DNSOP
Internet-Draft
Updates: 6761 (if approved)
Intended status: Standards Track
Expires: June 21, 2018

M. West
Google, Inc

December 18, 2017

Let ’localhost’ be localhost.
draft-ietf-dnsop-let-localhost-be-localhost-02

Abstract

This document updates the treatment of the special-use domain name "localhost" as specified in RFC6761, Section 6.3, with the goal of ensuring that it can be safely relied upon as a name for the local host’s loopback interface. To that end, stub resolvers are required to resolve localhost names to loopback addresses. Recursive DNS servers are required to return "NXDOMAIN" when queried for localhost names, making non-conformant stub resolvers more likely to fail and produce problem reports that result in updates.

Together, these requirements would allow applications and specifications to join regular users in drawing the common-sense conclusions that "localhost" means "localhost", and doesn’t resolve to somewhere else on the network.

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 June 21, 2018.

Table of Contents

1. Introduction

The "127.0.0.0/8" IPv4 address block and "::1/128" IPv6 address block are reserved as loopback addresses. Traffic to these blocks is assured to remain within a single host, and can not legitimately appear on any network anywhere. This turns out to be a very useful

property in a number of circumstances; useful enough to label explicitly and interoperably as "localhost". [RFC1537] suggests that this special-use top-level domain name has been implicitly mapped to loopback addresses for decades at this point, and that [RFC6761]'s assertion that developers may "assume that IPv4 and IPv6 address queries for localhost names will always resolve to the respective IP loopback address" is well-founded.

Unfortunately, the rest of that latter document's requirements undercut the assumption it suggests. Client software is empowered to send localhost names to DNS servers, and resolvers are empowered to return unexpectedly non-loopback results. This divide between theory and practice has a few impacts:

First, the lack of confidence that "localhost" actually resolves to the loopback interface encourages application developers to hard-code IP addresses like "127.0.0.1" in order to obtain certainty regarding routing. This causes problems in the transition from IPv4 to IPv6 (see problem 8 in [I-D.ietf-sunset4-gapanalysis]).

Second, HTTP user agents sometimes distinguish certain contexts as "secure"-enough to make certain features available. Given the certainty that "127.0.0.1" cannot be maliciously manipulated or monitored, [SECURE-CONTEXTS] treats it as such a context. Since "localhost" might not actually map to the loopback address, that document declines to give it the same treatment. This exclusion has (rightly) surprised some developers, and exacerbates the risks of hard-coded IP addresses by giving developers positive encouragement to use an explicit loopback address rather than a localhost name.

This document updates [RFC6761]'s recommendations regarding "localhost" by requiring that name resolution APIs and libraries themselves return a loopback address when queried for localhost names, bypassing lookup via recursive and authoritative DNS servers entirely.

In addition, recursive and authoritative DNS servers are required to return "NXDOMAIN" for such queries. This increases the likelihood that non-conformant stub resolvers will not go undetected. Note that this does not have the result that such resolvers will fail safe--it just makes it more likely that they will be detected and fixed, since they will fail in the presence of conforming name servers.

These changes are not sufficient to ensure that "localhost" can be assumed to actually refer to an address on the local machine. This document therefore further requires that applications that wish to make that assumption handle the name "localhost" specially.

2. Terminology and notation

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in [RFC2119].

IPv4 loopback addresses are registered in Table 4 of Section 2.2.2 of [RFC6890] as "127.0.0.0/8".

IPv6 loopback addresses are registered in Table 17 of Section 2.2.3 of [RFC6890] as "::1/128".

The domain "localhost.", and any names falling within ".localhost.", are known as "localhost names".

3. The "localhost." Special-Use Domain Name

Localhost names are special insofar as these names do not exist in the DNS, and querying the DNS for them is an error. With that principle in mind, the considerations outlined in [RFC6761] can be answered as follows:

1. Users are free to use localhost names as they would any other domain names. Users may assume that IPv4 and IPv6 address queries for localhost names will always resolve to the respective IP loopback address.

2. Application software MAY recognize localhost names as special, or MAY pass them to name resolution APIs as they would for other domain names.

   If application software wishes to make security decisions based upon the assumption that localhost names resolve to loopback addresses (e.g. if it wishes to ensure that a context meets the requirements laid out in [SECURE-CONTEXTS]), then it MUST directly translate localhost names to a loopback address, and MUST NOT rely upon name resolution APIs to do so.

   Application software MUST NOT use a searchlist to resolve a localhost name. That is, even if DHCP’s domain search option [RFC3397] is used to specify a searchlist of "example.com" for a given network, the name "localhost" will not be resolved as "localhost.example.com." but as "localhost.", and "subdomain.localhost" will not be resolved as "subdomain.localhost.example.com." but as "subdomain.localhost.".

