Network Working Group

Scott Bradner
Harvard University
Allison Mankin
USC/ISI
Jeffrey I. Schiller
Massachusetts Institute of Technology
June 2003

A Framework for Purpose-Built Keys (PBK)

<draft-bradner-pbk-frame-06.txt>

Status of this Memo

This document is an Internet-Draft and is subject to the provisions of Section 10 of RFC2026.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF), its areas, and its working groups. Note that other groups may also distribute working documents as Internet-Drafts.

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."

The list of current Internet-Drafts can be accessed at http://www.ietf.org/ietf/1id-abstracts.txt

The list of Internet-Draft Shadow Directories can be accessed at http://www.ietf.org/shadow.html.

Copyright Notice

Copyright (C) The Internet Society (2003). All Rights Reserved.

Abstract

This memo considers the need to authenticate the source of a network communication where the actual identity of the source is not important but it is important and that successive messages in the communication come from the same source. This memo defines the use of specially generated public/private key pairs, known as Purpose-Built Keys (PBKs), to provide this assurance. This memo is not a full specification of a PBK protocol, but rather a model or framework

for development of PBK in applications.

1.0 Introduction

There are many cases in Internet protocols where cryptographic mechanisms can add significant security improvement. However most such mechanisms rely on associating keys to entities, ultimately requiring an enterprise-wide, multi-enterprise, or even more widely deployed Public Key Infrastructure (PKI).

In the absence of security mechanisms, many protocols are continuously vulnerable to attack. However, there are many circumstances where we can improve overall security by narrowing the window of vulnerability, so that if we assume that some operation is performed securely, we can secure all future transactions.

There are also cases where the actual identity of the initiator of a network communication is not an important piece of information, yet it is important to know that successive packets are from that same source. One example of this is in mobile IPv6. Mobile IPv6 contains a rebinding option that enables a mobile node to tell the other end of a communication that the IP address for the mobile node has changed. It is clearly important to know that any such rebinding request actually came from the correct mobile node even if the identity of the user of that mobile node does not need to be known.

Note that it is not that the identity of the user here is unimportant to the network (the node user may well authenticate to an Authentication, Authorization and Accounting (AAA) service or other access manager at the start of network activity), but rather that it is unimportant to accomplish that level of authentication for the purpose of rebinding.

This memo describes the use of a temporary public/private key pair that is generated by a host for each case where the consistency of authentication needs to be assured. For example, if mobile IP binding were to use this technique, then a new key pair would be generated before each mobile ip session in which the mobile was roaming, and discarded after the session was completed.

This use of host-generated temporary keys is confined to the parties in a communication and does not require that the keys be registered

with or known by any third party. Thus this mechanism does not require that any support infrastructure exist outside of the protocol support in the corresponding hosts, and it can be deployed incrementally as host support becomes available. It also scales well since the operations are confined to the end systems involved in the

communication.

By not using registered keys, the PBK mechanism preserves user pseudonymity as long as the identities of the users are not disclosed by some other process during the communication. There is extensive literature about the lack of anonymity of persons stemming from their IP addresses ([Syverson] is a good starting point) as well as work that has kinship to the pseudonyms in this work [Brands], [Chaum88], [HIP], [SUCV].

The PBK mechanism is susceptible to man-in-the-middle attacks which affect its initialization. Such attacks may make it possible for a pseudonymous identity to be used by a party other than the party that generated the public/private key pair and then sent it to the recipient. There is an "initial leap of faith" about the pseudonymous identity since it has no parties, other than the party that generated the public/private key pair, vouching for it, and though only the party that generated the public/private key pair holds the private key, a man-in-the-middle attacker may appear to hold and use the identity without good care being taken in a protocol design that makes use of PBK. Therefore, the designer of such a protocol should be aware of this risk and include a challenge-response confirmation step. The challenge-response step should have the property of needing the private key for decryption and include a nonce.

The PBK mechanism is intended to used with transport or application protocols. It differs from IPSec in that it is applied on demand by an application or by a transport protocol.

2.0 Conceptual Overview

Following is a conceptual step-by-step description of the PBK process when operating below the transport layer.

First some definitions:

Initiating Node: the node initiating the conversation
Receiving Node: the node at the other end of the conversation

Before an Initiating Node initiates a connection during which it will need to prove that it is the same node that started the connection, it creates a public/private key pair for use during the connection. This is known as a purpose-built key (PBK) pair.

The Initiating Node then creates a Purpose-Built ID (PBID) by performing a cryptographic hash of the public part of the PBK pair. This PBID will be used as an identity token for the node.

The Initiating Node then initiates the connection. The PBID is sent along with the initial packets in the connection. In IPv6 this could be done in an end-to-end option header, in IPv4 as a header option. (These option ideas are for transport level use of the PBK - if the PBK was used from within HTTP or another application, the PBID's location would be in the application protocol.) The PBID does not need to appear in all of the packets; it just has to be reliably conveyed to the Receiving Node. Reliability may be obtained by carrying it on enough packets so that a return packet indicates it was received eventually. This is the simplest approach; depending on requirements and the application, the PBID may well be sent using a reliable transport protocol. Retransmission algorithms, where they are needed, must be conformant with RFC 2914 [RFC2914].

The Receiving Node stores the PBID and the source IP address from the received packet in a table.

