Workgroup:	Internet Engineering Task Force
Internet-Draft:	draft-augustyn-intarea-ipref-framework-00
Published:	14 January 2023
Intended Status:	Informational
Expires:	18 July 2023
Author:	W. Augustyn, Ed.

Framework for IP Addressing with References (IPREF)

Abstract

This document describes IP addressing with references (referred to as IPREF) and how it can be used with IPv4 and IPv6. IPREF is a private-to-private technology where hosts on private networks communicate with hosts on other private networks directly. Special addresses, called IPREF addresses, are used for the purpose. It also describes how hosts on private networks may publish their IPREF addresses via Domain Name System (DNS).

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 18 July 2023.

Copyright Notice

Copyright (c) 2023 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 (https://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 Revised BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Revised BSD License.

Table of Contents

1. Introduction
   1.1. Requirements Language
   1.2. IPREF Terminology
2. Overview
   2.1. IPREF Addresses
   2.2. Packet Exchange
   2.3. DNS with IPREF
3. IPREF Reference
4. IPREF Address
5. Embedding References in IP Packets
   5.1. IPv4 Option
   5.2. IPv6 Extension Header
   5.3. UDP Tunnel
6. Distributing IPREF Addresses
   6.1. DNS Records
   6.2. Local Network Resolver
   6.3. DNS Agent
7. Related Technologies
8. Distinct Properties
9. IANA Considerations
10. Security Considerations
11. References
   11.1. Normative References
   11.2. Informative References

Appendix A. Appendix 1

Acknowledgements

Contributors

Author's Address

1. Introduction

Normally, hosts on private networks are only reachable by other hosts on the same private networks. To make them visible to other hosts, techniques like NAT, popular with IPv4 private networks, or filtering, more likely found with IPv6 private networks, are used to make them appear on the public Internet. In most cases, only selected services, determined by their layer 4 port numbers, are made available through these methods. Services from different private hosts may share a single public address.

This document describes a different way of accessing hosts on private networks. It is done by enhancing capabilities of existing layer 3 protocols. The enhancement provides for addressing with references where the source and destination is specified by means of references rather than concrete addresses. The respective private networks, communicating in this manner, use these references to render actual addresses on their local networks. Reference are single values, unsigned integers, which are carried in addition to the existing addresses by the layer 3 protocols.

In theory, any layer 3 protocol can operate in the manner described. For practical purpose, this document discusses only the case of IPv4 and IPv6, and only with the Internet connecting communicating private networks.

1.1. Requirements Language

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.

1.2. IPREF Terminology

IPREF - short name referring to the technology described in this document, pronounced I-P-REF

third network - network connecting communicating private networks, e.g.: the Internet

encoding network - network representing hosts on peer private network

2. Overview

IP addressing with references (IPREF) is intended for communication between private networks connected via a *third* network, typically the Internet. The references are opaque integers that factor in calculating of actual addresses at the respective private networks. References are not

direct IP addresses. Each private network makes their calculations independently irrespective of how their peers might be interpreting the same references. The resulting addresses are normally different (they may be the same by chance) and are not known to the peers.

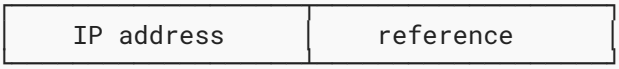
IPREF deals with private networks, there is no intention to communicate directly with 'public' hosts, i.e. hosts with addresses on the *third* network. There is no need to. These 'public' hosts normally reside on private networks anyway. They are made available on the 'public' network using techniques such as NAT (Network Address Translation) or filtering. Since they reside on private networks, they can be reached through IPREF directly without NAT mapping or creating special filters.

In case of IPv6 private networks, local addresses may be globally routable. In spite of that, they may or may not be reachable from peer private networks. IPREF is not concerned with that, it works the same way regardless. Similarly, IPv4 private networks may include hosts with globally routable addresses, it makes no difference. Peer networks still don't know, and don't need to know, what those addresses are and to which protocol they belong.

IPREF is a form of network address translation but it is never referred to it by this term to avoid confusion with well known NAT. It is always referred to as IPREF. It is an evolution of address rewriting technology.

