                    Changes in the Internet Threat Model
                  draft-arkko-arch-internet-threat-model-00

Abstract

   Communications security has been at the center of many security
   improvements in the Internet. The goal has been to ensure that
   communications are protected against outside observers and attackers.

   This memo suggests that the existing threat model, while important
   and still valid, is no longer alone sufficient to cater for the
   pressing security issues in the Internet. For instance, it is also
   necessary to protect systems against endpoints that are compromised,
   malicious, or whose interests simply do not align with the interests
   of the users. While such protection is difficult, there are some
   measures that can be taken.

   It is particularly important to ensure that as we continue to develop
   Internet technology, non-communications security related threats are
   properly understood. While the consideration of these issues is
   relatively new in the IETF, this memo provides some initial ideas
   about potential broader threat models to consider when designing
   protocols for the Internet or when trying to defend against pervasive
   monitoring. Further down the road, updated threat models could
   result in changes in RFC 3552 (guidelines for writing security
   considerations) and RFC 7258 (pervasive monitoring), to include
   proper consideration of non-communications security threats. It may
   also be necessary to have dedicated guidance on how systems design
   and architecture affects security.

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 https://datatracker.ietf.org/drafts/current/.

   Internet-Drafts are draft documents valid for a maximum of six months
   and may be updated, replaced, or obsoleted by other documents at any

   time. It is inappropriate to use Internet-Drafts as reference
   material or to cite them other than as "work in progress."

   This Internet-Draft will expire on November 1, 2019.

Copyright Notice

Table of Contents

1. Introduction

   Communications security has been at the center of many security
   improvements in the Internet. The goal has been to ensure that
   communications are protected against outside observers and attackers.
   At the IETF, this approach has been formalized in BCP 72 [RFC3552],

which defined the Internet threat model in 2003.

The purpose of a threat model is to outline what threats exist in
order to assist the protocol designer. But RFC 3552 also ruled some

threats to be in scope and of primary interest, and some threats out
of scope [RFC3552]:

   The Internet environment has a fairly well understood threat
   model. In general, we assume that the end-systems engaging in a
   protocol exchange have not themselves been compromised.
   Protecting against an attack when one of the end-systems has been
   compromised is extraordinarily difficult. It is, however,
   possible to design protocols which minimize the extent of the
   damage done under these circumstances.

   By contrast, we assume that the attacker has nearly complete
   control of the communications channel over which the end-systems
   communicate. This means that the attacker can read any PDU
   (Protocol Data Unit) on the network and undetectably remove,
   change, or inject forged packets onto the wire.

However, the communications-security -only threat model is becoming
outdated. This is due to three factors:

o Advances in protecting most of our communications with strong
   cryptographic means. This has resulted in much improved
   communications security, but also higlights the need for
   addressing other, remaining issues. This is not to say that
   communications security is not important, it still is:
   improvements are still needed. Not all communications have been
   protected, and even out of the already protected communications,
   not all of their aspects have been fully protected. Fortunately,
   there are ongoing projects working on improvements.

o Adversaries have increased their pressure against other avenues of
   attack, from compromising devices to legal coercion of centralized
   endpoints in conversations.

o New adversaries and risks have arisen, e.g., due to creation of
   large centralized information sources.

In short, attacks are migrating towards the currently easier targets,
which no longer necessarily include direct attacks on traffic flows.
In addition, trading information about users and ability to influence
them has become a common practice for many Internet services, often
without consent of the users.

This memo suggests that the existing threat model, while important
and still valid, is no longer alone sufficient to cater for the
pressing security issues in the Internet. For instance, while it
continues to be very important to protect Internet communications
against outsiders, it is also necessary to protect systems against

endpoints that are compromised, malicious, or whose interests simply
do not align with the interests of the users.

Of course, there are many trade-offs in the Internet on who one
chooses to interact with and why or how. It is not the role of this
memo to dictate those choices. But it is important that we
understand the implications of different practices. It is also
important that when it comes to basic Internet infrastructure, our
chosen technologies lead to minimal exposure with respect to the non-
communications threats.

It is particularly important to ensure that non-communications
security related threats are properly understood for any new Internet
technology. While the consideration of these issues is relatively
new in the IETF, this memo provides some initial ideas about
potential broader threat models to consider when designing protocols
for the Internet or when trying to defend against pervasive
monitoring. Further down the road, updated threat models could
result in changes in BCP 72 [RFC3552] (guidelines for writing
security considerations) and BCP 188 [RFC7258] (pervasive
monitoring), to include proper consideration of non-communications
security threats.

