Parametrización de anomalías en NIDS híbridos mediante etiquetado selectivo de contenidos

L. Sánchez, P. García, *Member IEEE,* J. Díaz, *Member IEEE,* G. Maciá
Dpto. De Teoría de la Señal, Telemática y Comunicaciones,
E.T.S de Ingeniería Informática y de Telecomunicación
sancale@correo.ugr.es; {pgteodor,jedv,gmacia}@ugr.es

***Resumen*— En este artículo se presenta un procedimiento para la generación automática de firmas en sistemas NIDS híbridos. Con objeto de llevar a cabo una realimentación en bucle cerrado desde el módulo A-NIDS, basado en anomalías, al S-NIDS, basado en firmas, el tráfico clasificado como anómalo será analizado siguiendo un proceso estocástico. A resultas, se seleccionarán aquellas partes específicamente anómalas del tráfico, de las cuales se derivará una firma a incluir en la base de datos de patrones del S-NIDS. Antes de proceder a su inclusión efectiva, y con objeto de optimizar el espacio de firmas considerado, cada nueva firma generada será comparada, agrupada y suavizada, en su caso, con otras "similares" ya existentes. Aunque de carácter preliminar, la experimentación llevada a cabo hasta el momento evidencia un comportamiento prometedor del sistema global propuesto por los autores.**

***Palabras clave*— Agrupamiento *(clustering)*, anomalía *(anomaly)*, detección de intrusiones *(intrusion detection)*, firma *(signature)*, intrusión *(intrusion)*, modelo de normalidad *(normality model)*, respuesta a intrusiones *(intrusion response)*, servicios de seguridad *(security services)*.**

I. Nomenclatura

A lo largo del documento se hará uso de los siguientes acrónimos principales:

IDS *(Intrusion Detection System)*; IRS *(Intrusion Response System)*; HTTP *(HyperText Transfer Protocol)*; URI *(Uniform Resource Identifier)*; FSA *(Finite State Automaton)*; LCSeq *(Longest Common Subsequence)*

II. Introducción

El constante aumento en la complejidad de las redes y sistemas de comunicación implica la aparición de un sinfín de nuevas vulnerabilidades y problemas en estos entornos. Así se recoge en los numerosos estudios que, sobre incidentes de seguridad, vienen realizando año tras año entidades especializadas tales como CERT (http://www.cert.org), FIRST (http://www.first.org) y SANS (http://www.sans.org), cuya labor de monitorizar e informar acerca de los principales riesgos y vulnerabilidades en los entornos TIC resulta imprescindible en la actualidad.

En el ámbito de la provisión de servicios de seguridad se hace necesario el desarrollo de herramientas capaces de garantizar la confianza de los usuarios. La importancia de dicho estudio y desarrollo se evidencia en la enorme actividad, tanto de carácter privado como público, existente actualmente a este respecto.

Son numerosas, así, las herramientas ideadas para dar respuesta a uno o varios de los aspectos involucrados en la seguridad en las TIC: confidencialidad, autenticación, disponibilidad, privacidad, etc. Entre otras varias posibles (cortafuegos, antivirus, anti-*spyware*, …) merecen ser destacados los *sistemas de detección de intrusiones*, o IDS [1]. Éstos fueron ideados para determinar la potencial ocurrencia de eventos susceptibles de causar un riesgo para las fuentes de información o recursos del sistema a proteger (intentos de acceso o manipulación, entre otros). En otras palabras, para posibilitar la detección de acciones no permitidas por las políticas de seguridad consideradas en el entorno a proteger [2].

En esta línea, el presente trabajo aborda uno de los principales retos actuales al respecto del diseño e implementación de sistemas IDS: mecanismos de *respuesta automática a intrusiones*, o IRS [3], [8]. La gran mayoría de los esquemas IRS actualmente disponibles se limitan a la mera generación de una notificación (por ejemplo, a través de un mensaje de correo electrónico) ante la aparición de un evento intrusivo. Dicha circunstancia deberá ser posteriormente estudiada y tratada de forma manual por un administrador humano, lo cual resulta de todo punto inadecuado por cuanto que los tiempos involucrados (en especial en los entornos de comunicaciones actuales, caracterizados por su alta velocidad y gran volumen de tráfico) invalidan por lo general el potencial sistema de protección desplegado.

Frente a la adopción del anterior u otros posibles mecanismos IRS, como es la actuación sobre las reglas de los cortafuegos, en el trabajo aquí planteado se propone la implementación de un esquema de generación automática de firmas que realimente y complemente a un IDS dado; en nuestro caso, uno comercial de amplio uso: Snort. Si bien es de señalar la existencia de numerosas referencias en este sentido en la literatura especializada [11], la particularidad de la presente contribución reside en varios hechos diferenciados. Por un lado, porque la generación de la firma correspondiente se realiza partiendo de la disposición de *alarmas de anomalía*s en el contexto de esquemas NIDS

basados en el estudio del *comportamiento normal* (o *anormal*) de un sistema dado. Por otra parte, las firmas se derivan siguiendo un proceso de análisis estocástico sobre los contenidos del tráfico de red monitorizado. También hay que reseñar que las nuevas firmas obtenidas, antes de ser incorporadas a la base de datos de patrones del NIDS, podrán ser "agrupadas" con objeto de minimizar el número de patrones específicos total a considerar.

