Tecnologías innovadoras para la detección y la comparación de los ataques informáticos

Versión traducida y editada del artículo original "Tecnologie innnovative per il rilevamento e il contrasto degli attacchi informatici", que aparece en el *Notiziario Tecnico Telecom Italia*, anno 14, no. 1 (Giugno 2005). La Redacción de esta revista, a través del grupo de Asistencia Técnica de ETECSA, ha cedido amablemente los derechos para su publicación en *Tono*.

Por Ing. Gerardo Lamastra e Ing. Luca Viale

Introducción

Los sistemas que realizan el levantamiento y la prevención de las intrusiones —(IDS) *Intrusion Detection System* / Sistemas de Detección de Intrusos— se han convertido, en los últimos veinte años, en uno de los elementos fundamentales de la arquitectura de seguridad de una red informática. Un Sistema de Detección de Intrusos es el similar informático de los sistemas de alarma que protegen una habitación o un banco. Como un verdadero sistema de alarmas, un IDS se caracteriza por los sensores para el sondeo de los hechos no deseados o de las condiciones anómalas en el ambiente monitoreado; el sistema de recogida y de correlación de las informaciones suministradas por los sensores para decidir si es efectivamente necesario emitir una señalización; y el mecanismo de señalización, típicamente multimodal —por ejemplo, la sirena es un aviso enviado a las fuerzas del orden—.

También los Sistemas de Detección de Intrusos se caracterizan fundamentalmente por los mismos componentes y, como los sistemas de alarmas tradicionales, los IDS al ser automáticos pueden ser abordados si se tiene un conocimiento muy profundo de cómo están colocados y configurados.

El objetivo de este artículo es presentar la temática a través de un análisis detallado del estado del arte, que ilustre, por lo tanto, los resultados obtenidos de la búsqueda desarrollada en el laboratorio Be-Secure de Telecom Italia Lab.

La evolución de los Sistemas de Detección de Intrusos

El año 1987 puede considerarse como el período en que se inicia públicamente el tratamiento de la tecnología de sondeo de las intrusiones; y el trabajo de D. Denning [1], el artículo científico de referencia sobre este importantísimo tema. Existen numerosos sistemas comerciales de Detección de Intrusos y varios proyectos desarrollados por la comunidad *Open Source*; pero aún son muchos los problemas por resolver y las oportunidades de investigación son significativas.

Desde el momento que los IDS surgieron como **sistemas de protección** para una infraestructura limitada, es natural que su ámbito tradicional sea aquel del *data center* o de la Intranet empresarial. El desarrollo reciente de las redes informáticas caracterizadas por un aumento casi exponencial de las bandas disponibles, un crecimiento de la conectividad de tipo *always on* y de la movilidad gracias al uso de las tecnologías celulares de nueva generación e inalámbricas, conlleva siempre un aumento de los límites a proteger. El IDS se convierte en un componente estratégico que debe operar en diferentes niveles, a partir del cliente inexperto pasa por la clientela **gran cliente** de tipo tradicional hasta llegar a involucrar directamente al gestor de la infraestructura y, luego, a los grandes operadores nacionales de telecomunicaciones.

Los Sistemas de Detección de Intrusos se originaron en un ámbito principalmente militar, aunque estaban muy relacionados con la actividad de algunos investigadores universitarios cuyos trabajos versaban sobre la seguridad de la información. En 1980, el artículo de J. Anderson [2] introdujo el problema del monitoreo de los sistemas informativos, en específico, los utilizados para las operaciones militares; poco después D. Denning comenzó a trabajar sobre los Sistemas de Detección de Intrusos también gracias al apadrinamiento de diferentes entidades gubernamentales. A finales de los años ochenta, muchas estructuras universitarias tenían un proyecto centrado en el tema de la detección de intrusos, por ejemplo, el Haystack de la universidad de Davis en California [3], el IDES en el *Computer Science Laboratory* del SRI [4] y el proyecto IDIOT [5] en el CERIAS de la universidad de Purdue.

Al inicio del decenio de los noventa, la red Internet salió del contexto puramente académico-militar convirtiéndose en la más grande estructura informática del

planeta, y la problemática de la seguridad pasó a segundo plano respecto a aquella de la usabilidad. Además, la mayor parte de los ataques fue asignada a una élite exclusiva y todavía no llegaba a todos. Se considera que el nivel de seguridad que ofrece un *firewall* sería indispensable para proteger la propia infraestructura. Mientras tanto, con una tendencia bastante típica, algunos grupos de investigación que tenían productos y prototipos interesantes de Sistemas de Detección de Intrusos, comenzaron a evolucionar en empresas *start up* —Haystack Labs, que comercializa el producto homónimo de NFR (*Network Flight Recorder*), aún presente en el mercado o Network ICE, fundada por R. Gram, hoy líder del área de I+D de *Internet Security Services* (ISS)—. Al mismo tiempo, los ataques fueron más difíciles de detectar, más fáciles de implementar y más al alcance de todos. Se hace evidente el fracaso de la seguridad con el uso de un solo dispositivo y, por lo tanto, los IDS comienzan a convertirse en elementos cada vez más esenciales de una infraestructura de protección cuyo paradigma es la llamada *Defense in Depth* [6].

También la tecnología IDS comienza a desarrollarse por otras vías. Aparecen los primeros sistemas que integran el *firewall* al IDS, la línea de demarcación que separa los sistemas de detección de los de contramedidas se vuelve más sutil. Así, de una parte los *firewalls* comienzan a integrar los mecanismos de inspección del tráfico cada vez más sofisticados, mientras que los IDS aprenden a responder directamente a los ataques detectados transformándose en sistemas de prevención.

La aparición de los ataques de tipo distribuido y los efectos drásticos del número creciente de virus que se propagan en la red, presagian la posibilidad de un enorme *black out* informático. Surgen, de este modo, los primeros sistemas que fueron ideados explícitamente para la protección de la infraestructura como los *routers* y otros sistemas de interconexión. Las empresas más importantes en el sector informático como la Cisco, Symantec, Internet Security Systems invierten cifras considerables para adquirir la tecnología de varias *start up* con el objetivo de brindar en su oferta de seguridad también un Sistema de Detección de Intrusos.

Los años sucesivos al 2000, se caracterizan por un relativo estancamiento, al menos en el sector de la investigación y el desarrollo. La tecnología dominante está representada por los sistemas de Network-IDS, que operan analizando con extremada optimización los flujos de paquetes que atraviesan los perímetros empresariales. La atención de los productores se desvía hacia la problemática de la administración, centrada en la necesidad de reducir la enorme cantidad de datos generados por estos sistemas. Comienzan a difundirse dudas sobre la validez efectiva de este tipo de tecnología [7]. Muchas empresas compran las sondas y las instalan pero la inversión necesaria para mantener un verdadero grupo interno que se ocupe del monitoreo resulta prohibitiva. A menudo, los costos del manejo de un servicio de seguridad son insostenibles a menos que se posea una tecnología propia de IDS y un equipo propio para la puesta en marcha de las actualizaciones necesarias al sistema para el reconocimiento de nuevos ataques.

Hoy, la tendencia de los productores es proponer soluciones de tipo *All in One* —todo en uno— que integren en una sola aplicación *firewall,* IDS y antivirus. Este tipo de mecanismo desplaza el centro de gravedad de la tecnología de los sistemas pasivos hacia tecnologías de prevención IPS —*Intrusion Prevention System*—. Los IPS, a diferencia de los sensores que operan en modo pasivo, se vuelven elementos activos de la infraestructura, actúan directamente en los niveles 2 y 3 de la secuencia protocolar, para intervenir sobre el tránsito de los paquetes, con un nivel de análisis notablemente más complejo con relación al del *firewall* tradicional. Su adopción, sin embargo, es contrastada por los efectos colaterales en relación con su elevada complejidad —debido, en muchos casos, a la necesidad de mantener el estado de las sesiones y de utilizar algoritmos más complicados respecto a los tradicionalmente empleados en el *firewall* y el antivirus— que podrían obstaculizar la misma infraestructura. Si, en efecto, un dispositivo de este tipo debiese por algún motivo volverse inoperante, la subred protegida estaría en riesgo de ser desconectada.

También, la tecnología basada indirectamente en el enfoque de tipo *network based* transita por una etapa difícil, rigurosa, enmarcada entre la disyuntiva de los sistemas antivirus y del siempre más difundido *firewall,* los sistemas IDS de tipo *host based* logran con dificultad difundirse seriamente. Predomina la tendencia del enfoque integrador *firewall* e IDS, especialmente en el puesto de trabajo donde es posible obtener con eficacia retroalimentación directa del cliente. Muchos administradores de sistemas continúan con dudas acerca de la instalación de un IDS *host based*, a causa del elevado potencial computacional que gravaría el sistema.