2.1. IPREF Addresses

The references are used in conjunction with standard IP addresses from the third network. These are typically addresses of gateway interfaces where the packets arrive at or where they are being sent from. A combination of a *third* network IP address and the reference is called an IPREF address, Figure 1.




```
198.51.100.11+700
2001:db8::abc:11+700
```

Figure 1: IPREF address

In the notation, a plus sign '+' is used to separate the IP address from the reference. In a packet exchange, there is a source IPREF address and a destination IPREF address, thus two references. The destination reference is allocated by the destination private network while the source reference is allocated by the source private network. There are no negotiations. Each peer defines their own references and accept peer references as opaque values.

2.2. Packet Exchange

Figure 2 depicts how two local networks communicate with one another. Local network A has two standard hosts A5, A7, and a gateway GWA. The gateway connects to the *third* network, the Internet. Local network B has hosts B4, B6, and a gateway GWB. The gateway connects to the

same *third* network as gateway GWA. Only the gateways are aware of, and implement, IPREF processing. The hosts on both local networks, as well as any networks hosts, routers, or nodes on the *third* network are standard network devices, unaware of IPREF. For simplicity, the examples shows the case of IPv4 running on all three networks. The IP addresses on local networks may overlap but for simplicity are shown as distinct. Further, for ease of following the example, addresses related to local network A are odd while addresses related to local network B are even.

Hosts on local network A communicate with hosts on other local networks, such as local network B, without knowing the IP addresses on the peer network or even without knowing the network protocol employed by the peer network. An *encoding* network is used in lieu of the peer private network addresses. The peer hosts appear as if they were hosted on the *encoding* network. This is a compatibility feature. It is possible to avoid such mapping at the cost of making local hosts IPREF aware. That case is not described here as it is dramatically simpler to use *encoding* networks and leave existing hosts as-is without any modifications.

