UNIVERSIDAD AUTONOMA DE MADRID

ESCUELA POLITECNICA SUPERIOR

PROYECTO FIN DE CARRERA
Ingeniería de Telecomunicación

Detección forense de ataques mediante el uso de registros NetFlow

José Antonio San Román Gil

Septiembre 2015

Detección forense de ataques mediante el uso de registros NetFlow

Autor: José Antonio San Román Gil

Tutor: Jorge E. López de Vergara Méndez

High Performance Computing and Networking group

Dpto. de Tecnología Electrónica y de las Comunicaciones

Escuela Politécnica Superior

Universidad Autónoma de Madrid

Septiembre 2015

Resumen

Este proyecto está dedicado a la caracterización de escaneos de puertos en registros NetFlow muestreados. Una vez caracterizado este tipo de ataque, se pretende crear un sistema que posibilite su detección a posterior. Con ello, se pueda aplicar a un estudio longitudinal sobre una red académica. El estudio longitudinal otorga una explicación y caracterización del comportamiento de la red a medio y largo plazo, es decir, meses o incluso años. El estudio da una respuesta y ayuda al gestor de la red de una forma complementaria. Mediante este estudio realizado se puede caracterizar, a nivel de exportador de red, los orígenes y los principales destinos de los equipos asociados al exportador, ya sean destinos nacionales o internacionales, observados en los escaneos detectados Además, se ofrece una vista a nivel diario y a nivel mensual de la concentración de estos escaneos vistos por cada exportador analizado. La última característica que ofrece este estudio es una caracterización del tipo de puerto involucrado en los escaneos detectados. Por medio de estas características que ofrecemos en el estudio podremos identificar patrones de comportamiento anómalos, comportamientos irregulares e identificar individualmente a los agentes causantes de riesgo potencial. De forma más concreta, este trabajo en primer lugar ha consistido en la familiarización sobre el sistema de monitorización de la red académica española RedIRIS, que nos ha facilitado medidas de red reales durante un periodo de un año basadas en flujos. Posteriormente, se ha trabajado en la automatización del pre-tratado de estas medidas. Estos datos se han pasado por nuestro sistema, que planteamos en este proyecto, para la obtención del tráfico involucrado en escaneos de puertos. Por último lugar, todo este desarrollo ha sido aplicado en las trazas de RedIRIS y los resultados son mostrados como un caso de estudio significativo.

Palabras clave

Estudio Longitudinal, Medidas de Red, Flujos de Red, RedIRIS, Escaneo de puertos, Sistema de detección, Netflow, Localización IP, Seguridad red, AWK, Ataques en red.

Abstract

This Project is devoted to the characterization of Port Scans in Netflow sampled records. Once characterized this type of attack, we have created a system that will later enable their detection. Thus, it can be applied to a longitudinal study of an academic network. The longitudinal study provides an explanation and characterization of the network behavior in the medium and long term, i.e., months or even years. The study gives an answer and helps network managers in a complementary way. By this study it can be characterized, in terms of net exporter, the origins and the main destinations of the equipment associated with the exporter, whether national or international destinations, observed in the detected scans. Furthermore, it offers a daily and monthly view of these scans seen by each analyzed exporter. The last feature offered by this study is to characterize the type of port involved in the discovered scans. By means of the features we offer in this study, we can identify anomalous behavior patterns, irregular conduct and individually identify potential risk-causing agents. More specifically, this work has first consisted on getting familiarized with the monitoring system of the Spanish RedIRIS academic network, which has given us real network measurements over a period of one year based on flows. Subsequently, the work has consisted on the automation of these pretreated measurements. The data has passed through our system, which we propose in this thesis, to identify the traffic involved in portscans. Finally, all this development has been applied to the RedIRIS traces and the results are shown as a significant case of study

Keywords

Longitudinal study, Network measurements, Network flows, Portscan, Detection systems, Netflow, IP Location, Network security, AWK, Network attacks.

Agradecimientos

Quiero empezar esta sección agradeciendo, en primer lugar, a Jorge E. López de Vergara Méndez, tutor de este trabajo. Su tiempo empleado, las largas reuniones y el sin fin de correos intercambiados durante la duración de este Proyecto Final de Carrera. Destaco la capacidad organizativa y la exigencia que demanda, necesaria para tener su aprobación. También agradezco a Jorge la gran libertad en la toma de decisiones de este Proyecto, así como todas las facilidades que se me han proporcionado para su desarrollo.

Mis padres y mi hermana. Ya iba siendo hora de finalizar y sé que tenéis tantas o más ganas que yo de ello. Por fin este objetivo ha sido logrado. Gracias por hacer posible que yo pueda tener estos estudios y también gracias por el apoyo tenido en los momentos más difíciles.

Mis compañeros de la carrera: Guly, Jorgito, Viejo, Xust, Escat, Diego, Alber Rastas, Isma, Charlie, Chiqui, Danielov, Mancebo, Guille, Will, Mochi, Tipi, Oscar Pedrezuela… También tengo que hacer mención al grupo de las Escaleritas, que lo conforma la nueva generación de la EPS. Yo he trabajado pero vosotros habéis ayudado y desde aquí es el mejor lugar para agradecéroslo, gracias por todo vuestro apoyo.

También debo agradecer a los Titos (Checkers, Early, m0mo, GnomoManolo, Albeniz, Shoer, Fonso, hayber, Kine, Iñaki…) amigos, compañeros y grandes personas, su apoyo en los momentos más duros de este trabajo.

A su vez, quiero acordarme de aquella persona que fue un antes y un después en mi futuro en la carrera. Cuando más negro veía la salida de la facultad, tú siempre eras un apoyo tanto en esos momentos como sigues siendo en mí día a día, gracias.

Son muchas las personas que se han interesado por el estado de este Proyecto durante su duración, a todos ellos debo agradecer su interés y este es el mejor sitio para ello.

Índice de contenidos

Índice de ilustraciones

Índice de tablas

Glosario

SNMP	Simple Network Management Protocol
AWK	Lenguaje de programación
GNU	GNU is Not Unix
MRTG	Multi Router Traffic Grapher
IP	Internet Protocol
TCP	Transmission Control Protocol
IPFIX	Internet Protocol Flow Information Export
BGP	Border Gateway Protocol
MPLS	Multiprotocol Label Switching
IETF	Internet Engineering Task Force
FIN	Fin
RST	Reset
CPU	Central Processing Unit
ISP	Internet Service Provider
MB	MegaByte
TB	TeraByte
IDS	Intrusion Detection System
DMZ	Demilitarized Zone
UDP	User Datagram Protocol
ICMP	Internet Control Message Protocol
SYN	Bit de control SYN
ACK	ACKnowledgement
OSI	Open Systems Interconnection
SDH	Synchronous Digital Hierarchy
HPCN	High Performance Computing and Networking
EPS	Escuela Politécnica Superior
GMT	Greenwich Mean Time
ASN	Autonomous System Numbers
FS	Field Separator
IANA	Internet Assigned Numbers Authority
ESP	España
EXT	Extranjero
HTTP	Hypertext Transfer Protocol
POP3	Post Office Protocol
SMTP	Simple Mail Transfer Protocol

Capítulo 1: Introducción

1.1. Motivación

Internet es un sistema complejo en constante evolución. Actualmente, una mayoría importante de los sistemas de red son víctimas de ataques por parte de hackers. El impacto económico que provocan dichos ataques nos hace concluir que es importante detectar y tomar, si es posible, medidas para su detección. Particularmente, es un tema muy complicado debido al tipo de acciones que se realizan en contra de la voluntad del usuario (robo de contraseña, recopilación de información, virus, troyanos, denegación de servicio DoS…).

Para la detección de este tipo de ataques a la red, se han desarrollado un tipo de sistemas denominados sistemas de detección de intrusiones en redes (NIDS, *Network Intrusion Detection Systems*) [1] [2]. Su funcionamiento se basa en la escucha de todo el tráfico de red, tratando de detectar anomalías que puedan llegar a suponer un riesgo potencial. Para ello, este tipo de sistemas realiza un análisis del tráfico en tiempo real, es decir, analizan los paquetes buscando patrones sospechosos del tráfico entrante, saliente y local.

Sin embargo, los nuevos tipos de usos indebidos surgen casi todos los días y su detección no es una tarea sencilla. Incluso la red más modesta puede producir enormes cantidades de datos [3] [4].

Las tareas de monitorización y gestión de la red siempre han sido muy complejas y están en constante evolución. Además, se añade un grado de dificultad a la hora de la realización de la tarea conforme el tráfico aumenta en el enlace de red. Con los nuevos avances tecnológicos, se nos ofrece una mayor potencia de cálculo y una mayor capacidad de almacenamiento en los equipos de red. Esto provoca, una ampliación de la disponibilidad de información que es posible tener en cuenta para la realización de las tareas de monitorización y gestión de una red. En un principio, el gestor de la red nos mostraba dicha información, de forma agregada (por ejemplo, bytes de entrada/salida por una interfaz de red), mediante el protocolo SNMP (*Simple Network Management Protocol*). Actualmente, se ha planteado la posibilidad del uso de NetFlow. Esta posibilidad, proporciona un gran volumen de información de tráfico en flujos de red, con mayor detalle que los datos agregados de SNMP[5].

La motivación de este Proyecto de Fin de Carrera es averiguar si es posible utilizar dicha información proporcionada por los NetFlows para la detección de ataques en la red. En caso afirmativo y, teniendo en cuenta las características de los registros NetFlow, se podría desarrollar un sistema de monitorización de una red para la detección de ataques/anomalías.

El escenario de aplicación, de este sistema de detección, es la red académica española (RedIRIS). Concretamente, hemos seleccionado el tráfico recogido en 6 exportadores de registros de NetFlow durante el año 2013. Se pretende así ofrecer un soporte al gestor de la red, del comportamiento anómalo visto en los exportadores, tanto a un nivel interno como un nivel externo, durante el periodo de análisis propuesto anteriormente.

1.2. Objetivos

El objetivo de este Proyecto Final de Carrera ha sido desarrollar un sistema de detección de escaneos de red utilizando registros NetFlow muestreados. Para ello, se trabaja con registros NetFlow reales de la Red Académica Española (RedIRIS), con una frecuencia de muestreo de 1 de cada 100 paquetes.

Para lograr este objetivo, también ha sido necesario alcanzar los siguientes sub-objetivos:

- Ser capaz de trabajar con un gran volumen de datos de tráfico de red.
- Aprender awk como lenguaje de programación útil para el manejo de grandes volúmenes de datos.
- Conocer las vulnerabilidades de los sistemas informáticos en la red.
- Caracterización del ataque escaneo de puertos.
- Aprendizaje de métodos estadísticos para la clasificación del tráfico.

1.3. Fases de realización

Seguidamente, se enumeran y citan brevemente, las fases de trabajo que se ha seguido en la elaboración de este Proyecto Final de Carrera.

- Estudio y posterior análisis del estado del arte de los ataques en red. En esta fase se ha establecido el tipo de ataque con el que se ha trabajado de los que existen en la actualidad. Concretamente, en el presente documento se ha focalizado en la detección de posibles escaneos de puertos en los registros NetFlow proporcionados.

- Estudio del estado del arte de escaneos de puertos. En esta fase se ha adquirido el modus operandi de este ataque y se han estudiado los diferentes métodos de detección del mismo.

- Preparación del entorno. Instalación del entorno de desarrollo. Estudio y búsquedas de software y librerías necesarias para el desarrollo.

- Implementación del sistema de detección basado en el algoritmo más eficiente desde el punto de vista computacional. Para llevar a cabo la implementación se ha seleccionado el lenguaje de programación a utilizar (AWK en el entorno GNU/Linux).

- Pruebas funcionales. Probar que el funcionamiento del sistema de detección planteado corresponde con el diseño para los registros capturados.

- Reprogramación, corrección de errores vistos tras las pruebas funcionales.

- Pruebas funcionales. Validación del proceso de reprogramación y corrección de errores.

- Estudio de los resultados obtenidos y obtención de gráficas y conclusiones.

- Redacción de la memoria del proyecto. En el presente documento se describen los pasos realizados durante las fases anteriores, con los resultados obtenidos en las mismas, y finalmente, las conclusiones obtenidas en el mismo y su trabajo futuro.

1.4. Estructura del documento

Como desenlace del presente capítulo, vamos a describir la estructura que se va a seguir a lo largo del documento. La memoria está organizada en los siguientes capítulos:

- En el capítulo 2, Estado del arte, se exponen las diferentes ramas de investigación actuales que se han considerado importantes en el desarrollo del presente proyecto. Comentaremos brevemente de las técnicas de monitorización de una red existentes actualmente. Definiremos el concepto de estudio longitudinal de una red y los estudios actuales en este ámbito para proseguir hablando sobre los sistemas implementados en la actualidad, para la detección de anomalías en la red. Por último, mostraremos la aplicación de los flujos de red en seguridad para centrarnos en el objetivo del presente proyecto.

- En el capítulo 3, RedIRIS, datos y sistema de detección de escaneos de puertos planteado, se hace una descripción del escenario de aplicación de nuestro sistema de detección, la red académica española (RedIRIS). Describiremos el conjunto de herramientas empleadas en el proceso de pre-tratado de la información enviada por los exporters de la red de análisis.

- En el capítulo 4, Implementación, hablaremos del sistema de detección de ataques implementado. También, expondremos las reglas que se han seguido para su elaboración y el banco de pruebas realizado para la verificación de su correcto funcionamiento.

- En el capítulo 5, Resultados y conclusiones, se muestran los resultados obtenidos tras el estudio longitudinal en la búsqueda de escaneos de puertos sobre la red de estudio y sus respectivas conclusiones derivadas del estudio realizado.

- Finalmente, en el capítulo 6, Conclusiones y trabajo futuro, se plantean las conclusiones obtenidas de los resultados del estudio y se proponen nuevas líneas de investigación que podrían desarrollarse a partir de los resultados previamente obtenidos.

Capítulo 2: Estado del arte

2.1 Introducción

En este capítulo, se revisa el estado del arte en el área de la monitorización de una red y su aplicación de este tipo de técnicas de análisis y detección de comportamientos anómalos en una red. El objetivo de esta sección, radica en conocer cómo se realiza esta tarea en la actualidad y apoyarnos en las diferentes técnicas que se van a mostrar, para la realización de una posible solución de un bajo coste computacional.

Empezamos hablando brevemente acerca de los esquemas de monitorización de una red más empleados actualmente. Después, expondremos el concepto de estudio longitudinal de la red. Posteriormente, se explican los sistemas de detección de ataques y los tipos de ataques que se observan en una red. Con ello, nos centraremos en la finalidad del proyecto (los escaneos de puertos). Como último concepto del presente capitulo, comentaremos la aplicación de los flujos de red en la detección de este tipo de irrupciones.

2.2 Monitorización

En redes de comunicaciones, definimos la monitorización a tres niveles.

- Monitorización basada en estadísticas agregadas.
- Monitorización basada en paquetes.
- Monitorización basada en flujos.

2.2.1 Monitorización basada en estadísticas agregadas

Este tipo de herramientas, muestran el número de eventos o volúmenes de tráfico a lo largo del tiempo. Su principal uso es para graficar el ancho de banda circulante, en ambas direcciones de un enlace de red, a distintas escalas de tiempo (días, semanas y años).

MRTG y RRDTool son ejemplos de herramientas que se emplean en este tipo de monitorización [6]. Mediante el empleo de la herramienta MRTG, se obtienen las medidas del ancho de banda empleado, realizando una petición SNMP a un router.

Dicha petición, permite conocer cuántos bytes han sido transmitidos y/o recibidos en dicho router desde el momento de inicialización del mismo. Una vez es conocida dicha información, la herramienta RRDTool nos permite la representación de estas peticiones a diferentes escalas temporales.

Además, es posible conocer otras medidas. Algunas de ellas son la carga de CPU o la memoria empleada por interfaz.

2.2.2 Monitorización basada en paquetes

Este tipo de monitorización se fundamenta en la captura de la totalidad o una fracción de los paquetes que circulan por el enlace de red, para su posterior estudio.

El problema principal que se observa en dicho esquema, ocurre en los enlaces de alta capacidad. Para ejemplificar dicha afirmación, vamos a suponer un enlace de red de 10 Gb/s. Asumiendo una captura de paquetes de pequeño tamaño, obtenemos una tasa de más de 14 millones de paquetes por segundo. Con estas cifras, en primer lugar, existe un problema en su captura [7]. Además, resulta un reto, proceder a realizar cualquier tipo de análisis de los datos capturados.

Las investigaciones han prestado especial atención a dicho problema para los enlaces de red de alta capacidad. Una solución que se plantea es la mejora del rendimiento de los PCs [8]. La manera que se ha propuesto para dicha mejora se divide en la aplicación de dos optimizaciones parciales. La primera consiste en la modificación de la pila de red o los drivers de red, con el objetivo de sacar un máximo partido a las CPUs multiprocesador. Un ejemplo de modificación de la pila de red consiste en saltarse procesos propios de desfragmentación/segmentación de IP o TCP. La segunda optimización parcial radica en la modificación de la forma de acceso a los paquetes, esto es, en procesarlos en bloques en vez de forma individual. [9]

Otra proposición que se plantea son los desarrollos hardware, que poseen objetivos similares[10][11]. Estos desarrollos ofrecen tasas de funcionamiento equivalentes o incluso ligeramente superiores. La contrapartida principal de este tipo de desarrollos es el coste del equipo.

Asimismo, tienen otras desventajas como su poca flexibilidad a los cambios tanto en el tipo de monitorización como en la red[12].

En conclusión, este tipo de monitorización nos otorga mayor resolución a costa de un mayor coste en recursos.

2.2.3 Monitorización basada en flujos de red

Se define como un flujo de red al conjunto de paquetes consecutivos que comparten una serie de valores a niveles 2, 3 y 4 de red [13][14]. En general, estos paquetes consecutivos comparten misma interfaz de entrada y salida, misma dirección IP origen y dirección IP destino, mismo protocolo y mismo número de puerto origen y destino.

Hace más de una década, Cisco introdujo en sus routers, la capacidad de generar un nuevo registro por cada nuevo flujo que llega a alguna de sus interfaces. Su intención inicial trataba de acelerar el proceso de encaminamiento de los paquetes, evitando tener que realizar una misma búsqueda en la tabla de rutas para paquetes con idénticas características. Esta idea inicial fue evolucionando y, de forma rápida, se conoció el potencial de estos flujos de red en otras áreas de investigación. Cisco los nombró como NetFlows. Dicha capacidad no solo está disponible en los routers de Cisco, sino también existen otros fabricantes como Juniper, Cflowd, Riverstone, etc. [15][16][17].

En la actualidad, tenemos 10 versiones de este protocolo abierto desarrollado por Darren Ker y Barry Brunis en Cisco Systems. Netflow V1-9 e IPFIX (también conocido como V10). Dependiendo de la versión NetFlow empleada en la exportación de los flujos, contendrán una serie de campos adicionales u otros.

La versión de exportación NetFlow más empleada a día de hoy es la versión 5 del protocolo. Se debe principalmente a la información adicional agregada sobre los sistemas autónomos (BGP) y números de secuencia a la primera versión NetFlow, ya en desuso. Tras la versión 5, la versión más utilizada es la versión 9 por la flexibilidad y extensibilidad que aporta. Permite la exportación en varios formatos de registros NetFlow, incluye etiquetas MPLS, direcciones y puertos IPv6 y permite agregación en los routers.

Tras este protocolo implementado por Cisco, surgió la necesidad de un protocolo estándar y universal para la exportación de flujos de red desde diferentes dispositivos de red. Esto provocó que NetFlow se convirtiera en un estándar. Por ello, la IETF, mediante el grupo de trabajo IPFIX [18] (*Internet Protocol Flow Information Export*) creo dicho estándar IPFIX, basado en la versión 9 del protocolo de exportación NetFlow[19].

La siguiente tabla, nos resume las versiones existentes actualmente de NetFlow.

Tabla 1: versiones NetFlow [20]

Versión	Información
v1	Primera implementación ya obsoleta y limitada a IPv4 sin máscaras
v2	Versión interna de Cisco nunca publicada
v3	Versión interna de Cisco nunca publicada
v4	Versión interna de Cisco nunca publicada
v5	La versión más extendida y utilizada en numerosos routers, sólo IPv4
v6	Versión no admitida por Cisco
v7	Igual que v5 pero con un campo del router origen
v8	Agregación de flujos sólo para información que ya está presente en v5
v9	Versión basada en plantillas. Orientado a flujos IPv6 y MPLS
v10	Versión basada en la V9, creada por el grupo de investigación IPFIX

Una característica importante a destacar sobre los registros NetFlow es que no contienen información de usuario, solo datos de conexión, lo que permite una visión detallada del comportamiento de una red. Asimismo, evita problemas relacionados con la privacidad de los usuarios.

En la siguiente figura, se expone la arquitectura empleada en la formación de los registros NetFlow a partir de los paquetes que llegan al router. Una vez el paquete atraviesa el router, se extrae su quíntupla y busca en una tabla de flujos abiertos (NetFlow cache) si tenemos una coincidencia con los registros previamente guardados o si, por el contrario, se debe crear una nueva entrada en la tabla.

Si estamos ante el primer caso, únicamente se actualizará la entrada previamente creada. En el segundo caso, se añadirá un nuevo registro en dicha tabla donde acogeremos la quíntupla extraída previamente.

En ambos casos, aparte de tener la quíntupla, se posee cierta información adicional como el número de paquetes y bytes agregados entre otro tipo de información según la versión de NetFlow [21].

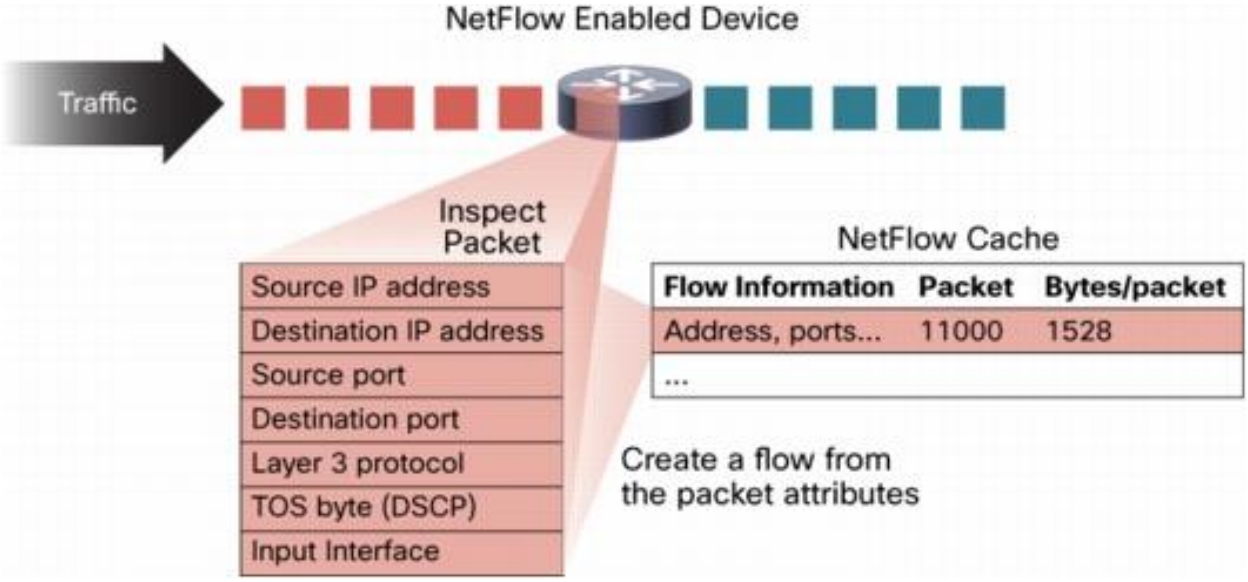


Ilustración 1: Creación de registros NetFlow [22]

Esta NetFlow cache se encuentra limitada por la capacidad que posee, el propio router. Por lo tanto, podemos distinguir entre dos tipos de flujos: flujos activos y flujos caducados. Las razones para pasar de un estado activo a un estado caducado son diversas. Cisco consideraba que si un flujo activo llevaba más de 15 segundos sin recibir nuevos paquetes con la misma quíntupla (denotado como tiempo de *timeout*) que la situada en la cache, significaba que éste había caducado. Tenemos otra serie de razones para considerar que un flujo activo, ha caducado [23]:

- La duración del flujo activo es superior a 30 minutos.
- El número de flujos en estado activo ha sobrepasado el límite de la capacidad del router.
- Se identifican banderas TCP de fin de conexión (FIN o RST).

Una vez tenemos alguno de los casos citados anteriormente, tenemos dos posibilidades para dictaminar qué hacer con el flujo [24]. Una de ellas es que pueden ser simplemente eliminado, con lo que se perdería información muy valiosa contenida en dichos flujos. La otra manera es realizar un proceso de exportación de los flujos caducados para su posterior uso. De esta tarea de almacenamiento de flujos caducados se encargan los colectores. El software más empleado para guardar de forma ordenada dichos flujos es el conjunto de herramientas Flow-Tools [25].

Esta figura resume el conjunto de fases que se dan en el proceso descrito anteriormente

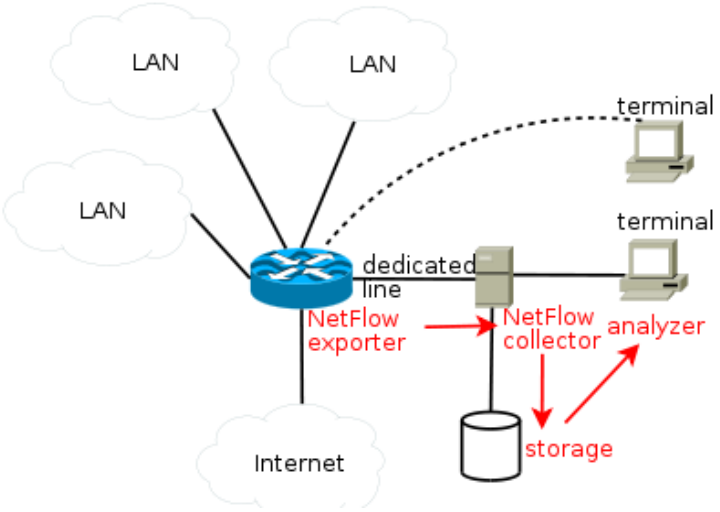


Ilustración 2: Arquitectura NetFlow [26]

En función de la política escogida en el proceso de captura del router, tenemos dos tipos diferentes de flujos [27].

- Full NetFlow es el sistema de mayor precisión, puesto que detecta todos los paquetes que atraviesan por el router. Este tipo otorga mayor fiabilidad pero, a su vez, es un sistema muy costoso en términos de recursos y memoria. En un enlace de alta capacidad, el router puede tener problemas y no responder con normalidad.

- Sampled NetFlow, a diferencia de lo anterior, escoge ciertos paquetes de acuerdo a una política definida previamente (por ejemplo, 1 paquete de cada 100 que atraviesa la interfaz). Esto nos permite recoger estadísticas para un subconjunto de entradas de tráfico por la interfaz.

La frecuencia de muestreo es una característica variable, que nos permite reducir, por una parte la carga de CPU y por otro lado el volumen de tráfico de la interfaz a almacenar. El muestreo es útil para ciertas tareas de planificación de una red, ya que no es necesario analizar la totalidad de flujos que atraviesan una interfaz para determinar su comportamiento. La contrapartida de uso de estas técnicas es la calidad de la aproximación, ya que depende tanto de la frecuencia de muestreo escogida como de la manera en la que se realiza la elección de los paquetes. Para ello, Cisco nos ofrece tres tipos de muestreo: muestreo determinista, muestreo aleatorio y periodo de muestreo [28]. El muestreo determinista, toma como frecuencia de muestreo 1/N, con lo que selecciona todos los paquetes múltiplos de N. El muestreo aleatorio, selecciona cada paquete con una probabilidad p de 1/N. Esto nos dice que, en media, muestreara un paquete de cada N, en posiciones no determinadas. El tercer método que propone Cisco, selecciona un paquete para muestrear cada N milisegundos. En la siguiente figura pretendemos aclarar el funcionamiento del muestreo aleatorio frente al muestreo determinista. En la parte superior, el muestreo determinista elige el primer paquete de los cinco que atraviesan la interfaz en ese momento. En la parte inferior de la figura, el muestreo aleatorio escoge, en media, un paquete de cada cinco que atraviesan la interfaz, sin importar el orden.

