proceso de virtualización que permitirá perfeccionar el proceso de backup y restauración de servicios en la División. Todo este proceso tiene una desventaja, y es el elevado costo del hardware, pero la garantía de una recuperación rápida de la información, así como de las configuraciones de los servidores, base de datos y servicios en un tiempo mínimo ante cualquier desastre es una garantía invaluable para cualquier organización.

Existen 2 alternativas para la implementación de redundancia de almacenamiento de información mediante la tecnología RAID: RAID basado en software, y RAID basado en hardware. Los Arreglos RAID, basados en hardware, dotan a los equipos de computación de una mayor capacidad de almacenamiento, a la vez que proveen acceso ininterrumpido a los datos. (Ver Anexo 31)
El RAID combina múltiples discos duros en un arreglo, y almacena la información procurando evitar que se pierdan datos si uno o más discos llegan a fallar. Existen distintos niveles de redundancia en los arreglos RAID (generalmente se reconocen desde RAID-0 hasta RAID-5 aunque existen proveedores que han especificado unilateralmente otros niveles), los que definen distintas especificaciones de almacenamiento.
La mayoría de los Sistemas Operativos de Red modernos (Windows NT, Netware, Solaris, SCO Unix, etc.), tienen capacidad de manejar algunos de los niveles antes mencionados de RAID (Sistemas RAID basados en software), pero cuando se buscan altos niveles de seguridad en la redundancia de la información almacenada, se recurre a

Sistemas RAID basados en hardware. Además de ser más seguras, las soluciones RAID basadas en hardware son también más rápidas que las soluciones basadas en software.

En caso de la red datos de la División se pretende realizar la compra del hardware antes mencionado u otras de similares características, para utilizar la opción de Raid basado en hardware, realizar el proceso de virtualización y utilizar las funcionalidades del Storage, logrando con esto un alto nivel de seguridad en cuanto a redundancia de la información. Además en cada uno de los nodos deberá existir una réplica de los servicios más importante de red que se brindan a los usuarios y clientes de la división, entre ellos pueden ser citados, AD, DNS, Proxy y Correo, buscando con esto la continuidad del proceso informático, si un servicio se cae por algún motivo la réplica de este en otra máquina virtual toma su lugar de forma automática.

2.8.2 Rsync

Con el Rsync se realizaran Bakups de forma incrementar un día determinado de la semana de los archivos de configuración y de las imágenes realizadas con el Proxmox, entre las diferentes máquinas virtuales con Linux, de forma tal siempre exista una copia de seguridad con los últimos cambios.

2.8.2.1 Opciones de rsync

-a: para copiar recursivamente manteniendo privilegios, fecha de creación, permisos, etc.;

-v: para incrementar el nivel de detalle de la operación;

-z: para comprimir los datos, así la transferencia es más rápida. Si el flujo de datos por red es rápido, es posible que no sea necesario usar ésta opción, ya que el echo de comprimir los datos también provoca carga de trabajo en ambos ordenadores, y lo que se gana en velocidad de transferencia se puede perder en tiempo de compresión / descompresión de datos.

-e ssh: para usar ssh para copiar los archivos de un servidor a otro. Este es el canal cifrado que se a usar para trasferir los datos con seguridad.

2.8.2.2 Pasos para la configuración

- apt-get install rsync
- Se ejecutará rsync en el servidor de backup

rsync -e ssh -avz administrator@192.168.50.100:/var/backups /home/salvas/tmp/

Si todo hasta en orden se podrá observar un mensaje indicando los archivos recibidos, la cantidad de bytes y la velocidad de transferencia.

Ahora si se necesita hacer esto forma automática cada hora mediante cron habrá que lograr que ssh no pida contraseña. Para ello se deberá realizar el siguiente procedimiento:

1- Serán creadas una llave pública y una llave privada con:

ssh-keygen -t dsa

Presionando solo enter en todas las preguntas que el comando anterior requiere se obtendrá la llave pública en /home/administrator/.ssh/id_dsa.pub. Esto se realizara en el servidor 192.168.50.200 (Backups).

2- Copiar esta llave al servidor 192.168.1.100:

- # cd /home/usuario/.ssh/
- # cat id_dsa.pub | ssh administrator@192.168.50.100 "cat - >> /home/administrator/.ssh/authorized_keys"
- Ahora se realizará la siguiente prueba para ver que todo esté funcionando correctamente ejecutando:

  # ssh administrator@192.168.50.100

Si no pide contraseña y accede directamente, se podrá proceder con el siguiente paso.

- Ahora que el servidor 192.168.50.100 no pedirá autenticación y el proceso se puede automatizar. Se deberá ejecutar este comando en 192.168.50.200 como una tarea cada n cantidad de tiempo y desde luego se podrá tener un registro de que archivos salvados en este último:
- Finalmente se realizara la configuración de esta tarea en el cron del servidor de backup, de la siguiente forma:

crontab -e

Para ejecutar el backup cada hora se agregará en 192.168.50.200:

0 * * * * rsync -e ssh -avz administrator@192.168.50.100:/var/backups /home/salvas/tmp/

2>&1 > /var/log/rsync-backup.log

2.9 Modelo seguro

2.9.1 Cortafuego simple

Un firewall es un dispositivo que filtra el tráfico entre redes, como mínimo dos. El firewall puede ser un dispositivo físico o un software sobre un sistema operativo. En debe ser visto como una caja con DOS o más interfaces de red en la que se establecen una reglas de filtrado con las que se decide si una conexión determinada puede establecerse o no. Incluso puede ir más allá y realizar modificaciones sobre las comunicaciones, como el NAT.
Esa sería la definición genérica, hoy en día un firewall es un hardware específico con un sistema operativo o una IOS que filtra el tráfico TCP/UDP/ICMP/../IP y decide si un paquete pasa, se modifica, se convierte o se descarta. Para que un firewall entre redes funcione como tal debe tener al menos dos tarjetas de red. (Ver Anexo 32)

2.9.2 DMZ

Dependiendo de las necesidades de cada red, puede ponerse uno o más firewalls para establecer distintos perímetros de seguridad en torno a un sistema. Es frecuente también que se necesite exponer algún servidor a Internet (como es el caso de un servidor Web, un servidor de correo, etc.), y en esos casos obviamente en principio se debe aceptar cualquier conexión a ellos. Lo que se recomienda en esa situación es

situar ese servidor en lugar aparte de la red, el que denominamos DMZ o zona desmilitarizada. El cortafuego tiene entonces tres entradas (Ver Anexo 33)
En la zona desmilitarizada se pueden poner tantos servidores como se necesiten. Con esta arquitectura, se permitirá que el servidor sea accesible desde Internet de tal forma que si es atacado y se gana acceso a él, la red local sigue protegida por el cortafuego.
También en una red se pueden configurar más de una DMZ (Ver Anexo 34).

2.9.3 Diseño modelo seguro

Los servidores de la división se montaran en una DMZ Lateral Three-part, configurada así en el Router, para protegerlos del acceso directo de usuarios desde redes exteriores a la de la División. Estos últimos se conectaran a través de los routers a ETECSA a Internet y a la WAN corporativa de Desoft (Ver Anexo 18). Las PCs del administrador de red y responsable de seguridad informática tendrán acceso también a la DMZ, en el Switch-L3 se declararán las diferentes VLAN. También en los servidores se habilitarán solo los puertos que se van utilizar a partir de los servicios que están instalados con el IPTABLE, evitando así puertos innecesarios abiertos en la red.

2.10 Redundancia de rutas

Se utilizara el empleo de técnicas de enrutamiento dinámico RIP. Se definirá en el Switch-L3 las VLANs para disminuir así el dominio de colisiones. Las VLANs van a ser direcciones del tipo clase A con mascara variable e IPv4 y se definirá que se utilice IPsec, para aumentar la seguridad de los paquetes IP que viajan a través de la red LAN en la división. En caso que se migre en un futuro a IPv6 no sería necesario utilizar IPSec debido a que el IPv6 ya lo tiene implementado por defecto. En los Switch-L2 de cada local se habilitarán dos puertos para cada VLAN y otros dos para redundancia, con esto se pretende minimizar el tiempo de caída utilizando para ello rutas alternativa, en estos Switch-L2 se definen untaggeando las VLAN, se les definirán los vid de cada VLAN y se realizara el untaggeo de los puertos para las mismas.

Se utilizará solo las normas de cableado horizontal, ya que la distancia entre el Switch-L3 y los Switch-L2 es corta menor de 100 metros. Solo va a existir una conexión con fibra óptica desde el local del nodo al de Desarrollo.

Las direcciones IP para las diferentes VLAN se pueden observar en la tabla 4, estas serán distribuidas por el servidor DHCP contra MAC Address de cada PC, limitando el acceso a la red de la empresa solo de las PCs autorizadas en la misma, y se pondrán filtros MAC en la puerta de enlace y en cada firewall de cada servidor.

VLAN	Departamento	Direcciones de red	Rango de Direcciones IP	Cantidad de PC por departamento
VLAN 1	Economía	10.15.2.0/29	10.15.2.1-10.15.2.6	4
VLAN 2	Desarrollo	10.15.3.0/27	10.15.3.1-10.15.3.30	25
VLAN 3	administradores	10.15.4.0/29	10.15.4.1-10.15.4.6	3
VLAN 4	Recursos Humanos	10.15.5.0/29	10.15.5.1-10.15.5.6	4
VLAN 5	Dirección	10.15.6.0/29	10.15.6.1-10.15.6.6	5
VLAN 6	Negocios	10.15.7.0/28	10.15.7.1-10.15.7.14	12
VLAN 7	Despliegue	10.15.8.0/27	10.15.8.1-10.15.8.30	25
VLAN 8	Implementación	10.15.9.0/27	10.15.9.1-10.15.9.30	25
VLAN 9	Seguridad Informática	10.15.10.0/28	10.15.10.1-10.15.10.14	10

Tabla 2.2: rango de direcciones IP por departamento

2.11 Redundancia de medios

En los locales de la dirección e informatización se habilitaran cuatro puntos de acceso (Ver Anexo 19) en modo infraestructura, para aquellos usuarios que cuenten con Laptop, Celulares, Tables, y que estén autorizados por la dirección de la empresa para acceder a los diferentes servicios de red. Con esto se logra un doble propósito: introducir redundancia en las comunicaciones y reforzar la cobertura inalámbrica.

Se habilitará en cada uno de los Switch el uso de *spanning tree protocol* o árbol de expansión 802.1d, el cual garantiza una sola trayectoria activa entre dos estaciones de red, indicándole mayor prioridad al Switch-L3, con esto también se refuerza la redundancia, porque si falla un enlace por algún motivo queda otro que automáticamente toma el lugar del que falló y el proceso informático no se detiene bajo ningún motivo.

2.12 Protocolos de Seguridad

2.9.4 Certificados

Una de las más comunes formas de criptografía hoy es la criptografía de clave pública. La criptografía de clave pública utiliza una clave pública y una privada. Los sistemas de trabajo por encriptación de información usan clave pública. La información puede entonces solamente ser desencriptada usando la clave privada.

Un común uso de la criptografía de clave pública es la encriptación del tráfico de la aplicación usando conexión *SSL (Secure Socket Layer)* o *TLS (Transport Layer Security)*. Un ejemplo: configurando Apache al proveer HTTPS, el protocolo finaliza en SSL. Este permite una vía de encriptación del tráfico usando un protocolo que en sí mismo no provee encriptación.
Un certificado es un método usado al distribuir una clave pública y otra información alrededor de un servidor y organización quien es el responsable de él. Un certificado puede ser digitalmente asignado por una *CA (Certification Authority)*. Una CA es una tercera organización en quien se puede confiar y que confirma que la información que contiene el certificado es exacta.

2.9.4.1 Tipos de certificados

Al asegurar un servidor utilizando criptografía de llave pública, en muchos casos, puede ser enviada la demanda de certificado (incluyendo la llave pública), probado la identidad de la compañía, y pagando a la CA. La CA verifica la demanda de certificado y su identidad, y entonces envía de regreso un certificado para asegurar el servidor de la empresa. Alternativamente, puede ser creado un certificado en la propia empresa llamado por sus siglas en ingles *self-signed certificate.* Este último no aconsejable utilizarlo en entornos de producción.

2.9.4.2 Proceso para obtener un certificado de una CA

El proceso para obtener un certificado de una CA es el siguiente:

1. Crear un par llaves una pública y otra privada.
2. Crear la demanda de certificado basado en clave pública. La demanda de certificado debe contener información del servidor y del hosting de empresa en él.
3. Enviar la demanda de certificado, con los documentos que prueben la identidad, a la CA. No se puede dar un criterio exacto de cual certificado autoritario debe ser seleccionado. Esa es una decisión que compete a cada responsable basado en experiencias anteriores, o en experiencias de amigo o colegas, o simplemente en un factor monetario. Solo se recomienda leer siempre las instrucciones de la CA.
4. Cuando la CA este satisfecha al saber que usted es quien dice ser, entonces ellos envían a usted un certificado digital.
5. Instalando este certificado en un servidor seguro, y configurar entonces las aplicaciones apropiadas al usar este certificado.

2.9.5 Otras consideraciones relacionadas con la seguridad

La seguridad no es sólo un problema técnico: más que nada, es sobre buenas prácticas y permitir los riesgos. Esta sección revisa algunos de los riesgos más comunes, así como también unas pocas prácticas recomendadas que deberían, dependiendo del caso, aumentar la seguridad o reducir el impacto de un ataque exitoso.

2.9.5.1 Riesgos inherentes de las aplicaciones Web

El carácter universal de las aplicaciones Web llevó a su proliferación. Usualmente se ejecutan varias en paralelo: correo Web, wiki, sistema de gestión, foros, galería de fotos, blog, etc. La mayoría de estas aplicaciones están basadas en la pila «LAMP» (Linux, Apache, MySQL, PHP). Desafortunadamente, muchas de estas aplicaciones también fueron escritas sin considerar los problemas de seguridad. Los datos que provienen del exterior, demasiado seguido, son utilizados luego de escasa o nula validación. Se pueden proveer valores creados especiales para generar que una llamada a un programa ejecute otro en cambio. Con el paso del tiempo se corrigieron muchos de los problemas más obvios, pero aparecen nuevos problemas regularmente.

Por lo tanto, es obligatorio actualizar las aplicaciones web regularmente, para que un «cracker» (sea un atacante profesional o un «script kiddy») no pueda aprovecharse de una vulnerabilidad conocida. El riesgo real depende de cada caso, varía entre la destrucción de datos a la ejecución de código arbitrario, incluyendo la desfiguración del sitio Web.

2.9.5.2 Posibles desastres que pueden ser causados por un atacante a un sitio Web

Generalmente se utiliza una vulnerabilidad en una aplicación Web como punto de partida para intentos de intrusión Lo que sigue es una breve revisión de las consecuencias posibles.

Las consecuencias de una intrusión tendrán varios niveles de obviedad dependiendo de las motivaciones del atacante. Los «script kiddies» sólo aplican recetas que encuentran en sitios web; generalmente desfiguran una página Web o borran datos. En casos más sutiles agregan contenido invisible a las páginas Web para mejorar las referencias a sus propios sitios en los motores de búsqueda.
Un atacante más avanzado irá más allá. Un escenario desastroso podría ser como sigue: el atacante obtiene la habilidad de ejecutar programas como el usuario WWW-data, pero ejecutar una orden necesita demasiadas manipulaciones. Para hacer su tarea más sencilla, instala otra aplicación web diseñada específicamente para ejecutar remotamente muchas órdenes distintas, como navegar el sistema de archivos, examinar permisos, subir o descargar archivos, ejecutar programas o inclusive proveer una consola de red. Generalmente, la vulnerabilidad le permitirá ejecutar wget para descargar algún malware en /tmp/ y luego ejecutarlo. Usualmente se descarga dicho malware de un sitio web extranjero que fue comprometido con anterioridad y servirá para cubrir sus huellas y hacer más difícil rastrear el origen real del ataque.
En este punto el atacante tiene suficiente libertad de movimiento y, generalmente, instalan un «bot» IRC (un robot que se conecta a un servidor IRC por el que se lo puede controlar). Generalmente se lo utiliza para compartir archivos ilegales (copias no autorizadas de películas o software, etc.). Un atacante tenaz inclusive podría desear ir más allá todavía. La cuenta www-data no provee acceso completo al equipo, el atacante intentará obtener permisos de administrador. Esto no debería ser posible, pero si la aplicación web no estaba actualizada es posible también que el núcleo y otros programas tampoco estén actualizados; esto a veces deriva de una decisión del administrador que, a pesar de conocer la vulnerabilidad, descuidó la actualización del sistema ya que no existen usuarios locales. El atacante podrá aprovechar una segunda vulnerabilidad para obtener permisos de root.

Ahora el atacante es dueño de la máquina; usualmente intentarán mantener este acceso privilegiado tanto como les sea posible. Esto involucra instalar un «rootkit», un programa que reemplazará algunos componentes del sistema para que el atacante pueda obtener privilegios de administrador más adelante; el «rootkit» también intentará esconder su propia existencia así como también cualquier rastro de la intrusión. Un programa ps comprometido omitirá algunos procesos, netstat no mostrará algunas conexiones activas, etc. Utilizando los permisos de root, el atacante pudo observar el sistema completo pero no encontró datos importantes; por lo que intentará acceder a otras máquinas en la red corporativa. Analizando la cuenta del administrador y los archivos históricos, el atacante encuentra las máquinas a las que se accede frecuentemente.
Puede interceptar la contraseña de alguno de los administradores reemplazando sudo o ssh con una versión comprometida, y luego utilizar esta información en los servidores detectados… y propagar la intrusión de allí en más.
Este es un escenario de pesadilla que se puede prevenir con varias medidas. Las siguientes secciones describirán algunas de estas medidas.

2.9.5.3 Selección prudente de software

Una vez que se conocen los problemas de seguridad, debe tenerlos en cuenta en cada paso del proceso de desplegado de un servicio, especialmente al elegir el software que instalar. Muchos sitios Web, como SecurityFocus.com, mantienen una lista de vulnerabilidades descubiertas recientemente, lo cual le puede dar una idea del historial de seguridad de un software antes de desplegarlo. Por supuesto, debe balancear esta información con la popularidad de dicho software: un programa más utilizado es un objetivo más tentador y, consecuentemente, será investigado más en detalle. Por el otro lado, un programa de nicho podría estar lleno de huecos de seguridad que nunca son publicados debido a la falta de interés en una auditoria de seguridad.

En el mundo del Software Libre, generalmente hay mucha variedad de opciones y elegir un software sobre otro debería ser una decisión basada en el criterio local. Más funcionalidad implica un aumento del riesgo de una vulnerabilidad escondida en el código; elegir el programa más avanzado para una tarea podría ser contraproducente, usualmente elegir el programa más simple que cumpla los requisitos es un mejor enfoque.

2.9.5.4 Gestión de una máquina como un todo

De forma predeterminada, la mayoría de las distribuciones Linux instalan una cantidad de servicios Unix y muchas herramientas. En la mayoría de los casos, no se necesitan estos servicios y herramientas para lo que el administrador configuró la máquina. Como guía general en materia de seguridad, es mejor desinstalar software innecesario. En efecto, no tiene sentido asegurar un servidor FTP si se puede utilizar una vulnerabilidad en otro servicio no utilizado para obtener permisos de administrador en todo el equipo.
De la misma forma, generalmente se configurarán los firewalls sólo para permitir acceder a los servicios que deban estar accesibles públicamente.
Los equipos actuales son suficientemente poderosos para poder albergar varios servicios en la misma máquina física. Desde un punto de vista económico, dicha posibilidad es interesante: un sólo equipo a administrar, menor consumo de energía, etc. Desde el punto de vista de seguridad, sin embargo, esta elección puede ser un problema. Un servicio comprometido puede proveer acceso a toda la máquina, que a su vez compromete los otros servicios en el mismo equipo. Se puede mitigar este riesgo aislando los servicios. Puede lograrlo mediante virtualización (cada servicio albergado en una máquina virtual dedicada) o bien con SELinux (que cada demonio de servicio tenga un conjunto de permisos adecuado). [85]

Conclusiones del Capítulo 2

Se realizó la propuesta del diseño y modelado del sistema de seguridad, a partir de la selección de tecnología de hardware, reestructuración lógica y física de la red de datos para mejorar su seguridad, utilizando para ello el modelo jerárquico de diseño de redes y fueron seleccionadas un grupo de herramientas de software libre que permitirán las pruebas y el modelado del sistema de seguridad en tiempo real.

CAPÍTULO III SUPERVISION DEL SISTEMA DE SEGURIDAD CON HERRAMIENTAS DE SOFTWARE LIBRE

Los cambios propuestos al sistema de seguridad ya se han comenzado a realizar de forma progresiva en la división, se espera mejorar las condiciones técnicas de los servidores, y posteriormente migrar los servicios para los clúster con máquinas virtuales en los nodos con Proxmox. La mayoría de las herramientas analizadas en el capítulo anterior han sido implementadas en máquinas virtuales y se han ido probando de forma paulatina el funcionamiento del sistema de seguridad actual y los cambios que se han ido realizando en cuanto a niveles de vulnerabilidades, análisis de tráfico, supervisión de los diferentes servicios de red, nivel de encriptación de la información que viaja a través de la red de datos y el sistema de Backup implementados entre servidores actuales.

3.1 Recolección de información en la red de datos

En esta fase se debe recolectar la mayor cantidad de información posible sobre la red objetivo de análisis, en este caso en particular el análisis de tráfico y monitoreo de servicios se realizara a la red de la división Desoft Villa Clara. De allí se deben extraer informaciones tales como: posibles nombres de usuarios y sus respectivas contraseñas, en caso de que no viajen por la red por conexión segura TLS y puedan ser capturadas en texto plano, direcciones IP, MAC Address, servidores de nombre, y otras informaciones relevantes. Durante esta fase cada fragmento de información obtenida es importante y no debe ser subestimada para buscar así las posibles vulnerabilidades o debilidades del sistema de seguridad.
El proceso en el cual se realiza la captura la información puede ser dividido de dos formas diferentes. La captura de información activa y la captura de información pasiva. En la primera de estas se recolecta información enviando tráfico hacia la red que va ser el objetivo principal de nuestro análisis, como por ejemplo realizar ping ICMP, y escaneos de puertos TCP/UDP. Para el segundo caso se puede obtener información sobre la red objetivo, esto es en caso de que exista información pública sobre la misma, utilizando servicios o fuentes de terceros, como por ejemplo Google, Bing, Yahoo o redes sociales.

