

HONEYPOTS (SERVIDORES TRAMPA)

Artículo extraído de: http://www.xombra.com/go_articulo.php?articulo=56

El papel de la tecnología del sistema de detección de intrusos basado en señuelos -- o "honeypots" - está evolucionando. Los honeypots, que alguna vez fueron utilizados principalmente por los investigadores como una forma de atraer a los hackers a un sistema de redes para estudiar sus movimientos y comportamiento, están adquiriendo una importancia cada vez mayor en la seguridad empresarial. En efecto, al brindar detección temprana de actividad no autorizada en las redes, los honeypots son ahora más útiles que nunca para los profesionales de seguridad de TI. Este artículo analiza el funcionamiento de los honeypots y su tecnología, que se está convirtiendo en el componente clave del sistema de capas de protección contra intrusos.

Los Honeypots son una emocionante tecnología nueva, con un enorme potencial para la comunidad informática. Los primeros conceptos fueron introducidos por primera vez por varios iconos en la seguridad informática, especialmente por Cliff Stoll en el libro "The Cuckoo's Egg" y el trabajo de Bill Cheswick "An Evening with Berferd". Desde entonces, los honeypots han estado en una continua evolución desarrollándose en una poderosa herramienta de seguridad hoy en día. El propósito del presente trabajo es el de explicar exactamente qué son los honeypots, sus ventajas y desventajas, y su importancia en la seguridad.

Los Honeypots (Potes de miel) "Consisten en activar un servidor y llenarlo de archivos tentadores, hacer que sea difícil, pero no imposible penetrarlo y sentarse a esperar que aparezcan los intrusos. Los honeynets (conjuntos de honeypots) dan a los crackers un gran espacio para recorrer. Presentan obstáculos que poseen el nivel de complejidad suficiente para atraerlos, pero sin irse al extremo para no desalentarlos... Ellos juegan con los archivos y conversan animadamente entre ellos sobre todo los fascinantes programas que encuentran, mientras el personal de seguridad observa con deleite cada movimiento que hacen. Francamente, siento una combinación de sentimientos con respecto a espiar a la gente, aunque no sean buenas personas". Dan Adams.

También existe el Honeynet, que es un conjunto de Honeypots, así abarca más información para su estudio. Incluso hace más fascinante el ataque al intruso, lo cual incrementa el número de ataques. La función principal a parte de la de estudiar las herramientas de ataque, es la de desviar la atención del atacante de la red real del sistema y la de capturar nuevos virus o gusanos para su posterior estudio. Una de las múltiples aplicaciones que tiene es la de poder formar perfiles de atacantes y ataques.

Son sistemas que deliberadamente se decide exponerlos a ser atacados o comprometidos. Estos, no solucionan ningún problema de seguridad, son una herramienta que nos sirve para conocer las estrategias que se emplean a la hora de vulnerar un sistema. Son una herramienta muy útil a la hora de conocer de forma precisa los ataques que se realizan contra la plataforma de trabajo que hemos elegido, o bien, las plataformas configuradas de la misma forma, y que sirven para guardar todos los procesos que se están ejecutando contra o en nuestro sistema con el claro objetivo de acceder a información sensible para una organización, empresa o corporativo. Así mismo nos permiten conocer nuevas vulnerabilidades y

riesgos de los distintos sistemas operativos y diversos entornos y programas, las cuales aún no se encuentren debidamente documentadas.

En general, existen dos tipos de honeypots:

Para la producción y para la investigación.

-* Honeypots para la investigación:

Gran parte de la atención actual se centra en los honeypots para la investigación, que se utilizan para recolectar información sobre las acciones de los intrusos. El proyecto HoneyNet, por ejemplo, es una organización para la investigación sobre seguridad voluntaria, sin ánimo de lucro que utiliza los honeypots para recolectar información sobre las amenazas del ciberespacio.

-* Honeypots para la producción:

Se les ha prestado menor atención a los honeypots para la producción, que son los que se utilizan para proteger a las organizaciones. Sin embargo, se les concede cada vez más importancia debido a las herramientas de detección que pueden brindar y por la forma cómo pueden complementar la protección en la red y en el host.

