

ALTO
Internet-Draft
Intended status: Standards Track
Expires: January 13, 2011

H. Song
Huawei
M. Tomsu
Alcatel-lucent Bell Labs
G. Garcia
Telefonica I+D
Y. Wang
Microsoft Corp.
V. Pascual
Consultant
July 12, 2010

ALTO Service Discovery
draft-song-alto-server-discovery-03

Abstract

Application-Layer Traffic Optimization (ALTO) service aims to provide distributed applications with information to perform better-than-random initial peer selection when multiple peers in the network are available to provide a resource or service. In order to discover an Application-Layer Traffic Optimization (ALTO) Server, a set of mechanisms are required. These mechanisms enable applications to find an information source which provides them with information regarding the underlying network. This document discusses various scenarios of ALTO discovery and specifies the use of several available options such as DHCP or 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 <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 January 13, 2011.

Copyright Notice

Copyright (c) 2010 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.

Table of Contents

1.	Introduction	4
1.1.	History	4
1.2.	Overview	4
2.	Terminology	4
3.	ALTO Service Deployment	5
3.1.	ISP-Centric ALTO Service Deployments	6
3.2.	Cross-domain vs. Localized ALTO Server Discovery	7
4.	ALTO Service Discovery Scenarios	7
4.1.	Discovery Metrics	7
4.1.1.	Discovery Clients	7
4.1.2.	Service Location	8
4.1.3.	Layering Perspective	8
4.2.	Discovery Scenarios	9
4.2.1.	Local ALTO service discovery by end terminals	9
4.2.2.	Local ALTO service discovery by application trackers	10
5.	ALTO Service Discovery Mechanisms	11
5.1.	ALTO service discovery using Domain Name System (DNS)	11
5.1.1.	DNS-based ALTO discovery	12
5.1.2.	Determine Service Name of Local ALTO servers	12
5.1.2.1.	Using DHCP option for access domain name	13
5.1.2.2.	Use IANA Database	13
5.1.2.3.	Reverse DNS lookup	14
5.2.	DHCP	14
5.3.	XRD	14
5.4.	Provisioning	15
5.5.	Manual Configuration	15
5.6.	Multicast and broadcast	15
5.7.	Caching	16
6.	Security Considerations	16
7.	IANA Considerations	16
8.	Acknowledgements	16
9.	References	17
9.1.	Normative References	17
9.2.	Informative References	18
	Authors' Addresses	19

1. Introduction

1.1. History

This document represents a merge of features from two previous drafts:

- (1). [draft-wang-alto-discovery-00](#)
- (2). [draft-song-alto-server-discovery-00](#)

The ALTO service architecture and protocol are currently under discussion and development within the IETF ALTO working group.

Although it is identified in the charter that a discovery mechanism is needed, the preference is to adopt one or more existing mechanisms for ALTO discovery rather than designing a new one. This document is consistent with the ALTO framework[I-D.ietf-alto-protocol], and presents different scenarios and available options based on prior and related discovery mechanisms. This document will be updated to track the progress of the ALTO requirements and solution.

1.2. Overview

The ALTO problem statement [[RFC5693](#)] describes that in P2P applications or client/server applications, resources or services are often available through multiple replicas and each of those are sometimes provided by different providers. ALTO service gives guidance to a consumer or directory about which resource provider(s) to select, in order to optimize the client's performance or quality of experience while optimizing resource consumption in the underlying network infrastructure.

In order to query the ALTO server, clients must first know one or more ALTO servers that might provide useful information. The purpose of the ALTO discovery mechanism is to find those servers in different network and application scenarios.

[Section 3](#) and [Section 4](#) discuss various scenarios of ALTO service deployment and discovery. [Section 5](#) provides a description of available discovery mechanisms and its application to the ALTO service discovery use case addressing potential issues and consideration for each.

2. Terminology

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](#)].

