Capítulo I

Marco Teórico

1.1 Software Libre

1.1.1 Historia, definición y características.

"El software libre es una cuestión de libertad, no de precio. Para entender el concepto, debería pensar en libre como en libre expresión, no como en barra libre" (Definición de Richard Stallman sobre el software libre)

El termino software libre nació en el año de 1988. Tiene como principal característica el permitir a cualquier usuario acceder a su código fuente sin pagar por su uso, a diferencia del software privativo en donde no se posible observar su código fuente y se tiene que pagar por su uso.

A principios de la década de los 70's y 80's comenzaron a desarrollarse notables proyectos como son SPICE y Tex, ambos enfocados a mantener la filosofía que poco a poco iba perdiendo fuerza, el primero desarrollado en 1973 en la Universidad de California por Donald Pederson estaba enfocado a la simulación de circuitos electrónicos y el segundo desarrollado por Donald Knuth en 1978 el cual es un sistema de escritura cuyo objetivo es producir documentos con un formato de calidad, además de uno de los más importantes en materia de software libre UNIX.

UNIX creado por Thompson y Ritchie desde 1972 e impulsado por los laboratorios Bellde AT&T, fue un pilar fundamental para el software libre, ya que éste fue desarrollado inicialmente bajo los términos de licencia que permitían su libre distribución,modificación y estudio, éste tuvo su mayor impulso en la Universidad de California en Berkeley quien posteriormente por problemas de licenciamiento y falta de acceso al mismo fue encareciendo el proyecto hasta llegar al grado en que dicha institución seguía un proceso legal con la división Unix System Laboratories de AT&T por publicar el código de este *sistema operativo*, lo que ocasionó que se perdieran los términos de distribución de versiones que actualmente están establecidos en la filosofía del software libre.

Otro principal fundador de la filosofía de software libre es Richard Stallman. Quien por problemas presentados con una impresora HP, y al no poder acceder a su código, lo motivó a desarrollar controladores de uso libre y distribuirlos sin ningún costo a usuarios que presentaran los mismos inconvenientes. Esta ideología fue rápidamente adoptada por personas que tenían los mismos problemas con software propietario, naciendo así la comunidad de software libre.

Los grupos de software libre comenzaron desarrollando controladores para hardware que presentaban problemas similares al presentado por Richard Stallman. Con el paso del tiempo se fueron creando alternativas de uso libre hacia el software propietario. En la actualidad se ha logrado avanzar a tal grado que existen sistemas operativos trabajando completamente con software libre: como es el caso de distribuciones Linux basadas en *Debian* o *Red Hat*, entre otras.

La idea del software libre radica en que al momento de liberar el código, se pretende que se realicen mejoras sobre este software a tal grado que logre igualar o inclusive superar en calidad al software propietario.

Todo software que sea considerado como software libre, deberá cumplir con las siguientes características:

- Libre redistribución.El poseedor del software tiene la libertad de venderlo o regalarlo. Sin embargo, existen licencias que obligan a que la distribución sea de forma gratuita.
- El código fuente podrá ser observado de manera libre sin tener que pagar por él o en su defecto se deberá obtener libremente.
- El poder realizar mejoras sobre dicho software; siempre y cuando se respete el copyright de algunas licencias.
- La integridad del código fuente del autor. Se debe respetar al autor de dicho software; los cambios realizados por terceras personas deberán ser publicados como parches o actualizaciones "las cuales pueden ser nuevas mejoras o nuevas funcionalidades" y no como un nuevo software libre.
- Toda persona que trabaje sobre alguna modificación deberá ser incluida.
- Todo software libre debe estar regido por alguna licencia; sin embargo esta licencia deberá ser neutral y no deberá obligar a que alguna versión o parche más reciente se deba distribuir sobre esta misma licencia, ni el volverlo software privativo.

1.1.2 GNU/Linux.

El proyecto GNU/Linux fue iniciado en 1983 por Richard Stallman. Esta filosofía tiene como principal objetivo el crear software libre con el fin de tener por lo menos alguna alternativa de uso libre hacia aplicaciones propietarias.

Fue tal el crecimiento de este proyecto que en cuestión de años se unieron cientos de personas. Logrando en 1991 la liberación del proyecto GNU/Linux cuyo objetivo radicó en la creación de un sistema operativo basado en su totalidad por software libre, bajo un núcleo Unix. Con la liberación de dicho proyecto se logró también el nacimiento del kernel Linux. En este kernen se han desarrollado diversos sistemas operativos, como lo son Red Hat, Centos, Fedora, Debian y Ubuntu entre otros.

Tal ha sido el éxito de la filosofía GNU/Linux que hoy en día es uno de los principales competidores de los sistemas operativos Windows y Unix. Teniendo un impacto a tal grado que cada día se unen más personas a esta filosofía.

Todo software que sea considerado como Software Libre, sin importar el kernel que utilice, debe cumplir con las siguientes libertades.

1.1.3 Libertades del Software Libre.

- **Libertad 0:** Libertad de ejecutar el programa con cualquier propósito.
- **Libertad 1:** Libertad de tener acceso al código fuente del software, poder estudiarlo y modificarlo para obtener el objetivo deseado.
- **Libertad 2:** Libertad de redistribución del software para ayudar al prójimo.
- **Libertad 3:** Libertad de redistribuir versiones modificadas a terceros, con el objetivo de realizar modificaciones y ayudar al mejor funcionamiento del software.

Todas las libertades mencionadas deberán cumplirse. Si por algún motivo se ignora alguna de ellas, el software no se considerará libre.

Aunque el software libre ha adquirido bastante popularidad en la actualidad, existen demasiadas trabas puestas por empresas, especialmente Microsoft, que evitan que siga creciendo esta ideología. En la siguiente sección se realiza una comparación entre el software propietario y el software libre.

1.1.4 Software propietario vs software libre

Como se ha mencionado, el principal objetivo del software libre es tener alguna alternativa de uso libre contra cada software propietario que exista. Sin embargo, en muchos casos las aplicaciones desarrolladas bajo software con licenciamiento de uso son mejores que las aplicaciones desarrolladas sobre la filosofía Open Source: el problema radica cuando los desarrolladores de software libre trabajan en desarrollar alguna alternativa de uso libre hacia el software propietario y no conocen, o pueden observar el código del software propietario. Este inconveniente causa dificultades en el desarrollo e implementación del nuevo software de uso libre.

Sin embargo, cada vez se logran desarrollar mayores aplicaciones de software libre que compiten a la par con el software propietario. Un ejemplo son los sistemas operativos (S.O.) Ubuntu y Fedora: en cada actualización disponible de sus versiones, los creadores logran volver a estos S.O. más amigables hacia el usuario final, ofreciendo software libre de fácil utilización y muy potente. Logrando que más personas prueben estos sistemas operativos y decidan instalarlos.

En contramedida, el software propietario ofrece una mejor calidad: las empresas desarrolladoras de software propietario se comprometen a dar mantenimiento y garantizar que el software funciona correctamente. Sin embargo, el software propietario es objeto de muchos ataques realizados por *perpetradores*. Como ejemplo, la cantidad de malware y ataques desarrollados a los S.O. Windows y software desarrollado en esta plataforma. Mientras que en los S.O. Linux prácticamente no se presenta este problema.

Otro inconveniente que presenta el software propietario es la piratería: la mayoría del software propietario es de alto costo, y no poder observar su código fuente motiva a los perpetradores a desarrollar aplicaciones con el objetivo de *crakear* o parchar el software propietario, logrando el uso de éste de forma ilegal.