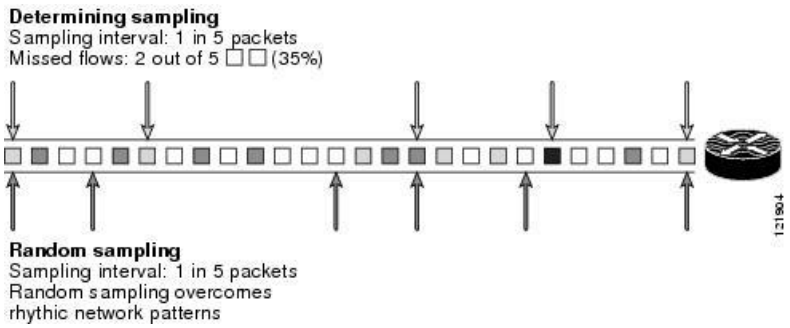


Ilustración 3: muestreo aleatorio y muestreo determinista [28]

2.3 Estudio longitudinal de una red

El estudio de una red a largo plazo, no ha tenido tanto impacto como la monitorización de la misma, ya que no permite dar una respuesta rápida a problemas acontecidos en la red. Este tipo de estudio, ofrece una respuesta a los problemas de la red a posteriori (meses después del inicio del estudio).

Este tipo de análisis de una red posee una serie de desventajas que han restado interés a su empleo. Actualmente, se tienen muchos problemas para encontrar realmente redes de comportamiento invariante. Asimismo, poseemos dificultades técnicas para la captura y almacenamiento de estas medidas de red por un periodo largo [29].

Aun habiendo poco interés, cabe destacar las siguientes investigaciones:

En este artículo [30], se presta atención a la evolución del ancho de banda, durante un periodo de 7 años y un día, de una red académica japonesa. Las primeras conclusiones que se alcanzaron fueron que las medidas varían con el tiempo, de forma constante. A pesar de ello, su ocurrencia es de forma aleatoria lo que dificulta el modelado de la red y dificulta la extracción de conclusiones firmes. Otra conclusión que lograron fue respecto al ancho de banda de la red. Concluyeron que el caudal medio por flujo aumenta anualmente un 15%. A su vez, observaron que la distribución de aplicaciones es estable y sufre un incremento de su popularidad (aplicaciones de tipo P2P) durante dicho periodo.

Tenemos otra investigación [31] en la cual, durante un periodo de 3 años y mediante el manejo de la entropía como una distribución de segundo orden para detectar cambios en el comportamiento del tráfico de la red. El escenario de aplicación es la red de la Universidad de Oregón.

Como tercer estudio relacionado con las medidas de red longitudinal, cabe destacar el siguiente [32]. En este estudio es más centrado en el usuario con medidas de flujos extendidos en un periodo de dos años. Se entiende por flujo extendido, aquel flujo compuesto de la quíntupla definida anteriormente más otras características que el gestor de la red considera importante.

Estas características mencionadas pueden ser la carga útil de una fracción de los paquetes, el tiempo entre paquetes, mecanismos de identificación de aplicaciones o incluso de que aplicación se trata tras inspeccionar la carga útil. Tienen suficientes conocimientos para relacionar direcciones IP y usuarios y además, establecen unos tipos de usuario. Afirman que el conjunto de usuarios se diferencia en tres clases: La primera clase, de mayor composición de usuarios y preferida por los ISPs, la conforman aquellos usuarios que usan su conexión para navegar subir y descargar volúmenes diarios alrededor de 1 MB. La segunda clase, de menor composición y desfavorable para las ISPs, aquellos usuarios que usan gran variedad de aplicaciones y descargan TBs diarios. También estudian de forma comparativa grupos de usuarios que tiene distinta capacidad de subida y descarga, encontrando patrones diferenciadores claros. En general más sensible al incremento del ancho de banda de subida que de bajada. Estas medidas son de utilidad para el operador de red. Permiten dimensionar su red de acuerdo al número de usuarios en vez de al número de flujos, o el caudal de estos que es un dato a medir y, habitualmente no conocido. Igualmente, le permite tener una estimación del impacto que tendría en su red un potencial incremento del ancho de banda, tanto subida como bajada, para ser ofertado a sus clientes.

En el siguiente artículo [33], tenemos diferentes trazas de una hora sobre la red de la Universidad de Auckland en Diciembre de 2003. Se utiliza la entropía de la distribución del tráfico como ayuda para la realización de una monitorización de la red.

Como última investigación [34], se estudia las redes P2P, más concretamente BitTorrent. El periodo de análisis es de 5 meses. Se analizan diferentes métricas como el caudal medio por conexión, por cada usuario, el tiempo de duración de la conexión, intervalo de peticiones realizadas, etc. Concluyeron que este tipo de red era una alternativa real a los esquemas cliente-servidor.

2.4 Sistemas de detección de ataques en red

Una vez presentados los esquemas de monitorización presentes en la actualidad, en la presente sección se explica, de forma conceptual, los sistemas de seguridad para la detección de ataques en red. Expondremos como ejemplos de este tipo de sistemas los Firewalls y los Sistemas de Detección de intrusos (IDS).

2.4.1 IDS (Intruder Detection System)

Este tipo de sistemas de detección de anomalías se encargan de monitorizar el tráfico circulante en una red y los posibles eventos ocurridos.

Con ello, se permite identificar intentos de ataques o cualquier tipo de actividad sospechosa o maliciosa, que puedan comprometer la seguridad de una red o host.

Como sistema, los IDS aportan una capacidad de prevención y alerta anticipada a cualquier actividad anómala avistada en la red. Además, incrementan la seguridad en la red, examinando los paquetes en búsqueda de patrones, que puedan afectar a los elementos que componen la red [35] [36].

Un Sistema de Detección de Intrusos se compone de los siguientes elementos:

- Fuentes de recolección de datos: El objetivo es conseguir de una forma eficiente, todos los datos necesarios durante el proceso de detección de intrusos. Pueden venir de logs o del software de base de datos.

- Reglas de contenido de datos: Poseen patrones de tráfico para la detección de anomalías en el sistema.

- Filtros: Sirven para un primer análisis de comparación del trafico monitorizado con las reglas de contenido de datos definidas previamente.

- Detectores de eventos anormales en el tráfico de red: Permite al sistema actuar como detector de amenazas e intrusos para evitar un posible ataque en la red.

- Dispositivo generador de informes y alarmas: Son los elementos encargados de avisar al administrador de la red de aquellas posibles amenazas que puedan afectar a la red.

La actividad de un sistema, generalmente, se suele registrar en un fichero o en una base de datos. Se denominan *logs* del sistema. Se les va añadiendo información, a medida que se van realizando acciones en el sistema [37].

Debido a las funciones que desempeñan estos tipos de sistemas se pueden clasificar de acorde a dos enfoques:

- El primer enfoque es de acuerdo a los sistemas que vigilan. Se encargan de analizar actividades que provienen de un host o de la propia subred en la búsqueda de posibles ataques. Tenemos los IDS basados en Red o los IDS basados en Host [36].
- El segundo enfoque se realiza en función del desempeño en la detección de ataques. Se subdivide, a su vez, en dos secciones pertenecientes al mismo grupo. Aquí tenemos los sistemas basados en la detección de anomalías y los sistemas basados en la detección de usos indebidos del sistema [36] [38].

2.4.2 Firewalls

Aparte de los sistemas IDS definidos en la subsección anterior, tenemos otros elementos en una red de comunicaciones que actúan como un filtro de datos o paquetes para la prevención de un posible ataque o intrusión. La finalidad de este tipo de filtros es el bloqueo de acceso no autorizado. Para ello, contienen varios dispositivos destinados a permitir, limitar y descifrar el tráfico circulante conforme a un conjunto de normas definidas.

La implementación de estos sistemas es tanto en Hardware como Software. Todo mensaje entrante o saliente de la propia intranet es examinado por el Firewall y se bloquean aquellos mensajes que no verifican los criterios de seguridad definidos (Véase figura adjunta).

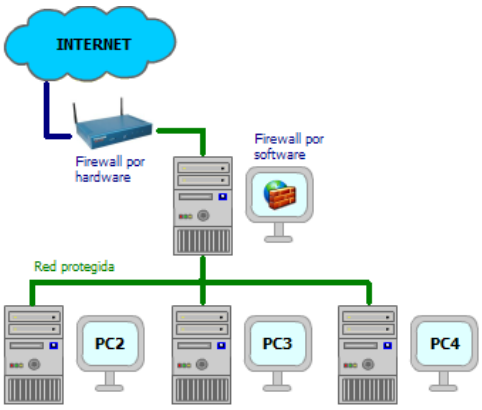


Ilustración 4 Esquema de una red empleando un firewall [39]

Habitualmente este tipo de sistema se conecta con una red interna, denotado como zona desmilitarizada (DMZ) [40].

Esencialmente, un Firewall se define como un filtro de paquetes bajo los siguientes parámetros de análisis:

- Dirección IP fuente y destino de los paquetes que atraviesan el Firewall.
- Análisis de puertos destino para los paquetes con protocolos TCP/UDP.
- Generación de mensajes de notificación de errores en el protocolo IP (ICMP)
- Análisis de los paquetes SYN/ACK.

Existen dos clases de Firewalls clasificados de acuerdo a la capa de nivel del modelo OSI en donde desarrollan su cometido definido.

- Firewalls a nivel de red (también denotados como filtros de paquetes IP), localizados en el nivel 3 del modelo OSI.
- Firewalls a nivel de aplicación, ubicados en el nivel 7 del modelo OSI [41].

2.5 Ataques en la Red

Un ataque o intrusión se define como un evento en la red que aprovecha cualquier tipo de vulnerabilidad de un sistema informático para causar un daño sin consentimiento propio del propio usuario de dicha red. Los usuarios intrusos poseen la capacidad de conocimiento de las vulnerabilidades ya sea por vía software o hardware.

Dado el tipo de finalidad de las amenazas en la red, se clasifican en cuatro categorías, de forma genérica:

- Interrupción: Cuando se pretende dañar o provocar una no disponibilidad de un recurso del sistema. Es un tipo de ataque contra la disponibilidad del servicio.

- Intercepción: Cuando un usuario, que no posee autorización, accede a un recurso. El atacante puede ser un ordenador, una persona o un programa. Es un tipo de ataque contra la confidencialidad.

- Modificación: Cuando una entidad no autorizada, consigue el acceso y manipulación de un recurso. Este tipo de ataque es contra la integridad del sistema.

- Fabricación: Se da cuando un usuario (no autorizado), inserta objetos falsificados en el sistema. Este ataque es contra la autenticidad [42].

En función de cómo afecte el tipo de ataque a la red, se clasifican en ataques activos o pasivos.

Un ataque activo, el intruso interfiere con el tráfico que fluye por la red, explotando las vulnerabilidades de la propia red o de una víctima en particular. En función del objetivo que se busque, se sub-clasifican los ataques activos en cuatro categorías:

- Suplantación de identidad: El usuario atacante se hace pasar por un cliente legítimo capturando secuencias de autenticación de varias direcciones IP de la red. Este proceso se conoce como *IP Spoofing*.

- Reactuación: Evento que surge cuando uno o varios mensajes fidedignos son capturados y repetidos para provocar un efecto no deseado en la red.

- Modificación de mensajes: Como su propio título indica, se produce una alteración de los mensajes emitidos, ya sea de tiempo como de orden, con el objetivo de producir un efecto no autorizado.

- Degradación fraudulenta del servicio: Provoca una denegación de los recursos informáticos y de comunicación de un elemento de la red. Por otro lado, el usuario que realiza este tipo de ataque puede camuflar a la víctima dicha denegación mediante mensajes no autentificados [43].

En un ataque pasivo, por el contrario, el intruso monitoriza el tráfico en la red para la captura de información para su posterior uso. Los objetivos, que se persiguen con este tipo de ataque, son la intercepción de datos y el análisis de tráfico, para la obtención de información de la propia red. Con esta información, pretende conocer la rutina del tráfico de la red de interés.

Este tipo de ataque es muy difícil de detectar, ya que no se produce ningún tipo de alteración en los datos interceptados [44].

2.6 Escaneo de puertos

Una vez definido los tipos de ataques existentes en la red y su comportamiento, exponemos el objetivo central del proyecto, que son los escaneos de puertos. Este tipo de ataque, atendiendo a las características citadas anteriormente, se considera un ataque pasivo de intercepción. Se denomina ataque pasivo porque no provoca una alteración de datos y de intercepción, ya que busca conocer accesos no autorizados de los elementos de la red. Los escaneos se caracterizan por pequeños paquetes que sondean los sistemas objetivos. Por ello, un escaneo genera una gran cantidad de flujos distintos. Establecemos tres tipos de escaneos: Escaneo Horizontal: caracterizado por un origen (host) escaneando un puerto especifico en varios hosts destinos. Escaneo Vertical: caracterizado por fijar un único host objetivo e ir escaneando algunos puertos concretos.

El tercer tipo se compone de una combinación de ambos dos tipos citados anteriormente. Es el más común de los definidos anteriormente. El resultado de ello es una variación del tráfico en el flujo de la red. Sobre este tema cabe destacar que poseen un menor impacto en el volumen total de tráfico [45].

Investigaciones acerca de dicho tema se centran en su característica más obvia que es el elevado número de conexiones salientes no naturales presentes en la fuente que escanea. En esta vía tenemos el trabajo [46] que examina el comportamiento de un host mediante sus conexiones en cuanto a sus conexiones entrantes/salientes. Con ello permite una detección de DoS y escaneos ya que siguen el mismo problema: Hosts con un inadecuado e inusual "fan-in/out". Otra investigación [47], se centra en describir un escaneo en términos de patrones de tráfico. Con ello, se permite diferenciar entre tipos de escaneos definidos anteriormente.

Otros tipos de investigaciones, como la planteada en las siguientes investigaciones, [48] [49] ya no se centran en la detección de anomalías en el volumen de tráfico. Se plantea el uso de la medida probabilística de la entropía para mostrar regularidad en flujos de red. La entropía se introdujo en la rama de teoría de la información desde 1948 [50] y se define como una medida aleatoria de incertidumbre en un proceso estocástico. También se relaciona, de manera más común, en el área de la compresión de datos sin perdidas. Shannon la definió como "el límite teórico de la tasa de compresión de una secuencia de bits es exactamente la entropía de la secuencia". Aplicando esto a la investigación anteriormente citada, podemos ver que en el caso de un host origen que desea realizar un escaneo, la entropía global en una ventana temporal corta de tiempo es propensa a un cambio. En particular, la presencia de muchos flujos con mismos orígenes IP dará lugar a una abrupta disminución de la entropía en la distribución de las direcciones IP de la fuente. Al mismo tiempo, el anfitrión de escaneo intentará ponerse en contacto con muchas direcciones IP diferentes en (posiblemente) diferentes puertos, lo que supone un aumento de estas medidas de entropía. Estas variaciones de entropía, sirve como método de validación de la presencia de un ataque.

Otro enfoque que se le da para paliar dicho problema es mediante regresiones logísticas [51], el empleo de las distancias a los denominados "modelos de referencia" [52] e incluso el uso de la transformada S para revelar componentes de frecuencia anómalos causados por un ataque. Para ello, se convierte el tráfico a una imagen bidimensional que describe el comportamiento tiempo-frecuencia del tráfico [53].

El muestreo es otro factor importante a la hora de detección de anomalías. En este ámbito, tenemos varias investigaciones, que se centran, en analizar el impacto que provocan las diferentes técnicas de muestreo en la detección de escaneo de puertos [54] [55]. Concluyen que dicho efecto posee un impacto importante en la precisión, aumentando las tasas de falsos positivos y negativos. Se informan de impactos menores al emplear muestreo en los flujos (NetFlow). Los flujos de red soportan muestreo sistemático y aleatorio. Sobre el muestreo sistemático, el trabajo [56] nos muestra que es especialmente problemático cuando los algoritmos de detección dependen de la observación de un paquete particular (por ejemplo, la bandera SYN). En otros artículos, como [57], también encontramos que algunas métricas de detección de anomalías son más resistentes al muestreo respecto a otras, especialmente los basados en entropía, y que los algoritmos de detección basados en el recuento de paquetes y bytes afectan menos que aquellos basados en el número de flujos.

2.7 NetFlow y su aplicación en seguridad

Como se vio en los apartados anteriores los esquemas de monitorización, basados en estadísticas y paquetes, nos muestran información muy detallada a costa de un consumo elevado de recursos.

Por lo expuesto en el siguiente estudio [58], nos enseña que, los flujos de red contienen, una serie de parámetros (a nivel de conexión), de forma explícita y sencilla.

Por ello, NetFlow nos proporciona una serie de ventajas frente a otro tipo de sistemas de monitorización de las que se destacan:

- Nos ofrece una vista única del tráfico total de la red a nivel de infraestructura. Una gran mayoría de los dispositivos de red poseen la capacidad de exportación del tráfico de la red en registros NetFlow.

- NetFlow no tiene acceso a información confidencial del usuario, únicamente se controlan los datos de conexión. Otorga mayor rapidez en el proceso de control de los paquetes de una red.

- Durante el proceso de análisis de registros NetFlow, es especialmente útil. Se emplean en la detección de ataques o evoluciones de los mismos, en donde los procesos de detección basados en firmas no logran actuar eficientemente.

Como es de esperar, el muestreo es un factor importante, el cual perjudica este tipo de aplicación. Por tanto, nos interesa un factor de muestreo el cual permita un almacenamiento, lo más detallado y fiel posible de la información.

Actualmente, NetFlow está demostrando su utilidad en el ámbito de la seguridad tanto para la detección pro-activa de ataques y anomalías de la red como para la investigación de incidentes ocurridos en la misma (denominado análisis forense de la red).

Por lo comentado en líneas anteriores, se posiciona como una alternativa a los esquemas de monitorización tradicionales y los sistemas de detección de ataques en la red.

2.7.1 NetFlow para análisis forense en Red

Proporciona un soporte adicional para la realización de análisis forense sobre incidentes detectados en la red. Se debe a la capacidad adicional de almacenamiento en "on/off line" relativa a su capacidad propia de almacenamiento [58].

Por tanto, se hace imprescindible un conjunto de herramientas que, posibiliten su recolección, almacenamiento y análisis para la simplificación de las consultas externas (búsquedas en línea de comandos). Estas consultas se programan en scripts diseñados para este fin.

2.7.2 NetFlow para la detección de ataques y anomalías en Red

NetFlow nos permite la detección, en un tiempo real, de ataques producidos en la red (algunos de ellos citados anteriormente).

Existen varias técnicas de análisis de los flujos de red con la finalidad de detección de ataques y anomalías. Algunos de ellos, se describen a continuación [58]:

- Análisis básico del tráfico: Este método se fundamenta en patrones de tráfico históricos para la clasificación del tráfico en "normal" o trafico malicioso. El problema principal radica en determinar el umbral para distinguir estos dos tipos de tráfico citados anteriormente, en otras palabras, poder eliminar falsos positivos/negativos debido a su costa comprobación, en términos de recursos empleados. También, este método no es automatizable y los casos vistos bajo este esquema deben de ser examinados manualmente por un operador de red.

- Análisis basado en expresiones regulares: Se basa en el aprovechamiento de ciertos campos de los registros NetFlow (puertos, direcciones IP, bytes por paquete, etc.) para realizar una consulta (habitualmente realizada en un script). Su principal desventaja de este análisis es la adaptación a las características específicas de la red. Además, necesitan una continua revisión debido al continuo progreso lógico del tráfico en una red.

- Análisis basado en algoritmos de aprendizaje y data mining: Este procedimiento es mucho más sofisticado que los dos anteriormente descritos. Permite la misma clasificación del tráfico que los dos anteriores pero requiere un uso de inteligencia adicional (por ejemplo, basándose en la entropía).

El algoritmo, objetivo principal del presente proyecto, no se puede categorizar en las herramientas explicadas anteriormente.

2.8 Conclusiones del capítulo 2

Todo el material expuesto en los apartados anteriores, nos sirve como fuentes de información, para la comprensión de cómo se resuelven estos problemas en la actualidad. Además, permite apoyarnos para desarrollar un sistema de detección de escaneos de puertos a partir de los registros NetFlow almacenados de nuestra red.

De esta forma, como hemos visto anteriormente, podríamos aportar un soporte fiable además de los medios ya existentes. Asimismo, podemos observar las motivaciones que hay detrás de estas investigaciones y la complejidad del trabajo con estas medidas a largo plazo recogidas de la red.

En el siguiente capítulo pretendemos dar una visión general de nuestra red de estudio y la descripción del proceso de creación de los registros NetFlow, a partir de los datos que disponemos. Ya en el cuarto capítulo, describiremos la segunda parte del sistema, es decir, el algoritmo empleado para la detección de escaneos de puertos, y el banco de pruebas realizado para su verificación.

Capítulo 3: RedIRIS, datos y sistema de detección de escaneos de puertos planteado

3.1 Introducción

En este capítulo se describe el escenario donde se va a realizar el estudio longitudinal (la red académica española RedIRIS).

A su vez, se expone el sistema completo empleado en la detección de posibles escaneos de puertos, de forma genérica. Además, explicaremos el conjunto de herramientas empleadas en la conversión de los flujos de red a registros NetFlow para su posterior análisis.

3.2 RedIRIS

RedIRIS es la red académica y de investigación española. Proporciona servicios avanzados de comunicaciones a la comunidad científica y universitaria nacional. Cuenta con más de cuatrocientos cincuenta instituciones afiliadas y ofrece este servicio a más de dos millones de usuarios [59].

Esta red ha estado en constante evolución desde su despliegue inicial en 1988, cuando el Plan Nacional de Investigación y Desarrollo propuso un plan para la Interconexión de los Recursos Informáticos de las universidades y centros de investigación. En cuanto a los router de red se ha mantenido constante desde sus inicios, un router por comunidad autónoma.

La topología de esta red inicialmente fue de tipo estrella con un nodo central en Madrid y 17 enlaces de red con capacidades entre 2 y 8 Mbps [60]. En 2002 se realizó una modernización de la red, con la construcción de RedIRIS-2, cuyos enlaces evolucionaron a una mayor capacidad (155 Mbps, 622 Mbps y 2,5 Gbps). Los enlaces externos eran de 2,5 Gbps. Cuatro años después, se quiso dotar de una nueva red troncal hibrida con enlaces de 10 Gbps con IP sobre SDH e IP sobre Ethernet. RedIRIS-10, que es como se denoto a este avance, entraría en funcionamiento a principios del año 2007. En la actualidad, tenemos la evolución a RedIRIS-NOVA. Entro en funcionamiento en el año 2011. Se compone de fibra óptica oscura que permite velocidades de 10 Gbps, ampliables a 40 y 100 Gbps en un futuro [61].

En la siguientes dos figuras expondremos la evolución de la red comentada anteriormente y su topología actual.

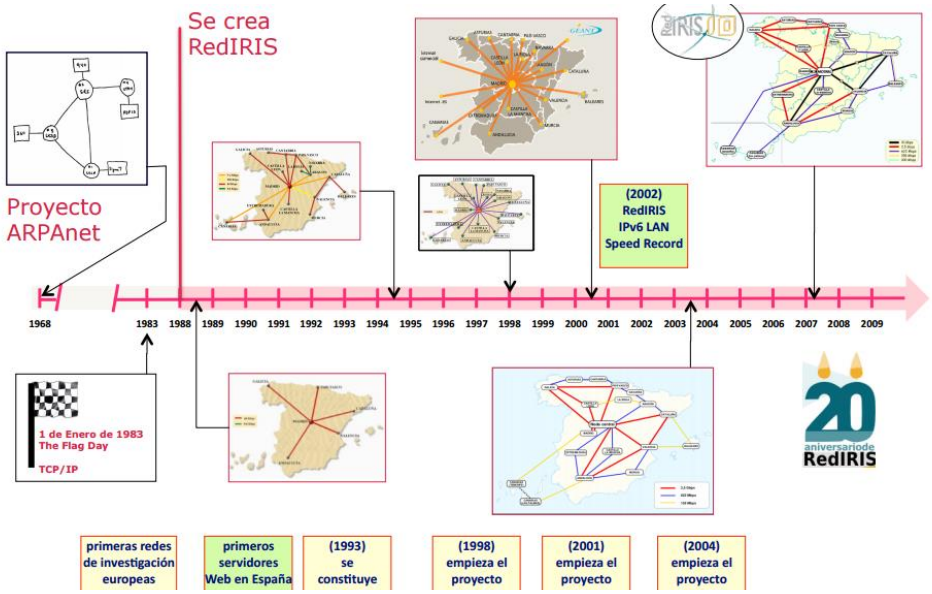


Ilustración 5: Evolución de RedIRIS [62]

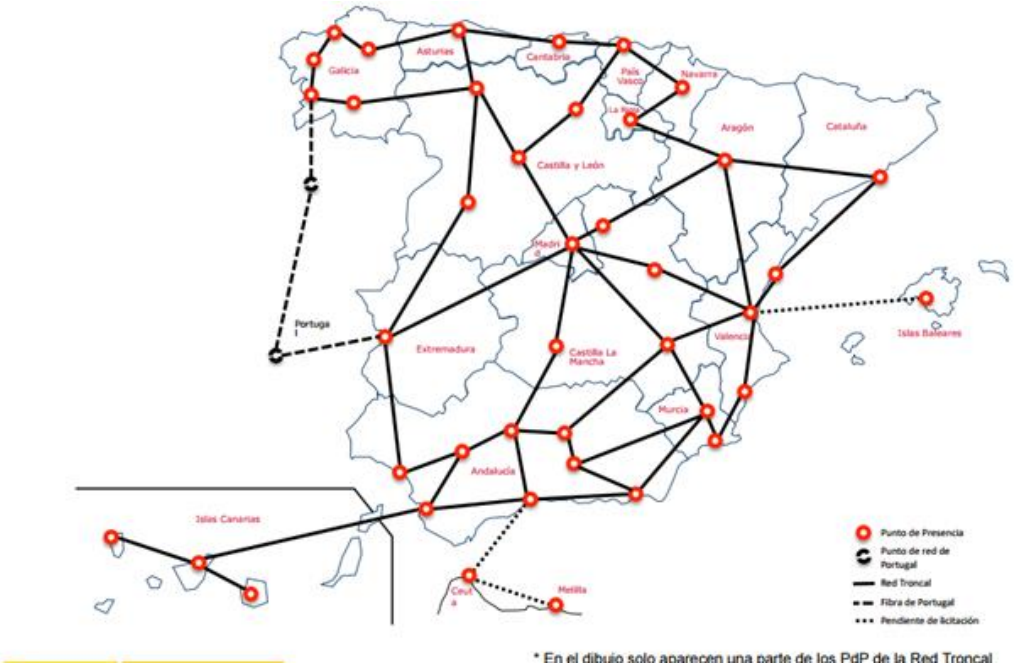


Ilustración 6: RedIRIS-NOVA [61]

Cada punto de presencia (PdP) consiste en un router troncal, que da servicios a la comunidad autónoma asociada y conectividad al tráfico itinerante. Poseen capacidades de registros NetFlow, y tienen una tasa de muestreo de 1 de cada 100 paquetes.

Dentro de los proyectos de investigación del grupo HPCN de la EPS, los routers de RedIRIS fueron configurados para que realicen una redirección del tráfico de NetFlow a un colector NetFlow situado en la Universidad Autónoma de Madrid. Los datos recogidos en dicho colector son los archivos fuentes empleados en el presente trabajo.

3.3 Descripción genérica del sistema propuesto

De forma más elemental, el sistema propuesto consta de tres etapas:

- Creación de registros NetFlow: Esta fase se comenta, de forma más detallada, en este capítulo. En esta fase se realiza un proceso de conversión de los flujos recogidos en los diferentes puntos de presencia tomados para el estudio y almacenados en el colector de la Universidad Autónoma de Madrid, a registros NetFlow para su posterior análisis.

- Análisis del tráfico: Este proceso se cuenta más detalladamente en el siguiente capítulo. En él se comentara el conjunto de reglas que nos permitirán clasificar el tráfico entrante. Con ello, seremos capaces de diferenciar los posibles escaneos de puertos del resto.

- Post-procesado de los posibles escaneos de puertos: Desarrollado en el capítulo 5. Se presentarán los resultados frutos de la fase anterior.

Ilustración 7: Fases del sistema propuesto

3.4 Creación de registros NetFlow

Los registros utilizados comienzan el 1 de enero de 2013 a las 00:00 GMT y finaliza el 31 de diciembre de 2013 a las 23:59 GMT. Por lo tanto el tiempo total de estudio es de un año en 5 orígenes distintos. El tamaño total de los registros NetFlow, preparados para el análisis, representan unos 20 Terabytes aproximadamente. Se quiere remarcar por tanto la dificultad que supone trabajar con este volumen de datos en un tiempo razonable.