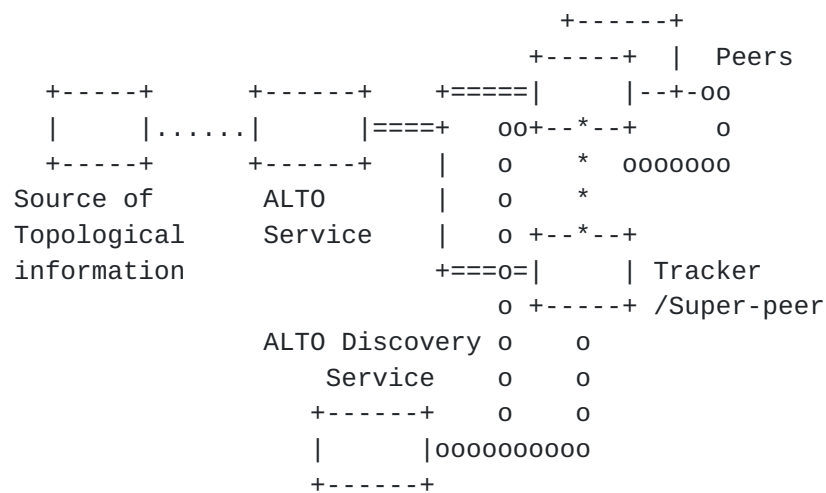
The document uses terms defined in [[RFC5693](#)].

In addition to the generic ALTO descriptions, the following terms are used to describe the discovery mechanisms in this document:

- o ALTO Discovery Client: The logical entity discovering the ALTO Service. Depending on the scenario, this could be a Peer or a Super-peer/Tracker.
- o ALTO Discovery Server: The logical entity providing information to locate the ALTO Service. Depending on the discovery mechanism, this could be another Peer or a dedicated entity in the network.
- o ALTO Discovery Domain: The scope of the network handled by a particular ALTO Discovery Server.

[3.](#) ALTO Service Deployment

This section explores the various dimensions of the ALTO service deployment and access scenarios, and briefly discusses their implications to the discovery mechanisms. Figure 1 below shows a generic ALTO framework diagram with discovery. .



Legend:

=== ALTO query protocol
 ooo ALTO service discovery protocol
 *** Application protocol (out of scope)
 ... Provisioning or initialization (out of scope)

Figure 1 ALTO Discovery Diagram

3.1. ISP-Centric ALTO Service Deployments

(Haibin: we delete the application-centric ALTO service deployment scenario as to keep consistent with the ALTO framework in the working group)

An ALTO Server is the logical entity that provides query interfaces for ALTO Clients. ALTO servers are deployed in an ISP-centric deployment.

A network operator which wants to optimize its traffic, e.g. to reduce its transit traffic volume across the network boundaries; a third party on behalf of one or even several ISPs.

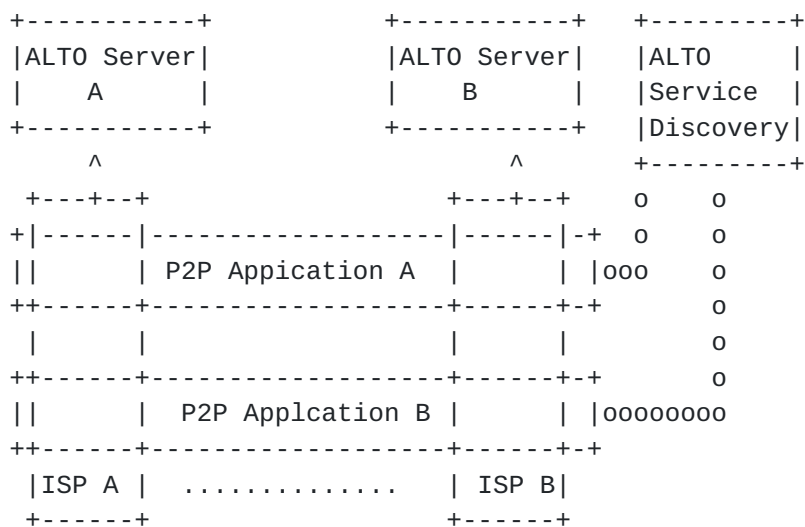


Figure 2: ISP-centric ALTO service deployment example

3.2. Cross-domain vs. Localized ALTO Server Discovery

For cross domain scenarios, the ALTO client is embeded in the application tracker or Resource Directory.