La consideración del protocolo HTTP como punto de inicio de la aproximación propuesta se fundamenta en el alto porcentaje actual de tráfico HTTP frente a otros servicios. No obstante este hecho, el procedimiento general aquí planteado resulta extrapolable a otros servicios y protocolos tras las oportunas particularizaciones.

Por lo demás, la organización del artículo es como sigue. En la parte III se presentarán los principios básicos sobre NIDS necesarios para situar adecuadamente en contexto el desarrollo realizado. Abordando ya el trabajo objeto de esta contribución, la parte IV discute la metodología específica propuesta por los autores de cara a la generación de firmas y su inclusión en el IDS, como mecanismo de respuesta a intrusiones en redes. En la parte V se presentarán y discutirán los principales resultados experimentales obtenidos al respecto, discutiéndose finalmente en la parte VI las principales conclusiones y algunas de las principales líneas de trabajo futuro.

III. Fundamentos IDS y Sistemas *H*-NIDS

Todo sistema IDS opera bajo el mismo procedimiento básico: extracción de información (*monitorización*) y análisis de la misma en busca de eventos intrusivos (*detección*) –*véase* RFC 4765, 4767 y el grupo IDWG en IETF, www.iertf.org–. De acuerdo con ello, dos son los principales criterios de clasificación aceptados: origen/procedencia de la información a considerar, y tipo de análisis realizado sobre los datos en el proceso de detección. Según el origen de la información, existen los IDS basados en *host*, o HIDS ("Host-based IDS"), en los que los datos manejados se refieren a máquinas y dispositivos varios (llamadas al sistema, identificadores de proceso, perfiles de usuario, etc.), y los IDS basados en red, o NIDS ("Network-based IDS"), en cuyo caso la información monitorizada es referida a eventos de tráfico relacionados con los protocolos de transmisión (cabeceras y direcciones IP, puertos origen y destino, y otros parámetros y variables relacionados).

Frente al estudio del origen de la información, el proceso de análisis da lugar a IDS basados en firmas, o S-IDS ("Signature-based IDS"), o a IDS basados en anomalías, o A-IDS ("Anomaly-based IDS"). El objetivo de los primeros es la detección de procesos de intrusión ya identificados y parametrizados, buscando en la información a analizar ciertos patrones ya definidos (firmas) para los ataques. Para ello, debe establecerse y actualizarse de forma periódica una base de datos de las firmas o patrones de ataques conocidos. Por su parte, los A-IDS llevan a cabo la estimación de la desviación de comportamiento entre la información monitorizada y el valor esperado, "normal" o "anormal", para la misma. Dicho comportamiento "normal" o "anormal", como ocurre con la base de datos de firmas en S-IDS, debe encontrarse especificado con anterioridad al proceso de detección, generándose una alarma cuando el grado de desviación obtenido supere un cierto umbral.

A. Sistemas NIDS híbridos

La adopción de una tipología concreta de IDS (NIDS *vs.* HIDS; A-IDS *vs.* S-IDS) pasa por la consideración de las principales diferencias entre los esquemas correspondientes, siendo los dos principales criterios a considerar la tasa o eficacia de detección y el coste involucrado en el análisis. Por lo que respecta a la elección NIDS-HIDS, a lo largo del presente trabajo se concreta el uso del primer tipo de IDS frente al segundo, si bien dicha elección no debe imputarse a otras cuestiones más allá de las puramente relacionadas con el tema de trabajo interés de de los autores: entornos de red frente a sistemas máquina finales. Por otro lado, algunas de las discusiones aquí realizadas para NIDS pueden ser también aplicadas a HIDS y, en todo caso, completadas por este último tipo de esquemas de detección de intrusiones.

Por su parte, los sistemas S-IDS presentan, frente a los A-IDS, como características principales su sencillez y alta eficacia. Ambas cuestiones son consecuencia directa de su operativa, una mera comparación de cadenas. Sin embargo, su rigidez funcional, lo que provoca una poca (o nula) capacidad de detección de ataques desconocidos (aun en el caso de que éstos sean mínimas variaciones de otros existentes), hace altamente atractiva la teórica funcionalidad que en este sentido presentan los sistemas A-IDS. Como principal contrapartida, sin embargo, los esquemas basados en anomalías suelen dar lugar a una alta tasa de falsas alarmas, o falsos positivos (eventos "normales" detectados como intrusivos por el sistema).

A partir de las consideraciones previas surgieron los IDS denominados *híbridos*, o *h*-IDS ("Hybrid IDS") [9], en donde se combinan S-IDS y A-IDS en un todo. Siguiendo esta línea de actuación, son diversas las propuestas y contribuciones que al respecto pueden encontrarse en la literatura especializada, concretamente por parte de los autores [4], [16]. Como principales aspectos de éstas últimas, se pueden indicar los siguientes:

– Se realiza una detección en dos pasos. En uno primero se lleva a cabo un análisis basado en firmas sobre el tráfico de red capturado (S-NIDS), determinando la existencia de ataques conocidos si se produce coincidencia con alguno de los patrones/firmas contenidos en la base de datos al efecto. En una segunda etapa, aquel tráfico no detectado como malicioso se someterá a un análisis

