Abstract

This document registers the ".onion" Special-Use Domain Name.

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 December 21, 2015.

Copyright Notice

Copyright (c) 2015 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust’s Legal Provisions Relating to IETF Documents (http://trustee.ietf.org/license-info) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.
1. Introduction

The Tor network [Dingledine2004] has the ability to host network services using the "\.onion" Special-Use Top-Level Domain. Such addresses can be used as other domain names would be (e.g., in URLs [RFC3986]), but instead of using the DNS infrastructure, .onion names functionally correspond to the identity of a given service, thereby combining location and authentication.

In this way, .onion names are "special" in the sense defined by [RFC6761] Section 3; they require hardware and software implementations to change their handling, in order to achieve the desired properties of the name (see Section 4). These differences are listed in Section 2.

Like Top-Level Domain Names, .onion addresses can have an arbitrary number of subdomain components. This information is not meaningful to the Tor protocol, but can be used in application protocols like HTTP [RFC7230].

See [tor-address] and [tor-rendezvous] for the details of the creation and use of .onion names.

1.1. Notational Conventions

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].

2. The ".onion" Special-Use Domain Name

These properties have the following effects upon parties using or processing .onion names (as per [RFC6761]):
1. Users: human users are expected to recognize .onion names as having different security properties, and also being only available through software that is aware of onion addresses.

2. Application Software: Applications (including proxies) that implement the Tor protocol MUST recognize .onion names as special by either accessing them directly, or using a proxy (e.g., SOCKS [RFC1928]) to do so. Applications that do not implement the Tor protocol SHOULD generate an error upon the use of .onion, and SHOULD NOT perform a DNS lookup.

3. Name Resolution APIs and Libraries: Resolvers MUST either either respond to requests for .onion names by resolving them according to [tor-rendezvous] or by responding with NXDOMAIN.

4. Caching DNS Servers: Caching servers SHOULD NOT attempt to look up records for .onion names. They MUST generate NXDOMAIN for all such queries.

5. Authoritative DNS Servers: Authoritative servers MUST respond to queries for .onion with NXDOMAIN.

6. DNS Server Operators: Operators MUST NOT configure an authoritative DNS server to answer queries for .onion. If they do so, client software is likely to ignore any results (see above).

7. DNS Registries/Registrars: Registrars MUST NOT register .onion names; all such requests MUST be denied.

3. IANA Considerations

This document registers "onion" in the registry of Special-Use Domain Names [RFC6761]. See Section 2 for the registration template.

4. Security Considerations

.onion names are often used to provide access to end to end encrypted, secure, anonymized services; that is, the identity and location of the server is obscured from the client. The location of the client is obscured from the server. The identity of the client may or may not be disclosed through an optional cryptographic authentication process.

These properties can be compromised if, for example:

- The server "leaks" its identity in another way (e.g., in an application-level message), or
The access protocol is implemented or deployed incorrectly, or
The access protocol itself is found to have a flaw.

.onion names are self-authenticating, in that they are derived from
the cryptographic keys used by the server in a client verifiable
manner during connection establishment. As a result, the
cryptographic label component of a .onion name is not intended to be
human-meaningful.

The Tor network is designed to not be subject to any central
controlling authorities with regards to routing and service
publication, so .onion names cannot be registered, assigned,
transferred or revoked. "Ownership" of a .onion name is derived
solely from control of a public/private key pair which corresponds to
the algorithmic derivation of the name.

Users must take special precautions to ensure that the .onion name
they are communicating with is correct, as attackers may be able to
find keys which produce service names that are visually or
semantically similar to the desired service.

Also, users need to understand the difference between a .onion name
used and accessed directly via Tor-capable software, versus .onion
subdomains of other top-level domain names and providers (e.g., the
difference between example.onion and example.onion.tld).

The cryptographic label for a .onion name is constructed by applying
a function to the public key of the server, the output of which is
rendered as a string and concatenated with the string ".onion".
Dependent upon the specifics of the function used, an attacker may be
able to find a key that produces a collision with the same .onion
name with substantially less work than a cryptographic attack on the
full strength key. If this is possible the attacker may be able to
impersonate the service on the network.

If client software attempts to resolve a .onion name, it can leak the
identity of the service that the user is attempting to access to DNS
resolvers, authoritative DNS servers, and observers on the
intervening network. This can be mitigated by following the
recommendations in Section 2.

5. References
5.1. Normative References


5.2. Informative References


Appendix A. Acknowledgements

Thanks to Roger Dingledine, Linus Nordberg, and Seth David Schoen for their input and review.

This specification builds upon previous work by Christian Grothoff, Matthias Wachs, Hellekin O. Wolf, Jacob Appelbaum, and Leif Ryge to register .onion in conjunction with other, similar Special-Use Top-Level Domain Names.
Authors’ Addresses

Jacob Appelbaum
The Tor Project, Inc

Email: jacob@appelbaum.net

Alec Muffett
Facebook

Email: alecm@fb.com