** ¿Cómo funcionan los honeypots?

Los honeypots también se pueden describir como de alta o baja interacción, distinción que se basa en el nivel de actividad que le permiten al atacante. Un sistema de baja interacción ofrece actividad limitada; la mayoría de las veces funciona al emular los servicios y sistemas operativos. Las principales ventajas de los honeypots de baja interacción es que son relativamente fáciles de instalar y mantener; también implican un riesgo mínimo porque el atacante nunca tiene acceso a un sistema operativo real para perjudicar a otros sistemas.

"Honeyd" es un ejemplo de honeypot de baja interacción, cuya función principal es monitorear el espacio de direcciones IP no utilizado. Cuando Honeyd detecta un intento de conectarse a un sistema que no existe, intercepta la conexión, interactúa con el atacante fingiendo ser la víctima para captar y registrar al ataque.

Por el contrario, los honeypots de alta interacción utilizan sistemas operativos reales y aplicaciones reales y no emulan nada. Al ofrecerles a los atacantes sistemas reales para que interactúen, las organizaciones pueden aprender mucho sobre su comportamiento. Los honeypots de alta interacción no imaginan como se comportará un atacante y proporcionan un ambiente que rastrea todas las actividades, lo que les permite a las organizaciones conocer un comportamiento al que de otra manera no tendrían acceso.

Los sistemas de alta interacción también son flexibles y los profesionales de la seguridad de TI pueden implementarlos en la medida que quieran. Además, este tipo de honeypot proporciona un objetivo más realista, capaz de detectar atacantes de mayor calibre. Los honeypots de alta interacción pueden ser complejos de instalar. Sin embargo, requieren que se implementen tecnologías adicionales para evitar que los atacantes los utilicen para lanzar ataques a otros sistemas.

** Sistemas de Detección de Intrusos.

Los sistemas de detección de intrusos se han convertido en un componente importante en la caja de herramientas de un buen administrador de seguridad.

Algunos aún no lo tienen claro, y piensan que un IDS (Intrusión Detection System) puede ser la panacea que nos lleve a la más absoluta tranquilidad y a una irreal sensación de seguridad, más peligrosa en muchos casos que la inseguridad en sí misma. Un detector de intrusos no es más que una de las medidas de seguridad que necesariamente hay que tomar para proteger una red.

En resumen, un sistema de detección de intrusos hace exactamente eso. Detectar posibles intrusiones. Específicamente, pretende detectar ataques o abusos al sistema, alertando con los pormenores del ataque. Proporciona una seguridad parecida a la que un sistema de alarma instalado en casa puede suponer. Mediante varios métodos, ambos detectan si un intruso, atacante o ladrón está presente, y en consecuencia disparan una alarma. Por supuesto, la alarma puede saltar sin motivo, al igual que el administrador puede no llegar a oírla. Ante estas irregularidades, solo resta estar atento y revisar frecuentemente con cautela los avisos recibidos.

-* Ventajas:

Los honeypots son un concepto increíblemente simple, las cuales ofrecen una fortaleza muy poderosa

- * Conjunto de datos pequeños pero de gran importancia: Los Honeypots recolectan pequeñas cantidades de información. En lugar de loguear 1 Gb por día, loguean sólo 1 Mb de datos por día. En vez de generar 10.000 alertas por día, pueden generar sólo 10 alertas por día. Recuerden, los honeypots sólo capturan actividad sospechosa ya que cualquier interacción con un honeypot es muy probablemente actividad no autorizada o una actividad maliciosa. Propiamente dicho, los honeypots reducen el "ruido" recogiendo sólo datos indispensables, de gran valor, los producidos únicamente por "chicos malos". Esto significa que es mucho más fácil (y barato) de analizar los datos que un honeypot recoge.

- * Nuevas herramientas y tácticas: Los honeypots son diseñados para capturar cualquier cosa que interactúa con él, incluyendo herramientas o tácticas nunca vistas.

- * Mínimos recursos: Los honeypots requieren mínimos recursos, sólo capturan actividad irregular. Esto significa que un viejo Pentium con 128 mb de RAM puede manejar fácilmente una entera red clase B en una red OC-12.

- * Encriptación o IPv6: A diferencia de la mayoría de las tecnologías para la seguridad, como los sistemas IDS, honeypots trabajan bien en entornos encriptados como IPv6. No importa lo que los "chicos malos" lancen hacia el honeypot, el honeypot lo detectará y lo capturará.

- * Información: Los honeypots pueden recoger información "en profundidad" como pocos, si es que existen tecnologías que se le parezcan.