It may also be necessary to have dedicated guidance on how systems
design and architecture affects security. The sole consideration of
communications security aspects in designing Internet protocols may
lead to accidental or increased impact of security issues elsewhere.
For instance, allowing a participant to unnecessarily collect or
receive information may be lead to a similar effect as described in

[RFC8546] for protocols: over time, unnecessary information will get
used with all the associated downsides, regardless of what deployment
expectations there were during protocol design.

The rest of this memo is organized as follows. Section 2 and
Section 3 outline the situation with respect to communications
security and beyond it. Section 4 discusses how the author believes
the Internet threat model should evolve, and what types of threats
should be seen as critical ones and in-scope. Section 4.1 will also
discuss high-level guidance to addressing these threats.

Section 5 outlines the author's suggested future changes to RFC 3552
and RFC 7258 and the need for guidance on the impacts of system
design and architecture on security. Comments are solicited on these
and other aspects of this document. The best place for discussion is
on the arch-discuss list (https://www.ietf.org/mailman/listinfo/
Architecture-discuss). This memo acts also as an input for the IAB
retreat discussion on threat models, and it is a submission for the

IAB DEDR workshop (https://www.iab.org/activities/workshops/dedr-
workshop/).

Finally, Section 6 highlights other discussions in this problem space
and Section 7 draws some conclusions for next steps.

2. Improvements in Communications Security

The fraction of Internet traffic that is cryptographically protected
has grown tremendously in the last few years. Several factors have
contributed to this change, from Snowden revelations to business
reasons and to better available technology such as HTTP/2 [RFC7540],
TLS 1.3 [RFC8446], QUIC [I-D.ietf-quic-transport].

In many networks, the majority of traffic has flipped from being
cleartext to being encrypted. Reaching the level of (almost) all
traffic being encrypted is no longer something unthinkable but rather
a likely outcome in a few years.

At the same time, technology developments and policy choices have
driven the scope of cryptographic protection from protecting only the
pure payload to protecting much of the rest as well, including far

more header and meta-data information than was protected before. For
instance, efforts are ongoing in the IETF to assist encrypting
transport headers [I-D.ietf-quic-transport], server domain name
information in TLS [I-D.ietf-tls-esni], and domain name queries
[RFC8484].

There has also been improvements to ensure that the security
protocols that are in use actually have suitable credentials and that
those credentials have not been compromised, see, for instance, Let's
Encrypt [RFC8555], HSTS [RFC6797], HPKP [RFC7469], and Expect-CT
[I-D.ietf-httpbis-expect-ct].

This is not to say that all problems in communications security have
been resolved – far from it. But the situation is definitely
different from what it was a few years ago. Remaining issues will be
and are worked on; the fight between defense and attack will also
continue. Communications security will stay at the top of the agenda
in any Internet technology development.

3. Issues in Security Beyond Communications Security

There are, however, significant issues beyond communications security
in the Internet. To begin with, it is not necessarily clear that one
can trust all the endpoints.

Of course, the endpoints were never trusted, but the pressures
against endpoints issues seem to be mounting. For instance, the
users may not be in as much control over their own devices as they
used to be due to manufacturer-controlled operating system
installations and locked device ecosystems. And within those
ecosystems, even the applications that are available tend to have
features that users by themselves would most likely not desire to
have, such as excessive rights to media, location, and peripherals.
There are also designated efforts by various authorities to hack end-
user devices as a means of intercepting data about the user.

The situation is different, but not necessarily better on the side of
servers. The pattern of communications in today's Internet is almost
always via a third party that has at least as much information than
the other parties have. For instance, these third parties are

typically endpoints for any transport layer security connections, and
able to see any communications or other messaging in cleartextx.
There are some exceptions, of course, e.g., messaging applications
with end-to-end protection.

With the growth of trading users' information by many of these third
parties, it becomes necessary to take precautions against endpoints
that are compromised, malicious, or whose interests simply do not
align with the interests of the users.

Specifically, the following issues need attention:

o Security of users' devices and the ability of the user to control
   their own equipment.

o Leaks and attacks related to data at rest.

o Coercion of some endpoints to reveal information to authorities or
   surveillance organizations, sometimes even in an extra-territorial
   fashion.

o Application design patterns that result in cleartext information
   passing through a third party or the application owner.

o Involvement of entities that have no direct need for involvement
   for the sake of providing the service that the user is after.

o Network and application architectures that result in a lot of
   information collected in a (logically) central location.

o Leverage and control points outside the hands of the users or end-
   user device owners.

