Una Arquitectura para la Protección de la Privacidad de las Comunicaciones

Marcelo Bagnulo[*], Alberto García-Martínez[†], Arturo Azcorra[†]
[*] Huawei Labs at U. Carlos III de Madrid, [†] Departamento de Ingeniería Telemática
Universidad Carlos III de Madrid, Avda. de la Universidad 39
28911 – Leganés (Madrid)
E-mail: {marcelo, alberto, azcorra}@it.uc3m.es

***Abstract**. SHIP6 is an architecture that aims to enable private communications by preventing eavesdroppers from using network or transport level parameters to correlate packets that belong to the same communication. SHIP6 extends the ability of SHIM6 to vary securely the locators used for a communication. The SHIM6 Context Establishment exchange is protected through the negotiation of Diffie-Hellman keys. A mechanism based in pseudo-random sequences is specified to vary the IP address, along with other relevant parameters of the communication such as transport ports, etc., for an on-going communication. Additionally, a synchronization mechanism that allows unlimited variation for the parameters considered is presented.*

1 Introducción[1]

La preocupación generada por la posibilidad de que tanto organizaciones estatales como empresas puedan hacer un uso cuando menos discutible de la información cursada por Internet es creciente. Un ejemplo de esta amenaza sería un nodo espía que analice accesos a servidores web para identificar los intereses de un usuario, o que inspeccione intercambios de correos para obtener información personal. La privacidad ha venido ligada tradicionalmente al cifrado de las comunicaciones, mediante el uso de protocolos como IPsec, TLS/SSL o SSH. No obstante, las soluciones basadas en cifrado hacen recaer un alto coste computacional en los extremos participantes, y en algunos mecanismos, resulta difícil gestionar las claves entre equipos que se comunican de forma ocasional. Y si esto no fuera poco, el uso de mecanismos de cifrado no impide por lo general que un espía conozca la cantidad de paquetes o duración de una comunicación.

En este artículo presentamos SHIP6 (SHIm6-based Privacy) una arquitectura para la provisión de privacidad en IPv6 que permite la variación de ciertos parámetros (entre ellos las direcciones IP) para los paquetes que pertenecen a una comunicación dada. De esta forma se dificulta de forma notable la capacidad de un nodo colocado en un punto intermedio de la comunicación para identificar los paquetes pertenecientes a una comunicación determinada, impidiendo la reconstrucción del flujo de contenidos, o la determinación de su duración o del número de bytes intercambiados.

[1] Este trabajo ha sido financiado parcialmente por los proyectos RiNG (IST-2005-035167) e IMPROVISA (TSI2005-07384-C03-02).

SHIP6 ofrece un interesante compromiso en funcionalidad y costes entre la habitual desprotección de las comunicaciones, y el uso de tecnologías basadas en cifrado. Adicionalmente, SHIP6 se puede combinar con el uso de cifrado para ofrecer una protección muy superior frente a nodos fisgones que no tienen información sobre las direcciones que pueden utilizar los extremos, que son incapaces de determinar los paquetes que pertenecen al flujo.

La solución presentada refina una arquitectura anterior [1], que está basada en el protocolo SHIM6 [2]. SHIM6 está siendo desarrollado por el IETF con el objetivo de brindar tolerancia a fallos a comunicaciones establecidas entre nodos que disponen de varias direcciones IPv6, y por tanto, de varios caminos entre sí. La arquitectura de privacidad definida en [1] oculta frente a terceros los parámetros intercambiados por SHIM6, y permite que los paquetes de una comunicación utilicen un número elevado de direcciones diferentes. No obstante, la variación de las direcciones está limitada a un conjunto de posibilidades establecido a priori. Otra restricción es que en [1] sólo se considera la variación de las direcciones IP, siendo necesario proteger todos y cada uno de los parámetros que permiten identificar un flujo. La arquitectura SHIP6 permite un número ilimitado de variaciones y oculta en tránsito los valores de todos los parámetros relevantes en los niveles de red y de transporte, ofreciendo una protección más eficaz que soluciones anteriores.

El resto del artículo se estructura de la siguiente forma: En la sección 2 se analizan las estrategias que podría utilizar un sistema espía para reconstruir un flujo a partir de una serie de paquetes capturados. De esta forma determinaremos los parámetros que deben ser protegidos frente a terceros. A continuación se presentan los fundamentos del protocolo SHIM6, que

constituye la base de la arquitectura de privacidad SHIP6. En la sección 4 desarrollamos la propuesta definida en [1], como principal antecedente en el uso del protocolo SHIM6 como herramienta de privacidad. A continuación se desarrolla la arquitectura de privacidad SHIP6, analizando cómo se protege el establecimiento de la comunicación, cómo se implementa la sincronización entre los nodos que se comunican, y cómo se protegen lo parámetros relevantes frente a posibles espías. La sección 6 se dedica a trabajo relacionado, para finalizar con las conclusiones.

