Internet Draft
draft-ietf-dnsop-interim-signed-root-01.txt
February 2003
Expires in six months

Johan Ihrén
Autonomica

An Interim Scheme for Signing the Public DNS Root

Status of this Memo

Abstract

This memo documents a proposed mechanism for a first stage of a transition from an unsigned DNS root to a signed root, such that the data in the root zone is accompanied by DNSSEC signatures to allow validation.

The underlying reason for signing the root zone is to be able to provide a more secure DNS hierarchy, where it is possible to distinguish false answers from correct answers.

For the special case of the DNS root zone, an interim scheme is proposed. This scheme is mostly aimed at securing the root zone itself for technical and operational reasons, and to give operational experience of DNSSEC.

1. Terminology

The key words "MUST", "SHALL", "REQUIRED", "SHOULD", "RECOMMENDED", and "MAY" in this document are to be interpreted as described in RFC 2119.

The term "zone" refers to the unit of administrative control in the Domain Name System. "Name server" denotes a DNS name server that is authoritative (i.e. knows all there is to know) for a DNS zone, typically the root zone. A "resolver", finally, is a DNS "client", i.e. an entity that sends DNS queries to authoritative nameservers and interpret the results.

2. Motivation for signing the DNS root

In the special case of the root zone there are very strong reasons to take a slow and conservative approach to any changes with operational impact. Signing the root is such a change.

DNSSEC[RFC2535, RFC3090] has been in development for a number of years now and still has not reached the point where the last flag day is behind us.

However, during the years of DNSSEC development and refinement [RFC2930, RFC3007, RFC3008, RFC3110, RFC3225, RFC3226, AD-secure, Opt-in, Wild-card-optimize], the Internet has matured and more and more businesses and other organizations have become dependent on the stability and constant availability of the Internet.

It is therefore prudent to do everything in our power to ensure that the DNS infrastructure works as well as possible and, when appropriate and possible, adding enhancements and functionality.

The time is now right for yet another step of improvement by signing the root zone. By doing that any Internet user that so wishes will obtain the ability of verifying responses received from the root nameservers.

Since this is new operational ground the objective is not maximum security but rather an "Internet-wide" controlled experiment with a signed root zone, where the trade off is that we utilize the fact that there are operators in place that can help even though they are not sufficiently staffed to do off-line signing in a 24x7 mode. For this reason it is fully possible to even do automatic signing, since the purpose is to ensure that DNSSEC works operationally with a signed root zone and gain experience from the exercise.

It should be pointed out, however, that the experimental part is only the added DNSSEC records. The traditional records in the root zone remain unchanged and the new records will only be visible to

DNSSEC-aware resolvers that explicitly ask to see these new records.

2.1. Motivation for signing the root zone now

The reason DNSSEC is not yet widely deployable is that the problem of key management remains unsolved, especially where communication between the zone administrators for a parent zone and child zone is required.

However, during the past year a solution to this problem has been found (in the form of a new record type, DS aka Delegation Signer) [DS], discussed, implemented and tested. Therefore, it is finally reasonable to consider DNSSEC deployment to be a real possibility within the next 12 months.

In the case of the root zone the particular importance of managing the transition with zero operational mistakes is a strong reason to separate the signing of the zone itself, with the associated key management issues, from the future management of signed delegations (of top level domains).

Although support for Delegation Signer has been implemented and tested it is not yet generally available except experimentally. For this reason signed delegations for productions zones will have to wait a bit longer. Furthermore, it will take some time to ensure that the entire RSS aka Root Server System is capable of supporting the protocol changes that accompany the new support for Delegation Signer.

By signing now it will be possible to work through the operational issues with signing the zone itself without at the same time having to manage the additional complexity of signed delegations. Also, by explicitly not supporting any signed delegations, it is also possible to recover in a graceful way should new operational problems turn up.

Because of these operational and technical issues there is a "window of opportunity" to sign the root zone before the status of implementation of "full DNSSEC", including Delegation Signer support, change to "generally available", thereby increasing the pressure for signed delegations from the root zone.

