Network Working Group
Internet-Draft
Intended status: Informational
Expires: August 29, 2015

F. Templin, Ed.
Boeing Research & Technology
S. Burleigh
JPL, Calif. Inst. Of Technology
February 25, 2015

DTN Security Key Management - Requirements and Design
draft-templin-dtnskmreq-00.txt

Abstract

Delay/Disruption Tolerant Networking (DTN) introduces a network model in which communications may be subject to long delays and/or intermittent connectivity. These challenges render traditional security key management mechanisms infeasible since round trip delays may exceed the duration of communication opportunities. This document therefore proposes requirements and outlines a design for security key management in DTNs.

Status of This Memo

This Internet-Draft is submitted in full conformance with the provisions of BCP 78 and BCP 79.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at http://datatracker.ietf.org/drafts/current/.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

This Internet-Draft will expire on August 29, 2015.

Copyright Notice

Table of Contents

1. Introduction

The Delay/Disruption Tolerant Network (DTN) architecture [RFC4838] introduces a data communications concept in which "bundles" of data are exchanged in store-and-forward fashion between endpoints that may be separated by long-delay or intermittently-connected paths. The Bundle Protocol Specification [RFC5050] provides the bundle message format and operations, including convergence layer transmission, fragmentation and custody transfer. Each bundle further may include extensions, among which may be security parameters designed to ensure confidentiality, integrity and authentication [RFC6257][I-D.irtf-dtnrg-sbsp]. These securing mechanisms (termed "Bundle Security Protocol") operate within the constraints imposed by various "ciphersuites". Prominent among these are ciphersuites that

rely on public/private key pairs where the public key is used to encrypt data and verify signatures while the private key is used to decrypt data and sign messages. Like any other public/private key system, however, Delay Tolerant Networks require some form of Public Key Infrastructure (PKI) to ensure that private key holders are properly authorized to use them as attested by a trusted Certificate Authority (CA) [RFC4210].

Public key cryptography in DTNs may be in some ways simpler than in traditional Internet security approaches. In particular, some BSP ciphersuites impose no need for peers to establish a long-term secret "symmetric" session key to be applied across a stream of bundles in the way that protocols such as the Internet Key Exchange (IKE) [RFC5996] establish session keys to be applied across a stream of packets. Instead, per the provisions of these ciphersuites, each bundle carries its own secret symmetric key in which the bundle is encrypted (in which case the symmetric key is itself encrypted in the public key of the receiver) or by which the bundle is signed (in which case the symmetric key is itself signed in the private key of the sender).

While the operation of the DTN securing mechanisms themselves can be applied independently of the key management scheme, in their current incarnation they can only be used with pre-placed irrevocable keys since there are no published mechanisms for automated security key management. On the surface, the use of standard PKI mechanisms would seem to be a natural fit, but traditional methods are not appropriate for long-delay and/or disrupted paths. This issue has prompted earlier IRTF investigations into an automated key management scheme for DTN [I-D.farrell-dtnrg-km][I-D.irtf-dtnrg-sec-overview], and it was also highlighted in "A Bundle of Problems" [WOOD08], Section 4.13 and "Security Analysis of DTN Architecture and Bundle Protocol Specification for Space-Based Networks" [IVAN09].

Therefore, an automated system for the publication and revocation of public keys will be necessary for many DTN applications, and that system must be designed to function in the presence of long delays and/or intermittent connectivity. The system must provide timely delivery of new public keys and security-key meta-data even though the delay inherent in the system may result in actual conveyance to DTN nodes long after transmission. Moreover the improper operation of this system, whether caused by malfunction or by a deliberate attack, could have significant impact on the usability of the network; the system must therefore be highly resistant to operational failure. In this document, we discuss the problem, provide requirements and propose a design for a suitable solution.