At some time in the connection before the proof of identity is needed, the Initiating Node sends its public key to the Receiving Node. This again could be done in IP-level options or in an application-level exchange. The public key could be sent as part of the initialization or anytime before it is needed for proof of identity. The Receiving Node verifies that the received public key hashes to the previously provided PBID.

When the Initiating Node wants to perform some operation for which it wants to prove its identity, it sends the PBID along with the operation request. The message is signed using the private part of the PBK. If replay protection is necessary, a nonce value (a

monotonically increasing value) or timestamp may be included with the operation request.

When the Receiving Node gets such an operation request it verifies the digital signature using the saved public key and returns a challenge packet. The challenge packet is sent to the IP address that was in the source IP address field of the packet that contained the request. The challenge packet contains a random number test value generated by the Receiving Node.

When the Initiating Node receives the challenge packet it encrypts the test value in its private key and sends the result back to Receiving Node.

When the Receiving Node gets the challenge response it decrypts the test value using the stored public key associated with the PBID. If the results match then the Receiving Node can be sure that the node that sent the operation request was the correct Initiating Node.

The PBKs would normally be discarded at the end of the communication but in those cases where a continuity of identity is needed over multiple sessions the PBKs could be retained until the requirement was over.

3.0 Notes on the design

The hash of the public key is used as the PBID so that the relationship between an offered PBID and public key can be established. If a Receiving Node is in possession of the private key and the hash of the corresponding public key matches an offered PBID, it can be sure that it has the correct PBID for that public key.

The challenge / response exchange has to be synchronized within the data stream if the processing of packets after the operation request would be different than before the operation request, as it would be for mobile IPv6. This would mean suspending normal transmission until the challenge / response exchange was completed.

The challenge is sent to the source address in the packet, and this address is not included in the digital signature on the operation request packet so that this mechanism can work through any address-

modifying devices that may be in the path.

In the cases where commands could be issued by both ends of a communication, as would be the case in mobile IPv6 if both ends were mobile, separate PBKs would be created by each end and the mechanism would be run independently by each end.

4.0 Security Considerations

This whole document is about security. Specifically the memo discusses how to perform authenticated operations in an environment where there is no existing security infrastructure or an environment where network addresses might change during the course of the communication.

In the absence of a security infrastructure such as a PKI, it is not always possible to authenticate one party to another. In the absence of any cryptographic security mechanism, internet transactions are continuously at risk of compromise. With PBKs it is possible to leverage an initial "leap of faith" so that presuming an initial transaction has not been tampered with (say the exchange of PBID's at the beginning of an association between two parties), future transactions can be secured.

5.0 Acknowledgements

We owe credit for some of the concepts in this draft to Bob Moskowitz and also (as anyone working in the area of privacy does) to David Chaum.

6.0 Author's Addresses

Scott Bradner
Harvard University
29 Oxford St.
Cambridge MA 02138

Phone +1 617 495 3864
email sob@harvard.edu

Allison Mankin
Bell Labs, Lucent
Phone: +1 301 728 7199
email: mankin@psg.com

Jeffrey I. Schiller
Massachusetts Institute of Technology
MIT Room W92-190
77 Massachusetts Avenue
Cambridge, MA 02139-4307
Phone: +1 617 253 0161
email: jis@mit.edu

Informative References

[RFC2914] Floyd, S., "Congestion Control Principles", RFC 2914, September 2000.

[Syverson] Syverson, P., Goldschlag, D. and Reed, M., "Anonymous Connections and Onion Routing," in 18th Annual Symposium on Security and Privacy, Oakland CA, 1997. http://www.onion-router.net/Publications/SSP-1997.pdf

[Brands] Brands, S.A., "Rethinking Public Key Infrastructures and Digital Certificates - Building In Privacy," MIT Press, 2000.

[Chaum88] Chaum, D., Fiat, A., and Naor, M. "Untraceable Electronic Cash", in S. Goldwasser, Editor, Advances in Cryptology - CRYPTO '88. Lecture Notes in Computer Science Volume 403, Springer-Verlag, 1988.

[HIP] Moskowitz, R., "Host Identity Payload Architecture", "Host Identity Payload Protocol", http://homebase.htt-consult.com/~hip, 2001.

[SUCV] Montenegro, G., Castellucia, C., "SUCV Identifiers and Addresses", IETF Work in Progress, July 2002.

Full Copyright statement

Copyright (C) The Internet Society (2003). All Rights Reserved. This document and translations of it may be copied and furnished to others, and derivative works that comment on or otherwise explain it or assist in its implementation may be prepared, copied, published and distributed, in whole or in part, without restriction of any kind, provided that the above copyright notice and this paragraph are included on all such copies and derivative works. However, this document itself may not be modified in any way, such as by removing the copyright notice or references to the Internet Society or other Internet organizations, except as needed for the purpose of developing Internet standards in which case the procedures for copyrights defined in the Internet Standards process must be followed, or as required to translate it into languages other than English.

The limited permissions granted above are perpetual and will not be revoked by the Internet Society or its successors or assigns.

This document and the information contained herein is provided on An "AS IS" basis and THE INTERNET SOCIETY AND THE INTERNET ENGINEERING TASK FORCE DISCLAIMS ALL WARRANTIES, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO ANY WARRANTY THAT THE USE OF INFORMATION HEREIN WILL NOT INFRINGE ANY RIGHTS OR ANY IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE.