Abstract

This document registers a set of Special-Use Domain Names for use with Peer-to-Peer (P2P) systems, as per RFC6761.

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 28, 2015.

Copyright Notice

Copyright (c) 2014 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
1. Introduction

The Domain Name System (DNS) is primarily used to map human-memorable names to IP addresses, which are used for routing but generally not meaningful for humans.

Peer-to-Peer (P2P) systems use specific decentralized mechanisms to allocate, register, manage, and resolve names. However, the hierarchical nature of DNS makes it unsuitable for various P2P Name Systems. Such P2P Name Systems operate entirely outside of DNS, independently from the DNS root and delegation tree.

As compatibility with applications using domain names is desired, these P2P overlay networks often define exclusive alternative Top-Level Domains to avoid conflict between the P2P namespace and the DNS hierarchy.

In order to avoid interoperability issues with DNS as well as to address security and privacy concerns, this document registers a set of Special-Use Domain Names for use with P2P systems (pTLDs), as per [RFC6761]: "GNU", "ZKEY", "ONION", "EXIT", "I2P", and "BIT".
The GNU Name System (GNS) ("GNU", "ZKEY"), the Tor network ("ONION", "EXIT"), the Invisible Internet Project ("I2P"), and the Dot-Bit Project ("BIT") use these pTLDs to realize fully-decentralized and censorship-resistant naming. The "EXIT" pTLD is used to control overlay routing and to securely specify path selection choices [TOR-PATH].

2. Applicability

[RFC6761] Section 3 states:

"[I]f a domain name has special properties that affect the way hardware and software implementations handle the name, that apply universally regardless of what network the implementation may be connected to, then that domain name may be a candidate for having the IETF declare it to be a Special-Use Domain Name and specify what special treatment implementations should give to that name. On the other hand, if declaring a given name to be special would result in no change to any implementations, then that suggests that the name may not be special in any material way, and it may be more appropriate to use the existing DNS mechanisms [RFC1034] to provide the desired delegation, data, or lack-of-data, for the name in question. Where the desired behaviour can be achieved via the existing domain name registration processes, that process should be used. Reservation of a Special-Use Domain Name is not a mechanism for circumventing normal domain name registration processes."

The set of Special-Use Domain Names for Peer-to-Peer Systems (pTLDs) reserved by this document meet this requirement, as they share the following specificities:

- pTLDs cannot be allocated administratively. Instead they use a variety of assignment mechanisms introduced in this document, and their respective protocol specifications: automated cryptographic assignment (".onion", ".zkey"), user-controlled assignment in a private scope (".gnu", ".i2p"), or in a global public ledger (".bit"), or used as a source-routing mechanism to delegate DNS resolution to a remote peer (".exit").

- pTLDs do not depend on the DNS tree hierarchy for their resolution: GNS and Namecoin domains MAY use it, as they return DNS-compatible results; and all pTLDs use specific P2P protocols for regular name resolution, covered by their respective protocol specifications.

- The appropriate pTLD protocols can be implemented in existing software libraries and APIs to extend regular DNS operation and
enable P2P name resolution. However the default hierarchical DNS response to any request to any pTLD MUST be NXDOMAIN.

Finally, in order for pTLDs to realize a censorship-resistant, fully-decentralized name system, and provide security and privacy features matching user expectations, this document specifies changes required in existing DNS software and DNS operations.

3. Terminology and Conventions Used in This Document

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].

The word "peer" is used in the meaning of an individual system on the network. Thus, "local peer" means the localhost.

The abbreviation "pTLD" is used in this document to mean a pseudo Top-Level Domain, i.e., a Special-Use Domain Name per [RFC6761] reserved to P2P Systems in this document. A pTLD is mentioned in capitals, and within double quotes to mark the difference with a regular DNS gTLD.

