Network Working Group
Internet-Draft
Intended status: Informational
Expires: 20 September 2025

D. Wing
Citrix
E. Nygren
Akamai Technologies
M. Richardson
Sandelman Software Works

19 March 2025

Requirements for HTTPS for Local Domains
draft-wing-settle-requirements-01

Abstract

When connecting to servers on their local network, users are surprised to encounter user interfaces that display errors, show insecure connections, and block some HTTP features when missing a secure context. However, obtaining PKIX certificates for those servers is difficult for a variety of reasons.

This document explores requirements for authenticating local servers.

About This Document

This note is to be removed before publishing as an RFC.

The latest revision of this draft can be found at https://danwing.github.io/settle-requirements/draft-wing-settle-requirements.html. Status information for this document may be found at https://datatracker.ietf.org/doc/draft-wing-settle-requirements/.

Discussion of this document takes place on the SETTLE mailing list (mailto:settle@ietf.org), which is archived at https://mailarchive.ietf.org/arch/browse/settle/. Subscribe at https://www.ietf.org/mailman/listinfo/settle/.

Source for this draft and an issue tracker can be found at https://github.com/danwing/settle-requirements.

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/.

Copyright Notice

Table of Contents

1. Introduction

Servers on local networks have historically settled for unencrypted communications -- printers, routers, network attached storage (NAS). However, with the advent of HTTPS everywhere [everywhere], browsers disadvantage unencrypted communications (e.g., [not-secure], [sec-context]). This increases importance of a secure context (HTTPS) to local domains.

In addition, it is recognized that home networks are not (and perhaps have never been) the idyllic secure gardens that many think they are. There are persistent threats in the home due to malware on devices within the home, as well as malware that might arrive on guest devices. Most home networks have little protection against various kinds of (layer-2) spoofing attacks, which means that active on-path attacks (MITM) must be assumed. Securing the administrative and regular connections within the home network would result in significant security gains for all devices in the home.

Today, a secure context is obtained with a PKIX certificate ([RFC5280]) signed by a Certification Authority (CA) that is trusted by the client.

However, servers on a local network cannot easily get PKIX certificates signed by a Certification Authority because: they are not directly reachable from the outside (due to firewall or NAPT), lack of domain name delegation, and need for ongoing certificate renewal.

The problem has been well recognized since about 2010 and several proposals have been suggested to solve this problem, each with their own drawbacks. This document is not intended to summarize these proposals or their drawbacks; for that detail see the pointers to previous work in Section 7. At a high level, the proposals have involved solutions such as:

* pre-shared secrets (scanned, printed, or displayed by the server)

* Public DNS pointing at local domain’s IP address (e.g., [plex])

* Local Certification Authority, where a CA is added to client’s certificate trust list and that CA signs certificates for devices within the local network

* Trust On First Use (TOFU), where a user verifies the first connection to a server and the client remembers that verification, similar to common use of ssh

* WebRTC and WebTransport, where a PKI-signed server provides a public key fingerprint of another server that it has previously bootstrapped

* Encoding server’s public key into the hostname [thomson-hld]

This document explores IETF requirements for an alternative server authentication system for local hosts.

2. Conventions and Definitions

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "NOT RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in BCP 14 [RFC2119] [RFC8174] when, and only when, they appear in all capitals, as shown here.

3. Technical Requirements

The goal is to work out the engineering tradeoff around [zookotriangle]. Specifically it says there are three aspects that must be traded off:

* Human-meaningful

* Secure

* Decentralized

3.1. Naming

PKIX certificates are a centralized naming scheme derived from DNS. These names have (the possibility of) having human-readable names. But the most significant property is uniqueness -- each name has its own identity and that identity can be proven.

A system that does not rely on centralized naming lacks this inherient uniqueness property.

