Domain Name System Operations
Internet-Draft
Intended status: Informational
Expires: April 26, 2015

P. Ebersman
Comcast
C. Griffiths
Dyn
W. Kumari
Google
J. Livingood
Comcast
R. Weber
Nominum

October 23, 2014

Definition and Use of DNSSEC Negative Trust Anchors
draft-livingood-dnsop-negative-trust-anchors-01

Abstract

DNS Security Extensions (DNSSEC) is now entering widespread deployment. However, domain signing tools and processes are not yet as mature and reliable as those for non-DNSSEC-related domain administration tools and processes. Negative Trust Anchors (described in this document) can be used to mitigate DNSSEC validation failures.

[Editor note: This document was originally draft-livingood-negative-trust-anchors-07 - renamved at the request of the DNSOP chairs]

Status of This Memo

This Internet-Draft is submitted in full conformance with the provisions of BCP 78 and BCP 79.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at http://datatracker.ietf.org/drafts/current/.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

This Internet-Draft will expire on April 26, 2015.

Table of Contents

1. Introduction

The Domain Name System (DNS), DNS Security Extensions (DNSSEC), and related operational practices are defined extensively [RFC1034] [RFC1035] [RFC4033] [RFC4034] [RFC4035] [RFC4398] [RFC4509] [RFC6781] [RFC5155].

This document defines a Negative Trust Anchor, which can be used during the transition to ubiquitous DNSSEC deployment. Negative Trust Anchors (NTAs) are configured locally on a validating DNS recursive resolver to shield end users from DNSSEC-related authoritative name server operational errors. Negative Trust Anchors are intended to be temporary, and should not be distributed by IANA or any other organization outside of the administrative boundary of the organization locally implementing a Negative Trust Anchor. Finally, Negative Trust Anchors pertain only to DNSSEC and not to Public Key Infrastructures (PKI) such as X.509.

DNSSEC has now entered widespread deployment. However, the DNSSEC signing tools and processes are less mature and reliable than those for non-DNSSEC-related administration. As a result, operators of DNS recursive resolvers, such as Internet Service Providers (ISPs), occasionally observe domains incorrectly managing DNSSEC-related resource records. This mismanagement triggers DNSSEC validation failures, and then causes large numbers of end users to be unable to reach a domain. Many end users tend to interpret this as a failure of their ISP or resolver operator, and may switch to a non-validating resolver or contact their ISP to complain, rather than seeing this as a failure on the part of the domain they wanted to reach. Without the techniques in this document, this pressure may cause the resolver operator to disable (or simply not deploy) DNSSEC validation. Use of a Negative Trust Anchor to temporarily disable DNSSEC validation for a specific misconfigured domain name immediately restores access for end users. This allows the domain's administrators to fix their misconfiguration, while also allowing the organization using the Negative Trust Anchor to keep DNSSEC validation enabled and still reach the misconfigured domain.

2. Definition of a Negative Trust Anchor

Trust Anchors are defined in [RFC5914]. A trust anchor should be used by a validating caching resolver as a starting point for building the authentication chain for a signed DNS response. The inverse of this is a Negative Trust Anchor, which creates a stopping point for a caching resolver to end validation of the authentication chain. Instead, the resolver sends the response as if the zone is unsigned and does not set the AD bit. This Negative Trust Anchor can

potentially be placed at any level within the chain of trust and would stop validation from that point in the chain down.

3. delete

4. Domain Validation Failures

A domain name can fail validation for two general reasons: a legitimate security failure such as due to an attack or compromise of some sort, or as a result of misconfiguration on the part of an domain administrator. As domains transition to DNSSEC the most likely reason for a validation failure will be misconfiguration. Thus, domain administrators should be sure to read [RFC6781] in full. They should also pay special attention to Section 4.2, pertaining to key rollovers, which appear to be the cause of many recent validation failures.

It is also possible that some DNSSEC validation failures could arise due to differences in how different software developers interpret DNSSEC standards and/or how those developers choose to implement support for DNSSEC. For example, it is conceivable that a domain may be DNSSEC signed properly, and one vendor's DNS recursive resolvers will validate the domain but other vendors' software may fail to validate the domain.