In this document, ".tld" (lowercase, with quotes) means: any domain or hostname within the scope of a given pTLD, while .tld (lowercase, without quotes) refers to an adjective form. For example, a collection of ".gnu" peers in "GNU", but an .onion URL. [TO REMOVE: in the IANA Considerations section, we use the simple .tld format to request TLD reservation for consistency with previous RFCs].

The word "NXDOMAIN" refers to an alternate expression for the "Name Error" RCODE as described in section 4.1.1 of [RFC1035].

The Tor-related names such as 'circuit', 'exit', 'node', 'relay', 'stream', and related Tor terms are described in [Dingledine2004] and the Tor protocol specification [TOR-PROTOCOL].

4. Description of Special-Use Domains in P2P Networks

4.1. The "GNU" Relative pTLD

"GNU" is used to specify that a domain name should be resolved using GNS. The GNS resolution process is documented in [Wachs2014].

The "GNU" domain is special in the following ways:
1. Users can use these names as they would other domain names, entering them anywhere that they would otherwise enter a conventional DNS domain name.

Since there is no central authority responsible for assigning .gnu names, and that specific domain is local to the local peer, users need to be aware of that specificity.

In any case, resolution of .gnu domains returns DNS compatible results, and thus should not affect normal usage of most Internet applications.

2. Application software SHOULD NOT recognize .gnu domains as special, and SHOULD use these names as they would other domain names.

Applications SHOULD expect DNS resolution to return NXDOMAIN for .gnu names, and MAY treat other responses as errors.

GNS-aware applications MAY also use GNS resolvers directly (in particular, when a DNS API or library responds with NXDOMAIN).

3. Name resolution APIs and libraries SHOULD either respond to requests for .gnu names by resolving them via the GNS protocol, or respond with NXDOMAIN.

4. Caching DNS servers SHOULD recognize .gnu names as special and SHOULD NOT attempt to look up NS records for them, or otherwise query authoritative DNS servers in an attempt to resolve .gnu names. Instead, caching DNS servers SHOULD generate immediate negative responses for all such queries.

5. Authoritative DNS servers are not expected to treat .gnu domain requests specially. In practice, they MUST answer with NXDOMAIN, as "GNU" is not available via global DNS resolution, and not doing so can put users' privacy at risk (see item 6).
6. DNS server operators SHOULD be aware that .gnu names are reserved for use with GNS, and MUST NOT override their resolution (e.g., to redirect users to another service or error information).

7. DNS registries/registrars MUST NOT grant any request to register .gnu names. This helps avoid conflicts [SAC45]. These names are defined by the GNS protocol specification, and they fall outside the set of names available for allocation by registries/registrars.

4.2. The "ZKEY" Compressed Public Key pTLD

The "ZKEY" pTLD is used to signify that resolution of the given name MUST be performed using a record signed by an authority that is in possession of a particular public key. Names in "ZKEY" MUST end with a domain which is the compressed point representation from [EdDSA] on [Curve25519] of the public key of the authority, encoded using Crockford's variant of base32hex [RFC4648] (with additionally 'U' being considered equal to 'V') for easier optical character recognition. A GNS resolver uses the key to locate a record signed by the respective authority.

"ZKEY" provides a (reverse) mapping from globally unique hashes to public key, therefore .zkey names are non-memorable, and are expected to be hidden from the user [Wachs2014].

The "ZKEY" domain is special in the following ways:

1. Users can use these names as they would other domain names, entering them anywhere that they would otherwise enter a conventional DNS domain name.

   Since there is no central authority necessary or possible for assigning .zkey names, and those names match cryptographic keys, users need to be aware that they do not belong to regular DNS, but are still global in their scope.

   In any case, resolution of .zkey domains returns DNS-compatible results, and thus should not affect normal usage of most Internet applications.
2. Application software SHOULD NOT recognize .zkey domains as special, and SHOULD use these names as they would other domain names.

Applications SHOULD expect DNS resolution to return NXDOMAIN for .zkey names, and MAY treat other responses as errors.

GNS-aware applications MAY also use GNS resolvers directly.