La siguiente tabla muestra una comparación realizada sobre el software propietario y el software libre.

Tabla 1.1 Software Propietario vs software libre

	Ventajas	Desventajas
Software Propietario	✓ Soporte proporcionado por el fabricante. ✓ Actualizaciones. ✓ Funcionamiento deseado. ✓ Se desarrollan aplicaciones muy complejas. ✓ Robusto. ✓ Página web del proveedor. ✓ Guía de usuario. ✓ Fácil instalación.	× Licencia de uso (En algunos casos es muy cara). × Susceptible a virus. × Tener que pagar nuevas licencias para actualizaciones. × En la mayoría de los casos solo se desarrolla sobre los sistemas operativos más comerciales. × No se puede observar el código fuente. × No se pueden realizar cambios. × No se puede redistribuir, a excepción de comprar licencias que soporten múltiples equipos.
Software Libre	✓ Gratuito. ✓ Se mejora constantemente. ✓ Libertad de distribución. ✓ Observar el código fuente. ✓ Realizar cambios de acuerdo a las necesidades requeridas. ✓ Multiplataforma. ✓ Generalmente está libre de virus. ✓ La licencia impide que se vuelva privativos. ✓ Mejoras desarrolladas por terceras personas respetando los derechos de autor.	× En algunos casos es difícil su instalación. × No se tiene soporte en algunas aplicaciones desarrolladas × En algunos casos no funcionan correctamente. × No siempre se tienen alternativas software libre que logren sustituir completamente a algún software propietario. × No se tienen actualizaciones constantes del software. × En algunos casos se tiene una página web del software obsoleta.

Además de lo descrito del software libre, se tiene el inconveniente que los desarrolladores al utilizar software libre crean una nueva aplicación y la traten de volver propietaria. Por este motivo existen diversas licencias que protegen al software libre contra cualquier tipo de privatización y garantizan el respeto de los derechos de autor (copyleft).

1.1.5 Tipos de licencias de Software Libre.

Una licencia es una autorización formal que proporciona el autor del software para su uso. Cada una de las licencias descritas tiene como objetivo proteger los derechos de autor. Dependiendo del tipo de tipo de licencia puede ser posible convertir software libre a software propietario, siempre y cuando se respeten los términos establecidos en la licencia.

Las siguientes licencias son las más usadas por el Software Libre.

1.1.5.1 General Public License (GNU GPL)

Licencia creada por la Free Software Fundation en 1988. Su principal propósito es el declarar al software licenciado sobre ella como software libre y protegerlo de cualquier tipo de privatización. Tiene como restricciones la libre distribución, modificación y uso del software.

Otra característica radica en el respeto de los derechos de autor "copyleft" de las mejoras realizadas en el software. También es importante señalar que las nuevas versiones creadas deberán de estar regidas sobre la misma licencia (no se podrá cambiar el licenciamiento del software). Todo esto con el objetivo de no volver el software propietario al momento de realizar mejoras sobre él.

Actualmente esta licencia se encuentra en la versión tres (GNU GPL V3) publicada el 29 de junio del 2007.

1.1.5.2 Berkeley Software Distribution (BSD)

Licencia creada por la Universidad de California en 1990 y es otorgada principalmente para sistemas BSD (Berkeley Software Distribution).

Su característica principal radica en que el autor renuncia a todas las modificaciones realizadas a su software y no se hace responsable de los efectos que tengan dichas modificaciones. Esta licencia es más permisiva a comparación de la licencia GNU GPL, debido a que la persona que realice modificaciones sobre el software regido con dicha licencia tendrá la libertad de cambiar a cualquier otra licencia de software libre o incluso volver su versión propietaria.

Uno de los objetivo a cumplir en este proyecto de tesis es buscar alguna alternativa de uso libre que soporte el protocolo Netflow. El protocolo Neflow es un estándar creado por Cisco que se basa en la generación de estadísticas obtenidas por router o switch mediante el monitoreo de red.

El monitoreo de red es una técnica altamente utilizada en empresas por administradores de redes para observar la salud de la red. En la siguiente sección se describe esta técnica.

1.2 Monitoreo de red.

1.2.1 Definición.

El monitoreo de red es una técnica utilizada para observar el tráfico que circula en una red de datos, siendo de gran utilidad esta técnica en las empresas, debido a que un administrador se puede dar cuenta mediante su uso sobre los servicios que no estén levantados, enlaces que estén fallando o detectar anomalías presentes en la red.

Normalmente al encontrar alguna anomalía se avisa a los administradores de red mediante alertas, como puede ser el envió de correos electrónicos o generando alarmas encargadas de notificar la anomalía que ha ocurrido en la red.

Existen programas dedicados exclusivamente al monitoreo de la red; como puede ser Nagius, Netflow y HpOpenView, entre otros. Cada software genera estadísticas del tráfico que circula a través de la red y por medio de análisis realizado a los flujos obtenidos se calcula:

- Utilización de ancho de banda de interfaces de red.
- Utilización de protocolos TCP, UDP, ICMP
- Utilización y disponibilidad de servicios (HTTP, FTP, SSH, SNMP, etc.).
- HostS, redes que consumen el mayor ancho de banda.
- Detección de comportamientos anormales en la red, entre otras funcionalidades.

Existen diversos métodos para realizar monitoreo de red, como son:

- Activar la tarjeta de red en modo promiscuo.
- Utilizando el servicio SNMP
- Mediante el protocolo Netflow.

La tabla I.1 mostrada en la introducción describe brevemente estos métodos que utilizan el monitoreo de red.

En este proyecto de tesis se decidió utilizar el protocolo Netflow que ha adquirido bastante popularidad en la actualidad como una nueva técnica eficiente en el monitoreo de red, en la siguiente sección se describe profundamente esta técnica.

1.3 Netflow.

1.3.1 Definición

NetFlow es un protocolo abierto desarrollado por Darren Kerr y Barry Bruins, de CISCO Systems, que permite la recolección de tráfico de red.

La tecnología NetFlow describe la manera en la que un *router* y/o un *switch* inteligente exportan estadísticas sobre el tráfico que pasa por el mismo, mediante la generación de registros NetFlow (denominados flujos) que se exportan vía datagramas UDP a un dispositivo o máquina recolectora.

Actualmente Netflow cuenta con 10 versiones, Netflow V1-V9 e IPFIX (también conocida como V10). Tras las versión 5, la versión más utilizada en el mercado es la versión 9. Esta versión se basa en plantillas, permitiendo varios formatos para los registros NetFlow, siendo así mucho más flexible y extensible.

Aunque inicialmente el protocolo NetFlow fue implementado por CISCO, la necesidad de un protocolo estándar y universal que permita la exportación de información en flujos de red desde distintos dispositivos de red, ha hecho que NetFlow haya emergido como un estándar en la *IETF* (RFC 3954 Netflow v9).

Un punto muy importante a señalar sobre los registros NetFlow es que éstos no contienen información del usuario, sólo datos de conexión, lo que permite tener una visión detallada del comportamiento del segmento de red (tanto para el monitoreo como para el análisis efectivo de dicho tráfico), evitando problemas relacionados con la privacidad de los usuarios.

1.3.2 Flujo.

Un flujo es una cadena unidireccional de paquetes entre una fuente y un destino, los cuales están definidos por una dirección IP, puerto origen y destino respectivamente.

