IMPLEMENTACIÓN DE REDES SEÑUELO COMO TÉCNICA DE MEJORAMIENTO DE LA CIBERDEFENSA

Esp. Ing. Fernando Boiero, Esp. Ing. Carlos Tapia, Ing. Miguel Sánchez
Investigadores de la Especialización en Seguridad Informática, del Instituto Universitario Aeronáutico ubicado en la Av. Fuerza Aérea 6500 Córdoba Capital.
Tutores: MCs. Ing. Eduardo Esteban Casanovas, Ing. Norberto Gaspar Cena.

Introducción

En el presente trabajo se plantea la hipótesis que es posible generar con recursos propios, un mecanismo de mejoramiento de la ciberdefensa a partir del tratamiento de la información obtenida del entendimiento de los ataques recibidos en una red señuelo. Si tomamos en consideración el devenir y actualidad de la Seguridad Informática y la Ciberdefensa, nuestro desarrollo se basa en la implementación de contramedidas ante ataques informáticos que no sólo ayuda a su prevención sino que habilita al estudio pormenorizado de las técnicas utilizadas por los intrusos.
Las herramientas que se describirán son las honeynets o redes señuelo, las cuales serán constituidas y operadas mediante software moderno que brinda simplicidad para su implementación y monitoreo.
Asimismo, se plantearán los beneficios de esta tecnología en el marco de Infraestructuras Críticas de un país, en pos del incremento de los niveles de Seguridad para aquellos servicios esenciales de una sociedad.

Objetivo

1. Diseñar una red virtual sobre la cual se realizará una implementación inicial de pruebas de una solución honeynet.
2. Realizar un estudio comparativo sobre distintas herramientas y soluciones de honeypots y honeynets disponibles en el mercado y se seleccionará las más adecuadas de acuerdo a una evaluación de criterios ponderados.
3. Instalar y configurar la solución de software de honeynet previamente seleccionada junto con sus herramientas sobre la infraestructura virtual, la que servirá de base para recibir los ataques que luego serán analizados
4. Sacar conclusiones de la utilización de los sensores, teniendo en cuenta su potencial utilidad para la protección de las Infraestructuras Críticas que están a cargo del Ministerio de Defensa de la Nación.

Metodología

La investigación se desarrolló en tres etapas. En la primera, se describieron teóricamente las diferentes Tecnologías, comparándolas y seleccionando una de ellas. En la segunda etapa, se implementó y parametrizó la solución seleccionada en un ambiente de test, y por último, se expuso los resultados obtenidos.

Desarrollo

Situación problemática

En los últimos años, el creciente grado de informatización así también como de interconectividad en las tecnologías de almacenaje y procesamiento de datos en prácticamente todos los ámbitos de nuestras vidas ha acarreado consigo, una nueva serie de aspectos a considerar en lo concerniente a la seguridad en las mismas. El formidable progreso en ámbitos como la electrónica, computación y redes informáticas dieron también origen a lo que, por su masiva relevancia en la nuestra vida cotidiana moderna, se reconoce como un nuevo escenario bélico, el Ciberespacio. Una de las amenazas más activas a la que nos enfrentamos en Internet hoy en día es el cibercrimen. Delincuentes con habilidades y capacidades que van en aumento se encuentran constantemente desarrollando métodos para sacar provecho de la actividad criminal en línea. En consecuencia, se vuelve imperante tanto en el ámbito estatal como en los entornos privados, la necesidad de establecer un sistema de políticas, conductas, procesos y procedimientos con fin de

proteger a las mismas ante el potencial peligro de eventuales ataques las infraestructuras de datos. Basta mencionar algunos casos recientes como Wikileaks (2007) o Stuxnet (2010) para entender que la denominada "ciberguerra" ya ha comenzado y debemos estar preparados para la misma. El presente trabajo se focalizara en la presentación, estudio e implementación de una de las contramedidas más importantes desarrolladas no solo con propósitos preventivos ante la amenaza de los ciberataques, sino también con el fin de poder realizar un estudio de los mismos en un entorno seguro. La herramienta de la que estamos hablando son las honeynets o "redes señuelo". Una honeynet es una red de computadoras denominadas comúnmente honeypots, diseñada con el único propósito de ser comprometidas por un intruso. No hay personal que utilice estos equipos para desempeñar ninguna función real o productiva, simplemente están conectados a la red en espera de que algún ciberdelincuente intente atacarlos de forma remota. Por lo tanto, todo el tráfico que circule en la misma, es en esencia, ilegal. El concepto es en cierta forma, similar al uso de una carnada en busca de una buena pesca.
Una de las ventajas del uso de honeynets para el monitoreo pasivo de red, es que los honeypots pueden ser implantados en computadoras de bajo costo o baja performance. De esta manera, se logra engañar a los posibles atacantes de una infraestructura de producción, desviando su atención y esfuerzos hacia un entorno controlado y carente de valor (desde el punto de vista de un atacante); evitando así comprometer los sistemas informáticos reales sobre los cuales los procesos organizacionales se sustentan.