2 Estrategias para la Reconstrucción de Comunicaciones

Supongamos un sistema que captura datos en varios nodos de Internet para obtener información privada de los usuarios (tomado este término en sentido amplio, es decir, refiriéndose tanto personas como aplicaciones). El objetivo del sistema podrá ser la obtención de la secuencia completa de los contenidos transferidos en una o varias de las comunicaciones, o bien simplemente conocer la duración de la comunicación o el número de bytes transferidos. Además, puede ser valioso para el sistema establecer relaciones entre distintas comunicaciones en las que participa un mismo usuario.

En las comunicaciones que se basan en los protocolos de transporte más populares (TCP y UDP), todos los paquetes pertenecientes a una misma comunicación utilizan el mismo par de direcciones IP, y las utilizan en ambos sentidos. Esto es debido a que las direcciones de nivel de red son utilizadas por TCP y UDP como parte de la identificación de los interlocutores. Como consecuencia, una estrategia interesante para el sistema espía es utilizar el par <dirección IP origen, dirección IP destino> de los paquetes capturados como primer criterio de clasificación. Si bien puede haber comunicaciones distintas entre dos equipos dados, no es de esperar que sean muchas, y éstas pueden ser fácilmente separables utilizando identificadores de niveles superiores. Por otro lado, las direcciones IP también aportan información muy interesante para la relación de diferentes comunicaciones entre sí, ya que un caso muy frecuente es que las comunicaciones relacionadas con un mismo usuario o actividad sean generadas en un mismo equipo, y que éste tenga direcciones estables.

La mayor capacidad de direccionamiento presente en IPv6 permite que cada equipo pueda disponer de direcciones públicas asociadas establemente a un equipo. De esta forma es posible prescindir de los NATs, que dificultan el inicio de comunicaciones desde el exterior del dominio privado, y la implantación de protocolos que dependen de que los identificadores de red se preserven extremo a extremo, como ocurre en IPsec, SIP, etc. Como aspecto negativo, la estabilidad de las direcciones puede afectar a la privacidad, al desvelar la relación que distintas comunicaciones pueden tener entre sí. La RFC 3041 [3] permite que los nodos varíen de forma aleatoria el identificador de interfaz (los 64 bits menos significativos de la dirección IPv6). De esta forma, comunicaciones establecidas en instantes distintos podrán utilizar diferentes direcciones IP. Nótese que en [3] no se proponen mecanismos para variar las direcciones en comunicaciones ya establecidas.

El protocolo SHIM6, en desarrollo en el IETF, sí permite la variación de las direcciones IP utilizadas para una comunicación en curso. Con las extensiones apropiadas, que describiremos en secciones posteriores, se pueden solventar ciertas limitaciones de la especificación básica, permitiendo el uso de un número ilimitado de direcciones en los paquetes de una comunicación.

Incluso si las direcciones de los paquetes de una comunicación cambian, el sistema espía puede relacionar los paquetes basándose en otros parámetros a nivel de red, de transporte o de aplicación que ofrezcan suficiente discriminación entre flujos. No obstante, es conveniente puntualizar que sólo el conocimiento de tanto las direcciones IP como de los identificadores de niveles superiores garantizan una precisa distinción de los flujos; si no se dispone del conocimiento de todos esos parámetros, pueden inferirse relaciones que no se correspondan con la realidad.

A continuación analizamos los parámetros de nivel de red y de transporte que pueden utilizarse para discriminar comunicaciones transportadas en IPv6, obviando el análisis de los parámetros de nivel de aplicación por requerir un análisis exhaustivo caso por caso:

Identificador de fragmentación. Si se ha aplicado fragmentación a los paquetes, el paquete IPv6 incorpora una cabecera que contiene un campo *Identificador* de 32 bits. El identificador, generado en los nodos origen, suele seguir una secuencia que se incrementa en uno en cada paquete, por lo que, a falta de otro criterio, paquetes con identificadores próximos podrían considerarse como pertenecientes a un mismo flujo.

SPI y número de secuencia de las cabeceras ESP o AH de IPsec. Si se utilizan las cabeceras de extensión ESP o AH de IPsec, el campo SPI (Security Parameters Index) de 32 bits se utiliza para indicar la Asociación de Seguridad (algoritmos de cifrado, claves utilizadas, etc.) utilizada en la comunicación. El SPI es constante para la vida de una Asociación de Seguridad (SA), aunque se pueden negociar nuevas SA durante una comunicación (por ejemplo, a través del protocolo IKE). Además, cada paquete incluye un número de secuencia de 32 bits con el objetivo de evitar ataques de repetición de paquetes. Este parámetro podría utilizarse para relacionar paquetes

con números de secuencia cercanos. No obstante, que su valor siempre comienza en cero para todas las asociaciones, su utilidad como criterio de discriminación de flujos es baja.

Información específica de protocolos de nivel de red. Protocolos que trabajan a nivel de red y que resultan en modificación de los paquetes de datos transmitidos, como ocurre con SHIM6 o MIPv6, pueden incorporar en dichos paquetes parámetros que faciliten la reconstrucción de un flujo. El estudio de este efecto debe realizarse considerando cada protocolo, quedando fuera del ámbito de este artículo.

Puertos de la capa de transporte. Si el paquete no está cifrado por los servicios de la cabecera ESP de IPsec, un sistema espía puede utilizar el protocolo de transporte y los puertos empleados por los extremos para relacionar paquetes. Tanto TCP como UDP multiplexan y demultiplexan las comunicaciones utilizando puerto origen y puerto destino, cada uno de 16 bits. Para valorar la capacidad de clasificación que ofrecen los puertos es interesante destacar que el uso de servicios estándar lleva a que generalmente uno de los puertos utilizados se encuentre dentro de un reducido conjunto de posibilidades, mientras que el otro puerto se debe escoger en el rango de puertos dinámicos [4], de 49152 a 65535 (16384 valores distintos). El conjunto de pares de puertos distintos es por tanto reducido, si consideramos la inspección de un número muy elevado de flujos, y la capacidad de discriminación alcanzable por este criterio, baja.

Número de secuencia TCP. Los números de secuencia de TCP, de 32 bit, también pueden utilizarse para identificar como relacionados a paquetes con valores suficientemente cercanos. Nótese que en este caso los valores del número de secuencia se incrementan con el número de bytes transferidos en un paquete, y no de uno en uno.

Finalmente, el sistema espía puede utilizar información concreta de la aplicación utilizada (identificadores de nivel de aplicación, conocimiento de la máquina de estados de la aplicación, etc.) para identificar a los paquetes pertenecientes a una comunicación, aunque en este caso el coste de procesamiento y el estado requerido en el sistema espía puede ser elevado.

Como conclusión, podemos afirmar que la variación de las direcciones IP durante la vida de una comunicación representa una medida sumamente efectiva para dificultar la reconstrucción de la comunicación en un sistema espía intermedio. Si se desea obtener una mayor protección, se pueden sincronizar estos cambios con la variación de los siguientes parámetros de nivel de red y transporte: identificador de fragmentación, SPI y número de secuencia de IPsec, puertos de nivel de transporte y número de secuencia TCP. Si estas medidas se aplican, sólo un análisis detallado y costoso de los contenidos, dado que hay que interpretar el contenido de un número elevadísimo de paquetes que podrían venir a continuación de uno dado, puede permitir reconstruir las comunicaciones. Y si se utiliza cifrado con ESP, la variación periódica de los parámetros de nivel de red impide por completo la posibilidad de delimitar el flujo completo de información.

3 Breve Descripción del Protocolo SHIM6

La abundancia de direcciones de IPv6, junto con políticas de encaminamiento orientadas a preservar la estabilidad del sistema de rutas interdominio, harán frecuentes configuraciones en las que un equipo IPv6 disponga de múltiples direcciones de alcance global. En efecto, se espera que las redes pequeñas o de tamaño medio, con múltiples proveedores, reciban delegaciones de rangos de direcciones diferentes, provenientes de cada proveedor a través del cuál se conectan. De esta forma, estas redes ya no necesitan propagar un anuncio específico suyo, haciendo más estable el encaminamiento interdominio. Como consecuencia, para que un nodo pueda ser alcanzable a través de cualquier proveedor, debe configurar tantas direcciones como prefijos haya en el sitio.

Para poder preservar una comunicación después de la ocurrencia de un fallo en alguno de los caminos, es necesario que la comunicación pueda continuar a través de otras direcciones distintas. Este cambio debe hacerse de forma transparente al nivel de transporte, ya que esta capa identifica las comunicaciones utilizando las direcciones IP. Para gestionar estos cambios se está desarrollando el protocolo de la capa de red SHIM6 [2]. SHIM6 establece una correspondencia entre los *identificadores*, las direcciones IP presentadas hacia los niveles superiores, que se mantienen constantes a lo largo de una comunicación, y los *localizadores* incluidos en los paquetes que viajan por la red. Al poder variar los localizadores, se permite que paquetes de una misma comunicación puedan tomar diferentes caminos hacia un destino dado. Para gestionar esta correspondencia, los nodos que se comunican deben haber intercambiado las direcciones que actúan como identificadores y localizadores, dando lugar a un estado llamado *contexto SHIM6*.

La capa SHIM6 se coloca dentro de la capa de red por encima de las funciones de encaminamiento de IP (determinación del interfaz de salida para un paquete, etc.), y por debajo de funciones como fragmentación, o IPsec.

A continuación describimos los mecanismos de seguridad utilizados para proteger la arquitectura SHIM6, que se basan en formatos especiales de direcciones con propiedades criptográficas, y los fundamentos del protocolo SHIM6.

