Network Working Group
Internet-Draft
Intended status: Standards Track
Expires: March 14, 2009

A. Kukec
University of Zagreb
S. Krishnan
Ericsson
S. Jiang
Huawei Technologies Co., Ltd

September 10, 2008

SeND Hash Threat Analysis
draft-ietf-csi-hash-threat-00

Status of this Memo

Abstract

This document analysis the use of hashes in SeND, possible threats and the impact of recent attacks on hash functions used by SeND. Current SeND specification [rfc3971] uses SHA-1 [sha-1] hash algorithm and PKIX certificates [rfc3280] and does not provide support for the hash algorithm agility. The purpose of the document is to provide analysis of possible hash threats and to decide how to encode the hash agility support in SeND.

Table of Contents

1. Introduction

SEND [rfc3971] uses the SHA-1 hash algorithm to generate the contents of the Key Hash and the Digital Signature fields of the RSA signature. It also uses a hash algorithm (SHA-1,MD5 etc.) in the PKIX certificates [rfc3280] used for the router authorization in the ADD process. Recently there have been demonstrated attacks against the collision free property of such hash functions [sha1-coll]. There have also been attacks on the PKIX X.509 certificates that use the MD5 hash algorithm [x509-coll] This document analyzes the effects of such attacks and other hash attacks on the SEND protocol and proposes changes to make it resistant to such attacks.

2. Terminology

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in RFC 2119 [rfc2119].

3. Impact of collision attacks on SeND

Due to the hash attacks demonstrated on the aforesaid hash algorithms a study was performed to assess the threat of these attacks on the cryptographic hash usage in internet protocols [RFC4270]. This document analyzes the hash usage in SEND following the approach recommended by [rfc4270]. Following the approach recommended by [rfc4270] and [new-hashes], we will analyze the impact of these attacks on SeND case by case in this section. Through our analysis, whether we should support hash agility on SeND is also discussed.

Up to date, all demonstrated attacks are attacks against a collision-free property. An attacker produces two messages which result in the same hash. Attacks against the one-way property are not yet feasible [rfc4270]. There are two attacks against one property. In the first-preimage attack, based on the hash of the real message, an attacker finds a false message which results in the same hash. In the second-preimage attack an attacker based on the real message itself, finds the false message which results in the same hash.

3.1. Attacks against CGAs in stateless autoconfiguration

Hash functions are used in the stateless autoconfiguration process that is based on CGAs. Impacts of collision attacks on current uses of CGAs are analyzed in the update of CGA specification [rfc4982], which also enables CGAs to support hash agility. CGAs provide proof-of-ownership of the private key corresponding to the public key used to generate CGA, and they don't deal with the non-repudiation feature, while collision attacks are mainly about affecting non-repudiation feature. CGAs are not suspectable to the collision attacks. In the collision attack, an attacker finds two keys (and other CGA parameters) K and K', where CGA = hash(K, ..) = hash(K', ..). Since both keys have to be choosen by an attacker, CGAs are not vulnerable to the collision attack. So, as [rfc4982] points out CGA based protocols, including SeND, are not affected by the recent collision attacks. But, CGAs are vulnerable to the preimage attack in which an attacker could manage to find the false key K' based on node's CGA = hash(K, ..), use key K' to produce the Key Hash field and to sign the SeND message afterwards. In that case, an attacker just has to break the CGA, and all other hashes are automatically broken, because an attacker can use the false key K' to produce all other hashes. But up to date, the preimage attacks are not yet feasible, but might be in the future.

3.2. Attacks against PKIX certificates in ADD process

The second use of hash functions is for the router authorization in ADD process. Router sends to host a certification path, which is a

path between a router and hosts's trust anchor and consists of PKIX certificates. Researchers demonstrated attacks against PKIX certificates with MD5 signature, in 2005 [new-hashes] and in 2007 [X509-COLL].