Un honeypot es un "sistema señuelo" monitoreado muy de cerca el cual sirve a varios propósitos: distraer adversarios de otros sistemas valiosos en una red, proveer de una alerta temprana acerca de nuevas tendencias de ataques y explotaciones, y permitir el examen en profundidad de adversarios durante y tras el ataque al mismo. Los honeypots, como un blanco fácil de ataques, pueden simular muchos hosts vulnerables en una red y nos proveen con valiosa información de los atacantes. Los honeypots no son la única solución a la seguridad en redes, pero son herramientas que se implementan para detectar actividad no deseada en las mismas. No son sistemas de detección de intrusos (IDS), pero nos enseñan cómo mejorar la seguridad de nuestras redes y, tal vez más importante que eso, nos enseñan a qué buscar o prestar atención [**01**]. Se trata de sistemas construidos y configurados con el propósito de ser atacados. Además, son "sistemas trampa" para los atacantes, los cuales son desplegados para contrarrestar los recursos de un atacante, aletargarlo para que éste pierda tiempo con el honeypot en lugar de atacar los sistemas de producción. En esta sección se analizan los principios básicos de los honeypots, sus tipos, varias soluciones honeypot concretas, sus ventajas y desventajas y una comparación entre ellas.
Las principales razones para contar con un sistema de honeypots son que:

1. Tienen baja tasa de falsos positivos y alta taza de aciertos.
2. Brindan capacidades para confundir y distraer a los atacantes.
3. Según su configuración, pueden funcionar virtualmente sin intervención de los administradores de seguridad.
4. Ayuda al entrenamiento de los equipos de seguridad.
5. Existen muchas opciones gratuitas.

Infraestructura Crítica

Teniendo en cuenta que nuestro principal objetivo de trabajo se encuentra centrado en la aplicación de este mecanismo sobre una infraestructura crítica, surge la necesidad de explicar y dar contexto a lo que a los fines de este trabajo es considerado como una infraestructura crítica.
A nivel mundial hay varios organismos que han definida a las infraestructuras críticas de diferente manera pero todas ellas con grandes similitudes. Nosotros nos hemos basado en la siguiente:
"El Plan Nacional de Protección de Infraestructuras Críticas las define como: *"Aquellas instalaciones, redes, servicios y equipos físicos y de tecnología de la información cuya interrupción o destrucción tendría un impacto mayor en la salud, la seguridad o el bienestar económico de los ciudadanos o en el eficaz funcionamiento de las instituciones del Estado y de las Administraciones Públicas".* Esta definición fue establecida por la Directiva europea: 2008/114/CE del 8 de diciembre de 2008, subrayando sobre la importancia de *"la identificación y*

designación de infraestructuras críticas europeas y la evaluación de la necesidad de mejorar su protección". "

Alternativas de software de honeypots analizados

Propiedades / Software	ManTrap	BOF	Specter	Honeyd	MHN
Nivel de interacción	Alto	Bajo	Alto	Bajo	Alto
Libre	No	No	No	Si	Si
Código abierto	No	No	No	Si	Si
Soporte de archivos log	Si	No	Si	Si	Si
Emulación de SO	Si	No	Si	Si	Si
Servicios soportados	No limitado	7	14	No limitado	No limitado

Luego de analizar algunas opciones de honeypots, se optó por adoptar la plataforma Modern Honey Network (abreviada MHN [**02**]) provee manejo de grado empresarial del software de honeypot de código abierto de uso más corriente, desde su seguro despliegue hasta el agregado de miles de eventos. MHN hace del manejo de honeypots seguros una tarea extremadamente simple [**03**] [**04**]. MHN es un software de código abierto en su totalidad, el cual soporta el despliegue de forma distribuida y en gran escala de honeypots internos y externos. MHN usa el estándar HPFeeds y honeypots de baja interacción para mantener la efectividad y seguridad a nivel de grado empresarial. MHN soporta los siguientes honeypots: Snort [**05**], Suricata [**06**], Dionaea [**07**], Conpot [**08**], Kippo [**09**], Amun [**10**], Glastopf [**11**], Wordpot [**12**], ShockPot [**13**], y p0f [**14**].

