Network Working Group
Internet-Draft
Intended status: Informational
Expires: January 31, 2013

S. Bellovin
Columbia University
R. Bush
Internet Initiative Japan
R. Housley
Vigil Security, LLC
S. Kent
BBN Technologies
S. Turner
IECA, Inc.
August 2012

Trust Anchor Publication Advice
draft-ymbk-ta-publication-00

Abstract

Many Internet protocols and services rely on credentials which use asymmetric keys. Many of these are hierarchic structures having certification authorities (CAs) that act as trust anchors (TAs). There is little general guidance on procedures for how these trust anchors can be distributed or otherwise published with prudence. To quote a well known security expert, "It's a matter of oral tradition in security circles." This document attempts to capture some of that lore.

Status of This Memo

Copyright Notice

Table of Contents

1. Introduction

Many Internet protocols and services make use of asymmetric keys distributed via certificates (e.g., X.509 or OPGP) or analogous formats (e.g., DNSSEC records). Many of these certificates are organized into hierarchic structures having one or more trust anchors (TAs). In any hierarchical structure, the choice of root is important. In PKIs, it is quite critical, since an untrustworthy or incompentent TA can issue credentials to imposters, vitiating the security guarantees for the entire structure. This in turn implies that anyone relying on a PKI must have accurate knowledge of the root of the tree. However, there is little general guidance on procedures for how these trust anchors can be distributed in a fashion that ensures their integrity and authenticity. To quote a well known security expert, "It's a matter of oral tradition in security circles." This document attempts to capture some of that lore.

In particular, the issue of publication of root TA(s) for the

Resource Public Key Infrastructure (RPKI) incited this document. We recommend it be handled similarly to the DNSSEC root keys, see Section 4.5.

We do not address the complex matter of generating key pairs for trust anchors. They range from exceedingly formal and complex, e.g. [icann-dnssec] for DNSSEC, to the exceedingly informal, e.g. [pgp-gen]. We assume the public key material and associated date have been created, and address the problem of distribution and/or publication of the TA materiel in a secure fashion.

The distribution/publication problem is how to give relying parties (RPs) who will use the TA confidence that the trust anchor is authentic. In this context authentic means that the public key and associated data has not been modified in an unauthorized fashion, and the data associated with the TA accurately identifies the principle that it represents. There is usually no external trust environment which is the same as that of the TA; after all this is a TA. So the problem often devolves to issues of identity and trust in the conveyance or conveyor of the TA. This is often referred to as 'Out Of Band' (OOB) verification of the TA.

Fundamentally, one can trust information if it came via a trusted path and/or was delivered by a trustworthy source. We refer to these as "conveyance" and "conveyor". In addition, one can build up trust by suitable conbination of information from many different sources; this gives rise to a variety of hybrid schemes.

Note carefully that there is no one solution for all situations. The proper answer depends on the operational needs, and often on the particular hardware and software involved.

2. Trust in the Conveyance

2.1. TLS / https

For some applications, an HTTP GET authenticated with TLS [RFC5785] may be sufficient. Given the number of certificates in the normal browser, many consider this imprudent and suggest that the user should ensure that the certification path validates to a particular TA that they trust for introducing other TAs. This may be beyond the average user

Use of further authenticity such as DANE, see [I-D.ietf-dane-protocol] is another approach.

Microsoft distributes new browser TAs in the same manner as software updates, which rely on certificate path validation. Thus the entropy of the browser's certificate store can only increase.

2.2. In Packaged Software

Embedding a TA in software is a common method of distribution in many contexts. In an enterprise context this may suffice, e.g., if software distribution is tightly controlled by the enterprise. Most operating systems and browsers use this method, as the vendors of these products are dealing with a set of RPs that is large, geographically dispersed, and unknown to the TA management. But, this approach is not without risks.

For example, an RP who receives an OS copy on a DVD in conjunction with the purchase of a laptop is probably confident that the TA(s) embedded in that OS have not been modified and that the vendor has vouched for the accuracy of the TA material. In contrast, if a copy of a browser is downloaded via the Internet, the set of TAs embedded in it may or may not be what the browser vendor intended. Attacks on the DNS (absent DNSSEC), or on the server from which the browser image was acquired could have resulted in bogus TA material.

