Resource Certification - A Public Key Infrastructure for IP Addresses and AS's

Geoff Huston, George Michaelson
Asia Pacific Network Information Centre
{gih, ggm}@apnic.net

DRAFT - November 2008

Abstract - X.509 Public Key certificates are typically used to validate attestations related to identity or role. The overwhelming number of large scale deployments seen in public networks serve this purpose. Here, we examine a different form of X.509 certificate that is used to describe IP address and AS number resources and bind them to a public/private key pair. These certificates are used to attest to resource allocation actions, so that digitally signed attestations relating to a party's right-of-use of IP addresses and AS numbers can be validated by relying parties, using a related Resource Certificate Public Key Infrastructure (RPKI). This has particular application in the area of demonstrable attestations related to the right-of-use of IP addresses, and in the area of inter-domain routing security. The issues related to the application of this RPKI to inter-domain routing security are considered, and the design, management and use of resource certificates, and the structure of the related Public Key Infrastructure are described in detail.

Index Terms – BGP security, Inter-domain routing security, X.509, Public Key Infrastructure

1. Introduction

In November 2008 the Asia Pacific Network Information Centre (APNIC) announced the release of a public 'resource certification service' that makes use of X.509 public key certificates [X.509] to publish public key certificates and associated signed objects that uniquely associate a private key holder with a 'right-of-use' of a collection of IP number resources (IPv4 addresses, IPv6 addresses and Autonomous System (AS) Numbers). This APNIC activity forms part of a larger certificate infrastructure effort that is ultimately intended to provide certification for all in-use number resources in the public Internet. This report describes this Resource Public Key Infrastructure (RPKI) in more detail, looking at the various aspects of the design that lie behind the construction of this particular PKI.

2. Motivation

Opinions vary as to what aspect of the Internet's infrastructure represents the greatest common vulnerability to the security and safety of Internet users, but it is generally regarded that the choice is one of the Domain Name System (DNS) or the inter-domain routing system.

Corrupting the name-to-address translation that is provided by DNS resolution services allows for site masquerading and 'passing off', traffic redirection, denial of service and other forms of service corruption, and with a selective attack on the DNS (i.e., one that harms only a subset of the global internet, perhaps to a single client, or users of a single DNS resolver) the problem cannot be seen outside of a very restricted scope. In such an attack on the DNS, the operation of the underlying packet transmission network, when considered as a set of sources and destinations of IP traffic, has not been corrupted, as the attack is directed at the function of mapping from names to the addresses (and vice-versa).

Corrupting routing can lead to a similar set of undesirable outcomes, including traffic inspection as well as masquerading, denial of service and selective corruption of services. It has also been argued that the deliberate corruption of routing makes DNS interception and corruption easier to undertake, and very probably harder to detect. On this basis routing is often positioned as the more critical security vulnerability of the two, and potentially the most critical security vulnerability of the Internet. It is certainly the case that corrupting the routing system to advertise an additional 'rogue' instance of a set of anycast name servers, and, in particular of a root name server, allows for a large set of consequent attacks that are based on DNS corruption [Chakrabarti 2002]. Others see the widely distributed trust model that lies behind the DNS as the most readily exploitable vulnerability, and argue that the DNS is the weaker link, as DNS attacks can be carefully crafted to selectively poison particle name resolvers with corrupt address information for a selected set of domain names [Kaminsky 2008]. Irrespective of any individual preference here, both realms, the DNS and routing, represent a continuing source of vulnerability for all users of the Internet. Both are therefore worthy of protection.

This is obviously not a novel observation, and measures to secure the operation of both the DNS and inter-domain routing have been considered by the Internet technical and engineering community for over a decade now. In the case of the DNS the preferred longer term approach is the universal adoption of DNSSEC [RFC4033]. By using public / private key technology, and exploiting the hierarchical structure of the DNS namespace, DNSSEC creates an interlocking key structure that allows a DNS end user to validate a response to a DNS query. With a single point of trust in the public key associated with the private key used to sign the root of the DNS it is possible, in a comprehensive DNSSEC world, to validate any DNS response, and even to validate a negative response of no such domain. At the same time as the technology for securing the DNS was being developed a similar study was underway with respect to securing the inter-domain routing system, which is, in turn, focussed on securing

the operation of the predominant inter-domain routing protocol, the Border Gateway Protocol (BGP) [RFC4271].

Just as in the DNS, an attractive approach to constructing an interlocking key structure in the address real is to leverage the strictly hierarchical nature of internet number resource assignment in constructing a corresponding PKI.

Securing the routing system requires a number of measures, including:

- securing access to routers to prevent unauthorized access and malicious reconfiguration,
- securing the connection between routing-active agents to prevent disruption of the communication channel used by the routing protocol, and
- validation of the protocol payload to detect efforts to inject false information into the routing system.

Each of these measures to secure routing requires a different form of response in terms of security infrastructure [Huston 2005].

Securing the routing devices themselves is normally undertaken by 'shared secret' mechanisms that secure the channel used to access the router (such as the secure shell protocol, ssh) as well as shared secret mechanisms to secure access to the device (access and authentication) and access parts of the device's configuration state (access permissions). This form of protection is basic, and widely deployed. It lies outside the scope of this report.

Securing the BGP communications channel is a specific instance of a more general function of securing a long-held TCP session, and approaches to this typically use MD5, and studies on the applicability of IPSEC have been undertaken. Again, this is not a consideration of this report.

The third objective in the above list encompasses the objective of enabling BGP speakers to validate the authenticity and validity of the routing information that is passed to them by a BGP 'peer'. This routing information is in the form of assertions of reachability of address prefixes, and assertions of an associated 'vector' of AS's that form the AS Path attribute. The validation questions that apply to this routing information include:

- Is this prefix a valid prefix to advertise into the routing system?
- Has the holder of the "right-of-use" of this address prefix authorized the originating AS to perform this advertisement?
- Did the authorized AS actually originate this route object?
- Does the AS Path of the route object represent the sequence of AS's through which the route object has been propagated?
- Are all the AS's in the AS Path valid?
- Does the next hop address represent a feasible forwarding path to reach the address prefix?[1]
- Does the forwarding path represent a series of switching decisions that are consistent with the local traffic forwarding policies at each step in the path?

The questions of potential relevance to the RPKI relate to establishing the validity of address prefixes and AS's, and the validity of authorities and attestations that are being made by the holders of addresses and AS's.

The objective of the RPKI is therefore to provide a means of validation of the authenticity of an IP address or AS. This authenticity means being able to determine that the address or AS number has been validly allocated or assigned, and that the address can be announced into the routing system and that the AS number can be used within the attributes of the routing information systems. In addition, the RPKI can validate the association between an address or AS number and its current right-of-use holder. This validation function can be interpreted as a validation of a title over the right-of-use of addresses, and this function of validation of title is expected to be of utility in a number of areas, and not strictly limited to that of routing-protocol security.