Con base en dicha plataforma se montó una red virtual con los principales sensores, en ambiente de laboratorio, a los efectos de estudiar su comportamiento ante distintos ataques, siendo la arquitectura inicial implementada la que se muestra a continuación.

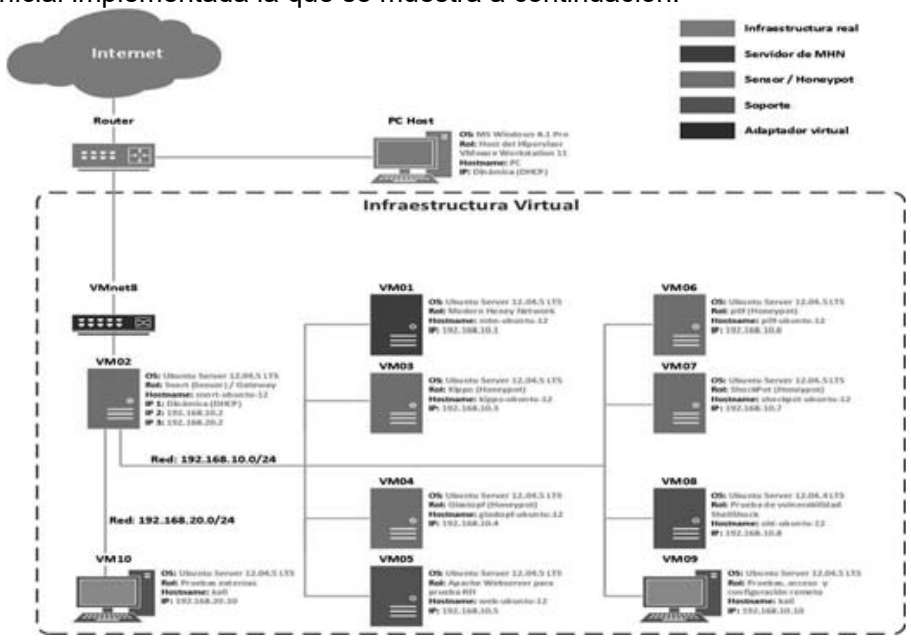

Fig. 1: Arquitectura de honeynet MHN implementada.

Uno de los componentes clave de MHN es su aplicación web de muy fácil manejo, la cual permite administrar la honeynet de manera centralizada. Esta aplicación permite al usuario seleccionar el sensor deseado (de una lista de sensores disponibles), a continuación genera una línea de comando Linux la cual se ejecutará en el sistema que hospedará a dicho sensor. Este comando instalará todas las dependencias necesarias junto al software del sensor en cuestión y lo enlazará

al servidor MHN. Una vez que la instalación haya finalizado, entonces comenzará a recopilar información a través de un protocolo de código abierto llamado “HPFeeds” [**13**].

Fig. 2: Interfaz de administración de MHN.

Sensores de MHN evaluados

En el presente trabajo se ha decidido desplegar, configurar y realizar pruebas con los siguientes sensores: Kippo, p0f, Glastopf, ShockPot y Snort. Sus principales características son:

Kippo: Es un honeypot de interacción media que emula un servidor SSH. Ofrece al atacante un completo sistema de archivos falso con el cual interactuar. Permitiéndole agregar y remover archivos.
Capacidad para agregar contenido falso a los archivos. Simula una conexión SSH con alguna máquina remota, y muchos “comandos” ejecutados por el atacante devuelven salidas simuladas.
p0f: Es una versátil herramienta de “OS fingerprinting” pasivo.
Glastopf: Es un honeypot que recopila información sobre ataques en base a aplicaciones web.
ShockPot: Es un honeypot emulador de aplicación web diseñado para detectar atacantes que intentan explotar la vulnerabilidad “Shellshock”.
Snort: Es un sensor basado en el Intrusion Detection System “Snort”, el cual es estándar de facto.