- * Simplicidad: Finalmente, los honeypots son conceptualmente simples. No hay por qué desarrollar algoritmos raros, ni complejas tablas que mantener, o firmas que actualizar. Mientras más simple sea la tecnología, menos posibilidades de errores o desconfiguraciones habrá.

-* Desventajas:

Como en cualquier otra tecnología, los honeypots también tienen su debilidad. Esto es debido a que no reemplaza a la actual tecnología, sino que trabaja con las existentes.

- * Visión Limitada: Los honeypots pueden sólo rastrear y capturar actividad que interactúen directamente con ellos. Los Honeypots no podrán capturar ataques a otros sistemas vecinos, al menos que el atacante o la amenaza interactúe con el honeypot al mismo tiempo.

- * Riesgo: Todas las tecnologías de seguridad tienen un riesgo. Los firewalls tienen

el riesgo de que sean penetrados, la encriptación tiene el riesgo de que sean rotos, sensores IDS tienen el riesgo de que fallen al detectar ataques. Los Honeypots no son diferentes, tienen un riesgo también. Específicamente, los honeypots tienen el riesgo de que sean apoderados y controlados por los "chicos malos" y que lo utilicen para dañar otros sistemas. El riesgo es variado para los diferentes honeypots. Dependiendo en el tipo de honeypots puede haber un riesgo no mayor a la de una falla del sensor IDS, mientras que en otros honeypots puede que haya que enfrentarse a una situación crítica.

**** Utilidades de Honeypots**

Específicamente, cómo se pueden usar los honeypots. Una vez más, tenemos dos categorías generales, los honeypots pueden ser usados con propósitos productivos o de investigación. Cuando es utilizado con propósitos productivos, los honeypots están protegiendo una organización. En este mundo se incluye, prevención, detección o la ayuda a organizaciones a responder a un ataque. Cuando es utilizado con propósitos de investigación, los honeypots son utilizados para recolectar información. La importancia de esta información depende según las organizaciones. Algunas organizaciones querrán estudiar la tendencia de las actividades hacker, mientras otras estarán interesadas en la predicción y prevención anticipada, o las fuerzas legales. En general, honeypots de baja interacción son utilizados con propósitos productivos, mientras honeypots de alta interacción son utilizados con propósitos de investigación. Sin embargo, los dos tipos de honeypots pueden funcionar para ambos propósitos. Cuando es utilizado con propósitos productivos, honeypots pueden proteger organizaciones en las tres formas al mismo tiempo: prevención, detección y reacción. Veremos con más profundidad cómo un honeypot puede trabajar en las tres formas perfectamente.

Los honeypots pueden ayudar a prevenir ataques en varias formas. El primero es contra ataques automatizados, como gusanos (worms) o auto-rooters. Estos ataques son basados en herramientas que aleatoriamente escanean (sondean) redes enteras buscando sistemas vulnerables. Si un sistema vulnerable es encontrado, estas herramientas automatizadas atacarán y tomarán el sistema (con gusanos que se auto-copian, copiándose a sí mismo a la víctima). Uno de los métodos para proteger de tales ataques es bajando la velocidad de su escaneo y potencialmente detenerlos. Llamados "sticky honeypots" (Tarros de miel "pegajosos"), estas soluciones monitorean el espacio IP no utilizado. Cuando los sistemas son escaneados, estos honeypots interactúan con él y disminuyen la velocidad del ataque. Hacen esto utilizando una variedad de trucos TCP, como poniendo el "Window size" a cero o poniendo al atacante en un estado de espera continua. Ésto es excelente para bajar la velocidad o para prevenir la diseminación de gusanos que han penetrado en la red interna. Un ejemplo de un sticky honeypot es el LaBrea Tarpit. Los "Honeypots pegajosos" son más comunes encontrarlos entre soluciones de baja interacción (hasta podría llamársele soluciones "no interactivas", ya que reducen tanto la velocidad que hacen gatear al atacante ;). Honeypots pueden también proteger su organización de perpetradores humanos. Este concepto se conoce como engaño o disuación. La idea es confundir al atacante, hacerle perder el tiempo y recursos interactuando con honeypots. Mientras tanto, su organización habría detectado la actividad del atacante y tendría tiempo para reaccionar y detener el ataque. Hasta se puede dar un paso más allá: si un atacante sabe que su organización está utilizando honeypots pero no sabe cuales son los sistemas honeypots y cuales son sistemas legítimos, quizás tenga miedo de ser capturado por honeypots y decida no atacarlo. Por lo tanto, honeypots disuaden al atacante. Un ejemplo de honeypot diseñado para hacer esto, es el Deception Toolkit, un honeypot de baja interacción.

