

Network Working Group A.
Kukec
Internet-Draft University of
Zagreb
Expires: January 2, 2009 S.
Krishnan

Ericsson S.

Jiang
Ltd Huawei Technologies Co.,

2008 July 1,

**SeND Hash Threat Analysis
draft-kukec-csi-hash-threat-02**

Status of this Memo

By submitting this Internet-Draft, each author represents that any applicable patent or other IPR claims of which he or she is aware have been or will be disclosed, and any of which he or she becomes aware will be disclosed, in accordance with [Section 6 of BCP 79](#).

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>.

This Internet-Draft will expire on January 2, 2009.

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. Based on previous analysis, this document suggests multiple hash support that should be included in the SeND update specification.

Table of Contents

1	Introduction	
3		
2	Terminology	
4		
3	Impact of collision attacks on SeND	
5		
3.1	Attacks against CGAs in stateless autoconfiguration	
5		
3.2	Attacks against PKIX certificates in ADD process	
5		
3.3	Attacks against Digital Signature in RSA Signature option	
6		
3.4	Attacks against Key Hash in RSA Signature option	
7		
4	Support for the hash agility in SeND	
8		
4.1	Hash algorithm option	
8		
5	Security Considerations	
11		
6	IANA Considerations	
12		
7	References	
13		
7.1	Normative References	
13		
7.2	Informative References	
13		
14	Authors' Addresses	
14		
15	Intellectual Property and Copyright Statements	
15		

Kukec, et al.
2]

Expires January 2, 2009

[Page

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 router authorization. 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](#)] that allow an attacker to generate two different certificates with different distinguished names and different public keys that contain identical signatures. This document analyzes the effects of such 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 collision 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. Attacks against the one-way property are not yet feasible [[rfc4270](#)].

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. While SeND is CGA based protocol, we are sure that the node that signs the message is the same as the node that creates the message and associated hash. So, as [[rfc4982](#)] points out

CGA based protocols, including SeND, are not affected by the recent collision attacks.

3.2. Attacks against PKIX certificates in ADD process

The second use of hash functions is for 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

Kukec, et al.
5]

Expires January 2, 2009

[Page

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 unsuspecting), the CA keeps track of those fields and during the fraud analysis, the false certificate can be revealed.

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. Additionally, if attacker, for example, manages to produce a false certificate with changed IP prefixes in IP subjectAltName extension (which is currently just theoretically possible), IP prefixes range will be broadened at maximum to the range from the Trust Anchor's certificate.

3.3. Attacks against Digital Signature in RSA Signature option

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. Attacker could generate a false message with the same hash and sign that false hashed message with authorized private key. However, the problem for the attacker is that they are very hard to predict the useful input data. It minimizes the possibility for a real-world collision attack and the fact that in order to perform a successful real-world attack he can not change a human-readable data. But we also 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.

Kukey, et al.
6]

Expires January 2, 2009

[Page

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 a typical example of data fingerprinting, which is potentially dangerous because input field is

a non human-readable data. But the problem for the attacker is that a public key, which is input data is authorized through CGAs, and sometimes additionally through a certification path if peer has configured trust anchor. For the succesful attack, attacker has to break both SHA-1 hashed public key, as well as corresponding CGA and possibly a certification path. Otherwise, changed key pair will be detected in the process of CGA verification. The same as in previous

cases, this attack is theoretically possible, but very hard to perform in practice.

4. Support for the hash agility in SeND

While all of analyzed hash functions in SeND are theoretically affected by recent collision 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 possibility to do that in SeND. We could decide to use by default the same hash function in SeND as in CGA, but this solution is architecturally strange and it does not really increase the security since the difficulty for attackers remain to break one single hash function. Furthermore, it may even reduce the security level by providing more relevant information of the hash function. On the other side, the use of two different hash algorithms makes attacker's life harder.

Another solution is to incorporate the hash function option into SeND

message. By putting a new hash function option in SeND message before RSA Signature option, attacker will have to break both the signature and the hash input at the same time since the new option will be input field for the Digital Signature in RSA Signature option. However, we can not avoid a downgrade attack totally

because

peer might be using just ND and not SeND. A completely safe

solution

here does not exist. A new hash function option in SeND message is

a

reasonable and the best solution for the hash algorithm agility support in SeND.

Each implementation SHOULD use different hash or signature algorithms

for each of the relevant fields (Key Hash field, Digital Signature, PKIX signature algorithm). Since all algorithms are in different procedures, making them the same does not make those procedures simpler, but making them different complicates possible attacks.

