Algorithm Agility for RPKI

Roque Gagliano
Stephen Kent
Sean Turner
Context

• SEC AD requested the SIDR WG to document a mechanism for algorithm migration in RPKI.
• We spent sometime reviewing key roll-over.
• The algorithm transition document will include:
  – A formal procedure to implement algorithm transition by updating rpki-algs document (and the CP?)
  – A set of steps/milestones to be achieved during the transition period.
  – Required behavior by CAs and RPs during migration
  – We will not have emergency mechanism for algorithm transition.
Algorithm migration (normal process):

- Materials **MIGHT** be issued under new suites.
- Materials **MUST** be issued under both suites.
- Materials **MIGHT** be issued under old suites.

**Events:**
- CA Ready
- CA Set
- CA Go
- RP Ready
- Twilight
- EOL

**Milestones:**
- Months? Years?

**Notes:**
- IETF updates Sidr-algs and Sets dates for milestones.
- IETF obsoletes old Sidr-algs document.
Question 1: Transition Path

- Top-down only: we implement a mechanism where a child CA can transition to a new algorithm only if its parent CA has already transitioned.

- Laissez faire: any CA can begin using the new algorithm suite at any point in the PKI hierarchy (if the parent can execute Proof of Possession).

Do we want to support only top-down algorithm transitions?
Answer 1

We recommend pursuing only the top-down option.
Reasoning:
  - It complies with the requirements from SEC-AD.
  - It may be simpler as it may avoid the need for cross-algorithm certificates (e.g., new-in-old)
  - If cross-algorithm certificates are needed, we still need support for PoP from parent CA (“CA Ready”).
  - It seems likely that the smaller number of higher tier entities will be ready for algorithm migration sooner than lower tiers (e.g., ISPs)
Question 2

Do we want to support multiple signatures in the CMS objects (ROAs & manifests)?

(See Section 3 draft-ietf-sidr-rpki-algs)

Support for multiple signatures may reduce:

- duplication of signed objects in repositories
- complexity of CA software
- complexity of repository operation
Answer 2

• We don’t recommend multi-signature objects
  – Certificates and CRLs MUST contain only a single signature, so only CMS objects are in play here
  – Spec would have to change
  – CAs may elect to publish separate, single signature objects anyway, so RPs have to accommodate
  – A multi-signature manifest is ambiguous, unless we modify the format to add an algorithm ID
ALG. ROLL-OVER (without KEY ROLL OVER) Repository Structure with single signature ROAs and Manifests.

CA X directory (RIR with no ROAs) Alg. A and B.

CA Y directory (ISP with ROAs and children) Alg. A and B

CA Z directory (End User, no child) Alg. A

Dir X

Dir Y

Dir Z

SIA: same PP for all CA Entity Y certs.

SIA: same PP for all CA Entity Z certs.
ALG. ROLL-OVER (without KEY ROLL OVER) Repository Structure with multiple signature ROAs and Manifests.

CA X directory (RIR with no ROAs)
Alg. A and B.

CA Y directory (ISP with ROAs and children)
Alg. A and B

CA Z directory (End User, no child)
Alg. A

CRL X\textsubscript{A-CUR}  CRL X\textsubscript{B-CUR}

CRL Y\textsubscript{A-CUR}  CRL Y\textsubscript{B-CUR}  CA Z\textsubscript{A-CUR}

roa\textsubscript{A-cur B-cur}  roa\textsubscript{A-cur}

manifest\textsubscript{A-cur B-cur}  manifest\textsubscript{A-cur}

Dir X  Dir Y  Dir Z

SIA: same PP for all CA Entity Y certs.
SIA: same PP for all CA Entity Z certs.
Question 3: Validation during transitions

• There are some interesting scenarios:
  – A prefix is validated using one algorithm suite (and stored in local cache) and the RP later receives a ROA signed using another algorithm set.
  – Two ROAs (identical content), each signed using different algorithm suites; one validates, the other does not. What does an RP do?

Comment: During transition both algorithm sets should be treated as equals.

Any thought on these issues or other validation issues during algorithm transition?
Summary

• Do we want to support only top-down algorithm transitions?

• Do we want to support multiple signatures in CMS signed objects?

• Any validation issue that should be addressed in transition document?
Backup
ALG. ROLL-OVER and KEY ROLL OVER Repository Structure with single signature ROAs and Manifests.

CA X directory (RIR with no ROAs)
Alg. A and B. Key Roll-over
CA Y directory (ISP with ROAs and children)
Alg. A and B. Key Roll-over
CA Z directory (End User, no child)
Alg. A. No key Roll-over
ALG. ROLL-OVER and KEY ROLL OVER Repository Structure with multiple signature ROAs and Manifests.

CA X directory (RIR with no ROAs)
Alg. A and B. Key Roll-over

CA Y directory (ISP with ROAs and children)
Alg. A and B. Key Roll-over

CA Z directory (End User, no child)
Alg. A. No key Roll-over