3. Prior Work in Routing Security

The initial approach that was used to provide some level of certainty regarding the legitimacy of the use of IP addresses in the routing system was the IP address allocation registry (and subsequently the AS number allocation registry), originally administered by the Internet Assigned Numbers Authority (IANA), and now undertaken by the five Regional Internet Registries (RIRs) as well as the IANA. The IANA now published registries for IPv4 address allocations, IPv6 allocations and AS number allocations. With some exceptions the IANA registries simply list the allocations to RIRs. The RIR registries collectively contain the current list of all validly allocated number resources and the details of the identity of the party to whom the resources were allocated.

There are some issues in using this published registry information to validate the authenticity of the use of number resources in a routing context and to authenticate the routing information with the registry-published details of the party to whom the resource was allocated or assigned:

- the RIR registry data is published in its complete form only under terms of a research agreement, due to community concerns over data mining of the registry
- the available query tool, "whois" [RFC3912], is insecure and readily disrupted by a number of forms of attack
- the query servers generally restrict the query rate from any single client, due to these same community concerns
- the collection of registry data is incomplete and out of date in certain parts, and inconsistencies

[1] This should be matched against the objectives of the routing system itself. One possible phrasing of the objective of the routing system is to ensure that every switching element in a network is configured with decision parameters that ensures that each packet is delivered to its destination along a network path which is the best possible path within the constraints of locally applied policies. This implies that packets should not be discarded if there exists at least one valid path to the destination and that packets should not loop. The validation questions noted here can be summarized by the question: "Is the protocol functioning correctly?" The routing system question related to the objectives of the routing system is subtly different: "Is the operation of the protocol maintaining a consistent set of forwarding decisions in each active switching element within the network?"

between different published "whois" entries have to be resolved by hand.

It appears to be a poor choice to use only whois queries to underpin a framework for secure routing for the Internet. Even if it were the case that the underlying registry data was to be corrected and all inconsistencies removed, the issues related to insecurity of access and inability to validate the data essentially relegate this "raw" registry data as unusable in any real time context of routing security.

A refinement to this approach is to refine the registry concept into "routing registries". Internet Routing Registries (IRRs) are registries than contain, in addition to information relating to AS numbers and IP addresses, structured entries that relate to the AS adjacencies that exist in the routing space and the applicable routing policies that AS's apply to these adjacencies. IRRs also contain entries that describe origination of routing information, binding together an address prefix and an originating AS. IRRs, by common agreement, use the RPSL [RFC2622] notation, an object model based on textual "type: value" descriptive fields. The major operational use for IRRs has been in the construction and maintenance of automated routing filters for inter-domain routers. By traversing an IRR, matching AS import and export routing policies, joining the inferred propagation information to the IRR-declared prefix origination for each AS, it is possible to construct the list of all prefixes that an adjacent AS may announce to its peer. From that complete list of possible announced prefixes a filter list can be constructed, which allows the local BGP instance the ability to declare any other routing information as "unauthorised" and filter it out of consideration.

In terms of positive attributes, this system has been able to prevent accidental route leaks from propagating out into the inter-domain routing space, and it ensures that routes are added into the routing system via a deliberative process rather than as an accidental outcome. However, IRRs are not used universally, and the partial use of IRR systems limits their general applicability, and this approach has had a number of problems:

- There is no method of authenticating the data retrieved from an IRR, and most methods of access to an IRR are unsecured. Having sourced IRR data, once dissociated from its point of publication there is no clear method to identify where it came from, and thus what level of trust to place in it.

- There are many IRRs and each have differing policies of admission, and can hold differing data. There is no capability to ensure consistency of information across IRRs.

- The IRR publication model is not inherently secure and very few IRRs implement a strict condition that IRR data should be derived from allocation registry data. IRRs use differing admission policies, the publication model is insecure and therefore there is no easy method for a client of an IRR to establish the currency and accuracy of IRR data [Steenbergen 2008].

Overall, the trust model of the IRRs appears to relate to trust in the data admission policies of the IRR, which, in turn, places an undue level of reliance in the location of publication of the data as distinct from establishing trust through explicit validation of the data. Efforts to improve this situation were studied in the late 1990s, but few IRRs have implemented the measures proposed by this Routing Policy System Security study [RFC2725].

Work has also focussed on the operation of BGP in an effort to secure the operation of the protocol and validate the contents of BGP Update messages. These studies have used a number of approaches to provide the appropriate validation mechanism, including referral to the DNS and the potential use of DNSSEC, "web of trust" techniques, simple signed assertions and (the focus of this report) by reference to an external certificate hierarchy that is aligned to the resource allocation hierarchy. Some major contributions in this area of study so far include sBGP [Kent 2000], soBGP [White 2003], psBGP [Oorschot 2007] , IRR [Goodell 2003], and the use of an AS RR in the DNS, signed by DNSSEC [Bates 1998].

The common factor in these approaches is that they all require as a basic input a means of validating two basic assertions relating to origination of a route into the inter-domain routing system:

1) that the IP address block and the AS numbers being used are valid to use, and
2) that the parties using these IP addresses and AS numbers are properly authorized to so do.

The mechanisms proposed to perform this validation vary from simple assertion through peer corroboration through to a comprehensive resource PKI. It appears that the proposals that rely on the existence of a comprehensive resource PKI do so in the face of the obvious fact that until now, no such PKI exists today. It appears that in most cases the proposals that make use of weaker models of assertion and web of trust could be replaced by a resource PKI with no loss of functionality and a significant improvement in the level of trust that could be placed in the outcome of the validation process. The essential common approach is to provide an associated "feed" of signed credential information, which could be used to validate the feed of routing information, and validation of these credentials could be performed through the RPKI.

4. Resource Certificates and the Resource Public Key Infrastructure

Resource Certificates are X.509 certificates that conform to the PKIX profile [RFC5280] and that also contain a mandatory certificate extension that lists a collection of IP resources (IPv4 addresses, IPv6 addresses and AS Numbers) [RFC3779]. These certificates attest that the certificate's issuer has granted to the subject a unique "right-of-use" of the associated set of IP resources by virtue of a resource allocation action. The certificates are not identity attestation certificates, nor are they role authority certificates, nor are they instances of permission certificates. The certificates do not attest to the identity of the certificate's subject. The unique "right-of-use" concept mirrors the resource allocation framework, where the certificate provides a means of third-party validation of assertions related to resource allocations. By coupling the issuance of a certificate by a parent CA to the corresponding resource allocation, a test of the certificate validity including the RFC3779 extension can

also be interpreted as validation of that allocation. Signing operations which descend from that certificate can therefore be held to be testable, under the corresponding hierarchy of allocation.