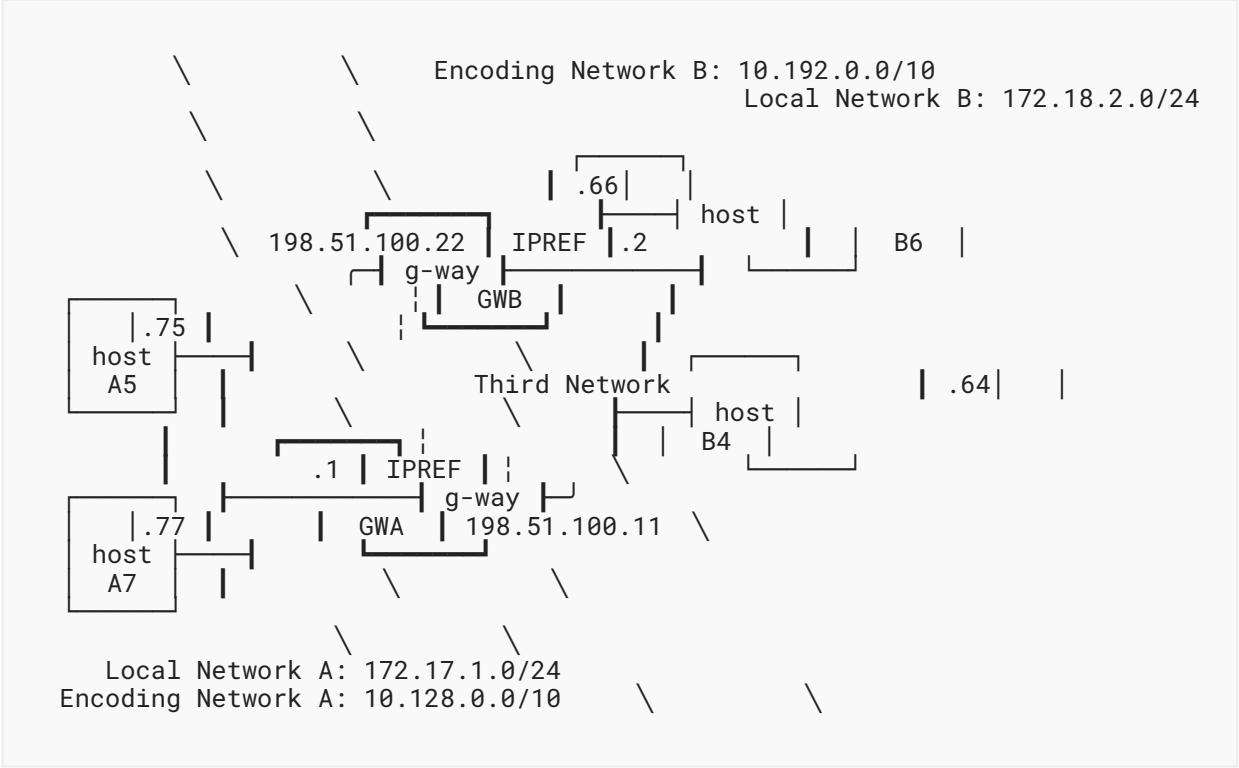


Figure 2: private networks

Let's assume host A5 wants to send a packet to host B4 and receive a response. Prior to the packet exchange, an administrator from local network B informs an administrator of local network A that host B can be reached at an IPREF address 198.151.100.22+400. Local administrator enters that information on gateway GWA which in turn allocates an *encoded* address of 10.128.0.40 to represent host B. This information is further passed to host A5 so that it can send packets to host B by using 10.128.0.40 as the destination.

In step (1) host A5 places its own address as source, 172.17.1.75 and the provided address of host B 10.128.0.40 as destination. As the packet traverses the network, both source and destination addresses will be rewritten. It is illustrated in Figure 3. In step (2), the packet arrives at gateway GWA. The gateway realizes the destination is an *encoded* address. It finds that the address corresponds to an IPREF address 198.51.100.22+400. It places this IPREF address in the packet as the new destination address. It also finds that the source address 172.17.1.75 does not correspond to any IPREF address so it allocates one through some algorithm, perhaps randomly, as 198.51.100.11+500. It places this IPREF address in the packet as the new source address. It then sends the packet into the *third* network. This is the common network, the Internet, that both local networks A and B connect to. In step (3) the packet arrives at gateway GWB. The gateway recognizes the destination IPREF address as representation of the internal host's B address 172.18.2.64. It places this address in the packet as new destination address. The gateway does not recognize the source IPREF address so it allocates an *encoded* address for it 10.192.0.70. It places this address in the packet as the new source address. It, then, sends the packet to the local host B4. In step (4), local host B4 receives the packet and recognizes the destination address as its own. It consumes the packet and processes it according to the payload.