3.1 Seguridad en SHIM6

La capacidad del protocolo SHIM6 para asociar varios localizadores a un identificador abre la puerta a ataques en los que una identidad pueda ser asociada a un localizador no legítimo. Para evitar estos ataques se ha propuesto el uso de HBA [5] (*Hash Based Addresses*, *Direcciones Basadas en Hash*). Las HBA son un nuevo tipo de direcciones globales para IPv6 que incorporan dentro del identificador de interfaz un *hash* de los prefijos disponibles en un nodo con múltiples direcciones. Como resultado, se genera un conjunto de direcciones, una para cada prefijo, ligadas criptográficamente entre sí para impedir que un localizador no legítimo se pueda asociar al conjunto. De modo general, un nodo X que tiene múltiples prefijos (PX_1::/64, PX_2::/64,…, PX_N::/64) genera el identificador de interfaz de cada una de sus direcciones como un *hash* de 64 bits del conjunto de prefijos disponible en el enlace, y un número aleatorio RN (*Random Nonce*) como:

$II_P = hash|_{64}$ (PX_P::/64, PX_1::/64,…, PX_{P-1}::/64, PX_{P+1}::/64, … PX_N::/64, RN)

Las direcciones que forman el conjunto HBA se obtienen de la concatenación de cada prefijo con el identificador de interfaz correspondiente. Nótese que los identificadores de interfaz son diferentes para cada dirección porque el orden de los prefijos que se usa de entrada para el *hash* varía para cada prefijo. Un nodo remoto puede verificar si una dirección alternativa está ligada o no a la dirección HBA que se utilizó inicialmente para establecer la comunicación, mediante la ejecución de un simple hash.

3.2 Protocolo SHIM6

El protocolo SHIM6 [2] crea y gestiona el contexto SHIM6 asociado a las comunicaciones establecidas entre dos nodos. Suponga que uno de los nodos de la comunicación decide crear un contexto SHIM6. A este nodo le llamaremos *iniciador*, y al otro nodo participante en la comunicación *correspondiente*. Para el ejemplo, consideraremos que al menos uno de los nodos puede configurar varias direcciones globales, en este caso el iniciador, y que estas direcciones se han generado como un conjunto de HBA asociado a los múltiples prefijos disponibles. El iniciador solicita la creación de un contexto SHIM6 asociado con los múltiples prefijos disponibles mediante un mensaje denominado *I1*. El nodo correspondiente recibe este mensaje, y genera a su vez un mensaje *R1*, sin crear todavía ningún estado, como medida de protección frente a posibles ataques de denegación de servicio. A continuación el iniciador genera un mensaje *I2* que contiene la siguiente información relevante:

- el par de identificadores para cuyas comunicaciones se va a utilizar el contexto SHIM6
- la Etiqueta de Contexto (ET de aquí en adelante) del iniciador, cuya semántica se explica más adelante
- el conjunto de localizadores disponible en el iniciador
- el contexto necesario para validar los localizadores en el receptor, p. ej. el Random Nonce requerido para validar la HBA

Cuando el nodo correspondiente recibe el mensaje *I2*, verifica que el identificador del iniciador esté incluido entre las direcciones de la HBA reconstruyendo el conjunto de direcciones con la información recibida. Si esta verificación es satisfactoria, crea el contexto SHIM6 y responde con un mensaje *R2*, en el que incluye su propia ET, su conjunto de localizadores y la información para que el iniciador valide los localizadores. A partir de este momento ya es posible que los paquetes de un flujo incorporen distintos localizadores.

Mientras la comunicación utilice como localizadores a los identificadores iniciales, la capa SHIM6 no realiza modificaciones a los paquetes de datos. No obstante, cuando se produce un cambio en los localizadores, es necesario que la capa SHIM6 del receptor identifique que esos paquetes deben ser traducidos, y sepa a qué par de identificadores corresponden. Para ello se incluye la ET, que está asociada de forma unívoca a un par de identificadores concretos, en una Cabecera de Extensión SHIM6 en todos los paquetes con localizadores alternativos. Nótese que cuando un paquete se envía desde el nodo correspondiente con destino al iniciador, por poner un ejemplo, la ET incluida es la generada por el iniciador. De esta forma es fácil asegurar que cada ET contenida en un paquete recibido se corresponde con una única comunicación.

4 Antecedentes sobre Privacidad en SHIM6

En un trabajo anterior [1] se proponen una serie de medidas para evitar que del uso de SHIM6 se deriven amenazas para la privacidad. La primera amenaza identificada es que la captura de los mensajes pertenecientes al establecimiento de contexto SHIM6 permita determinar el conjunto de localizadores asociados a cada interlocutor. Para evitarlo, se modifica el protocolo de establecimiento del contexto SHIM6 para que se genere clave Diffie-Hellman, y así cifrar con esta clave el intercambio de direcciones alternativas para la comunicación. Otra amenaza es que la ET, única para cada sentido de la comunicación SHIM6, se pueda utilizar como pista para determinar que paquetes que siguen distintos caminos pertenecen a la misma comunicación. La solución en este caso es que las ETs sean distintas cuando cambian los pares de localizadores.

Un paso más hacia una mayor protección, propuesto también en [1], consiste en definición de mecanismos que extiendan el rango de variación para una comunicación dada tanto para los localizadores como para las ET. La variación se basa en este caso en el uso de secuencias pseudoaleatorias basadas en semillas que se intercambian de forma secreta durante el establecimiento de contexto. De esta forma, ambos nodos participantes, y sólo esos nodos, conocen la lista ordenada de valores a ser utilizados. A partir del momento en el que uno de los nodos decide avanzar en la secuencia de valores, los paquetes generados por él incorporan los nuevos parámetros. Cuando el interlocutor recibe el primer paquete con los nuevos parámetros, interpreta de forma implícita que él también debe avanzar en la secuencia.