Name collisions can be engineered by attackers for nefarious purposes. For example, if a victim is configured to use the (likely) unique name "printer-12ab34cd56ef.local" (containing the printer's full or partial MAC address), an attacker can respond to connections to that name, potentially stealing the user's authentication credentials to that printer or seeing the content the user sent to that printer. Similar attacks are possible with file shares. This problem is exacerbated if non-unique names are used (e.g., simply "printer.local" or "router.local"), as it reduces the attacker's effort. Humans prefer simple, human-readable names, but a strong identity cannot be created with such names.

R-UNIQUE-NAME: The system MUST have a way to uniquely identify servers.

3.2. Cryptographic Binding

A server's name has to be mapped to its cryptographic identity.

R-BINDING: The Web Origin MUST be cryptographically bound to one or more key pairs, where the private keying material is on the service endpoint and where an attacker without the private key(s) is unable to access any state associated with the Web Origin.

A client has to be able to validate the name maps to the cryptographic identity.

R-VALIDATE: Clients MUST be able to cryptographically validate that the authenticating server matches the identity in the URI / Web Origin.

Web browsers and modern users both expect a URI.

R-URI: It MUST be possible to construct a URI that encapsulates a Web Origin and its cryptographically-bound identity information.

3.3. Abstract Naming

Using IP addresses in names is problematic if the server's IP address changes due to ISP renumbering or internal network DHCP server reconfiguration.

Given common NAT44 (NAPT), many many networks will share the same IPv4 addresses.

R-ABSTRACT: The solution SHOULD abstract names from IP addresses.

Any given name should be resolvable to a mixture of IPv4, IPv6 Link-Local (on an Interface), IPv6 ULA, and IPv6 Globally-Routable addresses. Operating a local DNS is beyond the scope of many administrators, so being able to advertise the server using [DNS-SD] is necessary.

R-DNS-SD: The name MUST be advertisable using [DNS-SD]

3.4. Avoid Central Authority

A solution needs to be self-contained and not use the central authority of PKIX. Being self-contained also removes reliance on a device vendor (to operate a centralized service).

R-AVOID-CENTRAL: A solution MUST NOT rely on central trust hierarchy.

Vendors go out of business or lose interest in continuing to service old products. The products may still be operational.

R-AVOID-VENDOR: A solution SHOULD NOT (MUST NOT?) have continued reliance on a service operated by a vendor, including if the device is reset to factory defaults (e.g., reset for troubleshotting or because sold).

3.5. Multiple Application Protocols

A solution has to support HTTPS because it is frequently used for device management.

R-HTTPS: A solution MUST support HTTPS.

Other encrypted protocols are also frequently used on local networks for DNS, file sharing, mail, and telephony.

R-MULT-APP: A solution SHOULD support other application-level protocols such as DoT [RFC7858], SMB over QUIC [smb-quic], IMAP [RFC8314], and SIP [RFC3261], as those protocols are routinely served within a local domain.

3.6. Cryptographic Agility

A solution has to support moving to new cryptographic functions.

R-AGILITY: A solution SHOULD support cryptographic agility (such as supporting more than one active key type and different hashes).

3.7. TLS Server Name Indication

TLS servers frequently use the TLS SNI [RFC6066] extension to support multiple identities on a single server.

R-TLS-SNI: A solution SHOULD support TLS SNI so a server knows which key pair/cert is expected.

3.8. W3C Private Network Access

To prevent various attacks, W3C has constrained how browsers operate on private networks

R-PNA: A solution SHOULD integrate well with an evolution of [w3c-pna] and both allow for an improved model there but should also provide more robust solutions to vulnerabilities that it tries to address

3.9. Operate with Local Resources

The TLDs ".local" and ".internal" are defined local domains and enterprise networks usually have a site domain ("internal.example.com"). A solution that scales from a home network to an enterprise network is desirable.

R-LOCAL: A solution MUST operate with .local and .internal, and SHOULD operate with an administratively-defined zone (e.g., internal.example.com).

Discuss: MAY constrain to the DHCP domain-search value?? Should we also allow any arbitrary name if the IP address is local (RFC1918 address), too?