3.1.1 Herramientas para capturar Información de los DNS públicos

3.1.1.1 DNSenum

El propósito principal de esta aplicación es capturar toda la cantidad de información que sea posible sobre un determinado dominio, realizando una amplia diversidad de operaciones. Un ejemplo práctico del mismo es la ejecución del siguiente comando en la red LAN de la división Desoft:

 # cd /usr/share/dnsenum/
 # dnsenum --enum vcl.desoft.cu

Seguidamente son definidas algunas de las opciones más utilizadas:

"--enum" es un atajo equivalente a la opción "--thread 5 -s 15 -w". Dónde: "--threads" define el número de hilos que realizarán las diferentes consultas.

"-s" define el número máximo de subdominios a ser arrastrados desde Google. "-w" realiza consultas Whois sobre los rangos de red de la clase C.

3.1.1.2 Fierce

Fierce es un escáner relativamente ligero para realizar una enumeración que ayude a los profesionales en pruebas de penetración a localizar espacios IP y los nombres de host no continuos para dominios específicos, utilizando cosas como DNS, Whois y ARIN. A continuación se puede observar un ejemplo práctico de su utilización ejecutado en la red de datos de la división:

 # fierce -dnsserver d.ns.desoft.cu -dns vcl.desoft.cu -wordlist /usr/share/dnsenum/dns.txt -file /tmp/resultado_fierce.txt

Seguidamente son definidas las opciones más utilizadas de este comando:

"-dnsserver" define el uso de un servidor DNS en particular para las consultas del nombre del host.
"-dns" define el dominio a escanear.
"-wordlist" define una lista de palabras a utilizar para descubrir subdominios.
"-file" define un archivo de salida.
[*] La herramienta dnsenum incluye una lista de palabras "dns.txt", las cual puede ser utilizada con cualquier otra herramienta que la requiera, como fierce en este caso.

3.1.1.3 Dmitry

Dmitry es una aplicación en línea de comando para sistemas Linux, el cual permite la captura de toda la información que sea posible sobre un host, desde un simple Whois hasta los reportes del tiempo de funcionamiento o escaneo de puertos.

 # dmitry -w -e -n -s vcl.desoft.cu -o /tmp/resultado_dmitry.txt

A continuación son definidas las opciones más utilizadas:

"-w" permite realizar una consulta whois a la dirección IP de un host.
"-e" permite realizar una búsqueda de todas las posibles direcciones de correo electrónico.
"-n" intenta obtener información desde Netcraft sobre un hot.
"-s" permite realizar una búsqueda de posibles subdominios.
"-o" permite definir un nombre de archivos en el cual guardar el resultado.

3.1.2 Información de la Ruta

3.1.2.1 Traceroute

Traceroute define la ruta tomada por los paquetes desde una dirección IP de red en su camino hacia un host en otra red. Este utiliza el campo “TTL” del protocolo IP e intenta provocar una respuesta ICMP TIME_EXCEEDED desde cada pasarela a través de la ruta hacia el host. Con el siguiente comando se puede observar las rutas tomadas por los paquetes desde la PC del administrador de red hasta la puerta de enlace de Desoft Habana, en la figura 3.1 se pueden observar los resultados obtenidos, en cuanto a rutas tomadas y tiempo de retardo en que llegan los paquetes.

 # traceroute 192.168.44.2

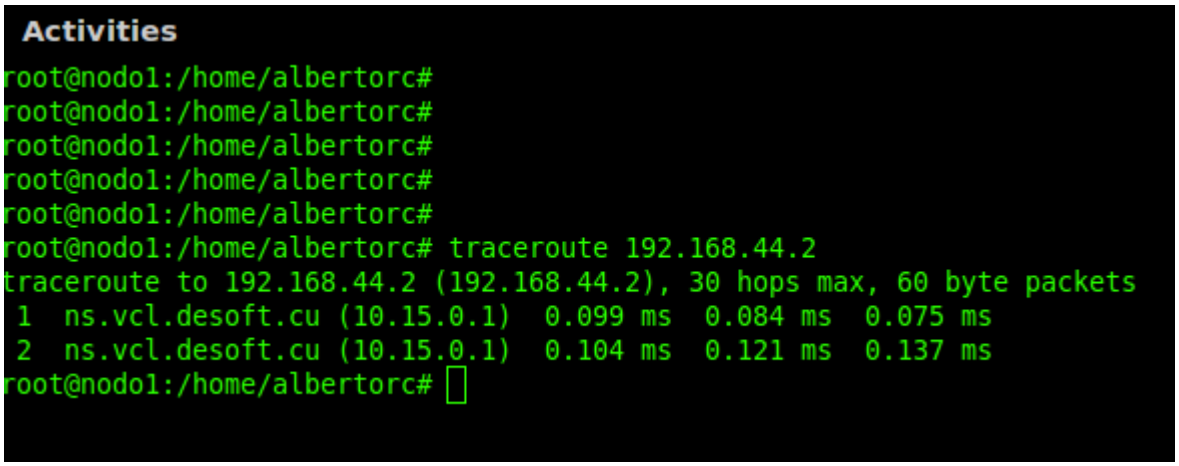


Figura 3.1 Resultados obtenidos con el comando traceroute

3.1.2.2 Tcptraceroute

Tcptraceroute utiliza paquetes TCP para trazar la ruta hacia el host objetivo. En la figura 3.2 pueden ser observados los resultados obtenidos con este comando cuando es ejecutado a través del puerto 80 (http). Se puede notar que los resultados obtenidos son muy similares que los obtenidos por traceroute.

 # tcptraceroute 192.168.44.2

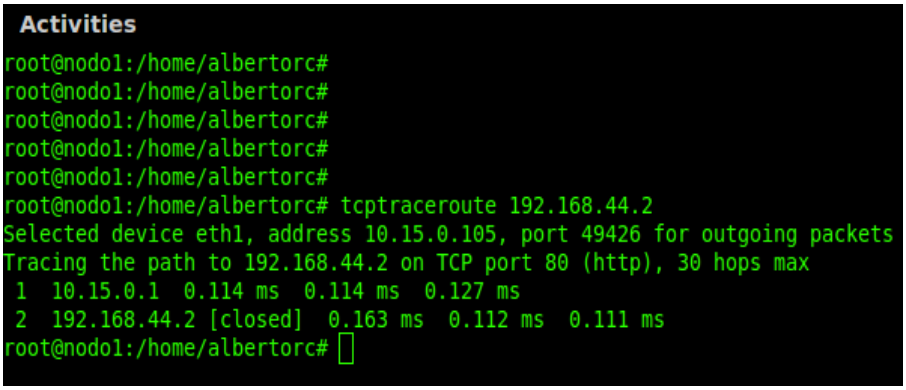


Figura 3.2 Resultado obtenidos por tcptraceroute

3.2 Detección de PCs en la red con puertos abiertos de forma arbitraria

Después de recolectar la mayor cantidad de información necesaria sobre la red de forma general, en cuanto a: dominios publicados para redes externas y puertas de enlace es necesario descubrir ahora las PCs activas en la red. Es decir encontrar cuales son las PCs que están disponibles o funcionando y cuál de estas pueden ser vulnerables. También se deben obtener indicios sobre el tipo y versión del sistema operativo utilizado en estas últimas. Toda esta información será de mucha ayuda para el proceso donde se deben detectar las vulnerabilidades.

3.2.1 Identificar las PCs de la red de la División

3.2.1.2 Nmap

Nmap “Network Mapper” o Mapeador de Puertos, es una herramienta open source para la exploración de redes y auditorías de seguridad. Con el siguiente comando se puede realizar un escaneo al rango de red 10.15.0.0/24 de la división, obteniendo como resultado las PCs activas en ese momento en el dominio. Los resultados de este escaneo pueden ser observados en la figura 3.3.

 # nmap -n -sn 10.15.0.0/24

Las siguientes opciones indican al nmap:

“-sn” a no realizar un escaneo de puertos después del descubrimiento del host, y solo imprimir los hosts disponibles que respondieron al escaneo.
“-n” a no realizar una resolución inversa al DNS sobre las direcciones IP activas que encuentre.

Nota: Cuando un usuario privilegiado intenta escanear objetivos sobre una red ethernet local, se utilizan peticiones ARP a menos que sea especificada la opción "--send-ip", la cual indica a nmap a enviar paquetes mediante sockets IP en bruto en lugar de tramas ethernet de bajo nivel.

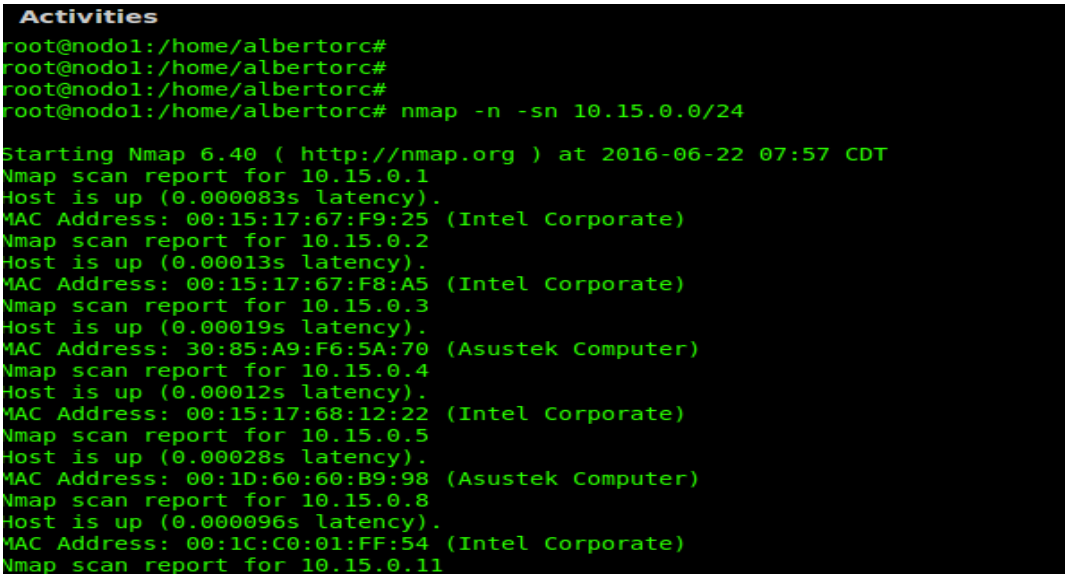


Figura 3.3 Escaneo a un rango de red Desoft con nmap

3.2.2 Reconocimiento de los sistemas operativos

Este procedimiento trata de determinar el sistema operativo funcionando en todas las PCs activas en la red, para conocer el tipo y versión del sistema operativo y definir si son versiones desactualizadas, es decir sin actualizaciones del WSUS vigentes, y por tanto vulnerables. Utilizando nuevamente la aplicación nmap con la opción –O, y analizando como ejemplo la PC de IP 10.15.0.48 se puede obtener los resultados expuestos en la figura 3.4, donde se muestra la versión de sistema operativo así como su Service Pack.

nmap -O 10.15.0.48

Las opciones siguientes son utilizadas para aportar información sobre los sistemas operativos de forma remota por nmap:

"-O" permite la detección del Sistema Operativo enviando un serie de paquetes TCP y UDP al host remoto, para luego examinar prácticamente cualquier bit en las respuestas. "-A" para habilitar la detección del Sistema Operativo junto con otras.

```
Starting Nmap 6.40 ( http://nmap.org ) at 2016-06-22 08:12 CDT

root@nodo1:/home/albertorc# nmap -o 10.15.0.48

Starting Nmap 6.40 ( http://nmap.org ) at 2016-06-22 08:13 CDT
WARNING: No targets were specified, so 0 hosts scanned.
Nmap done: 0 IP addresses (0 hosts up) scanned in 0.06 seconds
root@nodo1:/home/albertorc# nmap -O 10.15.0.48

Starting Nmap 6.40 ( http://nmap.org ) at 2016-06-22 08:13 CDT
Nmap scan report for desarrollo-12.vcl.desoft.cu (10.15.0.48)
Host is up (0.00043s latency).
Not shown: 988 closed ports
PORT      STATE    SERVICE
80/tcp    open     http
135/tcp   open     msrpc
139/tcp   open     netbios-ssn
445/tcp   open     microsoft-ds
912/tcp   open     apex-mesh
1110/tcp  filtered nfsd-status
1433/tcp  open     ms-sql-s
2383/tcp  open     ms-olap4
49152/tcp open     unknown
49153/tcp open     unknown
49154/tcp open     unknown
49155/tcp open     unknown
MAC Address: 20:25:64:C7:70:49 (Unknown)
Device type: general purpose
Running: Microsoft Windows 2008|7
OS CPE: cpe:/o:microsoft:windows_server_2008::sp2 cpe:/o:microsoft:windows_7::- cpe:/o:microsoft:windows_7::sp1 cpe:/o:microsoft:windows_8
OS details: Microsoft Windows Server 2008 SP2, Microsoft Windows 7 SP0 - SP1, Windows Server 2008 SP1, or Windows 8
Network Distance: 1 hop

OS detection performed. Please report any incorrect results at http://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 5.15 seconds
root@nodo1:/home/albertorc#
```

Figura 3.4 Información obtenida por nmap del Sistema Operativo

3.2.3 Enumerar las PCs activas en red

Enumerar las PCs que se encuentran en el entorno de red es el procedimiento utilizado para encontrar y recolectar información desde los puertos y servicios disponibles en estas. Usualmente este proceso se realiza luego de descubrir el entorno mediante el escaneo para identificar los hosts en funcionamiento.

3.2.3.1 Escaneo de Puertos

Teniendo conocimiento del rango de la red y las máquinas activas en la red, es momento de proceder con el escaneo de puertos para obtener los puertos TCP y UDP abiertos. Existen diversas técnicas para realizar el escaneo de puertos, entre las más comunes se enumeran las siguientes:

Escaneo TCP SYN
Escaneo TCP Connect
Escaneo TCP ACK
Escaneo UDP

3.2.3.1.1 Nmap

Por defecto *Nmap* utiliza un escaneo SYN, pero este es substituido por un escaneo Connect si el usuario no tiene los privilegios necesarios para enviar paquetes en bruto. Además de no especificarse los puertos, se escanean los 1,000 puertos más populares.

Nuevamente se utilizara la PC de IP 10.15.0.48 como ejemplo, ahora para obtener información sobre los puertos que esta última tiene abierto para la red de la División, los resultados obtenidos por el siguiente comando son mostrados en la figura 3.5.

nmap 10.15.0.48

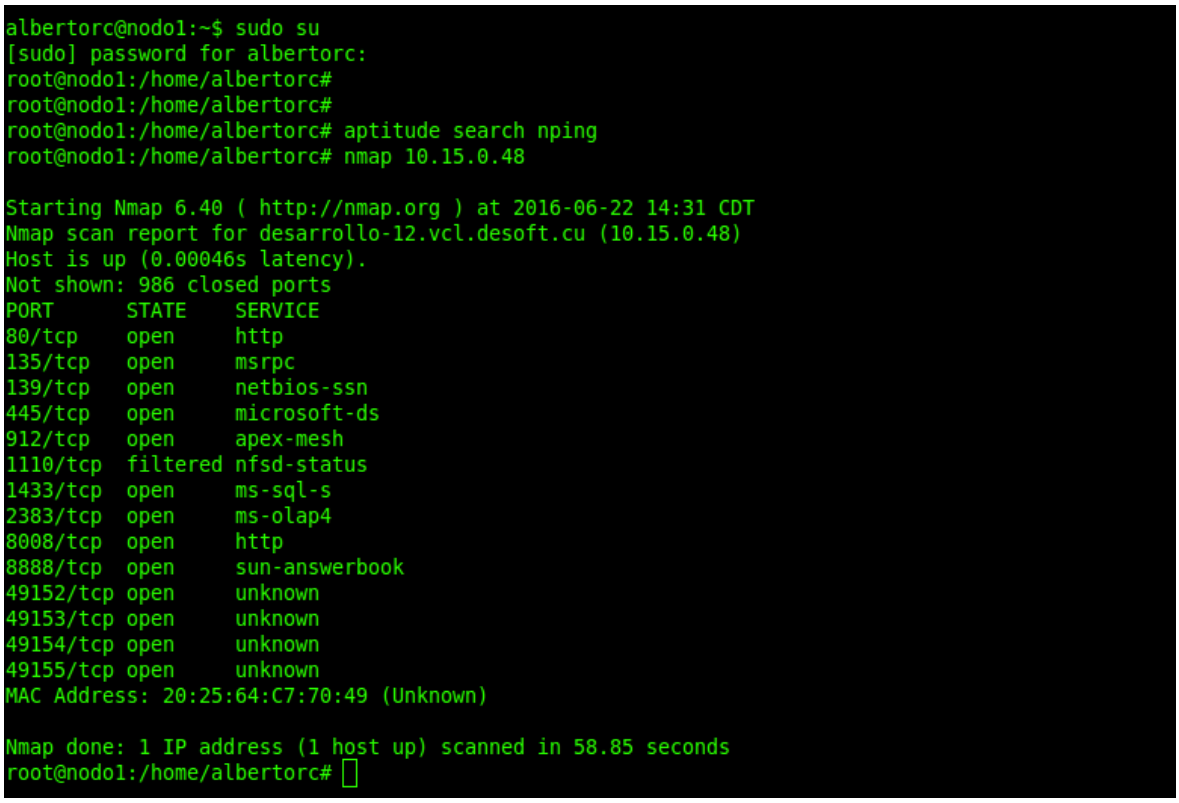


Figura 3.5 Información obtenida con una escaneo por defecto utilizando nmap

Para definir un conjunto de puertos a escanear contra un objetivo, se debe utilizar la opción “-p” en el nmap, seguido de los rango de puertos. Un ejemplo de los mismos seria:

nmap -p 80 10.15.0.0/24
nmap -p 80 10.15.0.0/24 -oA /tmp/Resultado_Scan_Nmap_p80.txt

La opción “-oA” le indica a nmap a guardar a la vez los resultados del escaneo en el formato normal, formato XML, y formato manejable con el comando “grep”. Estos serán respectivamente almacenados en archivos con las extensiones nmap, xml, gnmap.

3.2.3.1.2 Zenmap

El Zenmap es la versión grafica del nmap, esta no solamente permite un escaneo de los puertos abiertos de cada PC o servidor en la red sino que permite, a partir de un grupo de script que se le pueden adicionar, detectar vulnerabilidades en sistemas operativos y aplicaciones que no estén actualizadas y que corran sobre estos últimos, en las figuras 3.6 y 3.7 se pueden observar los resultados de un escaneo realizado en el rango IP 10.15.0.11-10.15.0.253. Allí se pueden observar puertos abiertos, últimas versiones de aplicaciones como el servidor Web Apache por ejemplo, entre otras, y sistemas operativos de cada PC con sus últimas actualizaciones.

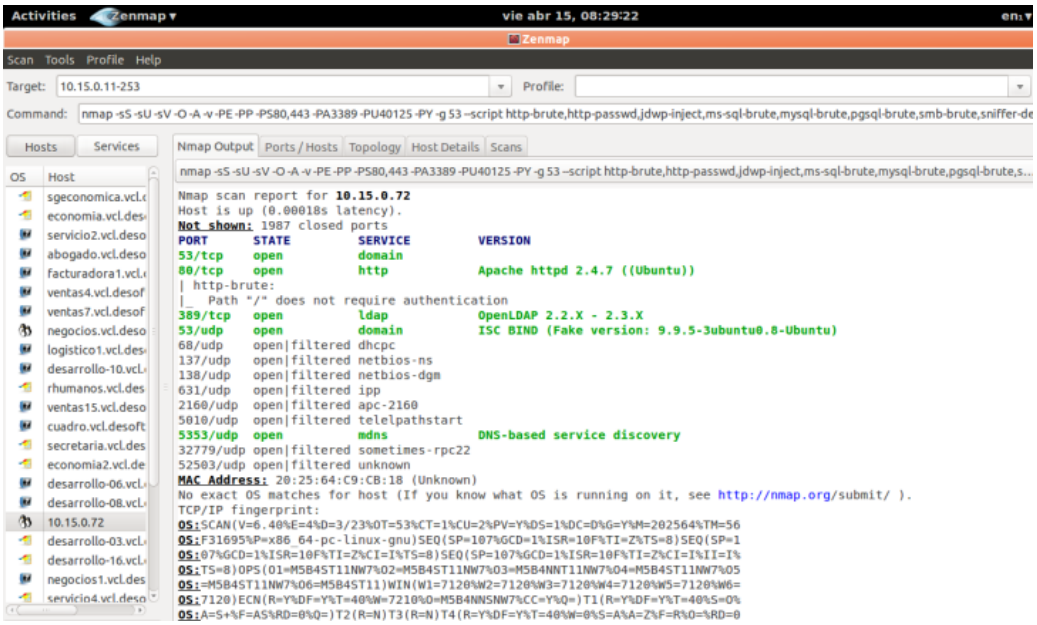


Figura 3.6 Información de salida de Zenmap para la PC 10.15.0.72

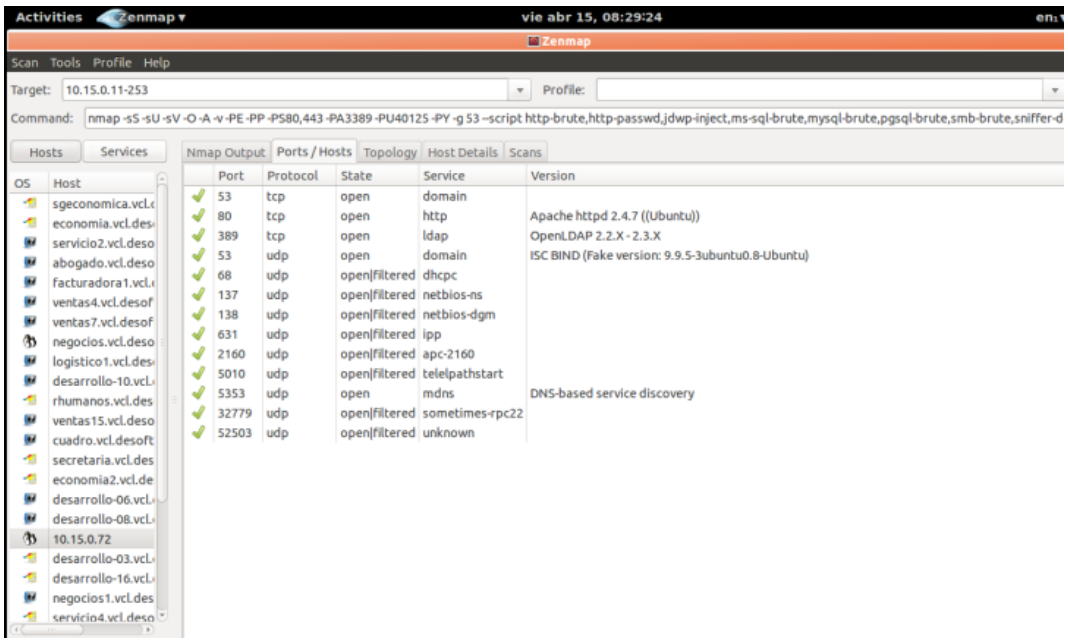


Figura 3.7 Puertos abierto en la PC 10.15.0.72

3.3 Enumeración de Servicios

Para determinar los servicios en funcionamiento en cada puerto específico es utilizado el nmap, un ejemplo simple de la utilización de este ultimo es la ejecución del siguiente comando tomando como PC de prueba la utilizada anteriormente con IP 10.15.0.48, los resultado obtenidos pueden ser observados en la figura 3.8, allí se muestran el puerto abierto y la aplicación que está corriendo en este último.