2. Discussion

Traditional automated PKI key management protocols allow for a subject (aka "end entity") to create a self-generated public/private key pair and then register the public key with a trusted Certificate Authority (CA) [RFC4210]. However, in a network based on DTN there may be significant delays between the time at which an end entity requests another entity's certificate and the time at which the requested certificate is delivered. Also, issues such as the publication of a new key pair can result in communication failures if end entities do not discover the new public key until some time after the old public key is deprecated. Alternatives such as a "web of trust" (e.g., via Pretty Good Privacy (PGP) [RFC4880]) may have application in some DTNs, but this is a topic for further study.

An old adage that also needs to be addressed is whether there is a "one-size-fits-all" solution. DTNs may come in various shapes and sizes, and various approaches may be better suited to some DTNs than others. More specifically, in the future there may not be one "DTN" in the same way that there is one public Internet. But rather, there may be many DTNs for public or private use - each with its own operational capabilities and constraints.

There will likely be ways to accomplish public key publication in the presence of long delays and/or disruptions, since keys can be published to take effect at some point in the future. However, timely certificate revocation may be infeasible due to the long delays inherent in many DTNs. DTN subjects therefore must be vigilant in ascertaining the degree to which long-delay correspondents can be trusted. These and many more issues must be carefully considered in any design.

3. DTN Security Key Management Core Requirements

A number of fundamental requirements must be satisfied by any security key management design for DTN. The requirements include the following:

3.1. REQ1: Must Provide Keys When Needed

The practical significance of this requirement is that the DTN security key management design must not rely on timely responses to queries directed to a Public Key Infrastructure (PKI). Low-delay online access using standard Internet connections (i.e., TCP/IP) may never be available. Even if the query is submitted using some delay-tolerant protocol, the opportunity to use the key to encrypt or verify data may have ended by the time the key arrives. In short, traditional PKIs are considered incompatible with DTN.

3.2. REQ2: Must Be Trustworthy

The design must be based on a trust anchor common to all nodes in the DTN network. A common trust anchor is needed to ensure that all DTN nodes will receive public keys from a secured key authority and not from an anonymous source. In particular, DTN nodes cannot simply accept public keys directly from one another with no prior trust basis. Otherwise, the network and all devices that use it could be compromised. The trust anchor should store and forward only authentic public keys from DTKA Key Authorities in an authentic manner so that the unavailability of DTKA Key Authorities will not prevent or delay communications between any two DTN nodes.

3.3. REQ3: No Single Point of Failure

The design must not introduce a single point of failure; the system must not fail in the event that one or more critical infrastructure elements are damaged. In particular, DTN nodes cannot always depend on receiving information from any single key authority node, since that node may not always be reachable over the network, may be subject to failures such as power outages, or may be compromised by an attacker. Much like the way RAID disc arrays operate, the system must be resilient to one or more failures.

3.4. REQ4: Multiple Points of Authority

The design must not introduce a single point of authority that could degrade the entire network if hijacked by an attacker. In particular, DTN nodes must never be forced to trust information provided by any single key authority node without corroboration by other key authority nodes.

3.5. REQ5: No Veto

Correspondingly, the design must never enable any single key authority node (possibly hijacked by an attacker) to degrade the network by declining to corroborate the information provided by other key authority nodes.

3.6. REQ6: Must Bind Public Key with DTN Node Identity

This requirement is about the claim for binding a public key with the ID of a DTN node. The key authority must certify the association of a public key with an identified DTN node when and only when that association is asserted by some entity that the key authority trusts. The mechanism by which such assertions are communicated must itself be secured. This requirement is a generic requirement for all secure Public Key Infrastructures.

3.7. REQ7: Must Support Secure Bootstrapping of a Node's Identity and its Public Key

The Key Authority must authorize the use of the association between a Node's identity and its public key, along with other administrative information, in its DTN. Such association is essentially random and cannot be verified in an automated manner. Thus, the association must be verified manually before the Key Authority can approve the use of the association in its DTN.