3.10. Operate Standalone

The system needs to operate without a connection to the Internet. This is necessary because Internet connectivity is sometimes flaky or unavailable (e.g., cabin in the woods, lengthy ISP outage).

R-STANDALONE: MUST NOT require Internet connectivity to operate securely, for its initial configuration, or to add or remove a device from list of authorized devices in the system.

Discuss: perhaps want to refine wording of this requirement, or split into separate requirements.

3.11. Web Origin

The Web Origin is comprised of the scheme (e.g., "https:"), hostname, and port. Today, when a key rotation occurs the Web Origin remains the same. In this way, things like stored web data (forms, passwords, cookies) can be used even after a key rotation.

R-WEB-ORIGIN: The Web Origin MUST be retained during key rotation.

3.12. Miscellaneous

1. It SHOULD be possible to have a way to represent a URI that includes a single specific IP address and the cryptographic identity of the service endpoint.

   Discuss: the above requirement needs to be re-written.

1. SHOULD support key rotation (even if via 301 redirect)

* Q: is it acceptable to state to be lost here? Note: likely cannot do 301 if doing TLS (HTTPS). Is this suggestion to start HTTP and upgrade to HTTPS? Could be useful for HTTPS but redirect unavailable for IPP, SMB, DoH.

  Discuss: the above requirement needs to be re-written.

1. SHOULD support building trust relationships within devices in the local environment

   Discuss: the above requirement needs to be re-written.

1. Could this help with HTTPS access to Wi-Fi login portals ([RFC8952], [RFC8910])?

   Discuss: the above requirement needs to be re-written.

4. Human Factors Requirements

4.1. Discoverable

Most local networks, especially home networks, do not operate their own DNS server. Many clients already support listening for DNS-SD broadcasts.

R-DISCOVER: A solution SHOULD have a way to do discovery of endpoints and their identities (for example, via [DNS-SD]).

4.2. Easy to Use

Successful solutions are usually also easy to deploy.

> R-EASY: A solution SHOULD have human factors and adversarial testing on proposed solutions to make sure that this solution provides a reasonable experience to average and novice end-users and does not introduce new security exploitation vectors

4.3. Bookmarkable

Names (or aliases to those names) should be simple for users -- ideally, user-defined so that if the underlying name is complex the user can create an alias that is meaningful to them.

> R-BOOKMARK: A solution SHOULD have a URI that users can Bookmark to create an association to a friendly name.
>
> Discussion: Can URL bar of the browser honor mDNS/DNSSD advertised names, or give a pull-down of them similar to how the "add printer" dialog does for printers? This would help ease the use of long FQDN so it’s almost as easy as router.local. Especially if it could show a nickname that is configured by the printer. Browser extensions exist for DNS-SD and mDNS ([Safari-ext], [Firefox-ext]).

4.4. Human-friendly Name

Names (or aliases to those names) should be simple for users -- ideally, user-defined so that if the underlying name is complex the user can create an alias that is meaningful to them.

> R-CONSISTENT: A solution SHOULD represent these URIs to humans in a consistent, readable, and non-confusing fashion. (In a browser, users shouldn’t see the key fingerprint by default but rather a representation of its presence)

5. Big Open Questions

5.1. Trust on First Use (TOFU)

As evidenced by web browser behavior over the years with self-signed certificates and their (increased) warnings, TOFU will not be acceptable.

5.2. User Experience

For a solution, what is the User Experience for any trust relationship / web-of-trust?

5.3. Trust Relationship

For a solution, what is the nature of the trust relationship?

* Peer trust web?
* Central CA within the local environment / trust clearing house?
* Client establishes its own trust to the server

5.4. Interaction with Matter/Thread

How does a solution tie into systems like Matter/Thread that have their own trust establishment frameworks?