Cada paquete enviado se clasificara en un flujo a través de los siguientes datos

1. IP Origen
2. IP Destino
3. Puerto Origen
4. Puerto Destino
5. Tipo de Protocolo de capa 3
6. ToS (Type of service)
7. Interfaz utilizada en el dispositivo activo (router o switch)

Por medio de los siete campos contenidos en cualquier flujo se logra que cada paquete capturado sea único y la agrupación de paquetes con contenido igual en un mismo flujo [http://www.cisco.com/en/US/products/ps6601/prod_white_papers_list.html].

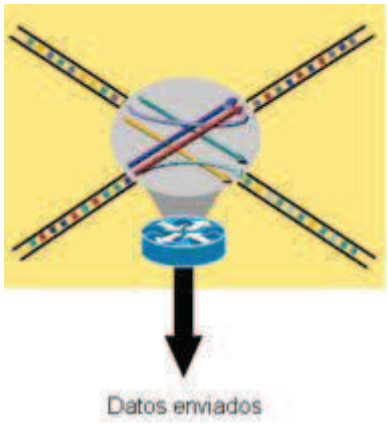

Figura 1.1 Representación de un flujo.

En la figura 1.1 se observa a un router que está clasificando la información que llega a él en distintos flujos de acuerdo con los siete campos que componen a un flujo. El dispositivo activo realizará las siguientes tareas:

- Enviar la información hacia su destino.
- Clasificar toda la información presente en flujos.
- Almacenar todos los flujos en una tabla (llamada cache).
- Enviar el contenido de la tabla hacia un dispositivo colector

1.3.2.1 Calculo del flujo

Una vez definido lo que es el flujo, los componentes presentes sobre él y los criterios utilizados para su clasificación. Para poder agrupar la información en flujos será necesario activar una cache (tabla) en el dispositivo activo. Toda cache activada sobre algún dispositivo activo, se encargará de:

- Contener toda la información que circule sobre el router en flujos
- Clasificar cada paquete presente en un flujo, tomando en cuenta los siguientes criterios:
  - Si el paquete analizado no se puede agrupar en algún flujo creado, se creara un nuevo flujo que contendrá información sobre este paquete y se añadirá a la cache.
  - Si el paquete analizado contiene información igual a algún flujo creado anteriormente. El paquete será agrupado sobre este flujo, sumando la cantidad de tráfico y paquetes del nuevo paquete al total contenido en el flujo.
- Crear un registro del total de flujos almacenados en la cache. Desglosando el tráfico presente en los protocolos TCP, UDP e ICMP y actualizar este registro cada que se añada un nuevo flujo a la cache.
- Agrupar el contenido de la cache en un paquete de exportación cada determinado tiempo.
- Enviar el paquete de exportación hacia un dispositivo colector cada determinado tiempo.

1.3.3 Funcionamiento del protocolo Netflow

El funcionamiento de esta técnica radica en la recolección e interpretación de los flujos creados en un determinado tiempo. Normalmente se realiza una comparación entre los flujos obtenidos actualmente y el promedio de los flujos almacenados en un lapso de tiempo específico. Si el resultado del análisis muestra un aumento inusual en el ancho de banda o se ha detectado un comportamiento anormal, se notifica inmediatamente sobre dicho evento a los administradores de red. Debido a la posibilidad de un ataque denegación de servicios o un malware presente en la red.

1.3.3.1 Conceptos Básicos

Los siguientes conceptos son necesarios para entender el funcionamiento del protocolo Netflow:

- **Observation Point:** Cualquier *dirección IP* que se desea observar dentro de la red.
- **Observation Domain:** Conjunto de Observation point que se encuentran dentro de un dispositivo activo en la misma red y tienen habilitado la exportación de datos vía Netflow.
- **IP Flow or flow:** Es el conjunto de paquetes que pasan a través del Observation Point en un intervalo de tiempo. También conocidos como flujos.
- **Flow record:** Provee información acerca de los flujos que se están observando en el Observation Domain.
- **Exportador:** Dispositivo activo (generalmente router) que tiene el servicio de Netflow habilitado; y que por medio de una *memoria volátil* dedicada almacenará, agrupará los flujos y creará el export packet que contendrá la cantidad de paquetes obtenidos de Observation Domain.
- **Export Packet:** Paquete originado en el dispositivo exportador con el objetivo de transportar los flow records generados por este dispositivo hacia el colector en un tiempo específico.
- **Colector:** Dispositivo que estará escuchando sobre un puerto UDP determinado con el objetivo de obtener los export packet enviados hacia él de uno o varios exportadores, este dispositivo almacenara la información en algún medio (Bases de datos, archivos, etc.) y tendrá la información lista para su interpretación mediante un software licenciado o de uso libre que soporte el protocolo Netflow.
- **Packet Header:** Es el primer campo que contiene un export packet. Contiene información acerca de ese paquete en específico. Ejemplo: la versión de Netflow utilizada.
- **Template record:** Define la estructura e interpretación de los campos en un flow data record.
- **Flow data record:** Son datos que contienen valores de los flujos (como puede ser dirección IP, puerto, protocolo, etc.) y que están asociados con el template record.
- **Options Template Record:** Define la estructura e interpretación de los campos de un Options Data Record.
- **Options Data Record:** Son datos que contienen valores e información sobre los parámetros de medición de los flujos, correspondientes a un Options Template Records.
- **Flowset:** Son flow record que tienen una estructura similar. En un export packet, uno o más flowset siguen el packet header. Se suelen dividir en:

- **Template Flowset:** Son uno o más templade records que han sido agrupados en un Export packet.
- **Options Template Flowset:** Son uno o más option templade records que han sido agrupados en un export packet.
- **Data Flowset:** Son uno o más flow records de un mismo tipo agrupados en un export packet. Cada registro es también un Flow Data Record o un Options Data Record definido por un Templade Record o un Options Template Record.

Para entender mejor acerca del formato de los datos presentados por Netflow la figura 2 muestra un ejemplo de un export packet generado para la versión 9 [http://www.cisco.com/en/US/products/ps6601/prod_white_papers_list.html].

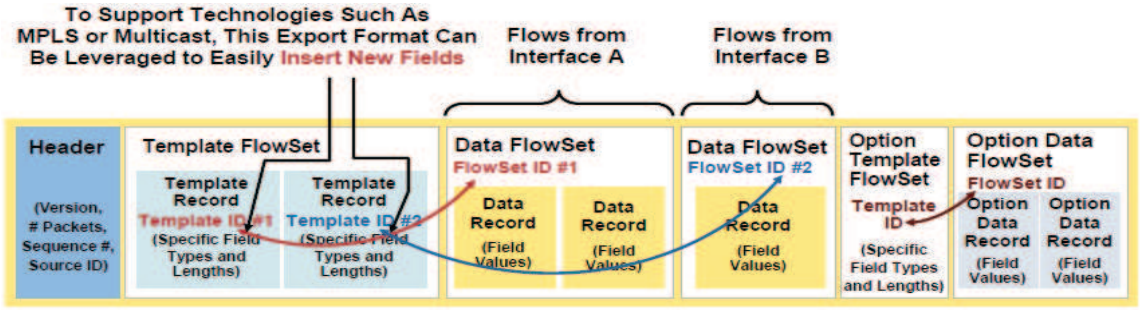

Figura 1.2 Formato Export Packet V9

Como se observa en esta figura, se tiene al inicio al packet header (Header) que indica: la versión utilizada, número de paquetes, número de templates ocupados, un número de identificación sobre el export packet (ID). Después del packet header se tendrán campos agrupados con una estructura similar (FlowSet). Los FlowSet se dividen en los mencionados anteriormente.