5. End User Reaction

End users generally do not know what DNSSEC is, nor should they be expected to at the current time (especially absent widespread integration of DNSSEC indicators in end user software such as web browsers). As a result, end users may misinterpret the failure to reach a domain due to DNSSEC-related misconfiguration . They may (incorrectly) assume that their ISP is purposely blocking access to the domain or that it is a performance failure on the part of their ISP (especially of the ISP's DNS servers). They may contact their ISP to complain, which will incur cost for their ISP. In addition, they may use online tools and sites to complain of this problem, such as via a blog, web forum, or social media site, which may lead to dissatisfaction on the part of other end users or general criticism of an ISP or operator of a DNS recursive resolver.

As end users publicize these failures, others may recommend they switch from security-aware DNS resolvers to resolvers not performing DNSSEC validation. This is a shame since the ISP or other DNS recursive resolver operator is actually doing exactly what they are supposed to do in failing to resolve a domain name, as this is the expected result when a domain can no longer be validated, protecting end users from a potential security threat. Use of a Negative Trust

Anchor would allow the ISP to specifically remedy the failure to reach that domain, without compromising security for other sites. This would result in a satisfied end user, with minimal impact to the ISP, while maintaining the security of DNSSEC for correctly maintained domains.

6. Switching to a Non-Validating Resolver is Not Recommended

As noted in Section 5 some people may consider switching to an alternative, non-validating resolver themselves, or may recommend that others do so. But if a domain fails DNSSEC validation and is inaccessible, this could very well be due to a security-related issue. In order to be as safe and secure as possible, end users should not change to DNS servers that do not perform DNSSEC validation as a workaround, and people should not recommend that others do so either. Domains that fail DNSSEC for legitimate reasons (versus misconfiguration) may be in control of hackers or there could be other significant security issues with the domain.

Thus, switching to a non-validating resolver to restore access to a domain that fails DNSSEC validation is not a recommended practice, is bad advice to others, is potentially harmful to end user security, and is potentially harmful to DNSSEC adoption.

7. Responsibility for Failures

A domain administrator is solely and completely responsible for managing their domain name(s) and DNS resource records. This includes complete responsibility for the correctness of those resource records, the proper functioning of their DNS authoritative servers, and the correctness of DNS records linking their domain to a top-level domain (TLD) or other higher level domain. Even in cases where some error may be introduced by a third party, whether that is due to an authoritative server software vendor, software tools vendor, domain name registrar, or other organization, these are all parties that the domain administrator has selected or approved, and therefore is responsible for managing successfully.

There are some cases in which the domain administrator is not the same as the domain owner. In those cases, a domain owner has delegated operational responsibility to the domain administrator. So no matter whether a domain owner is also the domain administrator or not, the domain administrator is operationally responsible for the proper configuration operation of the domain.

So in the case of a domain name failing to successfully validate, when this is due to a misconfiguration of the domain, that is the sole responsibility of the domain administrator.

Any assistance or mitigation responses undertaken by other parties to mitigate the misconfiguration of a domain name by a domain administrator, especially operators of DNS recursive resolvers, are optional and at the pleasure of those parties.

8. Use of a Negative Trust Anchor

Technical personnel trained in the operation of DNS servers MUST confirm that a failure is due to misconfiguration, as a similar breakage could have occurred if an attacker gained access to a domain's authoritative servers and modified those records or had the domain pointed to their own rogue authoritative servers. They should also confirm that the domain is not intentionally broken, such as for testing purposes as noted in Section 12. Finally, they should make a reasonable attempt to contact the domain owner of the misconfigured zone, preferably prior to implementing the Negative Trust Anchor.

In the case of a validation failure due to misconfiguration of a TLD or popular domain name (such as a top 100 website), this could make content or services in the affected TLD or domain inaccessible for a large number of users. In such cases, it may be appropriate to use a Negative Trust Anchor as soon as the misconfiguration is confirmed.

Once a domain has been confirmed to fail DNSSEC validation due to a DNSSEC-related misconfiguration, an ISP or other DNS recursive resolver operator may elect to use a Negative Trust Anchor for that domain or sub-domain. This instructs their DNS recursive resolver to temporarily NOT perform DNSSEC validation at or in the misconfigured domain. This immediately restores access to the domain for end users while the domain's administrator corrects the misconfiguration(s). It does not and should not involve turning off validation more broadly.

A Negative Trust Anchor MUST only be used for a limited duration. Implementors SHOULD allow the operator using the Negative Trust Andhor to set an end time and date associated with any Negative Trust Anchor. Optimally this time and date is set in a DNS recursive resolver's configuration, though in the short-term this may also be achieved via other systems or supporting processes. Use of a Negative Trust Anchor MUST NOT be automatic.