La segunda forma por la cual honeypots pueden ayudar a proteger una organización es por medio de la detección. La detección es crítica, su propósito es la identificación de una falla o ruptura para la prevención. No importa cuán segura sea la organización, siempre habrán fallas si los hombres están involucrados en el proceso... Detectando al atacante, podrán rápidamente reaccionar ante ellos, deteniéndolos, o mitigando el daño que hicieron. Tradicionalmente, la detección ha sido extremadamente difícil de hacer. Tecnologías como sensores IDS y sistemas de logueo han sido inefectivos por diversas razones: Generan muchos datos, grandes porcentajes de falsos positivos, inhabilidad de detectar nuevos ataques, y la inhabilidad de trabajar en forma encriptada o en entornos IPv6. Los Honeypots son excelentes en detección, solventando muchos de los problemas de la detección clásica. Los honeypots reducen falsos positivos, capturando pequeñas cantidades de datos de gran importancia, capturan ataques desconocidos como nuevos exploits o shellcodes polimórficos, y trabajan en forma encriptada o en entornos IPv6. Podrá aprender más acerca de esto en el trabajo Honeypots: Simple, Cost Effective Detection. En general, honeypots de baja interacción son la mejor solución para la detección, hacen fácil la tarea de utilizar y mantener que honeypots de alta interacción y tienen un riesgo limitado.

La forma tercera y final por la cual honeypots nos pueden ayudar a proteger una organización es en la respuesta. Una vez que una organización detecta una falla, ¿cómo debe responder? Esto es a menudo uno de los grandes retos que una organización debe enfrentar. Hay por lo general poca información acerca de quién es el atacante, cómo ingresó al sistema o cuánto daño hizo. En estas situaciones, la información detallada acerca de las actividades del atacante son cruciales. Hay dos problemas que afectan a la respuesta al incidente: el primero, a menudo los sistemas comprometidos no pueden ser desconectados de la red para ser analizados. Sistemas de producción, como el servidor mail de una organización, son tan críticos que aunque estén hackeados los expertos en seguridad no pueden desconectarlos y hacer un análisis forense como corresponde. Están limitados a analizar el sistema encendido mientras sigue proveyendo sus servicios productivos. Esto merma la habilidad para analizar qué sucedió, cuánto daño hizo el atacante, e incluso si el atacante accedió a otros sistemas de la red. El otro problema es que incluso en el caso de que este desconectado, hay tanta polución de datos que es muy difícil determinar qué es lo que hizo el "chico malo". Con polución de datos me refiero que hay tanta actividad (logging in de usuarios, lecturas de cuentas de mail, archivos escritos a bases de datos, etc...) que puede ser difícil determinar cuál es la actividad normal del día a día y qué es lo que hizo el atacante. Los Honeypots pueden ayudar a solventar ambos problemas. Honeypots son un excelente herramienta de incidencias ya que pueden rápidamente y fácilmente ser sacados de la red para un análisis forense completo, sin el impacto en las operaciones empresariales de todos los días. Recuerden que la única actividad que guardan los honeypots son las relacionadas con el atacante, ya que no las utilizan nadie, excepto de señuelo para los atacantes. La importancia de los honeypots aquí es la rápida entrega de la información, analizada en profundidad, a la organización que la necesita para responder rápida y eficientemente a un incidente. En general, honeypots de alta interacción son la mejor solución para la respuesta. Para reaccionar ante un intruso, se necesita conocimientos en profundidad sobre qué hicieron, cómo ingresaron, y qué herramientas utilizaron. Para la obtención de ese tipo de datos, se necesitarán las capacidades de un honeypot de alta interacción.