There may be several ALTO servers distributed in different operator's networks. Each operator may provide the ALTO service using their own ALTO servers. Each network operator may have its own traffic optimization policy based on his network topology, however it may not know other network operator's policies, nor be clear of other network operator's topologies (e.g. topology hiding). Each of the ALTO servers may have a FQDN.

The ALTO client (e.g. the Tracker) must be able to discover and choose the ALTO server that has the information that is specific to those clients located within that network.

In localized discovery deployments one or several ALTO servers provide the service only to clients in their own network or autonomous system.

4. ALTO Service Discovery Scenarios

4.1. Discovery Metrics

4.1.1. Discovery Clients

The ALTO Client can be the Peer in the end-user host or an external entity like a Super-peer or Resource Directory (aka Tracker) on

behalf of the Peer [[RFC5693](#)]. If a Super-Peer or Tracker acts as an ALTO Client it needs to know and select the suitable ALTO Service for the Peer being served. Possible mechanisms for third party ALTO discovery have been proposed in[I-D.kiesel-alto-3pdisc]

In a hybrid model the address info of the ALTO Server could be communicated from the Peer to the Super-Peer using the application protocol. It could also be discovered by the Super-Peer from other Peer information received implicitly (like the Peer public IP address) or received explicitly.

There could be scenarios where only the Peer (and not the Super-Peer/Tracker) is able to access the ALTO Service, for example if the ALTO Server is located in a private network. Also the ALTO server might not allow requests from the IP addresses that are out of its administrative domain.

4.1.2. Service Location

The ALTO service is provided by a centralized entity (the ALTO Server) for a given scope. A centralized ALTO Server is implicitly or explicitly assigned to a specific network scope, an out-of-band discovery mechanism is often required.

The ALTO Server for a Peer could be in the same Local Area Network (LAN), within the same ISP Network but not on the same LAN, or in the Global Internet outside the ISP Network. Different network scopes place different constraints on the discovery mechanisms. Multicast discovery generally works within a single LAN only, whereas DNS-based or DHCP-aabased discovery can span multiple subnets within a single ISP or a single network administrative domain. Internet scope discovery usually requires cross-domain indexing or directory services. Note that peers participating in a single P2P application may reside on the same or different ISP networks. Scenarios like this may require hybrid discovery solutions that can adapt to multiple network scopes at the same time. The discovery mechanisms listed in this document should take into account possible limitations of the ALTO service deployment in those network scopes.

4.1.3. Layering Perspective

The discovery process takes place before the first access to the ALTO server. This discovery process could be done at host network initialization time, at application initialization time or just before the first ALTO query is sent.

4.2. Discovery Scenarios

The ALTO service discovery scenarios are classified into two types: one is the ALTO server discovery by end terminals, and the other is the ALTO server discovery by application trackers.

4.2.1. Local ALTO service discovery by end terminals

In p2p applications without a tracker like DHTs and other conventional client/server applications, an end device needs to discover the local ALTO server by itself.

P2P application which has tracker(s) may also embed the ALTO client within the peers. And the peers can do the remote peer selection after retrieving peer list from the application tracker. Or the peer can send its ALTO server address information to the application tracker, and the application tracker will contact the specific ALTO server and do the peer selection for peers.

After the discovery of an ALTO server, the p2p client can get guidance from the ALTO server directly or through its application tracker.