Los flujos de red pueden ser de cuatro tipos diferentes [63]:

- Unidireccionales: Un flujo por cada sentido de la conexión.
- Bidireccionales: Ofrecen mayor información que los unidireccionales ya que, contienen tanto datos del origen como la respuesta del destino.
- De aplicación: Clasifican los paquetes que llegan a cada punto de presencia por su contenido acorde a la cabecera.
- Agregados: Son flujos adheridos acorde a un parámetro concreto.

El conjunto de herramientas de las Flow-Tools, que vamos a emplear en el para el fichero de texto fuente, son las siguientes [64]:

- Flow-cat: Permite la concatenación de varios flujos de red. Obtenemos información diaria.

```
eng1:% ls
ft-v05.2002-01-21.160001-0500   ft-v05.2002-01-21.170001-0500
ft-v05.2002-01-21.161501-0500   ft-v05.2002-01-21.171501-0500
ft-v05.2002-01-21.163001-0500   ft-v05.2002-01-21.173001-0500
ft-v05.2002-01-21.164501-0500   tmp-v05.2002-01-21.174501-0500

eng1:% flow-cat . | flow-print

srcIP            dstIP            prot  srcPort  dstPort  octets packets
138.26.220.46    192.5.110.20     17    62242    33456    40           1
143.105.55.23    18.123.66.15     17    41794    41794    40           1
129.15.134.66    164.107.69.33    6     1214     2222     4500         3
132.235.170.19   152.30.96.188    6     6346     1475     128          3
```

Ilustración 8: Ejemplo de uso de Flow-cat [64]

- Flow-filter: Realiza un filtrado de los flujos basándose en número de puerto, protocolo, ASN, dirección IP, bits de ToS, etc. Mediante las direcciones IP asociadas a cada exporter, podemos filtrar la información diaria acorde al origen del flujo.

```
eng1% flow-cat . | flow-filter -P119 | flow-print | head -10

srcIP            dstIP            prot  srcPort  dstPort  octets      packets
155.52.46.50     164.107.115.4    6     33225    119      114         2
128.223.220.29   129.137.4.135    6     52745    119      1438382     1022
155.52.46.50     164.107.115.4    6     33225    119      374         6
164.107.115.4    192.58.107.160   6     60141    119      5147961     8876
128.223.220.29   129.137.4.135    6     52745    119      1356325     965
128.223.220.29   129.137.4.135    6     52714    119      561016      398
130.207.244.18   129.22.8.64      6     36033    119      30194       121
155.52.46.50     164.107.115.4    6     33225    119      130         2
198.108.1.146    129.137.4.135    6     17800    119      210720652   216072
```

Ilustración 9: Ejemplo de uso de Flow-filter [64]

- Flow-print: Imprime una salida formateada de los flujos de red. Ofrece dicha salida en cualquier versión de NetFlow de las existentes.

```
eng1:% flow-print < ft-v05.2002-01-21.093345-0500 | head -15
srcIP            dstIP            prot  srcPort  dstPort  octets packets
131.238.205.199  194.210.13.1     6     6346     40355    221          5
192.5.110.20     128.195.186.5    17    57040    33468    40           1
128.146.1.7      194.85.127.69    17    53       53       64           1
193.170.62.114   132.235.156.242  6     1453     1214     192          4
134.243.5.160    192.129.25.10    6     80       3360     654          7
132.235.156.242  193.170.62.114   6     1214     1453     160          4
130.206.43.51    130.101.99.107   6     3226     80       96           2
206.244.141.3    128.163.62.17    6     35593    80       739          10
206.244.141.3    128.163.62.17    6     35594    80       577          6
212.33.84.160    132.235.152.47   6     1447     1214     192          4
132.235.157.187  164.58.150.166   6     1214     56938    81           2
129.1.246.97     152.94.20.214    6     4541     6346     912          10
132.235.152.47   212.33.84.160    6     1214     1447     160          4
130.237.131.52   130.101.9.20     6     1246     80       902          15
```

Ilustración 10: Ejemplo de uso de Flow-print [64]

Como se indicó en el estado del arte, un flujo puede ser eliminado o recopilado. En el segundo caso, los flujos se llevan hacia el colector para su almacenamiento. La información que llevan estos flujos de red se juntan, por defecto, en orden de byte, por lo que hace que estos ficheros sean portables.

El contenido de los flujos de red está almacenado siguiendo el formato de Flow-Tools que básicamente consiste en comprimir los registros en formato NetFlow cada 15 minutos. Por ello el proceso para trabajar consiste en:

Primeramente, descomprimir los registros y filtrarlos por los diferentes exporters para su pre-tratamiento. Aquellos flujos con objeto de listo se convierten a ficheros de texto. Con estos archivos, se comprueba la presencia de posibles escaneos de puertos.

Las figuras adjuntas a continuación, muestran lo comentado anteriormente.

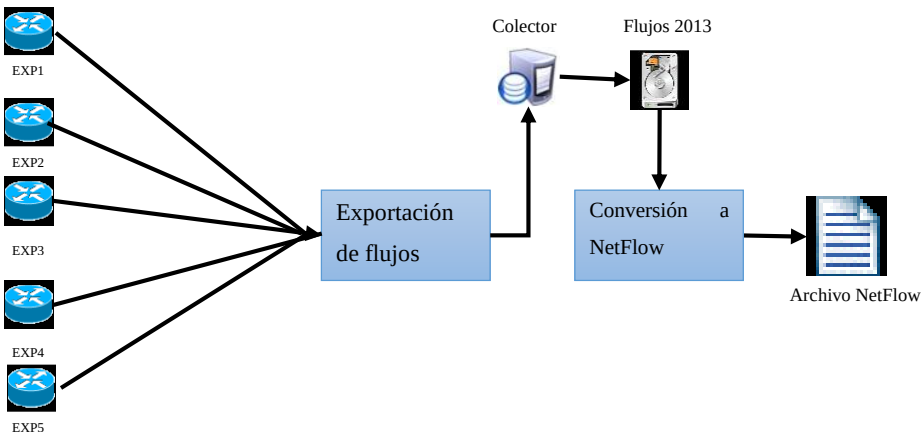


Ilustración 11: Diagrama de flujo de la etapa de creación de registros NetFlow

Debido a la cantidad de ficheros de entrada a procesar para la realización del estudio, se ha automatizado este proceso en un script en *Bash*. En este código, se emplea los tres comandos descritos anteriormente mediante un *pipe*. Un pipe se define como una técnica para pasar información de un proceso de programa a otro. En general, por cada fichero de flujos y día podemos utilizar las siguientes funciones de la librería Flow-Tools de esta manera:

```
flow-cat $(Directorio_FlujosDeRed) | flow-filter –e $(IP_Exportador) | flow-print –f5 > $(Directorio_RegistrosNetFlow)
```

A continuación, se describe lo que ocurriría con los archivos tras aplicar el código descrito anteriormente.

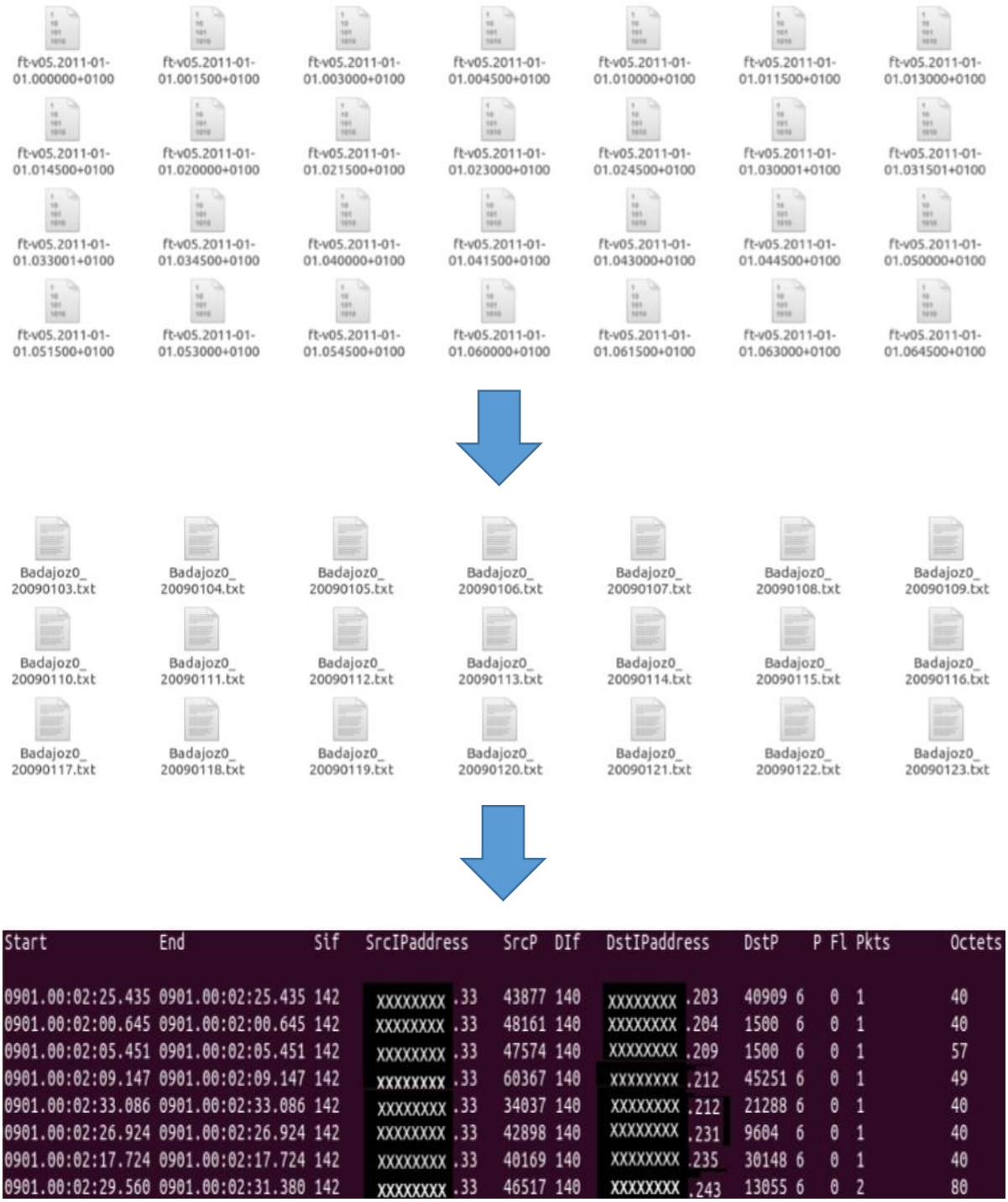


Ilustración 12: Proceso de transformación de flujos de red a registros NetFlow [63]

3.5 Conclusiones del capítulo 3

En este capítulo se expone un resumen acerca de la red elegida para la realización del estudio. Esto nos permite un primer conocimiento acerca de la red y su topología.

Además, de forma genérica, definimos el sistema de detección que se propone en este proyecto. Esto nos posibilita una total comprensión del conjunto de fases realizadas a lo largo del estudio longitudinal.

Una vez definidas esta serie de etapas exponemos la primera de ellas, el proceso de creación de registros NetFlow. Para ello, se define previamente el tipo de datos que obtenemos mediante los exportadores de red, así como el conjunto de herramientas empleadas para su correcta creación (*Flow-Tools*).

En el siguiente capítulo describiremos la segunda parte del sistema, es decir, el algoritmo empleado para la detección de escaneos de puertos y su correspondiente banco de pruebas, empleado como verificación.

Capítulo 4: Implementación

4.1 Introducción

En este apartado se expone el algoritmo creado para la detección de posibles escaneos de puertos. La finalidad de este capítulo es la exposición detallada de cómo, a partir de registros NetFlow, somos capaces de diferenciar el tráfico entrante.

Previamente a explicar dicho algoritmo comentaremos, de forma teórica, las bases del lenguaje de programación como las estructuras de datos empleadas en el mismo. Como final de esta sección se mostraran el procedimiento de verificación de este algoritmo. De forma concisa, el testeo del algoritmo se realizó a partir de un software de código abierto destinado a realizar rastreos de puertos. Su finalidad es la evaluación de la seguridad de sistemas informáticos.

4.2 Lenguaje y herramientas empleadas en el algoritmo

En esta sección se pretende comentar, en líneas generales, AWK, el de lenguaje de programación empleado en la elaboración del algoritmo, así como el conjunto de herramientas empleadas para el almacenamiento temporal del tráfico procesado.

4.2.1 AWK

AWK es un lenguaje de programación que fue diseñado para procesamiento de datos basados en texto, ya sean propios de ficheros o de flujo de datos. Su nombre se deriva de la primera letra del apellido de sus tres autores (*Alfred V. **A**ho, Peter J. **W**einberger*, y *Brian W. **K**ernighan).* Al último autor se le reconoce además por ser uno de los padres del lenguaje C. Precisamente, este lenguaje se creó para remplazar algoritmos de C para el procesamiento de archivos de texto [65] [66].

El funcionamiento elemental del AWK es la lectura del fichero que se pase como entrada línea a línea, y sobre cada línea procesada ejecuta una serie de operaciones.

Existen varias formas de usar AWK:

- En la propia línea de comandos siguiendo esta sintaxis:

```
>awk  PROGRAMA fichero_entrada
```

En donde *PROGRAMA* contiene la estructura de un programa AWK, que comentaremos a continuación, y *fichero_entrada* es el archivo de texto a procesar.

- Escribiendo el programa awk en un fichero (.awk) y ejecutándose mediante esta sintaxis:

```
>awk  -f  FICHERO_PROGRAMA_AWK fichero_entrada
```

- Ejecutando el *FICHERO_PROGRAMA_AWK* como si fuera un script. Básicamente seria añadir al principio del *FICHERO_PROGRAMA_AWK* la siguiente sentencia:

```
#!/usr/bin/awk -f
```

La estructura de un programa AWK consta de las siguientes partes:

- Parte inicial. Solo se ejecuta una única vez, antes de procesar el fichero de entrada.

```
BEGIN {operaciones}
```

- Parte central. Son una serie de instrucciones que se ejecutan para cada línea procesada del fichero de entrada.

```
{operaciones}
```

- Parte final. Se efectúa su ejecución una única vez, después del procesado de la entrada

```
END {operaciones}
```

Cada una de las siguientes expresiones AWK pueden incluirse varias veces para un mismo archivo a procesar. El fichero se procesa progresivamente y no atiende a ninguna ordenación previa de la estructura.

El manejo de los ficheros de texto se realiza de la siguiente forma. AWK divide las líneas del fichero de entrada en *campos*.

- La separación entre campos, viene determinado por la variable FS (por defecto, son espacios en blanco).

- Las variables $1, $2, $3,…, $N contienen el valor asociado a cada campo extraído del fichero. La variable $0 contiene la línea completa.

Este lenguaje posee un conjunto de variables predefinidas que se resumen en la siguiente tabla:

Tabla 2: Variables predefinidas AWK [65]

Nombre	***Significado***
FS	Carácter separador entre campos
NR	Número de registros provenientes de la entrada
NF	Número de campos en el registro de entrada
RS	Carácter separador entre registros de entrada
OFS	Carácter separador entre campos en la salida
ORS	Carácter separador entre registros de salida
FILENAME	Nombre del fichero de entrada.

Este lenguaje se creó para remplazar algoritmos de C para el procesamiento de archivos de texto [65] [66].

4.2.2 Arreglos asociativos (Associative *Arrays*) en AWK

Un *array* se define como una estructura para almacenar un conjunto de valores. Estos valores tienen algún tipo de relación entre sí. Los índices del *array* permiten el acceso a los elementos recolectados y se encierran entre corchetes.

En AWK, en particular, no se tiene que declarar el tamaño del *array* previamente como se realiza en otros lenguajes de programación.

La particularidad de este lenguaje de programación es el empleo de una estructura de datos más particular: los *arrays* asociativos. En este tipo de estructura, los índices pueden ser tanto numéricos (como en la mayoría de los lenguajes de programación vistos actualmente) como una cadena de texto. Esta doble posibilidad permite realizar una asociación entre índices y elementos que conforman el *array*. Los elementos no se almacenan siguiendo un orden particular como ocurre en el caso de un array típico.

Los arrays asociativos nos ofrecen una característica distintiva de AWK, y muy poderosa que nos ofrece el utilizar una cadena como un índice a un valor. Por ejemplo, se podría emplear una palabra como el índice a la definición de la misma. Si la palabra es conocida, podremos recuperar su definición, es decir, la búsqueda de la definición a partir de la palabra es muy rápida en términos computacionales.

Es importante explicar que todos los índices de los arrays en este lenguaje son cadenas. Incluso empleando como índices los números awk, de forma automática, los convierte en cadenas. Esto no afecta cuando se utilizan índices enteros pero si podría sufrir algún problema en la conversión número-cadena con índices reales [67].

4.2.3 Conclusiones

En estos apartados hemos dado a conocer las características del lenguaje de programación empleado en la realización del algoritmo y la herramienta de almacenamiento de los datos procesados de los registros NetFlow.

La explicación del algoritmo, objeto de la siguiente sección, va a seguir los pasos de acorde a la estructura de un programa AWK (Parte inicial, central y final). La fase de inicialización de variables (Parte inicial) no ha sido utilizada en el algoritmo.

La cabecera de todo registro NetFlow procesado presenta el siguiente formato:

Start	End	Sif	SrcIPaddress	SrcP	Dlf	DstIPaddress	DstP	P	Fl	Pkts	Octets

Ilustración 13: Campos NetFlow v5

A continuación mostramos una tabla con una correspondencia entre variables AWK y campos de los registros NetFlow que procesamos a lo largo del algoritmo.

Start	End	Sif	SrcIPaddress	SrcP	Dlf	DstIPaddress	DstP	P	Fl	Pkts	Octets
$1	$2	$3	$4	$5	$6	$7	$8	$9	$10	$11	$12

Ilustración 14: Correspondencia campos NetFlow - variables AWK

4.3 Algoritmo de detección de escaneo de puertos

Antes de comenzar a entrar en detalles del propio algoritmo vamos a resumir en este diagrama de bloques el conjunto de archivos entrada-salida que tendremos en el sistema.

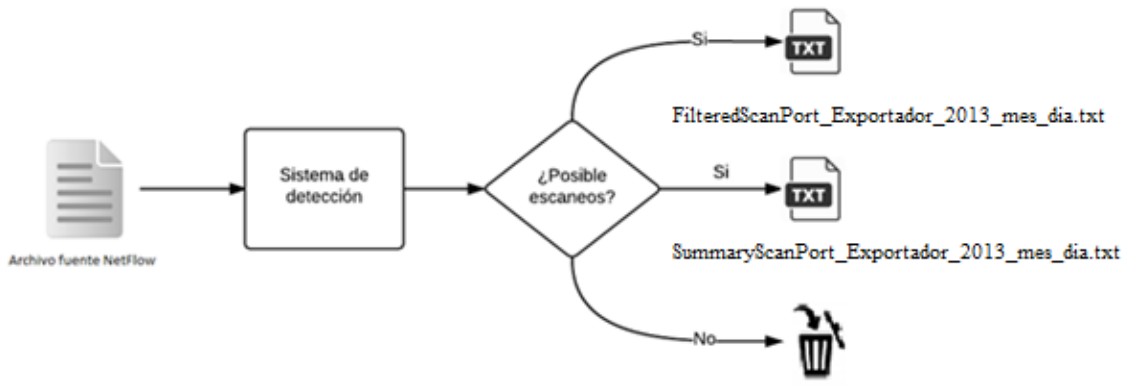


Ilustración 15: Diagrama E/S del sistema de detección

Los dos ficheros de salida contienen lo siguiente:

→ FilteredScanPort_Exportador_2013_mes_dia.txt es una copia del archivo fuente en aquellos puntos de dicho archivo donde nuestro sistema detecta los posibles escaneos. Por tanto, es un archivo NetFlow que contiene únicamente los posibles escaneos vistos.

→ SummaryScanPort_Exportador_2013_mes_dia.txt contiene un breve resumen de los escaneos avistados centrándose en la presentación de los siguientes campos.

Tabla 3: Campos del archivo SummaryScanPort_Exportador_2013_mes_dia.txt

Campos	Descripción
IPOrigen(ServerName)	Direccion IP del origen del ataque junto a su servidor de nombres asociado
IPDestino(ServerName)	Direccion IP del destino del ataque junto a su servidor de nombres asociado
(1/flproc)^nºpuertos]	Limite para el posible escaneo detectado. Variable que se compara con las puntuaciones bajo los escenarios definidos en el propio algoritmo.
Score(IPDst-Puerto)	Puntuación que mide el comportamiento IP-Puerto destino frente a todo el trafico que se procesa de posibles casos.
Score(Puerto)	Puntuación que mide el comportamiento del Puerto destino frente a todo el trafico que se procesa de posibles casos.
nºPuertosEscaneados	Total de puertos vistos en el escaneo
nºFlujosIPDst	Total de flujos implicados durante el escaneo
nºPaquetesIPDst	Total de paquetes transmitidos durante el escaneo
ListaPuertosEscaneados	Enumeracion de que numero de puerto concreto aparece como objetivo del escaneo.
Otras caracteristicas	Variables derivadas de los campos anteriores. Por ejemplo flujos por puerto, paquete por puerto, logaritmo decimal de la puntuacion IP-Puerto destino,…

4.3.1 Parte inicial

En esta primera fase del algoritmo se han dado valor a algunos límites de los sucesivos filtros, que han sido aplicados a lo largo de las siguientes etapas del algoritmo.

Tabla 4: Límites y valores asignados en la parte inicial del algoritmo

Variable	Valor asignado
lim_puerto	1024
lim_bytes	60

Esta etapa, de única ejecución, se realiza antes de procesar la entrada.

4.3.2 Parte central

En esta segunda etapa, se comienza a procesar la información que contiene el archivo de entrada. Para evitar el procesamiento de todo el fichero, estableceremos una serie de características típicas que verifican todo tráfico propenso a ser un barrido de puertos:

- La primera de ellas nos la aporta el siguiente estudio [68]. En él, nos explica que un posible escaneo de puertos se caracteriza por una transferencia baja (a nivel de bytes). De hecho, cuantifica dicha transferencia a un límite máximo de 60 bytes.

- La segunda característica se debe al tipo de flujos de red que estamos procesando. Debido a la característica bidireccional de estos flujos, nos refleja en un mismo archivo tanto la información enviada como su respuesta recibida. Debido a ello, se debe discriminar este tráfico de servidor que tenemos en la respuesta. Este tipo de tráfico se caracteriza por un uso de puertos destino superiores a 1024, es decir, un empleo de puertos efímeros.

Estos dos filtros que hemos comentado anteriormente discriminan una parte del tráfico antes de su procesamiento, ya que estos casos no aportan información relevante para los siguientes pasos del algoritmo.

Del resto de tráfico que verifica las dos condiciones expuestas anteriormente, se recogen ciertos campos de los registros NetFlow necesarios para la siguiente parte del algoritmo. Estas características recogidas se almacenan mediante el uso de *arrays* asociativos.

Tenemos un total de cuatro *arrays* diferentes, necesarios en la última parte del algoritmo.

- El primero recoge el número de veces a lo largo del archivo de entrada que aparece un puerto determinado. De forma esquemática se define de esta manera:

$$Array[DstP] + +$$

- El segundo tendremos el número de veces que hay conexiones a un destino bajo un puerto concreto.

$$Array[DstIPaddress][DstP] + +$$

- El tercer array nos muestra el número de veces a lo largo del archivo de entrada que se conecta la dirección IP origen con una dirección IP destino bajo un puerto determinado.

$$Array[SrcIPaddress][DstIPaddress][DstP] + +$$

- El último array permite una reconstrucción del archivo fuente con aquellos casos que se han determinado como posibles escaneos. Para ello, se guarda el número de fila implicada. Se exponen respectivamente tanto el caso de que no exista valor como el caso de si ya tuviera valor.

$$Array[SrcIPaddress][DstIPaddress] = NR$$

$$Array[SrcIPaddress][DstIPaddress] = Array[SrcIPaddress][DstIPaddress]""NR$$

Por otro lado, también se recoge el número total de registros NetFlow que se procesan por archivo de entrada.

En la siguiente figura se expone un diagrama de flujo de esta parte del algoritmo.

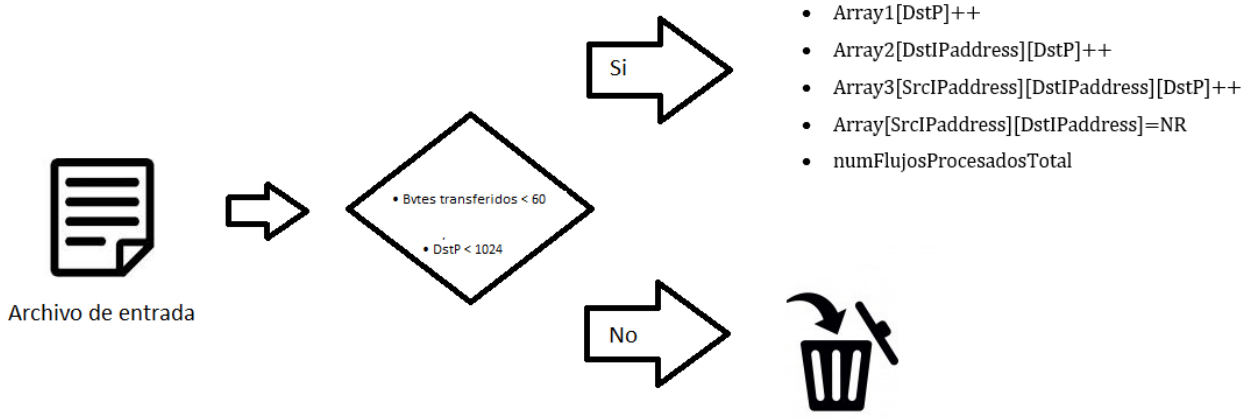


Ilustración 16: Diagrama de flujo de la parte central del algoritmo

4.3.3 Parte final

Una vez terminada el proceso de filtrado de la fase anterior, toda la información necesaria procedente del archivo fuente pasa a el conjunto de arrays. Además, se añade la variable que recoge el número total de flujos procesado del archivo entrante.

La ventaja de utilizar *arrays* asociativos permite una mayor aceleración en las consultas que se van a realizar entre las diferentes tablas. Por ello, se realiza un recorrido completo del *array*.

Sobre la información almacenada, se procede a determinar qué parte de este tráfico corresponde a posibles escaneos de puertos y qué parte no tiene dicha tendencia.

Para ello, como paso previo, se han ido estableciendo una serie de filtros más concretos, basándose en la información contenida en los *arrays*.

- El primero de ellos consiste en la idea que si un posible atacante va a realizar un barrido de puertos de un *host* objetivo, buscara probar más de un único puerto destino.

- El otro filtro empleado consiste en evaluar la conexión (IP, Puerto) $_{\text{destino}}$. Se busca descartar (IP, Puerto) $_{\text{destino}}$ con un número de conexiones altas bajo el posible barrido de escaneos que se ha detectado.

Como valor al primer filtro aplicado se ha considerado que, tras unas pruebas funcionales detalladas en la siguiente sección, sea mayor de tres puertos destino. Como segundo limite a tomar consiste en; dando valor al número de conexiones (IP, Puerto) $_{\text{destino}}$ y relativizando frente al número total de puertos identificados del posible escaneo no supere el valor de 3 $(\text{IP, Puerto})_{destino}/_{numPuertosEscaneo_{Total}}$ (conexiones/puerto).

Tras este filtrado de información, se procede al cálculo de una serie de dos puntuaciones (*Score*), que junto a la variable adicional de la fase anterior, permiten la identificación definitiva de posibles escaneos.