For instance, while e-mail transport security [RFC7817] has become
much more widely distributed in recent years, progress in securing
e-mail messages between users has been much slower. This has lead to
a situation where e-mail content is considered a critical resource by
mail providers who use it for machine learning, advertisement
targeting, and other purposes.

The Domain Name System (DNS) shows signs of ageing but due to the

legacy of deployed systems, has changed very slowly. Newer
technology [RFC8484] developed at the IETF enables DNS queries to be
performed confidentially, but its deployment is happening mostly in
browsers that use global DNS resolver services, such as Cloudflare's
1.1.1.1 or Google's 8.8.8.8. This results in faster evolution and
better security for end users.

However, if one steps back and considers the overall security effects
of these developments, the resulting effects can be different. While
the security of the actual protocol exchanges improves with the
introduction of this new technology, at the same time this implies a
move from using a worldwide distributed set of DNS resolvers into
more centralised global resolvers. While these resolvers are very
well maintained (and a great service), they are potential high-value
targets for pervasive monitoring and Denial-of-Service (DoS) attacks.
In 2016, for example, DoS attacks were launched against Dyn, one of
the largest DNS providers, leading to some outages. It is difficult
to imagine that DNS resolvers wouldn't be a target in many future
attacks or pervasive monitoring projects.

Unfortunately, there is little that even large service providers can
do to refuse authority-sanctioned pervasive monitoring. As a result
it seems that the only reasonable course of defense is to ensure that
no such information or control point exists.

There are other examples about the perils of centralised solutions in
Internet infrastructure. The DNS example involved an interesting
combination of information flows (who is asking for what domain
names) as well as a potential ability to exert control (what domains
will actually resolve to an address). Routing systems are primarily
about control. While there are intra-domain centralized routing
solutions (such as PCE [RFC4655]), a control within a single
administrative domain is usually not the kind of centralization that
we would be worried about. Global centralization would be much more
concerning. Fortunately, global Internet routing is performed a
among peers. However, controls could be introduced even in this
global, distributed system. To secure some of the control exchanges,
the Resource Public Key Infrastructure (RPKI) system ([RFC6480])
allows selected Certification Authorities (CAs) to help drive
decisions about which participants in the routing infrastructure can

make what claims. If this system were globally centralized, it would

be a concern, but again, fortunately, current designs involve at
least regional distribution.

In general, many recent attacks relate more to information than
communications. For instance, personal information leaks typically
happen via information stored on a compromised server rather than
capturing communications. There is little hope that such attacks can
be prevented entirely. Again, the best course of action seems to be
avoid the disclosure of information in the first place, or at least
to not perform that in a manner that makes it possible that others
can readily use the information.

4. The Role of End-to-end

[RFC1958] notes that "end-to-end functions can best be realised by
end-to-end protocols":

   The basic argument is that, as a first principle, certain required
   end-to-end functions can only be performed correctly by the end-
   systems themselves. A specific case is that any network, however
   carefully designed, will be subject to failures of transmission at
   some statistically determined rate. The best way to cope with
   this is to accept it, and give responsibility for the integrity of
   communication to the end systems. Another specific case is end-
   to-end security.

The "end-to-end argument" was originally described by Saltzer et al
[Saltzer]. They said:

   The function in question can completely and correctly be
   implemented only with the knowledge and help of the application
   standing at the endpoints of the communication system. Therefore,
   providing that questioned function as a feature of the
   communication system itself is not possible.

These functional arguments align with other, practical arguments
about the evolution of the Internet under the end-to-end model. The
endpoints evolve quickly, often with simply having one party change
the necessary software on both ends. Whereas waiting for network
upgrades would involve potentially a large number of parties from
application owners to multiple network operators.

The end-to-end model supports permissionless innovation where new
innovation can flourish in the Internet without excessive wait for
other parties to act.

But the details matter. What is considered an endpoint? What
characteristics of Internet are we trying to optimize? This memo
makes the argument that, for security purposes, there is a
significant distinction between actual endpoints from a user's
interaction perspective (e.g., another user) and from a system
perspective (e.g., a third party relaying a message).