Tipos de honeypots

Los honeypots vienen con diversidad de colores y gustos, haciendo difícil su comprensión. Para ayudarnos a entender mejor a los honeypots y a todos los diferentes tipos, dividiremos a dos categorías generales: honeypots de "baja

interacción" y de "alta interacción". Estas categorías nos ayudan a entender con qué tipo de honeypots está trabajando usted, sus fortalezas y debilidades. La interacción define el nivel de actividad que un honeypot le permite tener un atacante. Los honeypots de baja interacción tienen una interacción limitada, normalmente trabajan únicamente emulando servicios y sistemas operativos. La actividad del atacante se encuentra limitada al nivel de emulación del honeypot. Por ejemplo, un servicio FTP emulado escuchando en el puerto 21 probablemente estará emulando un login FTP o probablemente soportará algunos comandos FTP adicionales. Las ventajas de un honeypot de baja interacción es su simplicidad, estos honeypots tienden a ser fáciles de utilizar y mantener con un riesgo mínimo: Por lo general es instalar un software, elegir el sistema operativo y el servicio a emular y monitorear, y dejar que el programa camine por sí solo desde ahí... Este proceso cercano al "plug and play" hace que la utilización de éstos sea muy fácil para la mayoría de las organizaciones. Incluso, los servicios emulados mitigan el riesgo conteniendo la actividad del atacante: el atacante nunca tiene acceso al sistema operativo real donde puede atacar o dañar otros sistemas. Las principales desventajas de los honeypots de baja interacción es que registran únicamente información limitada y son diseñados a capturar actividad prevista, los servicios emulados sólo pueden llegar hasta ahí. También es fácil para un atacante detectar un honeypot de baja interacción, sin importar cuán buena sea la emulación. Un perpetrador hábil puede, con el debido tiempo, detectar su presencia. Ejemplos de honeypots de baja interacción se incluyen: Specter, Honeyd, y KFSensor.

Los honeypots de alta interacción son diferentes, éstos generalmente son soluciones complejas ya que implica la utilización de sistemas operativos y aplicaciones reales. Nada es emulado, le damos a los hackers lo real. Si quiere un honeypot Linux corriendo un servidor FTP, tendrá que construir un verdadero sistema Linux y montar un verdadero servidor FTP. Las ventajas de dicha solución son dos: Primero, usted capturará grandes cantidades de información dándoles a los hackers sistemas reales con la cual interactuar, podrá aprender la completa extensión de sus actividades, cualquier cosa desde rootkits nuevos hasta sesiones internacionales de IRC. La segunda ventaja es que los honeypots de alta interacción no asumen acerca del posible comportamiento que tendrá el atacante, en lugar de eso proveen un entorno abierto que captura todas las actividades realizadas. Esto permite a las soluciones de alta interacción conocer el comportamiento no esperado. Un excelente ejemplo de esto es cómo un honeypot capturó comandos "back door" codificados en un protocolo IP no estandar (específicamente protocolo IP 11, Network Voice Protocol). No obstante, esto también incrementa el riesgo de los honeypots ya que los atacantes pueden utilizar estos sistemas operativos reales para lanzar ataques a sistemas internos que no forman parte de los honeypots. En consecuencia, se requiere la implementación de una tecnología adicional que prevenga al atacante de dañar otros sistemas que no son honeypots. En general, honeypots de alta interacción pueden hacer todo lo que uno de baja interacción puede hacer y mucho más, pero pueden ser más complejos para utilizar y mantener. Ejemplos de honeypots de alta interacción son Symantec Decoy Server y los Honeyd. Puede encontrar una completa lista de ambos honeypots de baja y alta interacción en la página de Honeypot Solutions.

Para la mejor comprensión de ambos tipos de honeypots de baja y alta interacción, veamos los siguientes ejemplos. Empezaremos con el honeypot de baja interacción Honeyd.

Honeyd: honeypot de baja interacción

Honeyd es un honeypot de baja interacción. Desarrollado por Niels Provos, Honeyd es OpenSource y está diseñado para correr principalmente en sistemas Unix (aunque fue portado a Windows). Honeyd trabaja en el concepto del monitoreo del espacio IP no utilizado. Cualquier instante que observa un intento de conexión a un IP no utilizado, éste intercepta la conexión e interactúa con el atacante,

