NETWORK Working Group
INTERNET-DRAFT
Category: Standards Track
20 July 2001
Expires in six months

Erik Guttman
Sun Microsystems

Zeroconf Host Profile Applicability Statement
<draft-ietf-zeroconf-host-prof-01.txt>

Status of this Memo

Copyright Notice

Abstract

This document specifies a set of Zero Configuration Protocols which combined support the Zero Configuration domain of applicability. This host profile supports the same upper layer feature set as defined in STD 3 [RFC 1123] by hosts lacking any prior configuration, though in a restricted domain.

1. Introduction

The Internet Standards Process [RFC 2026], Section 3.2, defines how applicability statements are standardized to associate sets of protocols for a particular domain of applicabiliy. This

specification defines the Zero Configuration domain of applicability and a set of protocols which support it.

Requirements for Internet routers [RFC 1812] and hosts [RFC 1122] [RFC 1123] provide guidance to vendors and users of Internet communication software. They represent consensus arising from experience. This document similarly associates a set of protocols together for a particular purpose. In contrast to router and host requirements standards, the Zeroconf Host Profile does not arise out of experience, (though relevant experience is cited.) Instead, this comprises a set of protocols which complement each other when implemented together.

The goal of the Zero Configuration Networking (ZEROCONF) Working Group is to enable networking in the absence of configuration and administration. Zero configuration networking is required for environments where administration is impractical or impossible, such as in the home or small office, embedded systems' plugged together' as in an automobile, or to allow impromptu networks as between the devices of strangers on a train.

As noted in STD 3 [RFC 1122], the current internet suite of protocols fall short of this goal.

> It would be ideal if a host implementation of the Internet protocol suite could be entirely self-configuring. This would allow the whole suite to be implemented in ROM or cast into silicon, it would simplify diskless workstations, and it would be an immense boon to harried LAN administrators as well as system vendors. We have not reached this ideal; in fact, we are not even close.

This document describes a host profile which provides zero configuration operation. Like STD 3, this document describes a set of protocols and makes recommendations with respect to their implementation. Unlike the the mechanisms described in STD 3, we have limited experience with many Zeroconf protocols; some are only now emerging as IETF standards track specifications. Still, we have extensive experience with related protocols, which provided the inspiration for the Zeroconf working group and Zeroconf protocols, specifically the AppleTalk protocol suite [4].

2. Terminology

Terminology specific to discussion of particular zeroconf protocols is introduced in the appropriate section.

In this document, the key words "MAY", "MUST, "MUST NOT", "optional", "recommended", "SHOULD", and "SHOULD NOT", are to be

interpreted as described in [RFC 2119].

3. The Zero Configuration Domain of Applicability

Hosts which lack any specific configuration have zero configuration. The zero configuration domain of applicability concerns hosts with zero configuration for specific functions

3.1. Zero configuration is not all or nothing

A host may be configured with regard to some functions and have zero configuration for others. For example, a host may lack IP interface configuration (described in Section 4.1) but have naming configuration (described in Section 4.2) In this case, zero configuration IP interface autoconfiguration will be used by a host adhering to the Zeroconf Host Profile.

3.2. Configured vs. Zero Configuration Protocol behavior

Zero configuration behavior in each area is well defined. The specific behavior of a host when it becomes configured varies. Each protocol which supports the zero configuration protocol requirements varies in this respect.

IPv4 Link-local IP Interace Configuration [7] and IPv6 address autoconfiguration, [RFC 2461] and ZMAAP [12] are used whether an interface is configued or not.

Link-local Multicast DNS [10] by default is only used when a host has no configured DNS server, unless specifically configure to enable link-local Multicast behavior.

SLPv2 [RFC 2608] always operates in a zero configuration mode, transitions in behavior and reconfiguration occur automatically. (SLPv2 agents may also be configured manually, but that does this does not reduce or change their automatic functions.)

3.3. Scalability and network configuration

The zero configuration domain of applicability includes any IP network which supports multicast (though only broadcast is needed for IPv4 link-local interface configuration). Some protocols described in this applicability statement are defined to only operate using link-local addressing and link-local scope multicast. This is not an inherant limitation of this domain of applicability - for example, SLPv2 [RFC 2608] is defined to operate at admin local scope [RFC 2365] for IPv4 and site local scope for IPv6. [RFC 3111] In any case, the zero configuration domain of applicability is a network under a single common administration, and in some cases only a single network link.

4. Zeroconf Host Profile Requirements and Recommendations

IP Interface Configuration and name resolution services are host requirements (see section 3.3.1.6 [RFC 1122], 6.1.1 [RFC 1123]). A zeroconf host MUST implement these features.

Service discovery constitutes one of the most useful features of the AppleTalk protocol suite [4]. The ease of user configuration from standard service discovery facilities has proved so important that this feature alone has been decisive in continuing support for AppleTalk in many networks. For this reason, a zeroconf host SHOULD implement service discovery functions.

