SIDR
Internet-Draft
Intended status: Informational
Expires: November 21, 2015

S. Kent
BBN
D. Ma
ZDNS

May 20, 2015

Adverse Actions by a Certification Authority (CA) or Repository Manager in the Resource Public Key Infrastructure (RPKI) draft-kent-sidr-adverse-actions-00

Abstract

This document analyzes actions by or against a CA or independent repository manager in the RPKI that can adversely affect the Internet Number Resources (INRs) associated with that CA or its subordinate CAs. The analysis is based on examination of the data items in the RPKI repository, as controlled by a CA (or independent repository manager) and fetched by Relying Parties (RPs). The analysis is performed from the perspective of an affected INR holder.

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 November 21, 2015.

Table of Contents

1. Introduction

Both Suspenders [I-D.kent-sidr-suspenders] and RPKI Validation Reconsidered [I-D.ietf-sidr-rpki-validation-reconsidered] address mistaes by Resource Public Key Infrastructure (RPKI) [RFC6480] Certification Authorities (CAs) (with respect to subordinate CAs). However, mistakes are not the only way that adverse changes to RPKI data can arise. A CA or repository operator might be subject to an attack [RFC7132]. For a CA, if an attack allows an adversary to use the private keys of that CA to sign RPKI objects, then the effect is analogous to the CA making mistakes. There is also the possibility that a CA or repository operator may be subject to legal measures that compel actions that result in generating "bogus" signed objects or removing legitimate repository data. In many cases, such actions may be hard to distinguish from non-malicious mistakes, other than with respect to the time required to remedy the adverse action. Thus this document examines the implications of adverse actions with respect to Internet Number Resources (INRs) irrespective of the cause of the actions. The document proposes mitigation strategies that take into account the nature of adverse actions, e.g., distinguishing malicious vs. erroneous actions.

This document analyzes the various types of actions by a CA (or independent repository manager) that can adversely affect the Internet Number Resources (INRs) associated with that CA, as well as the INRs of subordinate CAs. The analysis is based on examination of the data items in the RPKI repository, as controlled by a CA (or independent repository manager) and fetched by Relying Parties (RPs). The analysis is done from the perspective of an affected INR holder.

1.1. 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 [RFC2119].

2. Analysis of RPKI Repository Objects

This section enumerates the RPKI repository system objects and examines how changes to them affect Route Origination Authorizations (ROAs) and router certificate validation. Identifiers are assigned to errors for reference by later sections of this document.

The RPKI repository [RFC6481] contains a number of (digitally signed) objects that are fetched and processed by RPs. The principal goal of the RPKI, until the deployment of BGPsec [I-D.ietf-sidr-bgpsec-overview], is to enable an RP to validate ROAs [RFC6482]. A ROA binds address space to an Autonomous System Number (ASN). A ROA can be used to verify BGP announcements (with respect to route origin) [RFC6483]. The most important objects in the RPKI (for origin validation) are ROAs; all of the other RPKI objects exist to enable the validation of ROAs in a fashion consistent with the INR allocation system. Thus errors that result in changes to a ROA, or to RPKI objects needed to validate a ROA, can cause RPs to reach different conclusions about the validity of the bindings expressed in a ROA.

When BGPsec is deployed, router certificates [I-D.ietf-sidr-bgpsec-pki-profiles] will be added to repository publication points. These are End-Entity (EE) certificates used to verify signatures applied to BGP update data, to enable path validation [I-D.ietf-sidr-bgpsec-protocol]. Router certificates are as important to path validation as ROAs are to origin validation.

The objects contained in the RPKI repository are of two types: conventional PKI objects (certificates and Certificate Revocation Lists (CRLs)) and RPKI-specific signed objects. The latter make use of a common encapsulation format [RFC6488] based on the Cryptographic Message Syntax (CMS) [RFC5652]. A syntax error in this common format

will cause an RP to reject the object as invalid. In turn, this may cause a ROA at a publication point to be considered invalid.