Por otra parte, los grandes administradores de infraestructuras tienen la necesidad siempre mayor de dotarse de sistemas específicos para proteger sus recursos tecnológicos. Los instrumentos estándares para la detección de intrusos se adaptan mal a este propósito. En particular,

para la previsión de las transformaciones de las redes telefónicas tradicionales en una red basada en el protocolo IP. Uno de los aspectos más críticos desde el punto de vista del administrador de red está representado por la problemática relacionada con los protocolos de encaminamiento. Si bien algunos protocolos integran determinadas características de seguridad —como la posibilidad de autenticar estrechamente las transacciones entre los encaminadores— o existen soluciones específicas a nivel de los equipos —como la posibilidad de realizar un filtrado sofisticado de los paquetes—, la mayor parte de los administradores de redes tiende a no utilizar estos mecanismos. Los motivos son múltiples: a menudo el potencial computacional es elevado —y por lo tanto se reduce la banda administrada por el equipo— o pueden crearse problemas de interoperabilidad entre las soluciones de los diferentes productores hasta, incluso, como ocurre en los *peer* BGP, la habilitación y la configuración de la funcionalidad de seguridad requiere la colaboración de diferentes operadores.

La actividad desarrollada a partir de 2003 en el grupo de seguridad informática de Telecom Italia Lab (Be-Secure) se concentra en dos aspectos fundamentalmente:

La evidencia de que los sistemas tradicionales de IDS no resultan siempre adecuados respecto a las exigencias de los operadores de telecomunicaciones. De hecho, estos sistemas han sido ideados para ámbitos de tipo Intranet/Corporativo. La infraestructura de red de un gran operador de telecomunicaciones requiere en cambio amplia flexibilidad al tener que administrar una multiplicidad de tipologías de redes, a partir de la clientela residencial bajo la forma de ADSL, hasta la oferta para empresas con requisitos específicos de seguridad como los bancos o la salud. Existe, por lo tanto, la dorsal de transporte —el *backbone*— que tiene obviamente las características y peculiaridades diferentes con relación a las redes convencionales. El uso de una solución estándar es difícil de aplicar.

La constatación de que el control de los costos asociados al IDS es bastante problemático. Más allá de los costos iniciales de las sondas, los IDS son sistemas que requieren una actualización continua. Si se observa el número promedio reciente de vulnerabilidades nuevas halladas por año [8], se nota claramente una tendencia en concordancia con la ley de Moore, la misma ley que caracteriza en general la difusión de todo aquello que es *computing technology*. Además, más allá de los costos debidos a la actualización, es necesario tener en cuenta también los costos de funcionamiento lo más confiable posible en el sistema de detección. Es bien conocido que el número de alarmas falsas generadas por los sensores puede resultar elevadísimo, sobre todo, en ambientes muy heterogéneos o en aquellos donde sea difícil operar una configuración muy fina de los sensores. Cada alarma falsa genera un costo debido a la necesidad de administrar, aunque de forma mínima, el hecho; pero tiene un efecto mucho más negativo y menos medible relacionado con la tendencia, por parte de quien ejecuta el monitoreo, de valorar todos los hechos como alarmas falsas y así, paradójicamente, un sensor IDS acaba siendo utilizado en reiteradas ocasiones como un instrumento para el análisis posterior de un hecho peligroso ocurrido en la red, en vez de ser utilizado como la primera señal de alarma para darse cuenta de que ocurre algo negativo.

El objetivo de la actividad de investigación desarrollada en Telecom Italia Lab en el último bienio ha sido crear, donde fuese posible, una serie de soluciones tecnológicas focalizadas en la detección de las intrusiones en los diferentes dominios conceptuales. Esto es, a nivel de sistema (*host*), a nivel de redes —*wired* y *wireless*— hasta llegar al nivel del sistema de infraestructura. La actividad de investigación ha sido distribuida en tres proyectos: EAGLE —*Innovative Intrusion Detection*—, RDS —*Routing Detection Security*— y WIDS — *Wireless Intrusion Detection System* —. Se han obtenido muchos resultados interesantes, ya sea en lo que concierne al aspecto innovador —que ha dado lugar a numerosos pedidos patentados— o en lo que concierne a los prototipos de los sistemas —que ha visto la realización de sistemas en funcionamiento, en laboratorios o durante pruebas de campo particulares—.

Estado del arte e investigación científica

Los Sistemas de Detección de Intrusos ya no son una tecnología innovadora, sino desde cierto punto de vista, se pueden considerar relativamente maduros a partir del momento que numerosos productores ofrecen soluciones comerciales. No obstante, los IDS requieren una competencia superior por parte del personal de administración respecto

a otros equipos de seguridad, y una puesta en marcha no optimizada puede generar un número elevado de hechos inútiles. Una valoración incorrecta de las prestaciones necesarias podría volver al sistema incapaz de responder de modo adecuado. Además, muchas veces, es difícil comprender qué ha ocurrido a partir de las indicaciones suministradas por el IDS.

Estas problemáticas del uso tienen un notable impacto práctico además del económico, lo cual está fuertemente amplificado en el contexto de grandes instalaciones. Es importante comprender cómo valorar, de modo apropiado, al menos los aspectos esenciales que caracterizan un IDS para ejecutar una correcta selección sobre la tecnología a emplear o, valorar a fondo, cuáles son las ventajas que cierto mecanismo puede suministrar con efectividad. Las métricas más importantes para valorar un IDS son tres:

Número de Falsos Positivos (#FP): un falso positivo es un hecho que el IDS considera significativo y que, por el contrario, no tiene ningún efecto real sobre el sistema. Podría derivarse, por ejemplo, de la identificación de un intento de ataque hacia una máquina que no es vulnerable del hecho que el paquete peligroso podría, por cualquier otro motivo, no alcanzar nunca el sistema de objetivos o más simple de un verdadero error de detección.

Número de Falsos Negativos (#FN): los falsos negativos son hechos dañinos que el sistema no logra identificar. Esto ocurre solamente porque se realiza un ataque que el sistema no conoce o porque el ataque está enmascarado de algún modo. La fragmentación es uno de los mecanismos **clásicos** para aproximarse a un IDS.

Capacidad de carga: a diferencia de los dos factores indicados anteriormente, relacionados con la eficiencia del funcionamiento del sistema, la capacidad de carga es una medida de la eficiencia. De hecho, es una valoración del máximo de carga que el IDS puede sostener manteniendo la capacidad de ejecutar el análisis. En el contexto de los sistemas de la red, se puede medir en términos de byte/s o en términos de paquetes. Incluso es importante disponer de ambos valores para tener un estimado total del sistema. En el contexto de los sistemas *host based* se miden típicamente los recursos sustraídos del IDS huésped. Es evidente que un IDS *host based* que sustrae más del 50 % de los recursos de cálculo y/o memoria disponible al sistema huésped tiene pésimas prestaciones.

Es obvio que en un ámbito estratégico-comercial, la medición relacionada con los costos es más importante que aquella de tipo tecnológico. Si bien este artículo no profundiza en el tema, es bueno observar que el costo asociado a un IDS puede ser considerado como la suma de cuatro contribuciones sustanciales: el costo del hardware necesario, el de las licencias de software, el de las actualizaciones, y el costo asociado a la administración del sistema —es decir, el costo debido a que el sistema es controlado por personal especializado—.

Desde el punto de vista científico, al tratarse de una tecnología anticuada, es oportuno basarse en un enfoque taxonómico para describir mejor los aspectos significativos relacionados con un IDS. La clasificación adoptada se resiente del hecho de que, históricamente, los primeros IDS fueron de tipo *network based*. Sin embargo, la clasificación es suficientemente general como para adaptarse también a los sistemas de tipo *host* y a aquellos pensados con precisión para la protección de las infraestructuras.

La clasificación a la que se hace referencia es a la del CIDF —*Common Intrusion Detection Framework*— [9]. Este resultado es uno de los frutos de la actividad comenzada por T. Luna en la Information Technology Office del ARPA y desarrollada después como esfuerzo dirigido a definir una arquitectura para la interoperabilidad de los IDS.