3.8. REQ8: Must Support Revocation

The DTN PKI must provide a mechanism that allows Certificate Authorities to revoke a certificate even before the certificate expires.

3.9. REQ9: Revocations Must Be Delay Tolerant

The propagation of information about revocation of issued and valid certificates must use DTN only. DTN certificate revocation must not assume the application will employ low-delay communications to verify public key certificates as is normal in the terrestrial Internet, where the Online Certificate Status Protocol (OCSP) is available to verify the absence of a public key in the revocation list in an on-demand manner.

4. DTN Security Key Management Design Criteria

We believe these core requirements imply several structural guidelines on security key management design for DTN. A candidate DTN security key management design can be formulated according to the following design criteria:

4.1. DC1: Must Perform Timely Key Provisioning

The design must ensure that security keys are put in place before they are actually needed. For example, if a source signs a bundle of data using its private key, each DTN node in the path may require access to the public key before the bundle arrives. Otherwise, the bundle could be rejected due to security policy. This means that DTN nodes must generate public/private key pairs and assert them to the key authority long in advance of when they would actually be needed.

4.2. DC2: Pub/Sub Model

The design must be based on a publish/subscribe model instead of an online (pull-based, or client/server) directory service, since on-demand retrieval from a traditional server is not possible in many

DTN environments due to delays/disruptions. One alternative is for the key authority to publish public key "bulletins" to which all DTN nodes subscribe. The bulletins must reach all DTN nodes in the network over the same long-delay links that carry ordinary data bundles. Bulletins therefore must convey keys to be used at some point in the future.

4.3. DC3: Publication Must Be Spread Over Multiple KAs

The key management system's responsibility for distributing key information bulletins must be spread across multiple Key Authority Nodes (KAs); a monolithic bulletin generated by a single KA would violate requirements 3, 4, and 5. The cooperating KA nodes must publish fractionated data that can be aggregated to reconstitute the original bulletin; it must never be possible for the compromise of any single KA to result in reception of an inauthentic bulletin. Specifically, the KAs must agree on a bulletin through control message exchanges, after which each KA publishes a few overlapping fragments of the bulletin instead of the full bulletin. Each DTN node then receives the fragments and reassembles them into a complete bulletin. In this way, it is OK if one or more of the KAs fails because the fragments are overlapping and DTN nodes will be able to reconstruct the full bulletin. It is also OK if one or more of the KAs has been hacked, because the integrity of the bulletin will be ensured by the consensus agreement of all KAs. However, at least a few non-compromised KAs (functioning as trust anchors) must be present and reachable for the system to survive with assured integrity.

4.4. DC4: Availability and Security

Like all other critical infrastructure elements, the key management system must be maintained as highly available and hardened against compromise. The latter requirement may require strong physical security, e.g., secured data centers, hardened mobile platforms, etc. This is no different than for other core network services such as the Domain Name System (DNS), Dynamic Host Configuration Protocol (DHCP) and many others. As in all other networking operations, nodes depend on at least occasional contact with critical infrastructure. Where fully ad-hoc networks are needed, dynamic key distribution may not be feasible. In that case, permanent Pre-Placed Keys (PPK) and/or limited-scope pairwise key exchanges may be the only solution alternatives.

5. Candidate DTN Security Key Management Design

We anticipate a security model for DTN that is based on ephemeral secret keys included on a per-bundle basis, i.e., in a similar manner as for S/MIME. That is, the symmetric keys used to secure DTN bundle traffic should normally be single-use (ephemeral) keys carried in individual bundles rather than persistent session keys. DTN nodes use public/private key pairs to encrypt/decrypt or sign/verify the ephemeral keys. The ephemeral keys are used to decrypt/authenticate bundle data efficiently.

In the design, DTN node public keys are registered with a Key Management System (KMS) that serves as the trust anchor for all secured DTN transactions. The KMS is organized as a group of N Key Authority (KA) nodes that act in an inter-dependent fashion to distribute public keys to all DTN nodes.