- Como puntuación primera se basa en el comportamiento de la par dirección IP-Puerto destino respecto al número de flujos de red total procesados. Sigue la siguiente fórmula.

$$Score_{ConexionIPPuerto} = Score_{ConexionIPPuerto} * \frac{a[Ip_{Dst}][P_{Dst}]}{numFlujos_{Procesados}}$$

- Como segunda puntuación se pretende conocer la conducta del conjunto de puertos objetivos del posible escaneo. Presenta la siguiente fórmula para su cálculo:

$$Score_{Puerto} = Score_{Puerto} * \frac{a[P_{Dst}]}{numFlujos_{Procesados}}$$

- Esta variable adicional actúa como comparador entre las puntuaciones descritas anteriormente. Se calcula de esta manera:

$$Límite = \left(\frac{1}{numFlujos_{Procesados}}\right)^{numPuertosEscaneo_{Total}}$$

Este cálculo previo de variables se realiza ya que las parejas (IP, Puerto) $_{destino}$ no presentan una distribución uniforme. Siguen lo denominado distribución de cola pesada. Debido a esta característica, unos pocos pares (IP, Puerto) $_{destino}$ llevan consigo la mayor parte del tráfico provocando que el resto de las conexiones, por poco comunes, se puedan considerar como *conexiones atípicas*. Dichas *conexiones* provocaran que su puntuación tanto a nivel (IP, Puerto) $_{destino}$ como a nivel de Puerto destino ($Score_{ConexionIPPuerto}$ y $Score_{Puerto}$) sean similares a la variable Límite. Todo ello, provocara que esos casos concretos sean detectables como posibles escaneos.

Mediante este cálculo previo, se tienen tres posibles escenarios para la determinación de posibles escaneos.

- El límite tiende a 0. En esta situación y ante un valor constante como es el número de flujos procesados, nos determina que el número de puertos involucrados en este escaneo es elevado.
- El límite tiene un valor distinto a 0.

De esto se generan dos sub-escenarios posibles:

  - En el primer caso, se debe comparar las puntuaciones respecto al valor de ese límite. Estos valores son muy pequeños para su comparación. Por ello, se debe cambiar el valor a otro escenario donde se pueda establecer una comparación menos costosa en términos computacionales. El escenario es pasar estos valores a la escala logarítmica. Para la clasificación en tráfico de posibles escaneos siguen estas dos condiciones, que se plantean a continuación:

$$(abs(\log_{10}(Limite)) - abs(\log_{10}(Score_{ConexionIPPuerto}))) < 6$$

$$(abs(\log_{10}(Limite)) - abs(\log_{10}(Score_{Puerto}))) < 6$$

    Se observan que siguen una diferenciación de valores en estos dos ámbitos descritos anteriormente, comparándose frente a un valor empírico. Se obtuvo este valor a través de las pruebas de verificación expuestas en el siguiente apartado. Tanto las puntuaciones como el límite están normalizados respecto al número de flujos procesados a lo largo del archivo de entrada. Por tanto, mediante el límite buscamos que los posibles escaneos se acerquen a ambas puntuaciones, cuantificando la diferencia de valores en 6 unidades en la escala logarítmica.
  - Puede ocurrir que las puntuaciones no se acerquen al límite fijado. En este caso se pretende comprobar si el número de puertos implicados es muy elevado. Concretamente en aquellos casos cuyo barrido de puertos sea superior a 6 puertos

En la siguiente figura se expone un diagrama de flujo de esta parte del algoritmo.

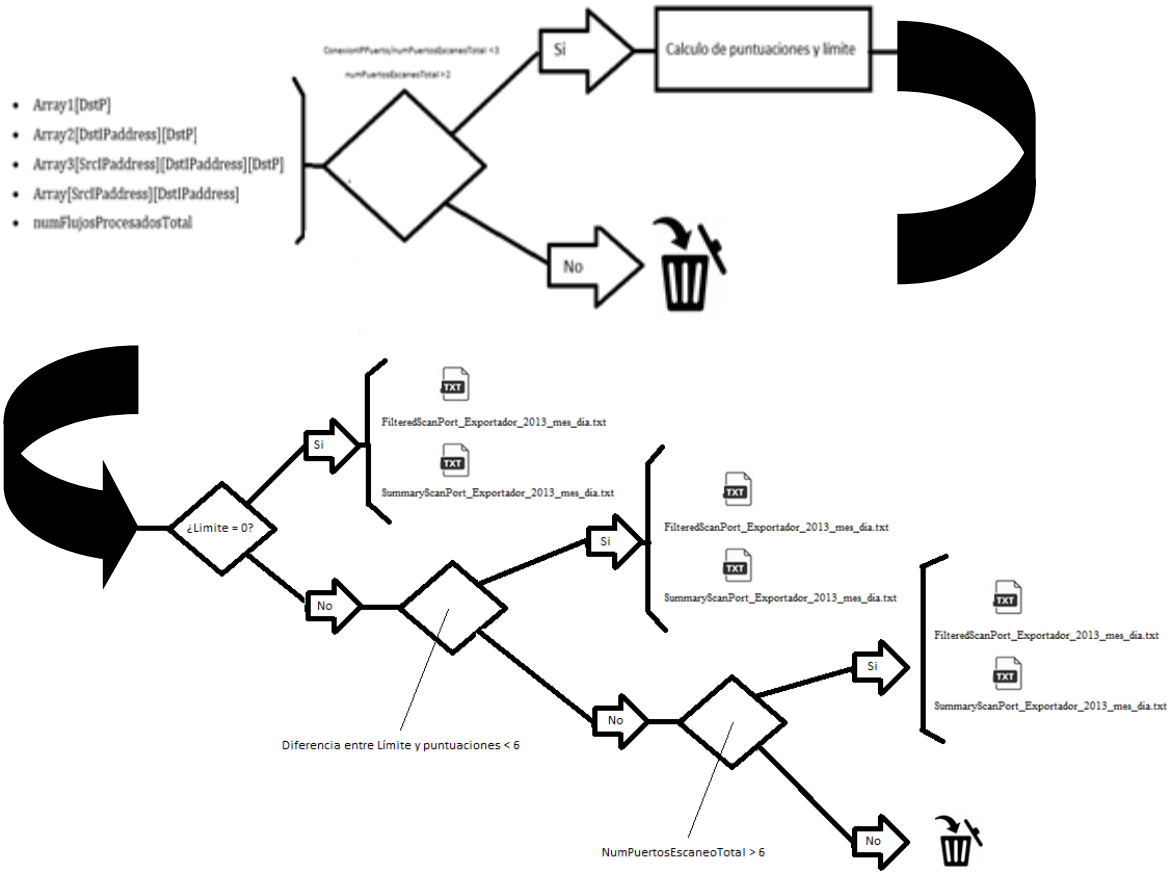


Ilustración 17: Diagrama de flujo de la parte final del algoritmo

4.4 Verificación del algoritmo

Como final del capitulo se va a exponer la metodología empleada en el proceso de verificación del algoritmo planteado anteriormente. Esta serie de pruebas consisten en dos tipos de comprobaciones:

- La primera se plantea a nivel funcional. Pretende verificar la parte central y el primer filtrado de la parte final del algoritmo. A partir de los datos extraídos de las referencias, pusimos a la entrada del presente Sistema de Detección, varios días aislados de prueba. Esto nos permitió detectar numerosos casos que pueden tener esta tendencia pero a su vez, encontramos numeroso trafico una índole diferente a la requerida.

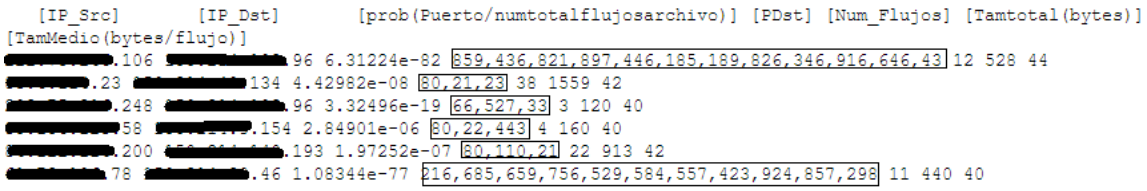


Ilustración 18: Prueba funcional del sistema de detección (I)

Como podemos observar tenemos casos claros de escaneos frente a casos que a priori podíamos denotar como tráfico normal. Por ello, surgió la puntuación a nivel de (IP, Puerto) destino para poder diferenciar, de manera completa, esta serie de casos que se generan.

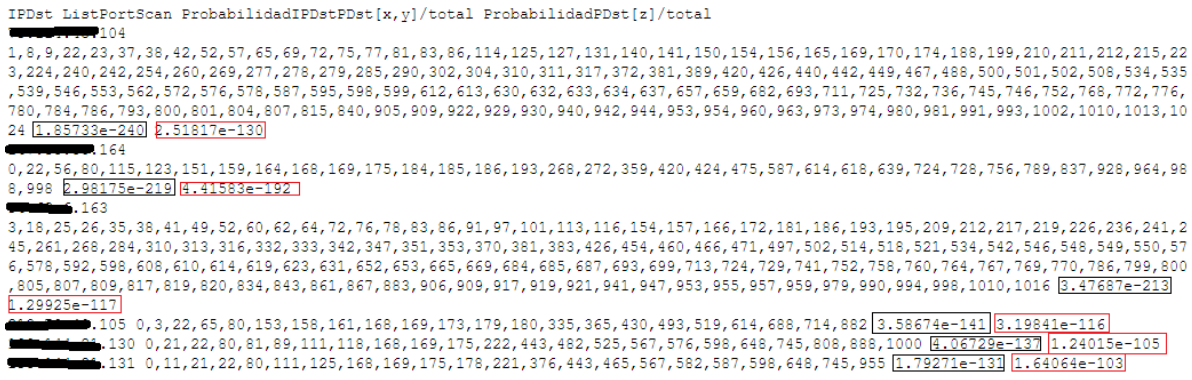


Ilustración 19: Prueba funcional del sistema de detección (II)

En la figura anteriormente expuesta extraemos dos conclusiones. La primera de ellas es que todavía no se ha diferenciado completamente el tráfico de posibles escaneos respecto al resto. La segunda conclusión es que los valores asociados a las puntuaciones son muy pequeños y es complicado establecer una comparación para poder diferenciar el tráfico. Debido a ello, decidimos tomar logaritmos a las puntuaciones previamente calculadas y apoyarnos en otra variable para poder realizar la clasificación del tráfico (Límite).

- La segunda prueba consistió en lo siguiente. Mediante el uso de un programa de código abierto (*Nmap*) que sirve para efectuar rastreo de puertos, se procedió a realizar una serie de barridos de puertos de un host dentro de la red de estudio RedIRIS. Se recogieron el tráfico entrante y saliente en ambos sentidos tanto mediante Wireshark. Ya en este paso, se observaron que el barrido de puertos se realiza completamente de forma satisfactoria en la entrada a la red y en la salida por el objetivo tomado se observa un bloqueo de paquetes. Dicho bloqueo se observó tanto a nivel de paquetes como a nivel de no realizarse los posteriores escaneos debido a que informan de que el ordenador objetivo se encuentra apagado.

```
Starting Nmap 6.40 ( http://nmap.org ) at 2014-10-20 22:37 CEST
NSE: Loaded 110 scripts for scanning.
NSE: Script Pre-scanning.
Initiating Ping Scan at 22:37
Scanning [redacted]138 [4 ports]
Completed Ping Scan at 22:37, 2.02s elapsed (1 total hosts)
Nmap scan report for [redacted]138 [host down]
NSE: Script Post-scanning.
Read data files from: /usr/bin/../share/nmap
Note: Host seems down. If it is really up, but blocking our ping probes, try -Pn
Nmap done: 1 IP address (0 hosts up) scanned in 2.44 seconds
           Raw packets sent: 8 (304B) | Rcvd: 0 (0B)
```

Ilustración 20: Salida *Nmap* Bloqueo

Se obtuvo a raíz de esta afirmación que existe un periodo de tiempo entre escaneos. Se tiene que dicho espacio temporal se sitúa en torno a 10-15 minutos aproximadamente.

```
Hora_i:22:37:42 IPOrigen:[redacted].89 IPDst:[redacted].138 Comando:nmap -T4 -A -v[redacted].138 Hora_f:22:37:42
Hora_i:22:52:44 IPOrigen:[redacted].89 IPDst:[redacted].136 Comando:nmap -T4 -A -v [redacted].136 Hora_f:22:52:44
Hora_i:23:13:25 IPOrigen:[redacted].89 IPDst:[redacted].208 Comando:nmap -T4 -A -v [redacted].208 Hora_f:23:13:25
```

Ilustración 21: Comando Nmap y tiempo entre escaneos

Posteriormente el tráfico guardado en el colector de la Universidad Autónoma es analizado bajo el sistema planteado. Tras esta fase, se pudieron comprobar los diferentes escenarios citados en la parte final de algoritmo. Además, esta auto-detección permitió cuantificar las diferencias entre puntuaciones y límite para la clasificación del tráfico.

```
Nmap scan report for [redacted] ([redacted].136)
Host is up (0.0076s latency).
Not shown: 840 filtered ports, 150 closed ports
PORT     STATE SERVICE            VERSION
22/tcp   open  ssh?
| ssh-hostkey: 1024 19:ec:06:57:17:d4:dd:e1:60:5a:a1:f8:70:7d:2a:73 (DSA)
| 2048 de:5a:68:ca:97:e6:20:62:22:d8:56:e9:d5:d8:7c:2e (RSA)
|_256 f8:a2:63:36:39:09:92:9c:36:8c:5a:f9:2b:62:41:42 (ECDSA)
53/tcp   open  domain?
80/tcp   open  http?
135/tcp  open  msrpc?
443/tcp  open  https?
|_http-methods: OPTIONS GET HEAD POST
|_http-title: Index of /
|_ssl-date: 2014-10-20T20:57:54+00:00; -1s from local time.
```

Ilustración 22: Salida *Nmap* Barrido de puertos

4.5 Conclusiones del capítulo 4

En este capítulo, de forma introductoria, se ha dado a conocer las características del lenguaje de programación empleado en el sistema de detección creado. También, presentamos la herramienta empleada para el almacenamiento de los datos procesados tras la creación de registros Netflow expuesta en el anterior capitulo. Se comenta adicionalmente la relación que guardan las variables empleadas en nuestro sistema con los campos que posee un registro Netflow.

Posteriormente se comienza la explicación del algoritmo y la serie de consideraciones a tener en cuenta para la detección de escaneos de puertos a partir de registros NetFlow. Se adjunta un diagrama de flujo para conocer la entrada/salida de los datos en cada punto concreto del algoritmo. Para finalizar el capítulo, se especifica el tipo de metodología empleada en la verificación del correcto funcionamiento del sistema frente a los escaneos de puertos.

En el siguiente capítulo, hablaremos sobre los resultados obtenidos del estudio longitudinal realizado durante el año 2013 para los cinco exportadores de red escogidos. Nos servirá para conocer el estado de cinco puntos de la red frente a este tipo de ataques. Caracterizaremos cada escaneo de puertos detectado a nivel temporal, daremos una localización Origen-Destino (espacial) y se mostrara un análisis frente al tipo de puerto involucrado.

Capítulo 5: Resultados

5.1 Introducción

En este apartado se recogen los resultados obtenidos tras la realización del estudio longitudinal de la red de análisis RedIRIS durante el año 2013. Se analizó el tráfico recogido en 5 exportadores de dicha red.

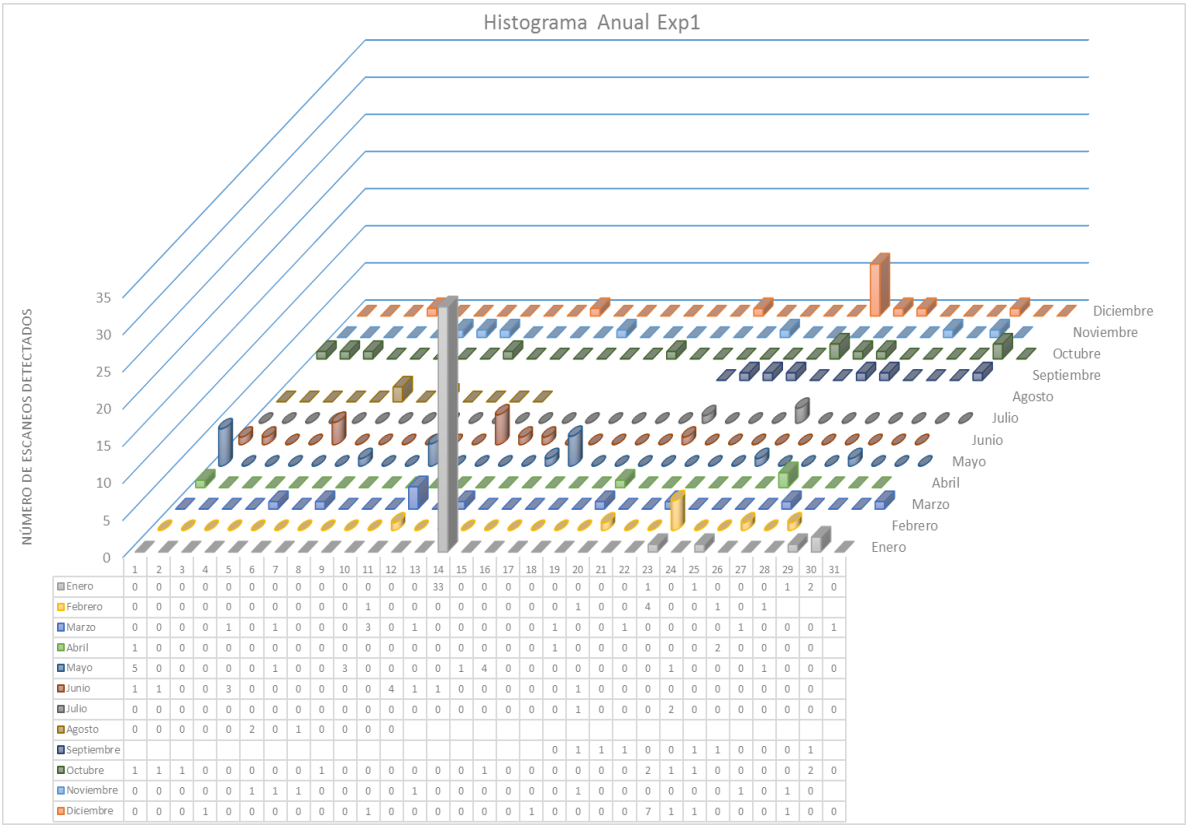
Los resultados que se exponen en las secciones siguientes pretenden dar un enfoque del posible escaneo detectado bajo distintos puntos de vista. El primero de ellos ofrece información acerca de cómo han sucedido los ataques desde un punto de vista diario, mensual y una serie temporal del año de análisis. El segundo nos aporta una localización Origen-Destino del posible escaneo. Sigue acorde a dos ideas:

- Mostrar una localización precisa del posible escaneo (a nivel de país).
- Mostrar la localización del mismo según la siguiente clasificación: Si el escaneo se realiza desde el exterior hacia dentro de la red, si es realizado desde dentro hacia fuera de la red, o si se hace entre hosts dentro de la propia red.

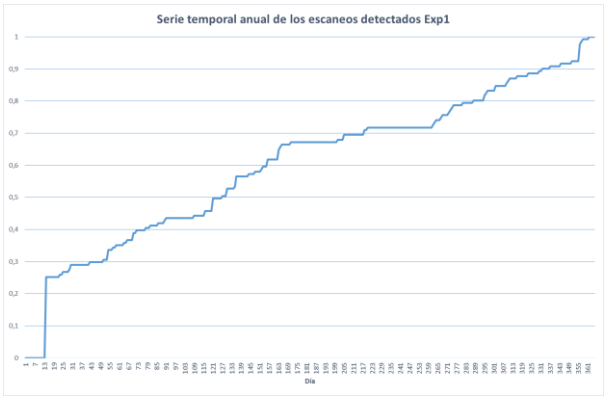
Como último enfoque se analizó la naturaleza de los puertos. Mediante la asignación de puertos de IANA [69] hemos clasificado los puertos empleados en los escaneos de acorde a su tipo de aplicación y uso. Además, se ha extraído un top 100 de los mismos.

5.2 Resultados temporales de los escaneos detectados

5.2.1 Exportador 1



	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
Enero	0	0	0	0	0	0	0	0	0	0	0	0	0	33	0	0	0	0	0	0	0	0	1	0	1	0	0	0	1	2	0
Febrero	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	1	0	0	4	0	0	1	0	1			
Marzo	0	0	0	0	1	0	1	0	0	0	3	0	1	0	0	0	0	0	1	0	0	1	0	0	0	0	1	0	0	0	1
Abril	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	2	0	0	0	0	
Mayo	5	0	0	0	0	0	1	0	0	3	0	0	0	0	1	4	0	0	0	0	0	0	0	1	0	0	0	1	0	0	0
Junio	1	1	0	0	3	0	0	0	0	0	0	4	1	1	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	
Julio	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	2	0	0	0	0	0	0	0
Agosto	0	0	0	0	0	2	0	1	0	0	0	0																			
Septiembre																			0	1	1	1	0	0	1	1	0	0	0	1	
Octubre	1	1	1	0	0	0	0	0	1	0	0	0	0	0	0	1	0	0	0	0	0	0	2	1	1	0	0	0	0	2	0
Noviembre	0	0	0	0	0	1	1	1	0	0	0	0	1	0	0	0	0	0	0	1	0	0	0	0	0	0	1	0	1	0	
Diciembre	0	0	0	1	0	0	0	0	0	0	1	0	0	0	0	0	0	1	0	0	0	0	7	1	1	0	0	0	1	0	0



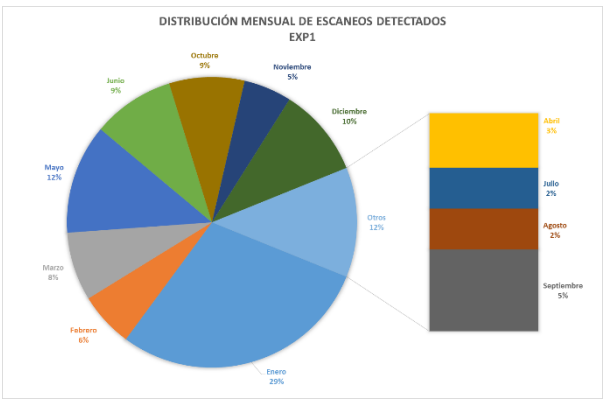


Ilustración 23: Histograma anual *(centro)*, serie temporal anual *(abajo-izquierda)* y distribución a nivel mensual *(abajo-derecha)* del número de escaneos detectados del exportador número 1.

La gráfica del histograma anual para el exportador 1 presenta lo siguiente:

Un valor máximo de 33 escaneos/día. Representan el mayor barrido diario visto en el colector de red a lo largo del año estudiado. Implica el mayor crecimiento de la serie temporal (de ¼ en un único día del mes). El mes implicado es enero y, como se observa en la distribución mensual, este crecimiento supuso un aumento del 25 % del total de escaneos.

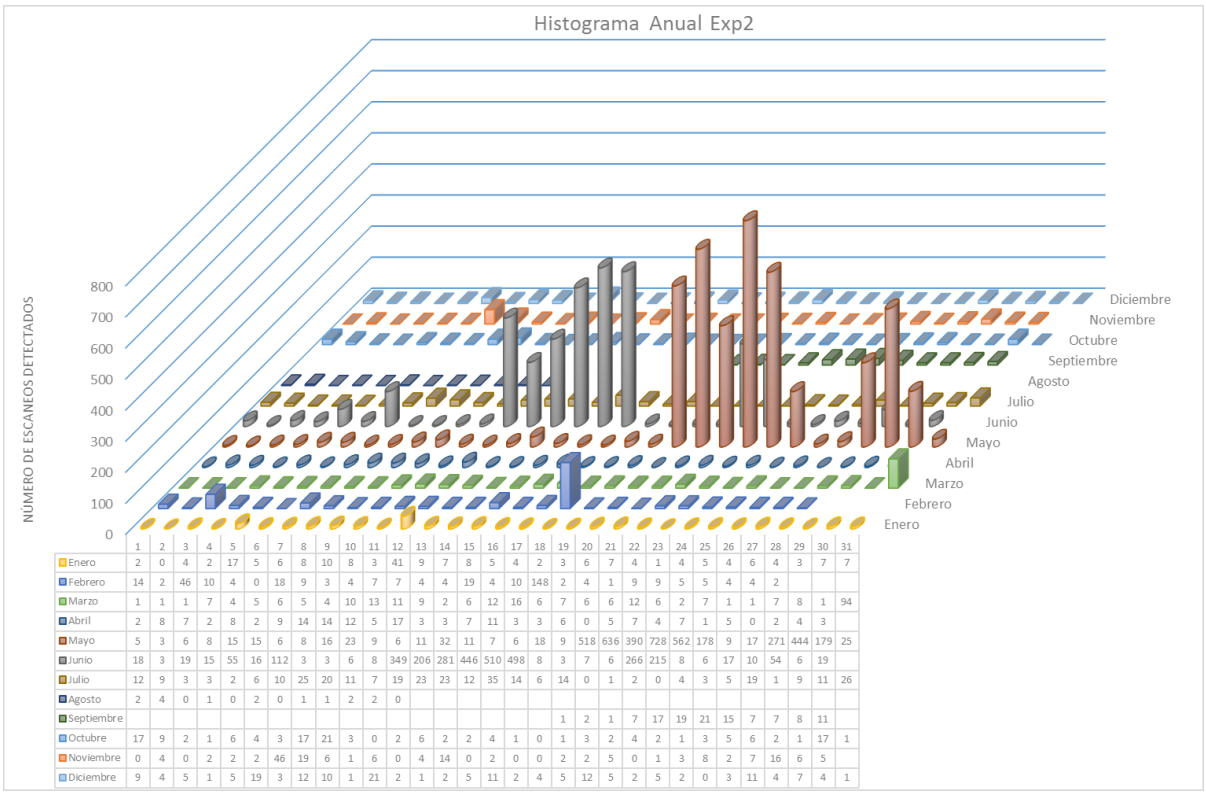
El siguiente máximo se observa en un día aislado en el mes de diciembre, de 7 escaneos detectados en dicho día. Entre ambos máximos de escaneos detectados/día representan un 30 % del total de escaneos detectados en un periodo bimensual (alrededor de un 15% del total mensualmente).

A partir de estos dos casos particulares tenemos otra serie de conclusiones a resaltar.

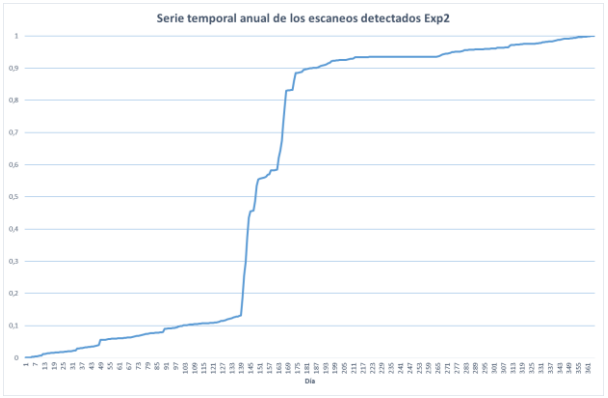
La primera de ellas es acerca del tipo de comportamiento que presenta el resto de escaneos vistos durante un periodo de 10 meses. Presentan una distribución uniforme de aparición que contiene un 70 % del total de escaneos vistos en el periodo anual (un 7% del total a nivel de mes). Como vemos, respecto a estos dos casos aislados citados anteriormente, corresponde a una mitad de concentración del número de escaneos detectados.

Respecto a la serie temporal, implica un crecimiento constante de pendiente un 0.05 por mes, teniendo en algunos casos de tendencia constante (que corresponden a periodos de poca actividad en la red). Estos periodos corresponden a los meses de abril, julio, agosto y septiembre.