De acuerdo con esta clasificación, un IDS está compuesto por 4 elementos esenciales, como se ilustra en la figura 1:

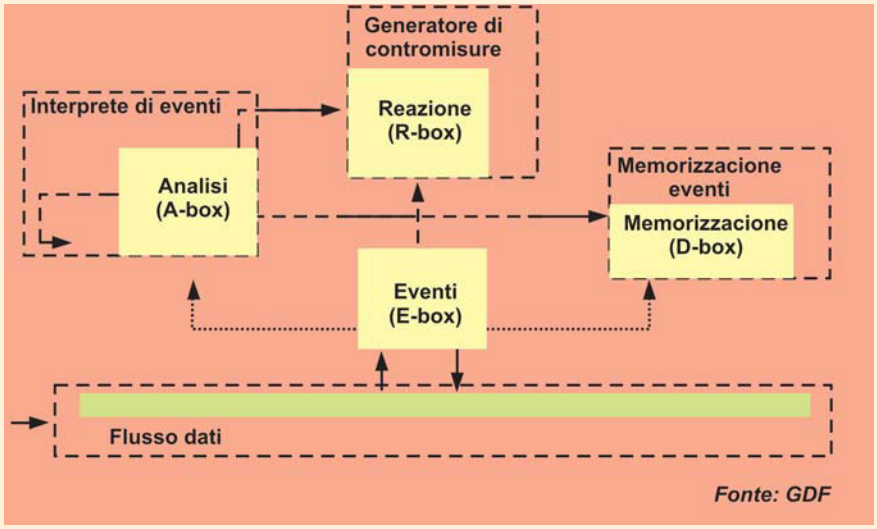

Figura 1 Arquitectura genérica de un sistema IDS

- E-Box: se ocupa de recoger los hechos relevantes del sistema bajo análisis. Este elemento es el verdadero sensor y realiza la interfaz con el sistema de modo pasivo para capturar todas las informaciones significativas para el análisis
- A-Box: es el corazón del sistema que efectúa el análisis y establece si efectivamente el hecho es un ataque o no
- D-Box: es el sistema de memorización que tiene por objetivo mantener la traza de aquello que ha ocurrido
- R-Box: es el sistema de reacción que puede limitarse a emitir una alarma para el personal de monitoreo o ejecutar automáticamente una contramedida apropiada al ataque identificado

De acuerdo con este esquema, es posible clasificar un IDS sobre la base de los cuatro mecanismos esenciales relacionados con las Box que lo componen. Por ejemplo, el mecanismo de recogida de los hechos —la E-Box— puede ser típicamente la red, el único *host*, un sistema híbrido —*host/network*— o un componente representado por un dispositivo específico.

Obviamente en lo que respecta a la metodología de análisis, existe la diversificación más amplia, aunque se pueden individualizar dos grandes paradigmas: *Misuse Detection* y *Anomaly Detection*. En las fases iniciales de la investigación se destinaron diferentes proyectos universitarios para que profundizaran en el uso de una técnica algorítmica específica. Así, están los IDS que trabajan con el mecanismo clásico del *pattern matching* o sistemas que trabajan con la utilización de algoritmos de tipo red neural, sistema experto, genético y estadístico.

Los mecanismos de reacción pueden ser de dos tipos: los pasivos, que se limitan a enviar un aviso al personal que se ocupa de la administración, o los activos que ejecutan, en cambio, alguna contramedida de manera automática. Si bien la adopción de los mecanismos de reacción activos ha llevado al desarrollo de los llamados Sistemas de Prevención de Intrusos (IPS), no hay que olvidar que un sistema de este tipo corre el riesgo de convertirse en un arma de doble filo en el caso de que las contramedidas se desaten frente a un falso positivo que está, por el contrario, asociado a un paquete correcto que se asemeja aparentemente a un paquete peligroso.

En lo que respecta al apoyo de memoria en general, se pueden tener sistemas que utilizan una infraestructura clásica fundamentada en una base de datos relacional o sistemas más sofisticados que, a menudo, evolucionan en productos independientes utilizados como verdaderos sistemas de recogida y correlación, los cuales agregan hechos provenientes de un número de orígenes muy elevado, con frecuencia, muy diferentes. Los hechos son transformados, primero, en una representación **normalizada** y, después, son confrontados con una serie de reglas escritas en cualquier lenguaje con especificación más o menos formal para evidenciar macroanomalías, comportamientos irregulares o cualquier hecho de interés que pueda ser descrito sin desaprovechar el lenguaje suministrado.

Sin dudas, el enfoque más difundido en el ámbito de los sistemas IDS está representado por el uso de sistemas de tipo *network* con una lógica basada en el *pattern matching*. Este enfoque caracteriza casi todas las soluciones comerciales más difundidas —*ISS Network Sensor, SourceFire, Enterasys Dragon*— y muchos sistemas *open source* ya sean nuevos (Snort) [11] o viejos —Shadow [12], NID [13] IDIOT [5]—. En el caso específico, al hablar de *pattern matching*, se entiende por una técnica de individualización de una o más secuencias bien especificadas y contiguas de símbolos, dichas signaturas, en el interior de un flujo continuo de símbolos. Típicamente, los símbolos, en la mayoría de los casos, son en esencia *byte*, pero también pueden ser caracteres *unicode*, *word* de 32 bit u otros.

Existen diferentes modelos formales para encuadrar el problema del *pattern matching*. En la práctica, casi siempre se utilizan las expresiones regulares por su inmediatez de uso y también porque existen algoritmos óptimos aplicables al caso en que se quiera seguir una investigación simultánea de numerosas signaturas sobre un único flujo de datos. No hay que olvidar que, además de disponer de un buen sistema para el análisis de las reglas, es necesario poder codificar los rasgos característicos de un ataque con una signatura específica. Por lo tanto, el uso de un enfoque más simple —aunque sea menos potente— es preferible, en vista de la rapidez con la cual es necesario escribir una signatura luego de que el ataque se hace visible.

En los últimos años, el esfuerzo de desarrollo de los IDS tradicionales en el ámbito *open source* se ha concentrado en torno al proyecto Snort [11] de M. Roesch que se ha vuelto con prontitud uno de los instrumentos de referencia de toda la comunidad de la ICT *security*. Como muchos proyectos *open source* de éxito, Snort ha generado una comunidad muy activa que continúa el desarrollo del sistema. Además, la posibilidad de disponer de un sistema abierto, permite probar rápido la eficacia de un nuevo algoritmo o de un mecanismo de análisis diferente al usual, lo que permite confrontar, de modo inmediato y transparente, el efecto en términos de eficacia y eficiencia.

Uno de los resultados más interesantes aparecidos durante la evolución del proyecto es la optimización del algoritmo de *pattern matching* en el contexto específico del IDS. Las primeras implementaciones del motor de *pattern matching* se basaron en el algoritmo de Boyer-Moore [15]. Este algoritmo es en esencia heurístico que en la mayoría de los casos es, desde el punto de vista computacional, más eficiente que el algoritmo óptimo. El número creciente de signaturas ha conducido recientemente a la adopción de un algoritmo de Aho-Corasic [16] que está creado para el análisis simultáneo de numerosas signaturas de una sola vez.

El enfoque apoyado en el uso de modelos formales —descritos por medio de reglas o por medio de modelos basados en las transiciones de estado— está caracterizado por un discreto éxito en el ámbito académico. Pero se ha encontrado con la dificultad práctica de codificar los ataques a través de los instrumentos suministrados. En particular STAT [17] [18] de la Universidad de California de S. Bárbara es uno de los ejemplos más interesantes: se aprovecha una descripción abstracta del sistema en términos de modelo reducido del estado y se busca configurar los ataques al sistema como conjunto de transmisiones sobre dicho modelo.

Si bien el enfoque sobre la base del paradigma del *Misuse Detection* resulta predominante, existen sin embargo, sistemas implementados con la lógica del *Anomaly Detection* que vale la pena citar. Ir a la investigación de anomalías en el sistema de monitoreo requiere medir una variación de alguna propiedad del objeto bajo examen. Es necesario tener una indicación medible de aquello que se tiene por **comportamiento admisible** y un umbral más allá del cual es lícito señalar un **comportamiento sospechoso**. Los enfoques pueden ser sustancialmente reconducidos a dos líneas principales: *Characteristic Deviation* y *Statistical Deviation*. Un desvío característico ofrece una medida cualitativa —por ejemplo, de manera general el cliente *root* no utiliza el servicio ftp— mientras una desviación estadística es una medida cuantitativa observable —por ejemplo, el tráfico icmp de la red monitoreada no supera normalmente el 15 % del tráfico total—.