6. Use Cases

For the below, "Secure communications" means being able to make a TLS connection to a service such that the service is able to authenticate itself in a way to prevent MitM attacks. The security model must be TOFU at a minimum, but when the identity of a service is none it should be possible to send it as a URI in such as a way to present a secure association rooted in the connection that sent it:

* Secure communications via HTTPS to admin interfaces on CPEs for both initial and ongoing configuration tasks of various servers (router, printer, NAS, etc.).
* Secure communications to DoH/DoT servers on CPEs
* Secure communications to printers (IPPS [RFC7472] printing)
* Secure communications to other local services (SMB over QUIC to another workstation or a NAS) and IoT devices
* Secure communications to localhost processes from a browser (e.g., admin tools)

7. Related Work

Martin Thomson wrote HTTPS for Local Domains [thomson-hld] which covers requirements, discusses several solutions and their tradeoffs, and suggests a solution where the client extends the Web Origin to include the server's public key. It does not allow the server to rotate its public key as that would change the extended Web Origin (see Section 3.11).

Dan Wing has proposed a Referee system [I-D.wing-settle-referee] which uses a new HTTPS-based server to authorize servers public keys (akin to an allowlist or to OCSP stapling) and encoding the server's public key into its hostname. Like [thomson-hld], it does not allow a server to rotate its key (see Section 3.11).

Michael Sweet has proposed a locally-deployed Certification Authority [I-D.sweet-iot-acme] that can be incrementally deployed.

W3C worked on this problem from 2017 through 2021 [w3c-httpslocal]. More recently, W3C had a workshop on the problem in September 2024 [tpac].

W3C has a working draft on OpenScreen Network Protocol [w3c-onsp] which establish trust between devices for the purposes of media casting and remote presentation using DNS-SD, TLS for an initial unauthenticated connection, SPAKE2 to validate mutual identity and exchange certificates.

W3C WICG also has a working draft on Peer-to-Peer API [w3c-local-peer] that layers a web API on top of the OpenScreen Network Protocol to allow local communication between browsers without the aid of a server. It contains proposed APIs for the following on a local network: peer advertising, peer discovery and authentication, and establishing a WebTransport.

The boundaries of a limited domain -- such as the local domain described in this document -- are explored in Section 6 of [RFC8799].

The IOTOPS working group and the associated IOT Security Foundation [iotsf] discussed the problem and some requirements in their white paper [iotops-suib] and presentation to IOTOPS working group at IETF112 [iotops-suib-prezo].

A threshold key system is described and implemented at [phb-mesh] with the following description:

> The Mesh is designed to provide users with the highest level of security that is possible without asking them to do anything at all. For this to become possible, the Mesh will have to be shipped to users as part of the machine Operating System.

A summary of the problem and analysis of several solutions (Locally-installed CAs, Plex, WebRTC and WebTransport, TOFU, shared secrets) and some drawbacks of those solutions is at [stark].

A method using PAKE and a shared secret (displayed on the server) is explained at [shared].

8. Security Considerations

TODO Security

9. IANA Considerations

This document has no IANA actions.

10. References

10.1. Normative References

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, DOI 10.17487/RFC2119, March 1997, <https://www.rfc-editor.org/rfc/rfc2119>.

[RFC8174] Leiba, B., "Ambiguity of Uppercase vs Lowercase in RFC 2119 Key Words", BCP 14, RFC 8174, DOI 10.17487/RFC8174, May 2017, <https://www.rfc-editor.org/rfc/rfc8174>.

10.2. Informative References

[DNS-SD] Cheshire, S. and M. Krochmal, "DNS-Based Service Discovery", RFC 6763, DOI 10.17487/RFC6763, February 2013, <https://www.rfc-editor.org/rfc/rfc6763>.

[everywhere] EFF, "HTTPS Everywhere", March 2025, <https://www.eff.org/https-everywhere>.

[Firefox-ext] Smith, D., "mDNS Discover", December 2020, <https://addons.mozilla.org/en-US/firefox/addon/mdns-discover/>.