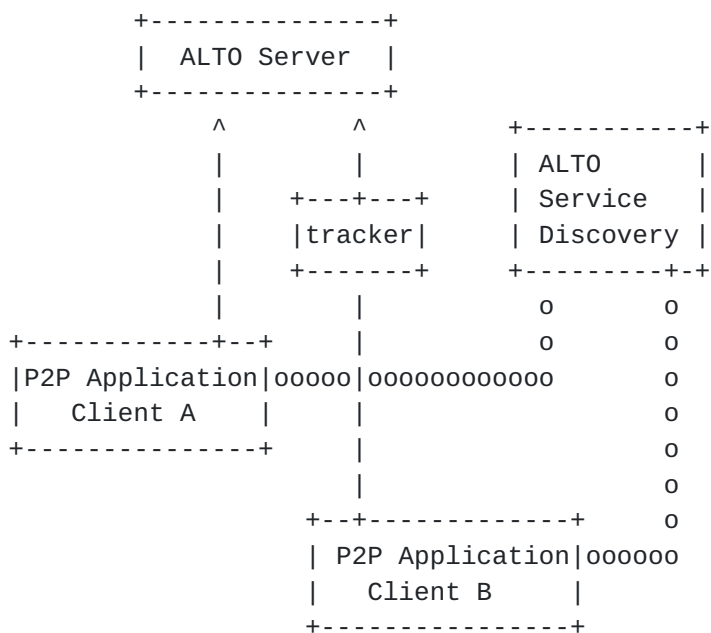


Figure 3: Local ALTO service discovery by end terminals (Example)

4.2.2. Local ALTO service discovery by application trackers

Some p2p applications have trackers, and these applications might not need to have their clients looking for the ALTO server guidance. Trackers query the ALTO servers for guidance, and then return the final ranked result to the application clients. However, application clients are distributed among different network operators and autonomous systems. Trackers must find different ALTO servers for the clients located in different network operators or autonomous systems.

Figure 4 shows an example for a tracker's ALTO server discovery. For client 1, the tracker has not cached yet the mapping between client 1's network operator and its ALTO server address, so it queries the DNS server for the ALTO server address in that operator's domain. And then the tracker interacts with the ALTO server on behalf of client 1 (to get the network map and cost map), finally, the ranked list is sent back to client 1. For client 2, the tracker has cached the mapping between client 2's network operator and its ALTO server address, so it does not need to query the DNS for the address of ALTO server 2. If the Application tracker already has the network map and cost map from ALTO Server2, then it does not query the ALTO Server for network map and cost map frequently.

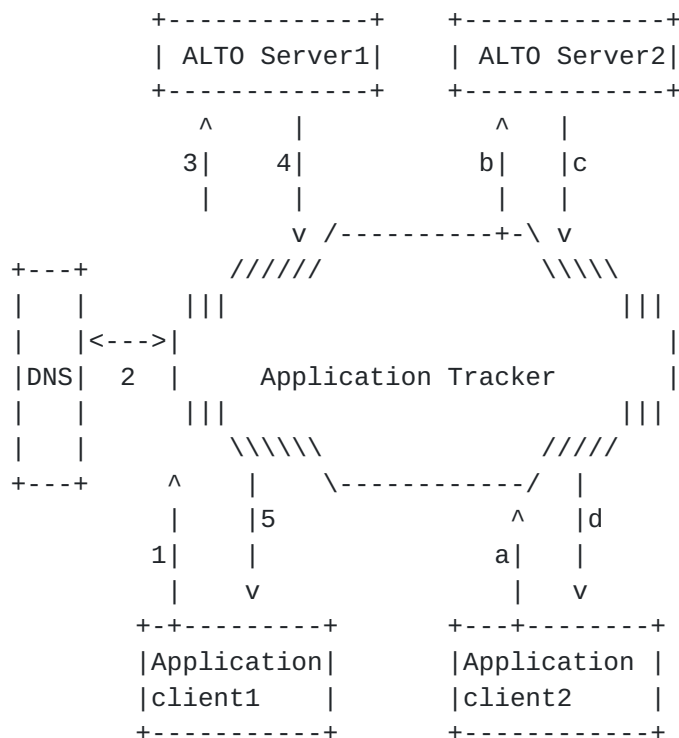


Figure 4: Local ALTO service discovery by application trackers (Example)

5. ALTO Service Discovery Mechanisms

One ALTO client should use one or several of the introduced discovery mechanisms according to its application scenario until it finally finds an appropriate ALTO server.