In 2005 they succeeded to construct the original and the false certificate that had the same identity data and digital signature, but different public keys [new-hashes]. The problem for the attacker is that two certificates with the same identity are not very useful in real-world scenarios, while Certification Authority is not allowed to provide such two certificates. Additionally, since identity field is humane readable data, certificates are not affected by collision attacks in practice. Implementations SHOULD use human-readable certificate extensions only if SeND certificate profile demands. We also have to take into account that attacker could produce such false certificate only if he could predict context- useful certificate data. So, although collision attacks against PKIX certificates are theoretically possible, they can hardly be performed in practice.

In 2007 were demonstrated certificates with the same identity data and signatures, which differed only in public keys. Such attacks are potentially more dangerous, since attacker can decide about contents of human readable fields, and produce for example certificates with the same signatures, but different identities or validity periods. However, in order to perform a real-world useful attack, attacker still needs to predict the content of all fields appearing before the public key, eg. serial number or validity periods. Although a relying party cannot verify the content of these fields (each certificate by itself is unsuspicious), the CA keeps track of those fields and during the fraud analysis, the false certificate can be revealed.

The certificate key in SeND is used both for the CGA generation and for message signing. In the future, CGA might not be used at all in SeND, just certificates. Thus, attacks against certificates are potentially very dangerous. Generally, the most dangerous are attacks against middle-certificates in the certification path, where for the cost of one false certificate, attacker launches attack on multiple routers. In such scenarios, we will be at least safe from attacks against Trust Anchor's certificate because it is not exchanged in SeND messages.

3.3. Attacks against Digital Signature in RSA Signature option

The digital signature in RSA Signature option is produced as the SHA-1 hash of IPv6 addresses, ICMPv6 header, ND message and other fields like Message Type Tag and ND options [rfc3971], and is signed with the sender's private key, which corresponds to the public key

from the CGA parameters structure and is authorized usually through CGAs. The possible attack on such explicit digital signature is typical non-repudiation attack. The Digital Signature field is vulnerable to the collision attack. In such collision attack an attacker produces two messages M and M', where hash(M) = hash(M'), underlays one of the messages to be signed with authorized keys (through CGAs), but uses another message afterwards. However, the prediction and production of the useful content of messages M and M' is just theoretically possible, since SeND message contains mostly human readable data. Additionally, the structure of at least one of two messages (M and M') is predefined [rfc4270]. But we have to take into account that a variant of SHA-1 was already affected by recent collision attacks and we have to prepare for future improved attacks.

3.4. Attacks against Key Hash in RSA Signature option

Key Hash field in the RSA Signature option is a SHA-1 hash of the public key from the CGA parameters structure in the CGA option of SeND message. Key Hash field is potentially dangerous because it contains a non human-readable data. Since in the collision attack an attacker itself chooses both keys, K and K', where hash(K) = hash(K'), the Key Hash field is not suspectable to the collision attack. The preimage attack in which an attacker derives the key K' based on hash(K) could be theoretically more useful. But even in that case, if an attacker signes the SeND message with the key K', he has to break also the CGA, since the Digital Signature is verified against the CGA and possibly against a certification path.

4. Support for the hash agility in SeND

While all of analyzed hash functions in SeND are theoretically affected by recent hash attacks, these attacks indicate the possibility of future real-world attacks. In order to increase the future security of SeND, we suggest the support for the hash and algorithm agility in SeND.

The most effective and secure would be to bind the hash function option with something that can not be changed at all, like [rfc4982] does for CGA - encoding the hash function information into addresses. But, there is no possibilty to do that in SeND. We could decide to use by default the same hash function in SeND as in CGA. The security of all hashes in SeND depends on CGA, ie. if an attacker could break CGA, all other hashes are automatically broken. From the security point of view, at the moment, this solution is more reasonable then defining different hash algorithm for each hash. Additionally, if using the hash algorithm from the CGA, no bidding down attacks are possible.