Some multicast applications require the allocation of multicast addresses which do not conflict with other address allocations. Zeroconf hosts MAY implement multicast address allocation functions to support these applications.

The protocols included in the Zeroconf Host Profile provide equivalent functions when run over IPv4 and IPv6. Where there are differences, these are noted.

4.1. IP Interface Configuration

4.1.1. Zeroconf Requirements

Hosts which support IPv4 and the Zeroconf Host Profile MUST implement IPv4 Link-local IP Interace Configuration. [7]

Hosts which support IPv6 and the Zeroconf Host Profile MUST implement IPv6 Stateless Address Autoconfiguration. [RFC 2461] [RFC 2462]

4.1.2. Discussion

IPv4 link-local address autoconfiguration provides an interface with the ability to communicate with hosts on the immediately attached link only. To obtain a routable IPv4 address, some additional mechanism is required.

Implementation issues likely to arise in implementing IPv4 link-local address autoconfiguration include potential mandatory address changes due to conflicts, support for more than one configuration per interface and complications arising from multihomed devices applying link-local autoconfiguration on more than one link. [7]

IPv6 stateless address autoconfiguration provides an interface with a link-local address. This address together with a routing prefix obtained via a router advertisement message enables the configured interface to communicate globally.

There is substantial experience deploying both of these protocols.

[Editor: Issues and observations arising from that experience to go here.]

4.1.3. Comparison against Zeroconf Requirements

The protocols recommended in section 3.1.1 fulfill all Zeroconf requirments (see Section 2.1 of [5]).

[Editor: Is further detailed analysis required?]

4.2. Translation between Host name and IP Address

4.2.1. Zeroconf Requirements

Hosts which support the Zeroconf Host Profile MUST support Multicast DNS. [10] This protocol is defined to work over IPv4 and IPv6.

4.2.2. Discussion

There has been no deployment experience with Multicast DNS to date. There has been extensive experience with the AppleTalk Name Bind Protocol (NBP) [4] and NetBIOS [RFC 1001]. [Editor: Issues and observations arising from experience go here.]

4.2.3. Comparison against Zeroconf Requirements

The protocols recommended in section 3.2.1 fulfill all Zeroconf requirments (see Section 2.2 of [5]).

[Editor: Is further detailed analysis required?]

4.3. IP Multicast Address Allocation

4.3.1 Zeroconf Host Profile Requirements

Hosts which will support applications which require unique multicast address allocation MAY support the Zeroconf Multicast Address Allocation Protocol (ZMAAP) [12].

4.3.2. Discussion

There has been no experience with ZMAAP to date.

4.3.3. Comparison against Zeroconf Requirements

The protocols recommended in section 3.3.1 fulfill all Zeroconf requirments (see Section 2.3 of [5]).

[Editor: Is further detailed analysis required?]

4.4. Service Discovery

SLPv2 [RFC 2608] and DNS SRV RRs [RFC 2782] conveyed over mDNS constitute two distinct options for service discovery for hosts conforming to the Zeroconf Host Profile. The options are discussed below.

This section employs the following terminology:

service
: A particular logical function that may be invoked via some network protocol, such as printing or storing a file on a remote disk.

service characteristics
: Characteristics provide a finer granularity of description to differentiate services beyond just the service type. For example if the service type is printer, the characteristics may be color, pages printed per second, location, etc.

service discovery protocol
: A service discovery protocol enables clients to discover servers (or peers to find other peers) of a particular service. A service discovery protocol is an application layer protocol that relies on network and transport protocol layers.

service protocol
: A service protocol is used between the client and the server after service discovery is complete.

distinct service
: A service is distinct if services of the same type cannot be used interchangeably by clients. Distinct services include those whose physical location, capabilities, state, permissions, performance characteristics or policy differs. A client will discern the difference between service instances. For example, a client seeking to print can only usefully send a job to a printer the user has physical access to. A client attempting to access data in a database can only do so if the correct database server (containing the data the client wishes to access) can be located.

indistinct service
: A service is indistinct if services of the same type can be used interchangeably by clients. For example, any SMTP relay, DNS server or IP gateway will generally provide the same function for a client.

4.4.1. Zeroconf Host Profile Requirements

Hosts implementing the Zeroconf Host Profile SHOULD implement the

Service Location Protocol, Version 2 (SLPv2) [RFC 2608] to enable discovery of distinct services. SLPv2 also enables discovery of

indistinct services. SLPv2 entails some modifications when implemented over IPv6. [RFC 3111]

Hosts implementing the Zeroconf Host Profile SHOULD implement Multicast DNS [10] and support the use of DNS SRV RRs. [RFC 2782]

4.4.2. Discussion

SLPv2 allows the use of service characteristics to distinguish different instances of services. This allows a client to request services on the basis of attributes, and locate the service which fulfills the client's needs.

DNS SRV RRs allow services to be located by name. A client is not able to distinguish between different services of the same named type except by using a service protocol distinct from the service discovery protocol.