A Resource Certificate describes an action by the certificate issuer that binds a list of IP Address blocks and AS Numbers to the subject of the certificate. The binding is identified by the implicit association of the subject's private key with the subject's public key contained in the Resource Certificate, signed by the private key of the certificate's issuer. Any instrument signed by the subject's private key that relates to an assertion of resource control can be validated through the matching public key contained in the certificate and validation of the certificate itself in the context of a resource PKI [Lepinski 2008b].

The intent of the Resource Public Key Infrastructure (RPKI) is to construct a robust hierarchy of X.509 certificates that allows relying parties to validate assertions about IP addresses and AS Numbers, and their use. The RPKI allows a relying party to determine if an address is valid to use in the context of the public Internet, and is able to validate assertions relating to the current "right-of-use" holder of an AS number or IP address.

The structure of the RPKI is designed to precisely mirror the structure of the distribution of addresses and AS's in the Internet, so a brief description of this distribution structure is appropriate. The Internet Assigned Number Authority (IANA) manages the central pool of number resources. The IANA publishes a registry of all current allocations. The IANA does not make direct allocations of number resources to end users or Local Internet Registries(LIRs), and, instead allocates blocks of number resources to the RIRs. The RIRs perform the next level of distribution, allocating number resources to LIRs, National internet Registries (NIRs) and end users. NIRs perform allocations to LIRs and end users, and LIRs allocate resources to end users. (Figure 1)

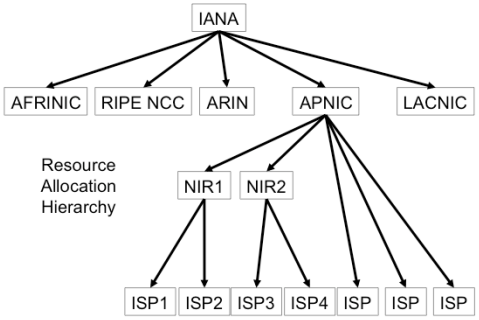

Figure 1. Address Distribution Hierarchy for the Internet

The RPKI mirrors this allocation hierarchy. In this model the IANA would issue PKIX certificates to each of the RIRs [RFC5280], describing in a resource extension to the certificate [RFC3779] the complete set of number resources that have been allocated to that RIR. The certificate would also hold the public key of the RIR and should be signed by the private key of the IANA. Each RIRs issues certificates that correspond to allocations made by that RIR, where the resource extension to the certificate lists all the allocated resources, and the certificate holds the public key of the recipient of the resource allocation, signed with the private key of the RIR. If the recipient of the resource allocation is an LIR or an NIR then it would also issue resources certificates in a similar vein (Figure 2).

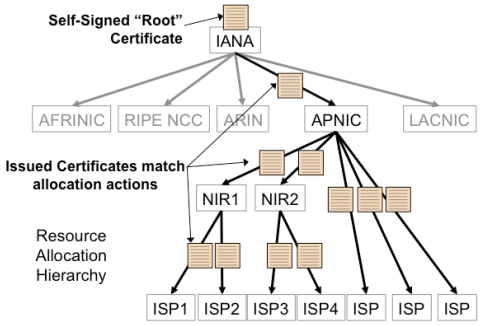

Figure 2. RPKI Resource Certificate Hierarchy

The common constraint within this certificate structure is that an issued certificate must contain a resource extension that contains a subset of the resources that are described in the resource extension of the issuing authority's certificate. This corresponds to the allocation constraint than an Internet Registry cannot allocate resources that were not allocated to the registry in the first place. The implication of this constraint is that if any party holds resources allocated from two or more registries then it will hold two or more resource certificates to describe the complete set of its resource holdings.

When an additional allocation action occurs, the associated certificate is reissued with a resource extension that matches the new allocation state. In the case of a reduction in allocated resources the previous certificates are explicitly revoked. In other cases there is no explicit revocation of the older certificates.

Validation within this RPKI validation of a certificate is similar to conventional certificate validation within any PKI, namely establishing a chain of valid certificates that are linked by issuer and subject from a nominated trust anchor CA to the certificate in question. The only additional constraint in the RPKI is that every certificate in this validation path must be valid resource certificates, and that the resources described in each certificate are a subset of the resources described in the issuing authorities certificate. In addition, the Trust Anchor in the RPKI is defined as a CA and an associated resource set. It is noted that the validation question is phrased as a general question, and is not phrased as a question relating to the validity of a specific IP address or AS number. For example, if party A has been issued a certificate with the AS resource set {1,2} and issues a subordinate certificate to party B with the AS resource set {1,2,3}, and then B issues itself an EE certificate with the AS resource set {1}, then if a relying party attempts to validate an assertion B is making about AS 1 using the EE certificate, the validation of the EE certificate will fail on the grounds that {1,2,3} is not a subset of {1,2}, even though AS 1 is a member of both sets.

The profile for Resource Certificates is described in Table 1, indicating all fields that must be included in a resource certificate. The certificate profile further requires that no other extensions be present in a resource certificate. [Huston 2008a]

Version	3
Serial Number	+ve number, unique per issuer
Signature Algorithm	minimum SHA-256
Issuer	Distinguished Name of the certificate issuer
Subject	Issuer-assigned Distinguished Name
Valid From / To	validity dates
Subject Public Key Info	Subject public key and algorithm
Basic Constraints	present for CA certificate
Subject Key Identifier	SHA-1 hash of subject public key
Authority Key Identifier	SHA-1 has of the issuer's public key
Key Usage	keyCertSign for CA, digitalSignature for EE
CRL Distribution Point	URL of issuer's published CRL
Authority Information Access	URL of issuer's superior certificate
Subject Information Access	URL of subject's repository publication point
Certificate Policies	Resource Certificate Policy Identifier
IP Resources	Issuer-allocated IPv4 and IPv6 addresses
AS Resources	Issuer-allocated AS Numbers

Table 1. Resource Certificate Profile.

The distinguished name of the subject in a certificate is normally nominated by the subject and verified by the issuer. In this case the certificate issuer is not making any attestation regarding the right of the subject to assert any particular identity, so within this PKI the distinguished name is selected by the issuer, and is to be generated in such a fashion that it does not convey any particular identity of the subject, other than uniqueness within the name space used by the issuer.

The validity dates of the certificate should reflect the contractual arrangements or agreement relating to the allocation of the number resource and not be limited by the validity dates of any superior certificate. The RPKI is capable of supporting validation queries in the present tense, and is not intended to support hypothetical validation questions relating to the past or future tense.

The Authority Information Access and Subject information Access fields represent "backward" and "forward" pointers in the RPKI respectively. The Authority Information Access points to the publication point of the resource certificate where this issuer is the subject, or, in other words, this is a backward pointer to the immediate superior certificate to this certificate. The Subject Information Access points to the location or directory where all published products of the subject are to be published. This corresponds to a forward pointer to all immediate subordinate products that have been issued by the subject of this certificate. It is a constraint within the RPKI that each certificate can only have one superior certificate, and therefore all access methods in the Authority Information Access field must resolve to a publication point for the same RPKI CA certificate.