Adverse actions take several forms:

* Deletion (D) is defined as removing an object from a publication point, without the permission of the INR holder.

* Suppression (S) is defined as not deleting an object, or not publishing an object, as intended by an INR holder. This action also includes retaining a prior version of an object in a publication point when a newer version is available for publication.

* Corruption (C) is defined as modification of a signed object in a fashion not requiring access to the private key used to sign the object. Thus a corrupted object will not carry a valid signature. Implicitly, the corrupted object replaces the legitimate version.

* Modification (M) is defined as publishing a version of an object that differs from the version authorized by the INR holder (but which is still syntactically valid). Implicitly, the legitimate version of the affected object is deleted and replaced by the modified object. The signature on the modified object will be valid in the RPKI.

* Revocation (R) is defined as revoking a certificate (EE or CA) by placing its serial number on the appropriate CRL, without authorization of the INR holder.

* Injection (I) is defined as introducing an instance of a signed object into a publication point. It assumes that the signature on the object will be viewed as valid by RPs.

The first three of these actions (deletion, suppression, and corruption) can be effected by any entity that manages the publication point of the affected INR holder. (An entity with the ability to act as a man-in-the-middle between an RP and a repository also can effect these actions with respect to the RP in question.)

The latter three actions (modification, revocation, and injection) nominally require access to the private key of the INR holder.

All six of these actions also can be effected by a parent CA. A parent CA could reissue the INR holder’s CA certificate, generate new signed objects using the private key associated with the reissued certificate, and publish these objects at a location of its choosing.

Most of these actions may be performed independently or in combination with one another. For example, a ROA may be revoked and deleted or revoked and replaced with a modified ROA. Where appropriate, the analysis of adverse actions will distinguish which of these individual actions, or combinations thereof, yield different outcomes for RPs. Recall that the focus of the analysis is the impact on ROAs and router certificates, with respect to RP processing.

2.1. ROA

In addition to the generic RPKI object syntax checks, ROA validation requires that the signature on the ROA can be validated using the public key from the EE certificate embedded in the ROA [RFC6482]. It also requires that the EE certificate be validated consistent with the procedures described in [RFC6482] and [RFC6487]. Adverse actions against a ROA can cause the following errors:

A-1.1 A ROA may be deleted from the indicated publication point.

A-1.2 A ROA may be revoked on the CRL for the publication point.

A-1.3 Publication of a newer ROA may be suppressed.

A-1.4 A ROA may be corrupted.

A-1.5 A valid ROA may be replaced with a corrupted ROA, which will be rejected by RPs.

A-1.6 A ROA may be modified so that the Autonomous System Number (ASN) or one or more of the address blocks in a ROA is different than the values authorized by the INR holder. (This action assumes that the modified ROA’s ASN and address ranges are authorized for use by the INR holder.)

A-1.7 If an INR holder intends to issue and publish two (or more) new ROAs for the same address space, one (or more) of the new ROAs may be suppressed while the other is published.

A-1.8 If an INR holder intends to delete all ROAs for the same address space, some of them may be held while the others are deleted.

2.2. Manifest

Each repository publication point contains a manifest [RFC6486]. The RPKI incorporates manifests to enable RPs to detect suppression and/or substitution of (more recent) publication point objects, as the

result of a mistake or attack. A manifest enumerates (by filename) all of the other signed objects at the publication point. The manifest also contains a hash of each enumerated file, to enable an RP to determine if the named file content matches what the INR holder identified in the manifest.

A manifest is an RPKI signed object, so it is validated as per [RFC6488]. If a manifest is modified in a way that causes any of these checks to fail, the manifest will be considered invalid. Suppression of a manifest itself (indicated by a stale manifest) also can cause an RP to not detect suppression of other signed objects at the publication point. However, RPs are not required to reject publication point entries in the face of an invalid manifest. If a signed object at a publication point can be validated (using the rules applicable for that object type), then an RP MAY accept that object, even if there is no matching entry for it on the manifest.