TABLA I
SISTEMAS NIDS CON CAPACIDADES DE DETECCIÓN HÍBRIDAS, SEGÚN EL DESARROLLADOR

Sistema	Desarrollador
AirDefense Guard	AirDefense, Inc.
Anagram	Intrusion Detection System Lab, Columbia University
Autonomous Agents for Intrusion Detection (AAFID)	CERIAS/Purdue University
Bro	Lawrence Berkeley National Laboratory
Checkpoint IPS-1	NFR Security
FireProof	Radware Ltd.
Firestorm NIDS	Gianni Tedesco
Mazu Profiler	Mazu Networks, Inc.
Minnesota INtrusion Detection System (MINDS)	Univ. of Minnesota
Network at Guard (N@G)	C-DAC (Formerly National Centre for Software Technology)
Nitro Security IPS	Nitro Security
nPatrol	nSecure
Prelude IDS	Yoann Vandoorselaere et al.
SecureNet IDS/IPS	Intrusion Inc.
Snort IDS	Marty Roesch
Strata Guard IDS/IPS	StillSecure
Symantec Intrusion Protection	Symantec
TippingPoint Intrusion Prevention System	3COM/TippingPoint Technologies

basado en anomalías (módulo A-NIDS), determinándose si se trata de tráfico "normal" (y por tanto "limpio") o "anormal".

- El procesado A-NIDS se implementa en base a la consideración combinada de técnicas estocásticas (cadenas/modelos de Markov) y esquemas basados en especificación.
- Aunque extrapolable a otros servicios, el trabajo actual se centra en el desarrollo de NIDS orientados al servicio HTTP, en base al análisis de URI del método GET.

Las tres principales ventajas obtenidas con el esquema *h*-NIDS planteado son así: alta velocidad de computación y fiabilidad en el proceso de detección, capacidad teórica de detección de eventos intrusivos desconocidos y disminución en la tasa de falsas alarmas, como consecuencia de la complementariedad entre los módulos S-NIDS y A-NIDS.

A modo de conclusión, la Tabla 1 indica algunos de los sistemas/plataformas IDS disponibles en la actualidad.

IV. NIDS EN BUCLE CERRADO

Una cuestión de gran relevancia en el contexto de los sistemas IDS se refiere a las posibles respuestas a adoptar ante la potencial detección de una intrusión. Como se ha apuntado con anterioridad, la gran parte de las soluciones propuestas en la literatura especializada se refieren a meras notificaciones al administrador de la red, el cual llevará a cabo su posterior tratamiento manual. Este hecho constituye por sí mismo un importante *hándicap* en la adopción de acciones de respuesta ante intrusiones en tiempo real eficaces de cara a la solución efectiva de los eventos intrusivos.

Este hecho se agrava enormemente si las alarmas de detección consideradas se refieren a eventos anómalos. Puesto que, por definición, una anomalía no es un ataque, ¿cómo actuar ante la ocurrencia de este tipo de situaciones? En este trabajo, sustentado sobre otros propios como [4], [16], se plantea la metodología operacional mostrada en la Fig. 1, como se describe a continuación.

Monitorizado el entorno a proteger, el tráfico capturado será procesado primeramente por el módulo S-NIDS del *h*-NIDS. Los ataques detectados serán convenientemente reportados y tratados de acuerdos con las políticas de actuación/respuesta correspondientes. En cambio, el tráfico "limpio" será analizado en una segunda etapa por el sub-sistema A-NIDS, generándose por parte de éste una

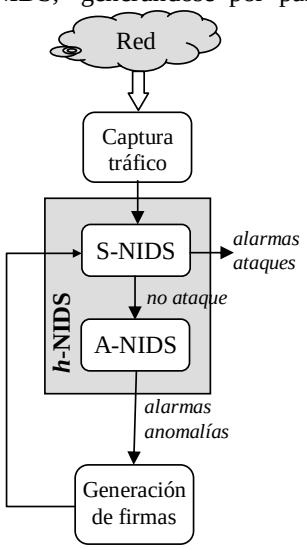

Fig. 1. Esquema h-NIDS realimentado, en el que la actuación sobre las alarmas del módulo A-NIDS se traduce en la derivación e incorporación de nuevas firmas al S-NIDS.

alarma en caso de determinarse la ocurrencia de un evento anómalo. Dicha situación disparará la puesta en marcha de un mecanismo de análisis del tráfico correspondiente, con un doble objetivo. En primer lugar, la generación automática de firmas/reglas que representen dicho tráfico. Ello permitirá un cierto grado de autonomía en cuanto a la actualización automática de la base de datos de patrones del S-NIDS. Por otra parte, sería de interés llevar a cabo el marcado de la firma en cuestión en base al grado de anomalía determinado para el tráfico correspondiente. A través de ello se indicaría la "fiabilidad" de la categorización del nuevo "ataque", lo que repercute en la naturaleza y severidad de las futuras posibles acciones de respuesta susceptibles de ser adoptadas para el mismo.

Centrando nuestra atención en este punto en la generación automática de firmas, en lo que sigue se indican las principales propuestas que en este sentido existen en la bibliografía. Vistas éstas, seguidamente se procederá a la discusión del esquema particular propuesto por los autores.

A. Generación automática de firmas

Son diversos los sistemas disponibles en los que se hace uso de mecanismos de generación automática de firmas. Una clasificación básica de éstos es la que sigue a continuación.

1) Técnicas basadas en comparación de patrones

Los esquemas de *pattern-matching* permiten la creación de firmas precisas sin necesidad de una inspección manual del tráfico. Una vez que se tienen los flujos de tráfico a analizar, se aplica algún tipo de algoritmo de comparación de cadenas o patrones con el fin de obtener similitudes en el *payload* de las PDU. Los patrones seleccionados se utilizarán para dar lugar a las firmas generadas de forma automática.

Entre otros sistemas basados en este tipo de esquemas se encuentra *Honeycomb* [12], en el cual se utiliza un algoritmo LCS *("Longest Common Substring")* basado en *suffix-trees*, para detectar la mayor subcadena común. Fundamentado en el uso de *Honeycomb*, *SweetBait* presenta la peculiaridad de la disposición de listas blancas (listas de firmas no permitidas), previniendo así la generación de firmas que produzcan falsos positivos [10]. Otro sistema es *PAYL* [13], el cual obtiene las firmas calculando la coincidencia de subcadenas (LCSeq, "*Largest Common Subsequence*"), con la característica propia de que realiza un estudio de la correlación de múltiples alertas para reducir las decisiones incorrectas.

2) Técnicas basadas en la frecuencia de las observaciones

Este tipo de esquemas permite la generación y el despliegue automático de firmas en base al análisis del contenido del tráfico de red. Los *payloads* son, así, examinados en busca de cadenas de bytes con una frecuencia de repetición elevada. La prevalencia del contenido se utiliza para identificar las potenciales secuencias comunes capaces de aprovechar alguna vulnerabilidad del entorno a proteger. Dichas cadenas son propuestas como candidatas para su derivación en firmas.

Varios son los sistemas que utilizan cálculos de frecuencia para realizar la generación de firmas. Entre ellos destacan: *AutoGraph* [18], que realiza una división en bloques de longitud variable utilizando la herramienta COPP ("*COntent-based Payload Partitioning*") y selecciona aquel bloque con mayor valor para la firma; también incluye la posibilidad de utilizar listas blancas. Análogamente, en [14] se introduce *EarlyBird*, sistema en el cual se propone una solución denominada "*content-shifting"*, que además de medir la prevalencia de los paquetes calcula un umbral de dispersión a partir de las direcciones origen y destino, de forma que se eviten falsos positivos.

3) Técnicas basadas en contenidos disjuntos

En este caso se generan firmas idóneas para la detección de ataques de tipo polimórfico. Las firmas generadas consisten en múltiples subcadenas disjuntas, y no en una única cadena continua de bytes, dando como resultado menores tasas de falsos positivos. Bajo esta perspectiva se pueden establecer tres nuevas clases de firmas:

- Conjuntos de cadenas de bytes: secuencias continuas de bytes que "coinciden" con un *payload* dado si todas ellas se encuentran en él, independientemente de su orden.
- Subsecuencias de cadenas de bytes: conjunto ordenado de secuencias que coinciden si y sólo si la secuencia aparece en un orden específico. Para ello suelen aplicarse algoritmos de alineamiento de cadenas.
- Conjuntos estadísticos: cadenas a las que se les asocia una puntuación estocástica y un umbral que determina la máxima tasa de decisiones incorrectas permitida. Existen dos tipos básicos: conjuntos con una estimación Bayesiana, y conjuntos basados en distribuciones de frecuencia.

Los sistemas *PolyGraph* [19] y *PADS* ("Position-Aware Distribution Signature") [20] generan firmas de alguna de las tres clases citadas. En el primero de ellos se realiza un preprocesamiento para extraer las distintas subcadenas de una longitud mínima, llevándose a cabo además un *clustering* jerárquico para generalizar las firmas en reglas. Por su parte, PADS genera sólo firmas de la tercera clase, basándose en distribuciones de frecuencia sensibles al contexto y creando firmas flexibles y precisas.

4) Técnicas basadas en análisis semántico

En estas técnicas se realiza un análisis semántico automático para identificar distintas partes del *payload* útiles para la generación de una firma, comprobándose si los datos se utilizan de forma peligrosa. La información acerca de la vulnerabilidad y de cómo ésta es explotada se utiliza para generar una firma precisa.

Un sistema que opera bajo esta técnica es *TaintCheck* [15],

el cual permite adicionalmente verificar la calidad de las firmas previamente generadas.

B. Generación de firmas en h-NIDS mediante análisis estocástico y etiquetado selectivo de contenidos

Aunque se apunta como de interés por parte de algunos investigadores, la gran mayoría de los esquemas de generación automática de firmas desarrollados en la literatura obvia el análisis de los contenidos de los paquetes de tráfico.

Frente a ello, como se ha descrito con anterioridad, se consideran como atributos principales para la parametrización pretendida las direcciones IP origen y/o destino, los puertos involucrados en las comunicaciones o los *flags* de TCP, entre otros. Aunque varios esquemas de los mostrados sí realizan un análisis del contenido de los paquetes, tratan dichos contenidos como meras cadenas de bytes, reduciéndose el estudio realizado sobre el *payload* a meros cálculos de similitud o frecuencias de aparición de las secuencias de bytes más comunes.