Finally, a Negative Trust Anchor SHOULD be used only in a specific domain or sub-domain and MUST NOT affect validation of other names up the authentication chain. For example, a Negative Trust Anchor for zone1.example.com would affect only names at or below zone1.example.com, and validation would still be performed on example.com, .com, and the root (".") . In another example, a Negative Trust Anchor for example.com would affect only names within

example.com, and validation would still be performed on .com, and the root (".")

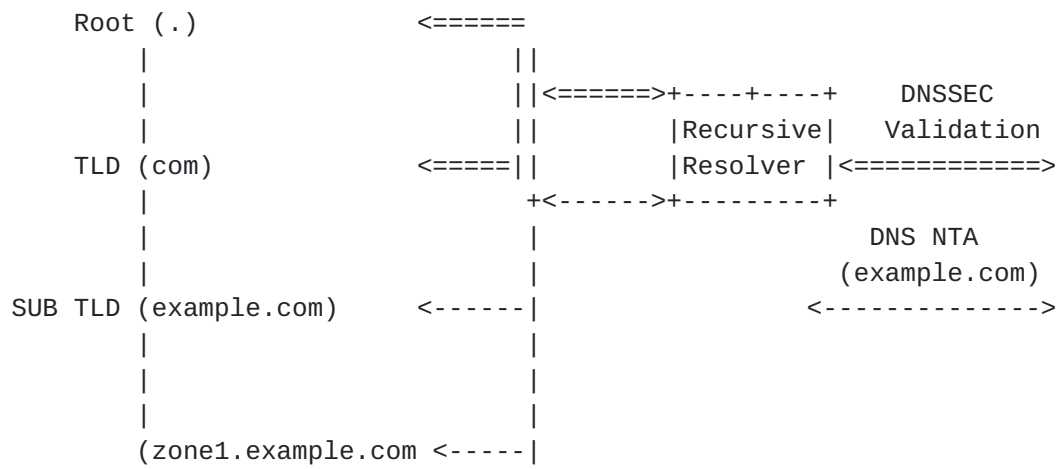

Figure 1: Negative Trust Anchor Diagram

9. Managing Negative Trust Anchors

While Negative Trust Anchors have proven useful during the early stages of DNSSEC adoption, domain owners are ultimately responsible for managing and ensuring their DNS records are configured correctly Section 7.

Most current implementations of DNS validating resolvers currently follow [RFC4033] on defining the implementation of Trust Anchor as either using Delegation Signer (DS), Key Signing Key (KSK), or Zone Signing Key (ZSK). A Negative Trust Anchor should use domain name formatting that signifies where in a delegation a validation process should be stopped.

Different DNS recursive resolvers may have different configuration names for a Negative Trust Anchor. For example, Unbound calls their configuration "domain-insecure."

[need to update reference to full Appendix A, not just unbound]

[Unbound-Configuration]

10. Removal of a Negative Trust Anchor

As explored in Section 13.1, using an NTA once the zone correctly validates can have security considerations. It is therefore recommended that NTA implementors should periodically attempt to validate the domain in question, for the period of time that the Negative Trust Anchor is in place, until such validation is again

successful. Before removing the Negative Trust Anchor, all authoritive resolvers listed in the zone should be checked. Due to AnyCast or load balancers, this may not be possible.

Once all testing succeeds, a Negative Trust Anchor should be removed as soon as is reasonably possible. Optimally this is automatic, though it may also be achieved via other systems or supporting processes.

11. Comparison to Other DNS Misconfigurations

As noted in Section 7 domain administrators are ultimately responsible for managing and ensuring their DNS records are configured correctly. ISPs or other DNS recursive resolver operators cannot and should not correct misconfigured A, CNAME, MX, or other resource records of domains for which they are not authoritative. Expecting non-authoritative entities to protect domain administrators from any misconfiguration of resource records is therefore unrealistic and unreasonable, and in the long-term is harmful to the delegated design of the DNS and could lead to extensive operational instability and/or variation.

12. Intentionally Broken Domains

Some domains, such as dnssec-failed.org, have been intentionally broken for testing purposes [Measuring-DNSSEC-Validation-of-Website-Visitors] [Netalyzr]. For example, dnssec-failed.org is a DNSSEC-signed domain that is broken. If an end user is querying a validating DNS recursive resolver, then this or other similarly intentionally broken domains should fail to resolve and should result in a SERVFAIL error. If such a domain resolved successfully, then it is a sign that the DNS recursive resolver is not fully validating.