3. Trust in DNSSEC Keys

In the end the trust that a validating resolver will be able to put in a key that it cannot validate within DNSSEC will have to be a function of

* trust in the key issuer

* trust in the distribution method

* trust in extra, out-of-band verification

For any security apex, i.e. a node in the DNS hierarchy that issues out-of-band "trusted keys", it is of critical importance that this function produces a positive result (i.e. the resolver gains sufficient confidence in the keys to decide to trust them). The particular case of the root zone is no different, although possibly it is more crucial than many other security apexes.

To ensure that the resulting trust is maximized it is necessary to work with all the parameters that are available. In the case of the key issuer there is the possibility of chosing a key issuer that has a large "trust base" (i.e. is already trusted by a large fraction of the resolver population). However, it is also possible to expand the aggregated trust base by using multiple simultaneous key issuers, as described in [Threshold-Validation].

Furthermore, with multiple trusted keys it will be possible for a validating resolver to optimize for the "right compromise" between

* maximum security (as expressed by all available signatures by all available keys verifying correctly

* maximum redundancy (as in the DNS lookup being validated if there is any signature by any trusted key available)

Without multiple, independent trusted keys the rollover operation will always be a dangerous operation where it is likely that some fraction of the resolver population lose their ability to verify lookups (and hence start to fail hard). I.e. the validating resolver will be forced to adopt the "maximum security" policy, since there is no alternative.

With multiple, independent trusted keys, however, it is possible to design for improved redundancy by trusting lookups that are only validated against a subset of the available keys. This will make rollovers much less risky to the extent that it will be possible to "survive" even emergency rollovers without any immediate configuration chagnes in the validating resolver.

4. Interim signing of the root zone

At present all the authoritative servers for the root zone pull the zone from a primary master. I.e. any changes at the primary master will propagate via NOTIFY[RFC1996] and subsequent AXFR/IXFR[RFC1995, AXFR-clarify] to the slave servers.

Between the primary master and the slaves transactions are signed with TSIG[RFC2845] signatures. This mechanism is already in place, and there is operational experience with periodic key rollover of the TSIG keys.

4.1. Design philosophy

By introducing a signing step into the distribution of the zone file complexity is increased. To avoid introducing new single points of failure it is therefore important to ensure that any added or changed steps are as redundant as possible.

The assumption is that the operators of the root servers are trusted and that consistency of the zone among the root servers is a crucial property that MUST be preserved in emergencies.

To ensure that consistency is maintained new single points of failure SHOULD NOT be introduced by the signing step. Therefore a structure where several parties have the ability to sign the zone is proposed. Furthermore, such a design helps avoid any individual party becoming a de facto single zone signing authority.

4.2. Proposed new management structure for the root zone

Taking into account the already existing infrastructure for management of TSIG keys and rollover of these keys the following structure is proposed:

* Day-to-day signing of the root zone is performed by a subgroup of the root server operators referred to as "signing operators". For this task individual, per-operator, Zone Signing Keys, ZSKs, are used.

* The set of Zone Signing Keys are signed by several Key Signing Keys, KSKs, at any particular time. The public part of KSKs in use have to be statically configured as "trusted keys" in all resolvers that want to be able to perform validation of responses.

* Key rollover, the operation when an old KSK is exchanged for a new KSK, is managed with minimal overlap and on a frequent basis of no less than three times per year per KSK. The rollover schedule is chosen to be frequent enough to not make long-term trust possible while being low enough to not become operationally infeasible.

4.2.1. Management and distribution of the zone file

The present, unsigned zone is published by the official slaves, the "root nameservers", transferring the zone securely from a primary master and subsequently using the data to respond to queries. This mechanism is changed into:

* The unsigned root zone is transferred securely from the primary master to a set of "signing primaries" managed by the operators participating in signing the zone. This is done via the traditional mechanisms NOTIFY + AXFR/IXFR + TSIG.

* The root nameservers change their configuration so that they replace the present, single, primary master as the source of the zone with a set of multiple possible "signing primaries" as masters that provide the signed zone.