En el ejemplo mostrado se observa que el primer campo después del packet header corresponde al Templade Flowset. Este campo contiene información general acerca de los templates generados. A cada template se le asignará un número ID único. En el ejemplo se observa que cada template record creado corresponde a información obtenida de diferentes interfaces en el router.

En los siguientes dos campos se observan todos los datos obtenidos del template ID 1 (interfaz A) y el *templade* ID 2 (Interfaz B) respectivamente. Por último se contiene el campo Option template records que puede contener información acerca del tráfico consumido de los templates anteriores u otra información agrupada en un nuevo template.

Para entender el funcionamiento del protocolo Netflow, se describirán los tres componentes principales utilizados.

1.3.3.2 Componentes básicos en el protocolo netflow.

Se distinguen tres componentes básicos en el protocolo Netflow. Estos son:

Exportador. Router o switch capaz de generar registros NetFlow (Export packets), que se exportarán a un colector vía UDP.

Colector. Dispositivo que escucha en un puerto UDP determinado y que es capaz de almacenar o reenviar los flujos recibidos a otros colectores según la arquitectura definida.

Analizador. Software encargado de filtrar, mostrar, analizar y/o visualizar los flujos recibidos.

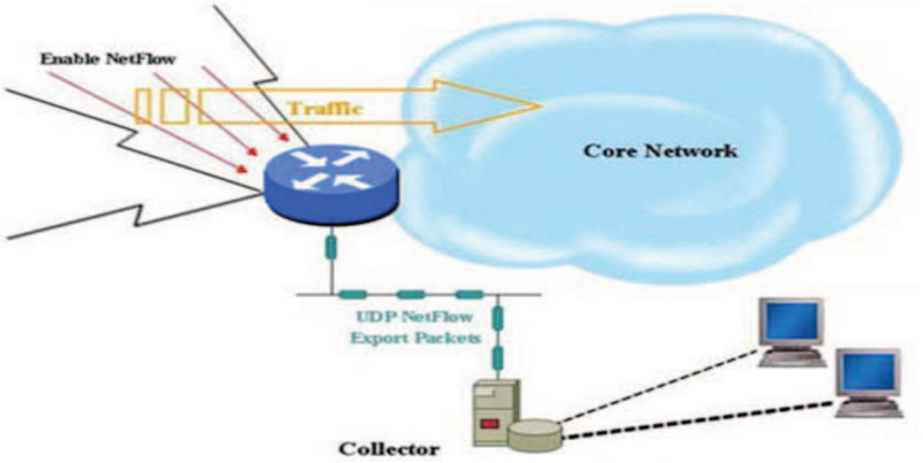

Figura 1.3 Funcionamiento protocolo Netflow

La figura 1.3 [http://www.cisco.com/en/US/products/ps6601/prod_white_papers_list.html] muestra el funcionamiento del protocolo Netflow de la siguiente forma:

- El tráfico presente sobre un router, que soporte el protocolo Netflow, será capturado, interpretado y clasificado en diferentes flujos. Dicho router se encargará de generar Export packets cada determinado tiempo (5 minutos por default).
- Los export packets se dirigirán hacia un equipo colector que se encargará de almacenar todos los datos provenientes de los exportadores.
- Con ayuda de un analizador podremos interpretar los archivos capturados y someterlos a un análisis detallado con ayuda de algún software licenciado o de uso libre.

Para entender mejor la forma en la que Netflow genera los Export Packets, es necesario explicar el concepto de Netflow cache.

1.3.3.3 Memoria volátil dedicada Netflow (cache).

Los exportadores operan construyendo una memoria volátil dedicada (Netflow cache) que contiene información sobre todos los flujos activos que pasan por el dispositivo. Cada flujo está representado por un flow record que contendrá los siete campos ocupados en la clasificación de un flujo, además de información extra relacionada con la conexión. La *caché* se actualiza cada vez

que se obtiene un nuevo flujo, llevando una cuenta de los paquetes y bytes por flujo. Después de un lapso de tiempo (definido por el administrador) se creará el *export packet* que contendrá todos los datos almacenados en la netflow cache. Al momento de ser enviado el export packet, la cache solo contendrá aquellos flow record que no hayan expirado.

Un flow record expira de la caché según una serie de criterios, algunos de ellos configurables en el dispositivo.

Estos criterios son:

- Cuando las conexiones TCP llegan a su fin (Flag FIN) o son reseteadas (se recibe un flag RST).
- Los flujos han estado inactivos por un tiempo determinado (Normalmente 15 segundos).
- La caché se llena o el router se queda sin recursos.
- Los flujos se mantienen activos en caché por un tiempo determinado (Normalmente 30 minutos). Una vez pasado este tiempo expiran de la caché, asegurando un reporte periódico. El envío se hace más frecuente si aumenta el tráfico de las interfaces configuradas con NetFlow.

Todos los flow record que han expirado en la caché se adjuntan en un datagrama de exportación UDP (export packet), que típicamente contendrá entre 20 y 50 registros. Este datagrama se envía a un puerto determinado al dispositivo colector configurado.

Para la colección y análisis de los flujos recibidos por los dispositivos de red se pueden utilizar diversos productos disponibles en el mercado tanto de libre distribución o comerciales.

En la figura 1.4 se muestra un ejemplo del contenido de una Netflow cache construyéndose en un router.

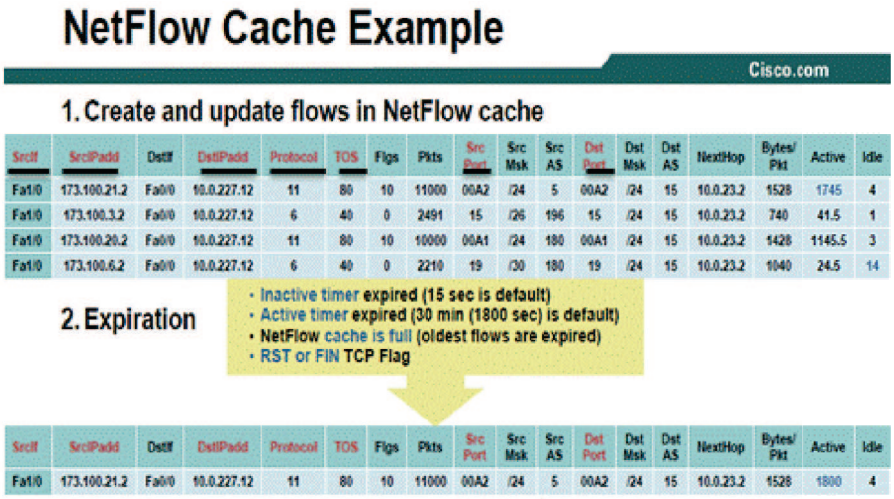

Figura 1.4 Netflow cache

Como se observa en la figura 1.4 se tienen cuatro flujos activos, definidos por los campos subrrayados (criterios para la clasificación de cada flujo); cualquier flujo nuevo que contenga los

mismos datos de clasificación a alguno almacenado en caché, se añadirá a ese flujo sumando la cantidad de paquetes y tráfico consumido del nuevo flujo a la información ya almacenada. En caso de que el nuevo flujo contenga datos diferentes a los presentes, se creará una nueva entrada en la caché conteniendo la información de este nuevo flujo.

