Seguridad y Protección de Sistemas Informáticos

Fco. Javier Lobillo Borrero

Departamento de Álgebra, Universidad de Granada

Curso 2017/2018

Índice

Índice

Claves públicas: propietarios y validez

Una clave pública:

- ¿Ha sido alterada?
- ¿Pertenece a quien dice ser su propietario?
- ¿Para qué funciones sirve?
- ¿Continúa siendo válida?

Los certificados digitales y las infraestructuras de clave pública (PKI) dan respuesta a estas y otras preguntas.

Qué es un certificado

Un certificado de llave pública es una afirmación, firmada digitalmente por una entidad emisora, que asegura que la clave pública (e información adicional) de la entidad propietaria del certificado tiene un valor específico.

Un certificado digital contiene:

- Claves públicas del propietario.
- Información del propietario (por ejemplo datos del usuario tales como el nombre, su identificador, etc.)
- Información de los emisores.
- Una o más firmas digitales de los emisores.

Si confiamos en la entidad que emite el certificado y éste es auténtico tenemos la garantía de que la información contenida en el certificado en veraz, en particular las claves públicas en él contenidas pertenecen a la entidad propietaria del mismo.

Índice

Anillos de confianza

Todos los participantes del anillo pueden convertirse en emisores de certificados. Es el esquema empleado en PGP y sus derivados como GnuPG y OpenPGP.

- Permiten establecer niveles de confianza y de extensión.
- Su fortaleza iguala a la menor de sus participantes.
- No necesitan infraestructura adicional.
- Cada certificado puede estar firmado por más de un emisor.
- Puede ser difícil encontrar la información necesaria para verificar una firma.

Anillos de confianza: transitividad

Uno de los elementos que los anillos de confianza pretenden destacar es la "transitividad" de las claves firmadas. La idea es que debería considerar como válida una clave firmada por alguien cuya clave es válida para mi. Este sistema tal cual presenta serias deficiencias que vamos a tratar de solventar. Una clave se considera **válida** para un usuario si

- ha sido firmada por el propio usuario,
- ha sido firmada por varias entidades tales que todas juntas proporcionan suficiente nivel de confianza para el usuario

Anillos de confianza: confianza en usuarios

Cuatro niveles de confianza

unknown No se sabe nada sobre el dueño de la clave firmante. Suele ser el nivel por defecto asignado a una clave.

none Se sabe que el propietario firma otras claves de modo impropio.

marginal El propietario comprende las implicaciones de firmar una clave y valida las claves de forma correcta antes de firmarlas.

full El propietario comprende perfectamente las implicaciones de firmar una clave y su firma sobre una clave es tan buena como la nuestra.

Anillos de confianza: validez de la clave

Una clave se considera válida si

1. ha sido firmada
   - personalmente,
   - por un usuario con nivel de confianza pleno,
   - por tres usuarios con nivel de confianza marginal;
2. el camino que nos lleva desde nuestra clave hasta la clave firmada de cinco pasos o menos.

Los principales problemas vienen dados de la subjetividad implícita a la hora de conceder niveles de confianza, y a la gestión de los certificados caducados o revocados.

Estructuras jerárquicas

Hay una Autoridad de Certificación (CA) principal (raíz) que emite certificados a CAs subordinadas. Éstas a su vez certifican a usuarios y dispositivos en función del uso y validez de las claves.

- El certificado está firmado únicamente por el emisor.
- Suele incluir los datos necesarios para verificar la identidad del emisor a partir de la CA raíz.
- Su fortaleza depende de la política de la autoridad certificadora.
- Facilidad para buscar la información necesaria para verificar la información.

Autoridades de Certificación (CA) I

Objeto y Funciones

- Constituyen el elemento principal de la cadena de certificación.
- Técnicamente es un certificado auto-firmado.
- Si la CA se ve comprometida, todo el sistema queda invalidado.
- Sus funciones son
  - Actuar como una entidad de confianza.
  - Verificar la identidad de los solicitantes.
  - Emitir los certificados.

Autoridades de Certificación (CA) II

Públicas vs. Privadas

Privadas: Son creadas por una entidad concreta y delimitada para dar servicios de certificación dentro de esa misma entidad. No tienen validez fuera de dicha entidad. Hay libertad plena para diseñar la información que aparece en el certificado, así como para modificar y adaptar la política de certificación.

Públicas: Funcionan a un nivel mucho más amplio, por ejemplo en todo Internet. Suelen estar mantenidas y operadas por empresas que tienen este propósito como negocio principal, cobrando una tasa por ello, o por instituciones gubernamentales. El contenido del certificado, tipos de nombres y atributos, está mucho más limitado y debe adaptarse completamente a estándares definidos.