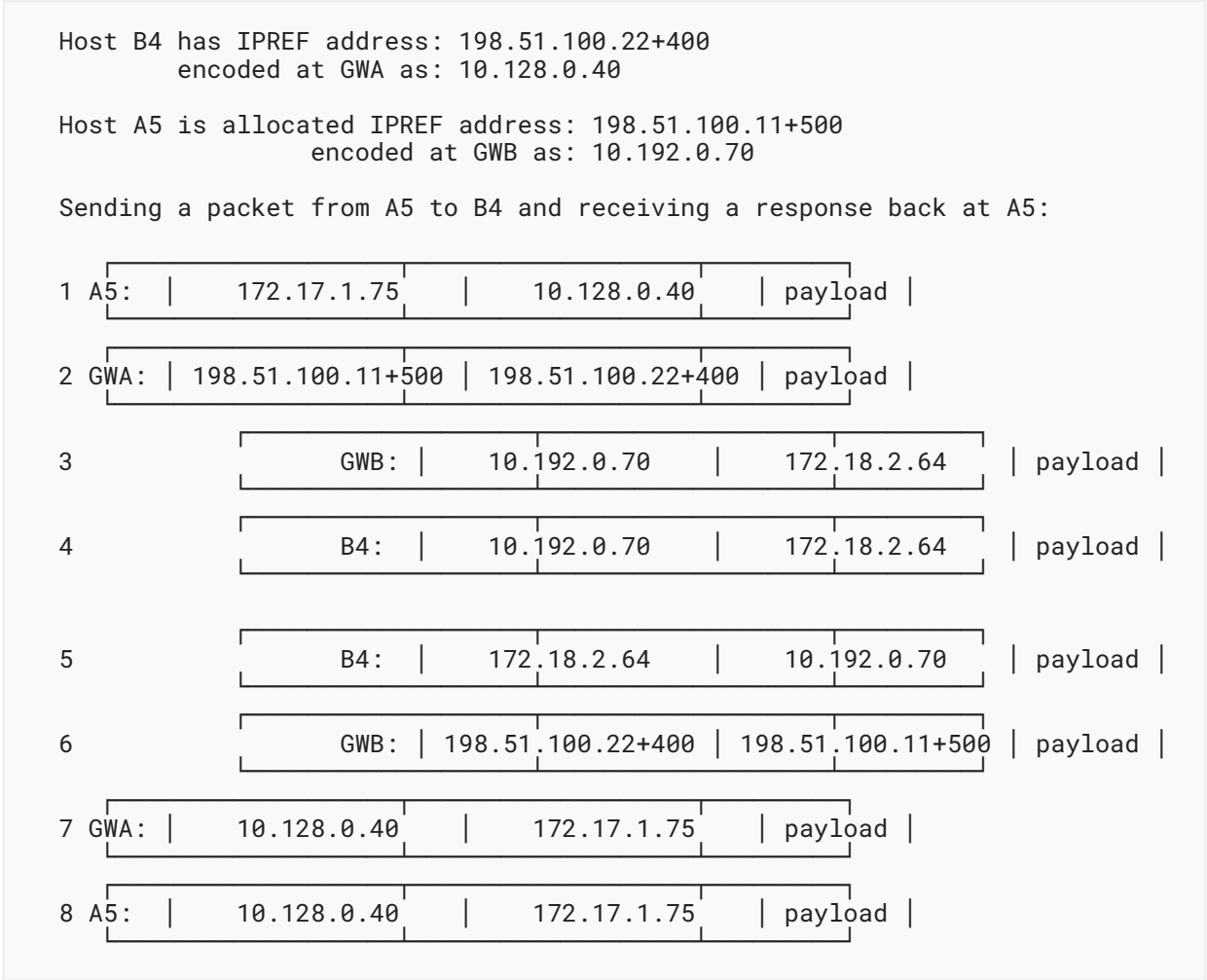


Figure 3: address rewriting

In step (5), host B4 sends a response back to host A5. I reverses source and destination addresses and sends a packet to gateway GWB. In step (6), the packet arrives at gateway GWB. The gateway recognizes the destination address as an *encoded* address previously created to represent IPREF address 198.51.100.11+500. It places this IPREF address in the packet as the new destination address. The gateway also recognizes the source address as corresponding to a preset IPREF address 198.51.100.22+400. It places this IPREF address as the new source address. It then sends the packet to the *third* network. In step (7), the packet arrives at gateway GWA. The gateway recognizes the destination IPREF address as corresponding to the local host A5 172.17.1.75. It places this address in the packet as the new destination address. It also recognizes the source IPREF address as one represented by an *encoded* address 10.128.0.40. It places that address in the packet as the new source address. It then sends the packet to the local host A5. in step (8), local host A5 recognizes the destination address 172.17.1.75 as its own and consumes the packet for processing. This concludes packet exchange.

In general case, networks may have overlapping addresses, they may have several local subnets, they may have more than one gateway to the *third* network. They may also run different network protocols. For example one private network may run IPv4 while its peer may run IPv6. The communicating networks do not need to know their peers' actual IP addresses or network protocols. The references are the stand ins representing those addresses in their native network protocol domains. The references are interpreted independently by both communicating networks. In the example above, network B allocates reference '400' to represent host B4. At network B, that reference is interpreted as 172.18.2.64 whereas that same reference is interpreted as 10.128.0.4 at network A. Neither network knows, or cares about, how the references are interpreted outside of their own administrative domains. In cases where interpretation involves changing network protocol, the local gateway must run dual stacks and perform proper repackaging of the packets. Of course, higher level protocol, such as TCP/UDP, must be compatible for this to be practical. IPv4 and IPv6 meet this criterion.

2.3. DNS with IPREF

Similarly to standard IP, IPREF can operate without DNS (Domain Name System) relying on IPREF addresses manually entered and distributed. For example, an /etc/hosts file could be used for the purpose. Such use makes sense in certain cases and is available with IPREF the same way as it is with standard IP. Especially on private networks, the practice of using direct addresses is common even if local domain names are allocated to the hosts.