Es importante hacer notar que la generación pseudoaleatoria de parámetros que deben ser únicos en un contexto dado puede dar lugar a *colisiones*. En concreto, una *colisión de ETs* ocurre cuando en un mismo nodo se asocia el mismo ET a dos comunicaciones distintas. De igual modo, ocurre una *colisión de direcciones* cuando el mecanismo de variación de direcciones para una comunicación dada da lugar a una dirección que ya está en uso en otro equipo del mismo segmento de red. La gestión de las colisiones propuesta en [1] se basa en el uso de un conjunto limitado de elementos de las secuencias pseudosaleatorias que son generados antes de establecer la comunicación, y para los que se asegura la no ocurrencia de colisiones. Una vez establecido el contexto SHIM6, sólo se pueden los localizadores y ETs hayan sido comprobados y reservados.

5 SHIP6: Privacidad Basada en SHIM6

Extender

SHIP6 es una arquitectura de privacidad basada en SHIM6 que extiende la arquitectura presentada en [1]. Por un lado, extiende la protección a todos los parámetros susceptibles de ser utilizados para relacionar flujos (localizadores, ET, identificador de fragmentación, SPI y número de secuencia IPsec, puertos, número de secuencia TCP[2]) dificultando aún más la reconstrucción de los flujos. Por otro lado, permite que la variación de estos parámetros sea ilimitada, protegiendo eficazmente a comunicaciones cuya duración o cantidad de tráfico no se conoce en el instante del establecimiento de la comunicación.

5.1 Establecimiento del Contexto Privado

Supongamos que un nodo X que desea activar las facilidades de privacidad para una comunicación con un nodo Y, utilizando en ambos casos como identificadores direcciones HBA. X inicia el establecimiento de contexto de SHIM6 con una nueva opción de *Privacy Request* que incorpora en el mensaje *I1*. De esta forma, notifica al nodo Y su deseo de obtener soporte de privacidad para el contexto SHIP6 a crear. Como una respuesta a esta solicitud (aunque también puede ocurrir de forma espontánea si el iniciador no incorporó el *Privacy Request*), el nodo correspondiente inicia la generación de clave Diffie-Hellman con el mensaje *R1*. Si el nodo correspondiente o el iniciador no intercambian el material criptográfico requerido para generar la clave Diffie-Hellman, la comunicación continúa sin soporte de privacidad. Dado que las opciones SHIM6 desconocidas son descartadas, una comunicación con un nodo que no implementa SHIP6 utiliza el intercambio SHIM6 convencional.

Si ambos nodos desean establecer una comunicación con privacidad, intercambian en los mensajes *R1* e *I2* el material necesario para generar la clave, por lo que después de la recepción del mensaje *R1* el iniciador puede crear el secreto y cifrar la información necesaria para ser incluida en el paquete *I2*. De igual forma, el mensaje *R2* puede incorporar información que sólo es visible para el iniciador.

5.2 Sincronización de Instancias de Privacidad

SHIP6 define un mecanismo que permite la variación ilimitada de localizadores y ETs. Comenzamos definiendo, para un nodo X que se comunica con un nodo Y, una *Instancia de Privacidad Local k* ($P_{loc}(X)^j$) como la siguiente tupla:

$$P_{loc}(X)^j := \langle II^j(X), ET^j(X) \rangle$$

Esta instancia $P_{loc}(X)^j$ es vista en el nodo Y como la *Instancia de Privacidad Remota k* ($P_{rem}(Y)^j = P_{loc}(X)^j$). A su vez, X mantiene una Instancia de Privacidad Remota

$$P_{rem}(X)^k = P_{loc}(Y)^k := \langle II^k(Y), ET^k(Y) \rangle$$

En un instante dado, los paquetes enviados y recibidos en el nodo X utilizan los parámetros definidos por $\langle P_{loc}(X)^j, P_{rem}(X)^k \rangle$, conocidos como los *Parámetros Actuales de Privacidad*. Además, los nodos mantienen un cierto número de Instancias de Privacidad Locales y Remotas, subsiguientes a las que definen los parámetros actuales de privacidad, para uso futuro. El mantenimiento de estas instancias permite procesar correctamente paquetes recibidos que incluyen localizadores o ETs de cualquiera de las instancias configuradas en un momento dado.

Es importante destacar que para que una instancia sea utilizable debe estar libre de colisiones, de forma que:

- Las ETs de la instancia local no estén ya en uso en otras instancias del nodo, o no sería posible

[2] A partir de ahora abreviados como LOC, ET, #FRAG, SPI, #IPsec, PRT y #TCP, respectivamente

identificar la comunicación a la que pertenece un paquete dado. La comprobación de colisión de ET implica comprobar una lista de ETs en uso o en reserva en un equipo cuando se va a generar otra ET.