* When a new, unsigned zone, is published by the primary master it will automatically be transferred to the pre-signing servers. The responsible operator will then sign the new zone and publish it from his signing primary. Since the serial number is higher than what the official root nameservers presently have the official root nameservers will all transfer the zone from this source (because of the redundant configuration with multiple possible masters for the signed zone).

* The operators that participate in signing rotate the signing responsibility among themselves. Should the responsible operator be unable to do this for any reason then any of the others are able to do the signing instead. Traceability is maintained since the zone signing keys are individual.

* To avoid having several "backup signing operators" possibly sign at exactly the same time backups are allocated "time windows" within which they are allowed to publish a signed zone.

  With N signers, each signer is assigned a unique number R in [1..N].

  $$T = 2*k*((S-R) \bmod N)$$

  T is time to wait in seconds, S current serial number. The length of the window is k, thereby separating each signing window with an interval where signing is not allowed.

  The constant k is used to create sufficient separation of the signers with respect to time used to sign and time needed to distribute the zone. A reasonable value for k would be in the order of 1800 seconds.

* Because the digital signatures in the signed root zone MUST NOT expire it is necessary to ensure that the unsigned zone does

change sufficiently often to cause new signing to occur within the validity period of the old signatures. This is most easily accomplished by a dummy update that only increments the serial number in the SOA record.

This new requirement will not cause any operational problems, since in fact the root zone is maintained this way since several years back.

4.2.2. Management of the Key Signing Keys

Key Signing Keys, KSKs, are periodically issued by a several "Key Signing Key Holders", KSK holders, individually. These KSKs consist of a private part that should always be kept secret and a public part that should be published as widely as possible since it will be used as a "trusted key" in resolver configurations.

The public part of each KSK should be included in the keyset for "." as described in [Threshold-Validation]. I.e. the keyset will be the union of the public parts of all KSKs and all ZSKs, Zone Signing Keys.

* Each KSK holder should generate a sequence of KSKs where the public parts will be used to include in the keyset for "." and the private parts will be used for signing the keyset.

* Each KSK holder should, on request from the signing operators, send in the public part of the KSK. The union of the public parts of KSKs and the corresponding public parts of ZSKs, as collected by the signing operators, constitute a "keyset".

* Each KSK holder should, on request from the signing operators, sign the complete keyset with the private part of the associated KSK and send in the resulting signature to the signing operators for inclusion in the signed zone.

4.2.3. Management of the Zone Signing Keys and the keyset signatures

A subgroup of the root operators that choose to participate in signing the zone each hold an individual "Zone Signing Key", ZSK. The subgroup of operators may include all operators, but that is not necessary as long as a sufficient number to achieve redundancy in "signing ability" participates.

* The complete keyset (i.e. all the public parts of KSKs and ZSKs) should be signed by each KSK holder individually, generating a new signature for the keyset which is sent back to the signing operators via an out-of-band mechanism.

* The signing operators should verify each received signature against the corresponding key in the keyset and, unless invalid, accept the signature into the set of signatures associated with this keyset as described in [Threshold-Validation].

* The signing operators should include one of the keysets together with all the KSK signatures in the zone file according to an agreed upon rotation schedule.

4.2.4. Management of key rollover

The Key Signing Keys should, for this interim scheme, be relatively short-lived and rolled over frequently. The direct intent is that it should not be possible to put long term trust in the keys. Therefore, by design, every resolver that chooses to configure these as "trusted keys" (to be able to validate lookups) will have to change the configuration periodically.

This is, after all, only an interim method of root zone signing.

* Key rollover is executed by changing from one signed keyset to another. I.e. from a keyset signed by one set of KSKs to a keyset signed by a partially different set of KSKs. By not rolling all the KSKs at the same time redundancy is improved.

* Technically the signing operators are able to initiate key rollover, although except for the case of a suspected key compromise (with subsequent immediate key rollover) this ability should only be used according to an established schedule.