Each DTN node generates its own public/private key pair and registers the public key with the KMS. The KMS in turn issues key assertions and revocations in periodic bulletins sent via multicast transmissions to all DTN nodes. The keys are designated for use at some time in the future, since delays/disruptions may preclude immediate delivery.

Each KA node in the KMS has all current public key information for the DTN, but for each bulletin publication it sends only a subset of blocks (or "fragments") of the entire bulletin. Each bulletin is erasure-coded for Forward Error Correction (FEC) in case some fragments are lost, corrupted, or deemed untrustworthy. The resulting parity blocks for error detection are also included in the publication. Receivers then reassemble the bulletin from the union of fragments and parity blocks received, i.e., even if some fragments are lost, and extract time-tagged public keys from the bulletin.

In subsequent operation, the public key that a node uses to encrypt or sign an outbound bundle will be selected based on bundle creation time. The node must ensure that when it creates a bundle it is using a key that other nodes have been informed of. This means that each DTN node must cache keys for sufficiently long times to account for delays in the path.

DTN nodes must therefore keep track of all recently-received public keys for each potential peer node in the DTN. A DTN node that receives a bundle then uses the newest key that is no younger than the bundle creation time to verify or decrypt the ephemeral key included with the bundle.

Since multiple keys are retained at each node with different creation times, there is no need to synchronize key transmission and reception; the receiving node has the appropriate key in place long before the bundle arrives.

Additionally, no information in the key distribution system is kept secret - it's all public information. The point of the KMS is to provide a critical infrastructure trust basis so that DTN nodes can tell whether a prospective correspondent is authorized to use the public key it claims.

Security is then based on the DTN node's trust relationship with the KMS. As a result, all public keys are distributed securely. The KMS service is automated, with potential human intervention for revocation. No multi-message exchanges over long-delay links are needed (i.e., as for services such as the Internet Key Exchange (IKE) protocol), since ephemeral keys are used instead of session keys. The system also provides no single point of failure or compromise.

6. Limitations and Challenges

The candidate KMS design requires a scalable, reliable multicast capability. The DTN Bundle Protocol (BP) reliably delivers bundles to one or more recipients based on convergence layer protocols such as TCP and LTP. Reliable delivery in the BP is "hop-by-hop", where each hop needs to receive data reliably from the previous hop to ensure that end-to-end delivery is reliable. Scalable reliable multicast delivery is also based on hop-by-hop convergence layers, but large-scale reliable multicast is an end-to-end consideration that is not dealt with well in the Internet and needs to be better understood in the DTN context.

Security of the KMS is a fundamental requirement for service integrity. Just as for core Internet services (e.g., the DNS, DHCP, etc.), the KMS must be protected against network-based and physical security attacks. The system design is resilient to one or more elements being compromised, but bringing down all nodes essentially brings down the DTN. History has proven that services of this nature in the public Internet can be protected against comprehensive destruction, but measures must be taken to ensure network and physical security.

Another measure that may be considered in this context is KMS confederation. The KAs of a "local" KMS might forward bulletins to the KAs of another KMS as well as to the local node populations they serve. Such a structure would tend to make the KMS not only more durable but also more scalable.

Nodes that (re)enter the DTN after a long time away can present a challenging bootstrapping situation. Sometimes DTN nodes can go offline for extended periods of time (days/weeks/months), which would essentially bring the same consideration as for a new DTN node entering service for the first time. Upon (re)entering the DTN, the node has to publish its public key via the KMS. This "first contact" trust establishment is crucial to the security of the entire system, i.e., ,there needs to be a way for the new DTN node to trust the KMS, and for the KMS to validate the identity of the DTN node. In effect, a trusted entity (a node or a human) must somehow "vouch" for the new node.