IPREF can take full advantage of DNS. IPREF addresses are publishable. When resolving names that render IPREF addresses, local resolvers must obtain translation into *encoded* addresses for standard hosts on local networks. This is the normal case, only IPREF gateways understand IPREF addresses, the vast majority of hosts are standard hosts which are unaware of IPREF addressing. The necessary mapping to *encoded* addresses is typically made by IPREF gateways and presented back to resolvers which pass them to the querying hosts. In addition, IPREF gateways must be aware of all local hosts whose IPREF addresses are published through DNS. This is illustrated in Figure 4.

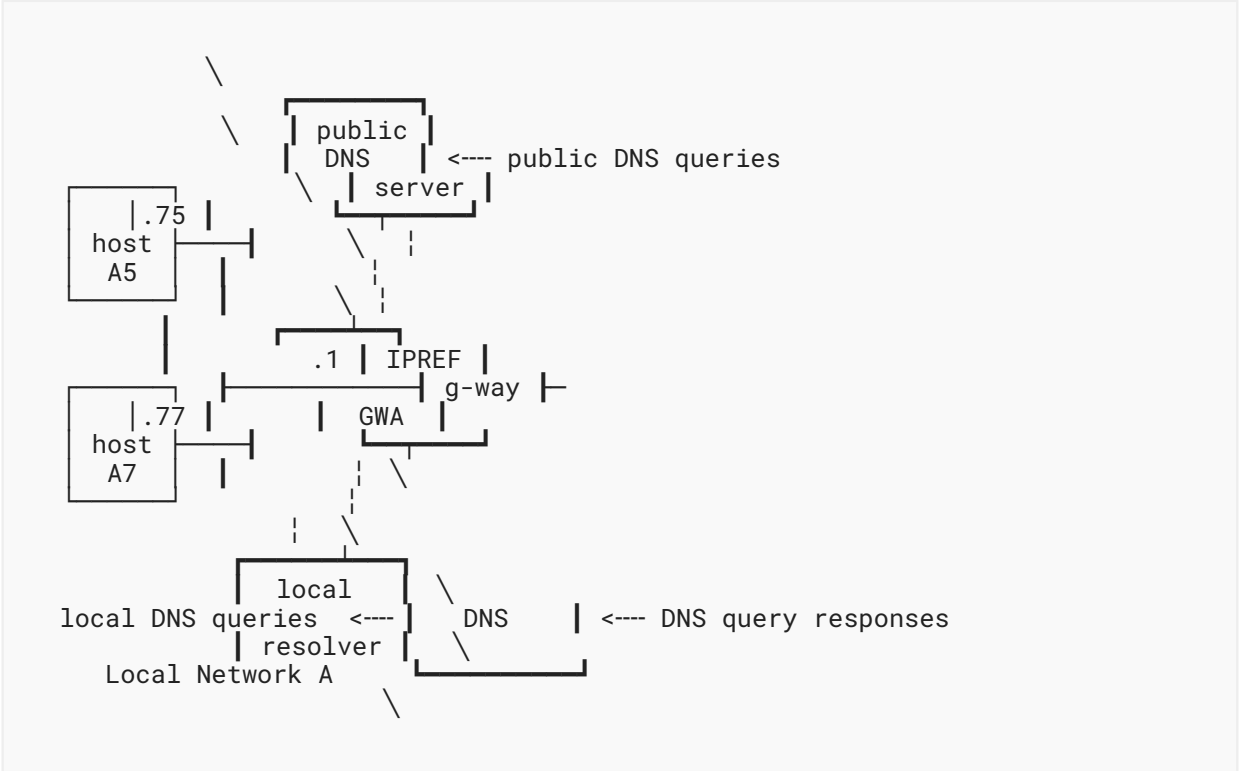


Figure 4: DNS with IPREF

Local hosts A5 and A7 resolve DNS names via a local resolver. The resolver queries DNS servers on their behalf. A query may return a standard IPv4 or IPv6 address, or it may return an IPREF address. In the former cases, the IP addresses are simply returned to the querying hosts. In case of IPREF addresses, the resolvers consults related IPREF gateway for a mapping to an *encoded* network. The gateway may allocate one on the fly if mapping is not available. The resulting *encoded* IP address is then returned to the querying hosts.