Corruption, suppression, modification, or deletion of a manifest might not affect RP processing of other publication point objects, as specified in [RFC6486]. However, many RP implementations ignore objects that are present at a publication point but not listed in a valid Manifest. Thus the following actions against can impact RP processing:

A-2.1 A Manifest may be deleted from the indicated publication point.

A-2.2 A Manifest may be revoked on the CRL for the publication point.

A-2.3 Publication of a newer Manifest may be suppressed.

A-2.4 A Manifest may be corrupted.

A-2.5 A valid Manifest may be replaced with a corrupted Manifest, which will be rejected by RPs.

A-2.6 A Manifest may be modified to remove one or more objects.

A-2.7 A Manifest may be modified to add one or more objects.

A-2.8 A Manifest may be modified to list an incorrect hash for one or more objects.

2.3. Ghostbusters Record

The Ghostbusters record [RFC6493] is a signed object that MAY be included at a publication point, at the discretion of the INR holder or publication point operator. The record is validated according to [RFC6488]. Additionally, the syntax of the record is verified based on the vCard profile contained in [RFC6493]. Errors in this record do not affect RP processing. However, if an RP encounters a problem with objects at a publication point, the RP may use information from the record to contact the publication point operator.

Adverse actions against a Ghostbusters record can cause the following error:

A-3.1 Suppression, deletion, or corruption of a Ghostbusters record could prevent an RP from contacting the appropriate entity when a problem is detected by the RP. Modification of a Ghostbusters record could cause an RP to contact the wrong entity, thus delaying remediation of an detected anomaly. All of these actions are viewed as equivalent from an RP processing perspective; they do not alter RP validation of ROAs or router certificates. However, these actions can interfere with remediation of the action when detected by an RP.

2.4. Certificate Revocation List

Each publication point contains a CRL that enumerates revoked (and not yet expired) certificates issued by the CA associated with the publication point [RFC6481].

Adverse actions against a CRL can cause the following errors:

A-4.1 If a CRL is deleted, RPs will continue to use an older, previously fetched Certificate Revocation List. As a result, they will not be informed of any changes in revocation status of subordinate CA or router certificates or affected subordinate signed objects, e.g., ROAs or CA certificates. This action is essentially equivalent to corruption of a CRL, since a corrupted CRL will not be accepted by an RP.

A-4.2 If publication of the most recent CRL is suppressed, an RP will not be informed of the most recent revocation status of subordinate CA or router certificates or affected subordinate signed objects. If an EE certificate has been revoked and the associated signed object is still present in the publication point, an RP might mistakenly treat that

object as valid. (This would happen if the object is still in the manifest or the RP is configured to process valid objects that are not on the manifest.) This type of action is of special concern if the affected object is a ROA, a router certificate, or a subordinate CA certificate (since suppression of revocation of any of these objects can have a substantial impact on the RPKI).

A-4.3 If a CRL is modified to erroneously list a signed object's EE certificate as revoked, the corresponding object will be treated as invalid by RPs, even if it is present in a publication point. If this object is a ROA, the (legitimate) binding expressed by the ROA will be ignored by an RP. If a CRL is modified to erroneously list a router certificate as revoked, a path signature associated with that certificate will likely be treated as invalid by RPs.

A-4.4 If a CRL is modified to erroneously list a CA certificate as revoked, that CA and all subordinate signed objects will be treated as invalid by RPs. Because RPs acquire RPKI data based on AIA and SIA extensions in CA certificates, revocation of a CA certificate may cause RPs to not retrieve all subordinate signed objects. Thus erroneous revocation of a CA certificate may have significant implications for RPs.

A-4.5 If a CRL is modified to omit a revoked EE, router, or CA certificate, RPs may continue to accept the revoked, signed object as valid. This contravenes the intent of the INR holder.

2.5. CA Certificates

Every INR holder is represented by one or more CA certificates. An INR holder has multiple CA certificates if it holds resources acquired from different sources. Also, every INR holder has more than one CA certificate during key rollover [RFC6489] and algorithm rollover [RFC6916].