pretendiendo ser la víctima. Por defecto, Honeyd detecta y loguea cualquier conexión a cualquier puerto UDP o TCP. Como si fuera poco, se pueden configurar a los servicios emulados para que monitoreen puertos específicos, como un servidor FTP emulado monitoreando el puerto TCP 21. Cuando un atacante conecta al servicio emulado, el honeypot no sólo detecta y loguea la actividad, sino que captura también toda la actividad del atacante con el servicio emulado. En caso del servidor FTP emulado podemos potencialmente capturar el login y password, los comandos que ingresa y quizás también descubrir lo que está buscando o su identidad. Todo depende del nivel de emulación del honeypot. La mayoría de los servicios emulados trabajan de la misma manera. Ellos esperan un tipo específico de comportamiento, y están programados para reaccionar en una forma predeterminada. Si el ataque A hace esto, entonces reaccionan de este modo. Si el ataque B hace aquello, entonces responden del otro modo. La limitación está en que si el atacante hace algo que la emulación no tiene previsto, entonces no saben cómo responder. La mayoría de los honeypots de baja interacción, incluyendo Honeyd, simplemente genera un mensaje de error. Podrá ver cuales son los comandos que soporta el servidor FTP emulado de Honeyd viendo el código fuente.

**** Otras características**

Algunos honeypots, como Honeyd, no sólo emulan servicios sino que también emulan el Sistema Operativo. En otras palabras, Honeyd puede aparentar ser un router Cisco, un webserver WinXP o un servidor DNS Linux. Existen varias ventajas en emular diferentes sistemas operativos. Primero, el honeypot puede encajar mejor con la red existente si el honeypot tiene la misma apariencia y comportamiento que las computadoras productivas. Segundo, se puede apuntar a atacantes específicos proveyéndoles sistemas y servicios que buscan a menudo o sistemas y servicios que usted quiere aprender al respecto. Hay dos elementos en sistemas operativos emulados. El primero es el servicio emulado. Cuando un atacante se conecta a un servicio emulado, usted puede tener al servicio comportándose y aparentando ser un sistema operativo determinado. Por ejemplo, si tiene un servicio emulando un webserver y quiere que su honeypot aparente ser un Win2000 servidor, entonces deberá emular el comportamiento de un IIS webserver, y para el caso de un Linux, debería emular el comportamiento de un webserver Apache. La mayoría de los honeypots emulan el SO de esta manera. Algunos honeypots sofisticados llevan la emulación un paso adelante (como lo hace el Honeyd): No sólo emulan en el nivel de servicio, sino que en el nivel del stack del IP. Si alguien realiza active fingerprinting para determinar el SO de su honeypot, muchos honeypots responderán al nivel del IP Stack del SO real en donde esté instalado el honeypot. Honeyd falsea la respuesta, emulando no sólo el servicio sino emulando también el stack del IP, comportándose como si fuera realmente otro sistema operativo. El nivel de emulación y sofisticación depende en qué tecnología de honeypot elige para usar.

*** Honeynets: Honeypot de alta interacción**

Los Honeynets son un ejemplo ideal de honeypots de alta interacción. Honeynets no son un producto, no son una solución software en donde se instala en una computadora. En lugar de eso, los Honeynets son una arquitectura, una entera red de máquinas diseñados para ser atacados. La idea es tener una arquitectura que sea una red altamente controlada, un lugar donde toda actividad sea controlada y capturada. En esta red nosotros ponemos a nuestras víctimas en forma intencionada, computadoras reales corriendo aplicaciones reales. Los "chicos malos" encuentran, atacan, rompen estos sistemas en su propia iniciativa. Cuando hacen esto, ellos no saben que están en un Honeynet. Toda su actividad, desde sesiones encriptadas SSH hasta emails y archivos subidos son capturados sin que lo noten. Esto es realizado introduciendo módulos en el kernel en los "sistemas víctima" que capturan toda las acciones de los atacantes. Al mismo tiempo, el Honeynet controla

la actividad del atacante. Los Honeynets hacen esto mediante la utilización de un gateway Honeywall . Este gateway permite el tráfico de entrada a los "sistemas víctima", pero controla el tráfico de salida usando tecnologías de prevención contra intrusos. Esto le da al atacante la flexibilidad de interactuar con las sistemas víctimas, pero previene al atacante de dañar otros sistemas que no forman parte del Honeynet. Un ejemplo:

Resumiendo,

Baja Interacción

Solución que emula sistema operativos y servicios.

Alta Interacción

No hay emulación, son suministrados sistemas operativos y servicios reales.

La necesidad de un IDS:

Los sistemas de detección de intrusos son un elemento integral y necesario en una infraestructura de información segura, representando el complemento ideal a los cortafuegos en las redes. De manera sencilla, los IDS permiten una completa supervisión de las redes, independientemente de la acción tomada, esa información siempre estará ahí y será necesaria para determinar la naturaleza del incidente y su fuente.