[I-D.sweet-iot-acme]
Sweet, M., "ACME-Based Provisioning of IoT Devices", Work in Progress, Internet-Draft, draft-sweet-iot-acme-07, 7 February 2025, <https://datatracker.ietf.org/doc/html/draft-sweet-iot-acme-07>.

[I-D.wing-settle-referee]
Wing, D., "A Referee to Authenticate Servers in Local Domains", Work in Progress, Internet-Draft, draft-wing-settle-referee-00, 7 January 2025, <https://datatracker.ietf.org/doc/html/draft-wing-settle-referee-00>.

[iotops-suib]
IOT Security Foundation, "SUIB: Router and IoT Vulnerabilities: Insecure by Design", August 2021, <https://iotsecurityfoundation.org/wp-content/uploads/2021/08/ManySecured-SUIB-White-Paper.pdf>.

[iotops-suib-prezo]
Geertsma, J., Ams端ss, C., Richardson, M., and N. Allott, "SUIB: Browsing local web resources in a secure usable manner", November 2021, <https://datatracker.ietf.org/meeting/112/materials/slides-112-iotops-suib-browsing-local-web-resources-in-a-secure-usable-manner-iot-device-configuration-as-a-special-case-00.pdf>. Presentation of IOT Security Foundation SUIB to IETF112 IOTOPS working group

[iotsf]
"IOT Security Foundation", September 2015, <https://iotsecurityfoundation.org>.

[not-secure]
Google, "A secure web is here to stay", 2018, <https://blog.chromium.org/2018/02/a-secure-web-is-here-to-stay.html>.

[phb-mesh] Hallam-Baker, P., "Mathematical Mesh", 2022, <https://github.com/hallambaker/Mathematical-Mesh>.

[plex]
Valsorda, F., "How Plex Is Doing Https for All Its Users", June 2015, <https://words.filippo.io/how-plex-is-doing-https-for-all-its-users/>.

[RFC3261] Rosenberg, J., Schulzrinne, H., Camarillo, G., Johnston, A., Peterson, J., Sparks, R., Handley, M., and E. Schooler, "SIP: Session Initiation Protocol", RFC 3261, DOI 10.17487/RFC3261, June 2002, <https://www.rfc-editor.org/rfc/rfc3261>.

[RFC5280] Cooper, D., Santesson, S., Farrell, S., Boeyen, S., Housley, R., and W. Polk, "Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile", RFC 5280, DOI 10.17487/RFC5280, May 2008, <https://www.rfc-editor.org/rfc/rfc5280>.

[RFC6066] Eastlake 3rd, D., "Transport Layer Security (TLS) Extensions: Extension Definitions", RFC 6066, DOI 10.17487/RFC6066, January 2011, <https://www.rfc-editor.org/rfc/rfc6066>.

[RFC7472] McDonald, I. and M. Sweet, "Internet Printing Protocol (IPP) over HTTPS Transport Binding and the 'ipps' URI Scheme", RFC 7472, DOI 10.17487/RFC7472, March 2015, <https://www.rfc-editor.org/rfc/rfc7472>.

[RFC7858] Hu, Z., Zhu, L., Heidemann, J., Mankin, A., Wessels, D., and P. Hoffman, "Specification for DNS over Transport Layer Security (TLS)", RFC 7858, DOI 10.17487/RFC7858, May 2016, <https://www.rfc-editor.org/rfc/rfc7858>.

[RFC8314] Moore, K. and C. Newman, "Cleartext Considered Obsolete: Use of Transport Layer Security (TLS) for Email Submission and Access", RFC 8314, DOI 10.17487/RFC8314, January 2018, <https://www.rfc-editor.org/rfc/rfc8314>.

[RFC8799] Carpenter, B. and B. Liu, "Limited Domains and Internet Protocols", RFC 8799, DOI 10.17487/RFC8799, July 2020, <https://www.rfc-editor.org/rfc/rfc8799>.