En relación con la naturaleza de las informaciones necesarias para efectuar la medición, los sistemas de tipo *Anomaly Detection* pueden clasificarse:

- *Behavioral Based*: se trata de sistemas que miden desviaciones de tipo característico —en ocasiones integrándolas con algunas medidas estadísticas—, los cuales se basan en la creación de un perfil que caracteriza el comportamiento estándar del objeto monitoreado. Generalmente estos sistemas trabajan en dos modalidades distintas: al inicio, durante la fase de adiestramiento (*training*), que muestra el objeto monitoreado para definir el comportamiento admisible. Es esencial que el sistema en esta fase se proteja de cualquier ataque. En una segunda fase aprovecha lo que ha recogido —y eventualmente reajustado de modo oportuno— para identificar las desviaciones de la norma. La tecnología subyacente a tales sistemas está típicamente basada en redes neurales, sistemas *fuzzy* o emulación del sistema inmunitario.
- *Traffic Pattern*: se trata de sistemas muy típicos en el ámbito de *network*/infraestructura. Trabajan mayormente aprovechando modelos estadísticos, que valoran la distribución temporal y espacial del tráfico, al extrapolar las características significativas —por ejemplo, número de conexiones por unidad de tiempo, número de paquetes intercambiados, etc.—. Este prototipo de sistema halla su ámbito aplicativo natural al detectar los ataques de tipo DDoS —*Distributed Denial of Services*— que buscan agotar los recursos disponibles al convoyar de modo coordinado un enorme número de solicitudes hacia los sistemas a atacar. Los perfiles de tráfico, durante estas condiciones particulares, difieren sensiblemente de las situaciones estándares. Por lo tanto, es posible identificar el comportamiento anómalo desde los primeros instantes.

El paradigma de la llamada Inmunología Computacional es uno de los enfoques que ha tenido mayor éxito en el ámbito de la investigación sobre los IDS de tipo *Anomaly Detection*. La inspiración de matriz biológica es tan precisa que los sistemas biológicos están en condiciones de reconocer una cantidad de ataques, incluso, si son desde el punto de vista estructural muy diferentes, y de emitir una respuesta adecuada por medio del sistema inmunitario. El funcionamiento real del sistema inmunitario de un organismo superior es muy complejo al estar en posición de reconocer un **ataque** obvio —en la confrontación con el mismo la reacción está dirigida y casi siempre es eficaz— o bien identificar ataques desconocidos sobre la base de un mecanismo que es similar al nivel de paradigma del *Anomaly Detection*. La idea de aplicar los principios que gobiernan el sistema inmunitario natural a la protección de los sistemas de cálculo, fue expuesta en el año 1994 por el grupo de investigación de S. Forrest de la Universidad de Nuevo México [19], [20]. El objetivo del proyecto es construir un sistema inmunitario artificial para las computadoras [21]. Los IDS realizados en el ámbito del proyecto cubren prácticamente todos los contextos apli-

cativos tradicionales: *Application Based, User Based, Host Based* y *Network Based*. El principio común implementado por tales sistemas está nombrado *Negative Selection*, un algoritmo que realiza la distinción entre aquello que es legítimo —clientes, acciones sobre el sistema operativo, conexiones de red— y aquello que no lo es.

El enfoque basado en el *Anomaly Detection* de tipo *Traffic Pattern* es dominante en el contexto de la seguridad del *backbone*. En ese sector la investigación está mucho menos desarrollada respecto al contexto más típico de la detección de intrusos y los problemas que han sido más estudiados son fundamentalmente dos: por una parte, la tentativa de individualizar y bloquear los DDoS que se realiza con técnicas estadísticas; y por otra, el análisis centrado en los protocolos de encaminamiento. Con la función de distribuir las informaciones relacionadas con la topología de la red, los protocolos de encaminamiento permiten enviar correctamente los paquetes hacia el destino final. En ausencia de informaciones de encaminamiento exactas o en presencia de informaciones falsas, la transmisión de los paquetes a través de la red resulta ineficiente o imposible. Este hecho es problemático por cuanto un eventual ataque a la infraestructura de encaminamiento que regula el funcionamiento normal del *backbone*, podría tener consecuencias muy graves pues el ataque tendría efectos serios también sobre las otras redes interconectadas al *backbone*, que se encontrarían en parte o completamente aisladas del resto de la red.

El estado del arte en este campo no ve emerger por el momento una solución de referencia, incluso, si desde hace un tiempo los proveedores de soluciones de seguridad han dispuesto algunas soluciones que se proponen responder al problema de la seguridad del *backbone*. Estas soluciones, en teoría, se proponen trabajar no sólo a nivel protocolar típico de los *end-point* —del transporte hacia delante—, sino también a nivel de *internetworking* —esencialmente en el nivel 3—. En la práctica, muchas de las soluciones disponibles no intervienen, de manera activa, al actuar en interfase con los sistemas de encaminamiento utilizados en el contexto a controlar, por el contrario, utilizan informaciones derivadas de los equipos mediante la técnica estándar —retiro de variables SNMP de las MIB dedicadas a los aspectos de encaminamiento— u otros recursos de información (*NetFlow*). El enfoque clásico de la *Network Intrusion Detection* es difícil de aplicar en este contexto desde el momento que el uso del *pattern matching* no permite identificar de inmediato una condición de funcionamiento anómala. En la actualidad, algunos productores específicos —en particular Arbor Network y Riverhead— ofrecen soluciones puntuales para determinados problemas, por ejemplo, el contraste de los ataques *Distributed-DoS*, mientras que los mayores productores —en específico Cisco y Juniper— desarrollan interesantes iniciativas de investigación y desarrollo.

Con el discurso sobre el *backbone*, concluye la panorámica acerca del estado del arte. Se ha considerado oportuno limitar la descripción a las soluciones que resultan más interesantes. El panorama ofrecido debe ser suficiente para enmarcar los aspectos relevantes de las tecnologías más difundidas relacionadas con el contexto de detección de intrusos.

De la tecnología al producto: Intranet y *Backbone* IDS

Desde finales de los años noventa, muchas de las soluciones elaboradas, en el ámbito militar o en el académico, comenzaron a implementarse directamente en contextos comerciales siguiendo la lógica más clásica de los productos de seguridad. La necesidad de ofrecer un producto completo requiere considerar otros aspectos que no están unidos a la problemática de la detección en sentido estrecho. En cualquier caso, el paso del laboratorio al ambiente de aplicación evidencia muchos problemas y, también, algunos límites estructurales de las soluciones identificadas en el contexto de la investigación.

La solución más difundida en el ámbito IDS está representada por un sensor de tipo *network-based* que trabaja esencialmente con un paradigma de detección de medidas, en particular con una lógica de tipo *pattern matching*. Todos los productos comerciales más importantes —ISS Proventia™, Sourcefire™, Dragon Enerasys™ o Symantec Manhunt™— implementan este tipo de lógica. Y luego la apoyan con varias tecnologías específicas que intentan reducir la incidencia de los falsos positivos y de los falsos negativos.

El problema de los falsos negativos es el más dramático en el contexto de los IDS. Un falso negativo está, a menudo, asociado a la posibilidad de modificar de modo más o menos arbitrario la forma del ataque sin variar la sustancia. Una clase entera de ataques basados en esta problemática está representada por el uso de la técnica de fragmentación. De hecho, la descompactación en paquetes más pequeños del *payload* aplicativo que contiene el ataque, impide al sistema de *pattern matching* funcionar de modo eficaz. Un enfoque más sutil consiste en el envío de fragmentos que se sobreponen en parte. Desde el momento que el criterio con que vienen tratados fragmentos sobrepuestos, de manera parcial, no es unívoco, es posible que el sensor y el sistema blanco vean dos contenidos diferentes y, por lo tanto, el efecto neto es aquel de

una errata de interpretación de lo que está ocurriendo. La figura 2 muestra gráficamente un ejemplo de lo que puede ocurrir en este caso.

Otra clase de ataques siempre basados en el mismo paradigma está representada por la llamada **denormalización**. Algunos protocolos soportan distintas formas de codificación por el contenido aplicativo. Con el aprovechamiento de las codificaciones alternativas es posible rodear la *signatura* específica por un ataque. Además es imposible insertar las formas diferentes con que puede ser codificado un ataque. Claramente el enfoque ideal consiste en reportar el contenido del paquete en una forma estándar antes de ejecutar el *pattern matching*.