3. Name resolution APIs and libraries SHOULD either respond to requests for .zkey names by resolving them via the GNS protocol, or respond with NXDOMAIN.

4. Caching DNS servers SHOULD recognize .zkey names as special and SHOULD NOT attempt to look up NS records for them, or otherwise query authoritative DNS servers in an attempt to resolve .zkey names. Instead, caching DNS servers SHOULD generate immediate negative responses for all such queries.

5. Authoritative DNS Servers are not expected to treat .zkey domain requests specially. In practice, they MUST answer with NXDOMAIN, as "ZKEY" is not available via global DNS resolution, and not doing so MAY put users' privacy at risk (see item 6).

6. DNS server operators SHOULD be aware that .zkey names are reserved for use with GNS, and MUST NOT override their resolution (e.g., to redirect users to another service or error information).

7. DNS registries/registrars MUST NOT grant any request to register .zkey names. This helps avoid conflicts [SAC45]. These names are defined as described above, and they fall outside the set of names available for allocation by registries/registrars.
4.3. Geographically Anonymous pTLDs

The Tor anonymization network makes use of several special pTLD labels, three of which have seen widespread usage to date [TOR-ADDRESS].

4.3.1. The "ONION" Hidden Service pTLD

The widely deployed "ONION" designates an anonymous Tor Hidden Service reachable via the Tor network [Dingledine2004]. These .onion hostnames are self-authenticating addresses for use with any TCP service. The purpose of the Tor Hidden Services system is to provide geographic anonymity for the .onion host and for all clients visiting the hidden service as well as other purposes such as NAT traversal, strong authentication, anonymity and censorship resistance.

Addresses in "ONION" are opaque, non-mnemonic, alpha-semi-numeric digest hashes corresponding to the unique identity key of a given Tor hidden service. Therefore such .onion addresses are self-authenticating. The algorithm to obtain the .onion hash from the Tor hidden service's public key is out of scope of this document, and described in the Tor Address specification [TOR-ADDRESS]. Tor generates this "Onion key" automatically when the hidden service is configured. Tor clients use it following the Tor Rendezvous specifications [TOR-RENDEZVOUS].

The "ONION" domain is special in the following ways:

1. Users can use these names as they would other domain names, entering them anywhere that they would otherwise enter a conventional DNS domain name.

Since there is no central authority necessary or possible for assigning .onion names, and those names correspond to cryptographic keys, users need to be aware that they do not belong to regular DNS, but are still global in their scope.

2. Application software MAY recognize .onion domains as special, and SHOULD use these names as they would other domain names.

Application software MAY implement mechanisms helping the user to ensure their privacy expectations are met, such as warning the user if they do not detect an active local Tor resolver, displaying a warning on first-use of an .onion domain to explain the necessity of a Tor resolver to reach such domains, etc.
If an application knows how to differentiate between DNS and P2P name resolution, it:

* SHOULD NOT pass requests for .onion domains to DNS resolvers or libraries,

* MUST expect NXDOMAIN as the only valid DNS response, and

* SHOULD treat other answers from DNS as errors.

Tor-aware applications MAY also use Tor resolvers directly.

3. Name resolution APIs and libraries SHOULD either respond to requests for .onion names by resolving them via the Tor protocol, or respond with NXDOMAIN.

4. Caching DNS servers SHOULD recognize .onion names as special and SHOULD NOT attempt to look up NS records for them, or otherwise query authoritative DNS servers in an attempt to resolve .onion names. Instead, caching DNS servers SHOULD generate immediate negative responses for all such queries.

5. Authoritative DNS servers are not expected to treat .onion domain requests specially. In practice, they MUST answer with NXDOMAIN, as "ONION" is not available via global DNS resolution, and not doing so MAY put users' privacy at risk (see item 6).

6. DNS server operators SHOULD be aware that .onion names are reserved for use with Tor, and MUST NOT override their resolution (e.g., to redirect users to another service or error information).