nmap -sV 10.15.0.48

“-sV” habilita la detección de versión. Después de descubrir los puertos TCP y UDP utilizando algunos de los escaneos proporcionados por nmap, la detección de versión interroga estos puertos para determinar más sobre lo que está actualmente en funcionamiento. La base de datos contiene pruebas para consultar diversos servicios y expresiones de correspondencia para reconocer e interpretar las respuestas. Nmap intenta determinar el protocolo del servicio, el nombre de la aplicación, el número de versión, nombre del host y tipo de dispositivo.

```
root@nodo1:/home/albertorc# nmap -sV 10.15.0.48

Starting Nmap 6.40 ( http://nmap.org ) at 2016-06-22 14:39 CDT
Nmap scan report for desarrollo-12.vcl.desoft.cu (10.15.0.48)
Host is up (0.00051s latency).
Not shown: 986 closed ports
PORT      STATE    SERVICE     VERSION
80/tcp    open     http        Microsoft HTTPAPI httpd 2.0 (SSDP/UPnP)
135/tcp   open     msrpc       Microsoft Windows RPC
139/tcp   open     netbios-ssn
445/tcp   open     netbios-ssn
912/tcp   open     vmware-auth VMware Authentication Daemon 1.0 (Uses VNC, SOAP)
1110/tcp  filtered nfsd-status
1433/tcp  open     ms-sql-s    Microsoft SQL Server 2008 R2 10.50.1617; RTM+ MS11-049
2383/tcp  open     ms-olap4?
8008/tcp  open     ajp13       Apache Jserv (Protocol v1.3)
8888/tcp  open     http        Apache Tomcat/Coyote JSP engine 1.1
49152/tcp open     msrpc       Microsoft Windows RPC
49153/tcp open     msrpc       Microsoft Windows RPC
49154/tcp open     msrpc       Microsoft Windows RPC
49155/tcp open     msrpc       Microsoft Windows RPC
MAC Address: 20:25:64:C7:70:49 (Unknown)
Service Info: OS: Windows; CPE: cpe:/o:microsoft:windows

Service detection performed. Please report any incorrect results at http://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 111.00 seconds
root@nodo1:/home/albertorc#
```

Figura 3.8 Información obtenida del escaneo de versiones en la PC 10.15.0.48

3.4 Detección de vulnerabilidades

La tarea de detectar vulnerabilidades en la red consiste en identificar y analizar las vulnerabilidades en los diferentes sistemas del entorno de red. Cuando ha finalizado el procedimiento de captura, descubrimiento, y enumeración de información, es el momento exacto de identificar las diferentes vulnerabilidades detectadas. La identificación de vulnerabilidades permite conocer cuáles son las más susceptibles y cuales brindan una mayor probabilidad de éxito para un atacante.

3.4.1 Vulnerabilidad Local

La vulnerabilidad local consiste en lo siguiente, se trata de cuando un atacante requiere acceso local para explotar una vulnerabilidad, ejecutando una sección determinada de código. Al aprovecharse de este tipo de vulnerabilidad un atacante puede elevar o escalar sus privilegios como usuario, para obtener acceso sin restricción en el sistema.

3.4.2 Vulnerabilidad Remota

Esta es aquella en la cual el atacante no tiene acceso previo, pero la vulnerabilidad puede ser explotada a través de la red. Este tipo de vulnerabilidad le permite a un atacante obtener acceso a un sistema sin enfrentar ningún tipo de barrera física o local.

3.4.3 Nmap Scripting Engine (NSE)

Es una de las características que hace más fuertes y flexibles a nmap. Esta permite a los usuarios a escribir y compartir scripts sencillos para automatizar una amplia variedad de tareas para redes. Estos scripts son luego ejecutados paralelamente con la velocidad y eficiencia. Los usuarios pueden confiar en el creciente y diverso conjunto de scripts distribuidos por nmap, o escribir los propios para satisfacer necesidades personales. Para realizar un escaneo utilizando todos los NSE de la categoría *“vuln”* o *vulnerabilidades* se puede utilizar:

```
# nmap -n -Pn --script vuln 10.15.0.48
```

La opción “--script” indica a nmap realizar un escaneo de scripts utilizando una lista de nombres de archivos separados por comas, categorías de scripts, o directorios. Cada elemento en la lista puede también ser una expresión boolean describiendo un conjunto de scripts más complejo.

Un listado completo e informaciones de forma detallada sobre las categorías y tipos de Scripts NSE, puede ser encontrado en la siguiente página:

http://nmap.org/nsedoc/

3.4.4 OpenVas

El OpenVas ofrece funcionalidades similares al Nessus además de una interfaz Web también es una interfaz de escritorio que permite realizar acciones. En la figura 3.9 se puede observar los resultados obtenidos luego de realizar un escaneo de la red, detectándose en uno de los servidores de prueba un hueco de seguridad en el servicio MS SQL, por el puerto 1433, protocolo tcp, en el mismo Openvas se brinda la solución a este problema.

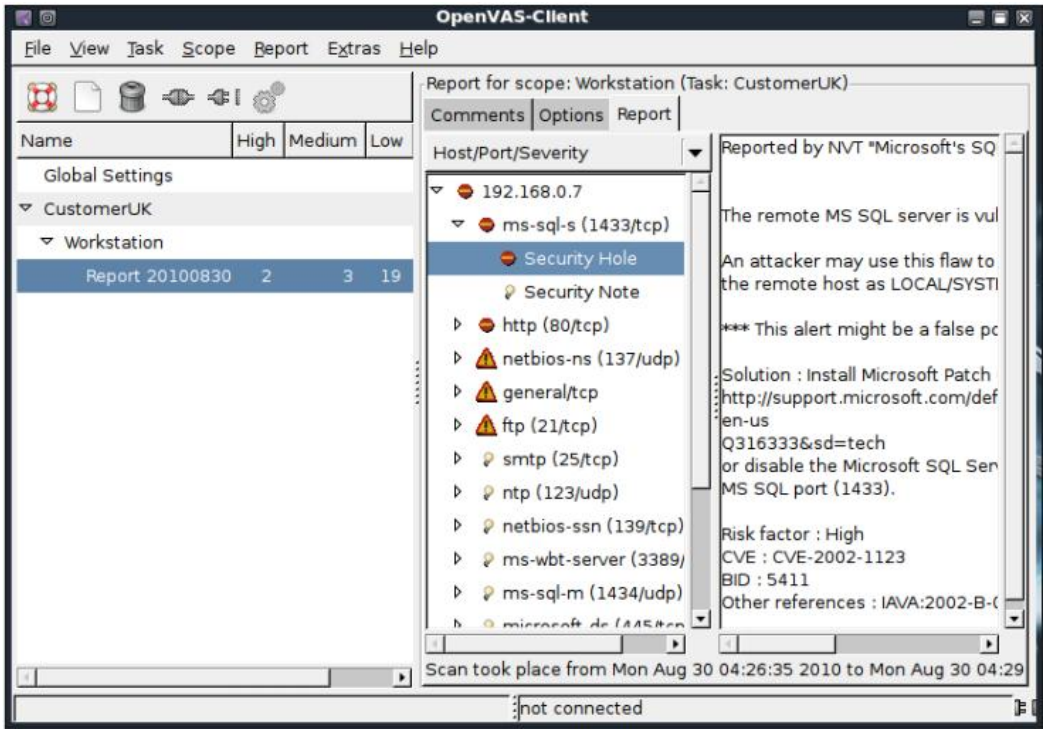


Figura 3.9 Vulnerabilidades detectadas por OpenVas

3.5 Análisis de tráfico

3.5.1. Tcpdump

En la figura 3.10 se puede observar el tráfico a través del proxy usando el comando tcpdump por el puerto 3128 a través de la interfaz de red eth2, como se puede observar no existe ocurrencia de saturación ni retardos en las peticiones a través de puerto 3128.

 # tcpdump –n –i eth2 | grep 3128

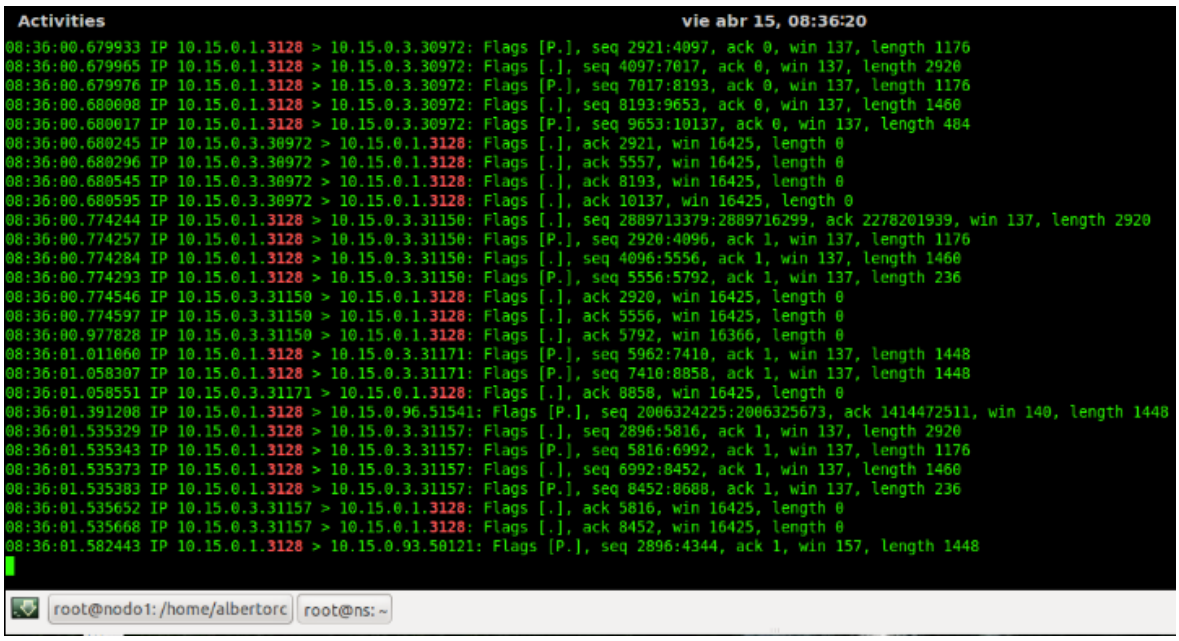


Figura 3.10 Análisis del tráfico por eth2 utilizando tcpdump

3.5.3 IPtraf

Para correr Iptraf es necesario ejecutar este último comando como root o superusuario, luego de esto se debe seleccionar la interfaz de red en la que se desee analizar el tráfico. En figura 3.11 se puede observar el análisis de tráfico realizado a la interfaz de red eth2 del servidor proxy de la división, en la interfaz de Iptraf se capturaron varios datos tales como: cantidad de paquetes, cantidad de bytes e IP de donde se accede al servidor, como se puede observar no hay existencia de saturación del trafico capturado ni retardos relevantes.

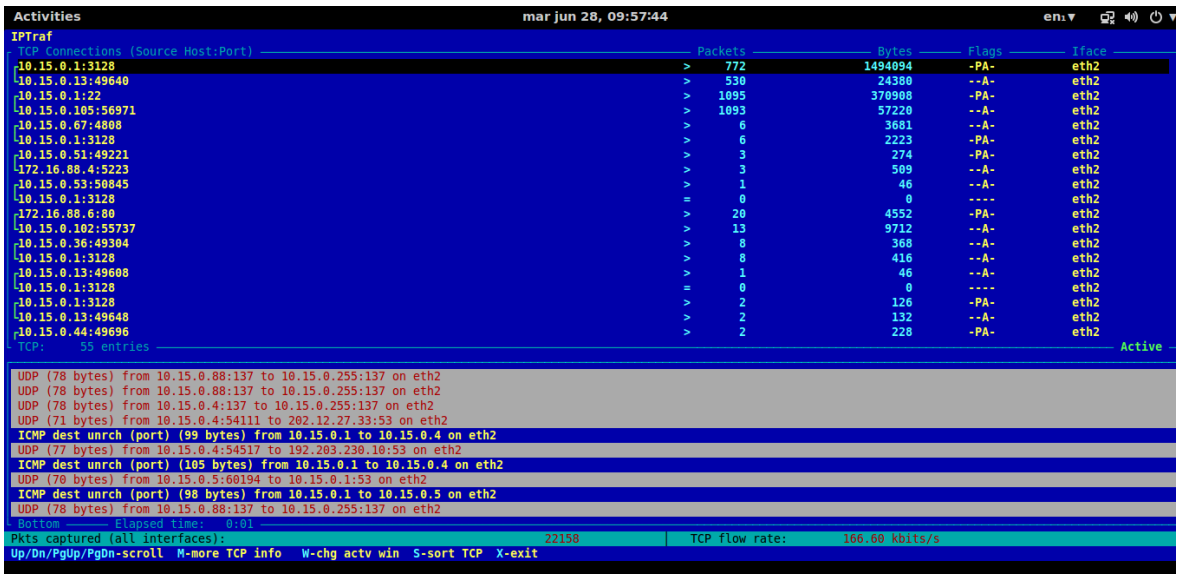


Figura 3.11 Tráfico capturado por IPtraf

3.6 Detectores de intrusos

3.6.1 Arpwatch

Este detector de intrusos puede ser configurado para enviar mensajes de aviso al correo o al teléfono móvil del supervisor de red, además puede ser ejecutado el comando *tail* con la opción –f para poder observar los cambios de MAC o IP que se estén realizando en ese momento en los ordenadores de la red de datos, como se puede observar la PC de MAC d0:67:e5:c:9:67 obtuvo la IP 10.15.0.72, y así sucede con todas las nuevas PCs que obtienen una nueva dirección IP impartida por el DHCP. Esto es de gran ayuda para supervisar la red, ya que indica cuando un atacante está utilizando un sniffer, porque lo que hace este último es poner en modo promiscuo su tarjeta de red y por lo tanto va cambiando de IP hasta obtener una con la que pueda escalar privilegios y así acceder a los servidores o PCs con información limitada o confidencial para la empresa.

 # tail –f /var/log/messages

La salida puede ofrecer:

Apr 15 12:45:17 tecmint arpwatch: new station 10.15.0.72 d0:67:e5: c: 9:67
Apr 15 12:45:19 tecmint arpwatch: new station 10.15.0.86 0:d0:b7:23:72:45
Apr 15 12:45:19 tecmint arpwatch: new station 10.15.0.86 0:d0:b7:23:72:45
Apr 15 12:45:19 tecmint arpwatch: new station 10.15.0.86 0:d0:b7:23:72:45
Apr 15 12:45:19 tecmint arpwatch: new station 10.15.0.86 0:d0:b7:23:72:45

3.6.2 Psad

En las figura 3.12 se puede observar la efectividad del Psad como detector de intrusos, dentro de /var/log/psad/ se encuentran las IP de las diferentes PCs que han sido bloqueadas por diferentes causas, por ejemplo escaneo de los puertos del servidor o exceso de peticiones por un puerto determinado. Dentro de /var/log/psad/10.15.0.3/ se almacenan un grupo de ficheros de las diferentes IP que accedieron al servidor de correo 10.15.0.3 y han sido bloqueadas por el Psad, ver figura 3.13. En la figura 3.14 se pueden observar los detalles de acceso de la IP 192.168.2.20 y las razones porque fue bloqueada esta última al acceder al servidor de correo Zimbra (IP 10.15.0.3).

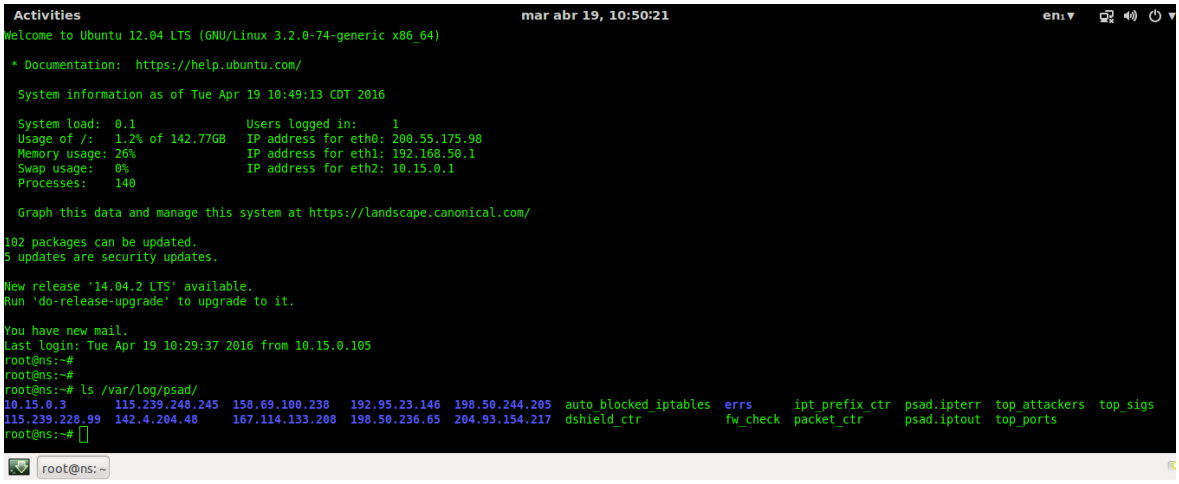


Figura 3.12: IP bloqueadas por el Psad

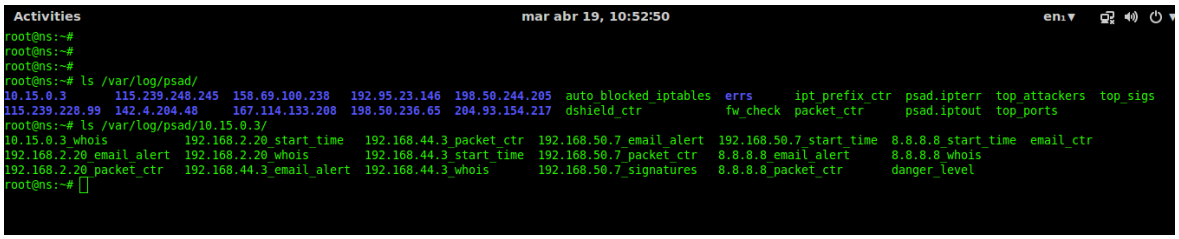


Figura 3.13: Acceso a la IP 10.15.0.3

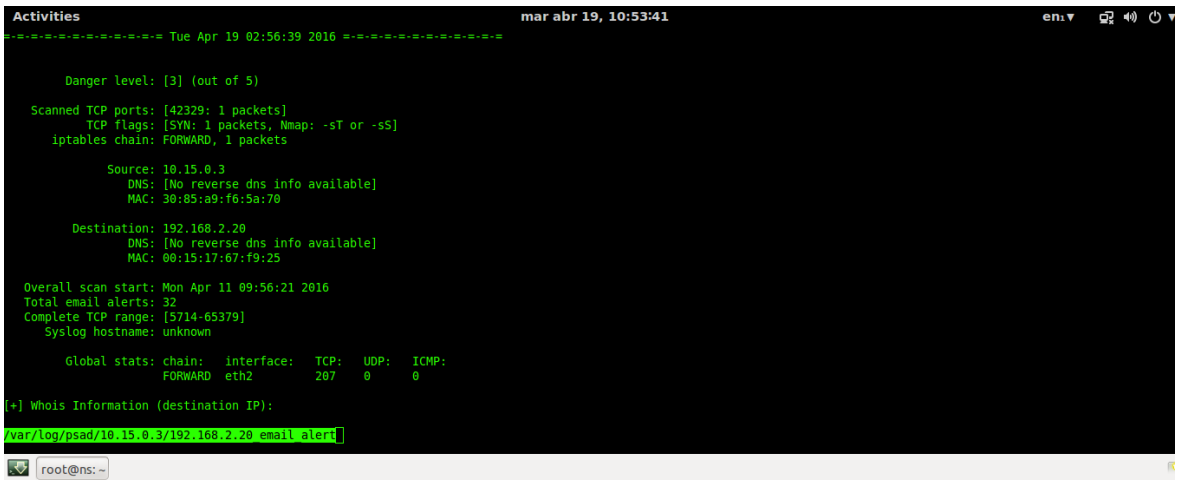


Figura 3.14: Detalles de accesos de la IP 192.168.2.20

3.7 Monitoreo de la disponibilidad de los servicios

3.7.1 Nagios

Nagios muestra información sobre el estado de los servicios, servidores en la red, así como también se puede monitorear un servidor determinado en cuanto a: procesos ejecutándose en este último, espacio del disco duro, servicio ssh, además puede ser configurado para enviar un sms a la dirección de correo electrónico o a nuestro número de teléfono móvil al supervisor de la red, esto es en caso de que algún servicio de mayor prioridad para la empresa no esté disponible, en la figura 3.15 se puede observar el servicio Nagios corriendo sobre de un servidor de prueba de la División.