Organizations that utilize Negative Trust Anchors should not add a Negative Trust Anchor for any intentionally broken domain.

Organizations operating an intentionally broken domain may wish to consider adding a TXT record for the domain to the effect of "This domain is purposely DNSSEC broken for testing purposes".

13. Other Considerations

13.1. Security Considerations

End to end DNSSEC validation will be disabled during the time that a Negative Trust Anchor is used. In addition, the Negative Trust Anchor may be in place after the point in time when the DNS

misconfiguration that caused validation to break has been fixed. Thus, there may be a gap between when a domain has have been re-secured and when a Negative Trust Anchor is removed. In addition, a Negative Trust Anchor may be put in place by DNS recursive resolver operators without the knowledge of the authoritative domain administrator for a given domain name. However, attempts SHOULD be made to contact and inform the domain administrator prior to putting the NTA in place.

End users of a DNS recursive resolver or other people may wonder why a domain that fails DNSSEC validation resolves with a supposedly validating resolver. As a result, implementors should consider transparently disclosing those Negative Trust Anchors which are currently in place or were in place in the past, such as on a website [Disclosure-Example]. This is particularly important since there is currently no special DNS query response code that could indicate to end users or applications that a Negative Trust Anchor is in place. Such disclosures should optimally include both the data and time that the Negative Trust Anchor was put in place and when it was removed.

13.2. Privacy Considerations

There are no privacy considerations in this document.

13.3. IANA Considerations

There are no IANA considerations in this document.

14. Acknowledgements

Several people made contributions of text to this document and/or played an important role in the development and evolution of this document. This in some cases included performing a detailed review of this document and then providing feedback and constructive criticism for future revisions, or engaging in a healthy debate over the subject of the document. All of this was helpful and therefore the following individuals merit acknowledgement:

- Joe Abley

- John Barnitz

- Tom Creighton

- Marco Davids

- Brian Dickson

- Patrik Falstrom
- Tony Finch
- Chris Ganster
- Olafur Gudmundsson
- Peter Hagopian
- Wes Hardaker
- Paul Hoffman
- Shane Kerr
- Murray Kucherawy
- Warren Kumari
- Rick Lamb
- Marc Lampo
- Ted Lemon
- Ed Lewis
- Antoin Verschuren
- Paul Vixie
- Patrik Wallstrom
- Nick Weaver
- Ralf Weber

15. References

15.1. Normative References

[RFC1034] Mockapetris, P., "Domain names - concepts and facilities", STD 13, RFC 1034, November 1987.

[RFC1035] Mockapetris, P., "Domain names - implementation and specification", STD 13, RFC 1035, November 1987.

[RFC4033] Arends, R., Austein, R., Larson, M., Massey, D., and S. Rose, "DNS Security Introduction and Requirements", RFC 4033, March 2005.

[RFC4034] Arends, R., Austein, R., Larson, M., Massey, D., and S. Rose, "Resource Records for the DNS Security Extensions", RFC 4034, March 2005.

[RFC4035] Arends, R., Austein, R., Larson, M., Massey, D., and S. Rose, "Protocol Modifications for the DNS Security Extensions", RFC 4035, March 2005.

[RFC4398] Josefsson, S., "Storing Certificates in the Domain Name System (DNS)", RFC 4398, March 2006.

[RFC4509] Hardaker, W., "Use of SHA-256 in DNSSEC Delegation Signer (DS) Resource Records (RRs)", RFC 4509, May 2006.

[RFC5155] Laurie, B., Sisson, G., Arends, R., and D. Blacka, "DNS Security (DNSSEC) Hashed Authenticated Denial of Existence", RFC 5155, March 2008.

[RFC5914] Housley, R., Ashmore, S., and C. Wallace, "Trust Anchor Format", RFC 5914, June 2010.

[RFC6781] Kolkman, O., Mekking, W., and R. Gieben, "DNSSEC Operational Practices, Version 2", RFC 6781, December 2012.

15.2. Informative References

[DNSSEC-Validation-Failure-Analysis]
Barnitz, J., Creighton, T., Ganster, C., Griffiths, C., and J. Livingood, "Analysis of DNSSEC Validation Failure - NASA.GOV", Comcast , January 2012, <http://www.dnssec.comcast.net/DNSSEC_Validation_Failure_NASAGOV_20120118_FINAL.pdf>.

[Disclosure-Example]
Comcast, "faa.gov Failing DNSSEC Validation (Fixed)", Comcast , February 2013, <http://dns.comcast.net/index.php/entry/faa-gov-failing-dnssec-validation-fixed>.