This memo proposes to focus on the distinction between "real ends"
and other endpoints to guide the development of protocols. A
conversation between one "real end" to another "real end" has
necessarily different security needs than a conversation between,
say, one of the "real ends" and a component in a larger system. The
end-to-end argument is used primarily for the design of one protocol.
The security of the system, however, depends on the entire system and
potentially multiple storage, compute, and communication protocol
aspects. All have to work properly together to obtain security.

For instance, a transport connection between two components of a
system is not an end-to-end connection even if it encompasses all the
protocol layers up to the application layer. It is not end-to-end,
if the information or control function it carries actually extends
beyond those components. For instance, just because an e-mail server
can read the contents of an e-mail message does not make it a
legitimate recipient of the e-mail.

This memo also proposes to focus on the "need to know" aspect in
systems. Information should not be disclosed, stored, or routed in
cleartext through parties that do not absolutely need to have that
information.

The proposed argument about real ends is as follows:

   Application functions are best realised by the entities directly
   serving the users, and when more than one entity is involved, by
   end-to-end protocols. The role and authority of any additional
   entities necessary to carry out a function should match their part
   of the function. No information or control roles should be
   provided to these additional entities unless it is required by the
   function they provide.

For instance, a particular piece of information may be necessary for
the other real endpoint, such as message contents for another user.
The same piece of information may not be necessary for any additional
parties, unless the information had to do with, say, routing
information for the message to reach the other user. When
information is only needed by the actual other endpoint, it should be

protected and be only relayed to the actual other endpoint. Protocol

   design should ensure that the additional parties do not have access
   to the information.

   Note that it may well be that the easiest design approach is to send
   all information to a third party and have majority of actual
   functionality reside in that third party. But this is a case of a
   clear tradeoff between ease of change by evolving that third party
   vs. providing reasonable security against misuse of information.

   Note that the above "real ends" argument is not limited to
   communication systems. Even an application that does not communicate
   with anyone else than its user may be implemented on top of a
   distributed system where some information about the user is exposed
   to untrusted parties.

   The implications of the system security also extend beyond
   information and control aspects. For instance, poorly design
   component protocols can become DoS vectors which are then used to
   attack other parts of the system. Availability is an important
   aspect to consider in the analysis along other aspects.

4.1. Guidelines

   As [RFC3935] says:

      We embrace technical concepts such as decentralized control, edge-
      user empowerment and sharing of resources, because those concepts
      resonate with the core values of the IETF community.

   To be more specific, this memo suggests the following guidelines for
   protocol designers:

   1. Minimizing information passed to others: Information passed to
       another party in a protocol exchange should be minimized to guard
       against the potential compromise of that party.

   2. End-to-end protection via other parties: Information passed via
       another party who does not intrinsically need the information to
       perform its function should be protected end-to-end to its

intended recipient. This guideline is general, and holds equally
for sending TCP/IP packets, TLS connections, or application-layer
interactions. As [I-D.iab-wire-image] notes, it is a useful
design rule to avoid "accidental invariance" (the deployment of
on-path devices that over-time start to make assumptions about
protocols). However, it is also a necessary security design rule
to avoid "accidental disclosure" where information originally
thought to be benign and untapped over-time becomes a significant
information leak. This guideline can also be applied for

different aspects of security, e.g., confidentiality and
integrity protection, depending on what the specific need for
information is in the other parties.

3. Minimizing passing of control functions to others: Any passing of
control functions to other parties should be minimized to guard
against the potential misuse of those control functions. This
applies to both technical (e.g., nodes that assign resources) and
process control functions (e.g., the ability to allocate number
or develop extensions). Control functions can also become a
matter of contest and power struggle, even in cases where their
function as such is minimal, as we saw with the IANA transition
debates.

4. Avoiding centralized resources: While centralized components,
resources, and function provide usually a useful function, there
are grave issues associated with them. Protocol and network
design should balance the benefits of centralized resources or
control points against the threats arising from them. The
general guideline is to avoid such centralized resources when
possible. And if it is not possible, find a way to allow the
centralized resources to be selectable, depending on context and
user settings.

5. Explicit agreements: When users and their devices provide
information to network entities, it would be beneficial to have
an opportunity for the users to state their requirements
regarding the use of the information provided in this way. While
the actual use of such requirements and the willingness of
network entities to agree to them remains to be seen, at the
moment even the technical means of doing this are limited. For
instance, it would be beneficial to be able to embed usage

requirements within popular data formats.