[RFC8910] Kumari, W. and E. Kline, "Captive-Portal Identification in DHCP and Router Advertisements (RAs)", RFC 8910, DOI 10.17487/RFC8910, September 2020, <https://www.rfc-editor.org/rfc/rfc8910>.

[RFC8952] Larose, K., Dolson, D., and H. Liu, "Captive Portal Architecture", RFC 8952, DOI 10.17487/RFC8952, November 2020, <https://www.rfc-editor.org/rfc/rfc8952>.

[Safari-ext]
Ballard, L., "Discovery - DNS-SD Browser", 2022, <https://apps.apple.com/ca/app/discovery-dns-sd-browser/id1381004916?mt=12>.

[sec-context]
W3C, "Secure Contexts", 2023, <https://w3c.github.io/webappsec-secure-contexts/>.

[shared] W3C, "APPROACH-2: Using Shared Secret", September 2019, <https://httpslocal.github.io/proposals/#approach-2>.

[smb-quic] Microsoft, "SMB over QUIC", December 2024, <https://learn.microsoft.com/en-us/windows-server/storage/file-server/smb-over-quic>.

[stark] Stark, E. M., "When a web PKI certificate won’t cut it", December 2021, <https://emilymstark.com/2021/12/24/when-a-web-pki-certificate-wont-cut-it.html>.

[thomson-hld]
Thomson, M., "HTTPS for Local Domains", September 2017, <https://docs.google.com/document/u/0/d/170rFC91jqvpFrKIqG4K8Vox8AL4LeQXzfikBQXYPmzU/edit>.

[tpac] IL, C., "HTTPS for Local Networks", September 2024, <https://github.com/w3c/tpac2024-breakouts/issues/78>.

[w3c-httpslocal]
W3C, "HTTPS in Local Network Community Group", 2019, <https://github.com/httpslocal>.

[w3c-local-peer]
W3C WICG, "Local Peer-to-Peer API", August 2024, <https://wicg.github.io/local-peer-to-peer/>.

[w3c-onsp] W3C, "Open Screen Network Protocol", December 2024, <https://www.w3.org/TR/openscreen-network/>.

[w3c-pna] W3C, "Private Network Access", September 2024, <https://wicg.github.io/private-network-access/>.

[zookotriangle]
Wikipedia, "Zooko’s triangle", March 2025, <https://en.wikipedia.org/wiki/Zooko%27s_triangle>.

Appendix A. Change History

A.1. Changes in -01

* Changed to 2010 (from 2017) when the problem of local domain authentication was first discussed.

* Rather than simple name collision ("printer.local"), discuss how most products include the device's (partial) MAC address -- which does help distinguishing devices on different networks. Also explain how an attacker can use that name.

* R-AVOID-CENTRAL changed from SHOULD to MUST.

* Justification text added to almost all R- requirements.

* Removed R-LOCALHOST, which had said "localhost" should be handled same as a local domain. This was removed because localhost is not a unique name causing other problems for a solution.

* R-LOCAL expanded to also cover administratively-defined zone (e.g., internal.example.com)

* Refined R-STANDALONE.

* Added R-WEB-ORIGIN, and moved key rotation requirement into R-WEB-ORIGIN to say the web origin has to stay the same if the key is rotated.

* Declared TOFU as unacceptable (was a question).

* As Related Work, added: Referee, locally-deployed CA, W3C OpenScreen Network Protocol, W3C WICG Peer-to-Peer API.

* Added pointers to DNS-SD and mDNS extensions for web browsers.

Acknowledgments

Thanks to Michael Sweet for his review and feedback. Thanks to Michiel De Backker for references to related W3C work.

Authors' Addresses

Dan Wing
Cloud Software Group, Inc.
Email: danwing@gmail.com

Erik Nygren
Akamai Technologies

Email: erik+ietf@nygren.org
URI: http://erik.nygren.org/

Michael Richardson
Sandelman Software Works
Email: mcr+ietf@sandelman.ca