1 April 2017

Networks & Trust

Introduction to PKIs & CAs

Author
Kevin Meynell

Internet Society

Introduction

The Internet Society recognises that in order to be trusted, the Internet must provide channels for secure and private communication between entities, which can be clearly authenticated in a mutually understood manner. The mechanisms that provide this function must support both the end-to-end nature of Internet architecture and reasonable means for entities to manage and protect their own identity details.

There are several commonly used mechanisms for supporting secure and private communication, transaction protection and identity assertion and management. These include the so-called Internet PKI commonly used for secure web browsing but which can be used for other applications, PKI for e-mail, RPKI used by Regional Internet Registries to assert the holders of IP resources, and DNSSEC that can be used to validate DNS queries. DANE is a new protocol that uses DNSSEC to allow owners to assert their own digital certificates, and therefore potentially incorporate the functionality of the Internet PKI into the global DNS. This document provides an explanation of how these mechanisms work and how they are deployed.

What is a Public Key Infrastructure?

A Public Key Infrastructure (PKI) is a system of managing linked pairs of cryptographic keys, one of which is private, and the other of which can be safely published in the form of a digital certificate. PKIs facilitate the secure transfer of data across the Internet, as well as being a means of identifying computer systems and users. This is undertaken through the issuing of digital certificates which associate a public key with a given individual or resource.

PKIs usually form part of a hybrid system in combination with symmetric key encryption; the symmetric and public key elements each being best suited to particular functions. Public key cryptography solves some of the problems of distributing keys among communicating partners and is useful for digitally signing messages and applications to provide evidence of their origin and integrity. However, public key encryption is computationally intensive and symmetric encryption offers much higher throughput, but needs a secure and convenient way of distributing the shared keys (which is where PKI can help).

Internet-related PKIs use certificates issued by a CA based on the ITU-T's X.509 version 3 standard, but using a specific profile defined in RFC 5280. The CA acts as a trusted third party that verifies the certain aspects of the identity of the domain holder, organisation or individual requesting a certificate, and thereby provides assurances about the identity of these entities to other parties.

The most common type of certificate is the server certificate which is also widely known as a TLS or SSL certificate. These are issued to domain holders and organisations to provide assurances about the identity of a server that a client wishes to communicate with. Their most familiar usage is for secure web browsing (using the TLS and previously the now deprecated SSL protocols) as indicated by the padlock icon that appears in web browsers when a secure session is established, although they are increasingly used for other applications such as e-mail, file transfers, video/audioconferencing and instant messaging.

Another type of certificate is the code-signing certificate which is issued to software authors. Vendors such as Apple and Microsoft are starting to mandate the use of code-signing as this reduces the possibility of malicious software being run on their operating systems, which has been a significant problem with programs downloaded from the Internet.

The third and final type of certificate is the client certificate, which is sometimes known as a personal certificate although a client certificate can actually be used to identify either an individual or a computer system. The issue of these certificates is less widespread largely due to the relative complexities of storing, securing and using these on end systems, although they have found applications in e-mail signing/encryption and document signing amongst others.

How does Public Key cryptography work?

Public Key or Asymmetric cryptography uses key pairs - a public key, and a private key. The public key is mathematically related to the private key, but given sufficient key length, it is computationally impractical to derive the private key from the public key. This allows the public key of the recipient to be used by the sender to encrypt the data they wish to send to them, but that data can only be decrypted with the private key of the recipient.

The advantage of asymmetric cryptography over symmetric cryptography is that the process of sharing encryption keys does not have to be secure, but the mathematical relationship between public and private keys means that much larger key sizes are required. The recommended minimum key length is 1024 bits, with 2048 bits preferred, but this is up to a thousand times more computationally intensive than symmetric keys of equivalent strength (e.g. a 2048-bit asymmetric key is approximately equivalent to a 112-bit symmetric key) and makes asymmetric encryption too slow for many purposes.

For this reason, public key cryptography is typically used for initial authentication and securely generating and exchanging a session key. The session key can then used for encrypting the data transmitted by one party, and for decrypting the data received at the other end, with the session key being discarded once the session is over. A variety of different key generation and exchange methods are in use, including RSA, Diffie-Hellman (DH), Ephemeral Diffie-Hellman (DHE), Elliptic Curve Diffie-Hellman (ECDH) and Ephemeral Elliptic Curve Diffie-Hellman (ECDHE). Public key cryptography can also be used to sign a message or document, thereby confirming the authenticity of the content as well as providing non-repudiation. The private key of the sender is used to sign the data being sent, which can then verified by the recipient using the public key of the sender. Commonly used cryptographic hash functions include MD5, SHA-1 and SHA-2, although MD5 and SHA-1 are now considered insecure and should not be used. The SHA-3 standard was released in August 2015, although this is intended to be an alternative hash function rather than a replacement for SHA-2.

Why should I care about PKIs?

The Internet was built on inherent trust as well as limited computing resources, and as a result data has historically been transmitted unencrypted. Where it was used, it was typically employed in a piecemeal fashion for sensitive information such as passwords or payment details. Whilst it was recognised back in 1996 (RFC 1984) that the growth of the Internet would require private data to be protected, it has become increasingly apparent over the intervening period that the capabilities of eavesdroppers and attackers are greater and more pervasive than previously thought. The IAB therefore released a statement in November 2014 calling on protocol designers, developers, and operators to make encryption the norm for Internet traffic, which essentially means making it confidential by default.

In addition, critical infrastructure such as the routing system and DNS is still largely run in a cooperative manner with little or no verification of the authenticity of routing updates or responses to DNS queries. This has led to erroneous information being propagated across the Internet

resulting in traffic being misdirected and parts of the Internet becoming inaccessible whether for accidental or malicious reasons.

Public Key cryptography can support the encryption of traffic across the Internet (using the TLS protocol), be used for validating the assignment of IP addresses (with RPKI), as well as for validating the ownership of domain names (with DNSSEC). However, a PKI is required in order to manage the public keys and link ownership of them to a particular entity.

The ownership and authenticity of a public key is normally asserted by a CA which issues a digital certificate. In some cases, a server may use a self-signed certificate which needs to be explicitly trusted by a client (browsers should display a warning when an untrusted certificate is encountered), but this may be acceptable in private networks and/or where secure certificate distribution is possible. It is highly recommended though, to use certificates issued by publicly trusted CAs.

What is a CA?

A Certificate Authority (CA) is an entity that issues digital certificates conforming to the ITU-T's X.509 standard for Public Key Infrastructures (PKIs). Digital certificates certify the public key of the owner of the certificate (known as the subject), and a CA therefore acts as a trusted middle man that gives other parties (known as relying parties) assurance they dealing with a validated entity (known as an end entity). End entities can be a domain holder, organisation or individual depending on the type of certificate being used.

End entity certificates are themselves validated through a chain-of-trust originating from a root certificate, otherwise known as the trust anchor. With public key cryptography it is possible to use the private key of the root certificate to sign other certificates, which can then be validated using the public key of the root certificate and therefore inherit the trust of the issuing CA. In practice, end entity certificates are usually signed by one or more intermediate certificates (sometimes known as subordinate or sub-CAs) as this can limit the necessity of revoking a root certificate in the event that an end entity certificate is incorrectly issued or compromised.

Root certificate trust is normally established through physical distribution of the root certificates in operating systems or browsers. The main certification programs are run by Microsoft (Windows & Windows Phone), Apple (OSX & iOS) and Mozilla (Firefox & Linux) and require CAs to conform to stringent technical requirements and complete an appropriate WebTrust, ETSI or ISO audit in order to be included in their distributions.

Root certificates distributed with major operating systems and browsers are said to be publicly or globally trusted and the technical and audit requirements essentially means the issuing CAs are multinational corporations or governments. There are currently around fifty publicly trusted CAs, although most/all have more than one root certificate, and most are also members of the CA/Browser Forum which develops industry guidelines for issuing and managing certificates.
It is however also possible to establish private CAs and establish trust through secure distribution and installation of root certificates on client systems. Examples include the Regional Internet Registries which operate RPKI CAs providing trust anchors for certificates issued to Local Internet Registries attesting to the IP addresses and AS numbers they hold. In these cases, the root certificates are securely downloaded and installed from sites using a certificate issued by a publicly trusted CA.

How do I establish a publicly trusted CA?

Becoming a publicly trusted root CA is a complex and expensive process. The major operating systems and browser vendors all have their own compliance programmes for including root certificates, which means it is necessary to fulfil the requirements for each operating system or browser distribution. The CA/Browser Forum aims to establish common standards to simplify this process (see 'Baseline Requirements for Issuance and Management of Publicly-Trusted Certificates'), but as yet there is no definitive list of publicly trusted CAs.

Nevertheless, there are certain commonalities in that a prospective root CA must conform to the requirements of the AICPA/CICA WebTrust Program for Certification Authorities, ETSI EN 319 411-3 (formerly TS 102 042) or ISO 21188:2006. This requires that the CA undergoes annual audits for compliance in the areas of policy and operational management; physical, environmental and personnel security; systems development and maintenance, business continuity; and monitoring and auditing requirements. Such audits typically take around 2 to 6 months to complete and cost USD 75 to 375K depending on complexity and number of sub-CAs.

The costs of the necessary hosting and support infrastructure are also significant, and a typical figure for establishing a root CA would in the order of USD 600-900K, with recurrent costs of around USD 375K per year. Furthermore, it should also be pointed out there are no guarantees that a CA will be accepted into a public root distribution even if it fulfils the necessary criteria. There are technical and security reasons not to propagate too many root certificates, and the inclusion of a CA must generally have some public interest. Even once accepted, it takes one or two years for the root certificate to fully propagate into operating systems and browsers, which means this requires a long-term commitment from the CA.

Given the complexity, expense and lead times required to become a publicly trusted root CA, it has therefore been common for governments, communities or organisations with a need for their own CAs to establish these as a sub-CA under an existing root CA. Although even this process has become more complex as the industry has needed to introduce more rigorous requirements on certificate issuance, it is still likely to be a more cost effective option for those CAs expecting to issue under 100,000 or so certificates.

What do I need to worry about?

One inherent weakness with Internet PKIs is that third parties (CAs) are able to issue certificates for any domain or organisation, whether or not the requesting entity actually owns or otherwise controls it. The risk of a CA issuing an incorrect certificate rises as the number of CAs increases, and trust is in the PKI system is only as strong as the weakest link which is the main reason why the public root distributions seek to limit the inclusion of CAs.

In the case of server certificates, validation is typically performed through domain validation - namely sending an e-mail with an authentication link to an address known to be administratively responsible for the domain. This is usually one of the standard contact addresses such as 'hostmaster@domain' or the technical contact listed a WHOIS database, but this leaves itself open to man-in-the-middle attacks on the DNS or BGP protocols, or more simply, users registering administrative addresses on domains that have not been reserved. Perhaps more importantly, Domain Validated (DV) certificates do not assert that a domain has any relationship with a legal entity, even though a domain may appear to have one.

This weakness is also of particular concern with code-signing certificates, as if a malicious entity is able to masquerade as a well-known software publisher, it is possible to distribute malware that will

be implicitly trusted and thereby installed by operating systems. With operating systems increasingly requiring digitally signed applications, a compromised certificate obviously has the potential to hack or disrupt a large number of systems.

For this reason, CAs are increasingly encouraging the use of Organisation Validated (OV) and Extended Validation (EV) certificates for both server and code-signing applications. With OV certificates, the requesting entity is subject to additional checks such as confirmation of organisation name, address and telephone number using public databases. With EV certificates, there are additional checks on legal establishment, physical location, and the identity of the individuals purporting to act on behalf of the requesting entity. Browsers normally display the validated organisation name in green when a valid EV certificate is encountered, although there is unfortunately no easy way of distinguishing an OV from a DV certificate.

Of course, even if user interfaces are considered an acceptable method for determining trust, this method is difficult to apply to automated systems such as e-mail servers. It also does not prevent CAs accidentally or fraudulently issuing incorrect certificates, and there have been incidents of security breaches where CAs were tricked into issuing fake certificates. Despite substantial tightening up of security procedures in the wake of several high-profile incidents, the system remains reliant on third party trust which has led to the recent development of the DNS-based Authentication of Named Entities (DANE) protocol.