MHN en funcionamiento

Resulta interesante compartir algunos resultados obtenidos durante la segunda etapa del proyecto.
Una vez completada la configuración de los distintos sensores los cuales fueron cuidadosamente elegidos y preparados con una estructura muy similar a una estructura real, se los expuso a la red en la esperanza de comenzar a recibir ataques y poder comenzar así con el análisis de los mismos.
A continuación se agrega una de las interfaces de monitoreo y control del sistema que permite ver la cantidad de ataques que está recibiendo cada uno de los sensores.

Podemos destacar que a poco de estar en funcionamiento nuestro sensor Snort lleva recibidos 59.381 ataques y que el sensor Glastopf viene recibiendo 9.424 ataques. Esta experiencia de campo nos demuestra que efectivamente se ha realizado una configuración interesante para los atacantes y también nos muestra que este es el punto de partida para el verdadero trabajo de análisis que hay que realizar sobre cada uno de los ataques recibido.

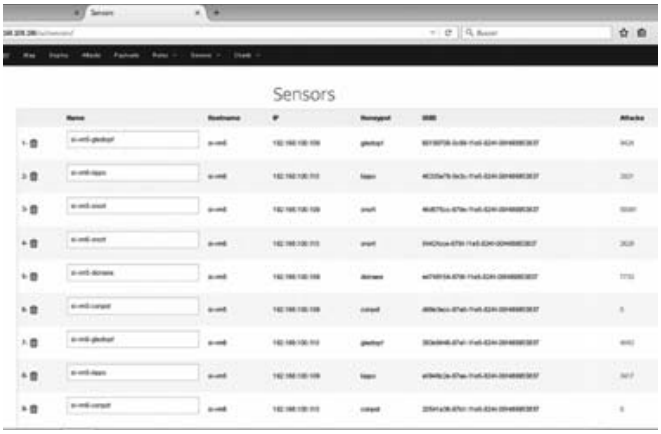

Fig. 3: Dashboard de MHN, resumen de sensores.

Fig. 4: Dashboard de MHN, resumen de ataques ultimas 24hs.

Otra de las vistas que brindan información del status de la cantidad de ataques recibidos, es el dashboard que resume los últimos recibidos dentro de las 24 hs anteriores, Figura 4.

Finalmente podemos mostrar resultados obtenido sobre el sensor Kippo. Es muy rico el análisis que se puede realizar sobre este tipo de sensor porque entre otras cosas se puede determinar si el atacante está automatizando su ataque o está haciendo un ataque inteligente teniendo en cuenta las condiciones de contexto asociadas al punto atacado.
En la figura siguiente se ven las contraseñas que los atacantes están eligiendo para romper el sistema de acceso.

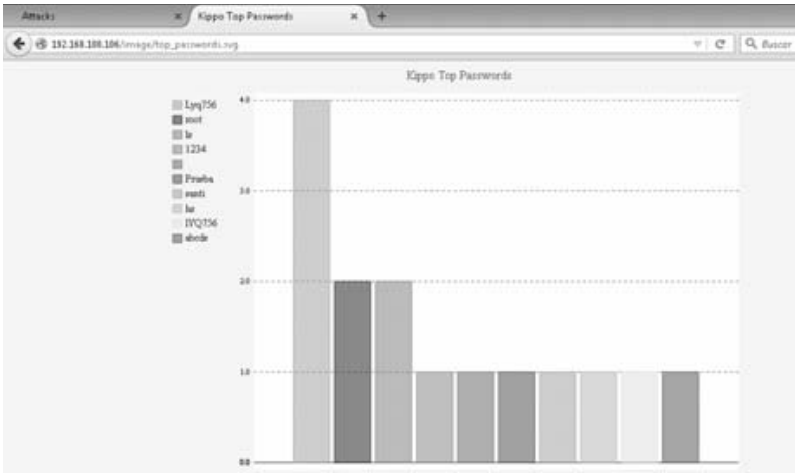

Fig. 5: Dashboard de MHN, resumen de ataques sobre Kippo.