If a publication point is not a leaf in the RPKI hierarchy, then the publication point will contain one or more CA certificates, each representing a subordinate CA. Each subordinate CA certificate contains a pointer (SIA) to the publication point where the signed objects associated with that CA can be found [RFC6487].

A CA certificate is a complex data structure and thus errors in that structure may have different implications for RPs depending on the specific data that is in error.

Adverse actions against a CA certificate can cause the following errors:

A-5.1 Revocation or deletion of a CA certificate would cause an RP to not be able to locate signed objects generated by that CA. Suppression of a CA certificate with a changed SIA value would have an equivalent effect. Thus an RP would be unaware of the INR bindings asserted in subordinate ROAs, and the RP would be unable to validate router certificates.

A-5.2 Corruption of a CA certificate will cause it to be rejected by RPs. In turn, this will cause any subordinate signed objects to become invalid.

A-5.3 If a CA certificate is modified, but still conforms to the RPKI certificate profile [RFC6485], it will be accepted by RPs. If an [RFC3779] extension in this certificate is changed to exclude INRs that were previously present, then subordinate signed objects will become invalid if they rely on the excised INRs. If these objects are CA certificates, their subordinate signed objects will be treated as invalid. If the objects are ROAs, the binding expressed by the affected ROAs will be ignored by RPs. If the objects are router certificates, BGPsec_Path attributes [I-D.ietf-sidr-bgpsec-protocol] verifiable under these certificates will likely be considered invalid.

A-5.4 If the SIA extension of a CA certificate is modified to refer to another publication point, this will cause an RP to look at another location for subordinate objects. This could cause RPs to not acquire the objects that the INR holder intended to be retrieved. In turn, RPs would not be able to acquire (much less validate) ROAs, router certificates, or any subordinate CA certificates associated with that CA.

A-5.5 If the AIA extension in a CA certificate is modified, it would point to a different CA certificate, not the parent CA certificate. This extension is used only for path discovery, not path validation. Path discovery in the RPKI is usually performed on a top-down basis, starting with TAs and recursively descending the RPKI hierarchy. Thus there

may be no impact on the ability of clients to acquire and validate certificates if the AIA is modified.

A-5.6 If the Subject Public Key Info (and Subject Key Identifier extension) in a CA certificate is modified to contain a public key corresponding to a private key held by the parent, the parent could sign objects as children of the affected CA certificate.

2.6. Router Certificates

Router certificates are used by RPs to verify signatures on BGPsec_Path attributes carried in Update messages.

Each AS is free to determine the granularity at which router certificates are managed [I-D.ietf-sidr-bgpsec-pki-profiles]. Each participating AS is represented by one or more router certificates. During key or algorithm rollover, multiple router certificates will be present in a publication point, even if the AS is normally represented by just one such certificate.

Adverse actions against router certificates can cause the following errors:

A-6.1 Suppression, revocation, or deletion of a router certificate would cause an RP to not be able to verify signatures applied to BGPsec_Path attributes on behalf of the AS in question. In turn, this would cause the route to be treated with lower preference than competing routes that have valid BGPsec_Path attribute signatures. (However, if another router certificate for the affected ASN is valid and contains the same AS number and public key, and is in use by that AS, there would be no effect on routing. This scenario will arise if a router certificate is renewed, i.e., issued with a new validity interval.) Modification of a router certificate that causes it to fail syntax checks will result in the certificate being rejected by RPs. Absent a valid router certificate, BGPsec_Path attributes associated with that certificate will be unverifiable. In turn, this would cause the route to be treated with lower preference than competing routes that have valid BGPsec_Path attribute signatures.

A-6.2 If a router certificate is modified to represent a different ASN, but it still passes syntax checks, then this action could cause signatures on BGPsec_Path attributes to be associated with the wrong AS. This could cause signed

routes to be inconsistent with the intent of the INR holder.