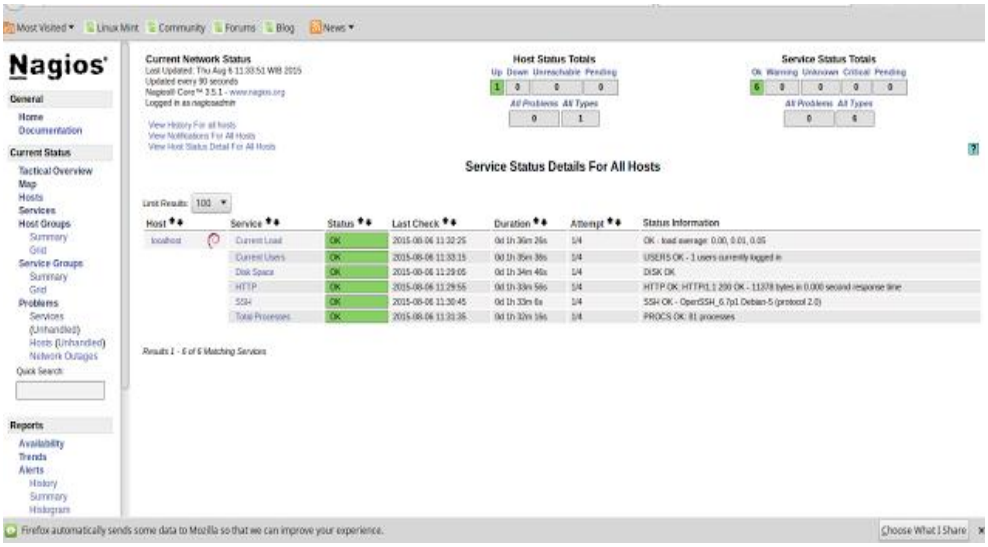


Figura 3.15: Ventana del estado de los servicios de Nagios

3.8 Monitoreo de la navegación Web y correo electrónico

3.8.1 Sarg

Sarg es una herramienta que brinda información sobre los accesos a Internet detallando fecha, hora, usuario, direcciones, así como archivos descargados. En la figura 3.16 se puede observar los accesos realizados por fechas y por usuarios esto aporta un control al supervisor de la red sobre los accesos de cada usuario a través del proxy de la División, dejando trazabilidad de cada una de sus operaciones.

Figura 3.19 Accesos por fechas y por usuarios

3.8.2 Squid-Analyzer

El Squid-Analyzer brinda una estadística de los sitios más accedidos de forma detallada. A partir del análisis de un grupo de Logs de acceso al proxy se obtuvo como resultado las estadísticas que puede ser observadas en la figura 3.20 de los sitios más visitados por los usuarios de la división.

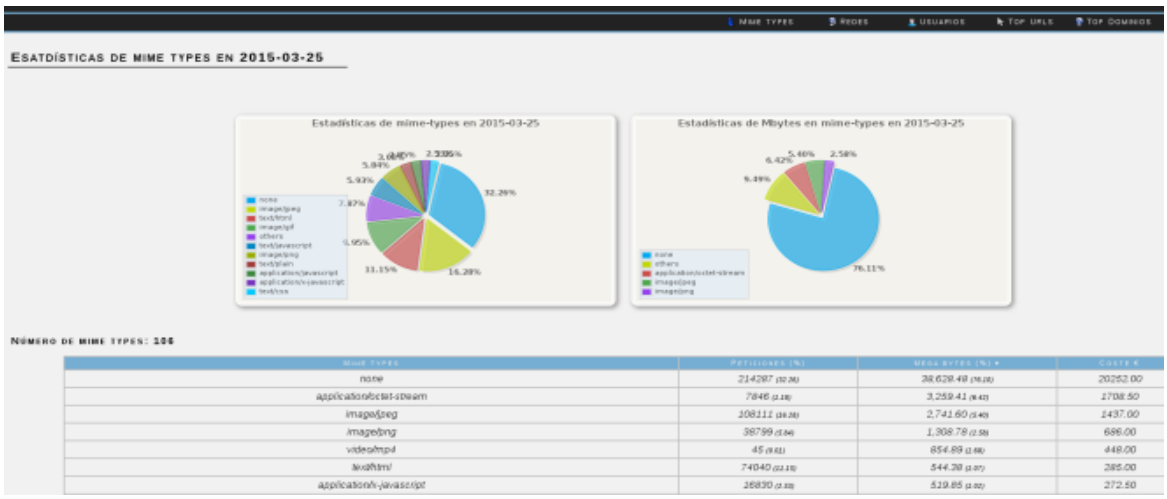


Figura 3.20 Estadísticas de acceso a sitios

3.8.3 SqStat

Con este Script se pueden observar los usuarios que se encuentran navegando en tiempo real, mediante una consulta a la cache del Squid. En la figura 3.21 se puede observar las descargas que están realizando los usuarios *alberto.rodriguez* e *hiram.perez* en ese preciso momento.

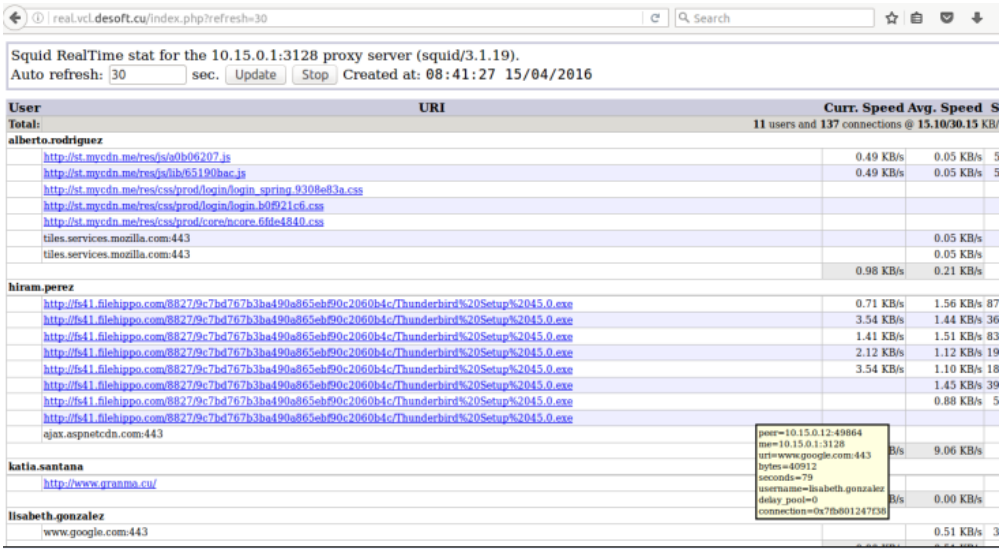


Figura 3.21 Ventana de Sqstat

3.8.4 MRTG

El MRTG realiza los gráficos del tráfico que pasa por el proxy. Genera gráficos diarios, semanales, mensuales y anuales, en la división se cuenta con dos routers, uno para la WAN que facilita el acceso a la VPN Desoft y otro para el acceso a Internet a través del proxy. En la figura 3.22 y 3.23 se puede observar el tráfico semanal y mensual de los

canales de 256 kbps y 512 kbps respectivamente, se puede ver la saturación del canal de 256 kbps debido a que existe muy poco ancho de banda, y el tráfico saliente en el canal de 512 kbps hacia la VPN de Desoft.

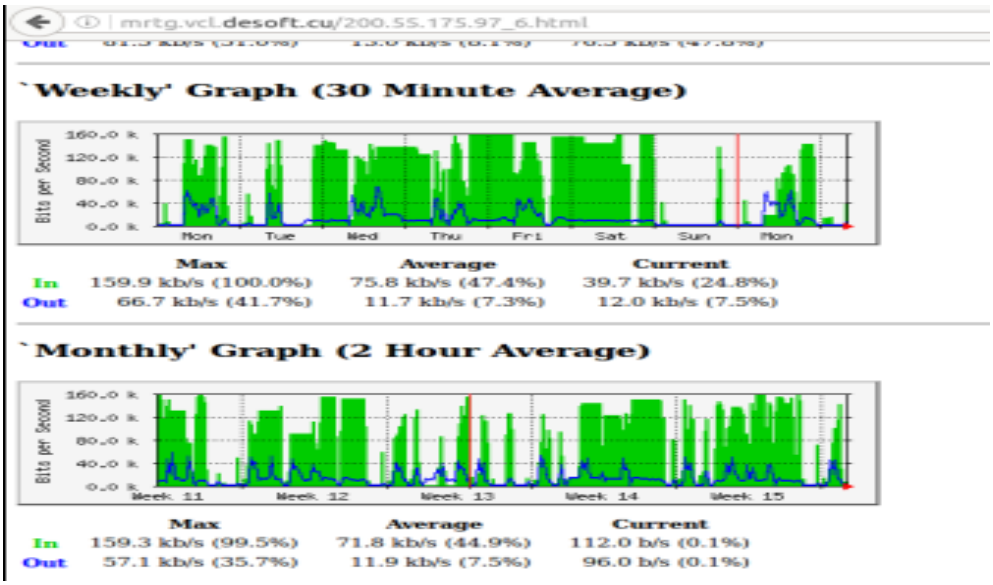


Figura 3.22 Tráfico semanal y mensual en canal 256 kbps acceso a Internet

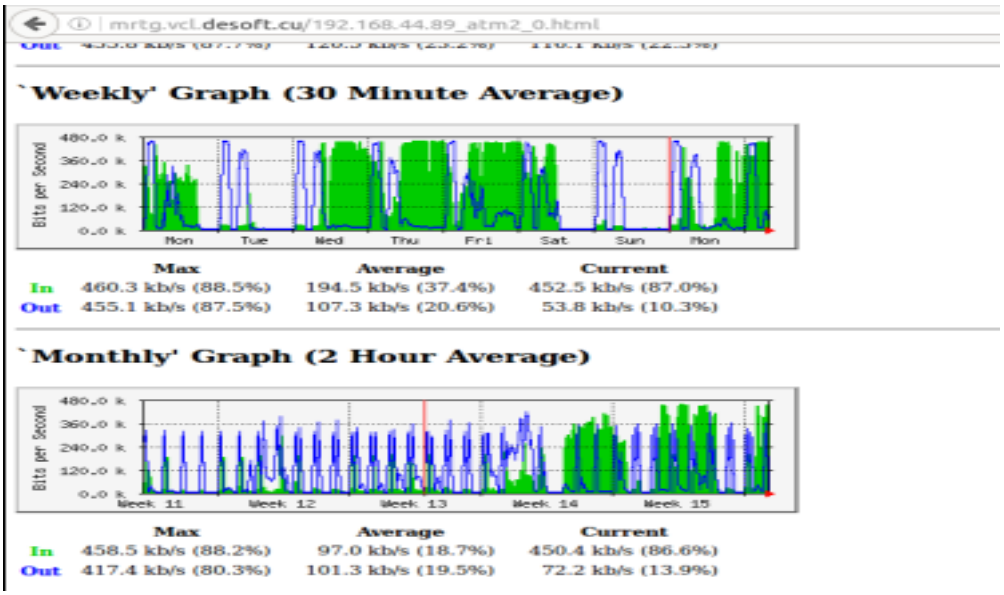


Figura 3.23 Tráfico semanal y mensual en canal 512 kbps WAN Desoft

3.8.5 Visitors

El Visitors ofrece detalles sobre las visitas al servidor Web, es una herramienta muy útil para tener una estadística de los accesos de los usuarios, así como de los exploradores que estos últimos utilizan. En la figura 3.24 se puede observar los exploradores más utilizados para acceder al servidor Web, así como una estadística semanal de los accesos a este último.

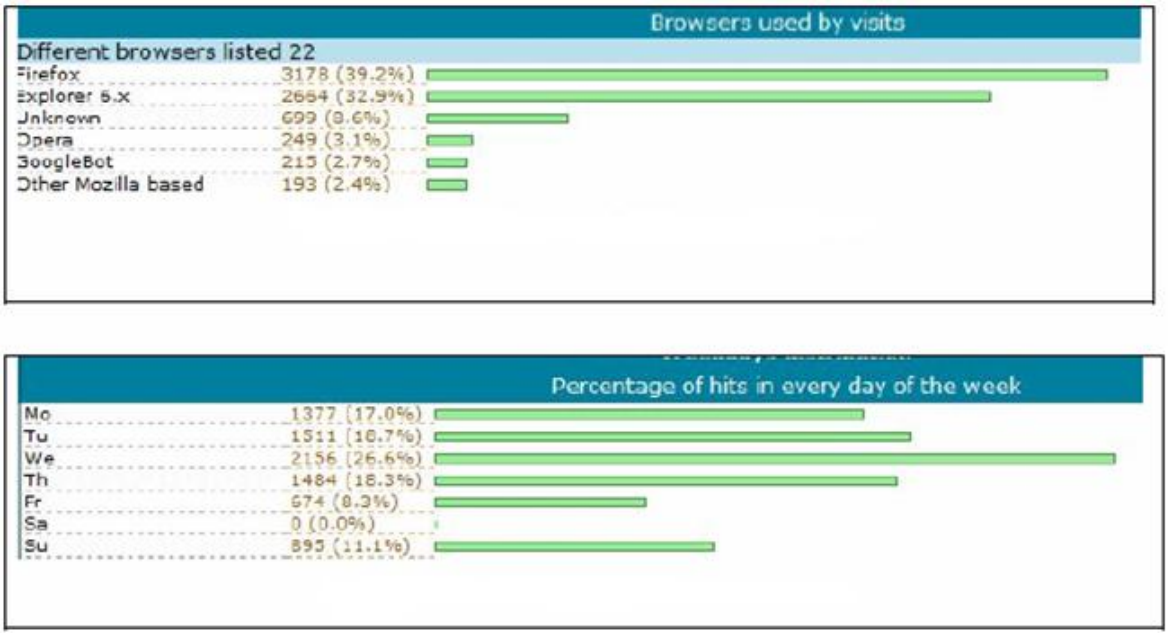


Figura 3.24 Navegador con los que se accedieron y distribución semanal de los accesos

3.8.6 Servidor de correo Zimbra y su interfaz Web de administración

La interfaz de administración del servidor de correos Zimbra trae implícita sus propias herramientas que permiten realizar diferentes operaciones, tales como: manejar las colas de correo, crear usuarios, grupos y también brinda estadísticas. En la figura 3.25 se puede observar las cuentas de usuarios, los alias y las listas de distribución, además muestra el estado de cada una de las cuentas, si están activas o bloqueadas.

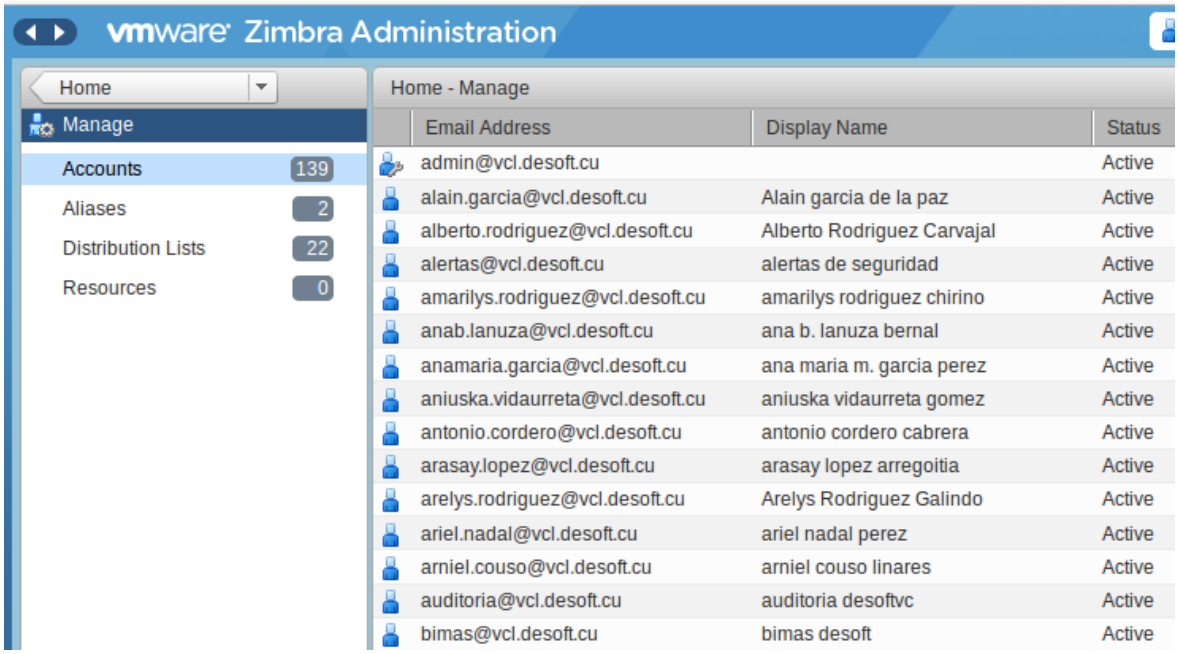


Figura 3.25 Administración de cuentas en el servidor de correos Zimbra

Conclusiones del Capítulo 3

A partir de los resultados obtenidos con las herramientas de software libre se puede constatar la seguridad de los paquetes que viajan a través de la red de datos, la eficacia de los IDS seleccionados cuando se realiza el escaneo intenso a un puerto de uno de los servidores, la detección de sniffer activos cuando existe un cambio de MAC e IP, la detección de vulnerabilidades en las diferentes PC, así como la utilidad de las herramientas de supervisión para la gestión y análisis de la seguridad del sistema informático. Se pudo observar el amplio abanico de posibilidades que brindan para una administración segura la interfaz Web del servidor de correo Zimbra, esta última ya trae incluidas sus propias herramientas para el manejo de las colas de correo, los Log y las estadísticas del servidor.

CONCLUSIONES

- Se definieron los requisitos necesarios que debe cumplir un sistema de seguridad para el ámbito empresarial, a partir de un estudio y análisis de los conceptos fundamentales. En particular se definió como aspectos necesarios la existencia de: herramientas para el análisis de la conectividad y su calidad, herramientas para el análisis de protocolos y tráfico en la red, herramientas para analizar la resolución de nombres, herramientas para el monitoreo de la red, herramientas para el análisis de logs de diferentes aplicaciones, herramientas para la detección de vulnerabilidades que complementan las que ya existen en la empresa.
- Se seleccionaron las tecnologías de hardware necesarias para dar comienzo al proceso de virtualización en la empresa seleccionando Proxmox por cumplir con los requisitos necesarios y ser software libre. Adicionalmente se hace una propuesta del hardware necesario para mejorar la conectividad y los servidores de la red.
- A partir de pruebas en tiempo real fueron seleccionadas un grupo de herramientas de software libre que permitirán mejorar la gestión, el funcionamiento y seguridad de la red de área local, en cuanto a: detección de vulnerabilidades, análisis de tráfico, supervisión de servicios de red, detectores de intrusos y pruebas.
- Se realizó la propuesta de diseño de sistema de seguridad para la División Desoft Villa Clara a partir de un análisis realizado de la topología de red, los servicios de más utilizados en la empresa, la tecnología de hardware existente, el sistema de backup implementado y el análisis de tráfico en la red de datos.
- Finalmente con máquinas virtuales fueron implementadas estas herramientas sobre software libre que permitieron realizar análisis, pruebas inicialmente sobre recreaciones sencillas de entornos virtuales y luego en lugares estratégicos de la red como el servidor Proxy, el servidor de correos y los demás servidores incluyendo los de producción.

RECOMENDACIONES

Como trabajo inmediato se recomienda:

- La adquisición de una tecnología de hardware con características iguales o similares a la sugerida para dar inicio al proceso de virtualización.
- La reestructuración de la red para una mayor seguridad desde el punto de estratégico para la DMZ y los Firewall de las subredes.
- Adquirir un Switch capa tres para disminuir el dominio de colisiones, segmentando la red mediante VLANs.
- Implementar los puntos de acceso con un servidor Radius para la cobertura inalámbrica y la redundancia en las comunicaciones.
- Habilitar en cada uno de los Switch que lo permitan el uso de *Spanning tree protocol*, protocolo utilizado para buscar una ruta alternativa en caso que una de las conexiones de red sea afectada, indicándole mayor prioridad al Switch-L3.
- Instalar en máquinas virtuales las herramientas de software libre utilizadas en auditorias de seguridad informática, análisis de tráfico y detección de intrusos para lograr con esto una gestión más favorable de la seguridad en la empresa.

BIBLIOGRAFÍA

[1] S. BosWorth, M. E. Kabay and E. Whyne, "*Computer Security Handbook"*, in Brief History and Mission of Information System Security, Sixth Edition, English ed. Canada: Wiley, 2014.

[2] Douglas E. Comer, "*Internetworking with TCP/IP Vol I: Principles, Protocols, and Architecture*", in chapter1 Introduction and Overview, Sixth Edition, English ed United States, 2014, pp 1-17.

[3] UIT, sala de prensa, "*La penetración de usuarios Internet se multiplico por siete desde el año 2000*" 2015, disponible en: https://www.itu.int/net/pressoffice/press_releases/2015/17-es.aspx.

[4] Ch. P. Pfleeger, Sh. Lawrence Pfleeger and J. Margulies,, "*Security in computing*", in Chapter1 The Vulnerability–Threat–Control Paradigm, fifth edition, English ed United States, 2015, pp 1-21.

[5] W. Stallings, "*Network Security Essentials: Applications and standards*" fourth editions, in Chapter1 Introduction, fourth edition, English ed United States, 2011, pp 1-26.

[6] Y. R. Wong, "*Monitoreo de servicios en redes LAN*", trabajo fin de carrera, Dep. Computación, Fac. Matemática, Física y Computación, Universidad Central ¨Marta Abreu¨ de las Villas, Villa Clara, 2012.

[7] Gary A. Donahue, "*Network Warrior*", Second Edition. English ed. United States: O'Reilly, 2011.

[8] Kenneth Geers, "*Strategic Cyber Security",* English ed. United States: Publication Filtri tee 12, 10132 Tallinn, Estonia, 2011.

[9] Eric Kramer, "*101 Successful Networking Strategies*", English ed. United States: Stacy L. Hiquet, 2012.

[10] Greg Tomsho, "*Guide to Networking Essentials*", Sixth Edition. English ed. United States: Boston, 2011.

[11] Scott Mueller, "*Upgrading and Repairing PCs*," 20th Edition, English ed. United States: Mueller, Scott. 2012.

[12] Todd Lammle and John Swartz "*CCNA Data Center Introducing Cisco Data Center Networking Study Guide*", English ed. Canada: John Wiley & Sons, Inc, 2013.

[13] Jason T. Luttgens and Matthew Pepe, "Incident Response & Computer Forensics", Third Edition, English ed. United States: Toronto, 2014.