4.1. Hash algorithm option

In order to provide hash algorithm agility, each SeND implementation MUST support the Hash algorithm option. The Hash algorithm option defines:

- o a hash algorithm that MUST be used for producing the RSA Signature option (Key Hash field) - HA-KH,

o a hash algorithm that MUST be used for producing the RSA
Signature
option (Digital Signature field) - HA-DS,

Kukey, et al.
8]

Expires January 2, 2009

[Page

- o a PKIX signature algorithm that the sender of the Hash algorithm option is ready to accept and validate. In order to enhance interoperability, implementations SHOULD also accept and validate PKIX certificates with a signature algorithm that has the higher encoding number than requested signature algorithm. Implementations MUST NOT accept PKIX certificates with signature algorithms marked with lower encoding.

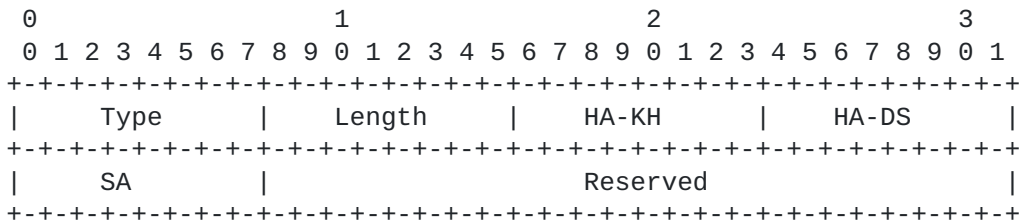


Figure 1

Type

TBA1

Length

The length of the option (including the Type, Length, Reserved, HA, SA, and Padding) in units of 8 octets.

Hash Algorithm for Key Hash field (HA-KH)

Hash algorithm used for producing the Key Hash field in the RSA Signature option. SEND [rfc3971] uses SHA-a. In order to provide hash algorithm agility, SHA-1 is assigned an initial value TBD1 in this document.

Hash Algorithm for Digital Signature field (HA-DS)

Hash algorithm used for producing the Digital Signature field in the RSA Signature option. SEND [rfc3971] uses SHA-1. In order to provide hash algorithm agility, SHA-1 is assigned an initial value TBD2 in this document.

Signature Algorithm (SA)

Signature algorithm of the PKIX certificate used in ADD process, in accordance with rfc3279. SEND [rfc3971] uses RSASSA-PKCS1-

v1_5

Kukec, et al.
9]

Expires January 2, 2009

[Page

algorithm. In order to provide algorithm agility, RSASSA_PKCS1-v1_5 is assigned an initial value TBD3 in this document.

Reserved

Field reserved for future uses. The value MUST be initialized to zero by the sender, and MUST be ignored by the recipient.

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.

- o Hash Algorithm for Key Hash field (HA-KH). The values in this name space are 5-bit unsigned integers.
- o Hash Algorithm for Digital Signature field (HA-DS). The values in this name space are 5-bit unsigned integers.
- o 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 Xixi Rd.,
Shang-Di Information Industry Base, Hai-Dian District, Beijing
P.R. China

Email: shengjiang@huawei.com

Full Copyright Statement

Copyright (C) The IETF Trust (2008).

This document is subject to the rights, licenses and restrictions contained in [BCP 78](#), and except as set forth therein, the authors retain all their rights.

This document and the information contained herein are provided on an

"AS IS" basis and THE CONTRIBUTOR, THE ORGANIZATION HE/SHE REPRESENTS

OR IS SPONSORED BY (IF ANY), THE INTERNET SOCIETY, THE IETF TRUST AND

THE INTERNET ENGINEERING TASK FORCE DISCLAIM ALL WARRANTIES, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO ANY WARRANTY THAT THE USE OF

THE INFORMATION HEREIN WILL NOT INFRINGE ANY RIGHTS OR ANY IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE.

Intellectual Property

The IETF takes no position regarding the validity or scope of any Intellectual Property Rights or other rights that might be claimed to

pertain to the implementation or use of the technology described in this document or the extent to which any license under such rights might or might not be available; nor does it represent that it has made any independent effort to identify any such rights.

Information

on the procedures with respect to rights in RFC documents can be found in [BCP 78](#) and [BCP 79](#).

Copies of IPR disclosures made to the IETF Secretariat and any assurances of licenses to be made available, or the result of an attempt made to obtain a general license or permission for the use

of

such proprietary rights by implementers or users of this specification can be obtained from the IETF on-line IPR repository

at

<http://www.ietf.org/ipr>.

The IETF invites any interested party to bring to its attention any copyrights, patents or patent applications, or other proprietary rights that may cover technology that may be required to implement this standard. Please address the information to the IETF at ietf-ipr@ietf.org.