3. Analysis of Actions Relative to Scenarios

This section examines the types of problems that can arise in four scenarios described below. We consider mistakes, (successful) attacks against a CA or a publication point, and situations in which a CA or publication point manager is compelled to take action by a law enforcement authority.

We explore the following four scenarios:

A. An INR holder operates its own CA and manages its own repository publication point.

B. An INR holder operates its own CA, but outsources management of its repository publication point to its parent or another entity.

C. An INR holder outsources management of its CA to its parent, but manages its own repository publication point.

D. An INR holder outsources management of its CA and its publication point to its parent.

Note that these scenarios focus on the affected INR holder as the party directly affected by an adverse action. The most serious cases arise when the INR holder appears as a high-tier CA in the RPKI hierarchy; in such situations subordinate INR holders may be affected as a result of an action. A mistake by or an attack against a "leaf" has more limited impact because all of the affected INRs belong to the INR holder itself.

In Scenario A, actions by the INR holder can adversely affect all of its resources and, transitively, resources of any subordinate CAs. (If the CA is a "leaf" in the RPKI, then it has no subordinate CAs and the damage is limited to its own INRs.)

In Scenario B, actions by the (outsourced) repository operator also can adversely affect the resources of the INR holder, and those of any subordinates CAs. (If the CA is a "leaf" in the RPKI, then it has no subordinate CAs and the damage is limited, as in Scenario A.) The range of adverse effects here includes those in Scenario A, and adds a new potential source of adverse actions, i.e., the outsourced repository operator.

In Scenario C, all signed objects associated with the INR holder are generated by the parent CA but are self-hosted. (We expect this scenario to be rare, because an INR holder that elects to outsource CA operation seems unlikely to manage its own repository publication point.) Because that CA has the private key used to sign them, it can generate alternative signed objects---ones not authorized by the INR holder. However, erroneous objects created by the parent CA will not be published by the INR holder IF the holder checks them first. Because the parent CA is acting on behalf of the INR holder, mistakes by or attacks against that entity are equivalent to ones effected by the INR holder in Scenario A.

The INR holder is most vulnerable in Scenario D. Actions by the parent CA, acting on behalf of the INR holder, can adversely affect all signed objects associated with that INR holder, including any subordinate CA certificates. These actions will presumably translate directly into publication point changes, because the parent CA is managing the publication point for the INR holder. The range of adverse effects here includes those in Scenarios A, B, and C.

3.1. Scenario A

In this scenario, the INR holder acts as its own CA and it manages its own publication point. Mistakes by the INR holder can cause any of the actions noted in Section 2. A successful attack against this CA can effect all of the modification, revocation, or injection actions noted in that section. (We assume that objects generated by the CA are automatically published). An attack against the publication point can effect all of the deletion, suppression, or corruption actions noted in that section.

3.2. Scenario B

In this scenario, the INR holder acts as its own CA and but it outsources management of it own publication point. Mistakes by the INR holder can cause any of the actions noted in Section 2. A successful attack against its CA can effect all of the modification, revocation, or injection actions noted in that section (assuming that objects generated by the CA are automatically published). Here, actions by the publication point manager (or attacks against that entity) can effect all of the deletion, suppression, or corruption actions noted in Section 2.

3.3. Scenario C

In this scenario, the INR holder outsources management of its CA to its parent, but manages its own repository publication point. Mistakes by the INR holder, acted upon by the parent CA, can cause

any of the actions noted in Section 2. Actions unilaterally undertaken by the parent CA also can have the same effect, unless the INR holder checks the signed objects before publishing them. A successful attack against the parent CA can effect all of the modification, revocation, or injection actions noted in Section 2. An attack against the INR holder can effect all of the deletion, suppression, or corruption actions noted in Section 2 (because the INR holder is managing its publication point). (An attack against the INR holder implies that the path it uses to direct the parent CA to issue and publish objects has been compromised.)

