Seguridad XML: Su Importancia en el E-Comercio

Greg Werner, CISSP
Advantage Security, S. de R.L. de C.V.
Av. Prolongación Reforma 625, Desp. 402
Torre Lexus
Paseo de las Lomas, Santa Fe
México, DF C.P. 01330

+52 55 5292 6000
www.advantage-security.com

Índice

Introducción

El XML (Extensible Markup Language) es un estándar para intercambiar información entre diferentes sistemas informáticos. Las ventajas del XML, tal como su implementación fácil, soporte nativo y flexibilidad han permitido su crecimiento explosivo en el e-comercio. Pero con estos crecimientos de implementación han incrementado los riesgos de implementación. Por ende se han desarrollado varios estándares de seguridad XML que se están implementando hoy en día, tales como firmas digitales con XML, cifrado XML, gestión de llaves XML (XKMS), registro de llaves XML (XKRSS), seguridad de los WebServices y Firewalls XML. Este documento presenta una breve historia como surgió el XML, cómo se está implementando y los diferentes estándares y soluciones de cómo asegurar estas transacciones importantes. Finalmente se presentarán ejemplos prácticos de configuraciones seguras XML.

Historia XML

El XML surgió como estándar definido por el W3C (World Wide Web Consortium) en Febrero de 1998, en esa fecha se identificó la versión 1.0. Antes de que se definiera el XML existía el SGML (Standard Generalized Markup Language, ISO 8879), el cual se definió como estándar en 1986, pero que empezó a desarrollarse a principios de los años 70, y el mismo fue basado en el GML creado por IBM en 1969. Como podrán ver, el XML como tal puede parecer nuevo pero tiene raíces que por lo menos tres décadas, lo cual implica que este lenguaje se quedará por muchos años más.

SGML es importante porque define métodos consistentes y precisos para definir etiquetas que definen las partes dentro de un documento con el fin de que se puedan intercambiar documentos entre diferentes plataformas. A pesar de los aspectos positivos del SGML es bastante complejo su implementación y por lo tanto se produjo el XML para simplificar el lenguaje SGML. XML, como metalenguaje, tiene como propósito resguardar y estructurar datos para que se puedan mandar por Internet. El SGML también fue la base para el idioma que todos hemos escuchado con tanta frecuencia, el HTML. Aunque el éxito del HTML no se puede ignorar, su propósito es muy diferente al XML. El HTML funciona para desplegar tablas, imágenes, celdas, etc. La razón por la cual es importante mencionar estas similitudes es porque la

información XML, tal como el HTML, viaja en la mayoría de los casos por los puertos 80 y 443 (HTTP y HTTPS, respectivamente).

El idioma del Internet es un protocolo que se llama el TCP/IP. La TCP/IP es un idioma flexible que permite la definición de varios miles de servicios, tal como servicios de correo electrónico, servicios de páginas web y servicios de intercambio de datos. Varios de estos servicios son predefinidos por lo que se llaman "puertos", se establecen estándares a lo que se refiere la definición de estos puertos para incrementar la productividad en las redes. Los servicios de HTTP y HTTPS, por ser servicios de navegación de Internet, casi siempre están habilitados por medio de la apertura de los puertos 80 y 443 en el Firewall[1]. Como podrán deducir el HTML, por ser el lenguaje de navegación de Internet, coincide con los servicios HTTP y HTTPS. No obstante el XML, por ser un lenguaje cercano al HTML, también usa los puertos 80 y 443 para intercambiar información. Esta característica permite la implementación de esquemas de intercambio de información usando XML con empresas ajenas sin tener que cambiar la configuración de muchos dispositivos y servidores, ya que probablemente dichas configuraciones tienen los puertos y servicios habilitados que son necesarios para permitir la navegación a Internet por parte de sus empleados, clientes y proveedores.

Es importante recalcar que en la mayoría de los casos cuando diferentes equipos intercambian información usando el lenguaje XML probablemente lo estén haciendo por medio de los puertos 80 y 443 y en casi todas las empresas se encuentran como puertos abiertos en los Firewalls. Adicionalmente es claro que el valor de las transacciones por Internet incrementa día a día y que dichas transacciones deben de tener esquemas que garanticen la integridad, autenticidad y confidencialidad de datos.