The following issues should be considered when designing the ALTO service discovery mechanism.

Load Balance: When more than one ALTO server provide identical service for the same area, we must find a mechanism to balance the processing load between the ALTO servers;

Well known port: If ALTO server provides service through a well known port, then the discovery mechanism only needs to discover the IP address of an ALTO server that can provide service for a client, otherwise, the discovery mechanism must discover both IP address and port number through which the ALTO server provides the service.

Note:It will depend on the ALTO protocol whether a well know port is used for the ALTO server. If there is no well known port for the ALTO server, we need to discover the port information with the discovery process.

IP address change: The IP address of the ALTO server may change in some circumstances. The ALTO service discovery mechanism must be well adaptable to this case when necessary.

Mobile Scenarios: When the end terminals are mobile equipments, the data traffic may route via a roaming client's home agent's router to the client, or route to the client directly. Which ALTO server to choose should depend on the routing optimization mode adopted for mobility. If the data traffic routes via the client's home agent, it should choose the ALTO server that serves its home area network, otherwise, it should choose the ALTO server that serves its current network.

5.1. ALTO service discovery using Domain Name System (DNS)

DNS is widely used on the Internet to discover the server address for applications. ALTO service is a conventional client/server mode service, which can use DNS lookup for its service discovery.

NAPTR [[RFC2915](#)] and SRV [[RFC2782](#)] DNS resource records are appropriate to provide service discovery mechanisms. The concrete application of these resource records depends on the final ALTO requests/response protocol. The use of NAPTR or SRV records is a

trade-off between flexibility and simplicity. S-NAPTR [[RFC3958](#)] and U-NAPTR [[RFC4848](#)] mechanisms provide a Dynamic Delegation Discovery System (DDDS) Application to map domain name, service name and protocol name to a target host and port or to a target URI. SRV records provide a mechanism to map domain name, service name and transport protocol name to a target host and port. The use of a NAPTR or SRV solution is open to discussion and depends on the requirements of the ALTO protocol. Next section will assume the use of SRV records.

5.1.1. DNS-based ALT0 discovery

Figure 5. shows a general DNS ALTO server discovery mechanism. A server must register its SRV resource record with a well known service name (e.g. `_ALTO._TCP.example.com`) in the DNS system. The service name in this document refers to the name used for DNS SRV query, which includes the service label, protocol name (TCP or UDP) and domain name. Any ALTO client that wants to get the IP address and port of the ALTO server sends a DNS SRV query to the DNS server in that domain .

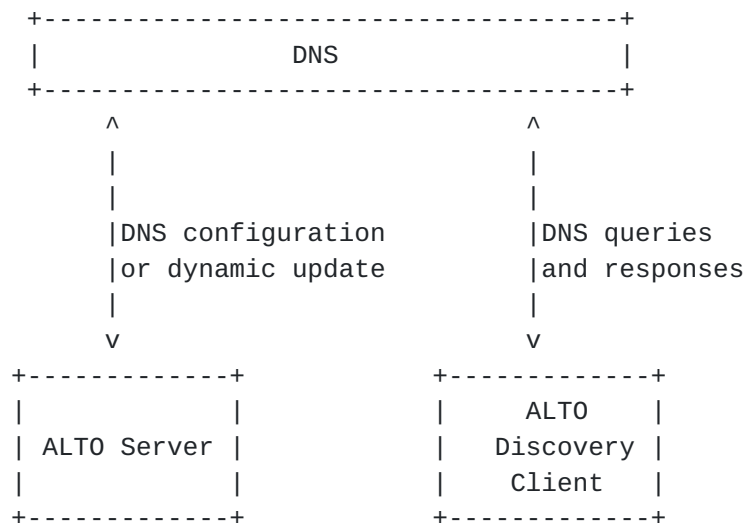


Figure 5: DNS query for well known ALTO servers

5.1.2. Determine Service Name of Local ALT0 servers

An ALTO discovery client must know its ALTO service name for it before sending a query to the DNS system. Some ALTO servers may provide service to the overall network, they may have well-known service name. But in most cases, one ALTO server will only provide service to its own local access network or autonomous system. There will be multiple ALTO servers in the overall network. An ALTO discovery client needs to find the service name of its local ALTO

server.