All Resource Certificates must have the IP Addresses and AS Resources present, and marked as a critical extension. The contents of these extensions correspond exactly to the current state of IP address and AS number allocations from the issuer to the subject.

Any holder of a resource who is in a position to make further allocations of resources to other parties must be in a position to issue Resource Certificates that correspond to these allocations. Similarly, any holder who wishes to use the RPKI to digitally sign an attestation needs to be able to issue an End Entity certificate to perform the digital signing operation. For this reason all issued certificates that correspond to allocations are CA certificates, and each CA certificate is capable of issuing subordinate CA certificates that correspond to further sub-allocations and subordinate EE certificates that correspond to generation of digital signatures on attestations.

EE resource certificates are used in the RPKI to sign "with resources". For example, a resource holder may with to authorize an AS to generate a route announcement for a particular address prefix. In this case the prefix holder would generate an EE resource certificate with the resource extension spanning the set of addresses that match the address prefixes that are the intended subject of the routing authority, and place validity dates in the EE certificate that correspond to the intended validity dates of the routing authority. The authority document would contain the AS that is being authorized in this manner, and a description of the range of prefixes that the prefix holder has authorized, and the EE certificate. The document would be signed by the EE certificate's private key. A relying party could validate the authority to route by checking that the digital signature is correct, that the resources in the EE certificate encompass he prefixes specified in the document, and the EE certificate itself is valid in the context of the RPKI.

The RPKI makes conventional use of Certificate Revocation Lists (CRLs) to control the validity of issued certificates, and every CA certificate in the RPKI must issue a CRL according to the CA's nominated CRL update cycle. A CA certificate may be revoked by an issuing authority for a number of reasons, including key rollover, the reduction in the resource set associated with the certificate's subject, or termination of the resource allocation. To invalidate the authority or attestation that was signed by a given EE certificate, the CA issuing authority that issued the EE certificate also revokes the EE certificate. The CRL only contains the serial number and date or revocation of each revoked unexpired certificate, or an empty list if there are no such revoked certificates at the present time. No revocation reason is specified in this profile, and revocation is an irreversible action for an issuer. It is also a property of this PKI that the key used to sign a CRL must match the key of the certificates it revokes, therefore binding a logical instance of a CA to a single key. Key rollover for a CA is performed by creating a new logical instance of a CA, as described in the next section.

Resource Certificates are intended to be public documents, and all certificates and objects in the RPKI are published in openly accessible repositories. The set of all such repositories forms a complete information space, and it is fundamental to the model of securing the public internet BGP that the entire information space is available. Other uses of the RPKI might permit

use of subsets, such as the single chain from a given end-entity certificate to a Trust Anchor, but routing security is considered against all known publicly routable addresses and AS numbers, and so all known resource certification outcomes must be available.

5. Certificate Management Procedures

Resource Certificates reflect an agreement between a resource registry, acting as an allocation body for IP number resources and a resource recipient. The recipient's resource holdings may change over time, expanding or shrinking as circumstances change, and, from time to time, the agreement between the registry and the recipient may be renewed for a further term, or rescinded. The implication of this characteristic of the RPKI is that resource certificates are not necessarily long-lived objects, and they may need to be re-issued from time to time at all levels in the hierarchy, and that relying parties have a continual need to access the most recently issued set of certificates to perform validation correctly.

There were a number of objectives when designing the certificate management procedures, including the need to avoid unnecessary proliferation of certificates in the RPKI, avoid inflation of the size of CRLs, and to avoid unnecessary dependencies. Dependencies that are best avoided include the consideration that a certificate reissuance event at the upper levels of the certificate hierarchy should not require all subordinate certificates in the sub-hierarchy rooted at the re-issued certificate to be re-issued, for example. The other major objective is that at all times the state of the certificate hierarchy must precisely mirror the state of the resource allocation registries.

To achieve these objectives the RPKI was designed such that each issuer maintains a single current certificate for each subject, where the current certificate encompasses the complete set of resources that have been allocated to that subject.

Further allocations of resources does not cause new certificates to be issued that reflect the incremental change in the allocation, but instead the issuer uses the most recent certificate request from the subject to issue a new certificate where the only changes from the previous are the issuer's serial number and the IP addresses and/or the AS resources fields in the certificate. The previous certificate with the smaller resource set is not revoked at this time, as the certificate is now considered to be incomplete rather than incorrect. As long as the new certificate is published at precisely the same repository publication point as the previous certificate, including the name of the object, then the AIA pointer of all subordinate certificates will point to the new 'current' certificate and relying party's validation of any subordinate certificate will correctly validate. This property means that the allocation of additional resources, a very common event in terms of resource administration, will not generate a cascading requirement for re-issuance of any subordinate certificates.

Where resources are being returned to the registry, then the situation arises that the current certificate is incorrect and should be revoked at the time a new certificate, with the adjusted resource extension, is issued. However the consideration here is that this may affect subordinate certificates, so the procedure prefers a "bottom-up" propagation of the resource return, where the certificate at the lowest level of the hierarchy that includes the returned resources are reissued, and the old certificates revoked, followed by its immediate superior certificate, and so on. This is a non-enforced constraint, which preserves the best possible outcome in routing validation. Re-issuance from the top down is of course always possible.

Key rollover also requires certificate reissuance. Because a logical instance of a CA is bound to a single key, key rollover is performed by creating a new key pair and a new associated logical CA instance, using the same Subject Information Access repository publication point as the previous CA, and requesting a CA certificate from the issuer. The new CA instance can then issue all the subordinate products of the previous CA, overwriting the previous products with the new CA's products in the publication repository. The old CA can then request its issuer to revoke the old CA certificate and the rollover is complete. In this procedure only the immediate subordinate products are affected by the key rollover, and the changes are not propagated any further in the certificate hierarchy than the immediate subordinates of the point of key change.

The common aspect of these certificate management procedures is the reuse of the original certificate request to re-issue subsequent certificates that share the subject's public key and the subject information access field and change the resource extensions and the validity dates, as appropriate. In this respect the certificate request is interpreted as a "standing request" against the CA, and will cause certificates to be issued without further notice to the subject.

Neither X.509 nor PKIX specify a standard mechanism for a CA and a subject to communicate. The specifications provide a standard protocol object in the form of a certificate request, but they do not define standard mechanisms to securely communicate the request, nor to inform the subject of the result of the request. Similarly there is no standard mechanism to request certificate revocation, nor widespread systems to query a CA for the current state of certificate issuance for a given subject.

To assist in structuring the interaction between a CA issuer and a subject, the RPKI framework includes the specification of a certificate management protocol [Huston 2008b]. This protocol is a simple client / server protocol that defines a basic set of interactions that allow a client to request certificate issuance, certificate revocation and status information from a server. In this case the server is a registry and the client is the recipient of resources. The protocol was designed to use existing Internet protocols, and avoid re-inventing the wheel as far as possible. A mix of technologies was chosen which reflected commonly available systems and techniques in the ISP community.