[14] William Stallings, "Network Security Essentials: Applications and Standards", Fourth Edition, English ed. United States: Toronto, 2011, pp 3-5.

[15] Taylor and Francis Group, LLC, "Cyber Security Essentials", English ed. United States: New York, 2011, p.1.

[16] María Lazarte "Una caja de herramientas de seguridad protege a las organizaciones de los ciberataques" (2015-12-17). Available:

http://www.iso.org/iso/home/news_index/news_archive/news.htm?refid=Ref2032 . [Último acceso: 20 Diciembre 2015].

[17] A. López Neira, J. Ruiz Spohr "El portal de ISO 27001 en Español", Disponible: http://www.iso27000.es/sgsi.html/ Fecha de consulta: Abril 2016

[18] ITU-T "Security in Telecommunications and Information Technology", sixth edition , English ed., septiembre 2015, p 12.

[19] Y. R. Wong, "*Monitoreo de servicios en redes LAN*", trabajo fin de carrera, Dep. Computación, Fac. Matemática, Física y Computación, Universidad Central ¨Marta Abreu¨ de las Villas, Villa Clara, 2012, pp. 8-10.

[20] William Stallings "*Cryptography and Network Security Principles and Practice*", English ed. United States: Toronto, sixth edition, 2014, pp. 17-18"

[21] William Stallings "*Cryptography and Network Security Principles and Practice*", English ed. Canadá: Toronto, sixth edition, 2014, pp. 20-21

[22] A. E. Caballero Quezada "*Hacking con kali Linux*"Versión 2.5, Edición Esp., Junio del 2015, p. 2-3. Disponible: http://www.reydes.com/d/?q=node/2

[23] Robert W. Beggs" Mastering Kali Linux for Advanced Penetration Testing", English ed. United States: Packt Publishing Ltd, 2014, p. 16.

[24] Official Kali Linux Documentation. (Mar 13, 2013) Introdución a Kali Linux. [website]. Available: www.docs.kali.org/

[25] Michael Rash**, "**Cipherdyne Security Software", Disponible: http://cipherdyne.org/psad/, Fecha de consulta: Abril 2016.

[26] Christos Pontikis "Install and Config Psad in Debian 7 Wheezy", Disponible: http://www.pontikis.net/blog/psad-install-config-debian-wheezy/ , Fecha de consulta: Abril 2016.

[27] Sitio Oficial de Ubuntu"The Port Scan Attack Detector", Disponible: http://manpages.ubuntu.com/manpages/saucy/man8/psad.8.html/, Fecha de consulta: Abril 2016.

[28] Ravi Saive April 16, 2013"Arpwatch Tool to Monitor Ethernet Activity in Linux", Disponible:http://www.tecmint.com/monitor-ethernet-activity-in-linux/, Fecha de consulta: Abril 2016.

[29] Sin autor, on April 2012 "Monitor your network for new hosts using arpwatch", Disponible: http://www.networkinghowtos.com/howto/monitor-your-network-for-new-hosts-using-arpwatch/, Fecha de consulta: Abril 2016.

[30] W. Baluja García, C. C. Caro Reina, F. A. Cancio Bello "*OSSIM, una alternativa para la integración de la gestión de seguridad en la red*", Revista Telem@tica. Vol. 11. No. 1, enero-abril, 2012. ISSN 1729-3804, p. 13-14

[31] R. Miller, David. Harris Shon, a. harper, Allen. Vandyke, Stephen. blask, chris: "*Security Information and Event Management (SIEM) Implementation*". ed. mcgraw hill. New York, usa. 2011. páginas 54-91.
[32] 7. Sin autor: "Monitoreo de Red", Articulo digitalizado disponible en: http://www.qualydat.com/en/exito/46-casos-de-exito/114-monitoreored.html
[Último acceso: 15 Enero 2016]

[33] R. Miller, David. Harris shon, a. Harper, Allen. vandyke, stephen. blask, chris: "*Security Information and Event Management (SIEM) Implementation*". Ed. Mcgraw hill. New York, USA. 2011. Páginas 140-142.
[34] «Alien Vault,» [En línea]. Available: https://www.alienvault.com/open-threat-exchange/projects . [Último acceso: 26 septiembre 2014].
[35] A. Ossim, «AlienVault OSSIM,» 15 Septiembre 2014. [En línea]. Available: https://www.alienvault.com/open-threat- exchange/projects/
.[Último acceso: 15 Septiembre 2014].

[36] W. BLOG, «Wolfant's BLOG,» 15 Octubre 2014. [En línea]. Available: http://wolfant.insuasti.ec/?p=29 . [Último acceso: 15 Octubre 2014].
[37] L. Martinez, «SecurityByDefault.com,» 03 Mayo 2013. [En línea]. Available: http://www.securitybydefault.com/2013/05/mi-analisis-de-alienvaultossim-421.html . [Último acceso: Septiembre 2014].
[38] A. A. Parriza, «angelalonzo.ec,» [En línea]. Available: http://www.angelalonso.es/doc-presentaciones/ossim-hakin9.pdf .
[39] N. C. L. M. jose alvarez orozco, «Blogdiario,» 12 08 2012. [En línea]. Available: http://networkadmin.blogspot.es/ . [Último acceso: 18 08 2013].
[40] K. Makino, «kinomakino.blogspot,» 18 03 2014. [En línea]. Available: http://kinomakino.blogspot.com/2014/03/ossim-pentesting-continuo-como-si.html . [Último acceso: 17 12 2014].
[41] Admin, «todoit.com.ve,» 16 05 2011. [En línea]. Available: http://todoit.com.ve/blog/2011/sobre-metodologia-de-gestion-de-redes/ .
[42] C. E. B., «coberturadigital,» 16 05 2014. [En línea]. Available: http://www.coberturadigital.com/2014/05/16/internet-en-ecuador-el-acceso-paso-del-3-al-404-en-10-anos/#comments .
[43] «Bajolared,» 16 05 2014. [En línea]. Available: http://www.bajolared.com/wordpress/ossim-como-plataforma-de-monitorizacion-y-gestion-de-informacion-de-seguridad/ . [Último acceso: 28 10 2014].
[44] AlienVault "Using USM and OSSIM 5.1 with OTX", rev. 2 September 8, 2015

[45] Alienvault OSSIM, https://www.alienvault.com/open-threat-exchange/projects.[Último acceso: marzo 2016]

[46] Alienvault OSSIM, https://www.alienvault.com/documentation/, [Último acceso: marzo 2016]

[47] AlienVault "*Unified Security Management Solution How to display Security Events from an external AlienVault Database*", English ed., edition 01, 2014, pp. 4-8

[48] AlienVault "Unified Security Management Solution Life Cycle of a log", English ed., 2014, pp. 4-13

[49] AlienVault "Unified Security Management Solution Managing remote components after upgrading to 4.8", English ed., 2014, pp. 4-5

[50] Blueliv "AlienVault Plugin Documentation", English ed., April 16, 2015, pp. 2-18

[51] William Stallings "Cyber Security Essentials: Application and Standards",English ed. United States: New York, Fourth edition, 2011, pp.29-30.

[52] Gary A. Donahue, "*Network Warrior*", Second Edition. English ed. United States: O'Reilly, 2011.pp. 459-467.

[53] Thomas M. Eastep,"Three-Interface Firewall" [En línea]. Available: http://shorewall.net/three-interface.htm/ , [Último acceso: Abril 2016]

[54] Hidalgo, Elías. Proxmox VE, una gran herramienta de virtualización. *Proxmox VE, una gran herramienta de virtualización.* [En línea] 30 de Enero de 2012. http://linuxzone.es/2012/01/30/proxmox-ve-una-gran-herramienta-de-virtualizacion/, [Último acceso: 02 de Julio de 2012.]

[55] S.Verma, R.Choubey, R.soni, "*An Efficient Developed New Symmetric Key Cryptography Algorithm for Information Security*", International Journal of Emerging Technology and Advanced Engineering Web-site: www.ijetae.com (ISSN 2250-2459, Volume 2, Issue 7, July 2012).

[56] S.Verma, R.Choubey, R.soni, "*An Efficient Developed New Symmetric Key Cryptography Algorithm for Information Security*", International Journal of Emerging Technology and Advanced Engineering

Web-site: www.ijetae.com (ISSN 2250-2459, Volume 2, Issue 7, July 2012).

[57] K.Loukhaoukha, J.Chouinard, A. Berdai, "*A Secure Image Encryption Algorithm Based on the Rubik's Cube Principle*", Journal of Electrical and Computer Engineering 01/2012; DOI:10.1155/2012/173931.

[58] S.Kilaru, Y.Kanukuntla, A.Firdouse, M.Bushra & S.chava, "*Effective and Key Sensitive Security Algorithm For An Image Processing Using Robust Rubik Encryption & Decryption Process*", ISSN (Print): 2278-8948, Volume-2, Issue-5, 2013.

[59] Dr. C. Vitalio Alfonso Reguera, conferencia"*Seguridad en redes y sistemas*" Universidad Central de Las Villas. Facultad de Ingeniería Eléctrica Departamento de Telecomunicaciones y Electrónica, curso 2012

[60] Douglas E. Comer "*Internetworking With TCP/IP Vol I: Principles, Protocols, and Architecture*", English ed.United States: New York, Sixth Edition, 2014, pp. 606-620

[61] Tim Polk, Kerry McKay, Santosh Chokhani, NIST Special Publication 800-52 Revision 1 "*Guidelines for the Selection, Configuration, and Use of Transport Layer Security (TLS) Implementations*", April 2014, English ed., Available: http://dx.doi.org/10.6028/NIST.SP.800-52r1/, [Último acceso: Abril 2016]

[62] "Server virtualization security best practices." Disponible: http://searchservervirtualization.techtarget.com/ tutorial/Server-virtualization-security-best-practicesguide, [Último acceso: Abril 2016]

[63] Wasim Ahmed "*Mastering Proxmox"*, July 2014, English ed. Birmingham – Mumbai, p.6"

[64] Wasim Ahmed "*Mastering Proxmox*", July 2014, English ed. Birmingham – Mumbai, p.26"

[65] Wasim Ahmed "*Mastering Proxmox*", July 2014, English ed. Birmingham – Mumbai, p.27"

[66] Dr. Héctor Cruz Enriquez, conferencia "*Diseño de redes LAN para ambientes Intranet*" Universidad Central de Las Villas. Facultad de Ingeniería Eléctrica Departamento de Telecomunicaciones y Electrónica, curso 2012

[67] Polze, A. and Tröger, P. (2012), "*Trends and challenges in operating systems—from parallel computing to cloud computing*". Concurrency Computat: Pract. Exper. 24: 676–686. doi: 10.1002/cpe.1903

[68] Sabnis, S., Verbruggen, M., Hickey, J. and McBride, A. J. (2012), "*Intrinsically Secure Next-Generation Networks*". Bell Labs Tech. J., 17: 17–36. doi: 10.1002/bltj.21556

[69] Shavlik Technologies. ", Disponible: http://totemguard.com/soporte/files/Shavlik_NetChk_Configure.pdf/

[70] "Symantec Backup Exec 12.5 for Windows Servers". Disponible: http://ftpandina.atv.com.pe/ManualesIT/ManualLTo.pdf

[71] "Seguridad y cumplimiento normativo". Disponible: http://www.vmware.com/latam/cloud-securitycompliance/cloud-security#sthash.wrmYQ5XX.dpuf

[72] "Check Point". Disponible: http://www.etekreycom.com.ar/tecno/proveedor/check_point.htm/

[73] "*Microsoft Virtualization*". Disponible: http://www.microsoft.com/spain/virtualizacion/solutions/technology/default.mspx/, [Último acceso: Abril 2016]

[74] CMAN Project, Disponible: https://www.sourceware.org/cluster/cman/, Fecha de consulta: Mayo 2015.

[75] "*Pacemaker A scalable High Availability cluster resource manager"*, Disponible: http://clusterlabs.org/ , Fecha de consulta: Mayo 2015.

[76] "*Corosync, The corosync cluster engine"*, Disponible: http://corosync.github.io/corosync/ , Fecha de consulta: Mayo 2015.

[77] "*Instalar el rol Hyper-V y configurar una máquina virtual*", Disponible: https://technet.microsoft.com/eses/library/hh846766.aspx, Fecha de consulta: Mayo 2015

[78] Wasim Ahmed "*Mastering Proxmox"*, July 2014, English ed. Birmingham – Mumbai, pp 7-8.

[79] A. E. Caballero Quezada "*Hacking con kali Linux*"Versión 2.5, Edición Esp., Junio del 2015, p. 4. Disponible: http://www.reydes.com/d/?q=node/2

[80] Robert W. Beggs" *Mastering Kali Linux for Advanced Penetration Testing*", English ed. United States: Packt Publishing Ltd, 2014, p. 23.

[81] H.Wang ; H.Zheng ; B.Hu ; H.Tang ."*Improved Lightweight Encryption Algorithm Based on Optimized S-Box", Computational and Information Sciences (ICCIS)"*, 2013 Fifth International Conference.

[82] T.Sharma. ; R. Thilagavathy, "*Performance analysis of advanced encryption standard for low power and area applications, Information & Communication Technologies (ICT)*", 2013 IEEE Conference, 2013.

[83] K.Loukhaoukha, J.Chouinard, A. Berdai, "*A Secure Image Encryption Algorithm Based on the Rubik's Cube Principle*", Journal of Electrical and Computer Engineering 01/2012; DOI:10.1155/2012/173931

[84] S.Kilaru, Y.Kanukuntla, A.Firdouse, M.Bushra & S.chava, "Effective and Key Sensitive Security Algorithm For An Image Processing Using Robust Rubik Encryption & Decryption Process", ISSN (Print): 2278-8948, Volume-2, Issue-5, 2013.

[85] Raphäel Hertzog and Roland Mas "*El Libro del administrador de Debian*", edit. Español, May 2013, p. 386]

[86] Wireshark. (n.d.). Chapter 1. Introduction. Retrieved from

https://www.wireshark.org/docs/wsug_html_chunked/ChapterIntroduction.html#ChIntroPurposes, Fecha de consulta: Enero 2016

[87] Snort. (2015, August 28). SNORT User's Manual. Retrieved from https://snort.org/#documents, Fecha de consulta: Enero 2016

[88] I.J. Intelligent Systems and Applications. (2014). Strategic sensor placement for intrusion detection in network-based IDS. Retrieved from http://www.mecs-press.org/ijisa/ijisa-v6-n2/IJISA-V6-N2-8.pdf, Fecha de consulta: Enero 2016

[89]Fuye Han, Zhen Chen, Hongfeng Xu and Yong Liang, A Collaborative Botnets Suppression System Based on Overlay Network, International Journal of Security and Networks, Vo. 7, No. 4, 2012.

[90] Zhen Chen, FuYe Han, Junwei Cao, Xin Jiang, Shuo Chen, "*Cloud Computing-Based Forensic Analysis for Collaborative Network Security Management System, Tsinghua Science and Technology*", 18 (1), pp.40-50, 2013.

[91] Xinming Chen, Kailin Ge, Zhen Chen and Jun Li, AC-Suffix-Tree: Buffer Free String Matching on Out-of-Sequence Packets, Proc. of the 7th ACM/IEEE Symposium on Architectures for Networking and Communications Systems (ANCS), 2011.

[92] Tianyang Li, Fuye Han, Shuai Ding, Zhen Chen, LARX: "*Large-scale Anti-phishing by Retrospective Data-Exploring Based on a Cloud Computing Platform"*, ICCCN GridPeer workshop, 2011.

[93] Beipeng Mu, Xinming Chen, Zhen Chen, "*A Collaborative Network Security Management System in Metropolitan Area Network"*, Proc. of the 3rd International Conference on Communications and Mobile Computing (CMC), 2011.

[94] Xinming Chen, Beipeng Mu, Zhen Chen, NetSecu: "*A Collaborative Network Security Platform for in-network Security"*, Proc. of the 3rd International Conference on Communications and Mobile Computing (CMC), 2011.

[95] H.Wang ; H.Zheng ; B.Hu ; H.Tang ."*Improved Lightweight Encryption Algorithm Based on Optimized S-Box*", Computational and Information Sciences (ICCIS), 2013 Fifth International Conference.

[96] T.Sharma. ; R. Thilagavathy,"*Performance analysis of advanced encryption standard for low power and area applications*", Information & Communication Technologies (ICT), 2013 IEEE Conference, 2013.

[97] Wireshark. (n.d.). Chapter 1. Introduction. Retrieved from

https://www.wireshark.org/docs/wsug_html_chunked/ChapterIntroduction.html#ChIntroPurposes/, Fecha de consulta: Enero 2016

[98] Snort. (2015, August 28). SNORT Users Manual. Retrieved from https://snort.org/#documents/, Fecha de consulta: Enero 2016

[99] I.J. Intelligent Systems and Applications. (2014)."*Strategic sensor placement for intrusion detection in network-based IDS*". Retrieved from http://www.mecspress.org/ijisa/ijisa-v6-n2/IJISA-V6-N2-8.pdf

[100] Zhen Chen, FuYe Han, Junwei Cao, Xin Jiang, Shuo Chen, "*Cloud Computing-Based Forensic Analysis for Collaborative Network Security Management System*", Tsinghua Science and Technology, 18 (1), pp.40-50, 2013.

[101] Beipeng Mu, Xinming Chen, Zhen Chen, "*A Collaborative Network Security Management System in Metropolitan Area Network*", Proc. of the 3rd International Conference on Communications and Mobile Computing (CMC), 2011.

[102] Xinming Chen, Beipeng Mu, Zhen Chen, NetSecu: "*A Collaborative Network Security Platform for in-network Security*", Proc. of the 3rd International Conference on Communications and Mobile Computing (CMC), 2011.

[103] Wang, Xiang, Zhi Liu, Yaxuan Qi, and Jun Li. "LiveCloud: A lucid orchestrator for cloud datacenters." *In Cloud Computing Technology and Science (CloudCom*)", 2012 IEEE 4th International Conference on, pp. 341-348. IEEE, 2012.

[104] VMWare Network security, http://www.vmware.com/products/nsx/resources.html/ , Fecha de consulta: Enero 2016

[105] Y. D. Lin, R. H. Hwang, F. Baker, "*Computer Networks: An Open Source Approach*," McGraw-Hill, February 2011.

[106] Schultz, Michael J., and Patrick Crowley. "*Performance Analysis of Packet Capture Methods in a 10 Gbps Virtualized Environment*". In Computer Communications and Networks (ICCCN), 2012 21st International Conference on, pp. 1-8. IEEE, 2012.

[107] Cardigliano, Alfredo, Luca Deri, Joseph Gasparakis, and Francesco Fusco. "*vPF_RING: towards wire-speed network monitoring using virtual machines*". In Proceedings of the 2011 ACM SIGCOMM conference on Internet measurement conference, pp. 533-548. ACM, 2011.

[108] Velázquez, Eugenio. Tecnología Pyme. [En línea] 8 de enero de 2009. [Citado el: 30 de Agosto de 2012.] http://www.tecnologiapyme.com/software/que-es-la-virtualizacion./, Fecha de consulta: Enero 2016

[109]Revista Summa. [En línea] 2 de abril de 2012. [Citado el: 15 de octubre de 2012.] http://www.revistasumma.com/tecnologia/24352-panduit-virtualizar-la-infraestructura-y-la-demanda- de-servicios.html./, Fecha de consulta: Enero 2016.

[109] Todd Lammle and John Swartz, "*CCNA Data Center: Introducing Cisco Data Center Networking Study Guide*", English ed. Canada, 2013.

[110] Joseph Muniz and Aamir Lakhan "*Web Penetration Testing with Kali Linux*", English ed. United States: Washington DC, July 2013.

[111] Shantanu Tushar and Sarath Lakshman "*Linux Shell Scripting Cookbook*", English ed. Birmingham – Mumbai, Second Edition, May 2013.

[112] Cao Li-ying, Zhang Xiao-xian, Liu He, and Chen Gui-fen, "*A Network Intrusion Detection Method Based on Combined Model*", International Conference on Mechatronic Science,Electric Engineering and Computer, pp.254-257, 2011.

[113] Volker Fusenig and Ayush Sharma. "*Security architecture for cloud networking*". Presented in Proceedings of the 2012 International Conference on Networking and Computing, ICNC 2012, IEEE Computer Society, 2012.

[114] Hoffman, Joshua. TechNet Magazine. [En línea] mayo de 2011. [Citado el: 15 de octubre de 2012.] http://technet.microsoft.com/es-es/magazine/hh126814.aspx/

[115] M. Souppaya and K. Scarfone "*Guidelines for Securiting Wireless Local Area Networks (WLAN)*", English ed. United States, February 2012.

