

Individual Submission
Internet-Draft
Intended status: BCP
Expires: January 27, 2011

G. Huston
G. Michaelson
APNIC
S. Kent
BBN
July 26, 2010

CA Key Rollover in the RPKI
draft-huston-sidr-keyroll-00.txt

Abstract

This document describes an algorithm to allow an entity who undertakes the role of a Certification Authority in the Resource Public Key Infrastructure to perform a rollover of its key pair. This document also notes the requirements placed on Relying Parties who maintain a local cache of the objects that have been published in the distributed Resource Public Key Infrastructure repository publication structure.

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 January 27, 2011.

Copyright Notice

Copyright (c) 2010 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to [BCP 78](#) and the IETF Trust's Legal Provisions Relating to IETF Documents (<http://trustee.ietf.org/license-info>) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect

to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

Table of Contents

[1.](#) Introduction [3](#)
 [1.1.](#) Terminology and Concepts [3](#)
[2.](#) CA Key Rollover Procedure [3](#)
[3.](#) Relying Party Requirements [6](#)
[4.](#) Security Considerations [7](#)
[5.](#) IANA Considerations [7](#)
[6.](#) Acknowledgements [7](#)
[7.](#) References [7](#)
 [7.1.](#) Normative References [7](#)
 [7.2.](#) Informative References [7](#)
Authors' Addresses [7](#)

Internet-Draft

Key Rollover

July 2010

1. Introduction

This document describes an algorithm to allow an entity undertaking the role of a Certification Authority (CA) in the Resource Public Key Infrastructure (RPKI) [[ID.ietf-sidr-arch](#)] to perform a rollover of its key pair.

The intent of this document is to define a conservative procedure for such entities to follow when performing a key rollover so that Relying Parties are in a position to be able to validate all authentic objects in the RPKI using the validation procedure described in [[ID.ietf-sidr-res-certs](#)] at all times.

1.1. Terminology and Concepts

It is assumed that the reader is familiar with the terms and concepts described in "Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile" [[RFC5280](#)], "X.509 Extensions for IP Addresses and AS Identifiers" [[RFC3779](#)], and the profile for RPKI Certificates [[ID.ietf-sidr-res-certs](#)] .

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](#).

2. CA Key Rollover Procedure

A CA instance is associated with a single key pair ([ID.ietf-sidr-res-certs](#)). The implication in the context of key rollover is that, strictly speaking, a CA does not perform a key rollover per se. In order to perform the equivalent of a key rollover, the entity who undertakes the role of a CA needs to instantiate a new instance of a CA, with the new key pair, and then substitute this new CA into the RPKI hierarchy in place of the old CA.

There are some considerations regarding this procedure that should be followed by an entity performing a key rollover operation. The critical consideration is that the RPKI has potential application in the area of control of routing integrity [[ID.ietf-sidr-arch](#)], and key rollover should not cause any transient hiatus where a Relying Party is led to incorrect conclusions regarding the authenticity of attestations and authorities made in the context of the RPKI. A CA should not assume that Relying Parties will universally use one form of construction of a potential validation path over any other, and therefore the key rollover procedure should endeavour at all times to preserve the integrity of the SIA and AIA pointers in RPKI

certificates.

In the procedure described here, the entity creates a "new" CA instance, and has the associated new public key published in the form of a "new" CA certificate. While the "current" and "new" CA instances share a single repository publication point, each CA has its own CRL, and its own manifest. Initially, the "new" CA publishes an empty CRL and a manifest that contains a single entry for the CRL. The "current" CA also is

The entity should then wait for a period of time to allow Relying Parties to discover and retrieve this "new" CA certificate and store it in their local RPKI Repository cache instances (this period of time is termed the "staging period"). During this period, the entity will have a "new" CA instance, with no subordinate products, and an "current" CA instance which has issued all subordinate products. At the expiration of the staging period the "new" CA instance can re-issue all subordinate products of the previous CA instance, overwriting the old subordinate products in the CA's repository publication point. When this is complete the "current" CA instance can be retired, and the "new" CA instance can be re-termed the "current" CA.

During the transition of the CA instances it is necessary for the "new" CA instance to re-issue all subordinate products of the "current" CA. The procedure described here specifies that, with the exception of manifests and CRLs, the re-issued subordinate products be published using the same repository publication point object names, effectively overwriting the old subordinate objects with these

re-issued subordinate objects. The intent of this overwriting operation is to ensure that the AIA pointers of indirect subordinate products at lower levels in the PKI hierarchy remain correct, and that CA rollover does not require any associated actions by any subordinate CA.

There are four CA states described here:

CURRENT:

The CA is the active CA used to process certificate issuance and revocation requests from subordinate entities.