CA subordinadas

Objeto y funciones

- Mismas funciones que una CA:
  - Actuar como una entidad de confianza.
  - Verificar la identidad de los solicitantes.
  - Emitir los certificados.
- Técnicamente es un certificado firmado por una CA de nivel superior.
- Pueden crearse en base a criterios diversos:
  - Tipo de propietario: usuario, dispositivo, etc.
  - Uso del certificado: cifrado, firma, firma de CRL, firma de certificados, etc.
  - Geográficos: país, región, ciudad...

Autoridades de Registro (RA)

Objeto y funciones

- Dependen de una CA de nivel superior, ya sea raíz o subordinada.
- Sus funciones son:
  - Recibir las solicitudes de certificados.
  - Verificar la identidad del solicitante.
- No realizan tareas de firma digital.

Índice

Estructura

Un certificado X.509 consta de tres partes

`tbsCertificate` Este campo contiene los nombres del emisor y del propietario, la clave pública asociada al propietario, un periodo de validez e información adicional.

`signatureAlgorithm` Contiene el identificador del algoritmo criptográfico usado por la CA para firmar este certificado.

`signatureValue` Aquí encontramos la firma digital calculada sobre el campo `tbsCertificate` codificado en formato ASN.1 DER.

tbsCertificate

version Es un entero que admite los valores 0, 1 o 2 según la versión sea v1, v2 o v3.

serialNumber Entero positivo asignado por la CA al certificado. Único para cada certificado emitido por esa CA.

signature El identificador del algoritmo usado por la CA para firmar el certificado. Debe coincidir con el valor de `signatureAlgorithm`.

issuer Emisor identificado mediante un DN (nombre completo).

validity Intervalo de tiempo durante el cual la CA garantiza que mantendrá la información sobre el estado del certificado. Contiene dos fechas, `notBefore` y `notAfter`.

subject Propietario identificado mediante un DN.

subjectPublicKeyInfo Este campo contiene el valor de la clave pública y el identificador del algoritmo para el que esta clave se usa (RSA, DSA...)

issuerUniqueID No válido para v1. Contiene un identificador único del emisor. El objetivo es la posible reutilización del nombre del emisor. Su uso no se recomienda hoy en día.

ubjectUniqueID Idem para el propietario.

extensions Válidas para la v3.

signatureAlgorithm y signatureValue

El campo `signatureAlgorithm` contiene el identificador del algoritmo criptográfico usado por la CA para firmar este certificado. Aunque algunos algoritmos son considerados como aceptables dentro de los diferentes estándares, otros no considerados inicialmente pueden también ser utilizados.

El campo `signatureValue` contiene la firma digital calculada sobre `tbsCertificate` codificada según el estándar ASN.1 DER. El valor de la firma se codifica como una BIT STRING y se incluye en este campo.
Mediante la generación de esta firma, la CA certifica la validez de la información contenida en `tbsCertificate`. En particular, la CA certifica la correspondencia entre la clave pública y el propietario del certificado.

DN (Nombre completo)

El DN está formado por atributos. En principio cualquier tipo de atributo está permitido, pero las implementaciones deben reconocer al menos los siguientes:

- país,
- organización,
- unidad organizativa,
- clasificación del nombre completo,
- estado o provincia,
- nombre común,
- número de serie.

También deberían poder interpretar los siguientes:

- localidad,
- título,
- apellido,
- nombre,
- iniciales,
- pseudónimo,
- clasificador de la generación (“Jr.”, “3rd”, “IV”)

Extensiones I

Definidas para la v3, proporcionan métodos para asociar atributos adicionales del usuario o de la clave pública, así como gestionar relaciones entre CAs. Las extensiones deben clasificarse como

críticas el certificado debe ser rechazado si se encuentra una extensión crítica no reconocida o si su valor no puede ser procesado.

no críticas si no son reconocidas pueden ignorarse, pero si son reconocidas deben procesarse.

Aunque todas ellas son opcionales, en ocasiones la presencia o valores de alguna de ellas obliga o prohibe la presencia de otras. De igual forma, los certificados propiedad de las CAs deben contener algunas de las extensiones estándar.

Extensiones II

Extensiones estándar

- Identificador de la clave de la autoridad.
- Identificador de la clave del propietario.
- Uso de la clave.
- Políticas de los certificados.
- Asignaciones de la política.
- Nombre alternativo del propietario.
- Nombre alternativo del emisor.
- Atributos del directorio del propietario.
- Restricciones básicas.
- Restricciones del nombre.
- Política de restricciones.
- Uso de la clave extendido.
- Puntos de distribución de las CRL.
- CRL más actualizado.

Extensiones III

Extensiones privadas de Internet

- Información de acceso a la autoridad.
- Información de acceso al propietario.

