Certificate Authority, its Problems and Alternative Approaches

Mbombui Nongho Fon-Pah

Uni Duisburg-Essen

***Abstract*—Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetuer id, vulputate a, magna. Donec vehicula augue eu neque. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Mauris ut leo. Cras viverra metus rhoncus sem. Nulla et lectus vestibulum urna fringilla ultrices. Phasellus eu tellus sit amet tortor gravida placerat. Integer sapien est, iaculis in, pretium quis, viverra ac, nunc. Praesent eget sem vel leo ultrices bibendum. Aenean faucibus. Morbi dolor nulla, malesuada eu, pulvinar at, mollis ac, nulla. Curabitur auctor semper nulla. Donec varius orci eget risus. Duis nibh mi, congue eu, accumsan eleifend, sagittis quis, diam. Duis eget orci sit amet orci dignissim rutrum.**

I. Introduction

Users of the internet have for over a decade needed a way to securely visit data(mail, websites, servers, etc.). Nonetheless, most frequently accessed data sources cannot for instance actively push their crypto certificates to all of their users (websites like Google would need to exactly know ahead of time who will visit them). Furthermore, pushing a crypto key over the same communication channel that could be a subject to a Man in the Middle (MitM) attack,could cause key learning vulnerable too. Still internet users require a method to ”securely” learn crypto certificates for all data sources that they will ever(inlcuding those they may never visit), and acertain if these certifcate holders(named entities) are ”trustworthy” (that means, a named entity may still act maliciouly even though it has a valid certificate).

Nowaredays, we depend on certain organizations called Certificate Authority(CAs) to do this task for us. That is, these are organisations the Relyign Parties can trust to confirm both ”*authenticity*” and ”*trustworthiness*” of all the named entities' certificates. Each of these CAs is tied to a certificated of its own. Large software vendors like Apple, Mozilla Foundation, Microsoft, etc often configure the X.509 certificates for approximately 160 CAs for internet users when their products are installed. These root certificates form the base-line for certificate validation by softwares like web browsers. Here, the reponsibility to choose and update the CA lists is done by the software vendors, and the verification of all data sources that a user may ever visit is done by these root CAs. However the CA Model has some ”weaknesses” which make it vulnerable to serveral attacks which are both difficult to detect and/or easy to implement. More details about these liaiblities will be discussed in Section IV and specific examples in Section V

The DNS Security Extensions(DNSSEC)[6], [7], [8] has recently become an operationally important technology, and their usage has been growing constantly for six years now[5]. It has provided the a means for domain owners to directly manage their security in same distributed database that internet users trust to find their service(DNS). The DNS-Based Authentication of Named Entities(DANE) offers the possibility to use DNSSEC to validate TLS Keys and certificates used by HTTPS and other TLS-based protocols[1]. Furthermore, a couple of commercial products such as Firefox have add-ons for most browsers while others like Google's Chrome [10] have integrated a native support for the DANE. A remarkable advantage with the DANE approach is that it uses an already existing infrastructure(DNS), which has been used for online transactions and which a vast majority of internet client acting as relying parties(RP) are alread addicted to,(for example it is rare to find a URL that is made up of IP address) attest certificates. Thereby decreasing the system's dependencies on addtional systems and

protocols and consequently reducing the overall attack chances.

Perspectives on the other hand tries to prevent these attack by using a collection of ”notary” servesr that observes named data source’s public key through serveral network vantage point and keep records fo a server’s key over time . Clients can download these records, when needed and compare them against unauthentuicated keys, there by preventing attacks. Key observations gathered over the multiple vantage points, makes it difficult for an attacker to compromise all the network paths to destination ,notary data that will allow the client to discover that an attack is eminent(*spial redundancy*) [0]. Futhermore, it enable clients to identify malicious notaries that supply inconsistent data,and there reducing the damage of attack on the notary infrastructure(*data redundancy*).

Name Constraints is an extension of X.509 Version 3 certifcate. It defines a name space in which all subsequent certificates in the certification path **MUST** be located [rfc]. That a top-level domain can only validate certificate of subdomain in its namespace. In case of an attack on a top-level domain(its cerificate is compromised) only certifacates of sub-domains in that namespace will be affected, therey reduicing the attack surface. Details on the above mentioned alternative approaches will be discussed in Section VI.

II. Certificate Authority(CA) Landscape

Certificate Authority or Certification Authority(CA), is an organization that is responsible for distributing digital certificates in accordance with a specified Certification Policy. This digital certificate confirms the ownership of a public key by the named subject of the certificate.These CAs make sure that relying parties(RA) can count on the signatures and assertions made by the private key that matches the public key that is certified.

III. Internet X.509 Public Key Infrastructure(PKI) Certificate

Without certificate authorities and X.509, users would be at risk from having their TCP connections (SMTP,HTTP, etc) hijacked or compromised. X.509 and certificate authorites are characteristics of public key infrastructure (PKI) schema.

A. Overview of PKI Model

PKI is an infrastructure that provide internet users with the means to securely and privately exchange information through the use of a public and a private key pair obtained and shared through a trusted authority. It also provides digital certificates that matches an individual or organization, a reporsitory that strores and when necessary can revoke the certficates.The component of a public key infrastructure are:

End Entity : Users, organizations, or systems who intends to use the private and public technology to securely exchange information.

Certificate Authority(CA) : The oraganization that issues digital certificates binding subject’s identity with subjects’s public key.