NEW:

The CA is in the process of being created. The CA is unable to process certificate issuance and revocation requests from subordinate entities. The CA may issue a CRL and an EE certificate in association with its Manifest, but has no other subordinate products.

PENDING:

The CA is in the process of being set up. The CA is able to able to issue certificates that were previously issued with the old key, but is not able to process new certificate issuance and revocation requests from subordinate entities.

OLD:

The CA is in the process of being removed. The CA is able to unable to process any certificate issuance and revocation requests from subordinate entities. The CA will continue to issue regularly scheduled CRLs and be permitted to issue an EE certificate as part of the process of updating its manifest to reflect the updated CRL.

To perform a key rollover operation the entity **MUST** perform the following steps in the order given here. Unless specified otherwise each step **SHOULD** be performed without any intervening delay. The process **MUST** be run through to completion.

1. Generate a NEW key pair.

2. Generate a certificate request with the NEW key pair and pass the request to the entity's immediate superior CA as the CA certificate Issuer.
3. Request the entity's Issuer to generate and publish a NEW CA certificate, with an issuer-selected Subject Name that is distinct from the Subject Name used in the CURRENT CA certificate for this entity.
4. Wait for a "Staging Period" following the completion of the NEW CA certificate request. This "Staging Period" is selected by the entity, and MUST be no less than 24 hours.
5. Upon expiration of the Staging Period, suspend the processing of subordinate certificate issuance requests and revocation requests. Mark the CURRENT CA as OLD and the NEW CA as PENDING. Halt the operation of the OLD CA for all operations except the further issuance of subsequent CRLs and EE certificates for Manifests.
6. Use the PENDING CA to generate new certificates for all existing subordinate CA and EE certificates, and publish those products in the same repository publication point and with the same repository publication point name as the previous OLD subordinate CA and EE certificates. The keys in these reissued certificates must not change.

7. Excluding manifests, where the signing structure uses a packaging format that includes the EE certificate within the signed data, signed objects that included OLD CA-issued EE certificates in their signed data will need to be re-signed using an EE certificate issued by the PENDING CA. In the case where the OLD CA-issued EE certificate is a "single use" EE certificate and the associated private key has been previously destroyed, this will entail the generation of a new key pair, the issuing of an EE certificate by the PENDING CA, and the signing of the data by the newly generated private key. In the case of a "multi-use" EE certificate, the EE certificate should be issued using the PENDING CA. The object, together with the issued EE certificate, should be signed with the associated private key, and published in the same repository publication point, using the same

repository publication point name, as the previously signed object that it replaces (i.e. overwrite the old signed object).

8. Use the OLD CA to issue a manifest that lists only the OLD CA's CRL, and use the PENDING CA to issue a manifest that lists all subordinate products that were issued by the PENDING CA.
9. Mark the PENDING CA as CURRENT and resume processing subordinate certificate issuance requests.
10. Generate a certificate revocation request for the OLD CA certificate and pass it to the entity's Issuer.
11. Wait for completion of the OLD CA certificate revocation request, then remove the OLD CA's CRL and Manifest and destroy the OLD private key.

3. Relying Party Requirements

This procedure defines a "Staging Period" for CAs performing a key rollover operation, which is defined as a period no shorter than 24 hours.

Relying Parties who maintain a local cache of the distributed RPKI repository MUST perform a local cache synchronisation operation against the distributed RPKI repository at regular intervals of no longer than 24 hours.

4. Security Considerations

[To be added]

5. IANA Considerations

[Note to IANA, to be removed prior to publication: there are no IANA

considerations stated in this document.]

[6.](#) Acknowledgements

The authors would like to acknowledge the review comments of Tim Bruijnzeels in preparing this document.

[7.](#) References

[7.1.](#) Normative References

- [RFC3779] Lynn, C., Kent, S., and K. Seo, "X.509 Extensions for IP Addresses and AS Identifiers", [RFC 3779](#), June 2004.
- [RFC5280] Cooper, D., Santesson, S., Farrell, S., Boeyen, S., Housley, R., and W. Polk, "Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile", [RFC 5280](#), May 2008.

[7.2.](#) Informative References

- [ID.ietf-sidr-arch]
Lepinski, M. and S. Kent, "An Infrastructure to Support Secure Internet Routing", [draft-ietf-sidr-arch](#) (work in progress), March 2009.
- [ID.ietf-sidr-res-certs]
Huston, G., Michaleson, G., and R. Loomans, "A Profile for X.509 PKIX Resource Certificates", Internet Draft [draft-ietf-sidr-res-certs-18.txt](#), February 2009.

Geoff Huston
Asia Pacific Network Information Centre

Email: gih@apnic.net
URI: <http://www.apnic.net>

George Michaelson

Email: ggm@apnic.net
URI: <http://www.apnic.net>

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

Email: kent@bbn.com