3. Trust in the Conveyor

For applications where the credential is that of an identity, authentication of the conveyor might be appropriate.

3.1. PGP

Pretty Good Privacy (PGP) [RFC4880] is based on personal identity, and "Uses a combination of strong public-key and symmetric cryptography to provide security services for electronic communications and data storage."

PGP itself actually has no root of trust, but rather is a web of trust sans root. It would not be of extreme interest here except it has some of the few well-documented rituals of authenticating exchange of credentials involving fingerprints (hashes) of keys [pgp-party]. There is also a system of coordinated key servers [REF NEEDED].

3.2. Physical Proximity

In some environments it is possible to provide good physical, personnel, and procedural security for TA distribution. This is especially easy if the set of RPs is small, geographically local, and known to the TA management.

For example, in an organization an employee might receive a smart

card loaded with a personal certificate and private key, and the TA for the organization. If the organization distributes this card to the employee in person, e.g., as a side effect of employee (or student) orientation, the employee can probably rely on the authenticity of the TA. The DoD Common Access Card (CAC) delivers TA material in this fashion, through a network of verification officers and associated work stations.

4. Advice yet to be Organized

4.1. Public Transport Plus Verification

Less secure TA distribution mechanism are often employed when the RP population is very large, or geographically dispersed, or not known by TA management a priori.

For example, a smart card loaded with a TA might be sent to an RP via the postal system. The RP, upon receipt if the card, can't be absolutely sure that the TA represents the entity identified on the card or in accompanying documentation. If registered mail is employed the likelihood of tampering en route might be considered very small, but the identity of the sender still would not be assured. Thus some means of independently verifying that aspect of TA security would still be needed. Depending on the context, such verification might be easy, or very difficult. For example, if the card is designed to enable access to a bank account, the RP might try to use it and see of the bank balances reported match what the RP expects. If the RP need not provide a password or other secret value to gain access to the account this is a reasonable way for the RP to verify that the card "works" and that it probably was issued by their bank, and thus the TA on the card is likely associated with the bank.

4.2. TAMP

The Trust Anchor Management Protocol (TAMP [RFC4255]) is a transport independent protocol for the management of trust anchors and community identifiers stored in a trust anchor store.

The core concept is not complex. Trust one signer to be the one to introduce other public keys as trust anchors, and those may have constraints (one for signed software, one for TLS, only for IPsec, or whatever). Complexity comes if you want to allow that one signer to pass the privilege to another signer.

More needed here.

4.3. SSH

Secure Shell (SSH [RFC4251]) authenticates [RFC4252] over the SSH Transport Layer Protocol, [RFC4253]. The server may authenticate the user's identity by a number of means, password, asymmetric key, challenge response, etc. The user authenticates the server by an asymmetric key. That key may have been transmitted out of band, e.g. using DNSSEC [RFC4255] or some other credible (to the user) means. SSH also offers a 'trust the transport' key conveyance, with manual hash verification, for the first connection to the server.

4.4. DNSSEC

DNSSEC relies on a diverse public distribution mechanism to distribute the TA material for the DNS root, see [icann-dnssec]. The DNS root TA material is available in multiple formats (e.g., S/MIME, PEM certificate request, XML, and OPGP), and from multiple sites (e.g., iana.org, ???). An RP that acquires the DNS TA material from multiple sources can verify that the public key values it acquires all match. An adversary would have to spoof replies from all of the queried sources to fool an RP. Moreover, if the RP received bogus DNS root TA data, the RP would not be able to validate legitimate DNSSEC records. Thus an adversary would have to insert itself in the RPs (DNESEC) communication path on a persistent basis to avoid detection. This is perceived as a difficult task and thus the DNSSEC mechanism for distributing TA material is viewed as adequate.

4.5. RPKI

Trust Anchor Locators (TALs) are used to distribute TAs in the RPKI. A TAL is a URI and a self-signed X.509 certificate. There is no plan to publish TALs in multiple formats, as the TAL format itself is quite simple.