This RPKI certificate provisioning protocol is expressed as a simple request/response interaction, where the client passes a request to the server, and the server generates a corresponding response.

The protocol is implemented as an exchange of messages that are passed over an HTTPS [RFC2818] transport connection that safeguards against

interception and replay attacks. The HTTPS session uses mutually authenticated Transport Layer Security (TLS) [RFC5246]. The TLS keys and associated certificate chain used to validate TLS transactions have been previously communicated between the two entities, as part of an initial configuration. A message exchange commences with the client initiating an HTTP POST with content type of "application/x-rpki", with the message object as the body. The server's response will similarly be the body of the response with a content type of "application/x-rpki".

The content of the POST and the server's response are both a well-formed Cryptographic Message Syntax (CMS) [RFC3852] objects, encoded using the Distinguished Encoding Rules for ASN.1. CMS is used as the signing format to sign the message object. The public part of the signing key and the associated certificate chain that is used to validate the CMS digital signature is communicated between the two entities as part of the initial configuration.

The protocol's request / response interaction is assumed to be reliable, in that all requests will generate a single corresponding response. The protocol requires sequential operation, where the server will not accept a client's request until it has generated and sent a response to the same client's previous request.

These mechanisms are intended to ensure that the communication between the client and server is secure in terms of protection against eavesdropping, replay or attempts to alter the message contents in any way.

The messages themselves are constructed using XML. A common XML wrapper identifies the server and client, allowing both parties to validate that the outer TLS and CMS certificates correspond to the party identified in the message content, ensuring that neither client nor server can masquerade as another party. There are three basic messages in this protocol:

- "**list**", to list all the server's current CA's where the client has allocated resources and the status if all certificates issued by each of the CA's against this client as the subject,
- "**issue**", to pass a certificate issuance request to a nominated server CA, and
- "**revoke**", to instruct a server CA to revoke all of the certificates issued to this client that share a subject public key.

A server may re-issue certificate for a subject at any time following a change in the state of the server's resource allocation database. In such cases the server will reuse the most recently received client's certificate request to generate the new certificate, and will not await a specific request from the client. In order for the client to maintain a synchronized certificate state with the server, the client polls the server at regular intervals and via the "list" command can check if the local state of issued certificates matches the server's state. The "list" command will provide the current issued certificate as part of the response, so the client can resynchronize state with the server within this single command. The "issue" command is used to request certificate issuance, or to commence a rekey operation, and the command carries as its payload a certificate request. The "revoke" command is used to request the server to revoke previously issued certificates, or to complete a rekey operation.

In addition, the specification includes a reference implementation of a RPKI certificate "engine" that implements the supporting functions for this client / server protocol. This engine is intended to bolt onto a registry system and use the client server protocol in client mode to communicate with one or more superior Internet Registries that have allocated resources to this registry to manage, and to operate in server mode in order to communicate with clients who have received resources from this registry. The components of this engine include a store of all current certificates issued by superior Internet Registries, and a store of all current certificates issued to clients of this registry, and all self-issued EE certificates. The engine also keeps a history of all issued certificates in a long term archive. The engine also has a local record of all client identifiers, allowing the engine to associate the TLS and CMS wrappers of a message with a given client identifier, which in turn can be mapped to a given resource recipient in the allocation database. For each superior certificate the local engine has a logical CA state and an associated key pair and a signing subsystem. It also maintains a set of keys for TLS and CMS wrappers of the external communications protocol, as well as holding copies of the public keys of all clients. A conceptual model of this RPKI "engine" is indicated in Figure 3.

The intent of this approach to certificate issuance and management is to make this task one that is entirely automated, and relies on the regular operation of scripts to ensure that the certificates remain synchronized with resource allocation activities.

The “cycle time” of the system is a 6 hour cycle, where each subordinate entity is expected to rendezvous with its immediate superior entity once within the “cycle time”. The 6 hour cycle time was chosen to permit relying parties to have a defined 24 hour daily cycle, with no public facing operation requiring more than 2 cycles to complete. Consideration of the expected depth of allocation and assignment chains, including the IANA, the RIR, NIR, LIR, and tiers of ISPs to end-customers suggested a chain of up to 8 parties was possible. By distributing the load across a day, but requiring at least 4 connections per day, the O(log2(n-1)) properties of the individual parties guarantee complete re-issuance down any chain of up to 8 participants.

A secondary intent was to change as little as possible in the existing resource management framework. A certificate issuing entity (an “Internet Registry”, or “IR”) already operates some form of resource allocation database and an associated set of IR management procedures. There is no intent in the certificate process to redefine this resource allocation database, nor any intent to alter the existing IR management procedures. The adopted approach is to allow the certificate subsystem to mirror the current state of the allocation database, and to issue certificates that are aligned to the allocated resource set. Accordingly the operations of the IR are unaltered and the outcome is a set of changes to the resource allocation database. The certificate system undertakes automated queries of the resource allocation data at regular intervals (once per “cycle time”) and adjusts the certificate state to conform to the resource allocation state.

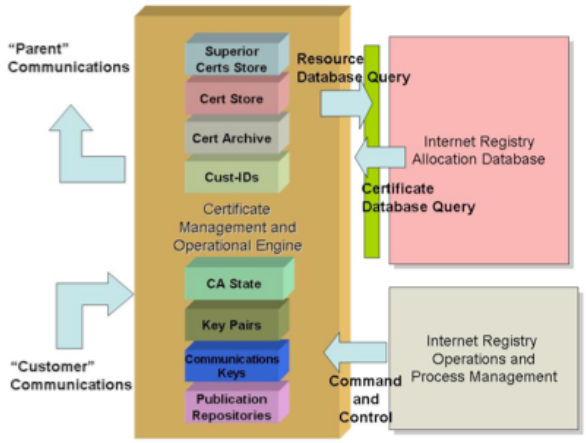

Figure 3. RPKI Engine Components

The RPKI Engine has five external interfaces:

1) An interface to its set of "superior" CA's, who have issued certificates where this entity is the subject. Once per "cycle time" (6 hours) this certificate engine uses a defined protocol to request the current state of issued certificates and allocated resources from the superior CA's. If the superior CA's resource allocation and/or certificate state has changed from the local stored state, then the local system will generate the appropriate set of certificate issuance requests and request from the superior CA's updated certificates that accurately reflect the current resource allocation state. This communication with the superior CA's uses the certificate management protocol described above.
2) An interface to its set of "subordinate" clients, where the CA is in a position to issue resource certificates to the subordinate client by virtue of having a current resource allocation to the client. This is precisely the same interface as that used to communicate to the "superior" CA's, with the essential difference here that in this case the CA is the server that will respond to requests rather than initiating requests. When a client requests the issuance of a certificate, the CA will respond with an issued resource certificate that has resource extensions that align to the allocated resource set. Changes to the resource allocation database are visible to the certificate system which in turn signals these changes to its subordinate clients.
3) An interface to the local resource allocation database, allowing the certificate system to query the resource allocation database to establish the current resource allocation state for any particular client.
4) An interface to the IR management system that allows for the creation of new clients (and new client communication keys) and the removal of clients, and the setting of a set of attributes for each certificate client. This is the "command and control" interface for the system.
5) An interface to an "object signing subsystem" that allows digital objects to be "signed" with specified resources and specified validity dates. This is intended to allow for the signing of objects such as Route Registry objects, Route Origination Attestations, proxy aggregation authorities, route filter requests and upstream route origination requests, for example. In this case "signing" involves the generation of one-off key pair(s), the issuance of one-off end entity certificates for these key pair(s), whose validity determines the validity of the signed object, and the signing of the object with the private key(s).