5.2.2 Exportador 2



	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
Enero	2	0	4	2	17	5	6	8	10	8	3	41	9	7	8	5	4	2	3	6	7	4	1	4	5	4	6	4	3	7	7
Febrero	14	2	46	10	4	0	18	9	3	4	7	7	4	4	19	4	10	148	2	4	1	9	9	5	5	4	4	2			
Marzo	1	1	1	7	4	5	6	5	4	10	13	11	9	2	6	12	16	6	7	6	6	12	6	2	7	1	1	7	8	1	94
Abril	2	8	7	2	8	2	9	14	14	12	5	17	3	3	7	11	3	3	6	0	5	7	4	7	1	5	0	2	4	3	
Mayo	5	3	6	8	15	15	6	8	16	23	9	6	11	32	11	7	6	18	9	518	636	390	728	562	178	9	17	271	444	179	25
Junio	18	3	19	15	55	16	112	3	3	6	8	349	206	281	446	510	498	8	3	7	6	266	215	8	6	17	10	54	6	19	
Julio	12	9	3	3	2	6	10	25	20	11	7	19	23	23	12	35	14	6	14	0	1	2	0	4	3	5	19	1	9	11	26
Agosto	2	4	0	1	0	2	0	1	1	2	2	0																			
Septiembre																			1	2	1	7	17	19	21	15	7	7	8	11	
Octubre	17	9	2	1	6	4	3	17	21	3	0	2	6	2	2	4	1	0	1	3	2	4	2	1	3	5	6	2	1	17	1
Noviembre	0	4	0	2	2	2	46	19	6	1	6	0	4	14	0	2	0	0	2	2	5	0	1	3	8	2	7	16	6	5	
Diciembre	9	4	5	1	5	19	3	12	10	1	21	2	1	2	5	11	2	4	5	12	5	2	5	2	0	3	11	4	7	4	1



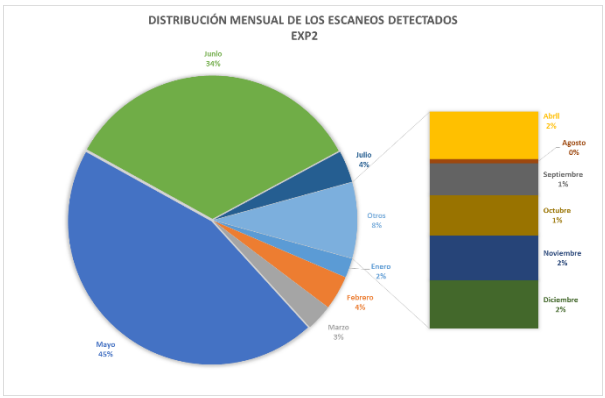


Ilustración 24: Histograma anual *(centro)*, serie temporal anual *(abajo-izquierda)* y distribución a nivel mensual *(abajo-derecha)* del número de escaneos detectados del exportador número 2.

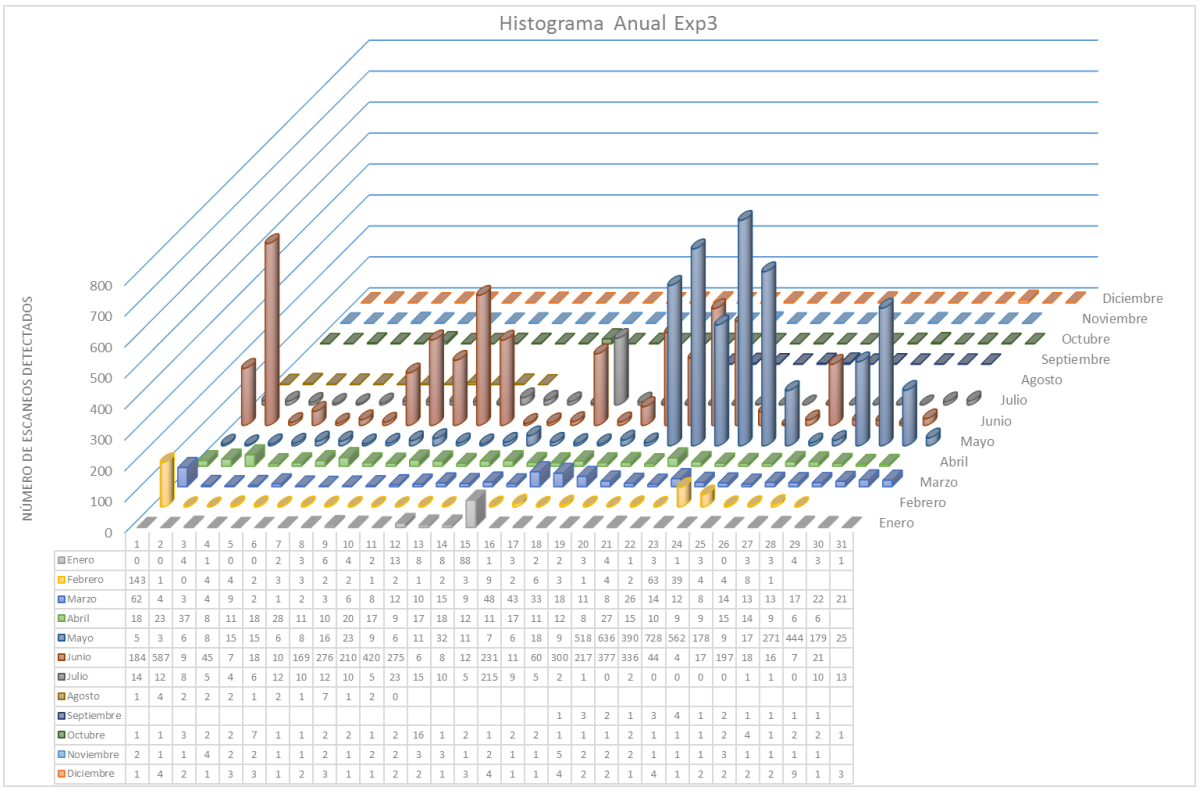
La gráfica del histograma anual para el exportador 2 presenta lo siguiente:

La mayor concentración de escaneos vistos se encuentra en el periodo bimensual de mayo-junio, con valores por encima de la centena de escaneos detectados por día. Algunos días aislados superan incluso los 500 escaneos detectados/día. Este periodo de dos meses contiene (viendo la distribución mensual) alrededor del 80 % del total de escaneos vistos en el periodo anual de análisis. Representa el mayor crecimiento en la serie temporal de 0.75 en dos meses o, de otra forma, un crecimiento del 0.01 diario.

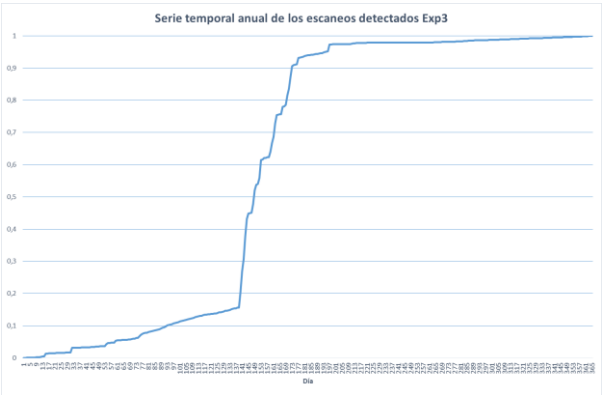
El 20 % restante aparece distribuido de forma uniforme a lo largo del resto del año del estudio. Esto se plasma en la serie temporal como un crecimiento constante de un 0.25 mensualmente y un 0.00083 diario. Diez veces menor que el crecimiento visto diario en el periodo bimensual de mayor concentración de barridos de puertos detectado en este exporter número 2. Al contrario que en el exportador previo, aparecen meses de no actividad como ocurre en el mes de agosto. Se sitúan los escaneos detectados/día, viendo el histograma por debajo de los 100 escaneos/día.

Presenta notables diferencias y alguna similitud respecto al caso anterior. La diferencia principal es que en el caso anterior se concentraban los escaneos en pocos días aunque en diferentes proporciones y en el segundo caso, existe una concentración más uniforme en el periodo de máxima concentración. Otra diferencia representativa existe en el periodo de concentración que, por una parte, son diferentes meses del año de estudio y por la otra, es diferente el número de meses, de máxima concentración, siendo de un periodo cuatrimestral en el caso anterior mientras que en este colector es de un periodo bimensual.

5.2.3 Exportador 3



	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
Enero	0	0	4	1	0	0	2	3	6	4	2	13	8	8	88	1	3	2	2	3	4	1	3	1	3	0	3	3	4	3	1
Febrero	143	1	0	4	4	2	3	3	2	2	1	2	1	2	3	9	2	6	3	1	4	2	63	39	4	4	8	1			
Marzo	62	4	3	4	9	2	1	2	3	6	8	12	10	15	9	48	43	33	18	11	8	26	14	12	8	14	13	13	17	22	21
Abril	18	23	37	8	11	18	28	11	10	20	17	9	17	18	12	11	17	11	12	8	27	15	10	9	9	15	14	9	6	6	
Mayo	5	3	6	8	15	15	6	8	16	23	9	6	11	32	11	7	6	18	9	518	636	390	728	562	178	9	17	271	444	179	25
Junio	184	587	9	45	7	18	10	169	276	210	420	275	6	8	12	231	11	60	300	217	377	336	44	4	17	197	18	16	7	21	
Julio	14	12	8	5	4	6	12	10	12	10	5	23	15	10	5	215	9	5	2	1	0	2	0	0	0	0	1	1	0	10	13
Agosto	1	4	2	2	2	1	2	1	7	1	2	0																			
Septiembre																			1	3	2	1	3	4	1	2	1	1	1	1	
Octubre	1	1	3	2	2	7	1	1	2	2	1	2	16	1	2	1	2	2	1	1	1	2	1	1	1	2	4	1	2	2	1
Noviembre	2	1	1	4	2	2	1	1	2	1	2	2	3	3	1	2	1	1	5	2	2	2	1	1	1	3	1	1	1	1	
Diciembre	1	4	2	1	3	3	1	2	3	1	1	2	2	1	3	4	1	1	4	2	2	1	4	1	2	2	2	2	9	1	3



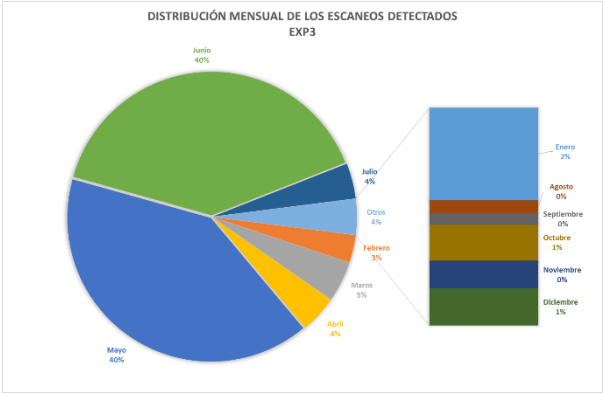


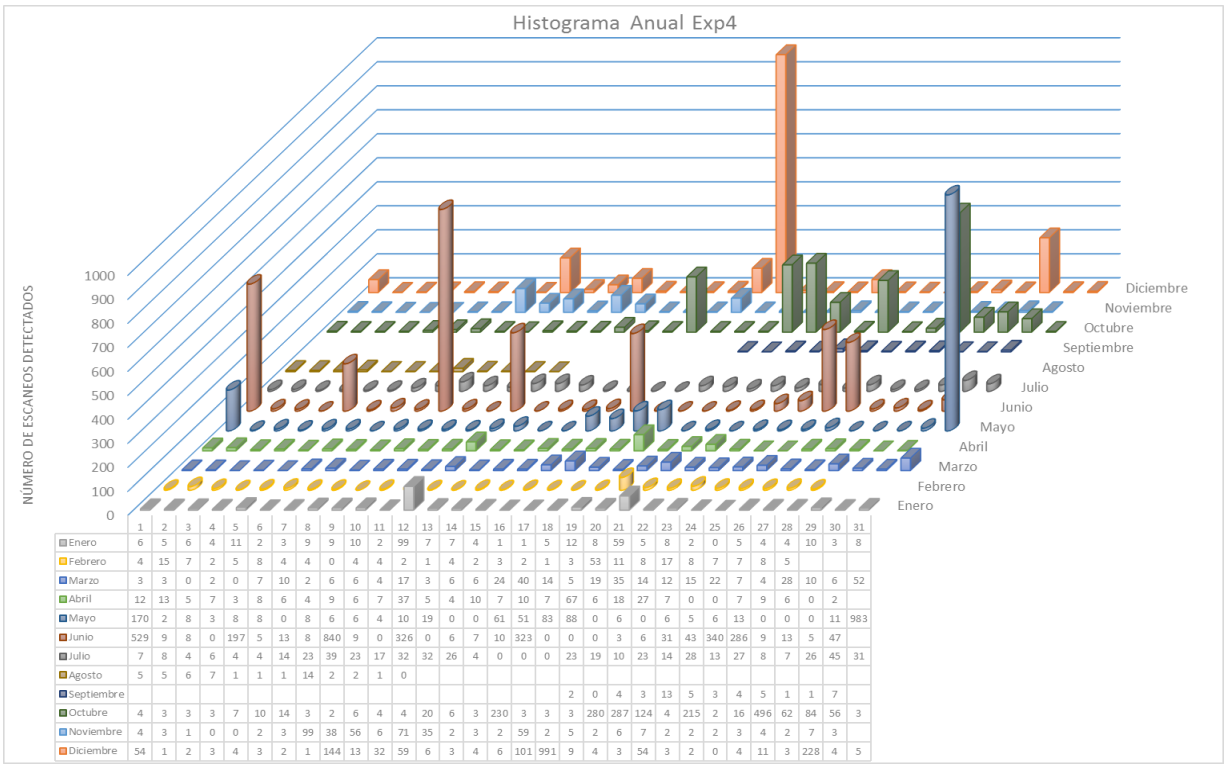
Ilustración 25: Histograma anual *(centro)*, serie temporal anual *(abajo-izquierda)* y distribución a nivel mensual *(abajo-derecha)* del número de escaneos detectados del exportador número 3.

Mediante la gráfica del histograma anual presenta la mayor concentración de escaneos detectados vuelve a situarse, como en el exportador anterior, en el periodo bimensual de mayo-junio, con valores por encima de la centena de escaneos detectados por día. Algunos días aislados superan incluso los 600 escaneos detectados/día. Este periodo de dos meses contiene (viendo la distribución mensual) alrededor del 80 % del total de escaneos vistos en el periodo anual de análisis. Representa el mayor crecimiento en la serie temporal, de 0.8 en dos meses o, de otra forma, un crecimiento del 0.013 diario.

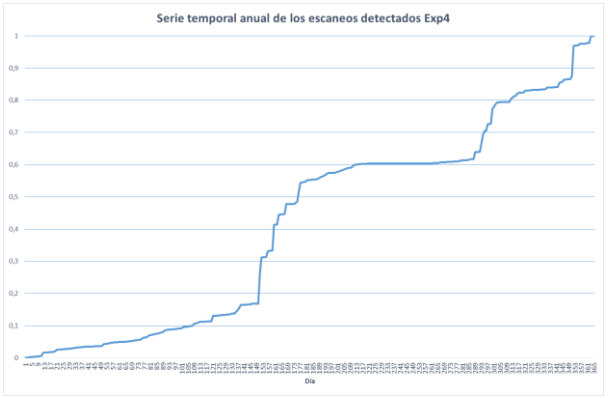
El 20 % restante aparece distribuido de forma uniforme a lo largo del resto del año del estudio. Presenta alguna serie de máximos en días aislados como ocurre en el primer día de febrero o en el 15 de enero llegando a la centena de escaneos/día e incluso superándola. Esto se plasma en la serie temporal como un crecimiento constante de un 0.25 en el resto de meses y un 0.001 diario. Diez veces menor que el crecimiento visto diario en el periodo bimensual de mayor concentración de barridos de puertos detectado en este tercer exporter. Al contrario que en el exportador previo, aparecen meses de no actividad como ocurre en los meses de agosto, septiembre y noviembre. Se sitúan los escaneos detectados/día, viendo el histograma en valores próximo a los 100 escaneos/día y concentrado en pocos días.

Presenta un comportamiento similar, salvando las proporciones de tráfico recolectado, respecto al exportador anterior, además de diferencias y alguna similitud respecto al primer exportador. La diferencia principal es que en el primer caso se concentraban los escaneos en pocos días aunque en diferentes proporciones y en este caso, existe una concentración más uniforme en el periodo de máxima concentración. Otra diferencia representativa que existe en el periodo de concentración es que, por una parte, son diferentes meses del año de estudio y por la otra, es diferente el número de meses, de máxima concentración, siendo de un periodo cuatrimestral en el primer caso frente a un periodo bimensual de este exportador.

5.2.4 Exportador 4



	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
Enero	6	5	6	4	11	2	3	9	9	10	2	99	7	7	4	1	1	5	12	8	59	5	8	2	0	5	4	4	10	3	8
Febrero	4	15	7	2	5	8	4	4	0	4	4	2	1	4	2	3	2	1	3	53	11	8	17	8	7	7	8	5			
Marzo	3	3	0	2	0	7	10	2	6	6	4	17	3	6	6	24	40	14	5	19	35	14	12	15	22	7	4	28	10	6	52
Abril	12	13	5	7	3	8	6	4	9	6	7	37	5	4	10	7	10	7	67	6	18	27	7	0	0	7	9	6	0	2	
Mayo	170	2	8	3	8	8	0	8	6	6	4	10	19	0	0	61	51	83	88	0	6	0	6	5	6	13	0	0	0	11	983
Junio	529	9	8	0	197	5	13	8	840	9	0	326	0	6	7	10	323	0	0	0	3	6	31	43	340	286	9	13	5	47	
Julio	7	8	4	6	4	4	14	23	39	23	17	32	32	26	4	0	0	0	23	19	10	23	14	28	13	27	8	7	26	45	31
Agosto	5	5	6	7	1	1	1	14	2	2	1	0																			
Septiembre																			2	0	4	3	13	5	3	4	5	1	1	7	
Octubre	4	3	3	3	7	10	14	3	2	6	4	4	20	6	3	230	3	3	3	280	287	124	4	215	2	16	496	62	84	56	3
Noviembre	4	3	1	0	0	2	3	99	38	56	6	71	35	2	3	2	59	2	5	2	6	7	2	2	2	3	4	2	7	3	
Diciembre	54	1	2	3	4	3	2	1	144	13	32	59	6	3	4	6	101	991	9	4	3	54	3	2	0	4	11	3	228	4	5



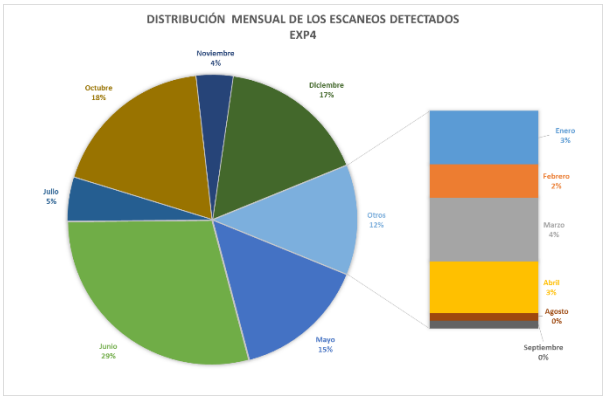


Ilustración 26: Histograma anual *(centro)*, serie temporal anual *(abajo-izquierda)* y distribución a nivel mensual *(abajo-derecha)* del número de escaneos detectados del exportador número 4.

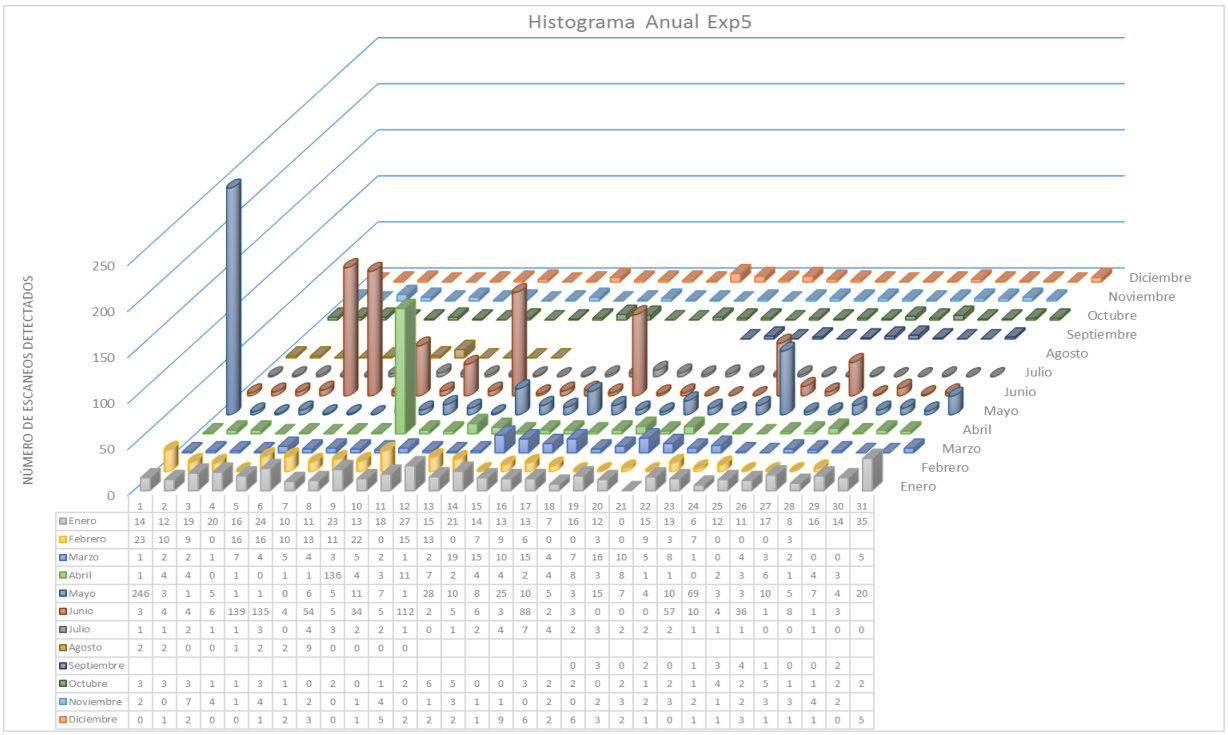
La grafica del histograma anual presenta la mayor concentración en un periodo cuatrimestral (meses de mayo, junio, octubre y diciembre), situando un 80% del total de escaneos detectados totales, vistos en la distribución mensual. Se observa el mayor crecimiento en la serie temporal de 0.7 en los cuatro meses, 0.005 diario. Ocurren máximos de hasta el millar de escaneos detectados/día. Al contrario que los exportadores anteriormente analizados, tenemos dicha concentración máxima en días puntuales de los cuatro meses que hemos tomado.

La distribución restante aparece de forma uniforme a lo largo del resto del año del estudio. Presenta algunos máximos en días aislados como ocurre en el último día de mayo llegando al millar de escaneos detectados/día pero el resto de días ni alcanzando la centena de escaneos/día. Esto se plasma en la serie temporal como un crecimiento constante de un 0.3 en el resto de meses y un 0.001 diario. Cinco veces menor que el crecimiento diario visto en el periodo cuatrimestral de mayor concentración de barridos de puertos detectado. Al contrario que en el exportador previo, aparecen meses de no actividad como ocurre en el mes de agosto y septiembre.

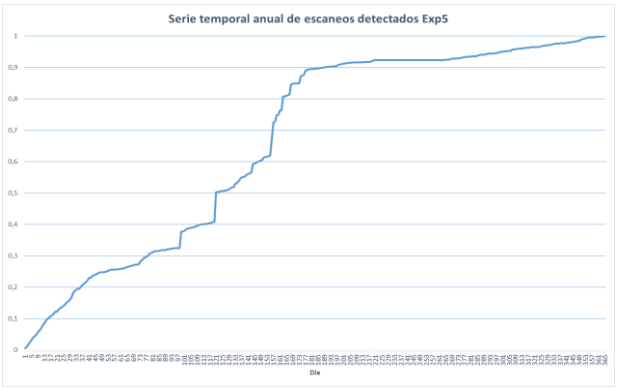
Frente a los exportadores anteriores, presenta diferencias y similitudes reseñables:

La primera diferencia respecto al tercero y segundo es en el periodo de máxima concentración (bimensual frente a cuatrimestral). Coincide en periodo con el primer caso pero se concentra en diferentes meses. La segunda es en cuanto al máximo de escaneos siendo parecido al caso anterior y diferente en proporción al resto de casos expuestos anteriormente.

5.2.4 Exportador 5



	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
Enero	14	12	19	20	16	24	10	11	23	13	18	27	15	21	14	13	13	7	16	12	0	15	13	6	12	11	17	8	16	14	35
Febrero	23	10	9	0	16	16	10	13	11	22	0	15	13	0	7	9	6	0	0	3	0	9	3	7	0	0	0	3			
Marzo	1	2	2	1	7	4	5	4	3	5	2	1	2	19	15	10	15	4	7	16	10	5	8	1	0	4	3	2	0	0	5
Abril	1	4	4	0	1	0	1	1	136	4	3	11	7	2	4	4	2	4	8	3	8	1	1	0	2	3	6	1	4	3	
Mayo	246	3	1	5	1	1	0	6	5	11	7	1	28	10	8	25	10	5	3	15	7	4	10	69	3	3	10	5	7	4	20
Junio	3	4	4	6	139	135	4	54	5	34	5	112	2	5	6	3	88	2	3	0	0	0	57	10	4	36	1	8	1	3	
Julio	1	1	2	1	1	3	0	4	3	2	2	1	0	1	2	4	7	4	2	3	2	2	2	1	1	1	0	0	1	0	0
Agosto	2	2	0	0	1	2	2	9	0	0	0	0																			
Septiembre																			0	3	0	2	0	1	3	4	1	0	0	2	
Octubre	3	3	3	1	1	3	1	0	2	0	1	2	6	5	0	0	3	2	2	0	2	1	2	1	4	2	5	1	1	2	2
Noviembre	2	0	7	4	1	4	1	2	0	1	4	0	1	3	1	1	0	2	0	2	3	2	3	2	1	2	3	3	4	2	
Diciembre	0	1	2	0	0	1	2	3	0	1	5	2	2	2	1	9	6	2	6	3	2	1	0	1	1	3	1	1	1	0	5



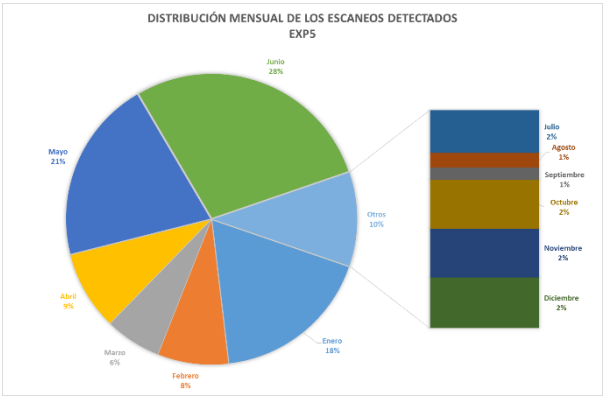


Ilustración 27: Histograma anual *(centro)*, serie temporal anual *(abajo-izquierda)* y distribución a nivel mensual *(abajo-derecha)* del número de escaneos detectados del exportador número 5.

Mediante la observación tanto en el histograma anual como en la distribución mensual, se encuentra en tres meses la mayor concentración de escaneos detectados en este exporter (alrededor del 70% del total de escaneos detectados). Los meses son los de enero, mayo y junio. En el mes de enero tenemos una distribución uniforme diaria a lo largo del mes mientras que en los otros dos meses mencionados, presentan una serie de picos de escaneos cercanos a los 250 escaneos/día. Coincide en la serie temporal con los crecimientos de dicha serie.

Otro mes a reseñar, a vista del histograma anual, es el mes de abril. En él, se ofrece un máximo de 100 escaneos en un único día.

En el resto del total un 20% se distribuye de forma más uniforme en el resto de meses del año de estudio. Al contrario que se ha visto en otros colectores analizados no presenta meses de no actividad en dicha red.

5.3 Ubicación Origen-Destino de los escaneos detectados

En esta sección del capítulo se busca dar una localización tanto a la dirección IP origen como a la dirección IP destino implicadas en cada uno de los escaneos detectados mediante nuestro sistema, a nivel de exporter de red.

Una de las dos gráficas que se presentan busca una clasificación realizada por nosotros acorde a estos parámetros (Gráficas de Clasificación):

- Escaneos de origen en España hacia destinos dentro de España (ESP-ESP).
- Escaneos de origen en España hacia el extranjero (ESP-EXT).
- Escaneos de origen en algún país fuera de España hacia un destino dentro de España (EXT-ESP).

En la otra gráfica se muestra una mayor precisión en la localización del origen-destino implicado en el barrido detectado, a nivel concreto del país origen y país destino (Gráficas de Localización).