Figura 2 El problema de la reconstrucción del *payload* a partir de fragmentos sobrepuestos

Con el propósito de no prolongar demasiado la descripción, se sugiere el trabajo de Ptacek *et al* [22] para profundizar en los ataques que son posibles hacia un IDS. Hoy la mayor parte de los sistemas comerciales utiliza la técnica de análisis del protocolo para administrar los problemas relacionados con la fragmentación, denormalización y ofuscamiento. Estas técnicas no son innovadoras, sin embargo, el esfuerzo requerido para realizar un sistema robusto y eficiente es notable, y puede ser afrontado sólo en sistemas que sean administrados con la lógica del producto.

Otro problema tradicional de los IDS es el enorme número de alarmas que estos dispositivos tienden a generar, sobre todo, cuando no están bien configurados. También, en este caso, las técnicas de análisis del protocolo pueden reducir bastante los falsos positivos, con la eliminación de todas aquellas alarmas que corresponden a situaciones que no son efectivamente admisibles. Por ejemplo, un paquete sospechoso en una conexión TCP que, sin embargo, no ha completado el 3-*way handshake* evidencia una falsa alarma. El paquete no será procesado jamás por el nivel aplicativo y, en consecuencia, está destinado al fracaso.

El problema de la administración de los falsos positivos ha sido enfrentado de varios modos por los diferentes productores. En muchos casos, la tendencia es ir a sistemas multiniveles que relacionen las informaciones generadas por el IDS con las obtenidas de otras fuentes, típicamente sistemas de escaneo de la vulnerabilidad o, en general, otros sistemas que permiten ejecutar una valoración de los sistemas que se pretenden proteger.

En el contexto *backbone*, los enfoques más comunes son los de tipo *Anomaly Detection*, las tecnologías más importantes son aquellas relacionadas con el análisis estadístico del tráfico. El problema esencial en la detección de los ataques de denegación del servicio está caracterizado por la dificultad de distinguir situaciones de carga elevada debido a una condición específica de servicio —por ejemplo, se piensa en lo que ocurre cuando se busca adquirir una entrada para un gran concierto y el intervalo temporal para ejecutar el pedido es muy pequeño: el número enorme de pedidos en un reducido espacio de tiempo se asemeja grandemente a un *SYN-flood* distribuido—.

Uno de los problemas fundamentales en este ámbito está representado por la necesidad de recoger un número de informaciones relevantes para los flujos. Es evidente que si cada aparato de red exporta estas informaciones de modo propietario, se torna mucho más complejo hacer funcionar cualquier sistema de monitoreo que es uno de los componentes esenciales de cualquier arquitectura tendente a resolver este tipo de problemas. El IETF se ha dirigido en esta dirección y ha hecho suya una específica de Cisco, denominada *NetFlow* [23], que permite a varios administradores de red obtener, en un formato estándar, informaciones sobre los flujos de datos que atraviesan la red. *NetFlow* nace, en principio, para apoyar la actividad de recogida de datos de tráfico y tarifarla. Pero, las informaciones que esto genera pueden ser usadas de modo natural para el análisis de los problemas de seguridad. Hoy, la mayor parte de los aparatos —Enterasys, Foundry Networks, Extreme Networks, Juniper, Riverstone, InMon Networks— integran este protocolo. Además existen varias aplicaciones capaces de interpretar y analizar estas informaciones. Pueden citarse Peakflow™ —creado por Arbor Networks— y Ntop, un producto *open-source* desarrollado en el centro SERRA de la Universidad de Pisa. NetFlow es capaz de capturar un buen conjunto de datos estadísticos, los cuales incluyen entre otras el protocolo, los puertos, el tipo de servicio que combinado suministran, como se ha dicho, una base informativa útil para una vasta gama de servicios.

La entidad medida por NetFlow y el **flujo**, identificado como un conjunto de paquetes entre una fuente y un destino, son reconocidos por los campos: *Source IP Address*, *Destination IP Address*, *Source Port Number*, *Destination Port Number*, *Protocol Type*, *Type of Service* e *Input Interface*.

Las informaciones generadas por NetFlow son transmitidas a través de paquetes UDP, el protocolo ha sufrido varias evoluciones —hoy la

versión utilizada es la 5—. La modificación más importante está representada por la introducción de las informaciones relativas al protocolo de encaminamiento BGP y a las *sequence numbers*. A un nivel más aplicativo, las tecnologías de mayor interés en este ámbito están representadas por Riverhead y Arbor Networks.

La actividad de Investigación y Desarrollo en TILAB

A partir del año 2003, han sido ejecutados en Telecom Italia Lab una serie de proyectos sobre el tema de la detección de intrusos. La motivación principal está vinculada al hecho de que las soluciones propuestas por el mercado, además de ser económicamente onerosas, no son siempre adaptadas, de modo funcional, a un contexto heterogéneo como aquel que caracteriza a un gran operador de telecomunicaciones que requiere una enorme flexibilidad para operar en varios segmentos de red —de la conectividad de la red de banda ancha para clientes domésticos hasta el *backbone*—.

La actividad de investigación se ha articulado en tres proyectos. El proyecto EAGLE se ha centrado en la ideación y la creación del prototipo de nuevas tecnologías que podrían ser utilizadas en el ámbito más tradicional con especial atención a los aspectos de flexibilidad y escalabilidad mencionados anteriormente. El proyecto RDS —*Routing Detection Security*— se ha centrado en las problemáticas relacionadas con la red de *backbone*, tiene énfasis en los ataques que son posibles sobre la infraestructura, en particular, a los protocolos de encaminamiento. En resumen, en el ámbito del proyecto dedicado a la seguridad de las redes inalámbricas —con especial énfasis en el mundo IEEE 802.11 y WiFi— se ha dispuesto estudiar y hacer un prototipo de los componentes definidos para este entorno, que, no obstante, podrían ser integrados en un sistema tradicional de tipo *network*.

Los resultados de la actividad de investigación son notables y pueden ser resumidos del siguiente modo:

- **Patentes**: toda la actividad ha conducido al pedido de seis patentes centradas en la tecnología tipo *network*, dos pedidos de patentes centrados en la tecnología *host*, uno centrado en el contexto WiFi y otro de patente para el contexto *backbone*.
- **Prototipos**: cada proyecto individual ha producido uno o más prototipos, en particular está disponible una solución que funciona para el sistema *network*-IDS. Asimismo, se han integrado al sistema dos componentes de prototipos relacionados con el contexto WiFi. El sistema *host*-IDS, por el contrario, está disponible únicamente en estado de prototipo avanzado y sólo para la plataforma Linux. En lo que respecta al ámbito *backbone*, están listas las sondas para el protocolo OSPF y para el protocolo BGP, además de un sistema definido para la recogida y la elaboración de las informaciones almacenadas por las sondas.
- **Líneas de códigos globales**: por cuanto esta cuantificación puede quedar imprecisa al describir la complejidad efectiva de los sistemas que se han realizado, no obstante, resulta útil suministrar al menos una indicación de máxima sobre su dimensión **cuantitativa**. Existen cerca de 150 000 líneas de códigos para el sensor de tipo red —más 2 000 para el soporte WiFi—, 20 000 líneas para el sistema de tipo *host-based* y otras 20 000 para los sistemas de tipo *backbone*.

A continuación se describen brevemente las características más importantes de cada una de las soluciones.

El sensor *Network*-IDS: nEAGLE™

nEAGLE™ es la solución *network*-IDS desarrollada en el ámbito del proyecto homónimo (Figura 3). El sistema está basado en la tecnología estado del arte en lo que respecta al *pattern matching*. Junto con esta tecnología implementa también tres mecanismos innovadores —en trámite de pedido de patente— descritos a continuación:

- El *Bidirectional Rule Matching* (BRM) es un mecanismo que analiza la respuesta del servidor después de que un paquete ha sido identificado como peligroso. La respuesta, a menudo, permite determinar si el ataque ha tenido éxito o no. De este modo, el número de falsos positivos se reduce drásticamente trabajando a nivel del sensor. También, mientras las reglas que describen

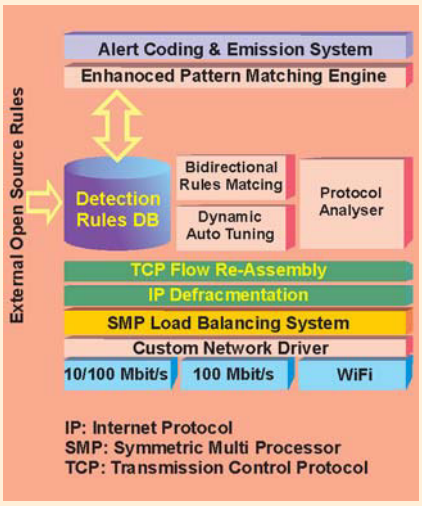

Figura 3 nEAGLE™: esquema de la arquitectura

los ataques requieren de actualización continua, las reglas de respuesta son relativamente estables y están relacionadas con las características del protocolo aplicativo. Un ejemplo de uso del BRM: el éxito de un ataque directo a un servidor Web, que aprovecha la vulnerabilidad de un *script* determinado instalado por defecto, puede ser deducido con cierto grado de precisión por la respuesta del servidor si, de hecho, el servidor responde con un código de error 400 —por ejemplo *Access Forbidden* u *Object not Found*— es extremadamente probable que el ataque no sea efectivo. Un resultado de tipo 200 (OK) seguido de paquetes que contienen elementos típicos de una interacción basada en una *command-line shell*, es una indicación clara de que el ataque ha tenido éxito. Por supuesto, este escenario no tiene como objetivo presentar de modo exhaustivo y profundo el BRM, sino ofrecer una idea de cómo funciona esto en la práctica.