5.1.2.1. Using DHCP option for access domain name

There are DHCP options (OPTION_V4_ACCESS_DOMAIN and OPTION_V6_ACCESS_DOMAIN) proposed in [[I-D.ietf-geopriv-lis-discovery](#)] to discover the local access domain names. The retrieved access domain name can be used to form a SRV name by prefixing the ALTO service label to the access domain name. If it failed with the SRV lookup with this service name, then it will remove one tag from the left hand of the access domain name and prefix the ALTO service label to form a new SRV name. It will iterate the process until it succeeds in getting an ALTO server information or failed.

It should be noticed that there are many residential gateways (RG) which can act as DHCP servers themselves. RG becomes a hindrance between the end terminals and the ALTO service provider's DHCP server if we use DHCP. It should not depend on the update of all these RGs to support this new DHCP Option for ALTO server discovery. A DHCP Container Option [[I-D.ietf-dhc-container-opt](#)] for server configuration should be used here. With the Container Option, the DHCP server for the consumer domain (e.g. RGs) can just pass the server configuration to the end terminals without explicit knowledge of the DHCP options contained in the Container. The DHCP Option for the access domain name could be contained in the Container Option.

5.1.2.2. Use IANA Database

The service name of a client's local ALTO server could be formed by adding the service and protocol label before its domain information. IANA and its subsidiary organizations (e.g. APNIC) database can be used to lookup the physical domain of a client through its public IP address, i.e. which network operator and/or autonomous system the client belongs to. The WHOIS service [[www.whois](#)] on the Internet is also available for this purpose. This mechanism requires ISPs assign the domain names to their ALTO servers according to the AS and ISP information (e.g. they have a rule to format the domain name, AS.ISP.COM), then you can rebuild the domain name with the information retrieved from WHOIS. Otherwise, you can't.

However, the mapping information may be changed due to the business deals and network adjustment. For example, an ISP could sell some part of its network (include all equipments, IP addresses, AS number, and so on) to another ISP, and the ISP does not have the responsibility to notify the IANA, and then the information in the IANA database is wrong.

5.1.2.3. Reverse DNS lookup

BEP 22 [[BEP-22](#)] framework uses reverse DNS lookup to determine the domain name of a client through its public address. And then use service label and the domain name to lookup the local server in DNS. The following limitations should be considered when use this mechanism.

(1) This method assumes that the access network provider also provides the reverse DNS record and they control the domain that is indicated in the "PTR" record. (In most cases it is true, but not always)

(2) Furthermore, this method might not apply where a host is given a domain name that is different from the domain name of the access network.

(3) In case of NAT and a public ALTO server, it requires the ALTO client to know its public IP address.

The advantage is that it doesn't require any update/configuration/change in the DHCP servers of any residential gateway.

5.2. DHCP

There are other ways using DHCP to locate an ALTO server. One suggestion is to use DHCP to obtain the ALTO server IP address and port information directly. New DHCP options are needed for this purpose. The residential gateways consideration for DHCP option must be considered as described in . ([Section 5.1.2.1](#))

With this mechanism, the DHCP server needs to support load balance if there are more than one ALTO servers for this access domain. The maintenance is costly when the address of ALTO server changes.

5.3. XRD

XRD is described in [[XEP-1.0](#)]. In order to begin the XRD discovery you need the URL (or XRI) of the resource you want to discover links/services related to. In other XRD use cases like OpenId or OpenSocial, it is clear that you know that URL (the OpenId url of the user, or the url of the OS container). But in case of ALTO Server Discovery, the obtainment of this initial URL also needs to use some discovery framework.