Another solution is to incorporate the Hash algorithm option into SeND message, and use different hash algorithms for different hashes, or the same algorithm for all hashes. As already mentioned, from the security point of view, it is reasonable to define just one algorithm, because additional algorithms does not increase the security. If that algorithm is defined in the Hash algorithm option in SeND message, it is vulnerable to the bidding down attack. On the other hand, different algorithms provides additional flexibility, and in the future SeND might be used completely without CGAs.

5. Security Considerations

This document analyzes the impact of hash attacks in SeND and offeres a higher security level for SeND by providing solution for the hash agility support.

6. IANA Considerations

This document defines three new registries that have been created and are maintained by IANA.

- Hash Algorithm for Key Hash field (HA-KH). The values in this name space are 5-bit unsigned integers.
- Hash Algorithm for Digital Signature field (HA-DS). The values in this name space are 5-bit unsigned integers.
- Signature Algorithm (SA). The values in this name space are 5-bit unsigned integers.

Initial values for these registries are given below. Future assignments are to be made through Standards Action [rfc2434]. Assignments for each registry consist of a name, a value and a RFC number where the registry is defined.

The following initial values are assigned for HA-KH in this document:

Name	Value	RFCs
SHA-1	TBD1	this document

The following initial values are assigned for HA-DS in this document:

Name	Value	RFCs
SHA-1	TBD2	this document

The following initial values are assigned for HA-KH in this document:

Name	Value	RFCs
RSASSA-PKCS1-v1_5	TBD3	this document

This document defines one new Neighbor Discovery Protocol [rfc4861] options, which must be assigned Option Type values within the option numbering space for Neighbor Discovery Protocol messages:

The Hash algorithm option (TBA1), described in Section 4.1.

7. References

7.1. Normative References

[new-hashes] Bellovin, S. and E. Rescorla, "Deploying a New Hash Algorithm", November 2005.

[rfc3971] Arkko, J., Kempf, J., Zill, B., and P. Nikander, "SEcure Neighbor Discovery (SEND)", RFC 3971, March 2005.

[rfc4270] Hoffman, P. and B. Schneier, "Attacks on Cryptographic Hashed in Internet Protocols", RFC 4270, November 2005.

[rfc4982] Bagnulo, M. and J. Arrko, "Support for Multiple Hash Algorithms in Cryptographically Generated Addresses (CGAs)", RFC 4982, July 2007.

7.2. Informative References

[rfc2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", RFC 2119, March 1997.

[rfc2434] Narten, T. and H. Alvestrand, "Guidelines for Writing an IANA Considerations Section in RFCs", RFC 2434, April 1998.

[rfc3280] Housley, R., "Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile", RFC rfc3280, April 2002.

[rfc4861] Narten, T., Nordmark, E., Simpson, W., and H. Solliman, "Neighbor Discovery for IP Version 6 (IPv6)", RFC 4861, April 1998.

[sha-1] NIST, FIBS PUB 180-1, "Secure Hash Standard", April 1995.

[sha1-coll] Wang, X., Yin, L., and H. Yu, "Finding Collisions in the Full SHA-1. CRYPTO 2005: 17-36", 2005.

[x509-coll] Stevens, M., Lenstra, A., and B. Weger, "Chosen-Prefix Collisions for MD5 and Colliding X.509 Certificates for Different Identitites. EUROCRYPT 2007: 1-22", 2005.

Authors' Addresses

Ana Kukec
University of Zagreb
Unska 3
Zagreb
Croatia

Email: ana.kukec@fer.hr

Suresh Krishnan
Ericsson
8400 Decarie Blvd.
Town of Mount Royal, QC
Canada

Email: suresh.krishnan@ericsson.com

Sheng Jiang
Huawei Technologies Co., Ltd
KuiKe Building, No.9 Xinxi Rd.,
Shang-Di Information Industry Base, Hai-Dian District, Beijing
P.R. China

Email: shengjiang@huawei.com

Full Copyright Statement

Intellectual Property