With DANE, a domain administrator can certify their public keys storing them in the DNS, or alternatively specifying which certificates should be accepted by a client. This requires the use of DNSSEC which cryptographically asserts the validity of DNS records, although DNSSEC does not yet have widespread deployment and major browsers currently require installation of an add-on in order to support DANE. Moreover, DNSSEC and DANE still requires some validation of the domain holders which will likely have to be undertaken by domain registries and/or registrars instead of CAs. An advantage though, is that the DNS contains Country Code Top-Level Domains (ccTLDs) that are managed on a national basis and could therefore be used for a national PKI.
The situation with client certificates is less clear as there are no common standards for validating individuals (or end systems). Individuals are typically validated with e-mail addresses, but whilst these are unique identifiers, they cannot definitively be linked to an actual person (or end system). Some CAs, especially those that are government operated, require the presentation of government issued ID and/or require the use of various forms of two-factor authentication (which can include sending an access code to a mailing address) to provide further assurances about certificate holders. However, this is a resource intensive process with privacy concerns, and there's an increasing trend towards other forms of electronic authentication given the usability issues with client certificates.

What is RPKI?

Resource Public Key Infrastructure (RPKI) is a specialised PKI that aims to improve the security of the Internet routing system, specifically the Border Gateway Protocol (BGP). It does this through the issuing of X.509-based resource certificates to holders of IP addresses and AS numbers in order to prove assignment of these resources. These certificates are issued to Local Internet Registries (LIRs) by one of the five Regional Internet Registries (RIRs) - AfriNIC, APNIC, ARIN, LACNIC and RIPE NCC - who have responsibility for allocation and assignment of these resources in their service regions. Number resources are not allocated or assigned on a national basis with the exception of seven legacy National Internet Registries (NIRs) in the APNIC region.

Resource certificates allow LIRs to generate Route Origin Authorisations (ROAs) which attest to which networks (specifically AS numbers) are authorised to originate which ranges of IP addresses

(specifically IP prefixes). This then allows other networks to determine whether route announcements are valid and should therefore be accepted, thus reducing the likelihood of erroneous or fake routes being propagated across the Internet.

Each RIR acts as a CA and trust anchor for the resources assigned within their service regions, although their root certificates are not included in any public root distributions. It is therefore necessary to download and install these from the RIR websites.

What is DNSSEC?

The purpose of the Domain Name System (DNS) is to translate human readable host names such as 'www.isoc.org' into machine readable IP addresses such as 212.110.167.157. It functions as a distributed hierarchy in which IANA (with approval from the US Department of Commerce) delegates authority from the root zone to each of the registries operating the 1,000 or so top-level domains (TLDs) including the Generic TLDs (gTLDs) and Country Code TLDs (ccTLDs). Under each TLD, there are a number of other domains run by the TLD registry or other organisations, which in turn may establish sub-domains under the domain for which they have authority. Thus the DNS can be visualised as a tree-like structure where administrative control is successively delegated from one organisation to another.

The DNS has become the main method by which to locate Internet services, largely due to its simplicity and scalability. Unfortunately though, it has several drawbacks in that its distributed nature means that changes do not propagate across the Internet instantly (due to caching and the need for zone transfers), and because many different organisations control the DNS which makes it difficult to ensure that information is being returned from a reliable source. In other words, there are no guarantees that a name server is not providing false information to direct users to hosts that monitor their transactions or masquerade as other sites.

As a result, DNS Security Extensions (DNSSEC) were devised by the IETF to authenticate DNS information. This uses public key cryptography that allows operators to digitally sign their DNS records which ensures only the domain holder can make changes, and that these records can be validated through a chain-of-trust as they will in turn by signed by the parent key of the delegating authority and so on up to the root zone. This ensures that a client making a query is able to verify that the returned answer is actually from an entity authorised to provide it.

DNSSEC can therefore be considered a specialised type of PKI as it adds cryptographic assertions to the DNS. This is why the DANE protocol was developed to extend its functionality to support X.509 certificates that can only be asserted by domain holders rather a third-party CA, although the domain holders still need to be validated by the delegating authority.