Índice

Distribución y renovación

Distribución

Para solicitar un certificado a una CA suelen producirse los siguientes pasos.

1. Generación de pareja de claves pública/privada.
2. `Certificate Signing Request` conteniendo exclusivamente la clave pública.
3. Verificación de identidad.
4. Firma con la clave privada de la CA.
5. Devolución al propietario.

Renovación

Sigue los mismos pasos que en la distribución con la excepción de que la verificación de la identidad puede hacerse mediante un certificado existente del mismo propietario y próximo a caducar.

Verificación y revocación

Un certificado es correcto si:

- La firma es correcta.
- Nos encontramos dentro del periodo de validez del mismo.
- No ha sido revocado.

Cómo revocar un certificado

- `Certificate Revocation List`
- `Online Certificate Status Protocol`

Los motivos para revocar un certificado pueden ser variados:

- Descuido del propietario.
- Problema de seguridad en el equipo del propietario.
- Problema de seguridad en cualquier algoritmo de los usados en cifrado o firma.
- Problema de seguridad en algún equipo del emisor.

CRL

- Una CRL es una lista que contiene números de serie de certificados emitidos por una CA junto que la fecha a partir de la cual están revocados. Permite impedir el uso de un certificado antes de su fecha de caducidad.
- Puede almacenarse en el servidor de la CA que lo gestiona o en un servidor externo (recomendado).
- También es recomendable que el lugar al que acceden usuarios y dispositivos sea distinto que el empleado por la CA.
- Una CRL tiene también un periodo de validez. Una vez caducada, una nueva CRL debe ser generada y descargada por los distintos servidores.
- Las CRL no actúan en tiempo real. Las CRL son actualizadas y descargadas periódicamente.

OCSP

- Trata de evitar dos de los problemas de las `CRL`: el tamaño y el tiempo de vida.
- Se solicita al `OSCP Server` información concreta sobre la validez de un certificado.
- El `OSCP Server` comunica con la CA puntualmente o periódicamente. En este último caso los periodos son mucho más cortos que los empleados por las `CRL`
- Las respuestas pueden ser: *good*, *revoked*, *unknown*.
- El tipo *unknown* genera problemas semánticos.

¿Es necesaria la revocación?

Certificado de una CA comprometido: Revocación fácil de hallar. Se publicita y se informa de modo amplio.

Certificado de un servidor comprometido: Revocación difícil de hallar.

- Puede darse con cierta facilidad en servidores pobremente protegidos.
- El certificado se envía habitualmente en el *handshake*, por lo que no es necesaria la revocación. Basta con cambiar el certificado.

Certificado de usuario comprometido: El número de usuarios que pueden necesitar acceder al certificado de un usuario concreto es pequeño. Pueden enviarse avisos personalizados.

Alternativas

- Asociar certificados a tarjetas de crédito.
- Asociar certificados a cuentas en servidores.

Índice

Concepto

Un sello temporal confiable es un sello temporal emitido por una tercera entidad en la que se confía, que actúa como TSA (*Time Stamp Authority*). Se emplea para demostrar la existencia de ciertos datos antes de una fecha concreta. Al contrario que con las CAs, varias TSA son admisibles y deseables para unos datos concretos.

Podemos encontrarlos en los siguientes estándares:

- RFC 3161 . Sistema básico
- ANSI ASC X9.95, que añade integridad en los datos mediante un sello temporal de confianza.

TSA en RFC 3161

- Debe tener acceso a un reloj altamente confiable.
- Debe incluir un valor temporal confiable en cada sello temporal.
- Debe incluir un número de serie único en cada sello temporal.
- Debe producir un sello temporal al recibir una solicitud válida.
- Debe incluir en cada sello temporal un identificador que determine la política de seguridad empleada para producirlo.
- Debe producir el sello temporal sobre un hash de los datos, y nunca sobre los datos propios.
- Debe verificar que el la longitud del hash coincide con la descripción del hash empleado.
- No debe tratar de examinar los datos a los que se va a aplicar el sello temporal.
- No debe incluir en el sello temporal ninguna identificación del solicitante.
- Debe firmar cada sello temporal con una clave privada generada con éste único propósito, y debe tener esta propiedad de la clave indicada en el correspondiente certificado.
- Solo debe incluir en el sello temporal aquellas extensiones soportadas por la TSA.