GLOSARIO DE TERMINOS

AES: *Advanced Encryption Standard*
CA: *Certification Authority*
CMP: *Certificate Management Protocol*
CMS: *Cryptographic Message Syntax*
CPS: *Certificate Practice Statement*
CRL: *Certificate Revocation List*
DES: *Data Encryption Standard*
DPD: *Delegated Path Discovery*
DPV: *Delegated Path Validation*
DSA: *Digital Signature Algorithm*
FQDN: *Full Qualified Domain Name*
HTTP: *HyperText Transfer Protocol*
IDEA: *International Data Encryption Algorithm*
IETF: *Internet Engineering Task Force*
IMAP4: *Internet Message Access Protocol*
IOS: *Internetwork Operating System*
LDAP: *Lightweight Directory Access Protocol*
MAC: *Message Authentication Code*
OSI: *Open Systems Interconnection*
POP3: *Post Office Protocol*
PS: *Personal Storage Enviroments*
RA: *Registration Authority*
RSA: *Rivest Shamir Adleman*
SAML: *Security Assertion Markup Language*
SCEP: *Simple Certificate Enrollment Protocol*
SCVP: *Server-Based Certificate Validation Protocol*
SHA: *Secure Hash Algorithm*
S/MIME: *Secure/Multipurpose Internet Mail Extensions*
SMTP: *Simple Mail Transfer Protocol*
SPKI: *Simple Public Key Infrastructure*
SSL: *Secure Socket Layer*
TCP: *Transmission Control Protocol*
TLS: *Transport Layer Security*

URL: *Uniform Resource Locator*

VPN: *Virtual Private Network*

W3C: *World Wide Web Consortium*

WTLS*: Wireless Transport Layer Security Protocol*

XML: *eXtensible Markup Language*

SIEM: *Security information and event management*

OSSIM: *Open Source SIEM*

SQL: *Structured Query Language*

IPS: *Intrusion Prevention System*

IDS: *Intrusion Detection System*

VLAN: *Virtual Local Area Network*

DNS: *Domain Name Server*

VPN: *Virtual Private Network*

SAN (Storage Area Network): *Sistema de Almacenamiento en Red.*
SCSI: *es un estándar que permite el uso del protocolo SCSI sobre redes TCP/IP.*

DMZ: *Zona Desmilitarizada es una red local que se ubica entre la red interna de una organización y una red externa, generalmente Internet.*

LAN: *Red de área local: Interconexión de varias computadoras y periféricos para que se comuniquen entre sí generalmente no más de 100 metros.*

NAT: *Es un mecanismo utilizado por los routers para intercambiar paquetes entre dos redes que se asignan mutuamente direcciones incompatibles convirtiendo en tiempo real las direcciones utilizadas en los paquetes transportados.*

Proxy: *Aplicación que intercepta las conexiones de red entre los clientes y destinos, comúnmente utilizado para la interconexión de múltiples computadoras a un único vínculo de Internet aprovechando una cache de contenidos que es compartido entre todos los equipos.*

Router o enrutador: *es un dispositivo para interconexión de redes de computadoras que opera en la capa tres (capa de red).*

Log: *Archivos donde se almacenan los mensajes generados por los programas o el sistema operativo.*

Sniffer: *un analizador de paquetes es un programa de captura de las tramas de una red de computadoras.*

Plugins: *es una aplicación que se relaciona con otra para aportarle una función nueva y generalmente muy específica.*

ACL: *lista de control de acceso o ACL (del inglés, access control list) es un concepto de seguridad informática usado para fomentar la separación de privilegios.*

Website: *colección de páginas web relacionadas y comunes a un dominio de Internet o subdominio en la World Wide Web en Internet.*

Exploit: *Es una pieza de software, un fragmento de datos, o una secuencia de comandos con el fin de automatizar el aprovechamiento de un error, fallo o vulnerabilidad.*

Auditoria de seguridad: *Una auditoria de seguridad es el proceso de leer y analizar a fondo el código fuente de algún software, buscando potenciales vulnerabilidades de seguridad que pueda contener. Usualmente, dichas auditorias son proactivas y se las realizan para asegurar que un programa cumple ciertos requisitos de seguridad.*

Inyección SQL: *Cuando un programa agrega datos a una consulta SQL de forma insegura, es vulnerable a inyecciones SQL; este nombre hace referencia al acto de cambiar un parámetro de forma que la consulta ejecutada por el programa resultará diferente a la esperada, bien para dañar la base de datos o para acceder a datos a los que normalmente no tendría acceso.*

Filtrado de consultas http: *Apache 2 incluye módulos que permiten filtrar consultas HTTP entrantes. Esto permite bloquear algunos vectores de ataque. Por ejemplo, limitar la longitud de los parámetros puede prevenir un desbordamiento de búfer. De forma más general, puede validar los parámetros inclusive antes de que sean pasados a la aplicación web y puede restringir el acceso según muchos criterios. Inclusive puede combinarlo con actualizaciones dinámicas del firewall, para prohibirle temporalmente el acceso al servidor web a un cliente que infrinja alguna de las reglas. Configurar estas verificaciones puede ser una tarea larga y tediosa, pero valdrá la pena cuando la aplicación web que deba desplegar tenga un historial de seguridad dudoso. mod-security (en el paquete libapache-mod-security) es el módulo principal de este tipo.*

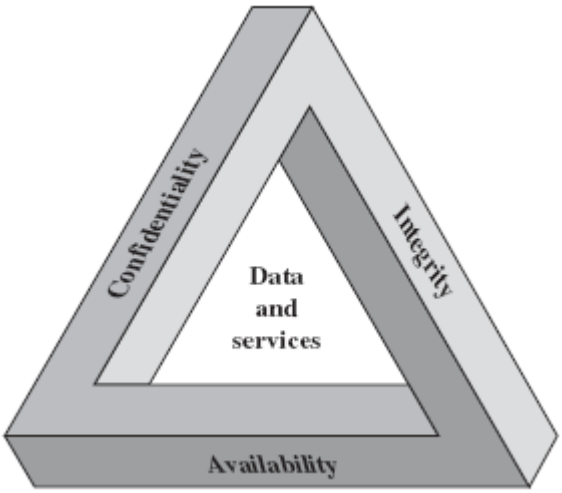
Escalada de privilegios: *Este término cubre cualquier cosa que pueda ser utilizada para obtener más permisos de los que normalmente tendría un usuario normal. El programa*

Sudo: *está diseñado específicamente para proveer permisos de administración a algunos usuarios. Pero también se utiliza el mismo término para describir el acto en el que un atacante aprovecha una vulnerabilidad para obtener permisos indebidos*

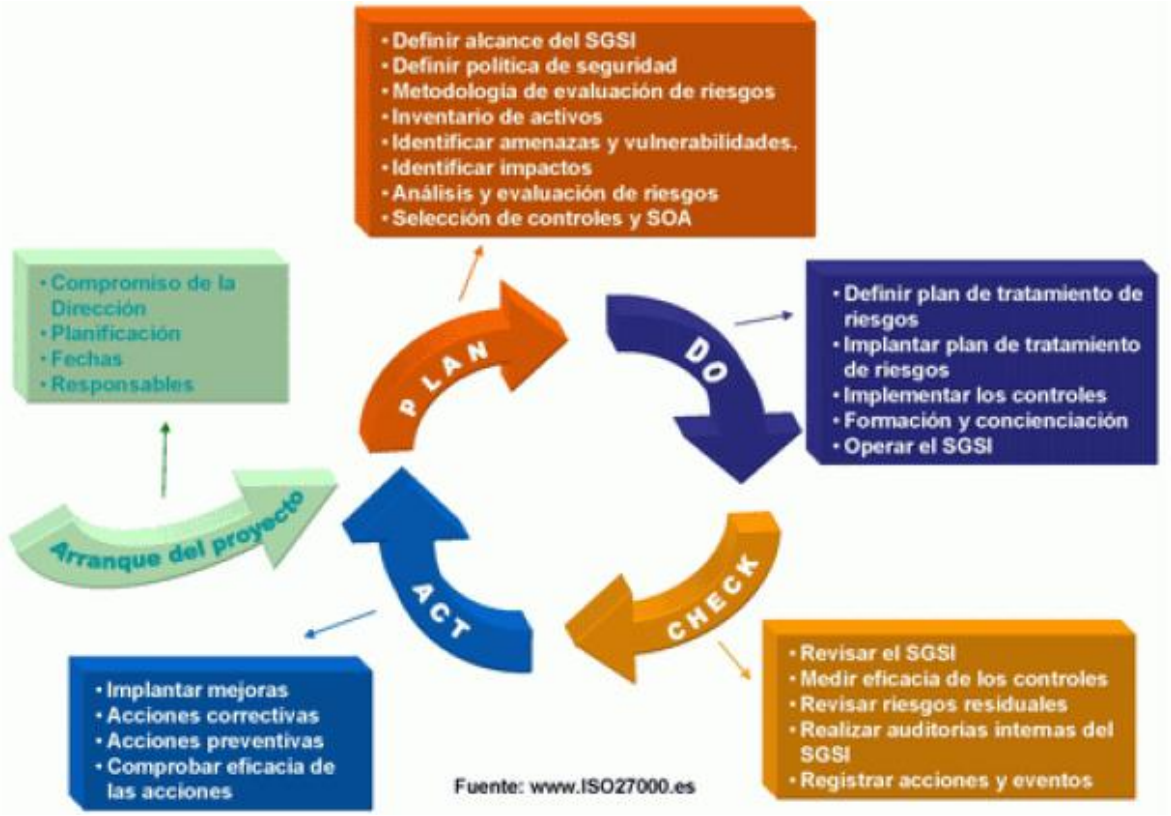
Vulnerabilidad de día cero («zero-day exploit»): *Un ataque mediante una vulnerabilidad de día cero es difícil de prevenir; el término abarca una vulnerabilidad que todavía no es conocida por los autores del programa*

ANEXOS

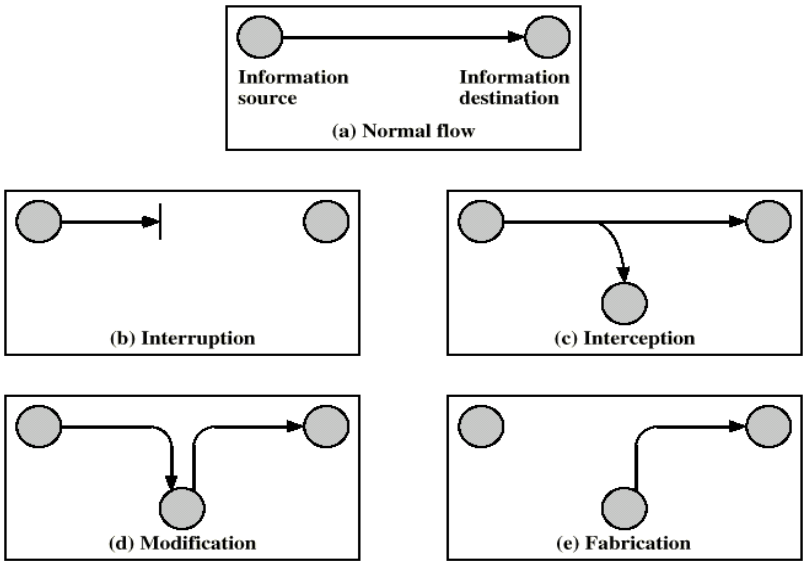
Anexo 1: Los triados requisitos de seguridad.



Anexo 2: Ciclo PHVA



Anexo 3: Atáques de Seguridad



Anexo 4: Seguridad de servicios x.800

Autenticación La identidad de que la entidad comunicante es la que dice ser. **Peer autenticación** Se utiliza en asociación con una conexión lógica para proporcionar la confianza a la identidad de las entidades conectadas. **Origen de los datos de autenticación** En una transferencia de conexión, proporciona seguridad de que la fuente de los datos recibidos es como se reivindica. **CONTROL DE ACCESO** La prevención del uso no autorizado de un recurso. Es decir, es el control de servicios que pueden tener acceso a un recurso. **Confidencialidad de los datos** La protección de los datos y de la no autorizada revelación de los mismos. **La confidencialidad de conexión** La protección de todos los datos del usuario en una conexión. **La confidencialidad sin conexión** La protección de todos los datos del	**Integridad de los datos** La seguridad de que los datos recibidos son exactamente los enviado por una entidad autorizada (es decir, no contienen modificación, inserción, eliminación, o bien la repetición). **Integridad relación con la recuperación** Prevé la integridad de todos los datos del usuario en una conexión y detecta ninguna modificación, inserción, eliminación, o la repetición de los datos dentro de un conjunto o secuencia de datos entero, con el intentó de la recuperación. **Integridad conexión sin recuperación** Como el anterior, pero proporciona solamente la detección sin recuperación. **Integridad selectiva por campo en la conexión** Prevé la integridad de campos seleccionados dentro de la los datos de usuario de un bloque de datos transferidos a través de una conexión y toma la forma de determinación de si los campos seleccionados han sido modificados, se inserta, eliminado o reproducido.

usuario en un único bloque de datos. **La confidencialidad selectiva por campo** La confidencialidad de campos seleccionados dentro de los datos del usuario sobre una conexión o en un solo bloque de datos. **La confidencialidad del flujo del tráfico** La protección de la información que podría ser derivado de la observación de los flujos de tráfico.	**Integridad sin conexión** Proporciona a la integridad de una sola conexión bloque de datos y pueden tomar la forma de detección de modificación de datos. Además, una forma limitada de detección de reproducción puede ser proporcionada. **Selectivo-campo sin conexión e Integridad** Proporciona a la integridad de los campos seleccionados dentro de un solo bloque de datos sin conexión; toma la forma de determinación de si los campos seleccionados han sido modificados. **NO RECHAZO** Proporciona protección contra la denegación por una de las entidades implicadas en una comunicación de tener participado en la totalidad o parte de la comunicación. **No repudio, Origen** La prueba de que el mensaje fue enviado por la parte especificada. **No repudio, Destino** La prueba de que el mensaje fue recibido por el especificado.

Anexo 5: Mecanismo de seguridad X.800

MECANISMOS DE SEGURIDAD ESPECÍFICAS Se puede incorporar en la capa de protocolo apropiado con el fin de proporcionar algunos servicios OSI de la seguridad. **Cifrado** El uso de algoritmos matemáticos para transformar datos en una forma que no es fácilmente inteligible. La transformación y la posterior recuperación de los datos dependen de un algoritmo y cero o más claves de cifrado. **Firma digital** Datos adjuntos o una transformación criptográfica que permita a un receptor demostrar el origen y la integridad de la unidad de datos y proteger contra la falsificación. **Control de acceso** Una variedad de mecanismos que hacen cumplir los derechos de acceso a recursos. **Integridad de los datos** Una variedad de mecanismos empleados para asegurar la integridad de una unidad de datos o de la corriente de unidades de datos. **Cambio de autenticación** Un mecanismo destinado a garantizar la identidad de una entidad por medio de intercambio de información **Control de enrutamiento** Permite la selección de especial físicamente seguro de rutas de determinados datos y permite a los cambios de itinerario, especialmente cuando se sospecha de una violación de la seguridad. **Notarización** El uso de un tercero de confianza para garantizar ciertas propiedades de un intercambio de datos.	**MECANISMOS DE SEGURIDAD GENERALIZADOS** Mecanismos que no son específicos de ningún servicio de seguridad OSI o capa de protocolo particular. **Funcionalidad de confianza** Lo que se percibe como correcta con respecto a algunos criterios (por ejemplo, según lo establecido por una política de seguridad). **Etiqueta de seguridad** El marcado unido a un recurso (que puede ser una unidad de datos) que designe los atributos de seguridad de ese recurso. **Detección de eventos** La detección de eventos relevantes para la seguridad. **Auditoria de seguridad** Los datos recogidos para facilitar una auditoria de seguridad, es una revisión independiente y examen de los registros y actividades del sistema. **Recuperación de seguridad** Se ocupa de las peticiones de los mecanismos, como evento Funciones para el manejo y gestión, y toma las acciones de recuperación.

Anexo 6: Relación entre servicios y mecanismos de seguridad

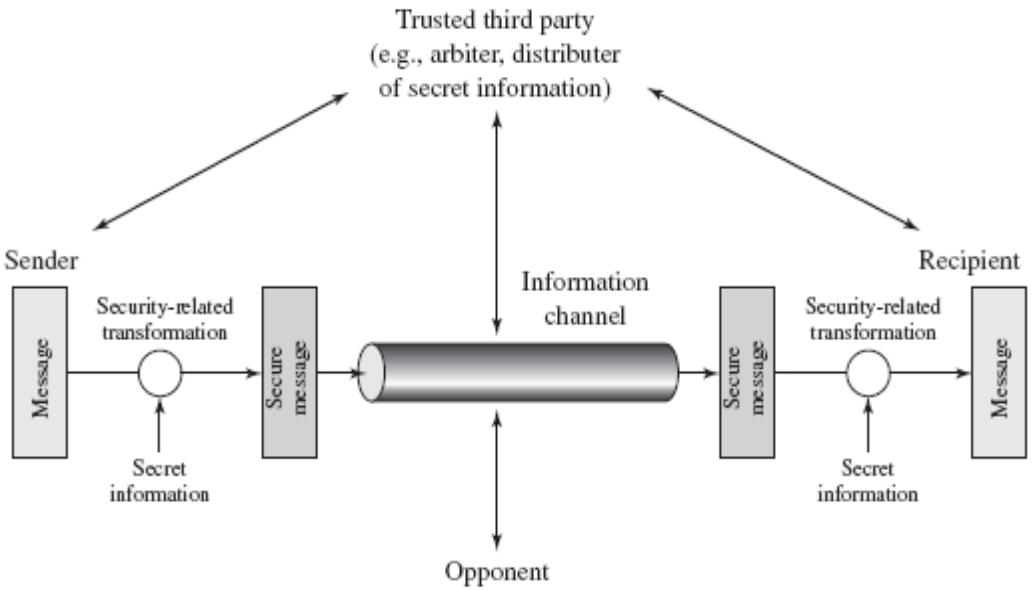
Mecanismo / Servicio	Cifrado	Firma digital	Control de acceso	Integridad de datos	Intercambio de automa-tización	Relleno de tráfico	Control de encami-namiento	Notari-zación
Autenticación de la entidad par	S	S	·	·	S	·	·	·
Autenticación del origen de los datos	S	S	·	·	·	·	·	·
Servicio de control de acceso	·	·	S	·	·	·	·	·
Confidencialidad en modo con conexión	S	·	·	·	·	·	S	·
Confidencialidad en modo sin conexión	S	·	·	·	·	·	S	·
Confidencialidad de campos seleccionados	S	·	·	·	·	·	·	·
Confidencialidad del flujo de tráfico	S	·	·	·	·	S	S	·
Integridad en modo con conexión con recuperación	S	·	·	S	·	·	·	·
Integridad en modo con conexión sin recuperación	S	·	·	S	·	·	·	·
Integridad de campos seleccionados en modo con conexión	S	·	·	S	·	·	·	·
Integridad en modo sin conexión	S	S	·	S	·	·	·	·
Integridad de campos seleccionados en modo sin conexión por campos selectivos	S	S	·	S	·	·	·	·
No repudio. Origen	·	S	·	S	·	·	·	S
No repudio. Entrega	·	S	·	S	·	·	·	S

· Se considera que el mecanismo no es apropiado.

S Sí: El mecanismo es apropiado, por sí mismo o en combinación con otros mecanismos.

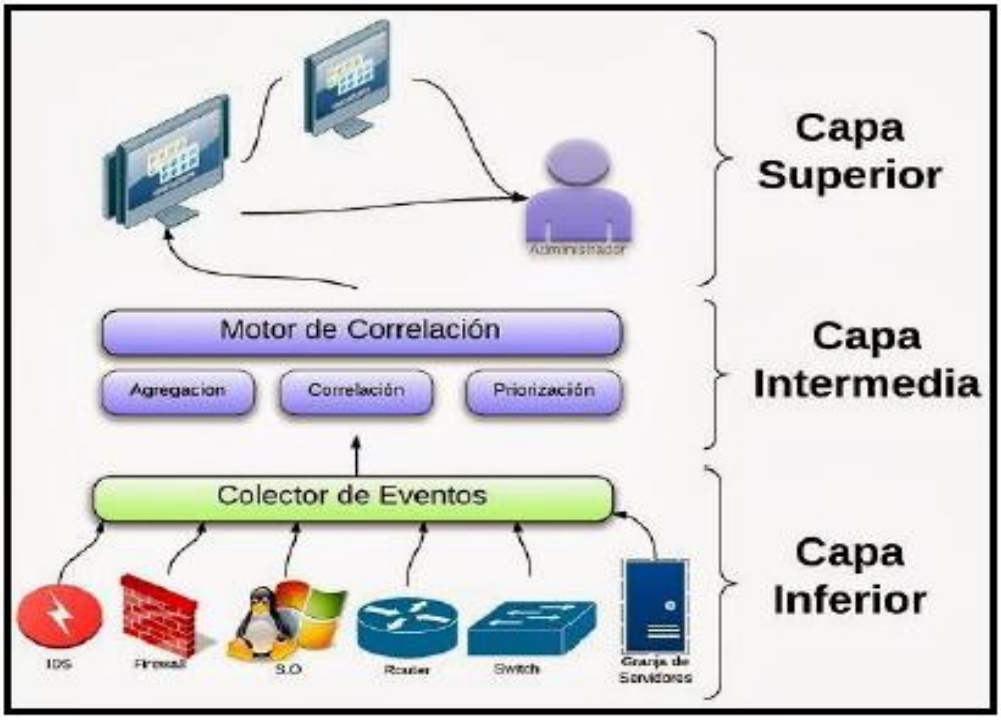
Nota – En algunos casos, el mecanismo proporciona más de lo que es necesario para el servicio en cuestión, no obstante lo cual podrá utilizarse.