- Las direcciones asociadas a la instancia local no se repitan en el mismo segmento de red. Si esta coincidencia ocurriera, la comunicación podría no desarrollarse de forma correcta. La colisión de direcciones se comprueba al configurar el equipo con las direcciones consideradas, y ejecutar el procedimiento estándar de Detección de Direcciones Duplicadas de IPv6.

Si falla alguna de estas comprobaciones, es decir, si existe una colisión, la instancia se marca como *sucia*, marcándose en caso contrario como *limpia*. Los parámetros generados para una instancia sucia, que no serán usados, son liberados para evitar que generen colisiones con otras instancias.

Dado que para las comunicaciones sólo deben usarse instancias de privacidad limpias, y que cada nodo sólo conoce inicialmente la limpieza de las instancias que le son locales, debemos definir un mecanismo que permita informar al nodo remoto acerca de la ocurrencia de colisiones. Para hacer esto, se asocia no sólo una, sino varias ETs, a cada instancia de privacidad, de forma que sus valores representen los distintos estados en la limpieza de las instancias locales de un nodo candidatas a ser utilizadas en el futuro. En concreto, para reflejar los distintos estados de limpieza de las W instancias consecutivas a la actual, es necesario disponer de 2^W ETs. Para nombrar cada ET, utilizaremos una notación basada en subíndices para los que el binario 0 representa una instancia limpia, y un 1 una sucia. Por ejemplo, para una instancia actual n y W=3, ET^n_2 (010 en binario) indicaría que las instancias n+1 y n+3 están limpias, mientras que la n+2 está sucia. Cuando se realiza un cambio en la instancia actual, el ET de entre los 2^W posibles que se incorpora en los paquetes de datos generados por el nodo X es el ET(Y) que indica el estado de las W siguientes instancias locales a X.

Como la secuencia de instancias sucias o limpias en los nodos X e Y puede ser distinta, cada nodo mantiene dos índices distintos, uno para apuntar a la instancia actual local y otro a la remota, de forma que un momento dado pueden estar en uso en el nodo X las instancias $<P_{loc}(X)^j, P_{rem}(X)^k>$ con $j \neq k$.

La activación de un cambio a la siguiente instancia tanto local como remota limpia, puede deberse a uno de los siguientes motivos:

- una decisión local, activada porque se detecta que un prefijo ya no es válido, porque ha vencido un temporizador, o porque se ha enviado una cierta cantidad de dados con la instancia actual
- la recepción de paquetes con una ET distinta de la usada en las instancias actuales. La ET debe pertenecer a alguna de las W instancias consecutivas a la actual que mantienen en un momento dado los dos pares.

Una vez que se ha producido un cambio, se actualizan los estados correspondientes a las instancias locales y remotas, de forma que siempre haya W instancias adicionales.

5.3 Generación de Parámetros de Comunicación Privados

A continuación detallamos cómo se generan los parámetros de las comunicaciones que varían con cada instancia de privacidad.

La comunicación se establece inicialmente utilizando una de las direcciones de la HBA, de forma que en la fase de establecimiento del contexto SHIM6 se validan las direcciones y prefijos a utilizar. A partir de aquí, dada una instancia de privacidad $j \geq 1$, el identificador de interfaz de un nodo X es generado como:

$$II^j(X) = hash|_{64} (PX_1::/64,\ldots PX_N::/64, RN, semX, j)$$

Siendo $PX_1::/64,\ldots PX_N::/64$ los prefijos asociados a la HBA de X, y semX la semilla generada por el nodo X. Todos estos parámetros son transmitidos privadamente en la fase del establecimiento de contexto de SHIP6. Cuando la instancia de privacidad j está activa (bien porque es la actual, o porque pertenece a las W instancias adicionales), el nodo configurará los siguientes localizadores en sus interfaces correspondientes para poder enviar y recibir paquetes con cualquiera de esas direcciones:

$$PX_1::II^j(X), PX_2::II^j(X)\ldots PX_N::II^j(X)$$

Respecto a las Etiquetas de Contexto, inicialmente no se utilizará ninguna, ya que el intercambio de datos se inicia con el identificador a ser utilizado por la comunicación. A partir de entonces, para $j \geq 1$, ya se ha razonado que en una instancia dada se configurarán 2^W ETs ($<ET^j_1(X), ET^j_2(X), \ldots ET^j_{2\wedge W}(X)>$, obteniéndose cada una como

$$ET^j_n(X) = hash|_{47} (semX, j, n)$$

Las semillas semX y semY deben ser escogidas de forma que se asegure que las instancias correspondientes a j=1 están limpias en ambos extremos de la comunicación. Esto es así porque la indicación de la limpieza al nodo remoto se realiza a través de la ET, y la ET sólo se transmite a partir del primer cambio en los localizadores, es decir, con j=1. Como consecuencia, en el momento de la generación de las semillas se comprobará que la primera instancia esté limpia, generándose en caso contrario otra semilla.