5. Potential Changes in IETF Analysis of Protocols

5.1. Changes in RFC 3552

   This memo suggests that changes maybe necessary in RFC 3552. One
   initial, draft proposal for such changes would be this:

   OLD:

      In general, we assume that the end-systems engaging in a protocol
      exchange have not themselves been compromised. Protecting against
      an attack when one of the end-systems has been compromised is
      extraordinarily difficult. It is, however, possible to design

---

      protocols which minimize the extent of the damage done under these
      circumstances.

   NEW:

      In general, we assume that the end-system engaging in a protocol
      exchange has not itself been compromised. Protecting against an
      attack of a protocol implementation itself is extraordinarily
      difficult. It is, however, possible to design protocols which
      minimize the extent of the damage done when the other parties in a
      protocol become compromised or do not act in the best interests
      the end-system implementing a protocol.

   In addition, the following new section could be added to discuss the
   capabilities required to mount an attack:

   NEW:

   3.x. Other endpoint compromise

      In this attack, the other endpoints in the protocol become
      compromised. As a result, they can, for instance, misuse any
      information that the end-system implementing a protocol has sent
      to the compromised endpoint.

5.2. Changes in RFC 7258

This memo also suggests that additional guidelines may be necessary
in RFC 7258. An initial, draft suggestion for starting point of
those changes could be adding the following paragraph after the 2nd
paragraph in Section 2:

NEW:

   PM attacks include those cases where information collected by a
   legitimate protocol participant is misused for PM purposes. The
   attacks also include those cases where a protocol or network
   architecture results in centralized data storage or control
   functions relating to many users, raising the risk of said misuse.

5.3. System and Architecture Aspects

This definitely needs more attention from Internet technology
developers and standards organizations. Here is one possible

   The design of any Internet technology should start from an
   understanding of the participants in a system, their roles, and

   the extent to which they should have access to information and
   ability to control other participants.

6. Other Work

See, for instance, [I-D.farrell-etm].

7. Conclusions

More work is needed in this area. To start with, Internet technology
developers need to be better aware of the issues beyond
communications security, and consider them in design. At the IETF it
would be beneficial to include some of these considerations in the
usual systematic security analysis of technologies under development.

In particular, when the IETF develops infrastructure technology for
the Internet (such as routing or naming systems), considering the
impacts of data generated by those technologies is important.

Minimising data collection from users, minimising the parties who get
exposed to user data, and protecting data that is relayed or stored
in systems should be a priority.

A key focus area at the IETF has been the security of transport
protocols, and how transport layer security can be best used to
provide the right security for various applications. However, more
work is needed in equivalently broadly deployed tools for minimising
or obfuscating information provided by users to other entities, and
the use of end-to-end security through entities that are involved in
the protocol exchange but who do not need to know everything that is
being passed through them.

Comments on the issues discussed in this memo are gladly taken either
privately or on the architecture-discuss mailing list.

8. Acknowledgements

The author would like to thank John Mattsson, Mirja Kuehlewind,
Alissa Cooper, Stephen Farrell, Eric Rescorla, Simone Ferlin,
Kathleen Moriarty, Brian Trammell, Mark Nottingham, Christian
Huitema, Karl Norrman, Ted Hardie, Mohit Sethi, Phillip Hallam-Baker,
Goran Eriksson and the IAB for interesting discussions in this
problem space.

9. Informative References

[I-D.farrell-etm]
          Farrell, S., "We're gonna need a bigger threat model",
          draft-farrell-etm-00 (work in progress), April 2019.

[I-D.iab-wire-image]
          Trammell, B. and M. Kuehlewind, "The Wire Image of a
          Network Protocol", draft-iab-wire-image-01 (work in
          progress), November 2018.

[I-D.ietf-httpbis-expect-ct]
          estark@google.com, e., "Expect-CT Extension for HTTP",
          draft-ietf-httpbis-expect-ct-08 (work in progress),
          December 2018.

[I-D.ietf-quic-transport]

                    Iyengar, J. and M. Thomson, "QUIC: A UDP-Based Multiplexed
                    and Secure Transport", draft-ietf-quic-transport-20 (work
                    in progress), April 2019.

    [I-D.ietf-tls-esni]
                    Rescorla, E., Oku, K., Sullivan, N., and C. Wood,
                    "Encrypted Server Name Indication for TLS 1.3", draft-
                    ietf-tls-esni-03 (work in progress), March 2019.

    [I-D.nottingham-for-the-users]
                    Nottingham, M., "The Internet is for End Users", draft-
                    nottingham-for-the-users-07 (work in progress), March
                    2019.