3.4. Scenario D

In this scenario an INR holder outsources management of both its CA and its publication point to its parent. Mistakes by the INR holder, acted upon by the parent CA, can cause any of the actions noted in Section 2. Actions unilaterally undertaken by the parent CA also can have the same effect. A successful attack against the parent CA can effect all of the modification, revocation, or injection actions noted in Section 2. An attack against the parent CA can effect all of the deletion, suppression, or corruption actions noted in Section 2 (because the parent CA is managing the INR holder’s publication point).

4. Detection and Remediation

Each INR holder SHOULD check the signed objects available in its publication point, to detect problems, on a regular basis. This document RECOMMENDS that each INR holder perform such checks as a side-effect of acquiring RPKI data for local processing, e.g., 3-4 times a day. Third parties also can perform checks in behalf of RPs, to detect adverse actions. This is consistent with the outsourcing options noted in the use cases in Section 1. In either situation, detection of adverse actions requires that the cognizant party have available a reference set of RPKI data for the INR holder. The reference set includes all signed objects in the publication point(s) of the INR holder plus the CA certificate for each publication point.

If an adverse action is the result of a mistake by, or an attack against, a superior CA or a repository manager, the INR holder SHOULD contact the relevant entity as soon as possible. It is expected that a mistake by these entities normally will be corrected and new objects published within 24-72 hours. An attack against a superior CA or repository manager also should be remedied by the affected parties, but the time to repair the damage may be longer, because of the additional activities that may accompany responding to an attack.

In both of these situations, the harmful effects of adverse actions can be mitigated if RPs delay acting on changes that might be attributable to adverse actions. This observation motivates the introduction of hysteresis into the RPKI validation process, at least with respect to changes that are indicative of an adverse action. The idea is that an RP would continue to accept as valid objects that were previously valid (based on local cache history), and ignore recent changes, for some interval. However, sometimes an INR holder may wish to make a change that might be viewed as an adverse action, without encountering such a delay.

To accommodate this situation, the RPKI can be extended to allow an INR holder to provide independent confirmation of such changes. A mechanism of this sort could prevent mistakes by a superior CA or repository manager from having an immediate, adverse effect. If the independent confirmation is not completely dependent on the RPKI repository and CA system, it can be immune to mistakes by or attacks against superior CAs and repository managers, and outsourced CA and repository managers.

The effects of an attack on an INR holder itself may not be countered by a mechanism of this sort; an adversary who can attack a CA and/or repository management function of an INR holder may be able to attack the independent confirmation mechanism at the same time. The use of a separate mechanism does create the potential for additional safeguards against such attacks. However, the extent to which this potential is achieved will depend on how an INR holder implements and manages this mechanism.

If a superior CA or repository manager is compelled to engage in an adverse action against an INR holder, e.g., by a law enforcement agency, use of an independent confirmation mechanism also may be able to counter such an action. In this situation, the actions are not likely to be reversed quickly, unlike a mistake or attack. This situation might argue for a longer period of delay. An INR holder and a subordinate INR holder may disagree about an action that invalidates the holdings of the subordinate. The addition of hysteresis and an independent confirmation mechanism to the RPKI ought not deprive an INR holder of the legitimate ability to take such actions. This argues for a time limit on the hysteresis and confirmation mechanism, consistent with the business practices of RPKI CAs. This time limit should be long enough to allow affected entities to remedy mistakes and recover from attacks. It also should be short enough to not impede the resolution of legitimate actions by an INR holder relative to subordinate INR holders.

5. Security Considerations

This informational document describes a threat model for the RPKI, focusing on mistakes by or attacks against CAs and independent repository managers. It is intended to support the design of future RPKI security mechanisms that seek to address the concerns associated with such actions.

6. IANA Considerations

This document has no actions for IANA.

7. Acknowledgements

The authors would like to thank Richard Hansen and David Mandelberg for their review feedback. The authors also thank Richard Hansen for his editorial assistance.