This RPKI engine is constructed in a modular manner, allowing multiple instances of the engine to be hosted within a single platform, as may be envisaged by the RIR's in hosting a certificate signing service on behalf of their clients.

6. Signed Objects in the RPKI

In itself, a PKI is of little value or utility. The utility of a PKI is best expressed as a means of validation of digitally signed information, and the particular value of the RPKI is not as attestation of identity or role, but a means of validation of the authority to use IP resources. While it is possible to digitally sign any digital artefact, the RPKI system defines a number of "standard" signed objects that have particular meaning in the context of routing security.

The common approach for all signed objects in the RPKI is to use a dedicated EE certificate to sign each object. In this way the issuer of the object can control the object's validity by having the ability to revoke the EE certificate at any time, so there is no need to create additional mechanisms within each signed object to control its validity: existing validation processes suffice.

The first of these objects is the Route Origination Authorization (ROA) [Lepinski 2008a]. A ROA is an explicit authority, created by a prefix holder than authorizes an AS to originate one or more specific route advertisements into the inter-domain routing system. A ROA is a CMS digital object expressed entirely in ASN.1 that contains a list of address prefixes and an AS number. The AS is the specific AS being authorized to originate a route advertisement, and the list of address prefixes are those that the AS is being authorized to originate. The CMS object also includes a EE public key resource certificate for the private key used to sign the ROA, where the IP Address extension in the EE certificate must match the IP address prefixes listed in the ROA's contents. As previously noted, the strong requirements in RPKI certificate issuance and validation is that Internet resources exactly follow allocation and assignment, and are a strict hierarchy. Therefore any valid subset of an RPKI 'branch' in the tree can be used to construct and sign an exactly matching subset of address resources. The EE certificate validation strongly verifies rights to manage the resources, and requiring the resources to match the content of the ASN.1 in the CMS associates these resources exactly with the ROA.

The ROA only conveys a simple authority, and does not convey any routing policy information, nor whether or not the AS holder has consented to actually undertake the routing action. The EE certificate is used to control the validity of the ROA and the CMS wrapper is used to bind the ROA and the EE certificate within a single digital signature in a secure fashion.

If the entire routing domain were to be populated with ROA's, then identification of an invalid route object in

that domain would be directly related to detection of an invalid ROA, or a missing ROA. However in a more likely scenario of partial use of ROA's (i.e., when only some legitimate route originations are authorised in a ROA) the absence of a ROA cannot be interpreted simply as invalid use of an address prefix. Similarly the presence of an invalid ROA does not necessarily invalidate a route object in such a partial deployment scenario. An attacker may deliberately generate an invalid ROA for a route object that is otherwise valid but not described in a valid ROA, and it would be inappropriate for a BGP speaker to discard a route object under such circumstances. In such a partial deployment scenario, what is useful for relying parties of the RPKI is a logical opposite of a ROA, namely an attestation by a prefix holder that the address prefix should not appear in the public routing domain.

This attestation, a Bogon Origin Attestation (BOA) [Hust08b], is similar to a ROA, but with the opposite intent, namely that the listed prefixes and AS numbers should not appear in any route object, and any use of these prefixes or AS's in a route object is unauthorized. Like the ROA, a BOA is a CMS ASN.1 object that contains a list of address prefixes and AS numbers. The CMS object also includes a EE public key resource certificate for the private key used to sign the BOA, where the IP Address extension in the EE certificate must match the IP address prefixes listed in the BOA, and the AS resources extension in the EE certificate must match the AS numbers listed in the BOA.

The combination of ROAs and BOAs allows a relying party to assess the validity of a route assertion from the perspective of the origination information. If a given route object matches exactly the information contained in a ROA whose EE certificate can be validated in the RPKI (a "valid" ROA) then the object can be regarded as a valid origination. But in a scenario of partial deployment of ROAs, all other objects cannot be regarded as simply "invalid". The complete set of situations that could occur are:

- **exact match**: where a valid ROA matches the route object
- **covering match**: where a valid ROA describes an aggregated of the route object with the same origin AS
- **exact mismatch**: where there is no valid ROA for this origin AS, but where one or more valid ROA's exist for the same prefix with differing origin AS's
- **covering mismatch**: where there is no valid ROA for this prefix, but where one or more valid ROA's exist that describes an aggregate of this route object, but with differing origin AS's
- **bogon**: where the route object or origin AS is described in a valid BOA
- **missing**: No ROA or BOA

In the case of a partial deployment scenario for RPKI route attestation objects, where some prefixes are described in ROAs or BOAs and others are not, then the relative ranking of validation outcomes from the highest (most preferred) to the lowest (least preferred) degree of preference is:

1) exact match,
2) covering match,
3) missing,
4) covering mismatch,
5) exact mismatch,
6) bogon.

One way of feeding this information back into BGP is via a BGP LocalPref setting, where validated outcomes are more preferred, missing validation credentials are essentially 'neutral', mismatched outcomes are less preferred and a valid bogon outcome is grounds to reject the route object completely. [Huston 2008d]

While ROAs and BOAs can be used to validate origination information, a related routing security question concerns the validity of the AS path information, that is, the sequence of AS's which runs from the origin, to the recipient BGP speaker.

In attempting to validate an AS path there are a number of potential validation questions. The first, and weakest, question is: are all AS's in the AS Path valid AS's? A slightly stronger validation question is whether all the AS pairs in the AS Path represent AS adjacencies that both AS's are willing to attest to (this question is used in soBGP [White 2003]). A yet stronger question is whether the sequence of AS's in the AS Path represent the actual propagation path of the BGP route object (this question is used in sBGP [Kent 2000]). These differences of degree expose differences of approach in path validation, and also the current uncertainty of the costs of path validation, and what can be achieved online in the routing framework, and what may have to be pre-validated outside of BGP. This is expected to remain an area of intense focus in routing security for some time.