* New Key Signing Keys will be published as widely as possible, however exactly how and where to publish the keys is by itself an area where it is necessary to acquire more experience. Especially for the case of individual hosts and other devices this will be a significant issue to deal with.

* Since the keys expire within a few months the aim is to make it as difficult as possible to configure an interim key and then forget about it long enough to still trust an interim key when a long term design for signing the root zone emerges.

4.2.5. The role of the KSK holder

With multiple KSKs it is possible to have multiple individual KSK holders. Each will perform the role of authenticating the identity of the signing operators, through the act of signing the keyset that includes the operator Zone Signing Keys.

The chain of authority that defines editorial control over the zone

contents is entirely separate from this and is in no way affected.

I.e. the KSK holder is only asserting identity of the holders of ZSKs and does not in any way take part in issues regarding zone contents. In this respect the role of a KSK holder is much like that of a public notary or a Certificate Authority.

The primary function that the KSK holder is performing is adding trust to the authenticity of the Zone Signing Keys and this trust has to be propagated down to validating resolvers. Therefore an obvious key characteristic of a KSK holder is that it should already be trusted by as large a fraction of the "resolver population" as possible. In this document that fraction is referred to as the "trust base" of a KSK holder.

The key characteristics of a KSK holder will be entities that

* already are trusted by some part of the "resolver population", i.e. have a "trust base"

* are multiple entities that complement each other (so that the aggregated "trust base" grows)

* are willing to help work on methods for distributing their trusted keys to the resolvers (hard problem)

The requirement on the individual KSK holders is that they must be able to

* establish a secure out-of-band communication path in collaboration with the signing operators which will be used for authenticated exchange of the unsigned keyset and generated signatures

* periodically generate strong keys using a good random number generator

* manage their keys (i.e. use them for signing the operator keyset and keeping the private key appropriately secret)

4.2.5.6. Suggestions for KSK holders

Regional Internet Registries, RIRs, are suggested to be one suitable choice of KSK holders. However, since every KSK holder will act on its own there is no requirement that all RIRs participate, although more than one would be good. Other suitable KSK holders may exist and as long as the requirements are met more would be better within the packe size limitations that are discussed in [Threshold-Validation].

The basis of the suggestion of RIRs is

* their neutrality

* their proven record of service to the Internet community

* that they don't have the domain name system as their primary field of activity

* their geographical diversity

* the fact that they are, by definition, not a single entity, but rather a collective of organizations.

5. Risk Analysis

A change in the management of the root zone is always a risk. But that is all the more reason to do it carefully and after due consideration, rather than as a result of an immediate and urgent need. The gains of a signed root zone have to be judged against the added complexity of the signing step.

However, added complexity, in one form or another, is basically unavoidable. It is possible to argue for or against implementation details, but in the end the benefits of a signed root will have to be compared against some amount of added complexity. If the cost or risk of this complexity is deemed to be too high, then it is not possible to sign the DNS root zone. If that is the conclusion; then it is obviously important to reach it as soon as possible.

The same is true for the need for operational experience with a signed root zone. There is no method of acquiring this experience except by signing the root zone, so that is what is being proposed.

Some identified scenarios:

5.1. What will the consequences of a signed root zone be for old resolver software?

Nameservers that are authoritative for signed zones only give out these signatures when explicitly asked for them. Therefore, the presence of signatures in the root zone will not have any impact at all on the majority of resolvers on the Internet that does not validate lookups.

5.2. What happens if a signing operator fails to sign the zone on time?

Literally nothing. I.e. the root zone that is published will be the version prior to the last change. This is not believed to be a major problem, since all changes to the data in the root zone are characterized by long overlaps in time. Furthermore every change is preceded by an administrative process that takes several days or even weeks. Because of this, a failure to install a new version of the root zone for even so long as a day will not noticeably change the turn-around time for changes in the root zone.

5.3. What happens if several signing operators by mistake sign the same version?

Literally nothing. One signing operator will be first, according to the mechanism designed to separate the different backup signing operators described in 3.3.1. The signed zone published by this operator will be the version of the signed root zone that all the root nameservers pick up.