7. DNS registries/registrars MUST NOT grant any request to register .onion names. This helps avoid conflicts [SAC45]. These names are defined the Tor protocol specification [TOR-PROTOCOL], and they fall outside the set of names available for allocation by registries/registrars.
4.3.2. The "EXIT" Client Source Routing pTLD

The .exit suffix is used as an in-band source routing control channel, usually for selection of a specific Tor relay during path creation as the last node in the Tor circuit.

It may be used to access a DNS host via specific Tor servers, in the form "hostname.nickname-or-fingerprint.exit", where the "hostname" is a valid hostname, and the "nickname-or-fingerprint" is either the nickname of a Tor relay in the Tor network consensus, or the hex-encoded SHA1 digest of the given node's public key (fingerprint).

For example, "gnu.org.noisetor.exit" will route the client to "gnu.org" via the Tor node nicknamed "noisetor". Using the fingerprint instead of the nickname ensures that the path selection uses a specific Tor exit node, and is harder to remember: e.g., "gnu.org.f97f3b153fed604230cd497a3d1e0815b007637.exit".

When Tor sees an address in this format, it uses the specified "nickname-or-fingerprint" as the exit node. If no "hostname" component is given, Tor defaults to the published IPv4 address of the Tor exit node [TOR-EXTSOCKS].

Because "hostname" is allegedly valid, the total length of a .exit construct may exceed the maximum length allowed for domain names. Moreover, the resolution of "hostname" happens at the exit node. Trying to resolve such invalid domain names, including chaining .exit names will likely return a DNS lookup failure at the first exit node.

The "EXIT" domain is special in the following ways:

1. Users can use these names as they would other domain names, entering them anywhere that they would otherwise enter a conventional DNS domain name.

   Since .exit names correspond to a Tor-specific routing construct to reach target hosts via chosen Tor exit nodes, users need to be aware that they do not belong to regular DNS and that the actual target precedes the second-level domain name.

2. Application software MAY recognize that .exit domains are special and when they do SHOULD NOT pass requests for these domains to DNS resolvers and libraries.

   As mentioned in items 4 and 5 below, regular DNS resolution is expected to respond with NXDOMAIN. Therefore, if it can
differentiate between DNS and P2P name resolution, application software:

* MUST expect NXDOMAIN as the only valid DNS response, and
* SHOULD treat other answers from DNS as errors.

Tor-aware applications MAY also use Tor resolvers directly.

3. Name resolution APIs and libraries SHOULD either respond to requests for .exit names by resolving them via the Tor protocol, or respond with NXDOMAIN.

4. Caching DNS servers SHOULD recognize .exit names as special and SHOULD NOT, by default, attempt to look up NS records for them, or otherwise query authoritative DNS servers in an attempt to resolve .exit names. Instead, caching DNS servers SHOULD, by default, generate immediate negative responses for all such queries.

5. Authoritative DNS servers are not expected to treat .exit domain requests specially. In practice, they MUST answer with NXDOMAIN, as "EXIT" is not available via global DNS resolution, and not doing so MAY put users' privacy at risk (see item 6).

6. DNS server operators SHOULD be aware that .exit names are reserved for use with Tor, and MUST NOT override their resolution (e.g., to redirect users to another service or error information).

7. DNS registries/registrars MUST NOT grant any request to register .exit names. This helps avoid conflicts [SAC45]. These names are defined by the Tor address specification, and they fall outside the set of names available for allocation by registries/registrars.
4.4. The "I2P" Addressbook pTLD

"I2P" provides accessibility to anonymous services ("eepsites") within the I2P network. I2P is a scalable, self-organizing, resilient packet switched anonymous network layer, upon which any number of different anonymity or security-conscious applications can operate.

The local I2P proxy resolves such names either by looking up a local table called the addressbook, or by decoding Base32-encoded [RFC4648] public keys and establishing a tunnel to the respective authority, similar to contacting .onion hidden services. The details of I2P's operation [I2P-NAMING] are outside of the scope of this document.