- El *Dynamis Auto Tuning* (DAT) permite al sensor adaptar el juego de reglas a la configuración de la red. De este modo, el sistema está en grado de identificar automáticamente la presencia de un servicio/protocolo —por ejemplo HTTP, FTP, Telnet— en puertos no estándares. Esto permite reducir los falsos negativos debidos a ataques no detectados porque se han dirigido hacia el servidor conectados en puertos no estándares. El DAT utiliza reglas de tipo *pattern matching* o de *protocol analysis* para identificar en un flujo de paquetes bien definidos la presencia de los elementos distintivos de un protocolo aplicativo específico. Frente a tal situación, el DAT reconfigura de manera automática el sensor, para que todas las reglas utilizadas por el protocolo específico sean de otro modo adoptadas por aquella conexión específica.
- El soporte para multiprocesador simétrico es un sistema de *network*-IDS que funciona analizando los paquetes. Para lograr extraer ventajas de la presencia de más CPU son posibles dos enfoques: el *pipeline* o la distribución de la carga sobre un único CPU. nEAGLE™ aprovecha el enfoque del *load balancing*, y reparte entre los CPU disponibles la actividad de detección de los ataques. El algoritmo de *scheduling* que selecciona la actividad determinada a ejecutar sobre cada CPU es de tipo *application based*, a diferencia de la mayor parte de los enfoques estándares que, por el contrario, son de tipo *kernel based* y no pueden, en modo alguno, aprovechar el conocimiento del dominio aplicativo para pilotar las decisiones del *scheduling*. Este enfoque permite obtener prestaciones muy superiores sobre los sistemas SMP respecto a aquellas que se obtendrían con sistemas tradicionales sin tener que recurrir a algún *load balancer* separado.

Además de la adopción de tecnologías específicas innovadoras, el sistema completo ha sido sometido a una programación, centrada en la reducción notable del elevado computacional asociado a la captura de los paquetes. Con la utilización de un sistema *ad hoc* realizado por medio de un *device-driver* personalizado, se pueden evitar desde la raíz los problemas del *livelock* [24] y aprovechar dos enfoques clásicos de la teoría de los sistemas operativos: implementación de canales de comunicación entre *device-driver* y *user space* que sean 0-*copy* y la administración del dispositivo de red a *poling* en vez de con las interrupciones.

nEAGLE™ soporta asimismo un mecanismo para la detección de los ataques específicos para redes WiFi. Este mecanismo está en grado de detectar ataques llevados al nivel físico y *datalink* del *snack* TCP/IP como los *Authentication-Deauthentication, Frame-Flooding, Association-Disassociation-Reassociation, Request-Flooding*. Además de estos mecanismos, la actividad de investigación y desarrollo ha llevado a la realización de una funcionalidad concreta —cubierta actualmente por pedidos de patentes—. Está orientada a la reducción de los falsos positivos y a la individualización de los ataques o tentativas de violación del mecanismo de autenticación para redes WiFi IEEE 802.1x —por ejemplo, THC-LeapCracker [25], ASLEAP [26], 4Way Handshake Start Dos [27]—. Tal funcionalidad está basada en la correlación de las informaciones intercambiadas en la red WiFi con las informaciones recibidas de los elementos de red como *Access-Point* y servidor de autenticación —por ejemplo RADIUS— que realiza la red WiFi.

nEAGLE™ ha sido verificada en laboratorio donde se han obtenido prestaciones verdaderamente notables —hasta en un orden de tamaño superior— en relación con aquellas basadas en la tecnología actualmente disponible sobre los sistemas hardware de bajo costo. En lo referido a la eficacia —falsos positivos y falsos negativos—, nEAGLE™ ha demostrado ser extremadamente válida, ya sea durante las pruebas de laboratorio o durante la actividad de *field trial*. En particular, nEAGLE™ está apta para administrar un número de paquetes superiores a un orden de magnitud respecto a soluciones convencionales —sobre hardware bajo costo en condiciones de sobrecarga— con un aumento de la banda media administrada de 5 Mbits/s acerca de 34 Mbits/s. En relación con el contexto *high end*, al aprovechar

un sistema SMP de dos CPU, nEAGLE logra administrar con detenimiento una banda del orden de Gbits/s.

El sensor HIDS: SPID™

La sonda del sistema SPID —*System Photo Intrusion Detection*— (Figura 4) utiliza un enfoque innovador basado en el paradigma del *Anomaly Detection* para identificar los ataques.

La metáfora en la que se basa está bien descrita por el nombre que lo distingue. De hecho, SPID ejecuta una serie de **fotografías instantáneas** del sistema centrándose en los elementos de uso de las diferentes fuentes disponibles

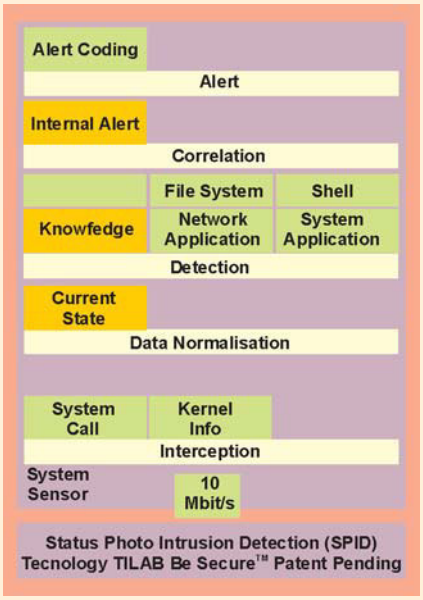

Figura 4 Arquitectura esencial de una sonda SPID

en el único *host*. Por ejemplo, se monitorean las relaciones *parent/child* existentes entre todos los procesos activos, las relaciones entre los clientes, aplicaciones y *files,* aquellas entre las aplicaciones y conexiones de red, etc. Cada aspecto específico del sistema es analizado por un componente *ad hoc* de la base de conocimiento.

Como todos los sistemas de tipo *Anomaly Detection*, SPID trabaja en dos fases. En la primera fase (adiestramiento), el sistema recoge las informaciones que caracterizan el uso normal de *host*. Al final del adiestramiento, la base de conocimiento obtenida de ese modo es normalizada. En otras palabras, se identifican todos los intervalos típicos de variabilidad del sistema, por ejemplo, la utilización de *files* temporales o eventuales periodicidades con las que se utilizan aplicaciones concretas. Estas informaciones extrapoladas de los datos observables, se incluyen en los datos que luego serán usados durante la segunda fase del funcionamiento (análisis). En esta fase SPID confronta el estado del sistema con las informaciones de la base de conocimiento. Si se detecta una anomalía, se emite una señal. Un sistema oportuno de recogida y agregación de estas señalizaciones se ocupa de codificar en una alarma verdadera la anomalía detectada por los componentes individuales que lo constituyen.