8. References

8.1. Normative References

[I-D.ietf-sidr-bgpsec-overview]
Lepinski, M. and S. Turner, "An Overview of BGPsec", draft-ietf-sidr-bgpsec-overview-06 (work in progress), January 2015.

[I-D.ietf-sidr-bgpsec-pki-profiles]
Reynolds, M., Turner, S., and S. Kent, "A Profile for BGPSEC Router Certificates, Certificate Revocation Lists, and Certification Requests", draft-ietf-sidr-bgpsec-pki-profiles-10 (work in progress), January 2015.

[I-D.ietf-sidr-bgpsec-protocol]
Lepinski, M., "BGPsec Protocol Specification", draft-ietf-sidr-bgpsec-protocol-11 (work in progress), January 2015.

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.

[RFC3779] Lynn, C., Kent, S., and K. Seo, "X.509 Extensions for IP Addresses and AS Identifiers", RFC 3779, June 2004.

[RFC5652] Housley, R., "Cryptographic Message Syntax (CMS)", STD 70, RFC 5652, September 2009.

[RFC6480] Lepinski, M. and S. Kent, "An Infrastructure to Support Secure Internet Routing", RFC 6480, February 2012.

[RFC6481] Huston, G., Loomans, R., and G. Michaelson, "A Profile for Resource Certificate Repository Structure", RFC 6481, February 2012.

[RFC6482] Lepinski, M., Kent, S., and D. Kong, "A Profile for Route Origin Authorizations (ROAs)", RFC 6482, February 2012.

[RFC6483] Huston, G. and G. Michaelson, "Validation of Route Origination Using the Resource Certificate Public Key Infrastructure (PKI) and Route Origin Authorizations (ROAs)", RFC 6483, February 2012.

[RFC6485] Huston, G., "The Profile for Algorithms and Key Sizes for Use in the Resource Public Key Infrastructure (RPKI)", RFC 6485, February 2012.

[RFC6486] Austein, R., Huston, G., Kent, S., and M. Lepinski, "Manifests for the Resource Public Key Infrastructure (RPKI)", RFC 6486, February 2012.

[RFC6487] Huston, G., Michaelson, G., and R. Loomans, "A Profile for X.509 PKIX Resource Certificates", RFC 6487, February 2012.

[RFC6488] Lepinski, M., Chi, A., and S. Kent, "Signed Object Template for the Resource Public Key Infrastructure (RPKI)", RFC 6488, February 2012.

[RFC6489] Huston, G., Michaelson, G., and S. Kent, "Certification Authority (CA) Key Rollover in the Resource Public Key Infrastructure (RPKI)", BCP 174, RFC 6489, February 2012.

[RFC6493] Bush, R., "The Resource Public Key Infrastructure (RPKI) Ghostbusters Record", RFC 6493, February 2012.

[RFC6916] Gagliano, R., Kent, S., and S. Turner, "Algorithm Agility Procedure for the Resource Public Key Infrastructure (RPKI)", BCP 182, RFC 6916, April 2013.

[RFC7132] Kent, S. and A. Chi, "Threat Model for BGP Path Security", RFC 7132, February 2014.

8.2. Informative References

[I-D.ietf-sidr-rpki-validation-reconsidered]
Huston, G., Michaelson, G., Martinez, C., Bruijnzeels, T., Newton, A., and A. Aina, "RPKI Validation Reconsidered", draft-ietf-sidr-rpki-validation-reconsidered-01 (work in progress), January 2015.

[I-D.kent-sidr-suspenders]
Kent, S. and D. Mandelberg, "Suspenders: A Fail-safe Mechanism for the RPKI", draft-kent-sidr-suspenders-03 (work in progress), April 2015.

Authors' Addresses

Stephen Kent
BBN Technologies
10 Moulton St
Cambridge, MA 02138-1119
USA

EMail: kent@bbn.com

Di Ma
ZDNS
4 South 4th St.
Zhongguancun
Haidian, Beijing 100190
China

EMail: madi@zdns.cn