Además, pueden ser utilizados para monitorizar objetos concretos, detectar abusos particulares o avisar de comportamientos anómalos en un sistema. Además de estas técnicas, el sistema puede estar orientado al host individual o a la red, algo así como la diferencia entre tomar los acontecimientos como individuales o colectivos. En general, son más útiles los orientados a red (distribuidos), por tomar el tráfico como un todo del que entran y salen paquetes constantemente, en vez de tomar cada máquina como posible objetivo de ataque. Esto último es lo que se ha venido haciendo desde siempre en los logs que genera por ejemplo, un sistema operativo con respecto a los acontecimientos internos.

Demasiada información: Uno de los problemas comunes de los IDS tradicionales es que generan un gigantesco volumen de alertas. El solo peso de este "ruido" consume tiempo, demanda demasiados recursos y costos para revisar los datos. Por el contrario, los honeypots recolectan los datos únicamente cuando alguien interactúa con ellos. Los grupos de pequeña información pueden facilitar y hacer más eficiente la labor de identificar las actividades no autorizadas y proceder en consecuencia.

Falsos positivos: Quizás el mayor inconveniente de un IDS es que muchas de las alertas generadas son falsas. Los falsos positivos son un gran problema incluso para las organizaciones que gastan mucho tiempo sincronizando sus sistemas. Si un IDS crea constantemente falsos positivos, los administradores eventualmente comenzarán a ignorar el sistema. Los honeypots obvian este problema porque toda actividad que se realice con ellos es por definición no autorizada. Esto hace que las organizaciones reduzcan las alertas falsas si es que no las eliminan.

Falsos negativos: Las tecnologías IDS también tienen dificultad para identificar los ataques o comportamientos desconocidos. Además, toda actividad con un honeypot es anómala haciendo que se destaquen los ataques nuevos o previamente desconocidos.

Recursos: Un IDS requiere demasiados recursos de hardware para seguir el tráfico de la red de la organización. Cuando una red aumenta en velocidad y genera más información, el IDS tiene que aumentar para no quedarse rezagado. (Administrar

toda esa información también demanda muchos recursos). Los honeypots requieren recursos mínimos, incluso en las grandes redes. De acuerdo con Lance Spitzner, fundador del Proyecto Honeynet, se puede utilizar una computadora Pentium con 128 Mb de RAM para monitorear millones de direcciones IP.

¿Como hacer un HoneyPots?

Sí nunca ha trabajado con honeypots antes, y quiere aprender más, recomendaría que comience con un simple honeypot de baja interacción, como el KFSensor (<http://www.keyfocus.net/kfsensor/> Versión Trial) o Specter (<http://www.specter.com/default50.htm> Versión Pago) para usuarios de Windows, o Honeyd para usuarios de Unix. Incluso hay un "Kit de herramientas Honeyd para Linux" (Honeyd Linux Toolkit <http://www.tracking-hackers.com/solutions/honeyd/>) para el fácil uso del Honeyd en computadoras Linux, otra alternativa es PatriotBox como servidor honeypot, de (<http://www.alkasis.com/?fuseaction=products.submenu&id=48> versión pago).

Los Honeypots de baja interacción tienen la ventaja de ser más fáciles de usar con poco riesgos, ya que contienen la actividad del atacante. Una vez que haya tenido la oportunidad de trabajar con soluciones de baja interacción, puede tomar las habilidades y el conocimiento que haya desarrollado y trabajar con soluciones de alta interacción.

¿Quieres crear tu propio HoneyPots?... Hazlo!!!

Ambientes Windows:

Paso 1.

Bajarnos el entorno Perl (ActivePerl 5.8.6) para windows

<http://www.activestate.com/Products/Download/Download.plex?id=ActivePerl> allí encontrará varios formatos, se recomienda bajar el .zip (<http://downloads.activestate.com/ActivePerl/Windows/5.8/ActivePerl-5.8.6.811-MSWin32-x86-122208.zip> , una vez descargado ejecute en línea de comando installer.bat, allí le presentará una serie de opciones que puede dejar por defecto.

Paso 2.

Añadir el directorio Perl al PATH. Entre en Mi PC con el click derecho entre en opciones avanzadas, variables de entorno y agregue lo siguiente:
c:\>path=%path%;c:\perl (o donde haya instalado el Perl).

Paso 3.