Solicitud de un sello temporal

TimeStampReq

- version
- messageImprint
- reqPolicy
- nonce
- certReq
- extensions

messageImprint

- hashAlgorithm
- hashedMessage

Respuesta de un sello temporal I

TimeStampResp

- status PKIstatusInfo
- timeStampToken TimeStampToken

PKIstatusInfo

- status PKIStatus
- statusString PKIFreeText
- failInfo PKIFailureInfo

Respuesta de un sello temporal II

`PKIStatus`

- granted (0),
- grantedWithMods (1),
- rejection (2),
- waiting (3),
- revocationWarning (4),
- revocationNotification (5)

Respuesta de un sello temporal III

PKIFailureInfo

- badAlg (0),
- badRequest (2),
- badDataFormat (5),
- timeNotAvailable (14),
- unacceptedPolicy (15),
- unacceptedExtension (16),
- addInfoNotAvailable (17),
- systemFailure (25)

TimeStampToken

- contentType id-signedData
- content SignedData

Respuesta de un sello temporal IV

TSTInfo

- version INTEGER v1(1) ,
- policy TSAPolicyId,
- messageImprint
- serialNumber
- genTime GeneralizedTime,
- accuracy Accuracy OPTIONAL,
- ordering BOOLEAN DEFAULT FALSE,
- nonce INTEGER OPTIONAL,
- tsa [0] GeneralName OPTIONAL,
- extensions [1] IMPLICIT Extensions OPTIONAL

Solicitud, respuesta y verificación

Resumen de solicitud y respuesta.

- El solicitante envía un hash de los datos a sellar junto con información adicional necesaria para la creación del sello.
- La TSA concatena dicho hash con el sello temporal, vuelve a calcular el hash de dicha cadena, firma ese hash con su clave privada (cuya finalidad es exclusiva para firmar sellos temporales), y envía el valor de dicha firma junto con el sello temporal.
- El solicitante almacena conjuntamente sus datos, la firma y el sello.

Verificación

- Se calcula el hash de los datos. Este hash se concatena con el sello temporal.
- Se calcula el hash de la cadena anterior y se verifica mediante la clave pública del TSA si la firma es válida para dicha cadena.

Bibliografía I


C. Adams, P. Cain, D. Pinkas, and R. Zuccherato.
Internet X.509 Public Key Infrastructure Time-Stamp Protocol (TSP).
IETF, August 2001.


Ingemar J. Cox, Matthew L. Miller, Jeffrey A. Bloom, Jessica Fridrich, and Ton Kalker.
Digital Watermarking and Steganography.
The Morgan Kaufmann Series in Multimedia Information and Systems. Elsevier, second edition edition, 2008.


D. Cooper, S. Santesson, S. Farrell, S. Boeyen, R. Housley, and W. Polk.
Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile.
IETF, May 2008.


Quynh H. Dang.
The Keyed-Hash Message Authentication Code (HMAC).
National Institute of Standards and Technology (NIST), July 2008.

Bibliografía II


Kresimir Delac and Mislav Grgic.
A survey of biometric recognition methods.
In *46th International Symposium Electronics in Marine, ELMAR-2004*, pages 184–193, 2004.


Hans Delfs and Helmut Knebl.
Introduction to Cryptography. Principles and Applications.
Information Security and Cryptography. Springer, third edition, 2015.


T. Dierks and E. Rescorla.
The Transport Layer Security (TLS) Protocol Version 1.2.
IETF, August 2008.


Educause.
7 things you should know about Federared Identity Management, September 2009.

Peter Gutmann.
Everything you Never Wanted to Know about PKI but were Forced to Find Out.
Technical report, University of Auckland.

Bibliografía III

ISO/IEC.
Information technology — Security techniques — A framework for identity management —, December 2011.

Joseph Migga Kizza.
Guide to Computer Network Security.
Computer Communications and Networks. Springer, 3rd. edition, 2015.

S. Kent and K. Seo.
Security Architecture for the Internet Protocol.
IETF, December 2005.

Andre Karamanian, Srinivas Tenneti, and Francois Dessart.
PKI Uncovered: Certificate-Based Security Solutions for Next-Generation Networks.
Cisco Press, 2011.

Bibliografía IV


Carlos Munuera.
Steganography from a Coding Theory Point of View, pages 83–128.
WORLD SCIENTIFIC, 2013.


Satoshi Nakamoto.
Bitcoin: un sistema de dinero en efectivo electrónico peer-to-peer.
Technical report, bitcoin.org.
Traducido por @breathingdog.


National Institute of Standards and Technology (NIST).
DATA ENCRYPTION STANDARD (DES), October 1999.


National Institute of Standards and Technology (NIST).
ADVANCED ENCRYPTION STANDARD (AES), November 2001.


National Institute of Standards and Technology (NIST).
SECURE HASH STANDARD, August 2002.

Bibliografía V


National Institute of Standards and Technology (NIST).
Digital Signature Standard (DSS), July 2013.


Josef Pieprzyk, Thomas Hardjono, and Jennifer Seberry.
Fundamentals of Computer Security.
Springer, 2003.