Por su parte, la metodología de análisis de intrusiones desarrollada por los autores, vista en la sección III-A, se sustenta en la detección estocástica de anomalías en URI de peticiones GET para el servicio HTTP [5] [16]. Es por ello que, en coherencia con el proceso A-NIDS llevado a cabo en el contexto de la operativa de la Fig. 1, la generación de firmas a asociar a los eventos anómalos observados también se abordará desde la perspectiva de un análisis probabilístico de las cadenas de caracteres que conforman los URI. Es de destacar lo novedoso del mecanismo de generación aquí presentado, el cual constituye una alternativa a los esquemas previamente descritos. El proceso general sobre un URI dado, URI^k, clasificado como anómalo, es el mostrado en la Fig. 2, cuyas etapas son tres:

1. Extracción selectiva de cadenas: Cada URI anómalo, URI^k, será analizado en mayor detalle de lo hecho por el módulo A-NIDS, de manera que se identificarán y extraerán aquellas subcadenas que contribuyan en mayor medida al carácter anómalo de la URI.
2. Derivación de firmas: En esta etapa, las subcadenas anómalas serán procesadas de cara a la representación, y consecuente generación de firma asociada, de URI^k.
3. Comparación y *clustering* de cadenas: Con objeto de evitar la generación de una firma particular para cada URI anómalo, las firmas derivadas serán categorizadas (agrupadas) en *clusters* y una única firma usada para todos.

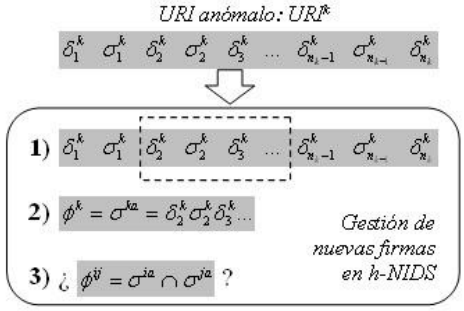

Fig. 2. Procedimiento en 3 pasos para la generación automática de firmas en *h*-NIDS.

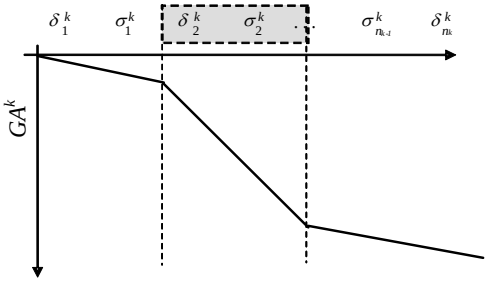

Fig. 3. Etiquetado selectivo de anomalías.

Seguidamente se describen en mayor detalle cada una de las fases citadas. Antes de ello, sin embargo, se recomienda ver el Apéndice A más adelante, donde se hace un breve, pero necesario, repaso de los fundamentos generales relativos al análisis estocástico llevado a cabo sobre las URI para la clasificación de anomalías [7].

1) Extracción selectiva de cadenas

Etiquetado como anómalo un URI observado, URI^k, la primera fase del proceso de generación de firmas aquí propuesto pasa por realizar un análisis en profundidad del mismo, a fin de identificar las subcadenas componentes del URI y motivo principal de la alarma. En la Fig. 3 se muestra un ejemplo conceptual del proceso.

Habida cuenta de la naturaleza logarítmica del grado de anomalía asociado a URI^k, GA^k (*véase* Apéndice A), su evolución será decreciente. El estudio de la pendiente de GA^k permitirá identificar aquellas subcadenas (y los estados asociados, en base a los delimitadores observados) que contribuyen en mayor medida al *anomaly score* total acumulado (y normalizado según se indica en el Apéndice A); por ejemplo, en base a un porcentaje sobre éste. Dichas subcadenas (en línea discontinua en la Fig. 3) serán marcadas como anómalas de cara a la derivación de la firma o patrón representativo de URI^k.

2) Derivación de firmas

Aceptada la notación $\sigma^{ka} = \delta_1^{ka}\sigma_1^{ka}\delta_2^{ka}\sigma_2^{ka}...\sigma_{n_k-1}^{ka}\delta_{n_k}^{ka}$ para identificar la secuencia, o secuencias, de símbolos y delimitadores anómalos dentro del URI^k analizado, será ésta directamente el patrón para representar la anomalía detectada: $\phi^k = \sigma^{ka}$.

Así, el proceso de detección llevado a cabo por el módulo S-NIDS, una vez incluida la nueva firma ϕ_k en la base de datos correspondiente, será tan simple como localizar la cadena en cuestión, σ^{ka}, en las URI GET recibidas en el servidor HTTP a proteger.

3) Comparación y clustering *de secuencias*

Con objeto de optimizar el espacio de firmas generadas, antes de proceder a la inclusión directa de ellas en la base de datos del S-NIDS se realizará una búsqueda en la misma a fin

de detectar potenciales "similitudes" con otros patrones ya existentes. Ello nos permitirá agrupar firmas, optimizando así el espacio de búsqueda y relajando el propio proceso S-NIDS llevado a cabo, tanto computacionalmente como desde la perspectiva de la eficacia de detección conseguida.