It is anticipated that the root TAL, like the DNSSEC root zone TA material, will be published by the IANA similarly to the DNSSEC root keys, see [icann-dnssec]. Until the IANA signs the root TAL for the RPKI, it is anticipated that the next level in the hierarchy, the RIRs, will each publish a TAL using analogous means.

Unlike the certificates in browsers, the IANA and RIRs are a small and static set of TAL publishers. It should be easier to distribute them in a more credible fashion.

4.5.1. Hardware Security Module

If a TA is used only offline, and one employs a good HSM (e.g., FIPS 140 level 4) then it is very, very unlikely that the private key will be compromised and not detected. IANA and the RIRs can afford to protect their keys this way, so one should rarely have to change these TAs. Thus the principle problem is how an RP can become confident that the TAL data it acquires is legitimate. The same principle applies here as for DNSSEC. Each RP should acquire TALs from multiple locations and verify that the data are consistent. Each RP will be downloading and verifying data from multiple RPKI repositories. If the TAL(s) acquired by an RP are not accurate, then legitimate RPKI data acquired from repositories will not validate. Thus an adversary would have to insert itself in the RPs RPKI repository communication path on a persistent basis to avoid detection. This is perceived as a difficult task and thus the RPKI mechanism for distributing TA material is viewed as adequate.

5. Acknowledgements

The authors would like to thank Shane Amante for inciting this effort.

6. References

[I-D.ietf-dane-protocol]
Hoffman, P. and J. Schlyter, "The DNS-Based Authentication of Named Entities (DANE) Transport Layer Security (TLS) Protocol: TLSA", Internet-Draft draft-ietf-dane-protocol-23, June 2012.

[RFC4251] Ylonen, T. and C. Lonvick, "The Secure Shell (SSH) Protocol Architecture", RFC 4251, January 2006.

[RFC4252] Ylonen, T. and C. Lonvick, "The Secure Shell (SSH) Authentication Protocol", RFC 4252, January 2006.

[RFC4253] Ylonen, T. and C. Lonvick, "The Secure Shell (SSH) Transport Layer Protocol", RFC 4253, January 2006.

[RFC4255] Schlyter, J. and W. Griffin, "Using DNS to Securely Publish Secure Shell (SSH) Key Fingerprints", RFC 4255, January 2006.

[RFC4880] Callas, J., Donnerhacke, L., Finney, H., Shaw, D. and R. Thayer, "OpenPGP Message Format", RFC 4880, November 2007.

[RFC5785] Nottingham, M. and E. Hammer-Lahav, "Defining Well-Known Uniform Resource Identifiers (URIs)", RFC 5785, April 2010.

[RFC5934] Housley, R., Ashmore, S. and C. Wallace, "Trust Anchor Management Protocol (TAMP)", RFC 5934, August 2010.

[icann-dnssec]

Ljunggren, F., Okubo, T., Lamb, R., Schlyter, J., "DNSSEC Practice Statement for the Root Zone KSK Operator", 2010, <https://www.iana.org/dnssec/icann-dps.txt>.

[pgp-gen] cets@seas.upenn.edu, "Generating PGP Keys", , <http://www.seas.upenn.edu/cets/answers/pgp_keys.html>.

[pgp-party] Brennen, A. V., "The Keysigning Party HOWTO", 2008, <http://cryptnet.net/fdp/crypto/keysigning_party/en/keysigning_party.html>.

Authors' Addresses

Steven M. Bellovin
Columbia University
1214 Amsterdam Avenue, MC 0401
New York, New York 10027
US

Phone: +1 212 939 7149
Email: bellovin@acm.org

Randy Bush
Internet Initiative Japan
5147 Crystal Springs
Bainbridge Island, Washington 98110
US

Email: randy@psg.com

Russell Housley
Vigil Security, LLC
918 Spring Knoll Drive
Herndon, VA 20170
US

Email: housley@vigilsec.com

Stephen Kent
BBN Technologies
10 Moulton St.
Cambridge, MA 02138
US

Email: kent@bbn.com

Sean Turner
IECA, Inc.
3057 Nutley Street, Suite 106
Fairfax, Virginia 22031
US

Email: turners@ieca.com