As the system is decentralized, "example.i2p" may also resolve differently for different peers, depending on the state of their respective addressbooks.

The "I2P" domain is special in the following ways:

1. Users can use these names as they would other domain names, entering them anywhere that they would otherwise enter a conventional DNS domain name.

   Since there is no central authority responsible for assigning .i2p names, and that the ultimate mapping is decided by the local peer, users need to be aware of that specificity.

2. Application software SHOULD recognize .i2p domains as special and SHOULD NOT use them as they would other domains.

   Applications SHOULD NOT pass requests for .i2p domains to DNS resolvers and libraries.

   As mentioned in points 4 and 5 below, regular DNS resolution is expected to respond with NXDOMAIN. Therefore, if it can differentiate between DNS and P2P name resolution, application software can expect such a response, and can choose to treat other responses from resolvers and libraries as errors.

3. Name resolution APIs and libraries SHOULD either respond to requests for .i2p names by resolving them via the I2P protocol, or respond with NXDOMAIN.
4. Caching DNS servers SHOULD recognize .i2p names as special and SHOULD NOT attempt to look up NS records for them, or otherwise query authoritative DNS servers in an attempt to resolve .i2p names. Instead, caching DNS servers SHOULD generate immediate negative responses for all such queries.

5. Authoritative DNS servers are not expected to treat .i2p domain requests specially. In practice, they MUST answer with NXDOMAIN, as "I2P" is not available via global DNS resolution, and not doing so MAY put users' privacy at risk (see item 6).

6. DNS server operators SHOULD be aware that .i2p names are reserved for use with I2P, and MUST NOT override their resolution (e.g., to redirect users to another service or error information).

7. DNS registries/registrars MUST NOT grant any request to register .i2p names. This helps avoid conflicts [SAC45]. These names are defined by the I2P protocol specification, and they fall outside the set of names available for allocation by registries/registrars.

4.5. The "BIT" Timeline System pTLD

"BIT" provides a name space where names are registered via transactions in the Namecoin currency [Namecoin]. Like Bitcoins, Namecoins are created using a proof-of-work calculation, which is also used to establish a decentralized, multi-party consensus on the valid transaction history, and thus the set of registered names and their values [SquareZooko].

The Namecoin used in a transaction to register a name in "BIT" is lost. This is not a fundamental problem as more coins can be generated via mining (proof-of-work calculations). The registration cost is set to decrease over time, to prevent early adopters from registering too many names.

The owner of a name can update the associated value by issuing an update, which is a transaction that uses a special coin which is generated as change during the registration operation. If a name is not updated for a long time, the registration expires.
The "BIT" domain is special in the following ways:

1. Users can use these names as they would other domain names, entering them anywhere that they would otherwise enter a conventional DNS domain name.

   From the user's perspective, the resolution of .bit names is similar to the normal DNS resolution, and thus should not affect normal usage of most Internet applications.

2. Application software SHOULD NOT recognize .bit domains as special and SHOULD treat them as they would other domains.

   Applications MAY pass requests to the "BIT" pTLD to DNS resolvers and libraries if A/AAAA records are desired. If available, the local resolver can intercept such requests within the respective operating system hooks and return DNS-compatible results.

   Namecoin-aware applications MAY choose to talk directly to the respective P2P resolver, and use this to access additional record types that are not defined in DNS.

3. Name resolution APIs and libraries SHOULD either respond to requests for .bit names by resolving them via the Namecoin protocol, or respond with NXDOMAIN.

4. Caching DNS servers SHOULD recognize .bit names as special and SHOULD NOT attempt to resolve them. Instead, caching DNS servers SHOULD generate immediate negative responses for all such queries.

   Given that .bit users typically have no special privacy expectations, and those names are globally unique, local caching DNS servers MAY choose to treat them as regular domain names, and cache the responses obtained from the Namecoin blockchain. In that case however, NXDOMAIN results SHOULD NOT be cached, as new .bit domains may become active at any time.
5. Authoritative DNS servers are not expected to treat .bit domain requests specially. In practice, they MUST answer with NXDOMAIN, as "BIT" is not available via global DNS resolution.