Para la comparación de las cadenas se utilizará la técnica bien conocida de alineamiento de secuencias LCSeq [6], a través de la cual se definirá la similitud entre dos firmas (cadenas) $\phi^i = \sigma^{ia}$ y $\phi^j = \sigma^{ja}$, como el número de subcadenas iguales que aparecen en ambas:

$$sim(\phi^i, \phi^j) = cardinal\left[\sigma_1^{ij} ... \sigma_m^{ij} \quad / \quad \sigma_k^{ij} \subset \phi^i, \phi^j, k = 1,..., m\right]$$

Sobre el valor de este parámetro se concluirá que las firmas son "iguales" cuando el número de coincidencias, *m*, supere un cierto umbral, declarándose como regla común representativa del *cluster* la secuencia de cadenas comunes correspondiente:

$$\phi^{ij} = \sigma^{ia} \cap \sigma^{ja} = \sigma_1^{ij} ... \sigma_m^{ij}$$

V. Resultados Experimentales

Como evaluación primera del trabajo desarrollado se ha realizado una experimentación preliminar, a partir de la cual, según se evidencia de lo que sigue, se ha obtenido un conjunto de resultados altamente prometedores.

La base de datos de tráfico considerada en la experimentación consta de un total de 140.000 paquetes, todos ellos correspondientes a solicitudes GET HTTP recibidas en un servidor web Apache 2.2.8 en explotación, perteneciente al grupo de trabajo y con acceso externo. Hay que reseñar, por tanto, que el tráfico analizado es real, y en modo alguno generado artificiosamente para la ocasión.

En esta misma línea, y no menos importante, hay que indicar que toda la experimentación efectuada parte de los desarrollos disponibles en la actualidad por el grupo de trabajo de los autores. Ello se refiere tanto al modelo de normalidad considerado en el procesamiento y detección A-NIDS, como al proceso de detección en sí mismo

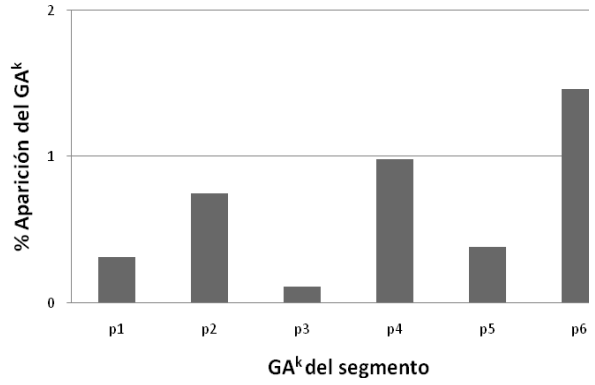

Fig. 4. Histograma de los distintos valores de GA^k para tráfico "limpio".

basado en la especificación de un valor umbral para la clasificación del tráfico en normal/anómalo. No obstante este hecho, en esta primera fase de evaluación se ha considerado como tráfico anómalo el total del tráfico disponible. Sobre éste se indican los resultados que siguen.

Del análisis de las URI correspondientes al tráfico monitorizado se deriva el histograma mostrado en la Fig. 4. Éste se refiere a las frecuencias de aparición correspondientes a 6 de los valores del GA^k (normalizado por el número de segmentos de que consta la URI^k asociada) más observados (*p1* a *p6*). Por razones de escala en la representación, otros 2 valores adicionales, uno con una frecuencia de observación en torno al 96% del total y uno inferior al 0,01%, no han sido incluidos en la figura.

Se ha realizado una segunda experimentación, análoga a la indicada previamente, pero utilizando en esta ocasión una base de datos de ataques, que consta de un total de 1.000 instancias GET HTTP correspondientes a 320 *exploits* diferentes recopilados de, entre otros, SecurityFocus (http://www.securityfocus.com).

El histograma en este caso obtenido se muestra en la Fig. 5, refiriéndose a las frecuencias de aparición de los distintos valores del GA^k observados (*p1* a *p12*). Los valores del *p1* al *p5* corresponden a segmentos etiquetados como anómalos, observándose que representan aproximadamente un 85% del total; resultado que cabía esperar, dado que el 100% de los paquetes corresponden a ataques, y en consecuencia, la gran mayoría de los segmentos evaluados serán considerados como intrusiones.

Como primer resultado se concluye una variabilidad no especialmente elevada de GA^k a nivel de segmento. Ello indica que a pesar de tratarse de una base de datos compuesta en su totalidad por ataques, el proceso de detección propuesto no experimenta una excesiva complejidad. Una vez extraídas las secuencias anómalas tal como se describe en la parte IV para la fase 2, se procede a la generación de las firmas correspondientes. Como ejemplo de ello a continuación se muestran algunas de las más representativas, junto con su correspondientes URI de partida:

/eManager/Email%20Management/cgi-bin/register.dll
y
/eManager/Email%20Management/cgi-bin/SpamExcp.dll

de firmas finales asociadas:

eManager/Email%20Management,
register.dll
y
SpamExcp.dll

Veamos también un ejemplo de los resultados obtenidos en relación a la etapa final de comparación y *clustering* de firmas para su agrupación en reglas.

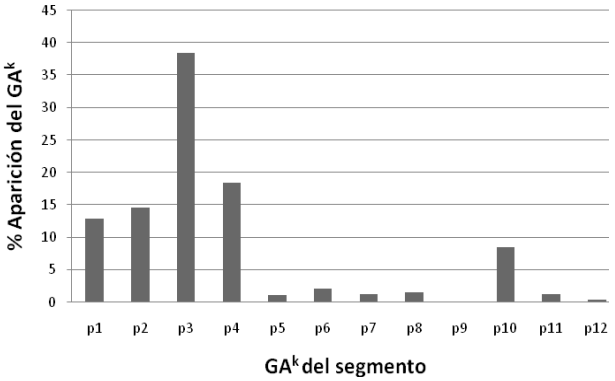

Fig. 5. Histograma de los distintos valores de GA^k para tráfico de ataque.