Según los criterios mencionados en la expiración del flow record, se observa que el primer flujo ha estado activo por un tiempo mayor a 1800 segundos. Por lo cual expira de la caché y se unirá al export packet a enviar al dispositivo colector.

Además se observa que en la caché se guardan solo datos de la conexión. No se observa ningún dato que contenga información referente al usuario. Lo que proporciona una gran ventaja en cuanto a la privacidad de la información.

1.3.4 Versiones de Netflow

Actualmente el protocolo Netflow cuenta con 10 versiones: La versión estándar es la 5; las versiones 1 a 4 son propietarias de cisco. La versión 9 es la que ofrece mejor compatibilidad con varios dispositivos mediante el uso de templates. Permitiendo la transmisión en diferentes formatos de datos Netflow en un mismo *export packet*, brindando con esto mayor flexibilidad.

La tabla 1.2 muestra un breve comentario sobre las versiones de Netflow más utilizadas.

Tabla 1.2 Principales características de las versiones de NetFlow

Versiones de Netflow	Comentario
1	Versión original, solamente soportada por router cisco.
5	Estandarizada y es la más usada en la actualidad, solo soporta Ipv4.
7	Solo es utilizada en las series de switch cisco C6500 y 7600
8	Múltiple compatibilidad entre swich, permite diferentes esquemas
9	Uso de templates. Al igual que la V8 tiene múltiple compatibilidad, además de esto es más flexible que las versiones anteriores, debido a que tiene soporte para campos adicionales y tecnologías como ejemplo: ➢ MLPS, BGP, Ipv6, entre otros.

Las versiones más utilizadas por el protocolo Netflow son la versión 5 y la versión 9. Por este motivo se realiza una comparación detallada entre cada una de ellas.

1.3.4.1 Netflow V5 vs V9

Si bien es cierto que la versión cinco de Netflow es la versión estándar, la versión nueve provee grandes mejoras en la representación de los datos en formato Netflow, la mejora más significativa entre estas versiones fue el agregar templates a la V9 de Netflow, permitiendo con esto mayor flexibilidad en los datos enviados.

La figura 1.5 [http://www.plixer.com/blog/netflow/netflow-v9-vs-netflow-v5/] muestra una comparación sobre el formato de un *export packet* entre la versión 5 y la versión 9.

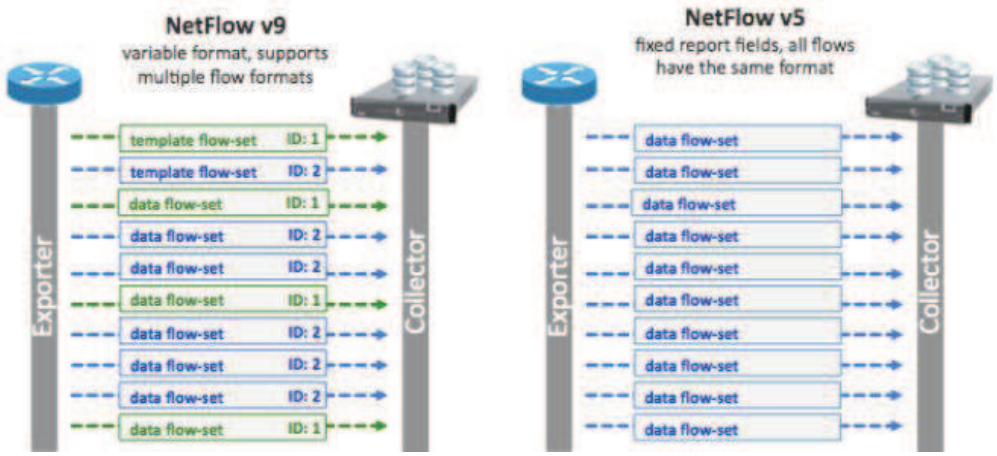

Figura 1.5 Netflow v5 Vs Netflow v9

El principal cambio entre estas versiones es el uso de templates en la versión 9. Logrando con esto el transmitir varios datos en distinto formato sin la necesidad de crear otro *export packet* para su transmisión.

Otra principal diferencia que se tiene entre la versión 5 (V5) y la versión 9 (V9) es que la V5 tiene un formato fijo de datos. Al utilizar la V5 solo podrán enviar datos hacia los routers que cumplan con el formato establecido por la V5. El export packet recibido no podrá ser interpretado por colectores que no soporten la V5. Mientras que la V9 con el uso de templates permite la combinación de varios campos en un mismo *export packet*. Logrando una configuración flexible dependiendo de la opción a realizar y que múltiples dispositivos puedan utilizar la V9 de Netflow adaptándola a sus necesidades.

Cada template creado contara con un único número que lo identificara (ID). Dependiendo de los datos recibidos se agruparan en el ID que les corresponda, logrando con esto tener múltiples datos con formatos distintos en un mismo export packet; pero cada dato o conjunto de datos solo pertenecerá al ID que hará referencia a su template.

1.3.5 Ventajas y consideraciones de utilizar Netflow

Netflow nos proporciona bastantes beneficios. Entre los cuales destacan:

- Responde a las preguntas qué, quién, cómo, dónde y cuándo acerca del tráfico cursado en la red.
- Provee estadísticas acerca de redes más utilizadas.
- Provee una visión detallada acerca del comportamiento de la red.
- Estandarizado**:** RFC 3954 Netflow V9.
- Monitoreo de la Red**:** Con técnicas de análisis de flujo.
- Monitoreo de Aplicaciones (basado en puertos TCP/UDP): Para planificar, entender nuevos servicios, y distribuir recursos y aplicaciones en la red.
- Monitoreo de Usuarios: para revisar de forma efectiva la utilización de los recursos por parte de los usuarios.

- Planificación de la Red: para anticiparse a los crecimientos de la red, ya sea en dispositivos, puertos y ancho de banda
- Análisis de seguridad: con el fin de detectar anomalías en el tráfico de la red.
- Contabilidad y la Facturación: Netflow es la principal tecnología desarrollada en estas áreas, debido a sus detalladas estadísticas. Permite poner precio al BW consumido.
- Almacenamiento de los Datos Netflow: Permite guardar estadísticas de los flujos capturados para realizar futuros análisis.
- No provee información sobre los usuarios: Solo nos muestra información sobre la conexión.
- Agrupar los datos: Mejor organización de los datos.
- Dispositivos Activos: Fue creado inicialmente para router y switch cisco, pero cada vez más empresas desarrolladoras de dispositivos activos incluyen este protocolo en sus equipos.
- Alcance: Debido a que NetFlow puede ser configurado en la mayoría de los routers y switch. Se tiene una excelente visión sobre el tráfico presente en la red.
- Entre otros.

1.3.6 Aplicación de la tecnología NetFlow en el área de seguridad informática.

El uso de NetFlow ha demostrado de gran utilidad para múltiples fines relacionados con el monitoreo, contabilidad y cobro de transmisión del tráfico o uso de red (del inglés “measurement, accounting and billing”).

Algunas funciones que provee Netflow orientado a seguridad informática son las siguientes:

- Detección en redes de la utilización de puertos TCP/UDP específicos.
- Propagación del malware en la red
- Detección de tráfico relacionado con un incidente ocurrido.
- Investigación de ataques DoS (Denial of Service).
- Host/subredes que consumen mayor tráfico en la red.