In looking at the AS adjacency question is it possible to construct an object similar in syntax to a ROA, that for a given AS lists all the adjacent AS's. These AS Adjacency attestation Objects (AAO's) are digitally signed objects that provide a means of verifying that an AS has made an attestation that it has a inter-domain routing adjacency with one or more other AS's. In this instance, the RPKI validation relates to the holder of the attesting AS. Therefore the EE certificate in question will relate to the signing AS, not the list of AS's declared to be adjacent.

It would be reasonable for a relying party to infer from a valid AAO that the signing AS has the intent to advertise route objects across this adjacency, and is prepared to learn route objects that are passed to it from the adjacent AS. However, it is noted that an AAO is an asymmetric assertion, where one AS is claiming that an inter-domain routing adjacency with at least one other AS exists, but this claim is not explicitly acknowledged by the remote AS in the context of a single AAO. Relying parties may elect to place greater levels of confidence in the existence of an inter-domain routing adjacency when both AS's have signed and published AAO objects that contain mutual references.

As for the ROA and BOA, the AAO is constructed from CMS and ASN.1. While the AAO is initially presented as a singly signed assertion, it is noted that CMS encompasses multiple signing, and that it would be possible to construct a CMS AAO which included all signing parties across a given list of AS adjacencies. This is held to be an optimization of the basic model, where discrete AAO can be compared to determine the mutual existence of signing across the relationship in both "directions".

It is also possible to apply RPKI digital signatures to a set of IRR objects, using the principles of the RPSS [RFC2725] to guide the decision as to which party should sign the object. RPSL Aut-num objects should be signed by the holder of the AS number, as the inetnum object should be signed by the holder of the IP address prefix. RPSL Route objects require the signature of both the AS holder and the IP address holder, signifying both the granting of an authority by the IP address holder, and the acceptance of this by the AS holder.

The advantage of using RPKI digital signatures in the context of an IRR is that it is then possible to divorce an IRR object from its point of publication, and allow relying parties the ability to validate assertions relating to origination and routing policy with the strong assurance that the IRR objects are authentic and have not been altered in any way. This is currently an area of active study, and consideration is being given to an approach that conserves much of the semantics of the fields in the IRR objects, and using a DKIM-style signature to digitally sign a subset of the IRR fields that require authentication [RFC4871], or revise the Routing Policy Specification Language [RFC2622] to remove all parts that refer to authentication and access control and substitute RPKI digital signatures in their place.

This approach would directly address the current weakness of the IRR dependency on the provenance of publication of IRR objects. Instead of weak trust in a "source" of IRR objects, a strong, and testable trust in the signatures can provide far greater assurance for relying parties that the IRR objects accurately represent the intentions and permissions of the object's maintainer.

7. Publication of RPKI Objects

To validate attestations made in the context of the RPKI, relying parties need access to the complete set of current Resource Certificates, CRLs, and signed objects that collectively define the RPKI.

Each issuer of a certificate, CRL or a signed object makes it available for download to replying parties through the publication of the object in a RPKI repository. The repository system is the ‘clearing-house’ for all signed objects that must be globally accessible to relying parties. When certificates, CRLs and signed objects are created, they are uploaded to a specified repository ‘publication point’, from whence they can be downloaded for use by relying parties.

The RPKI repository system is comprised of multiple repository publication points. Each repository publication point is uniquely associated with a single CA (or more precisely one or more CA's that are associated with the same resource allocation and differ in their key values) or a single EE certificate. The certificate's Subject Information Authority (SIA) extension provides a URI that references this repository publication point and a supported access mechanism [Huston 2008f]. A given publication point may of course lie on the same server, URI may share common parts, but it is not a requirement that any given publication point lie on the same physical or logical path as its parent, or children. The decision is ‘local’ at that level of the repository ‘tree’.

The unique characteristic of the RPKI repository system is the addition of a manifest to the repository [Austein 2008]. Because the repository access mechanism is unprotected it is possible for the access to be the subject of attack. Because the repository contains digitally signed RPKI objects any attempt to alter retrieved objects, or add bogus objects to the retrieval operation can be detected, because of a validation failure on the altered or bogus object. However the retrieval operation is susceptible to the deliberate omission of an object, and to the substitution of "stale" objects in place of current objects (a "stale" object is one that has been superseded by more recent information, but has not been explicitly revoked as it is not invalid per se).

For the RPKI this is a critical problem, as the intended use of the PKI, (that of routing security) is subtly different from other PKI's. For a conventional PKI a relying party may be presented with a small subset of the signed material, and wishes to validate this small subset of information. Routing presents a different problem, in that every BGP speaker could be considered a relying party and every relying party has a requirement to validate the complete routing information set. In other words each relying party is placed in the position of having to validate the major proportion of the PKI subject space as a continual task. In this case, the completeness of the information available from the RPKI repository system is a critical factor in the effectiveness of the PKI, and attacks on the completeness of the RPKI information can have consequences in terms of the integrity of the routing system.

To address this vulnerability the RPKI uses the concept of a manifest in every repository publication point. A manifest is a CMS/ASN.1 signed object that lists of all of the other signed objects issued by the authority responsible for a publication point in the repository system. For each certificate, Certificate Revocation List (CRL), or other signed object published by the authority, the manifest contains both the name of the file containing the object, and a cryptographically strong hash of the file content. Manifests allow a Relying Party to obtain sufficient information to detect whether the retrieval of objects from an RPKI repository has been compromised by unauthorized object removal, or by the substitution of "stale" versions of objects. Manifests are designed to be used both for Certification Authority (CA) publication points in repositories, that contain subordinate certificates, CRLs and other signed objects, and End Entity (EE) publication points in repositories that contain signed objects.

In terms of the use of scheduled update times as part of the signed data, manifests are modeled on CRLs, as the issues involved in detecting stale manifests, and detection of potential attacks using manifest replays, etc are similar to those for CRLs. Manifests also contain a list of file name and hash value pairs, corresponding to all the other objects held at this publication point. The manifest is signed with an EE certificate issued by the authority responsible for publication at this publication point, and are structured as a CMS object [RFC3852]. Manifests are a qualitatively novel addition to the PKI information model, considered for some time in general, but not included in current PKI standardization.

9. Use of the RPKI

Resource Certificates and the associated RPKI represent a major part of any effort to construct a secure inter-domain routing framework. An RPKI, even partially populated with signed information allows BGP speakers to make preferential selections to use routing information where the IP address block and the AS numbers being used are recognised as valid to use, and that the parties using these IP addresses and AS numbers are properly authorized to so do. The RPKI can also identify instances of unauthorised use of IP addresses and attempts to hijack routes.

However, the RPKI represents only one part of a larger framework of securing inter-domain routing, and the next step is that of applying the RPKI to the local BGP processing framework. There is also the need to move beyond validation of route origination and look at the associated issue of validation of the AS Path, and potentially the most challenging task of attempting to validate whether the initial forwarding hop associated with an offered route object actually represents the correct first hop along a useable forwarding path for packets to reach the network destination.