6. DNS server operators SHOULD be aware that .bit names are reserved for use with Namecoin, and MUST NOT override their resolution (e.g., to redirect users to another service or error information).

7. DNS registries/registrars MUST NOT grant any request to register .bit names. This helps avoid conflicts. These names are defined by the Namecoin protocol specification, and they fall outside the set of names available for allocation by registries/registrars.

5. Security Considerations

Specific software performs the resolution of the six Special-Use Domain Names presented in this document; this resolution process happens outside of the scope of DNS. Leakage of requests to such domains to the global operational DNS can cause interception of traffic that might be misused to monitor, censor, or abuse the user's trust, and lead to privacy issues with potentially tragic consequences for the user.

This document reserves these Top-Level Domain names to minimize the possibility of confusion, conflict, and especially privacy risks for users.

The reality of X.509 Certificate Authorities (CAs) creating misleading certificates for these pTLDs due to ignorance stresses the need to document their special use. X.509 Certificate Authorities MAY create certificates for "ONION", "BIT", and "ZKEY" given CSRs signed with the respective private keys encoded in the respective names. For "BIT", the Certificate Authority SHOULD limit the expiration time of the certificate to match the registration. Certificate Authorities MUST NOT create certificates for the "EXIT", "GNU", and "I2P" Top-Level domains. Nevertheless, clients SHOULD accept certificates for these Top-Level domains as they may be created legitimately by local proxies on the fly.
[SAC57] reports, page 11, that the CA/Browser forum stated: "Also as of the Effective Date [1 July 2012], the CA SHALL NOT issue a certificate with an Expiry Date later than 1 November 2015 with a subjectAlternativeName extension or Subject commonName field containing a Reserved IP Address or Internal Server Name."

It is not clear whether e.g., .onion sites are considered "Internal Server Names", however, we can expect that services doubling their public Web site with an onion site would use a single SSL certificate for both, as did Facebook with "facebookcorewww1.onion". Given this forum also declared the CAs would revoke such SSL certificates in October 2016, that opens a three (now two) years period of vulnerability for new gTLDs to suffer MiTM attacks over HTTPS. Such practice by CAs to validate certificates to invalid TLDs without verification may lead, e.g., to malicious third parties without any relation to an existing onion site to register a fake certificate for that site in order to facilitate attacks, especially when combined with name collision risk as explained in [SAC62].

6. IANA Considerations

The Internet Assigned Numbers Authority (IANA) reserved the following entries in the Special-Use Domain Names registry [RFC6761]:

- .gnu
- .zkey
- .onion
- .exit
- .i2p
- .bit

[TO REMOVE: the assignement URL is https://www.iana.org/assignments/special-use-domain-names/]

7. Acknowledgements

The authors thank the I2P developers for their constructive feedback, as well as Mark Nottingham for his proof-reading and valuable feedback.
8. References

8.1. Normative References


8.2. Informative References


[Wachs2014]

Authors' Addresses

Christian Grothoff
INRIA
Equipe Decentralisee
INRIA Rennes Bretagne Atlantique
263 avenue du General Leclerc
Campus Universitaire de Beaulieu
Rennes, Bretagne F-35042
FR

Email: christian@grothoff.org

Matthias Wachs
Technische Universitaet Muenchen
Free Secure Network Systems Group
Lehrstuhl fuer Netzarchitekturen und Netzdienste
Boltzmannstrasse 3
Technische Universitaet Muenchen
Garching bei Muenchen, Bayern D-85748
DE

Email: wachs@net.in.tum.de

Hellekin O. Wolf (editor)
GNU consensus

Email: hellekin@gnu.org

Jacob Appelbaum
Tor Project Inc.

Email: jacob@appelbaum.net

Leif Ryge
Tor Project Inc.

Email: leif@synthesize.us