When the second signing operator signs the zone the serial number in the zone will be unchanged, and therefore the root nameservers will not pick this zone up but instead stay with the first version.

5.4. What happens if the unsigned zone is compromised between the primary master and the signing primaries?

This is basically the same threat as the zone being compromised between the primary master and the root servers in the traditional unsigned case. To guard against this possibility every zone transfer is already protected by a TSIG signature.

Because of this the root zone file cannot get transferred to the signing primaries unless the transaction signature matches, thereby proving that the zone contents are uncompromised. The consequence is that if the zone transfers are somehow compromised in transit, they will not get signed and therefore the published root zone will remain the signed version of the last uncompromised transfer.

5.5. What are the security implications for the new "signing primaries"? Will they not have to be as secure as the primary master is now?

Yes. However, the entire root server system is based upon trust, i.e. the entire Internet is trusting the present root server system because there is no alternative to doing that. This proposal is not about changing the need for trust in the root server system. It is about providing a method of determining authenticity of data, something that is presently missing.

The root operators are already trusted to provide a root server system based upon secure servers lacking validation mechanisms. It is therefore a bit difficult to argue for not trusting them to provide an improved system that adds exactly the lacking validation mechanisms on a basis of not trusting them to secure the servers involved. In both cases trust is involved, the difference is that in the latter case validation is possible.

Furthermore, the proposed signing primaries will not need to have publicly known addresses (just as the present primary master is not publicly known), nor will they need to be able to communicate with anyone outside the well defined set of servers that are part of the root server system. Because of this it will be significantly easier to secure the signing primaries than the already present task of securing the DNS root nameservers.

5.6. What happens if a Zone Signing Key is compromised?

If this happens it is necessary to do an emergency rollover of the keyset that includes the compromised key. I.e. the old keyset (and its signatures) is replaced by a new keyset containing new ZSKs but the same, uncompromised, KSKs and its signatures. Then the root zone is re-signed using one of the new Zone Signing Keys.

This problem is not unique to a signed root zone, it is inherent in any activity involving cryptographic keys.

Also note that there will be no need to change any configuration in the resolver end. The rollover is an entirely atomic operation inside the management structure of the root zone.

5.7. What happens if a Key Signing Key is compromised?

Depending on the trust model used by individual validating resolvers one of two things will happen.

Resolvers that through local policy have chosen to trust this KSK alone will need to change their configuration to instead trust other KSKs, either available from other KSK holders or a new trusted key made available by the same KSK holder that held the compromised key.

Resolvers that have chosen to require multiple signatures by KSKs for the root zone will not have to do any emergency key rollover at all, since validation of lookups will still work, based upon the integrity of the other trusted keys that have not been compromised.

In the management end it is necessary to do an emergency rollover of the keyset including the compromised key and the suggested

method is by switching to a keyset that only changes the compromised KSK and the ZSKs and keeps all other KSKs. Such keysets should be prepared and signed in advance.

The new signed keyset is added to the root zone and then the zone is re-signed using one of the new Zone Signing Keys. In this case, since a Key Signing Key is changed, every resolver that choose to trust that KSK holder will over time have to configure the new key to be able to validate lookups.

This problem is not unique to a signed root zone, it is inherent in any activity involving cryptographic keys.

5.8. Does the out-of-band communication between the signing operators and the RIRs holding the key-signing keys add a new failure mode?

Yes. The traditional DNSSEC design assumes that (for an arbitrary zone, not particularly for the root zone) the key-signing key is managed by the same entity that manages the operator keys and hence no communication issue exists.

In this case it is important that the signing operators do not hold the responsibility for the key-signing keys. The root server operators should only act as a "publishing service" for the root zone contents, not as the entity that verifies correctness of the published data.

However, apart from established secure methods of out-of-band communication being available, there is also the very real possibility of managing these exchanges with actual eye-to-eye contact. Representatives for the RIRs and the root server operators already meet on a regular basis during IETF meetings and the different RIR meetings.

6. Security Considerations

Signing the DNS root zone is all about security. A signed root zone may aid applications and organizations all over the Internet in improving their security by enabling validation of DNS lookups.