Tal como el W3 y otras organizaciones, como el OASIS, desarrollaron estándares para el SGML, HTML y XML. Estas y otras organizaciones han desarrollado estándares de seguridad XML. En la mayoría de los casos los estándares que se han definido para el XML han tenido que ver con los esquemas de Infraestructura de Llaves Públicas (PKI, por sus siglas en Inglés). La razón por la cual muchos de los estándares de seguridad se han basado en tecnologías PKI es porque la PKI ya define estándares que se usan en el intercambio de datos para garantizar la integridad, autenticidad,

[1] Dispositivo de seguridad para filtar, bloquear y prevenir una variedad de protocolos desde y hacia las redes.

confidencialidad y en algunos casos el no-repudio de la información. Por lo tanto es importante definir los conceptos básicos de la PKI antes de explicar conceptos de firma digital con XML.

Criptografía y Llave Pública

Los problemas de autenticación y protección de la información en grandes redes de comunicación fueron analizados en 1976, en el plano teórico, por Whitfield Diffie y Martin Hellman, en un trabajo en el que explicaron sus conceptos respecto del intercambio de mensajes sin necesidad de intercambiarse llaves secretas. La idea fructificó en 1977 con la creación del Sistema Criptográfico de Llave Pública RSA, por parte de Ronald Rivest, Adi Shamir y Len Adleman, por aquel entonces profesores del Instituto de Tecnología de Massachusetts (M.I.T.). En lugar de emplear una sola llave para encriptar y desencriptar datos, el sistema RSA emplea un par combinado de llaves que desarrolla una transformación en un solo sentido. Cada llave es la función inversa de la otra, es decir, lo que una cifra, sólo la otra puede decifrar. La Llave Pública en el sistema RSA es publicada por su propietario, en tanto que la Llave Privada es mantenida en secreto.

Para enviar un mensaje privado, el emisor lo cifra con la Llave Pública del receptor deseado. Una vez que ha sido cifrado, el mensaje sólo puede ser descifrado con la Llave Privada del receptor. Inversamente, el usuario puede cifrar datos utilizando su Llave Privada. Es decir, las llaves del sistema RSA pueden ser empleadas en cualquier dirección. Esto sienta las bases para la firma digital. Si un usuario puede decifrar un mensaje con la Llave Pública de otro usuario, éste debe, necesariamente, haber utilizado su Llave Privada para cifrarlo originariamente. Desde el momento que solamente el propietario puede utilizar su propia Llave Privada, el mensaje cifrado se transforma en una especie de firma digital, un documento que nadie más ha podido crear.

La firma digital se crea tratando la información a enviar con un algoritmo de "hashing" que genera un mensaje comprimido. Este mensaje es cifrado a continuación utilizando la Llave Privada del usuario que está generando el mensaje. La firma digital sólo puede ser decifrada empleando la Llave Pública de ese mismo usuario. El receptor del mensaje decifra la firma digital y recalcula automáticamente el mensaje comprimido. El valor calculado de este nuevo mensaje comprimido se compara con el valor del

mensaje comprimido encontrado en la firma. Si los dos cálculos son iguales, significa que el mensaje no ha sido alterado. Desde el momento en que la Llave Pública del emisor fue usada para verificar la firma, el texto tiene que haber sido firmado con la Llave Privada, conocida exclusivamente por el emisor. Este proceso de autenticación será incorporado en toda aquélla aplicación que exija seguridad en las comunicaciones.

Criptografía y Llaves Públicas en XML

Firmas XML

Firmas XML son firmas digitales diseñadas para usarse con transacciones XML. El estándar define un esquema para capturar el resultado de una operación de firma digital aplicado a datos XML. La diferencia entre la firma digital XML y la firma digital de otro tipo de transacciones es que la firma digital XML fue diseñado específicamente para el XML y el Internet. Una característica fundamental de la firma digital XML es que se puede firmar ciertas partes del documento o el árbol completo. Esto puede ser relevante cuando un documento es creado y firmado por diferentes personas en diferentes tiempos y cada una de las personas firma digitalmente la información que es relevante. Puede ser posible por ejemplo que un formulario XML firmado digitalmente se mande a una persona para que la otra persona firme otra sección o bien se deja abierta una sección del formulario para que se pueda cambiar. Si se firmara el XML completamente el problema sería que cualquier cambio que se hace de ese punto en adelante invalidaría la firma digital original.