3. Name resolution APIs and libraries MUST recognize localhost names as special, and MUST always return an appropriate IP loopback

address for IPv4 and IPv6 address queries and negative responses for all other query types. Name resolution APIs MUST NOT send queries for localhost names to their configured recursive DNS server(s).

As for application software, name resolution APIs and libraries MUST NOT use a searchlist to resolve a localhost name.

4. (Caching) recursive DNS servers MUST respond to queries for localhost names with NXDOMAIN.

5. Authoritative DNS servers MUST respond to queries for localhost names with NXDOMAIN.

6. DNS server operators SHOULD be aware that the effective RDATA for localhost names is defined by protocol specification and cannot be modified by local configuration.

7. DNS Registries/Registrars MUST NOT grant requests to register localhost names in the normal way to any person or entity. Localhost names are defined by protocol specification and fall outside the set of names available for allocation by registries/ registrars. Attempting to allocate a localhost name as if it were a normal DNS domain name will not work as desired, for reasons 2, 3, 4, and 5 above.

4. IANA Considerations

4.1. Domain Name Reservation Considerations

This document requests that IANA updates the "localhost." registration in the registry of Special-Use Domain Names [RFC6761] to reference this document rather than [RFC6761].

Considerations for this reservation are detailed in Section 3.

4.2. DNSSEC

The ".localhost" TLD is already assigned to IANA, as per [RFC2606], but does not have an entry in the root-zone. This means that the root will return an NXDOMAIN response along with NSEC records constituting a secure denial of existence if queried. That’s consistent with the general principle that localhost names do not exist in the DNS, and the subsequent requirements to return NXDOMAIN that are laid out in Section 3.

5. Security Considerations

5.1. Applications are encouraged to resolve localhost names themselves.

Applications that attempt to use the local resolver to query "localhost" do not fail safely. If an attacker sets up a malicious DNS server which returns a non-loopback address when queried for localhost names, such applications will connect to that remote server assuming it is local. This risk drives the requirement that applications resolve localhost names themselves if they intend to make security decisions based on the assumption that localhost names resolve locally.

There may be cases in which the target runtime environment can be safely assumed to do the right thing with localhost names. In this case, the requirement that the application resolve localhost names on its own may be safe to ignore, but only if all the requirements under point 2 of Section 3 are known to be followed by the resolver that is known to be present in the target environment.

5.2. ’localhost’ labels in subdomains

Hosts like "localhost.example.com" and "subdomain.localhost.example.com" contain a "localhost" label, but are not themselves localhost names, as they do not fall within "localhost.". Therefore, they are not directly affected by the recommendations in this document. They have no resolution guarantees one way or another, and should not be given special treatment, either in DNS or in client software.

Note, however, that the admonition against searchlist usage could affect their resolution in practice, as discussed in Section 3. For example, even with a searchlist of "example.com" in place for a given network, the name "localhost" will not be resolved as "localhost.example.com." but as "localhost.", and "subdomain.localhost" will not be resolved as "subdomain.localhost.example.com." but as "subdomain.localhost.".

6. Implementation Considerations

6.1. Non-DNS usage of localhost names

Some application software differentiates between the hostname "localhost" and the IP address "127.0.0.1". MySQL, for example, uses a unix domain socket for the former, and a TCP connection to the loopback address for the latter. The constraints on name resolution APIs above do not preclude this kind of differentiation.

7. References

7.1. 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/info/rfc2119>.

[RFC2606] Eastlake 3rd, D. and A. Panitz, "Reserved Top Level DNS Names", BCP 32, RFC 2606, DOI 10.17487/RFC2606, June 1999, <https://www.rfc-editor.org/info/rfc2606>.

[RFC6761] Cheshire, S. and M. Krochmal, "Special-Use Domain Names", RFC 6761, DOI 10.17487/RFC6761, February 2013, <https://www.rfc-editor.org/info/rfc6761>.

[RFC6890] Cotton, M., Vegoda, L., Bonica, R., Ed., and B. Haberman, "Special-Purpose IP Address Registries", BCP 153, RFC 6890, DOI 10.17487/RFC6890, April 2013, <https://www.rfc-editor.org/info/rfc6890>.

7.2. Informative References

[I-D.ietf-sunset4-gapanalysis] LIU, W., Xu, W., Zhou, C., Tsou, T., Perreault, S., Fan, P., Gu, R., Xie, C., and Y. Cheng, "Gap Analysis for IPv4 Sunset", draft-ietf-sunset4-gapanalysis-09 (work in progress), August 2017.