Por un lado, una de las firmas generadas:

officescan/cgi/jdkRqNotify.exe?domain=%3Cdomain

es altamente recurrente en la experimentación realizada, generándose, sin embargo, una única nueva entrada en este sentido en la base de datos de firmas global. Así mismo, en el caso de la obtención de las cuatro firmas siguientes:

directory/showphoto.php?photo=OR
directory/showphoto.php?photo=%7C%7C6
directory/showphoto.php?photo=;
directory/showphoto.php?photo='

éstas se agregan "conjuntamente" a través de la regla común

directory/showphoto.php?photo=

VI. Conclusiones

En este artículo se propone una nueva metodología de cara a la generación automática de firmas para tráfico HTTP anómalo en sistemas *h*-NIDS. Ésta se sustenta en dos aspectos principales: análisis estocástico del contenido de los URI, y extracción selectiva de subcadenas anómalas. Adicionalmente, se introduce un procedimiento de comparación de firmas para posibilitar su agrupamiento y, así, optimizar el potencial análisis S-NIDS subsiguiente.

Aunque con resultados prometedores, los experimentos realizados hasta la presente para evaluar adecuadamente las prestaciones del esquema propuesto resultan escasos. Para una validación más rigurosa, se está planificando por parte de los autores un conjunto de tests más amplio y representativo.

Adicionalmente a este hecho, otras cuestiones técnicas a tratar en mayor profundidad se refieren a aspectos tales como: normalización del grado de anomalía en base a otros parámetros distintos a la longitud del URI, criterios alternativos para la extracción de las cadenas anómalas, secuenciación o no de éstas, etc.

VII. Agradecimientos

Este trabajo ha sido desarrollado dentro del proyecto del Plan Nacional de Investigación del MEC de código TSI2005-08145-C02-02 (70% fondos FEDER). Asimismo, el alumno Leovigildo Sánchez-Casado es becario de iniciación a la investigación por la Universidad de Granada.

Referencias

[1] E.D. Denning, "An Intrusion-Detection Model". IEEE Transactions on Software Engineering. Vol. 13-2; pp. 222-232, 1987.

[2] T. Sobh, "Wired and Wireless Intrusion Detection System: Classifications, Good Characteristics and State-of-the-Art". Computer Standards & Interfaces. Vol. 28; pp. 670-694, 2006.

[3] P. Kabiri, A. Ghorbani, "Research in Intrusion Detection and Response – A Survey". International Journal of Network Security. Vol. 1-2; pp. 84-102, 2005.

[4] P. García-Teodoro, J.E. Díaz-Verdejo, G. Maciá-Fernández, L. Sánchez-Casado, "Network-based Hybrid Detection and Honeysystems as Active Reaction Scheme". IJCSNS, Vol. 7:10, pp. 62-70, October, 2007.

[5] J.M. Estévez-Tapiador, P. García-Teodoro, J.E. Díaz-Verdejo, "Measuring Normality in HTTP Traffic for Anomaly-based Intrusion Detection". Computer Networks, Vol. 5:2, pp. 175-193, 2004.

[6] David Maier. "The Complexity of Some Problems on Subsequences and Supersequences". *J. ACM* **25**: 322–336. ACM Press, 1978.

[7] J.E. Díaz-Verdejo, P. García-Teodoro, P. Muñoz, G. Maciá-Fernández, F. Toro-Negro, "Una Aproximación Basada en Snort para el Desarrollo e Implementación de IDS Híbridos". *IEEE América Latina*, Vol. 5, No. 6; pp. 386-392, October, 2007.

[8] M. Rash, A. Orebaugh, G. Clark, B. Pinkard, J. Babbin, *Intrusion Prevention and Active Response*. Syngress Publishing, Inc. (2005).

[9] PMG, "Maximizing the Value of Network Intrusion Detection". A corporate white paper from the product management group of intrusion.com, 2001.

[10] Georgios Portokalidis, Herbert Bos, "Sweetbait: Zero-hour Worm Detection and Containment using Honeypots". Technical Report IR-CS-015, Vrije Universiteit, Amsterdam, The Netherlands, May 2005.

[11] U. Zurutuza, "Data Mining Approaches for Analysis of Worm Activity Toward Automatic Signature Generation", Ph.D. dissertation, directed by R. Uribeetxeberria and D. Zamboni, Univ. de Mondragón, Spain, January, 2008.

[12] Christian Kreibich, Jon Crowcroft, "Honeycomb – Creating Intrusion Detection Signatures using Honeypots". 2nd Workshop on Hot Topics in Networks (Hotnets II), Boston, November 2003.

[13] Kei Wang, Gabriela Cretu, Salvatore Stolfo, "Anomalous Payload-based Worm Detection and Signature Generation". 8th International Symposium on Recent Advances in Intrusion Detection (RAID 2005), September 2005.

[14] Sumeet Singh, Cristian Estan, George Varghese, Stefan Savage, "Automated Worm Fingerprinting". 6th Symposium on Operating Systems Design & Implementation (OSDI'04). USENIX Association, 2004.