Otra ventaja de firmas XML es que se puede firmar más de un tipo de recurso. Por ejemplo, una firma XML podría firmar datos HTML, datos binarios (JPG), datos XML o una sección específica en el archivo XML. A continuación se presenta un ejemplo de un XML firmado:

```
<Signature Id="EjemploXMLSignature"
 xmlns="http://www.w3.org/2000/09/xmldsig#">
   <SignedInfo>
     <CanonicalizationMethod
      Algorithm="http://www.w3.org/TR/2001/REC-xml-c14n-20010315"/>
     <SignatureMethod
      Algorithm="http://www.w3.org/2000/09/xmldsig#dsa-sha1"/>
     <Reference
        URI="http://www.w3.org/TR/2000/REC-xhtml1-20000126/">
        <DigestMethod
          Algorithm="http://www.w3.org/2000/09/xmldsig#sha1"/>
        <DigestValue>j6lwx3rvEPO0vKtMup4NbeVu8nk=</DigestValue>
      </Reference>
   </SignedInfo>
```

```
   <SignatureValue>MC0CFFrVLtRlk=...</SignatureValue>
   <KeyInfo>
      <KeyValue>
         <DSAKeyValue>
           <p>...</p><q>...</q><g>...</g><y>...</y>
         </DSAKeyValue>
      </KeyValue>
   </KeyInfo>
</Signature>
```

Ejemplo 1: Firma XML[2]

El receptor del archivo firmado debe de poder verificar la firma del certificado digital y validar el estatus del certificado digital que corresponde con la llave privada de la fuente del documento para generar la firma digital. Otra gran ventaja del estándar XML DSIG es que se definen ciertas etiquetas para: (i) definir el recurso a firmar (ii) pasos de procesamiento para generar el recurso a firmar (iii) la digestión del recurso y (iv) la representación de la firma digital. Con estos estándares claros XML DSIG el usuario puede verificar la firma digital de la fuente para garantizar la integridad de datos y la autenticidad de la persona. La firma digital XML, tal como la firma digital tradicional, garantiza la *integridad* de datos y que la información viene de la persona quien dice ser.

Cifrado XML

Al momento de querer agregar *confidencialidad* de datos el usuario puede usar la llave pública dentro del certificado digital del receptor para cifrar datos XML. Este mecanismo se define en el estándar XML-Encryption, y tal como el estándar XML-Signature, fue definido por el W3 Consortium. El estándar de XML Encryption define el proceso de cifrar datos y representar los resultados en XML. Los datos pueden ser de cualquier tipo, incluyendo un documento XML, un elemento XML o el contenido XML. El resultado de cifrar datos es un elemento de cifrado XML que contiene referencias a los datos cifrados. El estándar de descifrado especifica como aplicaciones de firmas XML pueden distinguir entre estructuras XML-Encryption que fueron cifrados antes de firmar digitalmente los datos y aquellos que se cifraron después de firmar los datos para que se verifique la firma digital. El cifrado XML es un método en donde el contenido XML es transformado para que los receptores de la información sean los únicos que pueden ver el contenido. Hay muchas aplicaciones para el cifrado XML, una de las transacciones que más usa el cifrado XML es con procesadores de pagos por Internet. El siguiente archivo XML presente un ejemplo del XML-Encryption:

[2] http://www.w3c.es/divulgacion/guiasbreves/Seguridad © 1994-2005 W3C® (MIT, ERCIM, Keio).

```
<?xml version='1.0'?>
 <Metodopago xmlns="http://ejemplo.org/pago">
  <Nombre>Cliente Ficticio</Nombre>
  <DatosEncriptados
xmlns="http://www.w3.org/2001/04/xmlenc#"
    Tipo="http://www.w3.org/2001/04/xmlenc#Element">
    <DatosClave>
      <DatosClave>A23B45C56</DatosClave>
    </DatosClave>
  </DatosEncriptados>
</Metodopago>
```

Ejemplo 2: Cifrado XML[3]

En este ejemplo un cliente ficticio encripta una tarjeta de crédito, es importante notar que el elemento de <TarjetaCredito> se sustituye con <DatosClave> y el mismo número de tarjeta de crédito se cifra para garantizar la confidencialidad.

Intercambio de Llaves con XML

Otro estándar muy importante en la seguridad XML es el estándar de intercambio de llaves públicas, el XML Key Management Specification (XKMS). Este estándar, también definido por el W3C, especifica los protocolos para distribuir e intercambiar llaves públicas, útil para la implementación conjunta con los estándares de cifrado y firmas XML. Otro propósito por el cual se desarrolló este estándar fue para reducir la complejidad del ambiente PKI. Este estándar XKMS tiene dos partes: la especificación de información de llaves X-KISS y la especificación del servicio de registro de llaves X-KRSS. El X-KISS es un protocolo para soportar la delegación de una aplicación a un servicio de procesamiento de llaves asociado con una firma XML, cifrado XML u otra llave pública. Sus funciones incluyen el lugar de las llaves públicas requeridas y la descripción de la relación entre la llave pública y la información de identificación. La especificación X-KRSS es un protocolo para soportar el registro de un par de llaves por el dueño de dichas llaves, con la intención de que el par de llaves se puede usar subsecuentemente con el X-KISS u otro mecanismo de confianza. Estos protocolos no requieren una infraestructura de llaves públicas, tal como el X.509, pero fue diseñado para ser compatible con tales infraestructuras.