[Measuring-DNSSEC-Validation-of-Website-Visitors]
Mens, J., "Is my Web site being used via a DNSSEC-validator?", July 2012, <http://jpmens.net/2012/07/30/is-my-web-site-being-used-via-dnssec-validator/>.

[Netalyzr]
Weaver, N., Kreibich, C., Nechaev, B., and V. Paxson, "Implications of Netalyzr's DNS Measurements", Securing and Trusting Internet Names, SATIN 2011 SATIN 2011, April 2011, <http://conferences.npl.co.uk/satin/presentations/satin2011slides-Weaver.pdf>.

[Unbound-Configuration]
Wijngaards, W., "Unbound: How to Turn Off DNSSEC", June 2010, <http://unbound.net/documentation/howto_turnoff_dnssec.html>.

Appendix A. Configuration Examples

The section contains example configurations to achive Negative Trust Anchor funcationality for the zone foo.example.com.

Please note: These are simply examples - nameserver operators are expected to test and understand the implications of these operations.

A.1. NLNet Labs Unbound

Unbound lets us simply disable validation eching for a specific zone. See: <http://unbound.net/documentation/howto_turnoff_dnssec.html> [TODO(WK): Make this a "real" reference]

```
server:
        domain-insecure: "foo.example.com"
```

A.2. ISC BIND

Use the "rndc" command:

rndc nta [-lifetime duration] [-force] foo.example.com [view]

Set a negative trust anchor, disabling DNSSEC validation for the given domain. Using -lifetime specifies the duration of the NTA, up to one day. Using -force prevents the NTA from expiring before its full lifetime, even if the domain can validate sooner.

A.3. Nominum Vantio

**

negative-trust-anchors

Format: name

Command Channel: view.update name=world negative-trust-anchors=(foo.example.com)

Command Channel: resolver.update name=res1 negative-trust-anchors=(foo.example.com)

Description: Disables DNSSEC validation for a domain, even if the domain is under an existing security root.

Appendix B. Document Change Log

[RFC Editor: This section is to be removed before publication]

Individual-00: First version published as an individual draft.

Individual-01: Fixed minor typos and grammatical nits. Closed all open editorial items.

Individual-02: Simple date change to keep doc from expiring. Substantive updates planned.

Individual-03: Changes to address feedback from Paul Vixie, by adding a new section "Limited Time and Scope of Use". Changes to address issues raised by Antoin Verschuren and Patrik Wallstrom, by adding a new section "Intentionally Broken Domains" and added two related references. Added text to address the need for manual investigation, as suggested by Patrik Falstrom. Added a suggestion on notification as suggested by Marc Lampo. Made several additions and changes suggested by Ralf Weber, Wes Hardaker, Nick Weaver, Tony Finch, Shane Kerr, Joe Abley, Murray Kucherawy, Olafur Gudmundsson.

Individual-04: Moved the section defining a NTA forward, and added new text to the Abstract and Introduction per feedback from Paul Hoffman.

Individual-05: Incorporated feedback from the DNSOP WG list received on 2/17/13 and 2/18/13. This is likely the final version before the IETF 86 draft cutoff date. Updated references to RFC6781 to RFC6781, per March Davids.

Individual-06: Added more OPEN issues to continue tracking WG discussion. No changes in the main document - just expanded issue tracking.

Individual-07: Refresh document - needs revision and rework before IETF-91. Planning to add more contributors.

WG-00: Renamed at request of DNSOP co-chairs, added co-authors

WG-00 (cont):

- Using github issue tracker - go see https://github.com/wkumari/draft-livingood-dnsop-negative-trust-anchors/issues for more details.

- A bunch of readability improvments.

- Issue: Notify the domain owner of the validation failure - resolved.

- Issue: Make the NTA as specific as possible - resolved.

Appendix C. Open Issues

[RFC Editor: This section is to be removed before publication]

Determine whether RFC 2119 language should be used or not when describing things like the duration of a NTA.

The DNSOP WG should discuss whether a 1 day limit is reasonable, whether a different time (more or less than 1 day, such as 1 hour or 1 week) should be specified, or whether no time should be specified (just a recommendation that it SHOULD generally be limited to X).