The issues here include not only a consideration of what can be secured and validated, but issues of scalability and efficiency in terms of deployment cost. The various approaches to routing security studied so far offer a wide variety of outcomes in terms of the amount of routing information that is validated, the level of trust that can be placed in a validation outcome and the overheads of generating and validating digital signatures on routing information. The next step appears to include the task of establishing an appropriate balance between the overheads of operating the security framework and the extent to which efforts to disrupt the routing system can be successfully deflected.

A characteristic of the RPKI which has not been noted thus far is that to all intents and purposes the PKI certificates at both EE and CA level are regarded as short-lived artefacts in that issuance, and re-issuance is normal and expected. Most PKIs in the realms of identity or attribute certification rely on relatively long-lived certificates throughout the information model, to reduce the overhead of information management. Given the highly dynamic nature of routing (where it is not uncommon for several significant updates per day to be made to an ISPs routing model, either locally or globally) and its criticality to the stability of the Internet, the decision was made to not emphasise the life of any level of the RPKI, but instead to require active management of current state, and frequent re-certification of the EE certificates associated with manifests and signed objects.

While this high degree of churn is a downside of the model, the number of participating entities is still high: it is expected that over time, a significant number of participants in Internet address management, of the order 10,000 to 20,000 entities worldwide, will routinely receive CA certificates from their superior address managing authorities, and therefore have repository publication obligations in the public view. Likewise, the number of relying parties is expected to be high, as the worldwide BGP routing community adopt the process of checks on ROA, BOA and AAO, and related objects from the IRR.

The RPKI has been designed as a robust, simple framework. As far as possible existing technologies and processes have been exploited, reflecting the conservatism of the routing community and the difficulty in securing rapid widespread adoption of novel technologies.

Bibliography

[Austein 2008] Manifests for the Resource Public Key Infrastructure, R. Austein et.al., work in progress, Internet Draft, draft-ietf-sidr-rpki-manifests--02.txt, August 2008.

[Bates 1998] DNS-based NLRI origin AS verification in BGP, T. Bates, R. Bush, T. Li and Y. Rekhter, Expired Internet Draft, draft-bates-bgp4-nlri-orig-verif-00.txt, July 1998.

[Chakrabarti 2002] Internet infrastructure security: a taxonomy, A. Chakrabarti, G. Manimaran, IEEE Networks, Vol. 16, No 6., pp 13-21, November 2002.

[Goodell 2003] Working Around BGP: An Incremental Approach to Improving Security and Accuracy of Interdomain Routing, G. Goodell, W. Aiello, T. Griffin, J. Ioannidis and P. McDaniel, Proc. of Internet Society Symposium on Network and Distributed System Security (NDSS'03), February 2003.

[Huston 2005] Securing Routing – An ISP's Perspective, G. Huston, ISP Column, February 2005. (http://www.potaroo.net/ispcol/2005-02/route-sec.html)

[Huston 2008a] A Profile for X.509 PKIX Resource Certificates, G. Huston, G. Michaelson, R. Loomans, work in progress, Internet Draft, draft-ietf-sidr-res-certs-12.txt, September 2008.

[Huston 2008b] A Protocol for Provisioning Resource Certificates, G. Huston, R. loomans, B. Ellacot, R. Austein, work in progress, Internet Draft, draft-ietf-sidr-rescerts-provisioning-03.txt, August 2008.

[Huston 2008c] A Profile for Bogon Origin Attestations (BOAs), G. Huston, T. Manderson, G. michaelson, work in progress, Internet Draft, draft-ietf-sidr-bogons-00.tx, August 2008.

[Huston 2008d] Validation of Route Origination in BGP using the Resource Certificate PKI, G. Huston, G. Michaelson, work in progress, Internet Draft, draft-ietf-sidr-roa-validation-00.txt, August 2008.

[Huston 2008e] A Profile for AS Adjacency Attestation Objects, G. Huston, G. Michaelson, work in progress, September 2008.

[Huston 2008f] A Profile for Resource Certificate Repository Structure, G. Huston, G. Michaelson, R. Loomans, work in progress, Internet Draft, draft-ietf-sidr-repos-struct-00.txt, August 2008.

[Kaminsky 2008] It's the End of the Cache as We Know It, D. Kaminsky, presentation to DEFCON16, August 2008 (http://www.defcon.org)

[Kent 2000] Secure Border Gateway Protocol (S-BGP), S. Kent, C. Lynn, and K. Seo, IEEE Journal on Selected Areas in Communications, Vol. 18, No.. 4,pp 582-592, April 2000.

[Lepinski 2008a] A Profile for Route Origin Authorizations (ROAs), M. Lepinski, S. Kent, D. Kong, work in progress, Internet Draft, draft-ietf-sidr-roa-format-03.txt, July 2008.

[Lepinski 2008b] An Infrastructure to Support Secure Internet Routing, M. Lepinski, S. Kent, work in progress, Internet Draft, draft-ietf-sidr-arch-03.txt, February 2008.

[Oorschot 2007] On Interdomain Routing Security and Pretty Secure BGP (psBGP), P. van Oorschot, T. Wan and E. Kranakis, ACM Transactions on Information and System Security, Vol. 10, No. 3, July 2007.

[Steenbergen 2008] Examining the Validity of IRR Data, R. Steenbergen, Presentation to NANOG 44, October 2008. (http://www.nanog.net/meetings/nanog44/presentations/Tuesday/RAS_irrdata_N44.pdf)

[White 2003] Securing BGP through secure origin BGP, R. White, Internet Protocol Journal, Vol. 6, No. 3, September 2003.

[RFC2622] Routing Policy Specification Language (RPSL), C. Alaettinoglu, et al, RFC2622, June 1999.

[RFC2725] Routing Policy System Security, C. Villamizar et al, RFC2725, December 1999.

[RFC2818] HTTP Over TLS, E. Rescorla, RFC2622, May 2000.

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

[RFC3852] Cryptographic Message Syntax (CMS), R. Housley, RFC3852, July 2004.

[RFC3912] WHOIS Protocol Specification, L. Daigle, RFC3912, September 2004.

[RFC4033] DNS Security Introduction and Requirements, R. Arends et al, RFC4033, March 2005.

[RFC4271] A Border Gateway Protocol 4 (BGP-4), Y. Rekhter, T. Li and S. Hares, RFC4271, January 2006.

[RFC4871] DomainKeys Identified Mail (DKIM) Signatures, E. Allman et al, RFC4871, May 2007.

[RFC5246] The Transport Layer Security (TLS) Protocol Version 1.2, T. Dierks, E. Rescorla, RFC5246, August 2008.

[RFC5280] Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile, D. Cooper et al, RFC5280, May 2008.

[X.509] Recommendation X.509: The Directory Authentication Framework, ITU-T, 2000.