SPID está constituido por un sistema de componentes organizados sobre la base de una arquitectura de tres estratos. El estrato más bajo se ocupa de la adquisición de las informaciones del sistema operativo. La técnica adoptada por SPID es aquella de intercepción de los sistemas de alertas. A diferencia de otros sistemas en que se interceptan indistintamente todos los sistemas de alertas para suministrar un modelo que caracterice la ejecución de un dato proceso, SPID se limita a interceptar sistemas de alertas que implican directamente una fuente del sistema operativo en particular: *file*, *socket*, *device-driver*, procesos, identificación del cliente. De este sistema de alertas se registran los parámetros y el proceso que ha ejecutado la solicitud. Con su empleo, el segundo estrato está en grado de construir un modelo analógico reducido del sistema; en la práctica este estrato tiene dos objetivos: traducir el sistema de alertas determinado en una solicitud abstracta de utilización de recurso del sistema operativo y ofrecer al estrato superior una visión abstracta de aquello que es el estado instantáneo del sistema monitoreado. El último estrato es el más complejo y se ocupa de recoger y controlar los esquemas con los que se usan varios recursos. Debido a que cada recurso tiene su especificidad, se han implementado diferentes módulos cada uno con una lógica:

- *User Table*: muestra las relaciones de tipo cliente/aplicativo/*file*, suministra una vista global del comportamiento del sistema recogido directamente a partir del modelo analógico reducido presente en el segundo estrato. El módulo tiene trazas de todas las fuentes que son utilizadas en el curso del funcionamiento del sistema. Además, por medio de contadores adecuados se mide también el número total de instancias de un objeto. Cada vez que aparece un objeto nuevo que no se había visto por el sistema durante la fase de adiestramiento, o bien cada vez que uno de los contadores excede el límite requerido por la fase de adiestramiento, se emite una señal.
- *Process Tree*: traza las relaciones de uso entre las aplicaciones. Es decir, se trata de comprender qué aplicaciones pueden ser ejecutadas a partir de una aplicación dada. La relación en forma de árbol es muy útil para capturar aquellos casos en que por efecto de un *overflow* se crea una *shell* abusiva en el contexto de un servidor de red.
- *Network*: este módulo tiene trazas de las conexiones de red que identifican las aplicaciones que desempeñan el papel de clientes y/o servidor, los puertos sobre los cuales operan y las características estadísticas de la duración de las conexiones.

- *File System*: este módulo tiene trazas de todas las modificaciones del sistema, en particular, puede detectar *mount/unmount* no correctos, la creación o la cancelación de un elevado número de *files*, la cancelación de *files* que están todavía abiertos o el excesivo número de vínculos simbólicos asociados a un *file*. También, en este caso, el uso de los contadores y la aplicación del procedimiento de normalización permiten discriminar las acciones admisibles de aquellas que no lo son.

SPID ha sido verificado en laboratorio sobre algunos servidores específicos de la red interna. Por el momento es considerado un sistema incompleto. Los resultados obtenidos durante la experimentación son positivos. En especial SPID parece inmune a algunos ataques que, por el contrario, caracterizan a otros sistemas basados en la *syscall interception* y en el paradigma de *Anomaly Detection*.

La Tecnología RDS

El proyecto RDS —*Routing Detection System*— es creado con el objetivo de realizar un Sistema de Detección de Intrusos dedicado por completo al análisis de los ataques dirigidos al protocolo de encaminamiento. Una de sus características esenciales es su capacidad de adaptación al ambiente de uso recogiendo de modo diferente, según el protocolo de encaminamiento monitoreado, todas las informaciones que determinan el comportamiento de la red. Tal enfoque permite identificar de modo preventivo, al verificarse, hechos específicos que pueden estar asociados a mal funcionamiento o a verdaderos ataques que, directa o indirectamente, provocan el mal funcionamiento de la infraestructura.

La arquitectura del sistema (Figura 5) prevé que las sondas sean intrusas en grado mínimo; pero en condiciones de recibir todas las informaciones de encaminamiento. El sistema, en su conjunto, está caracterizado por una estructura jerárquica en la cual las operaciones de detección están segmentadas en varios niveles en función de la complejidad y de la necesidad de entrada proveniente de otros componentes. La confiabilidad, que en el contexto del *backbone* es una

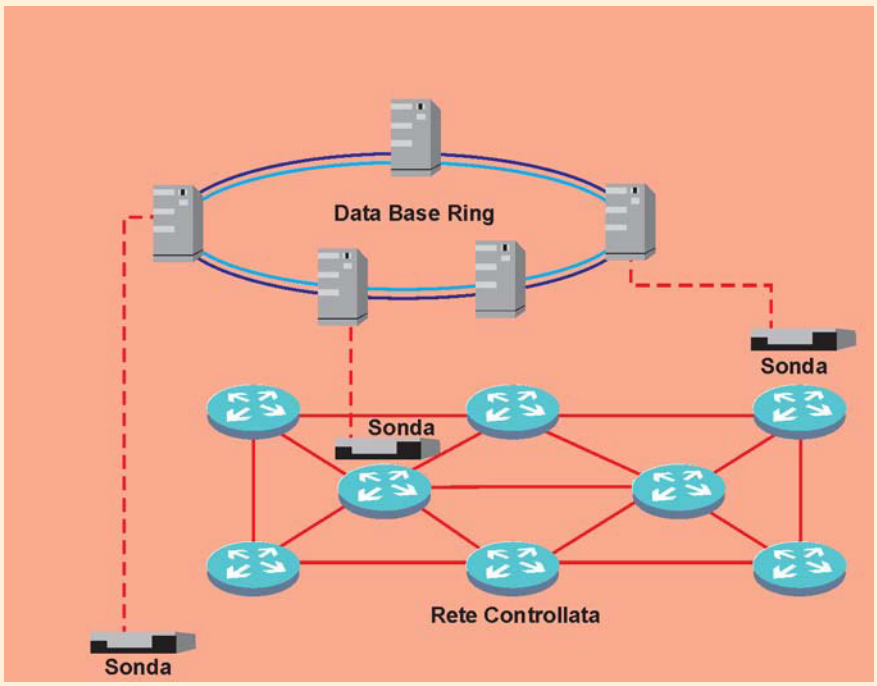

Figura 5 Arquitectura esencial del sistema RDS

característica irrenunciable, está garantizada por la presencia de más elementos redundantes no sólo a nivel de las sondas, sino también de otros componentes. Particular atención tiene la estructura de varios módulos del sistema que presentan una configuración, la cual facilita la realización e integración de nuevas lógicas de detección y nuevas tipologías de sondas para protocolos no consideradas aún en el sistema.

El sistema consta de dos niveles conceptuales: las sondas que son objetos especializados capaces de hacer interfase según diferentes modalidades con los procesos de encaminamiento, y el llamado *Data Base Ring* que es un sistema de alta confiabilidad que recibe las informaciones de las sondas, las procesa según lógicas codificadas y memoriza los datos obtenidos en una base de datos. Un ataque al sistema de encaminamiento puede provocar la **desaparición** momentánea de algunas zonas de red a monitorearse; la estructura de anillo garantiza que, incluso en condiciones críticas, el sistema pueda continuar su funcionamiento correcto. La presencia de más elementos capaces de recoger las informaciones, permite también subdividir la carga de trabajo y garantizar, de este modo, un buen nivel de escalabilidad. Obviamente, el operador puede interrogar la base de datos a través de una interfase clásica GUI. Hoy RDS es capaz de funcionar con dos protocolos de encaminamiento: OSPF —*Open Shortest Path First*— y BGP —*Border Gateway Protocol*— descritos a continuación:

◆ **Sonda OSPF**: ejecuta el análisis de los paquetes OSPF, ya sea mediante una lógica clásica de *sniffing* del tráfico o haciendo interfase directamente con el proceso de encaminamiento que une la sonda a la red monitoreada. La sonda es capaz de controlar varias **anomalías** entre las cuales pueden mencionarse:

-Redes de acceso sobre las que está activo OSPF: control que permite identificar y señalar la presencia o la aparición de redes de acceso sobre las cuales está activo el protocolo OSPF; tal situación no es necesariamente indicativa de un ataque pero señala la presencia de un punto de ataque potencial.

-Creación de nueva adyacencia: este hecho señala la creación de una nueva adyacencia sobre una red en la cual OSPF está activo y puede representar el inicio de un ataque.

-Detección del ataque MAXSEQ-MAXAGE y del ataque LOWCOST [28].

-Variaciones del número de ruta: este control sirve para señalar modificaciones en aumento o disminución de la tabla de encaminamiento. El hecho especialmente, en el caso de variaciones significativas o de numerosas variaciones en la unidad de tiempo, es a menudo indicación de un ataque.

-Aparición de rutas no pertenecientes al propio dominio IP: la inyección de rutas no pertenecientes al dominio es un error de configuración. La detección ocurre mediante el análisis cruzado de la tabla de encaminamiento OSPF con los *files* de configuración hechos por el operador.

-Detección de anomalías sobre la misma sonda: el sistema es capaz de detectar la presencia de criticidades en la conexión con las sondas que pueden indicar un mal funcionamiento o el inicio de un ataque a la infraestructura.