Olafur Gudmundsson has suggested that we may want to consider whether a non validatable RRSIG should be returned or not when a NTA is in place. This was raised in the context of NLnet Labs' DNSSEC-Trigger, which apparently acts like forwarding stub-validator. He said, "The reason for this is if NTA strips signatures the stub-validator thinks it is under attack and may a) go into recursive mode to try to resolve the domain, getting to the right answer the long way. b) Give the wrong error "Missing signatures" instead of the real error. If all the validator does is not to set the AD bit for RRsets at and below the NTA, stub-resolvers (and cascading resolvers) should be happy."

Determine whether an informative reference to X.509 in the Introduction is necessary.

Is it desirable to say that NTAs should not be distributed across organizational boundaries?

Per Warren Kumari, add examples to appendix. "it would be very helpful to actually show how this is used, with e.g and example in an Appendix, for -insert favorite resolver here-. The document contains a lot of really useful content about why you might use one, how to minimize damage, etc but (IMO) does't do a great job of explaining

how to actually do so". Rick Lamb and Joe Abley also agreed on the need for this.

Per Rick Lamb, "it might be useful to have section 2 "Definition .." make that clear for slow people like me - that the NTA is not an RR and is more of a configuration. Maybe simply replacing "placed" with "implemented" in section 2? "This NTA can potentially be -placed/ implemented- at any level within the chain of trust"

Per Olafur Gudmundsson, address fact that ALL authoritative name servers must be working. "section 10 you talk about possible early removal the NTA when validation succeeds but there may be instances where validation succeeds when using a sub-set of the authoritative servers thus NTA should only be removed if all servers are providing "good" signatures."

Per Olafur Gudmundsson, "Furthermore what to do if some names work but others do not, for example I remember a case where the records at the apex worked but all names below the apex were signed by a key not in the DNSKEY RRset, thus it is possible that either human or automated checks may assume there is no problem when there actually is one. What this is bringing to my mind is maybe you want a new section with guidelines on how to test for failures and in what cases failure justifies NTA and what tests MUST pass before preemttive removal of an NTA."

Per Olafur Gudmundsson, "Also should there be guidance that removal of NTA should include cleaning the caches of all RRsets below the name?"

Reference and text per Ed Lewis: One thing that seems to need repeating from time to time is this passage in RFC 4033. ... In the final analysis, however, authenticating both DNS keys and data is a matter of local policy, which may extend or even override the protocol extensions defined in this document set. See Section 5 for further discussion. A responsibility (one of many) of a caching server operator is to "protect the integrity of the cache." DNSSEC is just a tool to help accomplish that. It carries ancillary data that a local cache administrator may use to filter out undesired responses. DNSSEC is not an enforcement mechanism, it's a resource. When I see folks voice opinions that DNSSEC's recommended operation has to strictly followed, my gut reaction is that these folks have forgotten the purpose of all of our efforts. We don't secure protocols to make things work better. We don't operate the DNS because we like to run a well run machine. We don't run the Internet for the fun of it. (Some might enjoy running it, that's job satisfaction to some extent.) At the end of the day all that matters is that what is being done benefits society. We run the Internet to

enrich society. We prefer a well run DNS because it saps less resources than a poorly run DNS. We prefer secure protocols so that people don't become victims (in some sense of the word). Make it work. Do what it takes to make it work. "Local policy" rules.

Per David Conrad: I'd suggest that in the BCP/RFC/whatever, in addition to recommending that NTAs be time capped and not written to permanent storage, it should also recommend NTAs be written as specifically as possible. (Should be obvious, but doesn't hurt to reiterate I suppose).

Per Ralf Weber: Informing the domain owner on the validation failure. There should be a section in the document that the operator deploying an NTA has to inform the domain owner of the problem. (JL note: would prefer to say operator SHOULD take reasonable steps to notify the domain owner, etc.)

Authors' Addresses

Paul Ebersman
Comcast
One Comcast Center
1701 John F. Kennedy Boulevard
Philadelphia, PA 19103
US

Email: ebersman-ietf@dragon.net

Chris Griffiths
Dyn
150 Dow Street
Tower Two
Manchester, NH 03101
US

Email: cgriffiths@gmail.com
URI: http://www.dyn.com

Warren Kumari
Google
1600 Amphitheatre Parkway
Mountain View, CA 94043
US

Email: warren@kumari.net
URI: http://www.google.com

Jason Livingood
Comcast
One Comcast Center
1701 John F. Kennedy Boulevard
Philadelphia, PA 19103
US

Email: jason_livingood@cable.comcast.com
URI: http://www.comcast.com

Ralf Weber
Nominum

Email: Ralf.Weber@nominum.com
URI: http://www.nominum.com