Anexo 7: Modelo para la seguridad en redes

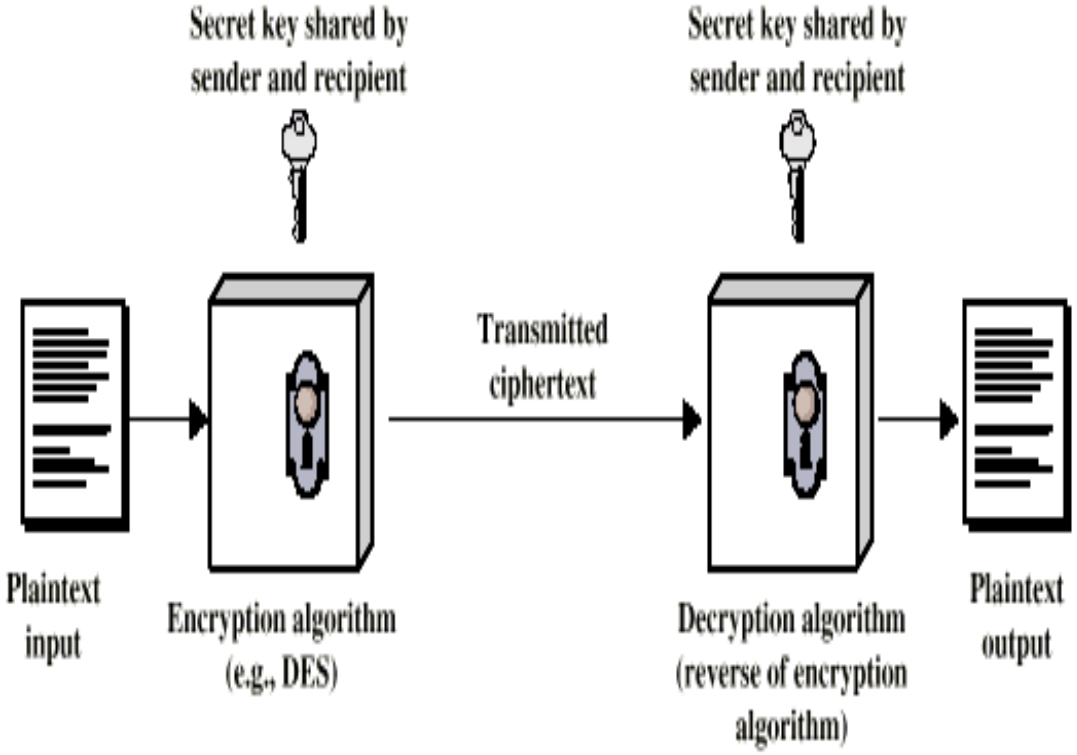


Anexo 8: Escritorio Kali Linux

Anexo 9: Arquitectura de OSSIM



Anexo 10: Modelo simplificado de encriptación simétrica



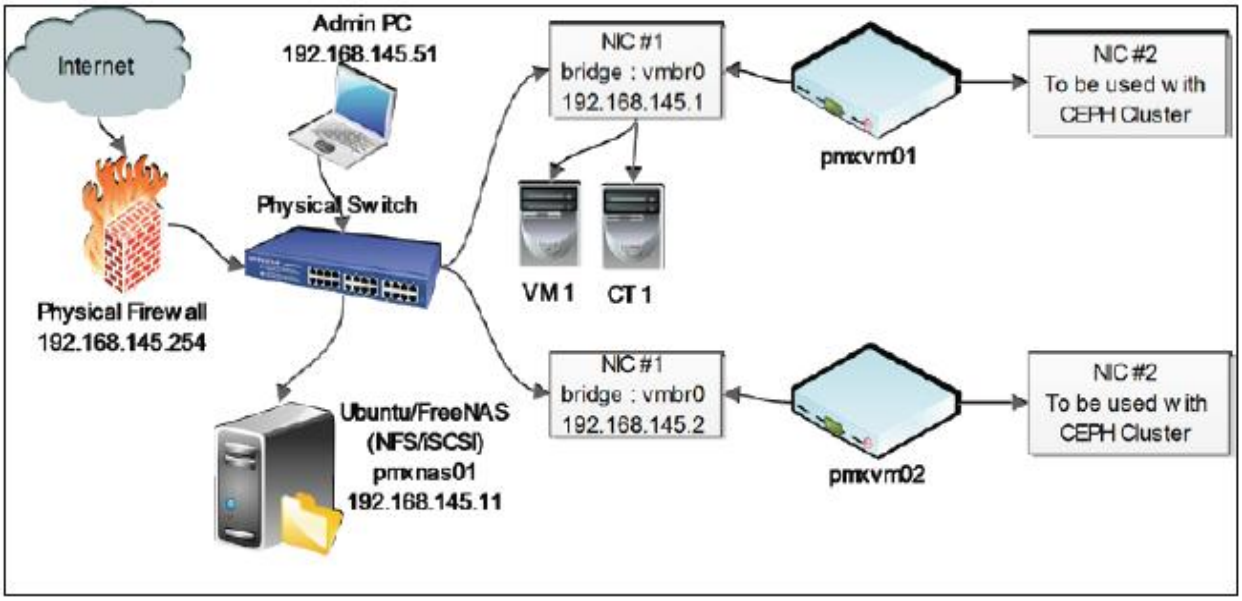
Anexo 11: Tipos de ataques en un mensaje encriptado

Solo texto cifrado	**Algoritmo de Encriptación, Texto cifrado.**
Texto nativo conocido	Algoritmo de Encriptación, Texto cifrado, uno o más texto nativos y sus correspondientes textos cifrados compuestos con la clave secreta.
Texto nativo escogido	Algoritmo de Encriptación, Texto cifrado, Texto nativo, escogido por el criptoanalista y su correspondiente texto cifrado compuesto con la clave secreta.
Texto cifrado escogido	Algoritmo de Encriptación, Texto cifrado, Texto cifrado, escogido por el criptoanalista y su correspondiente texto nativo compuesto con la clave secreta.
Texto escogido	Algoritmo de Encriptación, Texto nativo, escogido por el criptoanalista y sus correspondiente texto cifrado compuesto con la clave secreta, Texto cifrado, Texto cifrado, escogido por el criptoanalista y sus correspondiente texto nativo compuesto con la clave secreta.

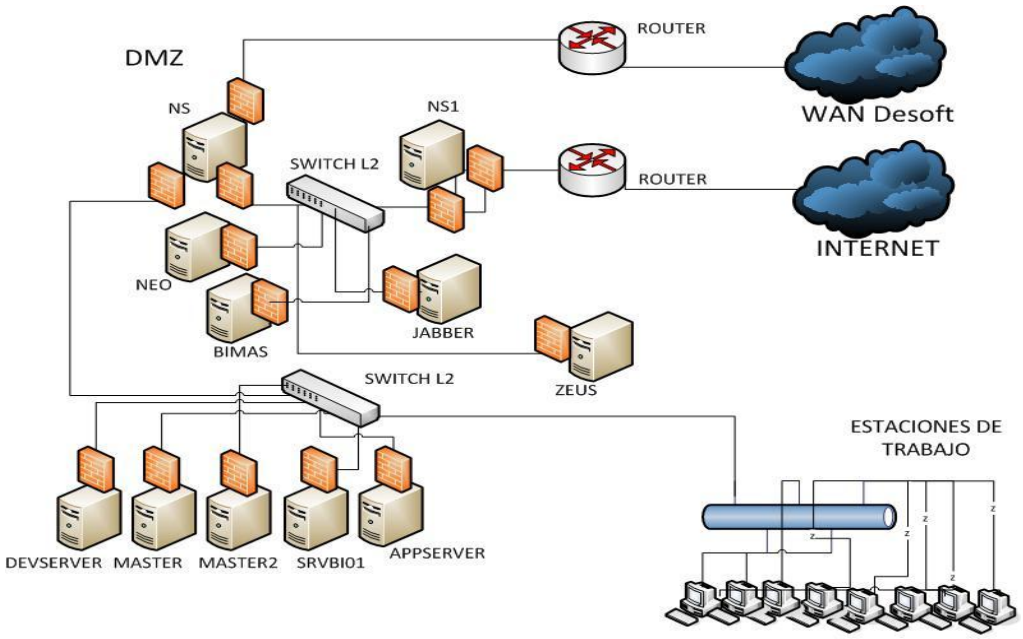
Anexo 12: Lista de componentes de hardware para un cluster básico con proxmox

Component type	Brand/model	Quantity
CPU/Processor	Intel i3-2120 3.30 Ghz 4 Core	2
Motherboard	Asus P8B75-M/CSM	2
RAM	Kingston 8 GB 1600 Mhz DDR3 240 Pin Non-ECC	3
HDD	Seagate Momentus 250 GB 2.5" SATA	2
USB stick	Patriot Memory 4 GB	1
Power supply	300+ Watt	3
LAN switch	Netgear GS108NA 8-Port Gigabit Switch	1

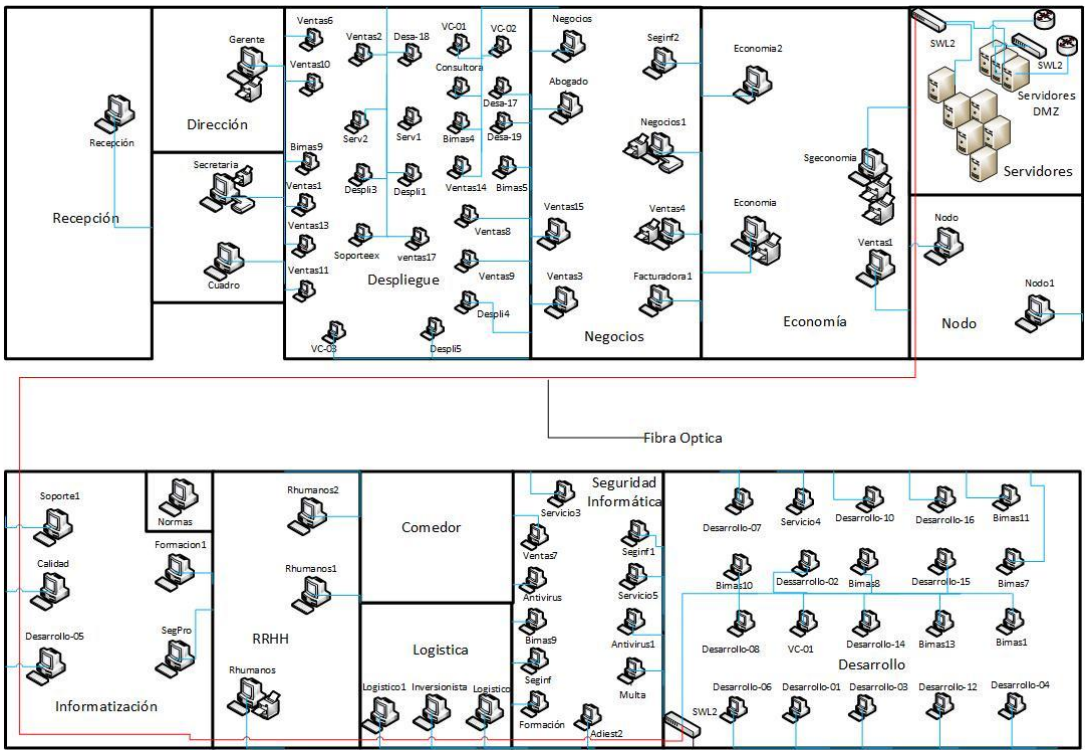
Anexo 13: Diagrama de red de un cluster básico con Proxmox



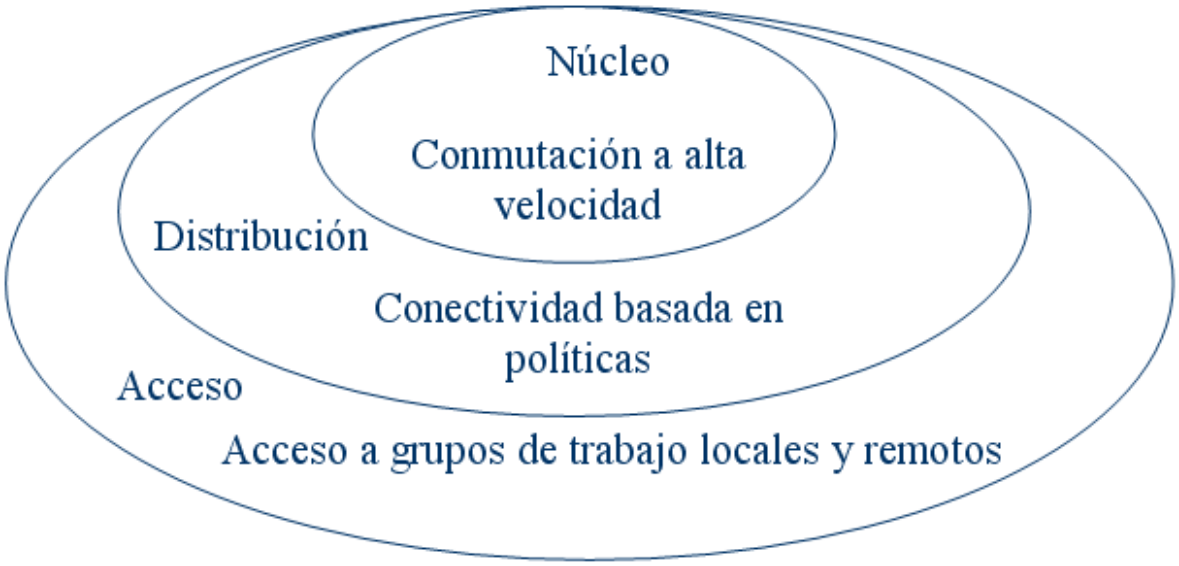
Anexo 14: Plano lógico de la actual red de la División Desoft Villa Clara



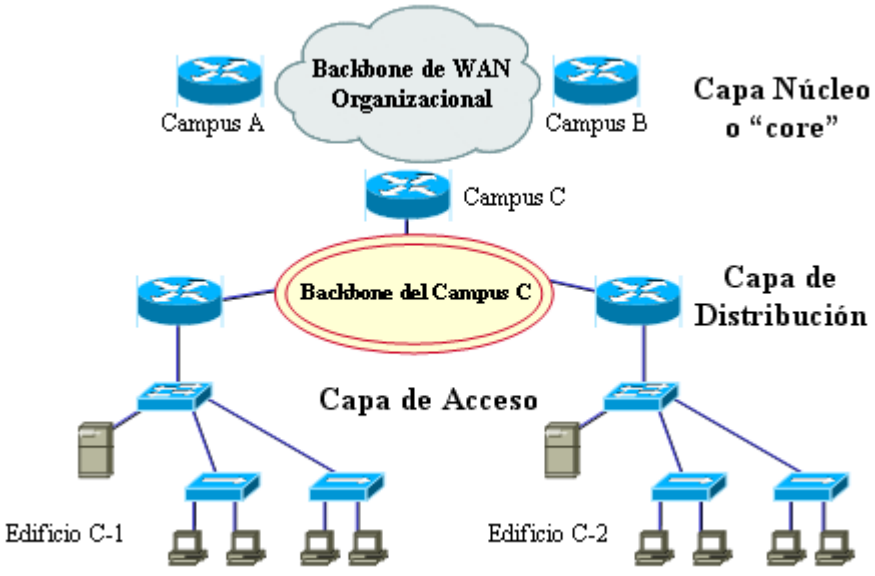
Anexo 15: Plano físico de la actual red de Desoft VC



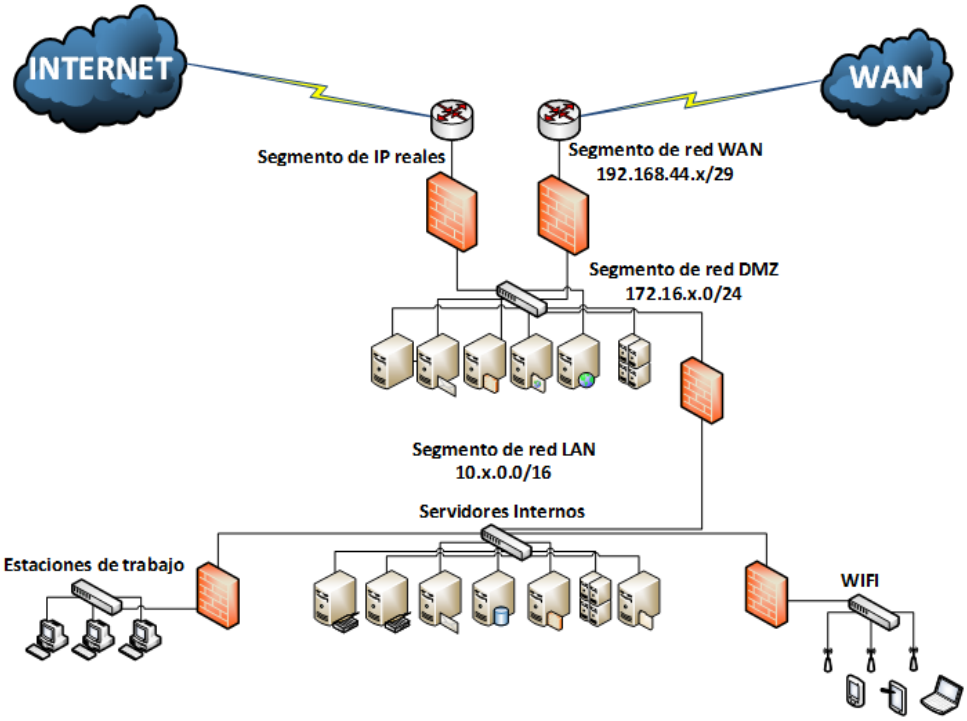
Anexo 16: Representación del modelo jerárquico



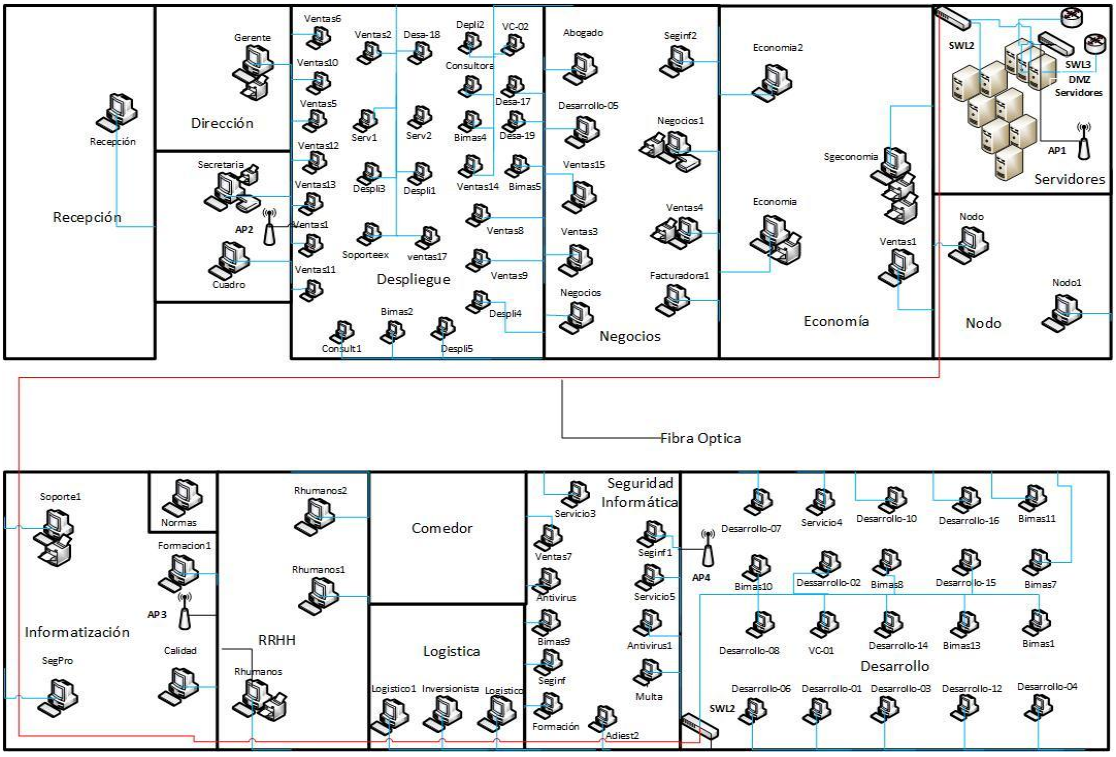
Anexo 17: Estructura del diseño jerárquico de redes



Anexo 18: Plano lógico de la futura red de la División Desoft Villa Clara



Anexo 19: Plano físico de la futura red de la División Desoft Villa Clara



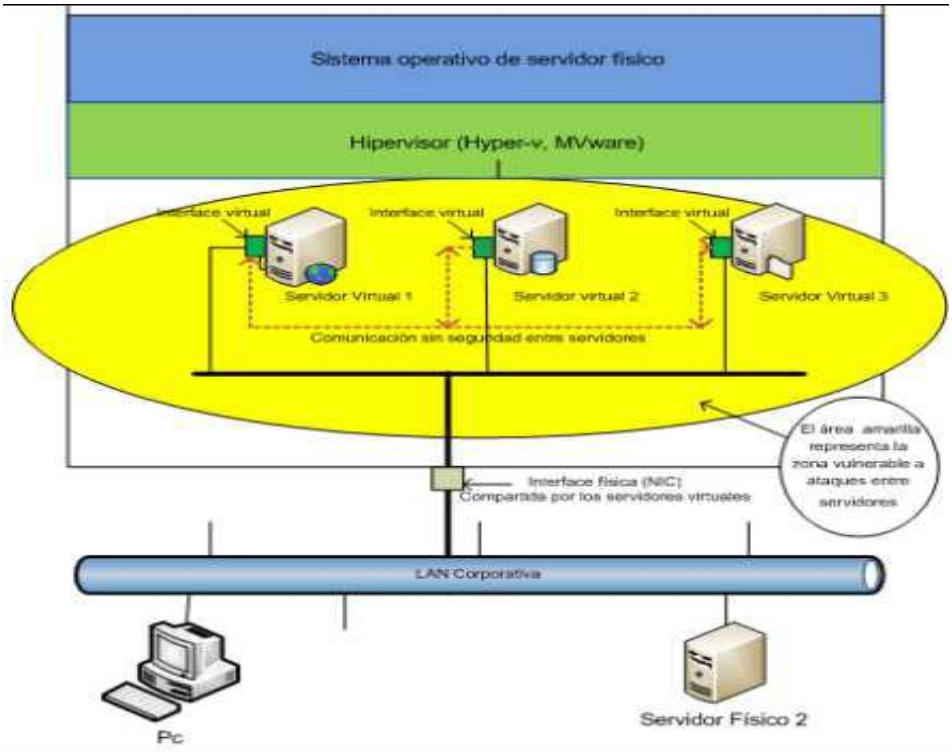
Anexo 20: Requerimientos de hardware para cada uno de los servicios instalados en la División Desoft Villa Clara

Servicio	Procesador	Memoria	Capacidad de Disco Duro
Correo	Intel Xeon E5-2630 @ 2.30GHz	8 GB	1 TB
Ftp	Intel Xeon E5-2630 @ 2.30GHz	1 GB	500 GB
jabber corporativo	Intel Xeon E5-2630 @ 2.30GHz	2 GB	40 GB
BIMAS	Intel Xeon E5-2630 @ 2.30GHz	4 GB	160 GB
AD, DNS, DHCP	Intel Xeon E5-2630 @ 2.30GHz	4 GB	160 GB
Gestores de base de datos	Intel Xeon E5-2630 @ 2.30GHz	8 GB	1 TB
Proxy	Intel Xeon E5-2630 @ 2.30GHz	8 GB	500 GB
Analizadores de tráfico	Intel Xeon E5-2630 @ 2.30GHz	4 GB	500 GB
Versionadores de código	Intel Xeon E5-2630 @ 2.30GHz	4 GB	1 TB
Servidor Radius	Intel Xeon E5-2630 @ 2.30GHz	4 GB	500 GB