DTN KMS services in fixed networks are not a problem, since the DTN topology does not change. On the other hand, Mobile Ad-hoc Networks (MANETs) typically show up in Unmanned Aerial Vehicle (UAV) networks, tactical military networks, etc. In that case, portions of the DTN may become detached from the rest of the DTN and re-attach at a different point of the DTN at a later time. This is more of a routing issue than a KMS issue, but routing aspects (especially in MANETs where there is no critical infrastructure) need to be understood.

Scaling considerations in terms of the size of the public key database must be analyzed on a per-DTN basis. For example, it may not be necessary for all DTN nodes to receive the public keys of all other DTN nodes since only a subset of all public keys may ever be needed. This is the same scaling consideration that motivated the design of the public Internet Domain Name System (DNS), when maintenance and distribution of a single, central repository at the SRI Network Information Center (SRI-NIC) became too unwieldy to maintain as the Internet grew exponentially.

7. IANA Considerations

There are no IANA considerations for this document.

8. Security Considerations

This document is entirely about security aspects of key management as a crucial component of DTN security; hence, security considerations appear throughout the document.

DTN security considerations are discussed in [RFC6257][I-D.irtf-dtnrg-sbsp].

9. Acknowledgments

Security key management has been discussed broadly in DTN mailing list discussions as well as in many of the documents cited in this publication. The candidate design discussed here is based on the original ideas of Scott Burleigh from NASA/JPL. Kapaleeswaran Viswanathan provided valuable review input.

10. References

10.1. Normative References

[RFC0791] Postel, J., "Internet Protocol", STD 5, RFC 791, September 1981.

[RFC2460] Deering, S. and R. Hinden, "Internet Protocol, Version 6 (IPv6) Specification", RFC 2460, December 1998.

[RFC4838] Cerf, V., Burleigh, S., Hooke, A., Torgerson, L., Durst, R., Scott, K., Fall, K., and H. Weiss, "Delay-Tolerant Networking Architecture", RFC 4838, April 2007.

[RFC5050] Scott, K. and S. Burleigh, "Bundle Protocol Specification", RFC 5050, November 2007.

10.2. Informative References

[I-D.farrell-dtnrg-km] Farrell, S., "DTN Key Management Requirements", draft-farrell-dtnrg-km-00 (work in progress), June 2007.

[I-D.irtf-dtnrg-sbsp] Birrane, E., "Streamlined Bundle Security Protocol Specification", draft-irtf-dtnrg-sbsp-01 (work in progress), May 2014.

[I-D.irtf-dtnrg-sec-overview] Farrell, S., Symington, S., Weiss, H., and P. Lovell, "Delay-Tolerant Networking Security Overview", draft-irtf-dtnrg-sec-overview-06 (work in progress), March 2009.

[IVAN09] Ivancic, W., "Security Analysis of DTN Architecture and Bundle Protocol Specification for Space-Based Networks", October 2009.

[RFC4210] Adams, C., Farrell, S., Kause, T., and T. Mononen, "Internet X.509 Public Key Infrastructure Certificate Management Protocol (CMP)", RFC 4210, September 2005.

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

[RFC5996] Kaufman, C., Hoffman, P., Nir, Y., and P. Eronen, "Internet Key Exchange Protocol Version 2 (IKEv2)", RFC 5996, September 2010.

[RFC6257] Symington, S., Farrell, S., Weiss, H., and P. Lovell, "Bundle Security Protocol Specification", RFC 6257, May 2011.

[WOOD08] Wood, L., Eddy, W., and P. Holliday, "A Bundle of Problems", December 2008.

Authors' Addresses

Fred L. Templin (editor)
Boeing Research & Technology
P.O. Box 3707
Seattle, WA 98124
USA

Email: fltemplin@acm.org

Scott Burleigh
JPL, Calif. Inst. Of Technology
4800 Oak Grove Dr.
Pasadena, CA 91109-8099
USA

Phone: +1 818 393 3353
Email: Scott.Burleigh@jpl.nasa.gov