La principal ventaja de Netflow contra otras técnicas de monitoreo de red enfocadas hacia seguridad se presenta en omitir información acerca del usuario. Proporcionando la información de forma sencilla y fácil de comprender.

Netflow proporciona una serie de ventajas frente al uso de IDS y otros mecanismos de detección perimetrales, entre las cuales se pueden destacar:

- Dado que la mayoría de dispositivos de red tienen la capacidad de exportar registros NetFlow, el monitoreo se puede realizar desde cualquier router en la infraestructura. Incluidos routers de acceso a Internet donde normalmente se ubican IDSs y Firewalls, teniendo así además una vista única del tráfico total de la red a nivel de infraestructura.
- A diferencia de los IDSs, con la utilización de Netflow no se tiene acceso a información confidencial; debido a que los flujos no contienen información del usuario, sólo datos de conexión. Esta es una de las mayores ventajas de esta tecnología.
- La detección de ataques de día cero”zero-day” o mutaciones de ataques por medio de un análisis realizado hacia los registros obtenidos por Netflow. Especialmente útil cuando la detección basada en firmas no es válida.

1.3.6.1 NetFlow enfocado en la detección de ataques y anomalías

NetFlow permite detectar, en tiempo real, ataques o anomalías presentados en la red. Como puede ser:

- Equipos posiblemente comprometidos (presenten alguna infección).
- Conexión hacia servidores clientes desconocidos.
- Envío de información a keyloggers.
- Ataques DoS/DDoS.
- Escaneos de puertos y redes.
- Infecciones específicas de determinados gusanos.
- SPAM, entre otros.

Existen varios métodos de análisis de flujos para detección de ataques y anomalías de seguridad. Como puede ser:

Realizando un análisis básico del tráfico. Se trata de un modelo basado en la descripción de las actividades consideradas como "normales" en la red de acuerdo a patrones históricos de tráfico. De manera que cualquier otro tipo de tráfico se marca como malicioso. La forma más básica de realizar esta tarea es mediante el uso de estadísticas, informes de datos y sesiones (estadísticas *TopN*).

Basado en expresiones regulares. Con el objetivo de detectar patrones de comportamiento anormal, cualquier campo de los incluidos en los registros NetFlow es susceptible de ser utilizado en una búsqueda por medio de expresiones regulares.

Por medio de alertas. La mayoría del software incorpora la opción de alertas. Esta opción nos permite realizar un análisis del comportamiento normal presente en la red. Dicho comportamiento sirve como línea base para realizar comparativos posteriores. Cuando se presenta un comportamiento anormal se ejecutará una alerta notificando un aumento en el tráfico o aumento en puertos específicos.

Con ayuda de filtros aplicados hacia los registros Netflow se logra acotar la información y poder realizar análisis más robustos.

Por medio de algoritmos. Se pueden realizar programas enfocados hacia la detección de actividades anormales presentadas en el tráfico capturado por Netflow. Como puede ser:

a. Escaneo de puertos o direcciones IP.
b. Ataques denegación de servicio.
c. Envío de información hacia servidores externos
d. Aumento del tráfico en la red, entre otros.

Una vez que se ha descrito lo que es el software libre, el monitoreo de red y el funcionamiento del protocolo Netflow, conceptos necesarios para el desarrollo de este proyecto de tesis. Se

explicarán brevemente conceptos requeridos de seguridad informática, necesarios para la correcta creación del detector de malware.

1.4 Conceptos de Seguridad Informática.

Lo que se pretende en esta sección es dar una visión general de lo que es la seguridad informática, los criterios usados para la clasificación de ataques y las buenas prácticas. Tomando como base estos criterios para la estrategia implementada en la detecciónel malware mediante el monitoreo de red.

1.4.1 Conceptos Básicos de Seguridad.

En esta sección se explicara a grandes rasgos diversos conceptos esenciales de seguridad informática

Seguridad: Diversas acciones y herramientas que me permiten proteger mis bienes o activos importantes para una persona o grupos de personas.

Tecnologías de la información: Son todos aquellos dispositivos o medios electrónicos que me permiten manipular la información.

Seguridad Informática: Diversas acciones y herramientas que me permiten proteger toda la tecnología de la información.

Servicios de seguridad.
Un servicio de seguridad es aquel que me permite mantener la seguridad en un sistema de cómputo.

Los servicios de seguridad están diseñados para contrarrestar los ataques a la seguridad y hacen uso de uno o más mecanismos de seguridad para proporcionar el servicio. Se clasifican en los siguientes.

- Control de acceso. Acceder a mi bien o activo siempre y cuando esté autorizado.
- Confidencialidad. Los usuarios deben de tener la privacidad de ver o realizar cambios en los bienes o activos.
- Integridad. Los usuarios deberán de obtener los bienes o activos tal y como ellos los modificaron (sin ninguna alteración por terceras personas).
- Disponibilidad. Todo bien o activo debe de estar presente al momento que un usuario autenticado acceda a ellos.
- No repudio. Al momento que un usuario realice cambios sobre los bienes o activos, deberá de aceptar que realizó dichas modificaciones.
- Autenticación. Serie de medios para garantizar que una persona es quien dice ser
  Factores de autenticación.
  - Algo que se sabe. Contraseña, clave.
  - Algo que se es. Administrador, jefe.
  - Algo que se posee. Certificado de seguridad, permisos.
  - Algo que se hace. Firma autógrafa, decir alguna frase
  - Desde algún lugar. Terminal remota, Lugar identificado por IP, *VPN(Virtual Private Network)*

Mecanismos de seguridad.
Son las herramientas o controles que permiten implementar un servicio de seguridad. Los mecanismos de seguridad pueden ser: aplicaciones (lógicos), físicos, buenas prácticas (reglas morales), reglas (políticas de seguridad), estándares, etc.

Las herramientas pueden ser: Preventivas, correctivas o detectoras.

1.4.2 Deficiniones de vulnerabilidades, amenazas y ataques

1.4.2.1 Vulnerabilidad.

Deficiencia o punto(s) débil(es) encontrado(s) que puede(n) ser explotado(s) por perpetradores con el objetivo de comprometer o dañar nuestros activos y bienes.

1.4.2.2 Amenaza.

Circunstancia o evento que aprovecha una vulnerabilidad y compromete la integridad, disponibilidad y confidencialidad. Generalmente pretende, puede o intenta destruir algo o a alguien. Siempre está latente, es importante señalar que puede o no materializarse.

Tanto las amenazas como las vulnerabilidades se clasifican en físicas, humanas, desastres naturales, hardware, software y de red.

1.4.2.3 Ataque.

Es una consumación de una amenaza, generalmente los ataques explotan una o varias vulnerabilidades y el grado de peligrosidad dependerá del perpetrador que los ha creado.

Los ataques responden al siguiente esquema: *Vulnerabilidad + Amenaza = Ataque*.

Todo ataque antes que el perpetrador lo ejecute sobre su objetivo específico, deberá de cumplir con las siguientes etapas:

1. *Planeación:* El perpetrador en esta etapa se encargará de recolectar información del objetivo(s) a atacar por cualquier medio posible. Además de visualizar el o los objetivos que pretende cuando se ejecute el ataque.
2. *Activación:* En esta etapa el ataque se encuentra en plena ejecución. Generalmente ya habrá explotado la vulnerabilidad(es) encontrada(s) resultado del análisis del punto anterior
3. *Ejecución:* Son los logros obtenidos por el perpetrador una vez que el ataque se encuentra en ejecución.