Para ocultar los parámetros relevantes del resto de un paquete que va a ser enviado, la capa SHIM6 modificada manipula ciertos campos de las capas superiores. Tras identificar los parámetros relevantes en el paquete a enviar (cabecera de transporte TCP o UDP, y si procede cabecera IPsec o de fragmentación), aplica a estos parámetros un XOR con un valor derivado de los secretos compartidos entre los extremos. Estas versiones modificadas son las que viajan por la red. Una vez en destino, la aplicación de la misma operación XOR retorna los valores iniciales. De esta forma, los parámetros son protegidos de forma transparente a las capas superiore (frgamentación, IPsec, o transporte). Las transformaciones se realizan de la siguiente manera:

$PRT^{j}(X) = PRT(X) \oplus hash(semX, semY, j, 1)$

$PRT^{j}(Y) = PRT(Y) \oplus hash(semY, semX, j, 1)$

$\#TCP^{j}(X) = \#TCP(X) \oplus hash(semX, semY, j, 2)$

$\#TCP^{j}(Y) = \#TCP(Y) \oplus hash(semY, semX, j, 2)$

$SPI^{j}(X) = SPI(X) \oplus hash(semX, semY, j, 3)$

$SPI^{j}(Y) = SPI(Y) \oplus hash(semY, semX, j, 3)$

$\#IPsec^{j}(X) = \#IPsec(X) \oplus hash(semX, semY, j, 4)$

$\#IPsec^{j}(Y) = \#IPsec(Y) \oplus hash(semY, semX, j, 4)$

Si un equipo va a enviar un paquete que necesita ser fragmentado, esta operación se habrá realizado por encima de la capa SHIP6. Es decir, que a la capa SHIP6 llegarán varios paquetes IP resultantes de la fragmentación. A estos paquetes se les aplica la siguiente transformación:

$\#FRAG^{j}(X) = \#FRAG(X) \oplus hash(semX, semY, j, 5)$

$\#FRAG^{j}(Y) = \#FRAG(Y) \oplus hash(semY, semX, j, 5)$

Dado que el valor concreto que tiene este parámetro en destino es irrelevante (siempre que todos los fragmentos muestren el mismo valor, como se asegura en la operación anterior), no es necesario que la operación se aplique también en el destino, sino que los paquetes se reconstruyen utilizando directamente el identificador de fragmentación protegido.

Es interesante destacar que desde el momento en el que se establece el contexto SHIP6 los parámetros transmitidos en los paquetes están protegidos. En concreto, los números de puertos TCP o UDP originales no aparecen en ningún momento en los paquetes que viajan por la red. Esto impide el uso de los números de puerto por parte de un nodo fisgón como pista para determinar el tipo de aplicación utilizada.

6 Trabajo Relacionado

El uso de secuencias pseudoaleatorias como mecanismo para la provisión de privacidad para el nivel de red y de transporte ha sido propuesto inicialmente en Arkko et al. [6], de donde tomamos algunas de las ideas presentadas en nuestro artículo. En este artículo también se considera el uso de permutaciones invertibles, basadas en secuencias aleatorias, como las utilizadas en nuestro caso para proteger la información de puertos, números de secuencia, etc. No obstante, no se define qué parámetros de los intercambiados actualmente son vulnerables ante un sistema espía. Tampoco se propone cómo coordinar las variaciones en las secuencias pseudoaleatorias entre dos nodos que se comunican, ni por supuesto aborda las dificultades que pueden surgir en dicho caso (como son las colisiones entre parámetros).

Otras propuestas ya han considerado la provisión de privacidad específica para protocolos de nivel de red que intercambian identificadores que pueden ser utilizados por un nodo fisgón. Por ejemplo, en [7] se analizan las vulnerabilidades de privacidad que presenta el protocolo MIPv6. En concreto, en el modo de optimización de rutas (Route Optimization), la dirección Home Address incluida en cada paquete transmitido puede utilizarse para identificar que paquetes generados desde direcciones IP distintas por un nodo que se mueve corresponden al mismo flujo. En este artículo proponen que la Home Address sea sustituida por una Etiqueta de Privacidad generada de forma pseudoaleatoria a partir de la información intercambiada a través del camino cifrado que pasa por el Home Agent. No obstante, este trabajo no contempla la provisión de privacidad para nodos que no se mueven, ya que no facilita la variación de los localizadores por otros motivos que el movimiento del nodo, y requiere del mantenimiento de un canal cifrado de comunicación. Adicionalmente, no indica cómo se sincronizan las variaciones en las etiquetas de privacidad de modo que se eviten posibles colisiones.

Es posible impedir que un nodo espía conozca la cantidad de datos reales intercambiados por una comunicación cifrada mediante la cabecera ESP. Para ello, la RFC 4303 [8] incorpora la capacidad de delimitar datos adicionales espurios en los paquetes transmitidos. El coste de esta protección es la necesidad de generar más tráfico del requerido, coste en el que no se incurre al combinar SHIP6 e IPsec. Además, el uso de SHIP6 con IPsec impide por completo (si hay un número suficiente de flujos similares) la reconstrucción del paquete en un nodo intermedio.