Local network administrator may decide to publish IPREF addresses of some internal hosts via DNS. The IPREF gateway must be aware which hosts are made available externally and what IPREF address have been assigned to them. Typically, this is a semi-static configuration which remains stable for long periods of time. The gateways may query those DNS servers for information periodically or some other mechanism may be employed to pass that information to the IPREF gateways.

Returning to the packet exchange scenario shown in Figure 3, the IPREF address of host B4 may be published via a public DNS server maintained by the administrator of local network B. In that case, host A5 may simply use a DNS name of that host and ask local resolver to provide the corresponding IP address. The resolver would query the DNS server, recognize the response as an IPREF address and consult local IPREF gateway for proper mapping to an *encoded* IP address. That IP address would then be returned to host A5 which in turn would use it as the destination IP address.

3. IPREF Reference

text ipref reference

4. IPREF Address

text ipref address

5. Embedding References in IP Packets

A reference is a network layer entity and the most natural place for embedding it would be a network layer header, such as an IPv4 option or an IPv6 extension header.

Unfortunately, many Internet Service Providers drop options and extensions headers that they deem not worthy processing. Even if they don't, network devices tend to put them on a slow processing path resulting in poor performance. For that reason, the most reliable way to embed references is to place them in the payload. One such common technique is tunneling where a reference could be placed in the payload along with the network packet being tunneled.

5.1. IPv4 Option

With IPv4, references may be embedded in an IPv4 header option. It would be a new option type. It would contain an octet with option type and option number, an octet with length, and two octets reserved for possible future use while padding to four octet boundary. The source and destination references would then follow.

A new option number would be registered with IANA. Until then, experimental option, 30, could be used. A separate document should describe it in detail.

5.2. IPv6 Extension Header

With IPv6, references may be embedded in an IPv6 extension header. It would be a new header type. It would contain an octet with next header type value, an octet with length, and two octets reserved for possible future use. This would be followed by four octets of padding to 8 octet boundary. Padding octets would not be intended for any future use. The source and destination references would then follow.

A new protocol type would be registered with IANA. Until then, experimental protocol, 254, could be used. A separate document should describe it in detail.

5.3. UDP Tunnel

Placing references in a UDP tunnel works for both IPv4 and IPv6. In both cases, the references are embedded in the form of respective option or extension header. In addition, a tunnel encapsulation record is added. The order of items is as follows:

UDP header

Tunnel encapsulation record

IPREF option

Packet payload

The encapsulation record would consist of four octets. First octet would contain the value of TTL copied from the original IP header. The second octet would contain protocol number copied from the original IPv4 header or next header value form an IPv6 extension header. The third octet would report number of hops detected for incoming packets. The fourth octet would be reserved for possible future use while padding to 4 octet boundary. The IPREF option would follow in the format matching respective IP protocol, IPv4 or IPv6.

The tunnel would operate at port 1045. This value would be registered with IANA. A separate document should describe the tunnel in detail.

6. Distributing IPREF Addresses

text distributing IPREF addresses

6.1. DNS Records

text dns records

6.2. Local Network Resolver

text local network resolver

6.3. DNS Agent

text dns agent

7. Related Technologies

text related technologies

8. Distinct Properties

text distinct properties

9. IANA Considerations

This memo includes no request to IANA.

10. Security Considerations

This document should not affect the security of the Internet.

11. References

11.1. Normative References

[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/info/rfc8174>.

11.2. Informative 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>.

[exampleRefMin] Surname, Initials., "Title", 2006.

[exampleRefOrg] Organization, "Title", 1984, <http://www.example.com/>.

Appendix A. Appendix 1

This becomes an Appendix

Acknowledgements

This template uses extracts from templates written by Pekka Savola, Elwyn Davies and Henrik Levkowetz.

Contributors

Thanks to all of the contributors.

Jane Doe
Acme
Email: jdoe@example.com

Author's Address

Waldemar Augustyn (EDITOR)
Email: waldemar@wdmsys.com