1.4.2.3.1 Clasificación de los ataques

Los ataques se suelen clasificar de acuerdo con los siguientes criterios:

- ¿Qué tan Intrusivo es? En este punto se clasifica al ataque de acuerdo con la capacidad de esconderse ante posibles detecciones de herramientas de seguridad o usuarios:
    A. *Ataque pasivo:* Son aquellos ataques que tienen el objetivo de pasar desapercibidos por las herramientas de seguridad y los usuarios.

B. *Ataque activo:* Son aquellos ataques que tienen el objetivo de causar alguna modificación evidente sobre los datos, bienes o activos.

- ¿A qué servicio de seguridad ataca? Se clasifica al ataque de acuerdo con el servicio de seguridad que pretende comprometer
  - A *Modificación:* Ataque realizado con el objetivo de crear algún cambio o alterar el flujo normal de comunicación. Atenta contra la integridad. Ejemplos: Backdoor, obtención de passwords, Arp Spoofing, IP Spoofing, Fuerza bruta, exploid, entre otros.
  - B *Suplantación:* Ataque realizado con el objetivo de obtener acceso hacia el flujo de comunicación robando la identidad de un usuario autorizado. Atenta contra la autenticación. Ejemplos: Ciber Graffiti, SQL Injection, Borrado de huellas, entre otros.
  - C *Interrupción:* Ataque que tiene como objetivo impedir el flujo normal en la comunicación. Atenta contra la disponibilidad. Ejemplos: DoS (Ataque de negación de servicios), DDoS (Ataque de negación de servicios distribuido), entre otros.
  - D *Intercepción:* Ataque que tiene como objetivo observar las acciones realizadas en el flujo de comunicación. Atenta contra la confidencialidad. Ejemplos: Sniffers, Keylogger, entre otros.

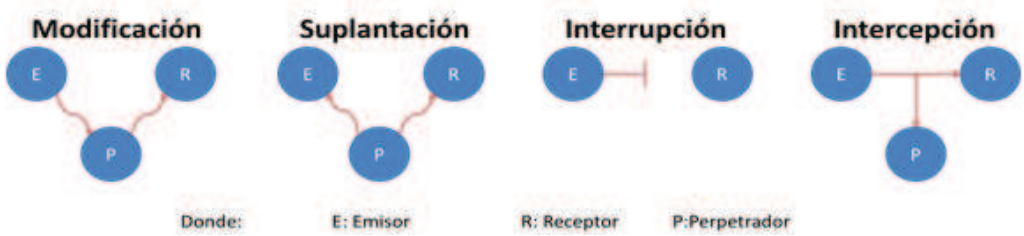

Figura 1.6 Esquema de ataques Fuente: Apuntes clase seguridad informática I, profesor: M.C. Cintia Quezada Reyes

¿Dónde se realiza? Se clasifica desde un punto de vista geográfico (lugar de ejecución).

- *Ataques Internos:* Ataque realizado por un usuario perteneciente a la empresa, institución, organización
- *Ataques externos:* Ataque realizado por un usuario externo a la empresa, institución, organización.

En la última sección de este capítulo se explicará a detalle el software malicioso, o también conocido como malware.

1.5 Malware.

1.5.1 Definición.

El malware es cualquier tipo de software dañino que puede afectar a un equipo electrónico de varias formas, como puede ser:

- Actuando como software espía.
- Obteniendo el password del equipo “víctima”.
- Realizando ataques pasivos o activos hacia los equipos “víctima”, con el fin de obtener información, causar daños o molestar.
- Engañando a la “víctima” mediante páginas web falsas, Ingeniería social, etc.
- Recolectar información del equipo “víctima”.

El Malware puede ser propagado por cualquier medio de comunicación disponible (Internet, Correo, P2P, Mensajería, etc.) y por supuesto por cualquier sitio web.

1.5.2 Clasificación del Malware

El malware se suele clasificar de acuerdo con la acción que realiza, su comportamiento o el grado de peligrosidad que suele tener en los equipos que logra infectar. El malware más común son:

1.5.2.1 Virus.

Programas creados para infectar sistemas u otros programas. Una vez que el virus logra infectar se encargara de realizar modificaciones y daños con el objetivo de provocar un mal funcionamiento general del equipo, registrar, dañar o eliminar datos, o bien propagarse por otros equipos a través de Internet.

1.5.2.2 Backdoor.

Son programas diseñados para abrir una "puerta trasera" en la víctima y permitirle a otro malware tener acceso.

1.5.2.3 Bootnets (Redes zombies).

Son computadoras infectadas por algún malware que actúan en conjunto enviando peticiones hacia servidores con el objetivo de saturarlo y por consecuente afectar la disponibilidad del sistema y/o aplicativo.

1.5.2.4 Exploit.

Programa o código que "explota" alguna vulnerabilidad del sistema o de parte de él, aprovechando esta deficiencia para beneficios propios.

Generalmente los exploit al encontrar alguna vulnerabilidad en el sistema accederán a ella y le permitirá el acceso a otro tipo de malware.

1.5.2.5 “Zero day” (Ataques de día cero).

Es cualquier tipo de malware del que no se tiene conocimiento. Generalmente cuando un exploit encuentra una nueva vulnerabilidad se encarga de ejecutar acciones para permitir a él mismo o a otro tipo de malware entrar y realizar ataques sobre el sistema.

Como es la primera vez que se ejecuta el ataque no se tendrá ningún parche, se categoriza como día cero al primer ataque realizado. Esta clasificación terminará cuando se tenga algún mecanismo de seguridad que contrarreste a este malware.

1.5.2.6 Gusanos (Worm)

Son programas desarrollados para reproducirse por algún medio de comunicación como el correo electrónico, redes P2P, memorias USB, internet, etc.

Su principal objetivo es llegar a la mayor cantidad de usuarios posible y lograr distribuir otros tipos de de malware (como Troyanos, Backdoors y Keyloggers, etc.).

Generalmente los gusanos están orientados a hacer ataques DoS contra sitios webs específicos y tiene como principal característica el saturar la red.

1.5.2.7 Hoax

Son mensajes enviados principalmente por correo electrónico que contienen contenido falso o tratan de engañan a sus víctimas por diferentes formas, como puede ser:

- Creación de nuevos virus y traen un adjunto con el parche que generalmente es un software espía o un virus.
- Mensajes de personas enfermas e incitan a ayudarlas por distintos métodos.
- Cadenas (SPAM).

Generalmente estos mensajes tienen el objetivo de saturar la red, los servidores de correo y obtener direcciones de correo.

1.5.2.8 Keylogger

Los keylogger son programas que capturan todo lo que teclea la máquina infectada e inclusive algunas aplicaciones capturan los clics efectuados con el mouse. Este tipo de malware envía las capturas a sus creadores. Por medio de un análisis obtienen información confidencial como nombres de usuario, contraseñas, números de cuentas, etc.

1.5.2.9 Phishing

Son páginas falsas creadas. Generalmente copias idénticas de las páginas de bancos o empresas importantes con la finalidad de engañar al usuario haciéndole creer que se encuentra en la página oficial. Al engañar al usuario el malware logra obtener información confidencial.

Generalmente los phishing llegan a las víctimas por medio de correos electrónicos haciéndose pasar por un correo de algún sitio oficial y al dar clic en sus ligas reenviará al usuario a la página web falsificada.