Bajarnos el entorno Python (ActivePython 2.4.0)

<http://www.activestate.com/Products/Download/Download.plex?id=Activepython>
Una vez instalado el SO (Sistema operativo) estará en la disponibilidad de ejecutar ficheros .py.

Ahora bien, es imprescindible equipar una máquina diseñada para ser atacada, con un sistema de seguridad. en caso contrario podría volverse en nuestra contra.

Paso 4

Bajarnos un script (free) desde

<http://www.megamultimedia.com/arroba/arroba78/descargas/intrusos.zip> de nombre windog, q peso solo 3k y es la mínima expresión de un honeypot, además de configurable es sencillo de instalar y su código es bastante entendible, solo basta tener ciertos conocimientos de programación.

Una vez descomprimido encontrarás 2 ficheros perl (sendmail.pl y telnet.pl) como tambien un readme del autor del script. Sí el entorno Perl esta corriendo solo tendremos que pulsar dos veces sobre cualquiera de los ficheros nombrados anteriormente y aparecerá una consola indicando que el script esta corriendo. NOTA: "Debemos estar atentos que no exista ningún servicio corriendo en los puertos que se indican en el script (23 si se ejecuta el telnet.pl o el 25 sí se ejecuta el sendmail.pl)"

Paso 5

Comprobemos que esta funcionando el script
Pulemos sobre telnet.pl dos veces
ahora por medio de la consola hagamos un telnet
telnet localhost

aparecerá algo similar a esto:
Unix(r) System V release 4.0
(Brooder)
Login:

Estará en espera que se introduzca el login y contraseña de acceso al sistema. El mensaje que muestra por consola puede ser modificado a gusto. Todos los intentos de login y contraseña quedarán reflejados en la ventana en la que se levanto el script y nunca serán válidos porque no esta programado de esa forma, además de que no hay servicio alguno que validar. Sí dejamos corriendo esto por un tiempo con vista a internet se observará como alguien intentará entrar por ese servicio ficticio.

Para modificar el banner que sale al principio indicando que el visitante acaba de conectarse a un sistema Unix, abre el fichero telnet.pl y modifica la línea:
\$banner = "\n\r\n\rUnix (r) System V Release 4.0 (brooder) \n\r";

Para el fichero sendmail es exactamente igual.

Sí usas el fichero sendmail.pl y al hacer conexión via telnet solo reconocerá el comando HELO y reponderá con un saludo y la dirección IP del visitante que acaba de conectarse.

Los script son tan sencillos que podemos hasta cambiar el puerto, si deseas trabajar con el puerto 23 solo debes cambiar el valor de la variable \$port, ejemplo:
\$port = 22;

** Miscelaneas

Recuerda que la verdadera potencia de los servidores trampa consiste en saber atacar al Script Kiddie donde más les duele. Ten presente que deberás combinar diferentes herramientas de analisis y monitoreo para sacar un verdadero provecho de un HoneyPots.

Para la configuración de un HoneyPots en Linux, lo dejaremos para otra oportunidad.

Fuentes consultadas:

HoneyNet Español
<http://his.sourceforge.net/trad/honeynet/>

Zonavirus

http://www.zonavirus.com/Detalle_Articulo.asp?Articulo=108

Symantec

http://www.symantec.com/region/mx/enterprisesecurity/content/framework/LAM_2371.html

Datafull

<http://www.datafull.com/datahack/informe.php?id=255>

Forzis

<http://www.forzis.com>

Tracking-Hackers

<http://www.tracking-hackers.com>

Bibliografía

Lanze Spitzner, Honeypots, Tracking Hack-ers, Addison Wesley, 2003.

The Honeynet Project, Know Your Enemy. Revealing the security tools, tactics, and motives of the blackhat community, Addison Wesley, 2002.

The Honeynet Project, Know Your Enemy. Honeynets, Noviembre de 2003.

The Honeynet Project, Know Your Enemy. GenII Honeynets, Noviembre de 2003.

Eduardo Gallego Revilla, Desarrollo de una Red Virtual de Señuelos para la Detección yAnálisis de Intrusiones en Sistemas Informá-ticos, ETSI de Telecomunicación, UPM. Di-ciembre de 2003.

Ryan C. Barnett, Monitoring VMware honey-pots, Septiembre de 2002.

The Honeynet Project, Know Your Enemy. Hot Zoneing, Febrero de 2003.

Lanze Spitzner, Honeypot Farms, Agosto de 2003.