Signing the root zone does add a new management step and therefore it is important to ensure that possible failures in this management step does not leave the published root zone in a state that is actually worse than the original unsigned state.

The various failure modes that have been identified all show this property of not being any worse than the unsigned case.

There is however one scenario that changes drastically with the

proposed distributed signing scheme and that is the consequences of one signing operator intentionally changing the contents of the root zone prior to the actual signing. In the unsigned case this will cause the root zone to become inconsistent (i.e. the responses vary depending upon which server it comes from), while in the signed case the root zone will remain consistent but with erroneous data in it.

It is my belief (this is impossible to prove) that the risk of human error among the operators, however small, is still far greater than the risk of willful misconduct. Therefore, the proposed design that enforces consistency among the roots, is actually also an improvement in stability and security.

Se further section 4 for a more complete risk analysis.

7. IANA Considerations

NONE.

8. References

8.1. Normative.

[RFC2535] Domain Name System Security Extensions. D. Eastlake. March 1999.

[RFC3090] DNS Security Extension Clarification on Zone Status. E. Lewis. March 2001.

[RFC1995] Incremental Zone Transfer in DNS. M. Ohta. August 1996.

[RFC1996] A Mechanism for Prompt Notification of Zone Changes (DNS NOTIFY). P. Vixie. August 1996.

[RFC2845] Secret Key Transaction Authentication for DNS (TSIG). P. Vixie, O. Gudmundsson, D. Eastlake, B. Wellington. May 2000.

8.2. Informative.

[RFC2930] Secret Key Establishment for DNS (TKEY RR). D. Eastlake. September 2000.

[RFC3007] Secure Domain Name System (DNS) Dynamic Update. B. Wellington. November 2000.

[RFC3008] Domain Name System Security (DNSSEC) Signing Authority. B. Wellington. November 2000.

[RFC3110] RSA/SHA-1 SIGs and RSA KEYs in the Domain Name System (DNS). D. Eastlake 3rd. May 2001.

[RFC3225] Indicating Resolver Support of DNSSEC. D. Conrad. December 2001.

[RFC3226] DNSSEC and IPv6 A6 aware server/resolver message size requirements. O. Gudmundsson. December 2001.

[Delegation-Signer] Delegation Signer Resource Record. O. Gudmundsson. October 2002. Work In Progress.

[AXFR-clarify] draft-ietf-dnsext-axfr-clarify-NN.txt DNS Zone Transfer Protocol Clarifications. A. Gustafsson. March 2002. Work In Progress.

[AD-secure] draft-ietf-dnsext-ad-is-secure-NN.txt Redefinition of DNS AD bit. B. Wellington, O Gudmundsson. June 2002. Work In Progress.

[Opt-In] draft-ietf-dnsext-dnssec-opt-in-NN.txt DNSSEC Opt-In R. Arends, M Kosters, D. Blacka. June 2002. Work In Progress.

[Wild-card-optimize] draft-olaf-dnsext-dnssec-wildcard-optimization-NN.txt DNSSEC Wildcard optimization. O. Kolkman, J. Ihren, R. Arends. September 2002. Work In Progress.

[Threshold-Validation] draft-ihren-dnsop-threshold-validation-00.txt Threshold validation: A Mechanism for Improved Trust and Redundancy for DNSSEC Keys. J. Ihren. February 2003. Work In Progress.

9. Acknowledgments.

To help me produce this document I have received lots and lots of comments from many people around the Internet with suggestions for lots and lots sorely needed improvements. Among the folks who helped out are, in no particular order: Paul Vixie, Bill Manning, Ôlafur Gumundsson, Rob Austein, Patrik F□ltstr÷m, Olaf Kolkman, Harald Alvestrand, Suzanne Woolf, John M. Brown, Lars-Johan Liman and M□ns Nilsson.

10. Authors' Address

Johan Ihr□n
Autonomica AB

Bellmansgatan 30
SE-118 47 Stockholm, Sweden
johani@autonomica.se