In some cases, DNS SRV RR functionality suffices - and since support for mDNS is already included in the Zeroconf Host Profile (as a REQUIRED feature), the lightest-weight implementation may exclude SLPv2 support.

The reason why one uses mDNS to issue requests for DNS SRV RRs is that network services may not be present. If a host is configured to use a DNS server, DNS [RFC 1034] is used instead of mDNS, as described in [10].

SLPv1 and SLPv2 have been deployed in networks for some time. [Editor: Include SLP deployment discussion here.]

DNS SRV RRs have been used by some applications to obtain service locations. These resource records have not been used in conjunction with mDNS so no guidance can be obtained from direct experience. AppleTalk Name Bind Protocol [4], however, provides a very similar function. [Editor: Include NBP observations here.]

Service discovery functionality can be considered as two complementary functions - client discovery and server advertising. A host which functions entirely as a service or as a client would need to implement only those aspects of a service discovery protocol which it needs to conform with the Zeroconf Host Profile. To be specific, a host offered network services but never needed to discover them could implement only SLPv2 Service Agent [12] or mDNS server [10] functions. A host which functioned as a client but never offered services would only implement SLPv2 User Agent or mDNS enhanced resolver functions.

4.4.3. Comparison against Zeroconf Requirements

The protocols recommended in section 3.4.1 fulfill all Zeroconf requirments (see Section 2.4 of [5]).

[Editor: Is further detailed analysis required?]

5. Requirement Levels

As required by [RFC 2026], Section 3.3, each technical specification which is cited must be associated with a requirement level.

FEATURE	SECTION	REQUIRED	RECOMMENDED	ELECTIVE
IP Interface Configuration	3.1	X		
Translation between Host Name and IP Address	3.2	X		
IP Multicast Address Allocation	3.3			X
Service Discovery -	3.4		X	

6. Security Considerations

Security considerations of Zeroconf protocols is discussed in [5]. Hosts conforming to the Zeroconf Host Profile MUST support the security features present in the protocols included in this profile which they implement.

References

[RFC 1812] Baker, F. (Editor), "Requirements for IP Version 4 Routers", RFC 1812, June 1995.

[RFC 1122] Braden, R. (Editor), "Requirements for Internet Hosts -- Communication Layers", STD 3, RFC 1122, October 1989.

[RFC 1123] Braden, R. (Editor), "Requirements for Internet Hosts -- Application and Support", STD 3, RFC 1123, October 1989.

[4] Sidhu, G., et. al., "Inside AppleTalk", Second Edition, Apple Computer, Inc, Addison-Wesley, ISBN 0-201-55021-0, 1990.

[5] Hattig, M., "Zeroconf Requirements", draft-ietf-zeroconf-reqts-08.txt, May 2001. Work in progress.

[RFC 2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.

[7] Cheshire, S., "Dynamic Configuration of IPv4 link-local addresses", draft-ietf-zeroconf-ipv4-linklocal-04.txt Work in progress.

[RFC 2461] Narten, T., Nordmark, E., Simpson, W., "Neighbor Discovery for IP Version 6 (IPv6)", RFC 2461, December 1998.

[RFC 2462] Thomson, S., Narten, T., "IPv6 Stateless Address Autoconfiguration", RFC 2462, December 1998.

[10] Ebisov, L., Aboba, B., Thaler, D., "Multicast DNS", draft-ietf-dnsext-mdns-02.txt. Work in progress.

[RFC 1001] Auerbach, K., Aggarwal, A., (editors), "PROTOCOL STANDARD FOR A NetBIOS SERVICE ON A TCP/UDP TRANSPORT: CONCEPTS AND METHODS", RFC 1001, March 1987.

[12] Catrina, O. (Editor), Thaler, D., Aboba, B., Guttman, E., "Zeroconf Multicast Address Allocation Protocol (ZMAAP)", draft-ietf-zeroconf-zmaap-01.txt. Work in Progress.

[RFC 2608] Guttman, E., Perkins, C., Veizades, J., Day, M., "Service Location Protocol, Version 2", RFC 2608, July 1999.

[RFC 3111] Guttman, E., "Service Location Protocol Modifications for IPv6", RFC 3111, May 2001.

[RFC 2782] Gulbrandsen, A., Vixie, P., Ebisov, L., "A DNS RR for specifying the location of services (DNS SRV)", RFC 2782, February 2000.

[RFC 1034] Mockapetris, P., "DOMAIN NAMES - CONCEPTS AND FACILITIES", RFC 1034, November 1987.

Authors' Addresses

Erik Guttman Sun Microsystems Eichhoelzelstr. 7 74915 Waibstadt Germany

Phone: +49 172 865 5497 Email: erik.guttman@sun.com

Full Copyright Statement

kind, provided that the above copyright notice and this paragraph are included on all such copies and derivative works. However, this

Acknowledgement

Funding for the RFC Editor function is currently provided by the Internet Society.