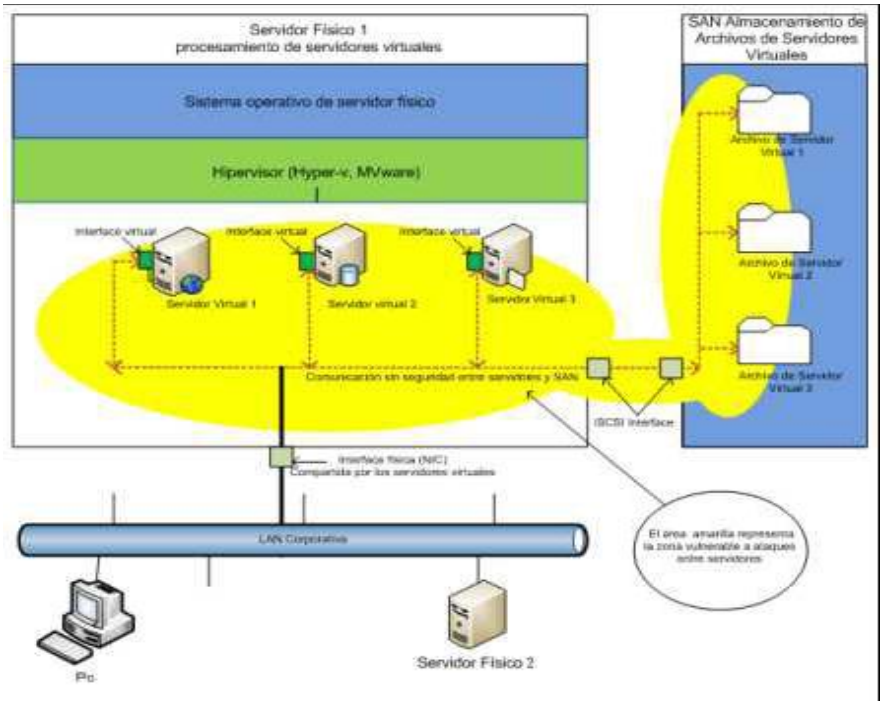
Anexo 21: Listado de precios y descripción del hardware

Código	Descripción	Cantidad	Categoría	Precios cuc	Precios totales
NF5280M3 GAMA ALTA RACK	Chassis 8 disk chassis (19¨ rack)// Processor E5-2630 / 6 cores / 2.2GHz Number of processors 2// Memory 8x8GB ECC DDR3 1600MHz// HDD 1TB 2.5¨ SAS hot swap Cant. HDD 5// RAID controller High performance RAID controller RAID level 5// NIC 4 x Gigabit ethernet NIC// Operating system No OS// Optical driver slim DVD-RW// Power supply	2	Servidor	$ 4,862.30	$ 9724.6
STORAGE AS1000G6	Chassis 12 disk chassis (19¨ rack)// Cache 48GB cache// HDD 1TB 2.5¨ SAS hot swap Cant. HDD 12// Host interface 4 x 10GB iSCSI// Operating system No OS// Optical driver slim DVD-RW// Power supply redundant power supply //Accesories Sliding rails for 19¨ rack	1	Servidor	$ 15,896.23	$ 15,896.23
KVM 2162DS	Dell KVM 2162DS Remote Console Switch Keyboard/Video/Mouse Analog Switch (4x USB 2.0 Server Interface Ports, includes CAT 5 cables, TAA	1	Remote Console Switch Keyboard/ Video/Mouse	$ 3,563.45	$ 3563.45
10201181	MONITOR LCD VIEWSONIC VG2228WM 22PULG	1	Monitor	$ 275.00	$ 275.00
AT-9924T	Switch layer 3 modular 24 port 10/100/1000	1	Switch layer 3	$ 4,173.25	$ 4,173.25
					$ 33632.53

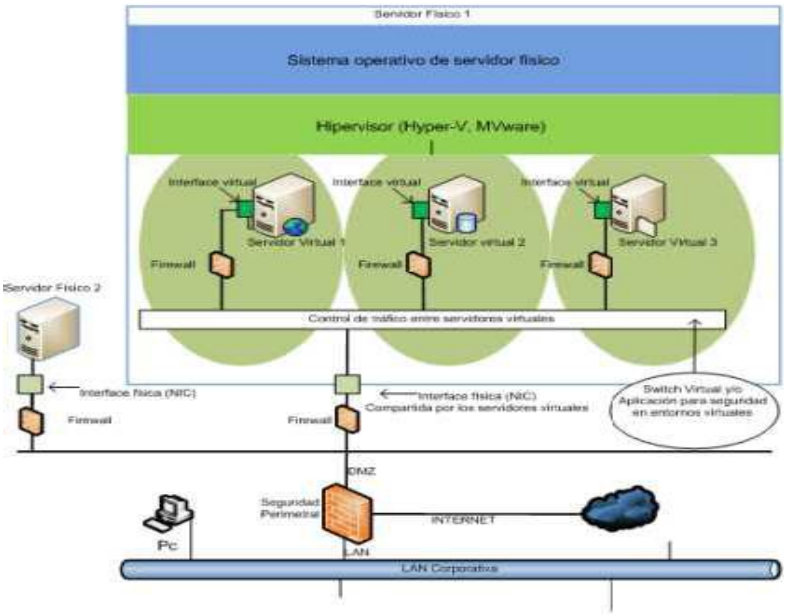
Anexo 22: Servidores virtuales en un mismo servidor físico sin seguridad entre ellos



Anexo 23: Esquema de servidores virtuales con procesamiento en un servidor y almacenamiento en una SAN

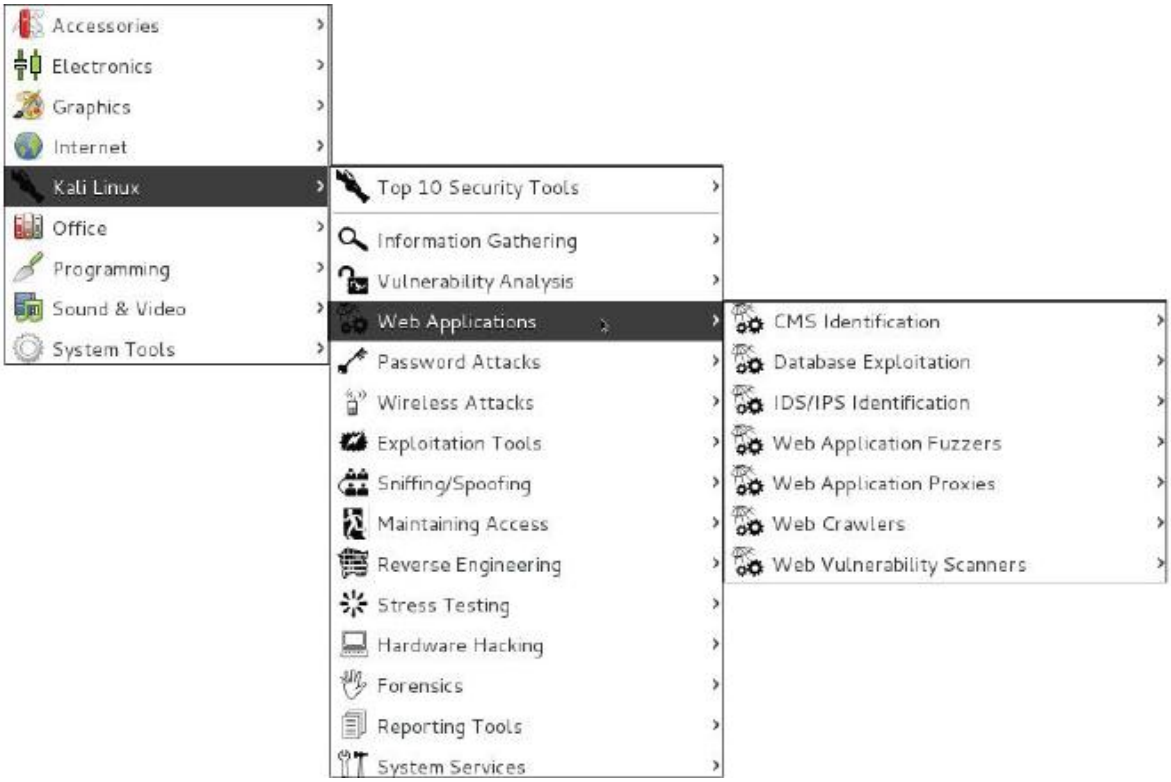


Anexo 24: Esquema de servidores virtuales con Mecanismos de seguridad instrumentados

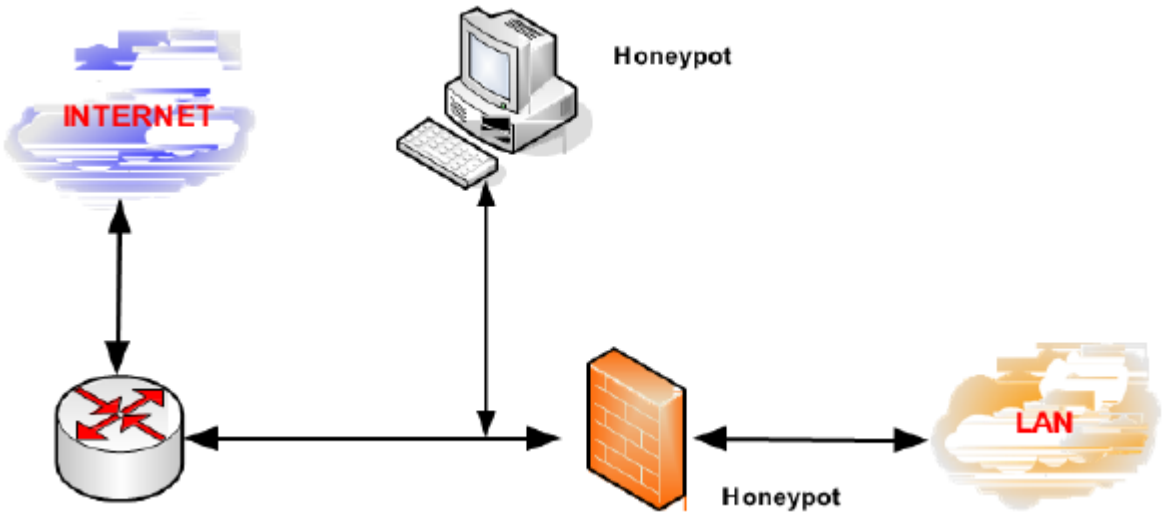


Anexo 25: Interfaz de administración de Proxmox

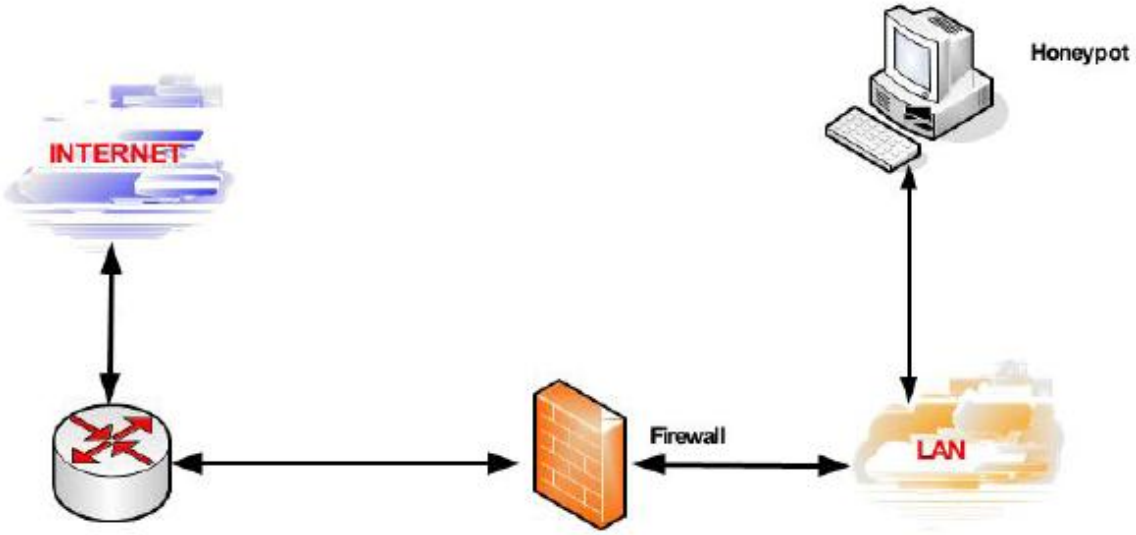
Anexo 26: Herramientas de Kali Linux utilizadas para los test de penetración



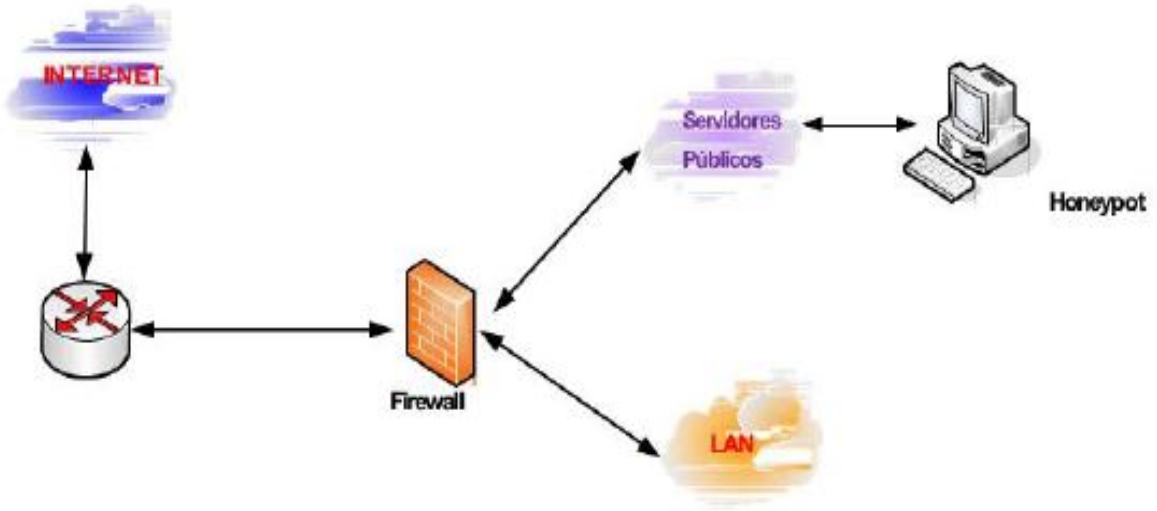
Anexo 27: Honeypots antes del firewall



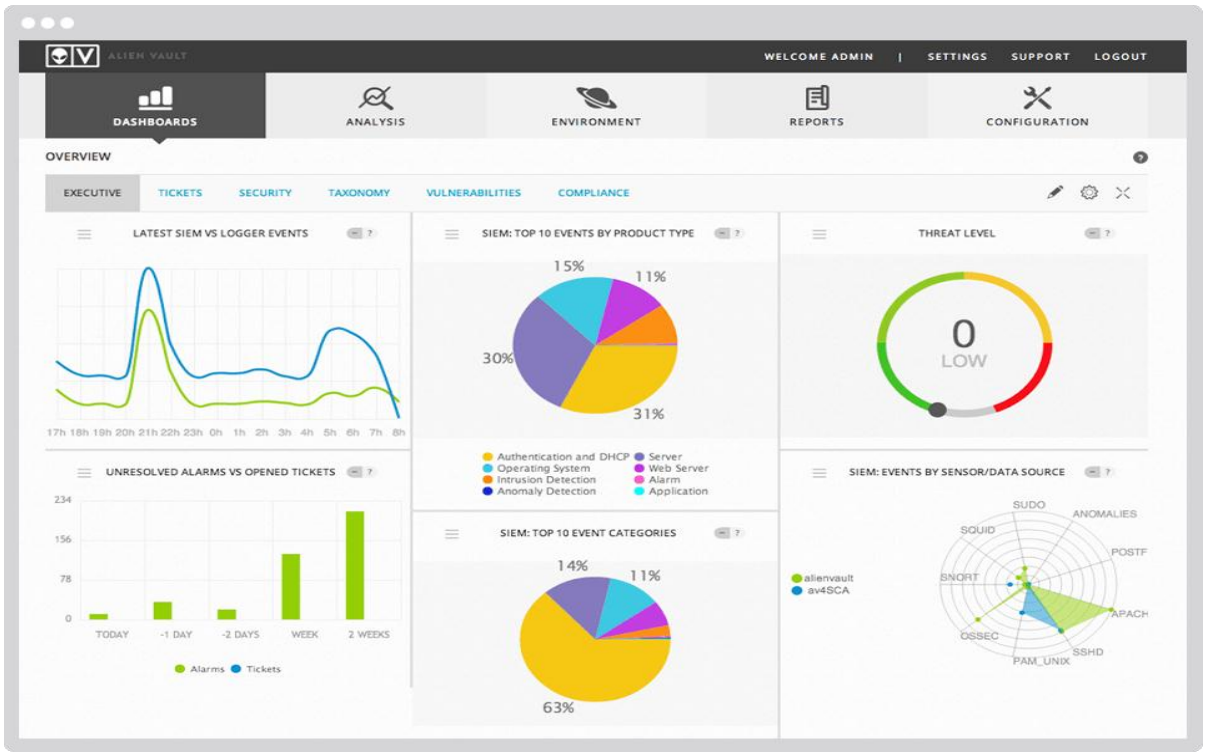
Anexo 28: Honeypots detrás del firewall



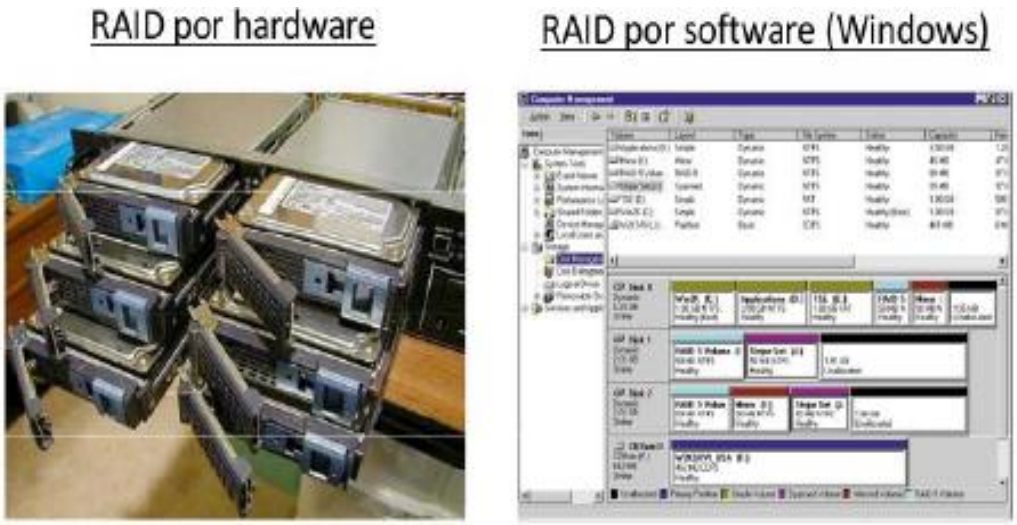
Anexo 29: Honeypots en DMZ



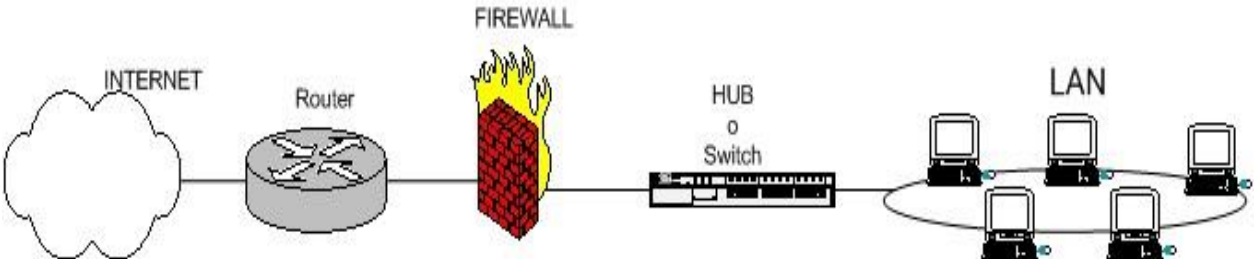
Anexo 30: Interfaz Web de administración de OSSIM



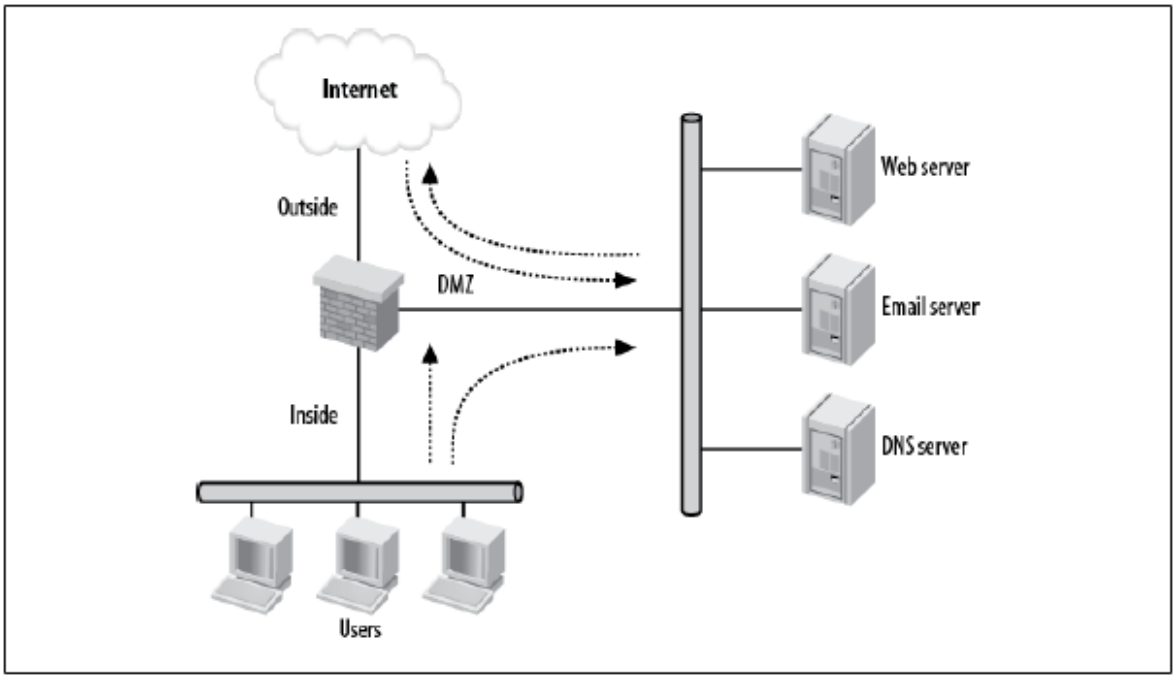
Anexo 31: Formas de Raid



Anexo 32: Esquema de cortafuego típico entre red local e Internet



Anexo 33: Esquema de cortafuegos entre red local e Internet con zona DMZ para servidores expuestos



Anexo 34: Red con múltiple DMZs