Registration Authority(RA) : An optional system to which CA delegate some functions like verifying the subject’s identity.

Certificate Revocation List(CRL) Issuer: An optional system to which CA delegate the function to publish the Certificate revocation list containing certificates revoked by the CA.

Validation Authority(VA) : An optional system to which CA delegate some functions like verifying the digital certificate subject.

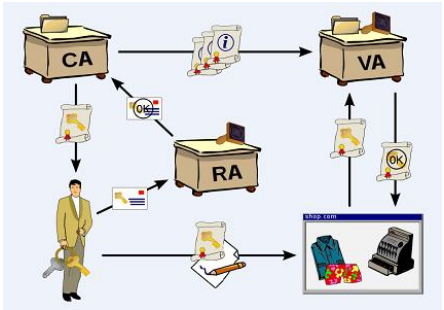

Figure 1.

B. X.509 Certificate Format

C. Certificate Validation Process

D. Subordinate Certificate Authority(SubCA) or Intermidiary CA

IV. Liabilities: An Assault on One Defeats All

V. Prominent Attacks

VI. Alternative Approaches

A. DNS-based Authentication of Name Entities(DANE)

Domain name authentication if a fundamental function for Internet security. In order for applications that communicate over the intertnet to protect their communication from eavesdropping, tampering or forgery, they need to make sure that the entity on the end point of a secure communcation actually reoresents the domain that the user intended to connect to. The Domain Name System Security Extentions (DNSSEC) provides an alternative path for distribution of secure information about domnain names, via Domain Name System(DNS) itself. The DANE working group in the Internet Engineering Task Force(IETF) has developed a new type of DNS record called TLSA record that permits a domain itself to ratify statements about which organisations are empowered to represent or vouch for it. End users' applications can use thes records either to establish a new chain of trust, rooted in the DNS or to increase the existing system of Certificate Authorities.

1) TLS Authentication: The Transport Layer Security(TLS) protocol provides secure client-server connection for many internet applications[2]. In all of these internet applications, the server that the user in the end want to connect to is identified by a DNS domain name[7][8]. An internet user might enter https://example.de into a web browser or send an e.mail to jane@xample.de. So in this case th using the TLS assures the user that the entity on the other far end of the connection actually represents example.de; that is to certify the server as a true representative of the domain name. It is also to be noted that these comments apply to Datagram Transport Layer Security(DTLS) due to the fact that it provides the same functions as TLS for User Datagram Protocol(UDP) packet flow.

2) The TLSA Resource Record(DANE Record): The TLSA DNS resource record (RR) is used to match a TLS server certificate or public key with the domain name where the record is found, thus forming a "TLSA certificate association"[RFC-6698]. The DANE use cases document[RFC-6698] lays out four main types of statements that permit domain operators to define how clients should judge TLS certificate of their domains. These statements are:

1) CA Constraints: The client should only accept certifcates published under a specific CA.
2) Service Certificate Constraints: The client should only accept specific certificates for specific services on a host.
3) Trust anchor assertion: The client should use domain-provided trust anchors to validate certificates fo that domain.
4) Domain-issued Certificate Constraints: The client should only accept certificate issued by the domain name adminstrator itself.

The major difference between the constraints 2. and 4. ist 2. requires that the certificate pass PKIX validation and 4. not.
All the above statements can be seen as constraining the scope of trust anchors. The first, second and forth types limits the scope of existing trust anchors; the third type provides the client with a new trus anchor(but still within a limited scope).

The DANE TLSA resource record has four major fields:

- The Certificate Usage Field: This field specifies one of the above mentioned statement types(constraints) as the assoiation that will be used to match the certificate presented in the TLS handshake.
- The Selector Field: This field specifies which part of the TLS certificate presented by the server with be matched against the association data. That is, the full certificate or its SubjecPublicKeyInfo
- The Matching Type Field: This field specifies how the certificate association is matched. These types are: Exact match on, SHA-256 hash of and SHA-512 of selected content[RFC-6234][RFC-6234].

- The Certificate Association Data Field: This field contains the actual data against which the TLS certificate chain should be matched.

The DANE RR is stored under the target domain with a prefix that indicate the port number of the TLS server and the transport protocol on which a TLS-based service is assumed to exist. TLSA RR Examples: if John runs a secure webservice at john.com and wants to tell clients to only accept certificates from Jane's CA, he could supply a TLSA record under $_{4}4_{t}cp_{j}ohn.comwithfollowingcontent$:

Usage: CA constraint
Selector: full certificate
Matching type: SHA-256
Certificate for Association: SHA-256 of Jane's certificate

So when client Donald wants to connect to "https://john.com", he can find these TLSA RR and appy John's constraint when he validates the server'certificate.

3) Comparing DANE to Public CAs:

4) Transition Chanllenges:

5) Security Considerations:

B. The Perpertive Approach

C. Name Constraint Approach

VII. Conclusion

References

[1] M. Castro, P. Druschel, A. marie Kermarrec, and A. Rowstron, "Scribe: A large-scale and decentralized application-level multicast infrastructure," *IEEE Journal on Selected Areas in Communications (JSAC*, vol. 20, p. 2002, 2002.

[2] J. Postel, "Transmission Control Protocol," IETF, RFC 793, September 1981. [Online]. Available: http://tools.ietf.org/html/rfc793

[3] Y. Wang, Z. Lu, and J. Gu, "Research on symmetric nat traversal in p2p applications," *Some Journal*, p. 59, 2006.