El XKMS, tal como el XML-Signature y XML-Encryption, usa la sintaxis XML y por lo tanto son diseñados para que se puedan usar los protocolos de Simple Object Message Protocol (SOAP) y WSDL (Web Services Description Language). De estos

[3] http://www.w3c.es/divulgacion/guiasbreves/Seguridad © 1994-2005 W3C® (MIT, ERCIM, Keio).

protocolos es posible generara API´s (Application Programming Interfaces) en otros idiomas comunes, tal como C.

Seguridad de WebServices

WSDL es un formato XML para describir servicios de red como una serie de nodos operando sobre mensajes que contiene información orientado a documentos o procedimientos. Las operaciones y mensajes son descritos abstractamente y después son limitados a un protocolo concreto de red y formato de mensaje para definir un nodo. Estos nodos relacionados se combinan en nodos abstractos para definir servicios web. WSDL es extensible para permitir la descripción de nodos y sus mensajes con cualquier tipo de protocolos que se usen en la comunicación, pero el WSDL generalmente trabaja en conjunto con SOAP 1.1, HTTP GET/POST, y MIME. Supongamos por ejemplo que un cliente desea intercambiar información con otra entidad pero no sabe cómo se definen los elementos de datos. El cliente obtiene la definición por medio del WSDL, estructura sus elementos como se define en el WSDL y empieza a intercambiar información.

El SOAP (ahora en la versión 1.1) también se definió por el W3C, es una especificación de mensajes XML que describe el formato del mensaje con reglas de serialización para diferentes tipos de datos. Adicionalmente describe como se puede usar el http como transporte para estos mensajes. Mensajes SOAP son solicitudes de servicios mandados a un nodo en la red, este nodo se puede implementar de varias meneras, por ejemplo con un Servlet de Java, un Modelo de Objeto de Componente (COM), un script de Perl, por ejemplo, y puede estar ejecutando en una variedad de plataformas. Es decir SOAP permite la interoperabilidad entre diferentes aplicaciones sobre plataformas diversas y en lugares diversos.

Ahora bien existe la firma de los datos XML, ¿entonces porqué es necesario definir estándares adicionales para los servicios web particularmente con SOAP y WSDL? Pues resulta que la firma XML se usa para firmar los encabezados SOAP para garantizar la integridad y autenticidad de los datos. Como se mencionó en el párrafo anterior el SOAP puede intercambiar una variedad de datos que pueden estar dentro de un XML, por lo tanto es importante garantizar los encabezados SOAP de estos mensajes para evitar definiciones erróneas. También es importante autenticar y garantizar la integridad de los mensajes SOAP que están solicitando servicios en

diferentes redes y plataformas. El SOAP se comunica en la mayoría de los casos via los puertos 80 y 443 que ya están abiertos en la mayoría de los firewalls en el mercado y por lo tanto es importante garantizar la seguridad de estos mensajes cuando entran y salen de las redes.

Firewalls XML

Los firewalls XML se desarrollaron específicamente para filtrar y prevenir ataques por medio de transacciones XML que entran y salen por la red de una institución o empresa. Un firewall tradicional es un dispositivo que filtra la información que entra y sale de la red por medio de la apertura de puertos específicos, visto de otra manera, son dispositivos que se usan para implementar políticas de seguridad para el egreso e ingreso de información de una red. Aunque los firewalls tradicionales hacen un excelente trabajo (con tal que estén configurados correctamente) en permitir la entrada y salida de información no inspeccionan paquetes XML individuales para detectar vulnerabilidades XML. Estas vulnerabilidades pueden ser verificaciones de formatos “well formed” (bien formados), verificaciones de buffer overflow, validación de esquemas XML y protección contra ataques de denegación de servicios.

Conclusión

La seguridad XML sigue evolucionando con el crecimiento acelarado del Internet y de redes IP pero la seguridad XML solamente puede proteger contra ataques si se implementa correctamente. Adicionalmente la seguridad XML se debe de implementar con herramientas que automaticen la implementación y prevención de ataques para reducir la complejidad inherente y para reducir el riesgo contra ataques y estas implementaciones de seguridad solamente son posibles con la capacitación a nivel dirección de las empresas para que se pueda invertir en los recursos necesarios para evitar pérdidas mayores en un futuro.

Referencias

Recursos sobre seguridad del W3C
Colección de recursos sobre diferentes temas en relación a la seguridad en la Web.

Sintaxis y Procesamiento de XML-Signature
Recomendación del W3C.

Sintaxis y Procesamiento de XML Encryption
Recomendación del W3C.