Conclusiones

El desarrollo de nuestro trabajo nos permite concluir que es posible generar con recursos propios, un mecanismo de mejoramiento de la ciberdefensa a partir del tratamiento de la información obtenida del entendimiento de los ataques recibidos en una red señuelo.
1. Se diseño una red virtual sobre la que se realizaron las pruebas con la particularidad que su diseño fue pensado para replicar una infraestructura crítica.
2. Luego de un profundo análisis de los diferentes factores de comparación explicitados en el trabajo se tomó la decisión que MHN es la mejor implementación de las honey net disponibles y la que mejor responde a nuestras exigencias en función de los objetivos planteados para nuestro estudio.
3. Fue posible montar varios tipos de sensores que emulen conductas de diferentes protocolos.
4. Esta solución y sus herramientas fueron instaladas en una infraestructura virtual construida con el especial propósito de probarlas y experimentar con ellas allí. Se analizaron numerosos tipos de

ataques como Denegación de Servicio Distribuidos, Escalación de Privilegios, Buffer Overflow entre otros.
5. Se generaron estadísticas de ataques en base de datos procesados en archivos log, permitiendo sentar las bases para poder hacer crecer el proyecto y asistir/capacitar para otras implementaciones. También se pudieron reconocer la utilización de sentencias no registradas en las bases de datos de ataques conocidos.
6. La utilización de herramientas para detectar ataques atrayendo intrusos permite entender las técnicas de ataques. Teniendo en cuenta esos resultados podemos mejorar sensiblemente nuestro esquema de Defensa en Profundidad y establecer nuestras estrategias defensivas en base a los ataques observados

Desafíos futuros

1) Incrementar el número y complejidad de honeypots de la red hasta llegar a la adaptación de las herramientas a las Infraestructuras Críticas, las cuales tienen mayor tamaño y complejidad.
2) Ganar en automatización de análisis de resultados.
3) Centralización de las estadísticas obtenidas por las herramientas para tomar decisiones respecto estrategias defensivas, teniendo en cuenta la multiplicidad de ejecutores.
3) Trabajar en forma conjunta con otros organismos para compartir estadísticas de ataques, obteniendo así datos agregados, tendencias y proyecciones.
4) Desarrollar nuevos sensores y herramientas tanto para atacar, como para contramedidas.
5) Realizar aportes al Honeynet Project mundial.

Bibliografía

[01] Marcin Nawrocki, Matthias Wahlisch, Thomas C. Schmidt, Christian Keil, Jochen Schonfelder, Universitat Berlin, “A Survey on Honeypot Software and Data Analysis”, 2016.
URL: https://arxiv.org/pdf/1608.06249.pdf
[02] Modern Honey Network, Documento “README.md”, Anomali, Inc., 2014.
URL: https://github.com/threatstream/mhn
[03] Shah Manthan Jigneshkumar, Vishwakarma Institute of Information Technology, “Modern Honey Network”, 2016.
URL: http://www.ijrat.org/downloads/ncpci2016/ncpci-34.pdf
[04] Asif Al Ferdous Khan, UNIVERSITAT AUTÒNOMA DE BARCELONA, “Creation of a Collaborative Model of Honeypots”, 2015.
URL: https://ddd.uab.cat/pub/tfg/2015/tfg_28071/TFG_Honeypots.pdf
[05] The Snort Project - SNORT Users Manual v2.9.8.3, 2015.
URL: http://manual.snort.org
[06] Suricata, Suricata User Guide, 2016.
URL: https://redmine.openinfosecfoundation.org/projects/suricata/wiki/Suricata_User_Guide
[07] Dionaea Honeypot, Documento “README”, 2013.
URL: https://github.com/rep/dionaea
[08] Conpot ICS/SCADA Honeypot, Documento “Usage Guide”, 2013.
URL: http://mushorg.github.io/conpot/usage/usage.html#http
[09] Kippo SSH Honeypot, Documento “README.md”, 2010.
URL: https://github.com/desaster/kippo
[10] Amun Honeypot, Documento “News”, 2012.
URL: http://amunhoney.sourceforge.net/
[11] MushMush Foundation, Documento “README.rst”, 2015.
URL: https://github.com/mushorg/glastopf
[12] Gianluca Brindisi, Documento “README.md”, 2012.
URL: https://github.com/gbrindisi/wordpot
[13] ThreatStream, Documento “README.md”, 2014.
URL: https://github.com/threatstream/shockpot
[14] Michal Zalewski, Documento “README”, 2012.
URL: http://lcamtuf.coredump.cx/p0f3/README
[15] HPfeeds, Documento “README.md”, 2013.
URL: https://github.com/rep/hpfeeds