5.3.1 Exportador 1

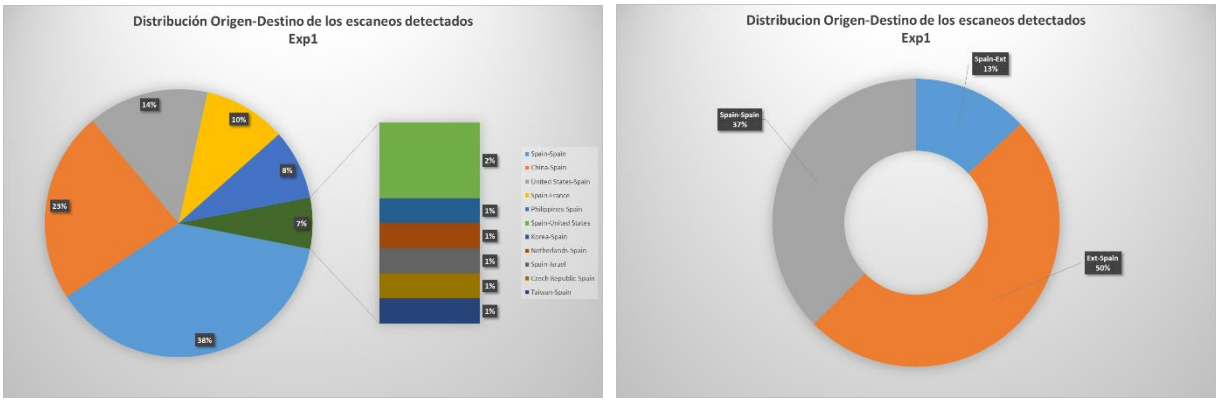


Ilustración 28: Gráfica de Localización (izquierda) y Gráfica de Clasificación (izquierda) para el exportador 1

Mediante la *Gráfica de Clasificación* se destacan las siguientes conclusiones.

La mitad de los escaneos totales detectados para este exporter pertenecen al grupo 3 (EXT-ESP) descrito anteriormente, es decir, provienen de equipos ubicados fuera de España hacia equipos situados dentro del territorio español.

La otra mitad restante se reparten entre los otros dos grupos mencionados anteriormente (ESP-ESP y ESP-EXT). Los barridos entre equipos dentro de España predominan casi 3 veces más, en proporción, respecto a escaneos entre equipos dentro de España hacia el extranjero.

Acudiendo a la Gráfica de Localización se reseñan las siguientes conclusiones.

Tenemos una máxima proporción del 38% del total entre equipos dentro de España (ESP-ESP). Como segundo máximo aparecen orígenes de países extranjeros, como China, Estados Unidos e Islas Filipinas, hacia destinos dentro de España (EXT-ESP). Componen un 47% del 50 % que tenemos de la *Gráfica de Clasificación*, pertenecientes a este perfil. Esto nos determina que el 95% de los escaneos pertenecientes a este grupo viene de estos tres destinos citados anteriormente.

Otra característica importante que tenemos sobre este *Grafico de Localización* es acerca de los escaneos de origen España hacia el exterior (ESP-EXT). Tenemos mayoritariamente destinos como Francia con un 10% del 13% total visto en este grupo, es decir, un 77 % de los escaneos que se hacen desde España van con destino algún equipo de Francia.

5.3.2 Exportador 2

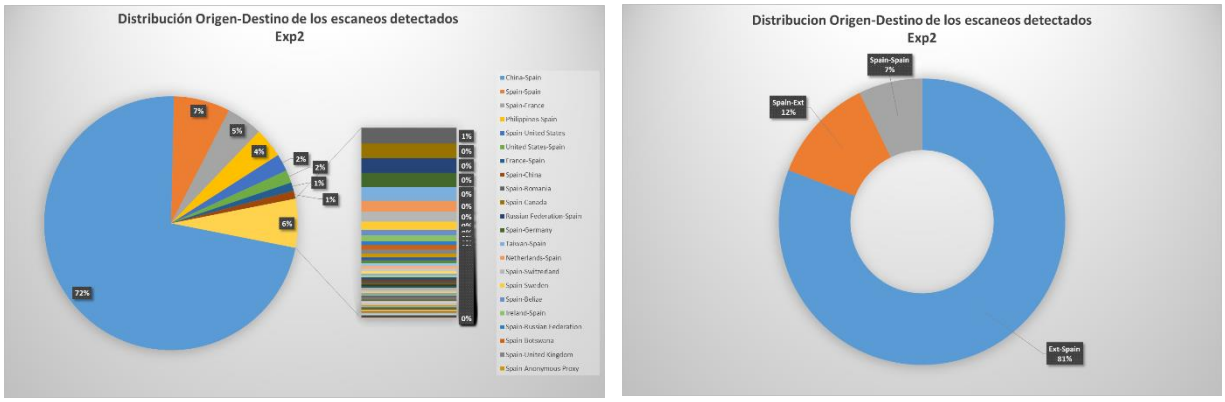


Ilustración 29: Gráfica de Localización (izquierda) y Gráfica de Clasificación (izquierda) para el exportador 2

Por medio de la *Gráfica de Clasificación* tenemos lo siguiente:

Un 80 % del total de escaneos detectados por el sistema provienen de origen extranjero hacia España (EXT- ESP). De forma mayoritaria un 72 % de las $^{4}/_{5}$ partes pertenecientes a este tipo se dan como origen China con destino España. Otros orígenes de posibles escaneos de importancia son Filipinas y Estados Unidos con un 4 y 2 % respectivamente. En otras palabras, un 97.5 % de los escaneos del tipo EXT-ESP son hacia estos tres destinos antes mencionados.

Del resto, un 20% se distribuyen entre los otros dos tipos definidos anteriormente (ESP-ESP y ESP-EXT). Predominan los barridos con origen España hacia países extranjeros frente a los escaneos entre origen y destino nacional (un 12 y 7 % respectivamente).

Este 7 % de escaneos a nivel nacional se ve reflejado en el *Grafico de Localización* como segundo máximo.

Los principales objetivos elegidos por orígenes españoles son Francia y Estados Unidos. De forma conjunta hacen un 7% del 12% detectado en este tipo ESP-EXT.

5.3.3 Exportador 3

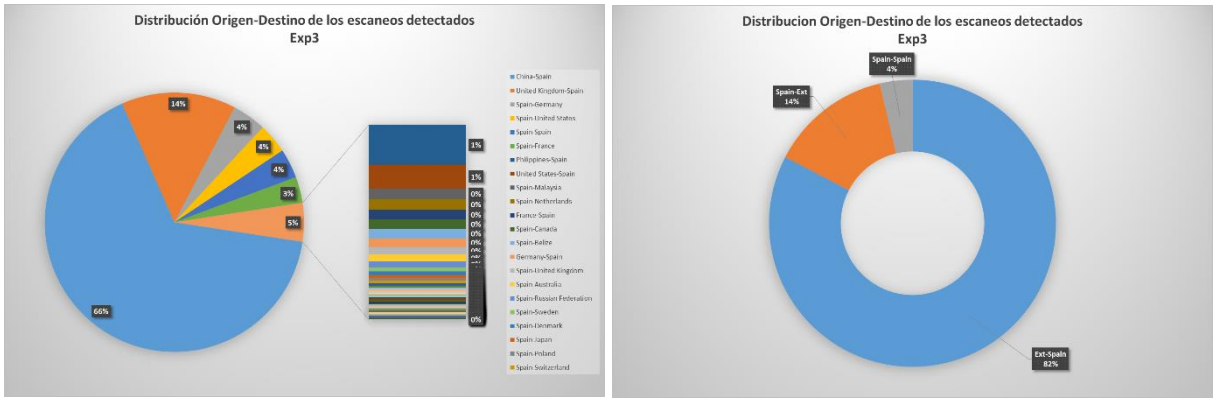


Ilustración 30: Gráfica de Localización (izquierda) y Gráfica de Clasificación (izquierda) para el exportador 3

Mirando la *gráfica de Clasificación* tenemos una cantidad superior al 80%, 82% de los escaneos totales, que corresponden a barridos de puertos realizados desde el extranjero hacia España (EXT-ESP).

Del resto de escaneos totales (un 18%) quedan distribuidos entre escaneos con origen y destino España (ESP-ESP) y barridos de puertos con origen España hacia el extranjero (ESP-EXT), con un 4 y un 14 % respectivamente.

Combinando estas conclusiones con la gráfica de Localización obtenemos lo siguiente:

Del 82 % de los escaneos totales, correspondientes al tipo EXT-ESP, tenemos que un 66% son barridos desde China hacia España. Además de ello, un 14 % corresponde a escaneos desde Reino Unido hacia España. Otros países que se ven involucrados en esta categoría son Filipinas y Estados Unidos, con un 1% cada uno.

Como tercer máximo tenemos el 4% citado del grupo ESP-ESP. Del 14 % visto para el tipo ESP-EXT se tienen como objetivos de posibles escaneos Francia (3%), Alemania (4%) y Estados Unidos (4%).

5.3.4 Exportador 4

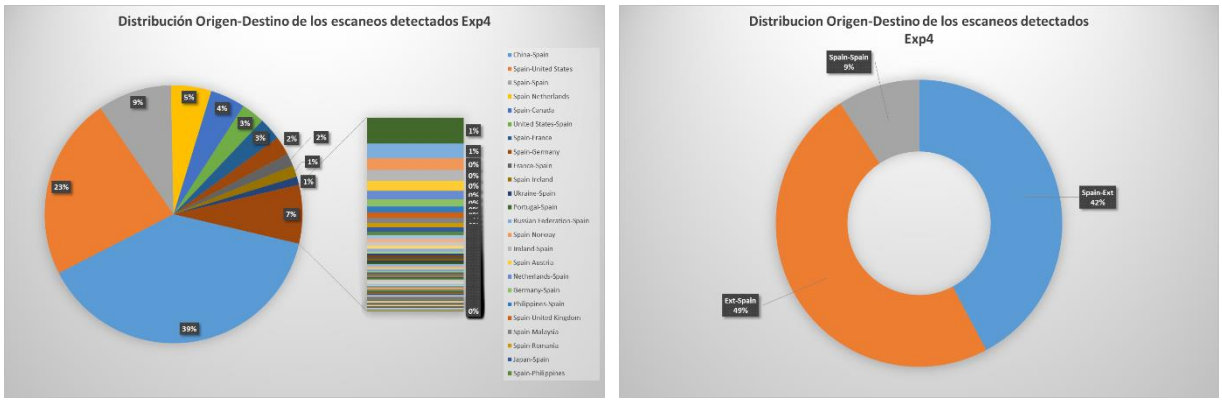


Ilustración 31: Gráfica de Localización (izquierda) y Gráfica de Clasificación (izquierda) para el exportador 4

Observando la gráfica de Clasificación tenemos una misma proporción entre escaneos desde el extranjero hacia España (EXT-ESP) y barridos entre España hacia el extranjero (ESP-EXT) con un 49 y un 42 % respectivamente.

Con un 9 % se destina a escaneos entre origen y destino nacional (ESP-ESP).

Con estos resultados descritos anteriormente e inspeccionando la gráfica de Localización se concluye lo siguiente.

Como principales orígenes de escaneos con destino España (EXT-ESP) tenemos China (39 %), Estados Unidos (3%), Francia (2%), Ucrania (1%) y Portugal (1%). Entre ellos suman un total de 46% de un 49% visto para este tipo.

Tenemos como tercera máxima proporción de la gráfica el 9% de escaneos entre origen y destino nacional (ESP-ESP).

Como principales destinos de escaneos con origen España (ESP-EXT) tenemos Estados Unidos (23%), Países Bajos (5%), Canadá (4%), Francia (3%), Alemania (2%) e Irlanda (1%). Entre ellos suman un total de 38% de un 42 % visto para este tipo de barridos.

5.3.5 Exportador 5

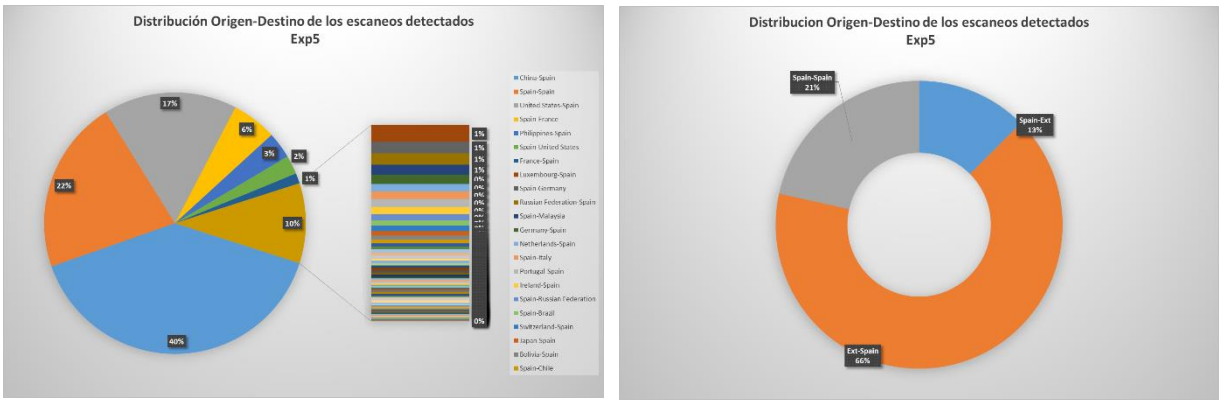


Ilustración 32: Gráfica de Localización (izquierda) y Gráfica de Clasificación (izquierda) para el exportador 5

Por medio de la gráfica de Clasificación concluimos lo siguiente:

Las $^{2}/_{3}$ partes de los escaneos detectados para este exporter vienen de orígenes extranjeros hacia equipos dentro de España (EXT-ESP).

Del resto de escaneos, predominan aquellos con origen y destino España (ESP-ESP) respecto al otro tipo comentado (ESP-EXT) en 1.7 veces superior, en proporción.

Por los datos expuestos en la Gráfica de Localización tenemos que:

Un 40% de los escaneos totales detectados se realizaron entre origen China y destino España (EXT-ESP). Como segundo dato a destacar es el 22% de los escaneos totales detectados entre origen y destino español (ESP-ESP).

Otra conclusión a reseñar es que orígenes como China, Estados Unidos y Filipinas hacia destino España (EXT-ESP) componen un 60% de un 66% visto anteriormente destinado para este tipo.

Como último punto destacar que los barridos de puertos realizados desde España hacia el extranjero (ESP-EXT) lo conforman, en su mayoría, destinos como Francia y Estados Unidos. Componen un 8% del 13 % total destinado a este grupo de clasificación.

5.4 Analisis de los puertos empleados en los escaneos detectados

Después de aportar una vista localizada tanto en tiempo para poder situar los escaneos en el año de estudio elegido. Estos resultados permiten conocer cómo se han ido sucediendo los posibles escaneos detectados en el sistema que se plantea.

Otra vista que se aporta en el capítulo es una ubicación de los mismos. Esto nos posibilita el conocimiento tanto de los países que escanean equipos de España (EXT-ESP) como aquellos destinos involucrados en barridos de puertos de origen España (ESP-EXT). A su vez, en los resultados, obtuvimos el tercer tipo mencionado que son escaneos entre origen y destino España.

Una vez expuestas ambas visiones temporal y localizada de los escaneos de puertos detectados, se pretende ofrecer una visión a nivel de puerto implicado en estos escaneos vistos. Mediante el IANA [69] (*Internet Assigned Numbers Authority*) podemos establecer tres tipos asociados a puertos menores al 1024.

- Puertos conocidos (*Well Known Ports*): Son puertos asignados por el sistema operativos y se emplean en protocolos "bien conocidos" como por ejemplo HTTP (servidor Web), POP3/SMTP (servidor de e-mail) y Telnet. Si queremos usar uno de estos puertos tendremos que arrancar el servicio que los use teniendo permisos de administrador.

- Puertos reservados (*Reserved Ports*): Son aquellos puertos reservados para un uso futuro pero no tienen asignados una aplicación concreta.

- Puertos sin asignación (*Unassigned Ports*) Aquí se componen de aquellos números no asignados como puertos bien conocidos.

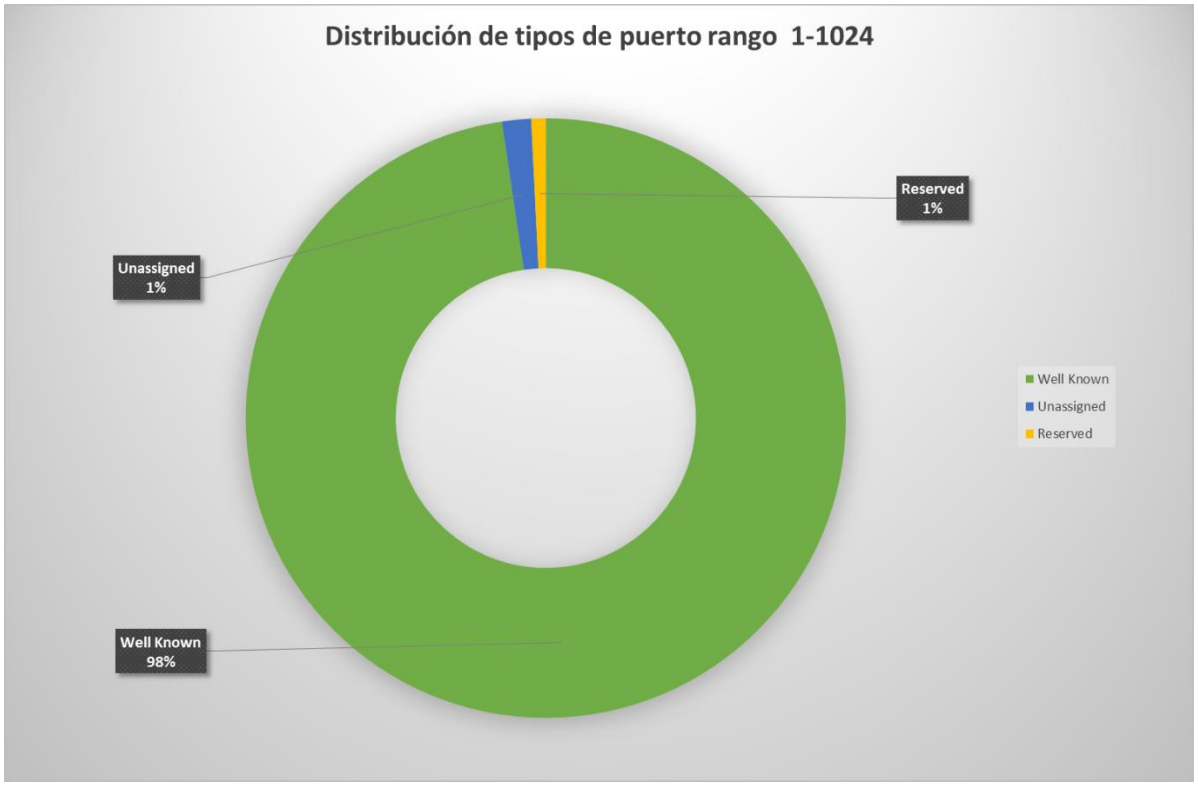


Ilustración 33: Distribución de los puertos 1 al 1024 [69]

Del resto de puertos, mayores al 1024 se clasifican en:

- Puertos Registrados (*Registered Ports*): Se componen en el rango desde el 1024 al 49151 y se pueden usar por cualquier aplicación.

- Puertos Dinámicos o Privados (*Private Ports*): se comprenden en el rango de 49152 al 65535. Se asignan a aplicaciones de clientes al inicio de la conexión. Se utilizan mayormente en conexiones P2P (*Peer to Peer*).

A continuación exponemos los resultados de los cinco exportadores de red analizados en el año 2013. La primera gráfica muestra una lista concreta del número de puerto a partir de un top 100 de puertos empleados en los escaneos. Nos permite ver aquellos números más relevantes empleados por parte de los atacantes. Mediante la otra gráfica vamos a valorar respecto a los tres tipos citados anteriormente. Esto nos permite conocer bajo que rango de puertos se emplea más en los escaneos detectados.

5.4.1 Exportador 1

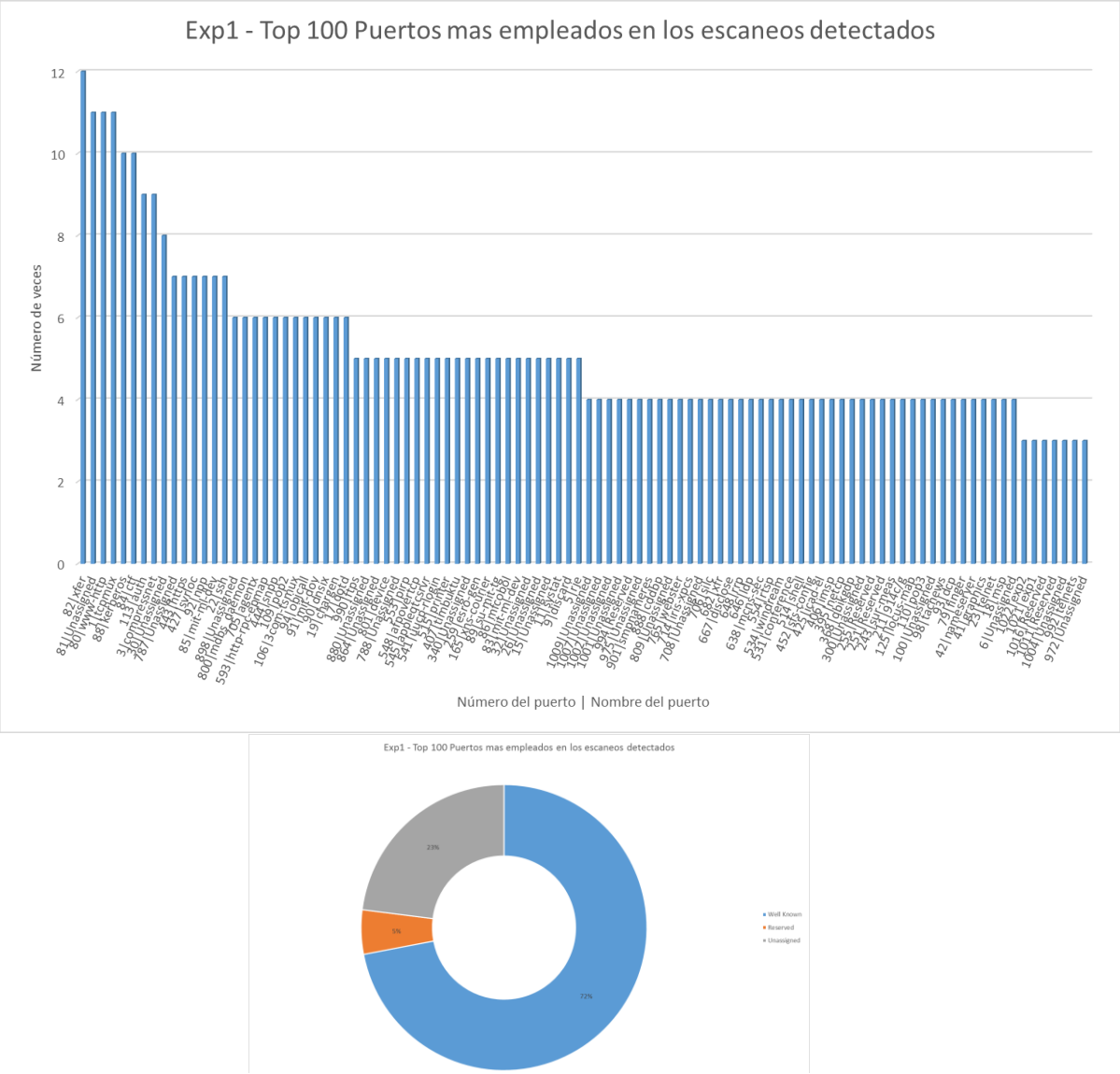


Ilustración 34: Top 100 de puertos más empleados en los escaneos detectados (arriba) clasificación del exporter bajo tipo de puerto (abajo) para el exportador 1.

En el gráfico de tarta tenemos una proporción alta de empleo de puertos bien conocidos y como segunda proporción alta los puertos sin asignación quedando en una contribución de solo el 5 % de puertos reservados.

Estos resultados se plasman claramente en el histograma. En él se presenta una mayoría distribuida entre puertos bien conocidos y puertos sin asignar. Fijándose en los primeros 10 de los 100 expuestos, tenemos proporciones similares a las vistas en el gráfico de tarta.

5.4.2 Exportador 2

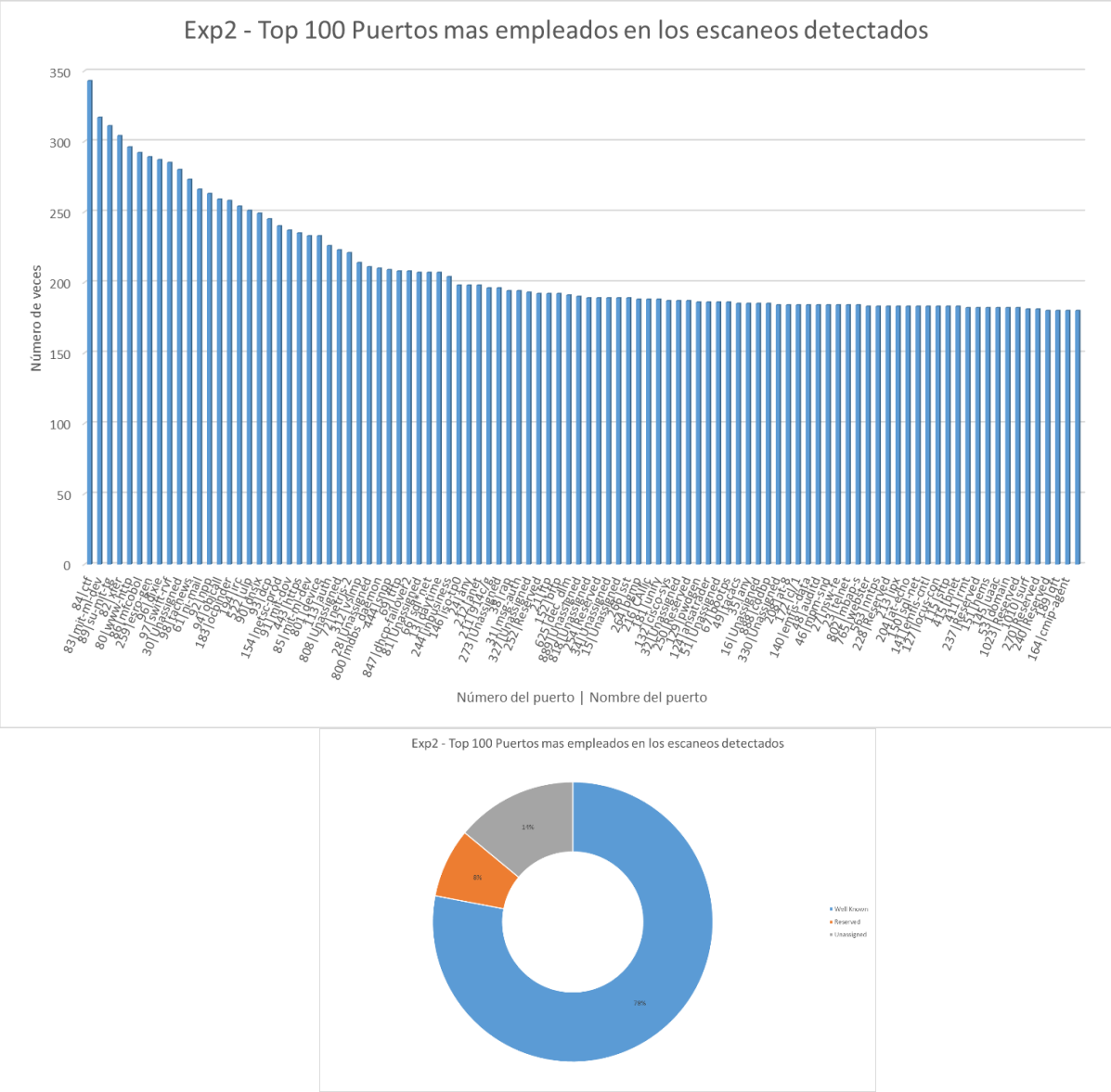
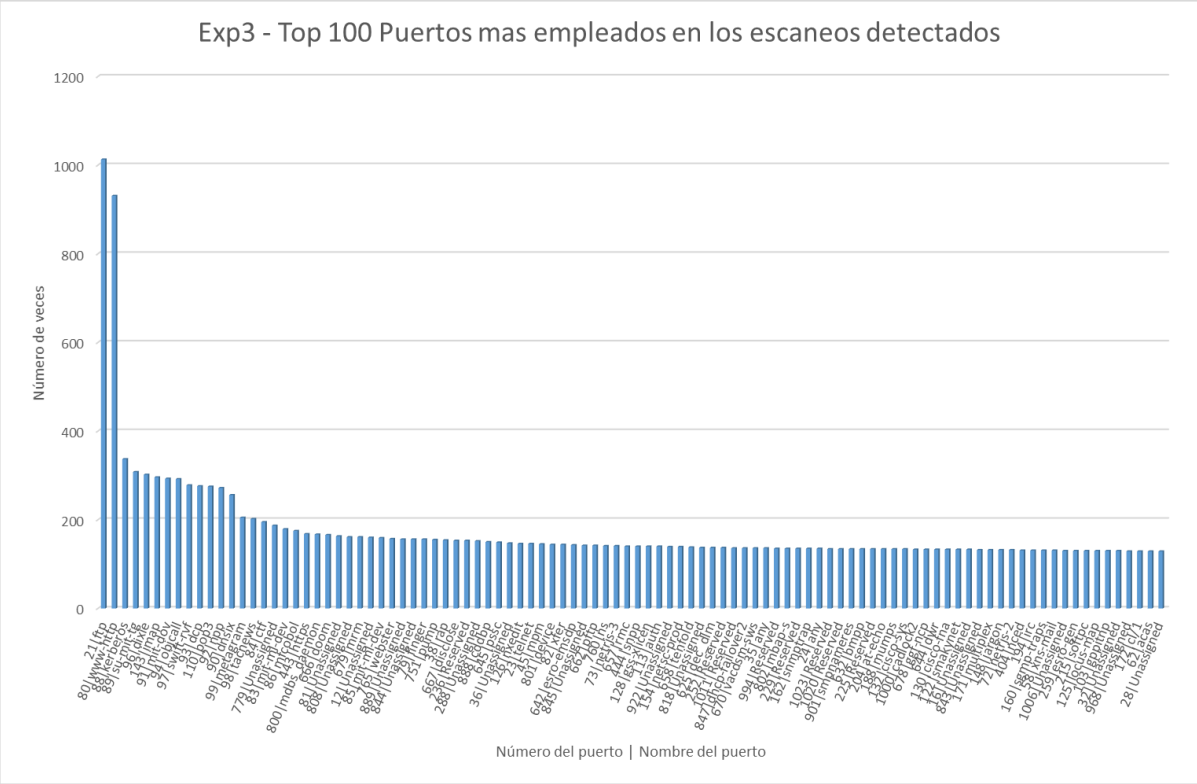


Ilustración 35: Top 100 de puertos más empleados en los escaneos detectados (arriba) clasificación del exporter bajo tipo de puerto (abajo) para el exportador 2.