5.4. Provisioning

A network operator can simply provide a configuration file that contains the ALTO server address for its clients, provided that there are only one or a few ALTO servers which provide identical service for its network. An application can also provide such a configuration file containing the ALTO server address if an existing ALTO server provides identical service to the overall network.

5.5. Manual Configuration

Manual configuration of the ALTO service location(s) could work in a single ISP network scope, but is not scalable when multiple ISPs or cross-domain ALTO services are required. P2P applications often connect peers from ISPs that they may not have contacted before, and manual configuration will not work without any prior knowledge of the ALTO servers.

5.6. Multicast and broadcast

Multicast or broadcast MAY be used in some scenarios for ALTO discovery.

IP-multicast-based discovery generally works in two ways:

1. Clients send out multicast discovery requests and listen for responses (usually unicast) from available servers or service providers.
2. Servers or service providers send out multicast announcements when they become available or periodically, and clients wait for the next available multicast announcement to identify the servers or service providers.

The on-demand requests and periodic announcements are not mutually exclusive. An implementation can choose to utilize both simultaneously. The configuration effort of multicast discovery is fairly straightforward, only the multicast address and port are needed. Service types and additional information are often encoded in the requests or announcements messages, enabling the same multicast channel to support discovery of different resources or services. There are two main constraints of multicast-based discovery - scopes and flooding messages. Routers disable multicast forwarding by default, making it practically a single-subnet solution. Some forms of discovery proxies are needed to extend the scope of multicast discovery to multiple subnets. The second issue is the flooding of multicast messages to all hosts on the same subnet. The total bandwidth consumed by multicast depends on the

arrival rate the client application requests, and/or the frequency of the service announcements. Older generations of 802.11-based wireless access points often slow down the transmission of multicast messages or generally have a higher packet loss rate for those, causing some multicast discovery implementation to automatically re-send multicast requests or announcements by default. This mitigation further increases the amount of flooding messages on the LAN. Examples of multicast-based discovery include [I-D.cheshire-dnsext-multicastdns], [[I-D.cai-ssdp-v1](#)], [[WSD](#)], SLP [[RFC2165](#)], and LLMNR [[RFC4795](#)].

[5.7.](#) Caching

Once a client has located an ALTO server for the first time, it can cache it for use as future ALTO server. There are implications in case of mobility of devices.

[6.](#) Security Considerations

As this document mainly proposes to use DNS and DHCP for ALTO service discovery, it will depend on the DHCP security and DNS security for this service discovery.

[7.](#) IANA Considerations

The service label for the ALTO service will depend on the final protocol name for application-layer traffic optimization(TBD).

[8.](#) Acknowledgements

The authors would like to give special thanks to Roni Even for his continuous contribution to this document.

We would also like to thank the following experts for their contribution.

Sebastian Kiesel

Yunfei Zhang

Y. Richard Yang

Xingfeng Jiang

Jay Gu

Ning Zong

David Bryan

Enrico Marocco

9. References

9.1. Normative References

- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", [BCP 14](#), [RFC 2119](#), March 1997.
- [RFC2782] Gulbrandsen, A., Vixie, P., and L. Esibov, "A DNS RR for specifying the location of services (DNS SRV)", [RFC 2782](#), February 2000.
- [RFC3958] Daigle, L. and A. Newton, "Domain-Based Application Service Location Using SRV RRs and the Dynamic Delegation Discovery Service (DDDS)", [RFC 3958](#), January 2005.
- [RFC4848] Daigle, L., "Domain-Based Application Service Location Using URIs and the Dynamic Delegation Discovery Service (DDDS)", [RFC 4848](#), April 2007.
- [RFC5693] Seedorf, J. and E. Burger, "Application-Layer Traffic Optimization (ALTO) Problem Statement", [RFC 5693](#), October 2009.
- [RFC2915] Mealling, M. and R. Daniel, "The Naming Authority Pointer (NAPTR) DNS Resource Record", [RFC 2915](#), September 2000.
- [I-D.ietf-alto-protocol]
Alimi, R., Penno, R., and Y. Yang, "ALTO Protocol", [draft-ietf-alto-protocol-04](#) (work in progress), May 2010.
- [I-D.ietf-geopriv-lis-discovery]
Thomson, M. and J. Winterbottom, "Discovering the Local Location Information Server (LIS)", [draft-ietf-geopriv-lis-discovery-15](#) (work in progress), March 2010.
- [I-D.ietf-dhc-container-opt]
Droms, R., "Container Option for Server Configuration", [draft-ietf-dhc-container-opt-05](#) (work in progress),