[RFC1537] Beertema, P., "Common DNS Data File Configuration Errors", RFC 1537, DOI 10.17487/RFC1537, October 1993, <https://www.rfc-editor.org/info/rfc1537>.

[RFC3397] Aboba, B. and S. Cheshire, "Dynamic Host Configuration Protocol (DHCP) Domain Search Option", RFC 3397, DOI 10.17487/RFC3397, November 2002, <https://www.rfc-editor.org/info/rfc3397>.

[SECURE-CONTEXTS] West, M., "Secure Contexts", n.d., <http://w3c.github.io/webappsec-secure-contexts/>.

Appendix A. Changes from RFC 6761

Section 3 updates the requirements in section 6.3 of [RFC6761] in a few substantive ways:

1. Application software and name resolution APIs and libraries are prohibited from using searchlists when resolving localhost names, and encouraged to bypass resolution APIs and libraries altogether if they intend to make security decisions based on the "localhost" name.

2. Name resolution APIs and libraries are required to resolve localhost names to loopback addresses, without sending the query on to caching DNS servers.

3. Caching and authoritative DNS servers are required to respond to resolution requests for localhost names with NXDOMAIN.

Appendix B. Changes in this draft

B.1. draft-ietf-dnsop-let-localhost-be-localhost-02

- Based on some feedback from Suzanne Woolf, this draft:
  - Clarified the abstract (https://github.com/mikewest/internetdrafts/commit/837b89f35e08e98b0e02df87032c4ccc19cd06eb)
  - Addressed nits in the "IANA considerations" section (https://github.com/mikewest/internetdrafts/commit/d65d4fbaec6afbbae70496ffb98dfb60e8d3e2eb)
  - Reworded the "Non-TLD localhost" section (https://github.com/mikewest/internetdrafts/commit/44b1d7d4cfcb65aab3c46ff1c436a75a2fb3403f)
  - Made the reference to [RFC2606] normative (https://github.com/mikewest/internetdrafts/commit/cd94988a966b93d2239de03d54513031a5823c0a)

B.2. draft-ietf-dnsop-let-localhost-be-localhost-01

- Explicit adoption of the principle Wes Hardaker proposed in https://www.ietf.org/mail-archive/web/dnsop/current/msg21039.html , and that Warren Kumari reiterated in https://www.ietf.org/mail-archive/web/dnsop/current/msg21129.html : localhost names do not exist in the DNS, there is no authoritative source for these names, and querying resolvers for them is an error.
- Slight tightening of the admonition against search lists.
- Addressed "localhost" labels in non-localhost names.

B.3. draft-ietf-dnsop-let-localhost-be-localhost-00

- No change since draft-west-let-localhost-be-localhost-06, just renaming the document after DNSOP adopted it.

B.4. draft-west-let-localhost-be-localhost-06

- Incorporated Ted Lemon's further feedback from https://www.ietf.org/mail-archive/web/dnsop/current/msg20769.html
- Explicitly waffling on DNSSEC.

B.5. draft-west-let-localhost-be-localhost-05

- Updated obsolete references to RFC 5735 and 5156 in favor of [RFC6890].
- Clarify that non-caching recursive DNS servers are also addressed by #4 in Section 3.
- Reformulating the abstract and introduction based on feedback like Ted Lemon's in https://www.ietf.org/mail-archive/web/dnsop/current/msg20757.html
- Added a request that an insecure delegation for "localhost." be added to the root-zone.

B.6. draft-west-let-localhost-be-localhost-04

- Restructured the draft as a stand-alone document, rather than as set of monkey-patches against [RFC6761].

B.7. draft-west-let-localhost-be-localhost-03

- Explicitly referenced [I-D.ietf-sunset4-gapanalysis].
- Added a prohibition against using searchlists to resolve localhost names.
- Noted that MySQL has special behavior differentiating the connection mechanism used for "localhost" and "127.0.0.1".

B.8. draft-west-let-localhost-be-localhost-02

- Pulled in definitions for IPv4 and IPv6 loopback addresses.

B.9. draft-west-let-localhost-be-localhost-01

- Added a requirement that caching DNS servers MUST generate an immediate negative response.

B.10. draft-west-let-localhost-be-localhost-00

First draft.

Appendix C. Acknowledgements

Ryan Sleevi and Emily Stark informed me about the strange state of localhost name resolution. Erik Nygren poked me to take another look at the set of decisions we made in [SECURE-CONTEXTS] around "localhost."; this document is the result. They, along with Warren Kumari, Ted Lemon, John Levine, Mark Andrews, and many other members of DNSOP offered substantive feedback that markedly improved the quality of this document.

Author’s Address

Mike West
Google, Inc

Email: mkwst@google.com
URI: https://mikewest.org/