[15] James Newsome, Dawn Song, "Dynamic Taint Analysis for Automatic Detection, Analysis, and Signature Generation of Exploits on Commodity Software". 12th Annual Network and Distributed System Security Symposium (NDSS 05), Feb. 2005.

[16] M. Bermúdez-Edo, R. Salazar-Hernández, J.E. Díaz-Verdejo, P. García-Teodoro, "Proposals on Assessment Environments for Anomaly-based Network Intrusion Detection Systems". Lecture Notes on Computer Science, Vol. 4347, pp. 210-221, 2006.

[17] J.M. Estévez-Tapiador, P. García-Teodoro, J.E. Díaz-Verdejo, "Detection of Web-based Attacks through Markovian Protocol Parsing". 10th IEEE Symposium on Computers and Communications (ISCC), Vol. 5:2, pp. 457-462, Cartagena (Spain), 2005.

[18] Hyang-Ah Kim, Brad Karp, "Autograph: Toward Automated, Distributed Worm Signature Detection". 13th USENIX Security Symposium, pp. 271-286, San Diego, CA, 2004.

[19] James Newsome, Brad Karp, Dawn Song, "Polygraph: Automatically Generating Signatures for Polymorphic Worms". In Proceedings of the 2005 IEEE Symposium on Security and Privacy (S&P'05), pp. 226-241, Washington, DC, USA, 2005. IEEE Computer Society.

[20] Yong Tang, Shigang Chen, "Defending against Internet Worms: a Signature-based Approach". 24th Conference on Computer Communications (IEEE INFOCOM 2005), pp. 1384-1394, May 2005.

APÉNDICE A. ANÁLISIS DE ANOMALÍAS EN HTTP

A partir de los RFC 1945, 2616, 2396 y 3986 se puede derivar la estructura FSA aceptada para las peticiones HTTP correspondientes al método GET. Dicha estructura puede modelarse haciendo uso de la teoría de Markov, de manera que los estados y sus transiciones queden representados a través de un modelo $M=\{\Sigma,\Lambda,B\}$, siendo Σ el conjunto de estados aceptados, Λ la matriz de probabilidades de transición entre ellos y B la matriz de observaciones correspondiente a las probabilidades de aparición de *símbolos* en cada uno de los estados.

En la Fig. 6 se muestra el FSA considerado por los autores para GET [7]. De éste cabe destacar:

- Σ (S_i): los estados del FSA son S_R, estado de inicio de recurso, S_A, estado de atributo, y S_V, estado de valor, además de los estados inicial y final S_I y S_F.
- Λ (a_{ij}): las transiciones entre cualesquiera dos estados *i* y *j* vienen determinadas por la potencial aparición de los separadores (δ_k): '/', delimitador de recurso, '?', delimitador de parámetro, '=', delimitador de asignación de recurso, '&', delimitador entre parámetros, y ' ', o SP (ASCII 32), delimitador de fin de recurso (*EOR*).
- B (b_i): la matriz de probabilidades de observación refiere a las cadenas de caracteres o símbolos (σ_i) existentes entre cada pareja de delimitadores consecutivos, correspondiente, en suma, a un estado dado.

De acuerdo con todo ello, el grado de anomalía de un URI dado URI^k, denotado como GA^k, una vez éste segmentado en una secuencia de símbolos y delimitadores, $URI^k = \delta_1^k \sigma_1^k \delta_2^k \sigma_2^k \delta_3^k ... \delta_{n_k-1}^k \sigma_{n_{k-1}}^k \delta_{n_k}^k$, se obtendrá en base al *score* dado por la función *logMAP* ("logarithmic Maximum a Posteriori Probability") como:

$$GA^k = \log b_{\sigma_1^k} + \sum_{j=2}^{n_k} \left(\log a_{S_{\delta_{j-1}^k} S_{\delta_j^k}} + \log b_{\sigma_j^k} \right)$$

donde $a_{S_{\delta_{j-i}^k} S_{\delta_j^k}}$ es la probabilidad de transición entre los estados definidos por los delimitadores δ_{j-1}^k y δ_j^k, $S_{\delta_{j-1}^k}$ y $S_{\delta_j^k}$, mientras que $b_{\sigma_j^k}$ representa la probabilidad de observación del símbolo σ_j^k en el estado definido por el delimitador δ_{j-1}^k, $S_{\delta_{j-1}^k}$.

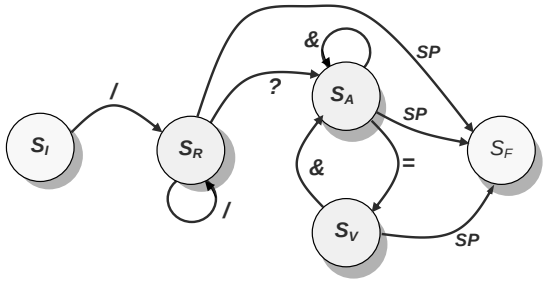

Fig. 6. FSA considerado para el método GET HTTP.

Finalmente, una alarma de anomalía será generada para el URI^k si su GA^k asociado (en valor absoluto, dada su naturaleza logarítmica), supera un cierto umbral especificado. Al respecto, es importante hacer notar la conveniencia de normalizar el valor de GA^k en algún sentido; por ejemplo, a la longitud, en número de segmentos constitutivos, de URI^k (n_k). En otro caso, aquellas URI más largas se verán claramente perjudicadas frente a las de mejor longitud.