March 2009.

9.2. Informative References

- [RFC2629] Rose, M., "Writing I-Ds and RFCs using XML", [RFC 2629](#), June 1999.
- [RFC3552] Rescorla, E. and B. Korver, "Guidelines for Writing RFC Text on Security Considerations", [BCP 72](#), [RFC 3552](#), July 2003.
- [RFC2165] Veizades, J., Guttman, E., Perkins, C., and S. Kaplan, "Service Location Protocol", [RFC 2165](#), June 1997.
- [RFC4795] Aboba, B., Thaler, D., and L. Esibov, "Link-local Multicast Name Resolution (LLMNR)", [RFC 4795](#), January 2007.
- [I-D.kiesel-alto-3pdisc]
Kiesel, S., Tomsu, M., Schwan, N., Scharf, M., and M. Stiernerling, "Third-party ALTO server discovery", [draft-kiesel-alto-3pdisc-03](#) (work in progress), July 2010.
- [I-D.cheshire-dnsext-multicastdns]
Cheshire, S. and M. Krochmal, "Multicast DNS", [draft-cheshire-dnsext-multicastdns-11](#) (work in progress), March 2010.
- [I-D.wang-alto-p4p-specification]
Wang, Y., Alimi, R., Pasko, D., Popkin, L., and Y. Yang, "P4P Protocol Specification", [draft-wang-alto-p4p-specification-00](#) (work in progress), March 2009.
- [I-D.narten-iana-considerations-rfc2434bis]
Narten, T. and H. Alvestrand, "Guidelines for Writing an IANA Considerations Section in RFCs", [draft-narten-iana-considerations-rfc2434bis-09](#) (work in progress), March 2008.
- [I-D.cai-ssdp-v1]
Goland, Y., Cai, T., Leach, P., Gu, Y., and S. Albright, "Simple Service Discovery Protocol/1.0 Operating without an Arbiter", October 1999, <[draft-cai-ssdp-v1-03](#)>.
- [WWW.WHOIS]
"http://www.whois.net".

- [BEP-22] Harrison, D., Shalunov, S., and G. Hazel, "BitTorrent Local Tracker Discovery Protocol", October 2008, <http://bittorrent.org/beps/bep_0022.html>.
- [XEP-1.0] Hammer-Lahav, E., "Extensible Resource Descriptor (XRD) Version 1.0", May 2009, <<http://www.oasis-open.org/committees/download.php/32686/xrd-1.0-wd01.html>>.
- [WSD] Beatty, J., "Web Services Dynamic Discovery (WS-Discovery)", April 2005, <<http://specs.xmlsoap.org/ws/2005/04/discovery/ws-discovery.pdf>>.

Authors' Addresses

Haibin Song
Huawei

Email: melodysong@huawei.com

Marco Tomsu
Alcatel-lucent Bell Labs
Lorenzstrasse 10
70435 Stuttgart
Germany

Email: marco.tomsu@alcatel-lucent.com
URI: www.alcatel-lucent.com/bell-labs

Gustavo Garcia
Telefonica I+D
Emilio Vargas
Madrid, Madrid
Spain

Phone: +34 913129826
Email: ggb@tid.es

Yu-Shun Wang
Microsoft Corp.
One Microsoft Way
Redmond, WA 98052
USA

Email: yu-shun.wang@microsoft.com

Victor Pascual
Consultant

Email: victor.pascual.avila@gmail.com