◆ **Sondas BGP**: el análisis de los paquetes BGP está basado en un verdadero sistema de encaminamiento insertado directamente dentro de la sonda. Unido a un mecanismo de *protocol analysis* desarrollado especialmente para el protocolo BGP. El diálogo entre la sonda y los aparatos de red se realiza mediante una sesión de *peering* BGP. La sonda es capaz de trabajar en configuración iBGP por cuanto permite transportar información de encaminamiento con un nivel de detalle superior. En este caso no se ejecutan variaciones sobre el campo *Next Hop* de los mensajes BGP y, también, se utiliza el campo Local Preferente. Al mismo tiempo, en detalle, la sonda es capaz de ejecutar muchos controles entre los que están:

-Detección de un excesivo intercambio de mensajes entre los puntos.

-Detección del *flapping*: se refiere a una modificación muy frecuente de las informaciones presentes en la tabla de encaminamiento, ya sea en *update* o en *withdrawn*.

-Distribución de los prefijos/redes no autorizados, no asignados y de uso privado.

-Detección de los conflictos MOAS —*Multiple Origin AS*— [29]: este control permi-te verificar la correspondencia de los AS de origen de un *update* con lo existente en los archivos del RIPE y evidenciar discrepancias y/o conflictos.

-Detección de anomalías sobre los *Next-Hop*: surgidas, desaparición, modificación de los *Next-Hop* asociadas a las rutas.

-Detección de ataques a la sonda: realización de un *honeypot* BGP capaz de identificar eventuales tentativas de instauración de sesiones BGP no autorizadas.

-Detección de anomalías en el campo AS-Path: análisis para valorar las variaciones significativas en el tiempo.

-Monitoreo de las rutas de los AS clientes: monitoreo de la accesibilidad de las redes de los AS clientes.

El sistema RDS ha sido verificado en los laboratorios de Telecom Italia Lab, para tal finalidad se utilizó una red simulada consistente en 18 encaminamientos distintos. Los experimentos han sido exitosos y se ha iniciado con un primer *field trial* con las sondas posicionadas en la red de Telecom Italia Lab. En este ambiente las sondas han evidenciado buena capacidad de detección de las anomalías, ya fuese en ámbito OSPF ó BGP.

Conclusiones

Con este artículo se pretende ofrecer una panorámica, lo más amplia posible, en relación con el ámbito de la detección de intrusos informáticos. En ese sentido, por tratarse de un tema muy extenso, se limita a brindar un punto de partida para profundizar en las características de las diferentes tecnologías disponibles. El problema de la detección es crucial en el contexto de una arquitectura de seguridad construida según el paradigma de la *Defense in Depth,* porque es evidente que sólo a través de un Sistema de Detección

verdaderamente eficiente es posible pensar en la adopción de tecnologías automáticas de respuesta y protección. La investigación desarrollada en Telecom Italia Lab sobre este tema, en el pasado bienio, se ha centrado en la problemática de la eficacia y la eficiencia en la capacidad de hacer el proceso de detección lo más exacto posible con reducción de los falsos positivos y negativos, con el aprovechamiento máximo del desempeño del hardware disponible y la intervención humana centrada sólo en los problemas verdaderamente significativos. La experiencia madurada en el curso de los diferentes experimentos indica que estos aspectos son fundamentales para permitir un uso óptimo de las tecnologías IDS también en el contexto de las grandes redes de telecomunicaciones.

Traducción: Lic. Joaquín A. Ferrer Álvarez

Bibliografía

[1] Denning D. E: "An Intrusion Detection Model" IEEE Transactions on Software Engineering, Vol SE- 13, No. 2, february 1987, 222-232

[2] Anderson, J: "Computer Security Threat and Surveillance", Tech. Rep. James P. Anderson Co, Fort Washington, Pa, 1980

[3] Smaha, S. E: "Haystack. An Intrusion Detection System", in Proc. of the IEEE 4th Aerospace Computer Security Applications Conference, Orlando, FL, Dec. 1988

[4] D.E. Denning, Neumann P.G.: "Requirements and Model for IDES – a Real-Time Intrusion Detection Expert System", Tech. Rep., Computer Science Laboratory, SRI International, 1985

[5] Kumar S., "Classification and Detection of Computer Intrusion", M.S. Thesis, Computer Science Dep., Purdue University, Agosto 1995

[6] "Information Assurance through Defense in Depth", Directorate for Command, Control Communications, and Computer Systems, U.S. Department of Defense Joint Staff, February 2000

[7] Gartner Group: "Gartner Information Security Hype Cycle Declares Intrusion Detection Systems a Market Failure", 2003 Press Release www.gartner.com/5_about/press_releases/pr11june2003c.jsp

[8] Cormack, A.: "Moore's Law of Computer Security", Terena Networking Conference 2002, 3-6 june 2002, Ireland.

[9] Stanford-Chen, S.: "Common Intrusion Detection Framework", http://seclabs.cs.ucdavies..edu/cidf/

[10] Verwoerd, T. and Hunt, R.: "Intrusion Detection Techniques and Approaches", Computer Communications, Elsevier, U.K, Vol. 25, No.15, Spetember 2002, pp 1356-1365

[11] Roesch M: "The Snort Intrusion Detection System", www.snort.org/

[12] Naval Surface Warfare Center – Dahlgren Lab: "The Shadow Intrusion Detection System", www.nswc.navy.mil/ISSEC/CID/

[13] Department of Energy. "The NID Network Intrusion Detection System", Il sistema è stato ritirato del 6/11/2004. L'accesso alla documentazione è ristetto al personale del Departamento della Difusa (DoD – USA)

[14] Kumar S., Spafford E. H., "An Application of Pattern Matching in Intrusion Detection", Technical Report, Giugno 1994

[15] Boyer R.S, Moore J. S,: "A Fast String Search Algorithm" *Communications of ACM*, 20(10), pp 762-772, Ottobre 1977

[16] Aho A Corasic M.: "Fast Pattern Matching:an Aid to Bibliographic Search", *Communications of ACM*, 18(6), pp 333-340, Giugno 1975

[17] Ilgun K, Kemmerer R.A, and Porras, P. A: "State Transition Analysis: A Rule-Based Intrusion Detection Apporach", *IEEE Transaction on Software Engineering* 21, Marzo 1995

[18] Vigna, G., Eckmann S.T., and Kemmerer R.A., "The STAT Tool Suite", in Proceedings of DISCEX 2000, Hilton Head, South Carolina, Gennaio 2000 IEEE Press

[19] Forrest S., Hofmeyr S.A, Somayali A, Loongstaff T.A. "A Sense od Sell for Unix Processes", Proceedings of the 1996 IEEE Symposium on Security and Privacy, 1996

[20] Forrest S., Perelson A.S, Allen L,. Chrukur R., "Self-Nonself Discrimination in a Computer", IEEE Symposium on Security and Privacy, 1994

[21] Forrest S., Perelson A.S, Allen L, Somayali A., "Computer Inmunology", *Communications of the ACM*, 40 10), 1994

[22] Ptacek t., Newsham, T., "Insertion, Evasion and Denial of Service: Eluding Network Intrusion Detection". Secure Network Inc. Whitepaper, 1998

[23] RFC 3954: "Cisco Systems NetFlow Services Export Version 9" y RFC 3955 "Evaluation of Candidate Protocols for IP Flow Information Export (IPFIX)"

[24] Mogul J.C., Ramaknshhann K. K.: "Elininating Receive Live Lock in an Interrupt Driven Kernel", in *ACM Transcations on Computer Systems*, 15(3), Agosto 1997, pp 217-252

[25] V. Hauser: THC-LeapCracker (v.0.1) released by The Hacker's Choice on 2/10.2004. www.thc.org/

[26] Wright J.: "The AS- LEAP ", http: // asleap.sourceforge.net/

[27] He C., Mitchell J.C.: "Analysis of the 802.11/4-Way Handshake", in Proceedings of the 3rd ACM International Workshop on Wireless Security (WISe'04)Philadelphia, PA, Oct.2004

[28] Jones E., Le Moigne O.: "OSPF Security Vulnerabilities Analysis", Internet Draft, 1 Diciembre 2004

[29] Zhao X., Pei D., Wang L., Massey D., Mankin A., Wu F.S. Zhang L., : "An Analysis of BGP Multiple Origin AS (MOAS) Conflicts", Proc. ACM SIGCOMM Internet Measurement Workshop, 2001.

Nota: se ha decidido no traducir la bibliografía y las frases que aparecen, tanto en italiano como en inglés, dentro de las figuras; y se ha respetado la forma en que los autores han empleado las referencias bibliográficas debido, en ambos casos, a su particularidad.