1.5.2.10 Spam

Son mensajes enviados a destinatarios sin su consentimiento, generalmente son mensajes publicitarios o contienen información de páginas falsas (phishing). Son enviados en forma masiva y tienen como objetivo el ser vistos por el usuario o saturar el servidor de correo utilizado.

1.5.2.11 Spyware

También conocidos como software espías. El objetivo de este malware es recolectar información de alguna persona o institución sin su consentimiento y distribuirlo a personas interesadas en esta información obtenida ilegalmente.

Generalmente recolectan información como nombres de usuario, contraseñas, direcciones IP, páginas web visitadas, etc.

1.5.2.12 Troyanos

Son virus informáticos o algún programa malicioso capaz de alojarse en computadoras y permitir el acceso a usuarios externos, a través de una red local o de Internet, con el fin de recabar información.

Suele ser un programa pequeño alojado dentro de una aplicación, una imagen, un archivo de música u otro elemento de apariencia inocente, que se instala en el sistema al ejecutar el archivo que lo contiene.

Generalmente cuando un troyano es ejecutado o instalado realiza acciones imperceptibles para el usuario y el sistema operativo, pueden actuar como espías o inclusive modificar registros del sistema logrando con esto tener un control total en sus acciones efectuadas y evitar ser detectados.

Como se observa, existe una enorme clasificación de malware. Por este motivo es de esperarse que cada malware tenga un comportamiento diferente, dependiendo del tipo de malware y el ataque que pretenda realizar.

1.5.3 Comportamiento del Malware

Debido a la extensa clasificación del malware no es posible definir un comportamiento general que se presente al momento de infectar a una víctima. A continuación se muestran algunos ejemplos del comportamiento del malware y las diferentes acciones realizadas.

1. Algún malware puede infectar a un equipo explotando alguna vulnerabilidad. Una vez infectado el equipo, el malware tratará de propagarse hacia otros equipos y realizar un ataque de negación de servicios.
2. Para la ejecución de algún malware será necesario que el usuario ejecute la aplicación que lo contiene. Una vez activado el malware recolectará información y la enviará hacia IP's desconocidas por el usuario.
3. Algún malware aprovechará alguna vulnerabilidad encontrada para su instalación en el sistema. Una vez instalado, el malware tratará de pasar desapercibido y abrirá una *backdoor* para la ejecución de otro malware.
4. El malware ha detectado alguna vulnerabilidad de día cero, se instalará y realizará acciones como el borrado de datos, cambio de registros, robo de contraseñas etc.
5. Se ha recibido un correo electrónico sospechoso. Algún usuario accede al contenido del correo y el malware se ejecuta. El malware tratará de consumir todo el ancho de banda

disponible en la red e infectar a la mayor cantidad de equipos en el menor tiempo posible. Entre otros.

Como se observa en los ejemplos mostrados, el comportamiento del malware varía dependiendo de la acción que se desee realizar con él. Sin embargo, se ha detectado que la mayoría del malware presenta algunos patrones similares. Normalmente un equipo infectado con algún tipo de malware tratará de realizar lo siguiente.

- Pasar desapercibido: Es de vital importancia para la mayoría del malware el pasar desapercibido. Debido a que si las herramientas de seguridad o el usuario no saben que su equipo se encuentra infectado no realizaran alguna acción para poder eliminar al malware.
- Tratar de infectar a otros equipos. Una vez que el malware ha logrado infectar a una víctima, tratará de explotar la vulnerabilidad encontrada en este equipo sobre otras víctimas pertenecientes a la misma red. Por medio de un escaneo de puertos o escaneo de IPs tratará de infectar a la mayor cantidad de equipos en el menor tiempo posible.
- Consumir recursos. Si el objetivo del malware es el de realizar algún ataque denegación de servicios, tratará de infectar a la mayor cantidad de computadoras posibles y utilizar todo el ancho de banda asignado en ellas para enviar múltiples peticiones hacia la víctima con el objetivo de saturar y tirar la conexión.
- Envió de información hacia IPs ajenas a la red infectada. El malware que presente este comportamiento recolectará información confidencial del equipo infectado y la enviará hacia IPs desconocidas para el usuario.

Estos son los principales comportamientos realizados por el malware, en esta sección me limite a describirlos brevemente. En la sección 4.3 se describe a detalle el funcionamiento de cada técnica descrita. Normalmente un equipo infectado por algún malware suele presentar las siguientes características.

- Se consume demasiado ancho de banda sin razón aparente.
- El equipo trabaja lento.
- Borrado de archivos sin autorización.
- Instalación de software sin autorización.
- Cambio en el registro del sistema; creación de cuentas de usuario sin autorización.
- No poder ejecutar herramientas de seguridad.
- Pérdida de archivos del sistema.
- Reinicio del S.O o imposibilidad de acceder a él.
- Software instalado deja de funcionar sin razón aparente.
- Aparición de páginas web no solicitadas.
- Pérdida de acceso a unidades rígidas.
- Entre otros.

Si el equipo presenta uno o varios de estos síntomas es recomendable el ejecutar alguna herramienta de seguridad en busca de malware que lo haya infectado.

Como último punto en este capítulo, se incluyen algunos consejos para evitar que un equipo sea infectado por malware

1.5.4 Mecanismos de prevención.

Para evitar daños ocasionados por algún malware. Es aconsejable seguir las siguientes reglas como medidas de prevención.

- Tener actualizado el S.O. y software.
- Instalar diversas herramientas de seguridad (como antivirus, antispyware, firewall, IDS, etc.).
- Realizar escaneos con herramientas de seguridad a su equipo de forma periódica (recomendable cada 2 meses).
- Evitar el uso de cuentas con privilegios de administrador.
- No abrir correos cuyo remitente sea sospechoso o desconocido.
- Evitar el uso de "cafés internet". De ser necesario su uso no teclear nombres de usuario o contraseñas en estas máquinas. Además al introducir dispositivos extraíbles en estos lugares realizar un escaneo con algún antivirus. (Más vale prevenir que lamentar).
- No proporcionar información confidencial.
- Usar contraseñas fuertes (al menos ocho dígitos incluyendo: letras, números, símbolos).
- Cambiar contraseña periódicamente (recomendable cada 6 meses).
- Tener un respaldo de la información.
- Evitar la instalación de software crackeado.
- Evitar el uso de barras en navegadores (como las barras de ask, yahoo, etc.) debido a que suelen instalar o descargar contenido malicioso sin el consentimiento del usuario.
- Buenas prácticas.

En caso de detectar algún malware que ha infectado el equipo recomiendo:

- No desesperarse.
- Investigar en internet acerca de los síntomas presentados por el equipo en busca de una solución.
- Aislar el equipo de cualquier red (evitar la propagación del malware).
- Ejecutar herramientas de seguridad en busca del código malicioso.
- Ejecutar el administrador de tareas y buscar procesos sospechosos.
- Ejecutar alguna herramienta enfocada hacia el monitoreo de red. En msdos es posible con el comando "netstat –na" obtener información acerca de las conexiones realizadas por su equipo a internet.
- Ejecutar el S.O en modo aprueba de fallos.
- De ser posible restaurar el sistema a un estado antes de la infección del malware.
- En caso de no poder acceder al S.O ejecutar un *live cd* de alguna distribución Linux y correr alguna herramienta de seguridad en busca del código malicioso.
- Respaldo de información en caso de ser necesaria la reinstalación.