A partir del gráfico de tarta tenemos una proporción cercana al 80 % de puertos bien conocidos. Con una proporción cuatro veces menor aparecen, como vimos para el anterior exportador, una proporción del 14 % para puertos sin asignación. Nos queda con un valor cercano al 10 % la contribución de los puertos reservados.

Alguna de las conclusiones planteadas anteriormente se observa también en el histograma adjunto. Con una predominancia en empleo de puertos bien conocidos seguido de puertos sin asignación. Para los puertos reservados su proporción es mínima pero tienen presencia como ocurre para el exporter anterior.

5.4.3 Exportador 3



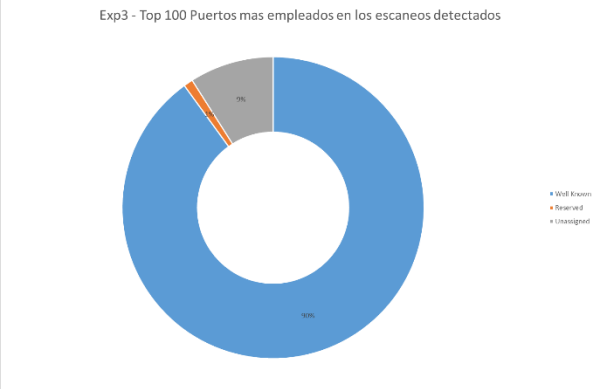


Ilustración 36: Top 100 de puertos más empleados en los escaneos detectados (arriba) clasificación del exporter bajo tipo de puerto (abajo) para el exportador 3.

Tanto en el histograma como en el gráfico de tarta se puede apreciar, para este exportador de red número 3, una predominancia muy notoria del empleo de puertos bien conocidos (valor del 90 % del total). Como segunda proporción a reseñar son los puertos sin asignar con un valor del 9%. Estos dos tipos de puertos son los más predominantes dejando con un valor del 1% a puertos reservados.

Estas mismas conclusiones se pueden trasladar al histograma. De forma mayoritaria aparece el uso de puertos bien conocidos junto a puertos sin asignación. De forma mínima tenemos contribuciones también del empleo de puertos reservados. Se comporta de la misma forma que los dos exporters anteriores pero posee diferente proporción para los tipos asociados.

5.4.4 Exportador 4

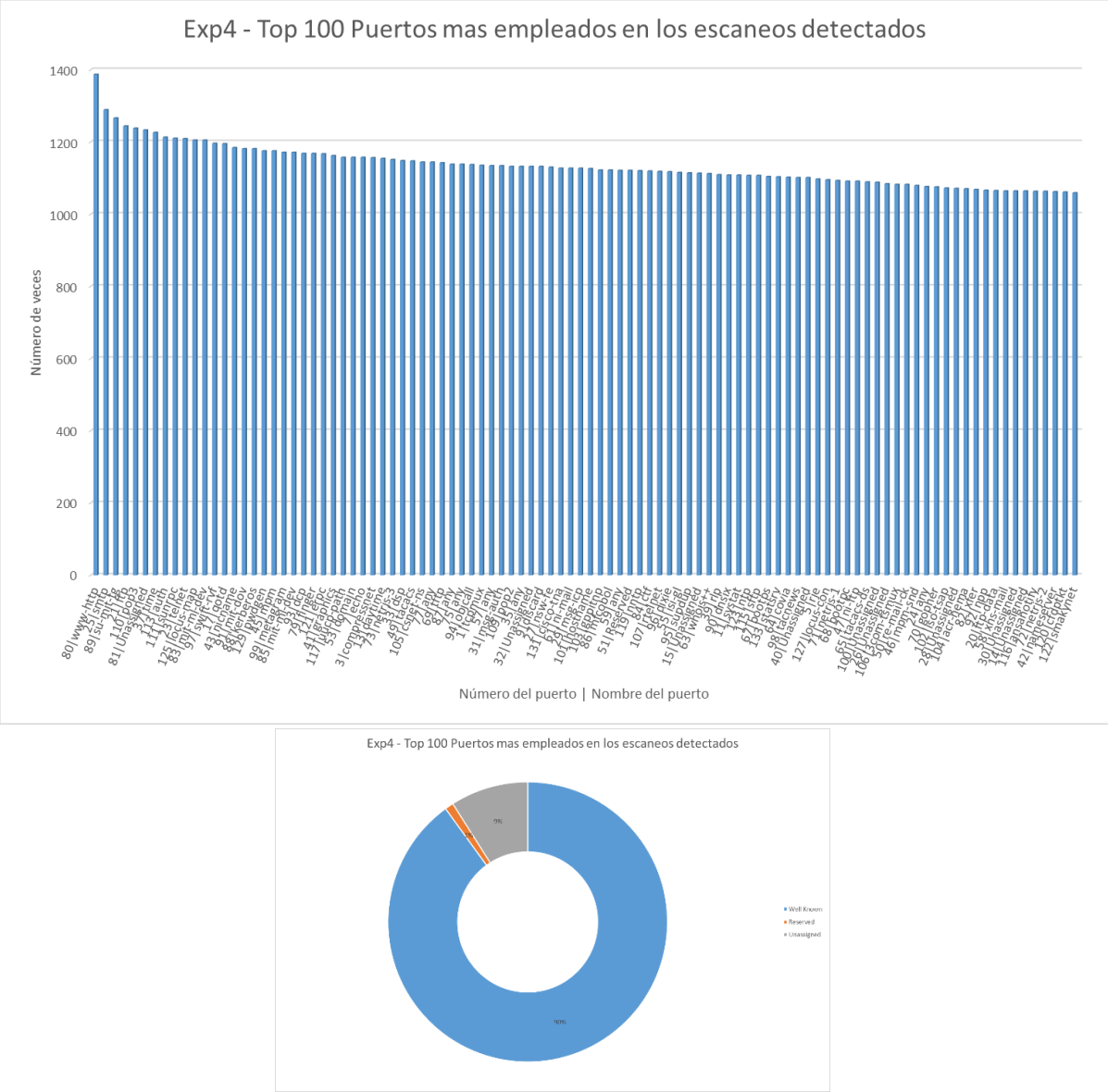


Ilustración 37: Top 100 de puertos más empleados en los escaneos detectados (arriba) clasificación del exporter bajo tipo de puerto (abajo) para el exportador 4.

Observando el grafico de tarta tenemos una alta proporción de empleo de puertos bien conocidos (90 %) seguido de una proporción nueve veces menor para puertos sin asignar (9%) para dejarnos con una mínima contribución de empleo de puertos reservados (1%). Todo ello con las mismas proporciones que en el exporter anterior

Mirando el anterior histograma y el adjunto se puede concluir que presentan mismo comportamientos en proporciones diferentes.

5.4.5 Exportador 5

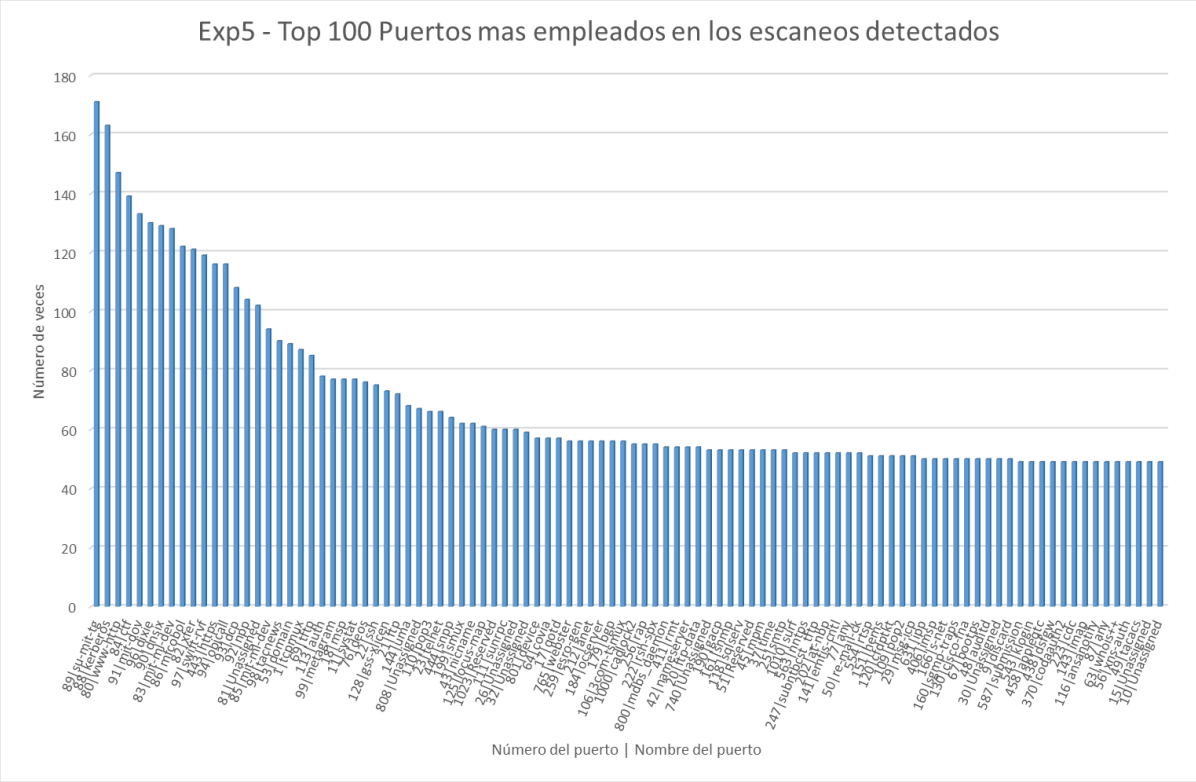


Ilustración 38: Top 100 de puertos más empleados en los escaneos detectados (arriba) clasificación del exporter bajo tipo de puerto (abajo) para el exportador 5.

En este exportador presenta misma proporción que los dos exporters vistos anteriormente para el empleo de puertos bien conocidos, con un 90 %. Respecto a las otras proporciones cambia para aumentar muy ligeramente el empleo de puertos reservados al 2% dejando a la proporción de puertos sin asignar con un 8% de uso.

Mirando el histograma se observa una menor proporción respecto a los dos exportadores anteriores pero poseen un comportamiento similar.

5.5 Conclusiones globales del estudio

Anteriormente se han expuesto una serie de conclusiones a nivel de exportador de red, de una forma particular. En esta sección se pretende ofrecer misma visión temporal, localizada y categorizada de los exportadores de red a nivel más general. En primer lugar y siguiendo el esquema propuesto en la particularización, ofreceremos una visión temporal. Se sigue con una visión localizada global para finalizar con la categorizada. Como último punto de esta sección, ofreceremos una tabla resumen con algunas características adicionales no citadas anteriormente.

5.5.1 Visión temporal

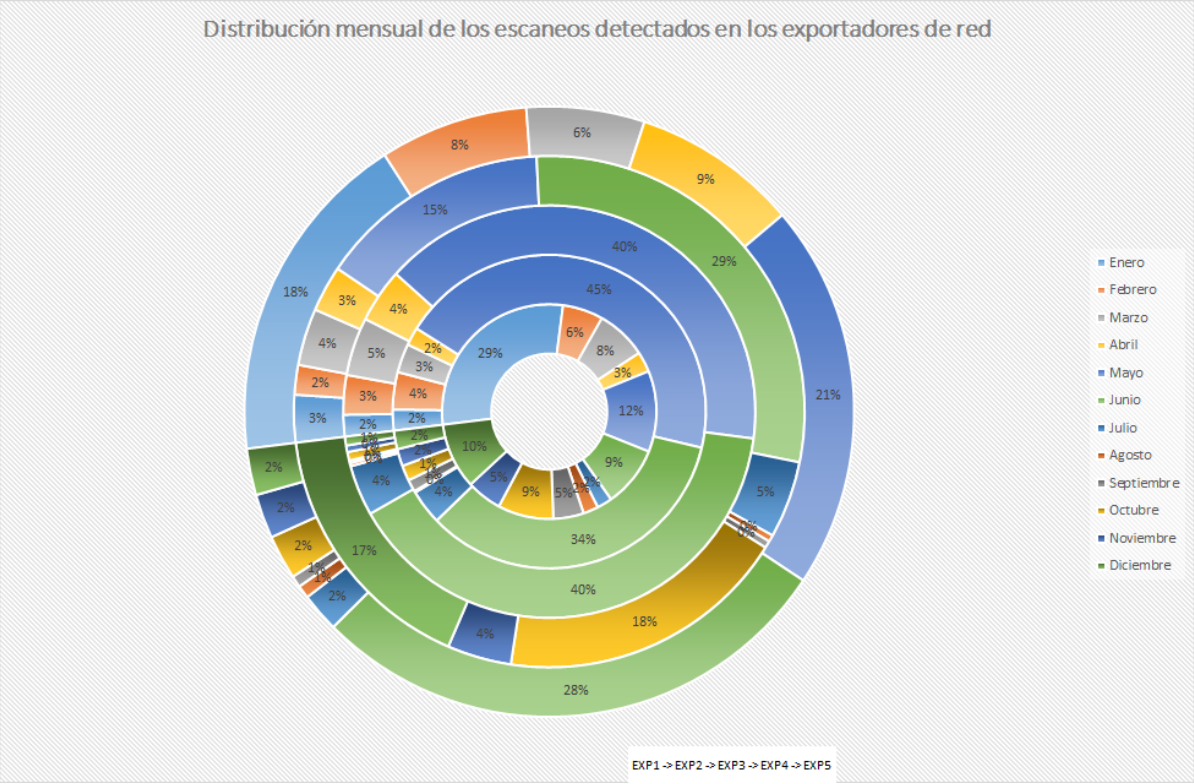


Ilustración 39: Distribución mensual de los escaneos detectados en los exportadores de red.

En la presente gráfica mostramos como se distribuyeron los escaneos detectados para todos los exportadores de red, a nivel mensual. En ello se observa un periodo claro tri-mensual (mayo, junio y julio) en donde en todos los exportadores analizados se concentra una gran mayoría de escaneos. A su vez se observa un periodo de menor actividad, que en algunos exportadores llega a la inactividad, durante los meses de agosto y septiembre.

En cuanto a similitudes, podríamos decir que tanto el exportador 2 y 3 presentan un comportamiento parecido, esto es, las concentraciones máximas de escaneos suceden en los meses de mayo y junio con proporciones parecidas, y los periodos de poca actividad suceden en los meses de octubre, noviembre y diciembre con periodos de inactividad durante agosto y septiembre.

Buscando más similitudes cabe destacar entre los exportadores 1 y 5. En ambos existe una mayor concentración de escaneos durante enero, mayo y junio aunque en diferente proporción. Los periodos de poca actividad suceden en los meses de agosto, septiembre, octubre y noviembre en ambos exportadores.

El exportador 4 actúa de una forma ligeramente diferente respecto a los anteriores. Se concentran dichos escaneos entre los meses de mayo, junio, octubre y diciembre. En este exportador no se observan periodos de inactividad como sucede en los exportadores 2 y 3.

5.5.2 Visión localizada

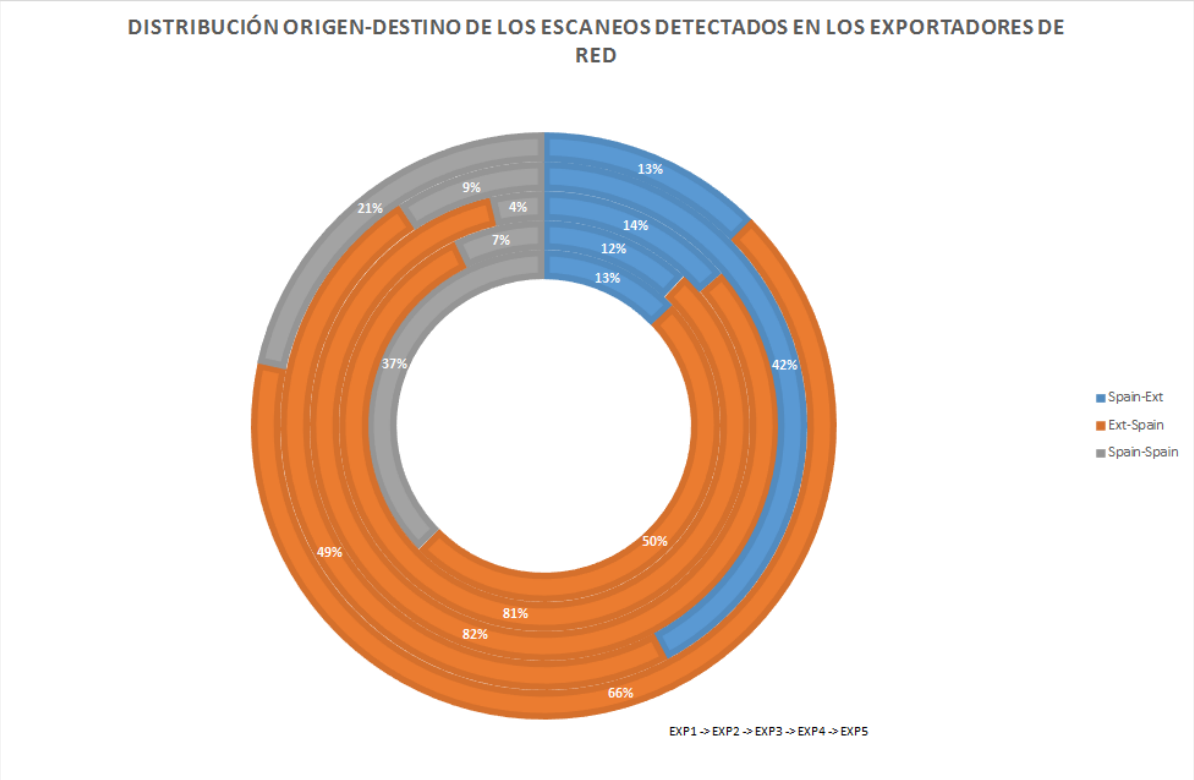


Ilustración 40: Distribución Origen-Destino de los escaneos detectados en los exportadores de red.

En esta gráfica se presenta la distribución Origen-Destino de los escaneos vistos para todos los exportadores de red tomados en el estudio. De forma global presenta una mayor proporción en escaneos con origen foráneo hacia equipos dentro de España. Con menor proporción aparece tanto escaneos con origen España hacia el exterior como escaneos con origen España hacia equipos destino en España.

En cuanto a similitudes, los exportadores 2 y 3 siguen un comportamiento Origen-Destino muy similar. En ambos las proporciones entre los diferentes tipos de Origen-Destino difieren no más de un 3%.

El resto de exportadores de red sigue comportamientos muy diferentes con lo que no nos posibilita establecer ningún tipo de analogías.

En lo que comparten similitudes es en los principales orígenes y destinos vistos en ellos. Como principales orígenes de escaneos, para todos los exportadores de red, son China y Estados Unidos principalmente. Como destinos de escaneos, para todos los exportadores de red, son España, Estados Unidos y Francia.

5.5.3 Visión categorizada

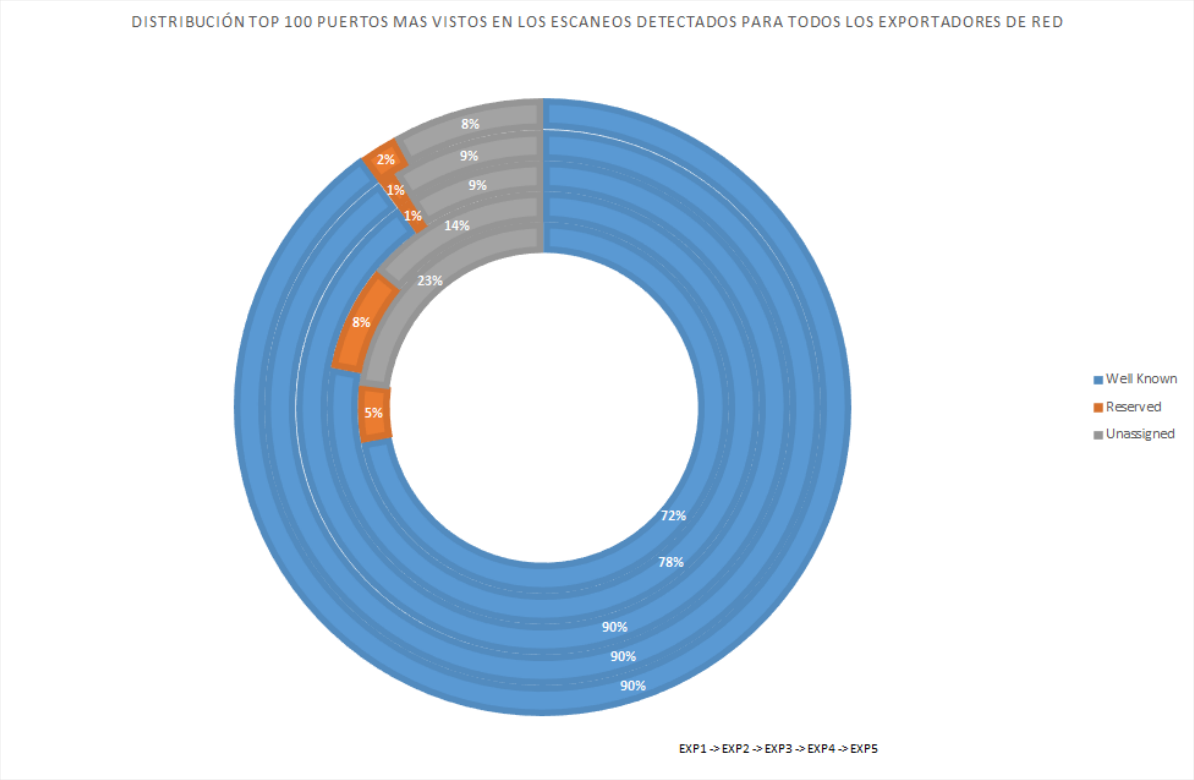


Ilustración 41: Distribución Top 100 más vistos en los escaneos detectados para todos los exportadores de red.

En esta visión categorizada, se expone una distribución top 100 más vistos en los escaneos detectados para todos los exportadores de red. En ello se observa lo siguiente:

En primer lugar la gran similitud entre los exportadores de red 3, 4 y 5 por un lado y los exportadores de red 1 y 2 por otro lado, en cuanto a proporción bajo los tres tipos expuestos. Esto junto a la distribución expuesta en la sección 5.4 nos viene a concluir que: el objetivo de la persona que realiza estos barridos es el conocimiento de la vulnerabilidad de su víctima, mezclando en un mismo escaneo puertos conocidos, puertos reservados y puertos sin asignación.

5.5.4 Otras características obtenidas del estudio

Tabla 5: Otras características obtenidas del estudio

Analisis de escaneo de puertos Año 2013	Port_max	Total_ports	Total Escaneos	TotalPuertos/TotalEscaneos	TotalPuertos/DíasRegistrados	TotalPort/Hora	TotalScan/Dia	TotalScan/Hora
Exportador 1	61	1567	131	11,96	4,78	0,20	0,40	0,02
Exportador 2	941	137230	9312	14,74	418,38	17,43	28,39	1,18
Exportador 3	910	107126	10080	10,63	326,60	13,61	30,73	1,28
Exportador 4	1003	264989	10607	24,98	807,89	33,66	32,34	1,35
Exportador 5	394	33518	2598	12,90	102,19	4,26	7,92	0,33

En esta tabla se recoge alguna serie de características no comentadas en anteriores secciones y algunos resultados derivados de los parámetros que vamos a exponer.

En primer lugar destacaremos el número máximo de puertos visto en un escaneo por exportador de red. Como se observa tenemos una gran diferencia en dicho número a nivel de exportador, siendo de 61 para el primer exportador de red hasta los 1003 puertos vistos para el exportador 4.

Como se observa en la tercera columna, es un tipo de ataque que se sufre en mayor medida en los exportadores de red 2, 3 y 4 (con cifras cercanas al millar de escaneos). Se puede deber dichos números a la cantidad de tráfico que pasa por dichos exportadores de red, que los convierte más "vulnerables" teniendo en cuenta el factor de muestreo que sufren los exportadores de la red de estudio.

Fijándonos en la columna de escaneos por día tenemos que los exportadores de red 2, 3 y 4 comentados anteriormente poseen cifras cercanas a los 30 escaneos diarios mientras que en los otros dos exportadores tomados para el análisis no llegan a los 10 escaneos. Esto confirma lo que anteriormente comentamos y nos permite concluir que un posible atacante, el cual conozca la red, preferirá realizar dichos barridos en aquellos exportadores donde más perjudique el muestreo a la detección que en aquellos exportadores que su caudal de trafico sea mucho menor y el muestreo no afecte en la misma medida que el caso anterior.

5.6 Conclusiones del capítulo 5

En este capítulo, se muestran todos los resultados obtenidos tras el estudio longitudinal planteado. En este estudio se muestran las siguientes características: temporal, espacial y específica del tipo de puerto involucrado en los escaneos vistos por nuestro sistema. De este modo, damos un soporte a los gestores de la red del comportamiento de la misma frente a este tipo de ataques.

La conclusión más relevante que se puede extraer, en base al escenario temporal, es que aparece una gran concentración de estos escaneos en un periodo bimensual situado entre los meses de mayo, junio para la gran mayoría de exportadores de red analizados. En base a la localización de estos ataques, se observa una mayoría de los mismos provenientes desde países asiáticos (China principalmente) para todos los exportadores de red tomados en el estudio. Acorde al tipo de puertos involucrados en cada escaneo detectado, notamos que existe una mezcla del tipo de puertos que se utilizan, es decir, se emplea una combinación tanto de puertos conocidos como reservados y puertos sin asignación. Esto nos muestra que se emplea esta combinación de puertos como cebo para conseguir evitar ser detectados.

Ya en el siguiente y último capítulo de este documento, se recogen las conclusiones obtenidas de este proyecto de fin de carrera y se plantean las líneas de trabajo futuro a seguir.

Capítulo 6: Conclusiones y trabajo futuro

El objetivo principal de este proyecto ha sido principalmente la realización de un sistema capaz de detectar posibles escaneos vistos en una red para, posteriormente, emplear dicho sistema en un caso de estudio concreto como es la red académica RedIRIS. Para lograr este objetivo, se llevó a cabo un estudio exhaustivo del Estado del Arte y las líneas de investigación relevantes para poder desarrollar el sistema que se plantea en este trabajo y conocer así la naturaleza asociada a un escaneo de puertos.