    [RFC1958]   Carpenter, B., Ed., "Architectural Principles of the
                    Internet", RFC 1958, DOI 10.17487/RFC1958, June 1996,
                    <https://www.rfc-editor.org/info/rfc1958>.

    [RFC3552]   Rescorla, E. and B. Korver, "Guidelines for Writing RFC
                    Text on Security Considerations", BCP 72, RFC 3552,
                    DOI 10.17487/RFC3552, July 2003,
                    <https://www.rfc-editor.org/info/rfc3552>.

    [RFC3935]   Alvestrand, H., "A Mission Statement for the IETF",
                    BCP 95, RFC 3935, DOI 10.17487/RFC3935, October 2004,
                    <https://www.rfc-editor.org/info/rfc3935>.

    [RFC4655]   Farrel, A., Vasseur, J., and J. Ash, "A Path Computation
                    Element (PCE)-Based Architecture", RFC 4655,
                    DOI 10.17487/RFC4655, August 2006,
                    <https://www.rfc-editor.org/info/rfc4655>.

    [RFC6480]   Lepinski, M. and S. Kent, "An Infrastructure to Support
                    Secure Internet Routing", RFC 6480, DOI 10.17487/RFC6480,
                    February 2012, <https://www.rfc-editor.org/info/rfc6480>.

    [RFC6797]   Hodges, J., Jackson, C., and A. Barth, "HTTP Strict
                    Transport Security (HSTS)", RFC 6797,
                    DOI 10.17487/RFC6797, November 2012,
                    <https://www.rfc-editor.org/info/rfc6797>.

   [RFC6973] Cooper, A., Tschofenig, H., Aboba, B., Peterson, J.,
              Morris, J., Hansen, M., and R. Smith, "Privacy
              Considerations for Internet Protocols", RFC 6973,
              DOI 10.17487/RFC6973, July 2013,
              <https://www.rfc-editor.org/info/rfc6973>.

   [RFC7258] Farrell, S. and H. Tschofenig, "Pervasive Monitoring Is an
              Attack", BCP 188, RFC 7258, DOI 10.17487/RFC7258, May
              2014, <https://www.rfc-editor.org/info/rfc7258>.

   [RFC7469] Evans, C., Palmer, C., and R. Sleevi, "Public Key Pinning
              Extension for HTTP", RFC 7469, DOI 10.17487/RFC7469, April
              2015, <https://www.rfc-editor.org/info/rfc7469>.

   [RFC7540] Belshe, M., Peon, R., and M. Thomson, Ed., "Hypertext
              Transfer Protocol Version 2 (HTTP/2)", RFC 7540,
              DOI 10.17487/RFC7540, May 2015,
              <https://www.rfc-editor.org/info/rfc7540>.

   [RFC7817] Melnikov, A., "Updated Transport Layer Security (TLS)
              Server Identity Check Procedure for Email-Related
              Protocols", RFC 7817, DOI 10.17487/RFC7817, March 2016,
              <https://www.rfc-editor.org/info/rfc7817>.

   [RFC8446] Rescorla, E., "The Transport Layer Security (TLS) Protocol
              Version 1.3", RFC 8446, DOI 10.17487/RFC8446, August 2018,
              <https://www.rfc-editor.org/info/rfc8446>.

   [RFC8484] Hoffman, P. and P. McManus, "DNS Queries over HTTPS
              (DoH)", RFC 8484, DOI 10.17487/RFC8484, October 2018,
              <https://www.rfc-editor.org/info/rfc8484>.

   [RFC8546] Trammell, B. and M. Kuehlewind, "The Wire Image of a
              Network Protocol", RFC 8546, DOI 10.17487/RFC8546, April
              2019, <https://www.rfc-editor.org/info/rfc8546>.

   [RFC8555] Barnes, R., Hoffman-Andrews, J., McCarney, D., and J.
              Kasten, "Automatic Certificate Management Environment
              (ACME)", RFC 8555, DOI 10.17487/RFC8555, March 2019,
              <https://www.rfc-editor.org/info/rfc8555>.

   [Saltzer]   Saltzer, J., Reed, D., and D. Clark, "End-To-End Arguments
               in System Design", ACM TOCS, Vol 2, Number 4, November
               1984, pp 277-288. , n.d..

Author's Address

   Jari Arkko
   Ericsson

   Email: jari.arkko@piuha.net