Otras propuestas se basan en el uso de dispositivos intermedios para la provisión de privacidad. Por un lado se propone el uso de *proxies* de nivel de aplicación para el transporte de datos cifrados a

través de la red superpuesta Onion Routing [9]. No obstante, los parámetros que pueden usarse para identificar el flujo sólo están protegidos en el camino entre los dispositivos intermedios, la protección depende de la confianza en terceras partes, y se requiere una infraestructura costosa.

7 Conclusiones

En este artículo se ha presentado SHIP6, una arquitectura para conferir privacidad a las comunicaciones establecidas entre dos nodos dificultando que un sistema espía pueda establecer relaciones entre paquetes de la misma comunicación. La arquitectura SHIP6 se desarrolla a partir del protocolo SHIM6, y permite la variación dinámica de los parámetros susceptibles de ser utilizados en cualquier punto intermedio para reconstruir un flujo.

En función de si la comunicación utiliza o no cifrado basado en ESP, se pueden considerar dos escenarios distintos, que ofrecen diferentes niveles de privacidad. Por un lado, si no se utiliza cifrado, SHIP6 permite variar todos los parámetros que pudieran ser utilizados para identificar un flujo de forma fácil, a saber: puertos locales y remotos, números de secuencia TCP (si se utiliza), identificador de fragmentación y localizadores. Si se utiliza ESP, SHIP6 permite variar los identificadores que quedan sin proteger, es decir, localizadores, SPI y número de secuencia IPsec, de forma que para un nodo espía sea imposible la reconstrucción del flujo completo.

La secuencia de variación de los parámetros tiene una longitud ilimitada, basada en la gestión de secuencias pseudoaleatorias para las que se gestiona la posibilidad de que existan colisiones entre los parámetros requeridos para demultiplexar las comunicaciones. Esto permite que los periodos entre cambios de parámetros sean tan bajos como se desee.

SHIP6 puede ser utilizado también para comunicaciones entre nodos que disponen de un solo proveedor, sobre todo si se configuran varios prefijos distintos para los segmentos de red, aun cuando estos prefijos no determinen caminos diferentes para los paquetes.

SHIP6 introduce una serie de costes en comparación con el uso de SHIM6. En primer lugar, en el instante de establecimiento del contexto es necesario establecer una clave simétrica mediante un intercambio Diffie-Hellman, y cifrar ciertos parámetros del establecimiento de contexto con dicha clave. Una vez establecido el contexto, los interlocutores deben mantener estado adicional, en forma de más direcciones, etiquetas de contexto, semillas, etc. En cuanto a costes de computación, el mayor impacto viene de la configuración de todas las direcciones asociadas a una instancia cada vez que es necesario añadir una instancia nueva al conjunto de instancias activas.

Como trabajo futuro, sería conveniente explorar de forma cuantitativa el compromiso para W, el número de instancias activas adicionales a la actual. Por un lado, debería ser suficientemente elevado para hacer insignificante la probabilidad de que W instancias consecutivas estén sucias. Por otro lado, debería ser tan pequeño como sea posible, para limitar el estado necesario.

Por otro lado sería conveniente disponer de una implementación que permita evaluar los costes, especialmente computacionales de la solución, comparándola con comunicaciones sin ningún tipo de privacidad, y con privacidad basada en IPsec.

Referencias

[1] M. Bagnulo, A. García-Martínez, A. Azcorra. “An Architecture for Network Layer Privacy”. Proceedings of the IEEE International Conference on Communications (ICC 2007). Glasgow, aceptado para su publicación, Junio 2007.

[2] E. Nordmark and M. Bagnulo, “Level 3 Multihoming Shim Protocol” IETF Internet-Draft draft-ietf-shim6-proto-07.txt (trabajo en curso), Noviembre 2006.

[3] T. Narten and R. Draves, “Privacy Extensions for Stateless Address Autoconfiguration in IPv6”, IETF RFC 3041, Enero 2001.

[4] Port Numbers, http://www.iana.org/assignments/port-numbers.

[5] M. Bagnulo, A. Garcia-Martinez and A. Azcorra, “Efficient Security for IPv6 Multihoming”, ACM Computer Communications Review, Vol. 35, n. 2, ACM Press, pp. 61-68, Abril 2005.

[6] J. Arkko, P. Nikander and M. Näslund, “Enhancing Privacy with Shared Pseudo Random Sequences”. 13th International Workshop on Security Protocols, Cambridge, 2005.

[7] R. Koodli, V. Devarapalli, H. Flinck and C. Perkins, “Solutions for IP Address Location Privacy in the presence of IP Mobility”. IETF Internet-Draft, draft-koodli-mip6-location-privacy-solutions-00.txt (trabajo en curso), 2005.

[8] S. Kent, “IP Encapsulating Security Payload (ESP)”, IETF RFC4303, Diciembre 2005.

[9] M. Reed, P. Syverson and D. Goldschlag, “Anonymous connections and Onion Routing”. IEEE J. Selected Areas in Communications 16, 4, pp. 482–494, Mayo 1998.