En este estudio se han aprendido los aspectos más relevantes en el proceso de monitorización de una red, lo que nos ha permitido adquirir su propio entendimiento a la vez que su funcionamiento. Esto nos muestra el conjunto de motivaciones encontradas en un estudio longitudinal y las dificultades de trabajar con una gran cantidad de información en un periodo de tiempo siempre limitado y a posteriori. Además cabe destacar como una motivación adicional la presencia del muestreo en las trazas de red analizadas. Añade un factor importante a la hora de prestar una exactitud en los escaneos de puertos detectados pero no imposibilita su detección como se ha demostrado en el presente documento.

Una vez se han expuesto todos los resultados extraídos tras el análisis, se considera alcanzado el objetivo propuesto por el trabajo con las siguientes conclusiones generales:

- Se ha realizado un aprendizaje y familiarización con el conjunto de mecanismos que se emplean para la detección de escaneos de puertos así como un lenguaje de programación que posibilita, mediante la ejecución periódica de *scripts,* su monitorización y obtención de posibles escaneos significativos en la red de análisis.

- Se han ejecutado estas herramientas sobre los registros de flujos de la red académica española, RedIRIS, para entender y conocer la evolución longitudinal de cinco exportadores de flujos de red que forman parte de RedIRIS.

- Se han obtenido una serie de patrones comunes, tras la ejecución sobre la red académica, que nos permite conocer la evolución longitudinal de cada exportador de red. Además, nos permite ofrecer una visión geográfica y nos ofrece una caracterización completa de los tipos de puertos, que han sido empleados en los escaneos que hemos detectado en el sistema.

Los resultados obtenidos del presente trabajo se consideran satisfactorios, dado que hemos sido capaces de conocer el estado actual de la red RedIRIS frente a este tipo de ataques. Sin embargo, se identifican nuevas líneas de trabajo futuro para la profundización del estudio realizado y su correspondiente aportación sobre algunos aspectos no tratados en este trabajo, así como la necesidad de seguir trabajando para mejorar las prestaciones de las herramientas desarrolladas.

Debido a ser un estudio longitudinal y haber desarrollado las herramientas que nos permiten automatizar todo el proceso, resulta de evidente interés, poder repetir el estudio con un nuevo período significativo de datos en un momento anterior y posterior al período analizado. Esto nos permitiría una mejor caracterización de la presencia y/o persistencia, de una manera más detallada profunda. En resumen, obtendríamos una visión más globalizada del comportamiento que presenta este tipo de redes. Por la duración acotada del trabajo se desestimó esta opción pero se mantiene como un posible punto de interés importante para mejorar el sistema ya propuesto y conseguir una mayor eficiencia en los resultados. Como otro posible punto de interés sería establecer una comparación entre nuestro sistema planteado y herramientas que realicen el mismo estudio.

Referencias

[1] SPEROTTO, Anna, et al. An overview of IP flow-based intrusion detection. Communications Surveys & Tutorials, IEEE, 2010, vol. 12, no 3, p. 343-356. http://goo.gl/OWQ0Pe

[2] ARBOR NETWORKS, Protecting IP Services from the Latest Trends in Botnet and DDoSAttacks, 2012. http://goo.gl/6PAma5

[3] BARFORD, Paul; PLONKA, David. Characteristics of network traffic flow anomalies. En Proceedings of the 1st ACM SIGCOMM Workshop on Internet Measurement. ACM, 2001. p. 69-73. http://goo.gl/FY7A8P

[4] MCHUGH, John. Sets, bags, and rock and roll. En Computer Security–ESORICS 2004. Springer Berlin Heidelberg, 2004. p. 407-422. http://goo.gl/68QWUB

[5] GARCIA-TEODORO, Pedro, et al. Anomaly-based network intrusion detection: Techniques, systems and challenges. computers & security, 2009, vol. 28, no 1, p. 18-28. http://goo.gl/5GyO2Z

[6] MORENO, Victor, et al. Multi-granular, multi-purpose and multi-Gb/s monitoring on off-the-shelf systems. International Journal of Network Management, 2014.

[7] FUSCO, Francesco; DERI, Luca. High speed network traffic analysis with commodity multi-core systems. En Proceedings of the 10th ACM SIGCOMM conference on Internet measurement. ACM, 2010. p. 218-224.

[8] HAN, Sangjin, et al. PacketShader: a GPU-accelerated software router. ACM SIGCOMM Computer Communication Review, 2011, vol. 41, no 4, p. 195-206.

[9] GARCÍA-DORADO, José Luis, et al. Low-cost and high-performance: VoIP monitoring and full-data retention at multi-Gb/s rates using commodity hardware. International Journal of Network Management, 2014, vol. 24, no 3, p. 181-199.

[10] SZABÓ, Géza, et al. Traffic classification over Gbit speed with commodity hardware. IEEE J. Communications Software and Systems, 2010, vol. 5.

[11] FORCONESI, Marco, et al. Bridging the Gap between Hardware and Software Open-Source Network Developments

[12] RIZZO, Luigi. Revisiting network I/O APIs: the netmap framework. Communications of the ACM, 2012, vol. 55, no 3, p. 45-51.

[13] CISCO, I.O.S. NetFlow introduction. [2006-09]. http://goo.gl/Gmnkug , 2006.

[14] LI, Yifan, et al. Canine: A combined conversion and anonymization tool for processing netflows for security. En International conference on telecommunication systems modeling and analysis. 2005.

[15] NETWORKS, Configuring J-Flow Statistics. http://goo.gl/MzWn8X , 2011

[16] SMITH, Matt; HUNT, Ray. Network security using NAT and NAPT. En Networks, 2002. ICON 2002. 10th IEEE International Conference on. IEEE, 2002. p. 355-360.

[17] GRAY. GNU Cflow Manual. http://goo.gl/wseLXB , 2011.

[18] http://www.ietf.org/rfc/rfc3954.txt

[19] http://www.ietf.org/dyn/wg/charter/ipfix-charter.html

[20] CALIGARE. NetFlow Export Format, http://goo.gl/159pT4, 2006

[21] http://www.ietf.org/rfc/rfc3917.txt

[22] http://goo.gl/OMzRyk

[23] NETFLOW, Cisco. Introduction to Cisco IOS NetFlow - A Technical Overview. Cisco System, Inc., http://goo.gl/BaQhxu , 2007.

[24] SOMMER, Robin; FELDMANN, Anja. NetFlow: Information loss or win? En Proceedings of the 2nd ACM SIGCOMM Workshop on Internet measurment. ACM, 2002. p. 173-174.

[25] FULLMER, Mark. Flow-tools. (2006-03-01) [2006-04-15]. http://goo.gl/prrtJ4 , 2007.

[26] http://en.wikipedia.org/wiki/NetFlow

[27] BASTIAS LÓPEZ, Xavier. Evaluación de técnicas de captura y muestreo de tráfico en redes de ordenadores, TFC EETACT, 2012.

[28] http://www.cisco.com/c/en/us/td/docs/ios/solutions_docs/netflow/nfwhite.html#wp1031447

[29] FLOYD, Sally; PAXSON, Vern. Difficulties in simulating the Internet. IEEE/ACM Transactions on Networking (TON), 2001, vol. 9, no 4, p. 392-403.

[30] BORGNAT, Pierre, et al. Seven years and one day: Sketching the evolution of internet traffic. En INFOCOM 2009, IEEE. IEEE, 2009. p. 711-719.

[31] HARRINGTON, Edward F. Measuring Network Change: Renyi cross entropy and the second order degree distribution. En Proceedings of passive and active measurement conference. 2006.

[32] GARCÍA-DORADO, José Luis, et al. Characterization of isp traffic: Trends, user habits, and access technology impact. Network and Service Management, IEEE Transactions on, 2012, vol. 9, no 2, p. 142-155.

[33] EIMANN, Raimund, et al. Network Event Detection with T-Entropy. Department of Computer Science, The University of Auckland, New Zealand, 2005.

[34] IZAL, Mikel, et al. Dissecting bittorrent: Five months in a torrent's lifetime. En *Passive and Active Network Measurement*. Springer Berlin Heidelberg, 2004. p. 1-11.

[35] VILLALON HUERTA, Antonio. Sistemas de Detección de Intrusos. 2005

[36] SIYAN, Karanjit. Internet y seguridad en redes. 1995. p. 1 y 9

[37] RAMOS, Jibran De La Rosa. Seguridad en redes inalambricas IEEE 802.11 (WLAN) con WEP mejorado. 2006.

[38] NORTHCUTT, Stephen. Snort: IDS and IPS toolkit. Syngress Press, 2007.

[39] http://www.alegsa.com.ar/Diccionario/Imagen/75

[40] ZWICKY, Elisabeth. Building Internet Firewalls, 1995

[41] ZWICKY, Elisabeth. Check Point FireWal-I v 3.0, 1997.

[42] SEGURIDAD Y TECNOLOGIA, Cooge theme. Seguridad y amenazas en Redes. 2008.

[43] FERNANDEZ, Arnaldo. Seguridad en Redes. 1987.

[44] SPEROTTO, Anna; SADRE, Ramin; PRAS, Aiko. Anomaly characterization in flow-based traffic time series. En IP Operations and Management. Springer Berlin Heidelberg, 2008. p. 15-27.

[45] ZHAO, Qi; XU, Jun; KUMAR, Abhishek. Detection of super sources and destinations in high-speed networks: Algorithms, analysis and evaluation.Selected Areas in Communications, IEEE Journal on, 2006, vol. 24, no 10, p. 1840-1852.

[46] KIM, Myung-Sup, et al. A flow-based method for abnormal network traffic detection. En Network operations and management symposium, 2004. NOMS 2004. IEEE/IFIP. IEEE, 2004. p. 599-612.

[47] WAGNER, Arno; PLATTNER, Bernhard. Entropy based worm and anomaly detection in fast IP networks. En Enabling Technologies: Infrastructure for Collaborative Enterprise, 2005. 14th IEEE International Workshops on. IEEE, 2005. p. 172-177.

[48] ZSEBY, Tanja, et al. Nightlights: Entropy-Based Metrics for Classifying Darkspace Traffic Patterns. En Passive and Active Measurement. Springer International Publishing, 2014. p. 275-277.

[49] SHANNON, Claude Elwood. A mathematical theory of communication. ACM SIGMOBILE Mobile Computing and Communications Review, 2001, vol. 5, no 1, p. 3-55.

[50] GATES, Carrie, et al. Scan detection on very large networks using logistic regression modeling. En Computers and Communications, 2006. ISCC'06. Proceedings. 11th IEEE Symposium on. IEEE, 2006. p. 402-408.

[51] STOECKLIN, Marc Ph; LE BOUDEC, Jean-Yves; KIND, Andreas. A two-layered anomaly detection technique based on multi-modal flow behavior models. En Passive and Active Network Measurement. Springer Berlin Heidelberg, 2008. p. 212-221.

[52] PUKKAWANNA, Sirikarn, et al. Investigating the utility of S-transform for detecting Denial-of-Service and probe attacks. En Information Networking (ICOIN), 2014 International Conference on. IEEE, 2014. p. 282-287.

[53] MAI, Jianning, et al. Is sampled data sufficient for anomaly detection?. En Proceedings of the 6th ACM SIGCOMM conference on Internet measurement. ACM, 2006. p. 165-176.

[54] MAI, Jianning, et al. Impact of packet sampling on portscan detection. Selected Areas in Communications, IEEE Journal on, 2006, vol. 24, no 12, p. 2285-2298.

[55] PAREDES-OLIVA, Ignasi; BARLET-ROS, Pere; SOLÉ-PARETA, Josep. Portscan detection with sampled netflow. En Traffic Monitoring and Analysis. Springer Berlin Heidelberg, 2009. p. 26-33.

[56] ANDROULIDAKIS, Georgios, et al. Understanding and evaluating the impact of sampling on anomaly detection techniques. En Military Communications Conference, 2006. MILCOM 2006. IEEE. IEEE, 2006. p. 1-7.

[57] BRAUCKHOFF, Daniela, et al. Impact of packet sampling on anomaly detection metrics. En Proceedings of the 6th ACM SIGCOMM conference on Internet measurement. ACM, 2006. p. 159-164.

[58] POYATO, Chelo Malagón. NetFlow y su aplicación en seguridad. RedIRIS: boletín de la Red Nacional de I+ D RedIRIS, 2009, no 87, p. 37-42.

[59] http://rediris.es/

[60] ALVAREZ-CAMPANA, M., et al. Castba: Internet traffic measurements over the Spanish R&D ATM Network. En HPOVUA Workshop, Rennes (France). 1998.

[61] http://wiki.bandaancha.st/RedIRIS

[62] https://www.redirisnova.es/mm/presentacion-RedIRIS-NOVA.pdf

[63] CLIMENT, Jordi. Análisis longitudinal de medidas de red, TFG-UAM, 2014.

[64] http://www.sanog.org/resources/sanog6/gaurab-sanog6-flow-tools.pdf

[65] http://es.wikipedia.org/wiki/AWK

[66] http://www.marblestation.com/?p=761

[67] DOUGHERTY, Dale; ROBBINS, Arnold. Sed & awk. " O'Reilly Media, Inc.", 1997.

[68] GALTSEV, Aleksey A.; SUKHOV, Andrei M. Network attack detection at flow level. En Smart Spaces and Next Generation Wired/Wireless Networking. Springer Berlin Heidelberg, 2011. p. 326-334.

[69] http://www.iana.org/assignments/service-names-port-numbers/service-names-port-numbers.xhtml

Anexo A: Pliego de Condiciones

El proyecto presenta por un lado la implementación de un mecanismo de detección de escaneos de puertos a partir de una traza NetFlow. Además, se expone un caso de estudio para demostrar la veracidad de dicho mecanismo.

Por ello, la memoria en un primer lugar se analiza las investigaciones que existen sobre dicho tema y se plantea una nueva vía a las ya existentes.

A.1 Entregables

- Memoria del proyecto. Se entrega una versión original y dos copias de la misma, encuadernadas de forma normalizada. Cesión a la Universidad Autónoma de Madrid, de forma gratuita y no exclusiva, por el máximo plazo legal y con ámbito universal, para que pueda ser utilizada de forma libre y gratuita por todos los usuarios del repositorio y del portal eciencia, los derechos de reproducción, distribución y de comunicación pública, tal y como queda descrito en la Ley de Propiedad Intelectual.

- Código. En él se expone en lenguaje awk el mecanismo de detección de escaneo de puertos en NetFlow y se tiene a su vez un script en lenguaje bash de automatización del proceso para el caso de estudio presente.

A.2 Condiciones de desarrollo – Recursos Hardware

- Equipo de desarrollo HP Pavilion Elite HPE con procesador Intel® Core™ i7-2600 3.4 GHZ, memoria RAM 8GB y disco duro 1 TB. Utilizado para el desarrollo del software y la realización de pruebas.

- Equipo de desarrollo de la documentación Packard-Bell TJ-66 con procesador Intel® Core™ 2 Duo CPU P8700 2.53 GHZ, memoria RAM 4GB y disco duro 160 GB. Utilizado para el desarrollo del software y la realización de pruebas.

A.3 Condiciones de desarrollo – Recursos Software

- Sistema operativo Windows 8.1 utilizado en el equipo de la elaboración de la documentación.

- Sistema operativo Linux Mint 10 ‘Julia’ utilizado en el equipo de desarrollo.

- Matlab r2013a empleado en la elaboración de gráficas presentes en la documentación.

- Microsoft Office Word 2013 utilizado para la elaboración de la documentación.

Anexo B: Presupuesto del proyecto

En este presente apartado desglosaremos el presupuesto total empleado en el proyecto en los siguientes apartados:

- Presupuesto de Ejecución Material.
- Gastos generales y Beneficio industrial.
- Honorarios por redacción y dirección del proyecto.
- Costes totales.

El presupuesto de ejecución material junto a los gastos generales y el beneficio industrial forman parte de lo denominado presupuesto por ejecución por contrata que, junto a los honorarios percibidos por la redacción y dirección del proyecto, conforman el presupuesto de costes total.

Todas estas cantidades están presentes en euros.

B.1 *Presupuesto de Ejecución Material (PEM)*

El presupuesto de ejecución material consta de los costes de mano de obra y el coste de los recursos materiales empleados a lo largo del desarrollo del proyecto. En este presupuesto no se incluyen los honorarios de la dirección del proyecto que son considerados aparte.

B.1.1 Descomposicion del proyecto en tareas

Para la compresión de esta serie de costes empleados, se distribuye el proyecto en una serie de tareas. Para una visualización de la precedencia temporal de las actividades realizadas, se elaborará un diagrama de Gantt del proyecto.

El presente proyecto consta de las siguientes tareas:

Tarea 1

Objetivo: Estudio y posterior análisis del estado del arte de los ataques en red. En esta fase se dictaminará que tipo de ataque se va a estudiar de los que existen en la actualidad.

***Duración*:** 3 meses

Esfuerzo: 1,5 personas-mes (Ingeniero Superior)

Tarea 2

Objetivo: Estudio del estado del arte de este tipo de ataque en concreto. En esta fase se adquirirá el *modus operandi* de este ataque y se estudiarán los diferentes métodos de detección del mismo

***Duración*:** 1 mes

Esfuerzo: 0,5 personas-mes (Ingeniero Superior)

Tarea 3

Objetivo: Preparación del entorno. Instalación del entorno de desarrollo. Estudio y búsquedas de software y librerías necesarias para el desarrollo.

***Duración*:** 0,5 meses

Esfuerzo: 0,25 personas-mes (Ingeniero Superior)

Tarea 4

Objetivo: Implementación del sistema de detección basado en el algoritmo más eficiente desde el punto de vista computacional. Para llevar a cabo la implementación se decidirá el lenguaje de programación a utilizar (por ejemplo, AWK en el entorno GNU/Linux).

***Duración*:** 3 meses

Esfuerzo: 1,5 personas-mes (Ingeniero Superior)

Tarea 5

Objetivo: Pruebas funcionales. Probar que el funcionamiento del sistema de detección planteado corresponde con el diseño para los registros capturados.

***Duración*:** 0,5 meses

Esfuerzo: 0,25 personas-mes (Ingeniero Superior)

Tarea 6

Objetivo: Reprogramación, corrección de errores vistos tras las pruebas funcionales.

***Duración*:** 0,5 meses

Esfuerzo: 0,25 personas-mes (Ingeniero Superior)

Tarea 7

Objetivo: Pruebas funcionales. Validación del proceso de reprogramación y corrección de errores.

***Duración*:** 1 mes

Esfuerzo: 0,5 personas-mes (Ingeniero Superior)

Tarea 8

Objetivo: Estudio de los resultados obtenidos y obtención de gráficas y conclusiones.

***Duración*:** 1,5 meses

Esfuerzo: 0,75 persona-mes (Ingeniero Superior)

Tarea 9

Objetivo: Redacción de la memoria del proyecto. En esta fase, se describen los pasos realizados durante las fases anteriores, con los resultados obtenidos en las mismas. Como final de este proyecto, se redactaran las conclusiones obtenidas en el mismo y su trabajo futuro.

***Duración*:** 2,5 meses

Esfuerzo: 1,25 persona-mes (Ingeniero Superior) y 0,5 personas-mes (Administrativo)

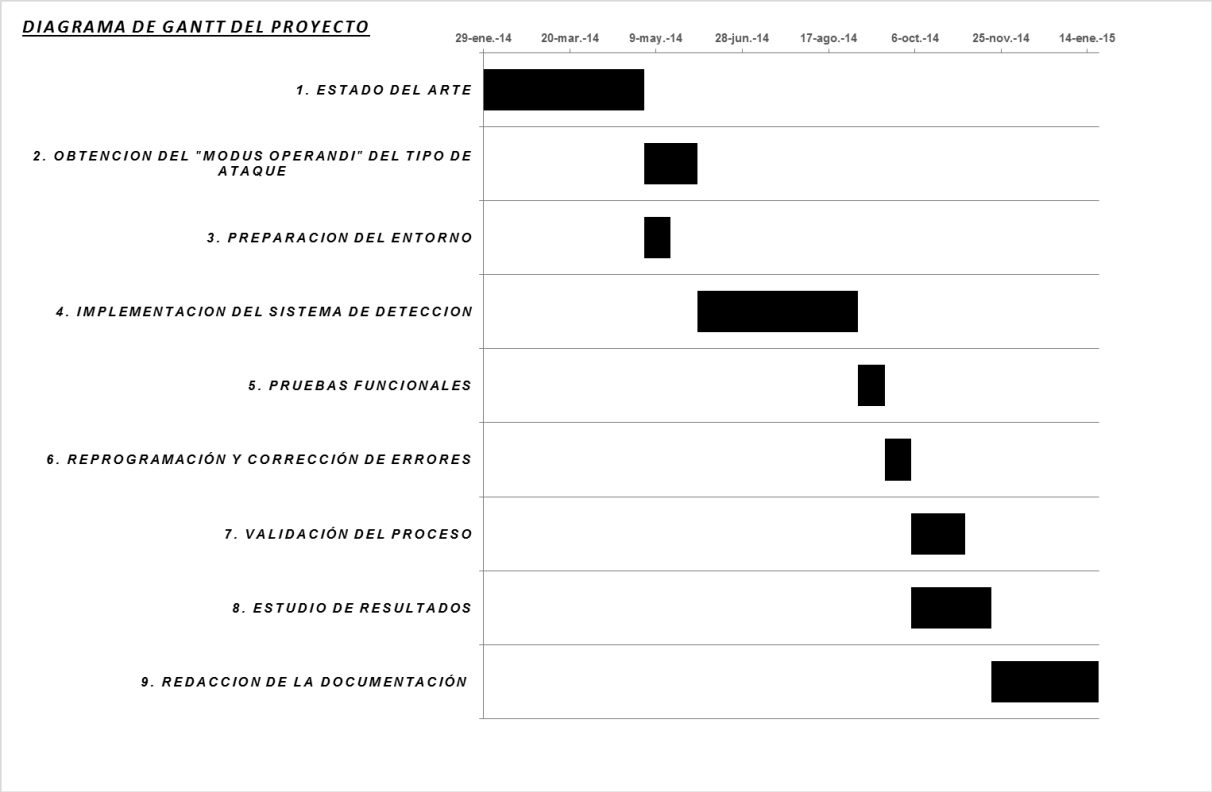


Ilustración 42: Diagrama de Gantt del proyecto

En la siguiente figura se expone un diagrama de Gantt dónde se observa, las relaciones de dependencia entre las diferentes actividades realizadas durante el desarrollo del proyecto.

Con ello, tenemos un plazo de ejecución del proyecto de 12 meses.

B.1.2 Costes de mano de obra

Para el desarrollo del proyecto, se requieren los siguientes perfiles profesionales.

- Un Ingeniero de Telecomunicación, encargado del planteamiento, desarrollo e implementación del trabajo técnico.

- Un Administrativo, encargado de la redacción, presentación y encuadernación del proyecto.

La estimación de estos costes se realiza en base a los siguientes datos:

- Cotizaciones según el Régimen General de la Seguridad Social. El perfil de Ingeniero Superior, pertenece al grupo 1 y el perfil de Administrativo, pertenece al grupo 7.

- Asumiendo una jornada laboral de 8h/día y de 21 días laborables/mes.

- Por el artículo 33 del XVII CONVENIO COLECTIVO NACIONAL DE EMPRESAS DE INGENIERÍA Y OFICINAS DE ESTUDIOS TÉCNICOS publicado en el BOE, nos muestra el salario anual que deben percibir las categorías profesionales empleadas en el presente proyecto.

- Por el artículo 22.1 del XVII CONVENIO COLECTIVO NACIONAL DE EMPRESAS DE INGENIERÍA Y OFICINAS DE ESTUDIOS TÉCNICOS publicado en el BOE, nos muestra que el número de horas efectivas de trabajo anual (contando con todas las festividades que se presenten) son 1800 h

Con la distribución del trabajo en el conjunto de tareas y los siguientes datos expuestos, tenemos lo siguiente:

Tabla 6: Costes salariales

Costes salariales		
Concepto	**Ingeniero Superior**	**Administrativo**
Base cotizable máxima anual	**43.272,00 €**	**43.272,00 €**
Contingencias comunes(23,6%)	**5.529,67 €**	**2.459,56 €**
Desempleo, FOGASA y Formación profesional((6,7+0,2+0,6)%)	**1.757,31 €**	**781,64 €**
Coste Seguridad Social	**7.286,99 €**	**3.241,20 €**
Salario bruto anual	**23.430,82 €**	**10.421,88 €**
Coste salarial anual	**30.717,81 €**	**13.663,08 €**
Coste salaral por hora	**17,07 €**	**7,59 €**
Numero de horas	**900**	**125**
Coste total	**15.358,90 €**	**948,83 €**

Tabla 7: Costes mano de obra

Costes mano de obra	
Concepto	**Coste**
Ingeniero Superior	**15.358,90 €**
Administrativo	**948,83 €**
Total	**16.307,73 €**

B.1.3 Costes recursos materiales

En las siguientes tablas muestran el coste de los recursos materiales empleados durante el desarrollo del presente proyecto, considerando un periodo de amortización, tanto hardware como software de 4 años.

En la primera tabla, buscamos indicar los costes totales y posteriormente se atribuirán las cuantías correspondientes a la amortización de los recursos prevista durante el periodo de utilización en el proyecto.

Tabla 8: Gastos en Recursos Hardware

Recursos Hardware			
Concepto	**Coste total**	**Meses**	**Coste real**
Equipo de desarrollo	1.250,00 €	10	**260,42 €**
Equipo de generación de la documentación	700,00 €	2	**29,17 €**
Disco Duro Interno Seagate Barracuda 2000 GB	68,00 €	10	**14,17 €**
Disco Duro Externo Toshiba Ebasics 1000 GB	55,00 €	12	**13,75 €**
Total			**317,50 €**

Tabla 9: Gastos en Recursos Software

Recursos Software			
Concepto	**Coste total**	**Meses**	**Coste real**
Windows 8.1 Pro x64	129,00 €	2	**5,38 €**
Matlab r2014a Student Version	82,00 €	3	**5,13 €**
Microsoft Office 2013 Hogar y Estudiantes	99,00 €	2	**4,13 €**
Total			**14,63 €**

Tabla 10: Gastos recursos materiales

Costes materiales	
Concepto	**Coste**
Recursos Hardware	**317,50 €**
Recursos Software	**14,63 €**
Total	**332,13 €**

B.1.4 Coste total de los recursos

La suma de los costes de la mano de obra junto a los costes de los recursos materiales conforman lo denominado ***Presupuesto de Ejecución Material*** (P.E.M).

Tabla 11: Presupuesto de Ejecución Material

Presupuesto de ejecución material	
Concepto	**Coste total**
Costes materiales	**332,13 €**
Costes mano de obra	**16.307,73 €**
Total	**16.639,85 €**

B.2 Gastos generales y Beneficio industrial

Como Gastos generales comprenden aquellos gastos derivados del empleo de las instalaciones, cargas fiduciarias, amortizaciones, etc. Por ello, el ***Presupuesto de Ejecución por Contrata*** queda como sigue a continuación.

Tabla 12: Presupuesto de Ejecución por Contrata

Presupuesto de ejecución por contrata	
Concepto	**Coste total**
Presupuesto de ejecución material	16.639,85 €
Gastos generales (17 % del P.E.M.)	2.828,77 €
Beneficio industrial (6 % del P.E.M.)	998,39 €
Total	**20.467,02 €**

B.3 Honorarios por redacción y dirección del proyecto

Los Honorarios que recomienda aplicar el Colegio Oficial de Ingenieros de Telecomunicación, tanto para la redacción como para la dirección del proyecto son los asociados a Trabajos tarifados por tiempo empleado, con un valor de un 5.6%.

B.4 Costes totales

La suma de la serie de imputaciones anteriores y aplicando el IVA correspondiente, obtenemos el presupuesto total del presente proyecto.

Tabla 13: Presupuesto total del proyecto

Presupuesto total	
Concepto	**Coste total**
Presupuesto de ejecución por contrata	**20.467,02 €**
Honorarios por dirección	**1.146,15 €**
Honorarios por redacción	**1.146,15 €**
Subtotal	**22.759,33 €**
IVA(21%)	**4.779,46 €**
Total	**27.538,78 €**

El presupuesto total obtenido del proyecto asciende a la cantidad de VENTISIETE MIL QUINIENTOS TREINTA y OCHO euros con SETENTA y OCHO céntimos.

Madrid, Septiembre de 2015.

Fdo.: José Antonio San Román Gil

Ingeniero de Telecomunicación