An Architecture of ALTO for P2P Applications

draft-yang-alto-architecture-00.txt

Status of This Memo

This Internet-Draft is submitted to IETF in full conformance with the provisions of BCP 78 and BCP 79.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF), its areas, and its working groups. Note that other groups may also distribute working documents as Internet-Drafts.

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."

The list of current Internet-Drafts can be accessed at http://www.ietf.org/ietf/1id-abstracts.txt.

The list of Internet-Draft Shadow Directories can be accessed at http://www.ietf.org/shadow.html.

This Internet-Draft will expire on September 5, 2009.

Copyright Notice

Copyright (c) 2009 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 in effect on the date of publication of this document (http://trustee.ietf.org/license-info). Please review these documents carefully, as they describe your rights...
and restrictions with respect to this document.

Abstract

ALTO enables Internet Service Providers (ISPs) and network application software distributors to work jointly and cooperatively to reduce network resource consumption and to improve application performance. In this document, we specify an architecture for integrating ALTO into peer-to-peer (P2P) applications.

Table of Contents

1. Requirements Notation ........................................... 3
2. Introduction ..................................................... 3
  2.1. Overview ................................................... 3
  2.2. Terminology ................................................ 3
  2.3. Requirements ................................................. 4
  2.4. Comparisons with Alternatives .............................. 4
3. Architectural Model .............................................. 4
  3.1. Basic Concepts .............................................. 4
    3.1.1. My-Internet View ..................................... 4
    3.1.2. ALTO Cost Type ......................................... 4
    3.1.3. Hosting ALTO Server ................................... 5
    3.1.4. Location Grouping ...................................... 5
  3.2. Basic Functional Components ............................... 7
    3.2.1. ALTO Query/Response Protocol .......................... 8
    3.2.2. ALTO Service Discovery ................................. 8
  3.3. Extensions .................................................. 8
    3.3.1. Redistribution and Caching of ALTO Info ............... 9
    3.3.2. ALTO Effectiveness Monitoring ......................... 9
4. Example Functional Components Instantiations .................. 9
  4.1. Tracker-based Peer Selection using ALTO .................. 10
  4.2. Client Peer Selection using ALTO .......................... 10
5. P2P Application ALTO Integration Guidelines .................. 10
  5.1. Peer Selection Guidelines ................................. 10
  5.2. Interoperability with Non-ALTO Clients ................... 11
6. ALTO Service Discovery Guidelines ............................. 11
  6.1. Delegation ................................................ 11
  6.2. NAT ........................................................ 11
  6.3. Load Balancing and Robustness ............................. 11
7. Security Considerations ........................................ 11
8. References ...................................................... 11
  8.1. Normative References ...................................... 11
  8.2. Informative References .................................... 11
Appendix A. Contributors ........................................... 12
Appendix B. Acknowledgments ....................................... 12
1. Requirements Notation

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in [RFC2119].

2. Introduction

2.1. Overview

ALTO is a service to allow Internet service providers (ISPs) and network applications to work more cooperatively. Since in general peer-to-peer applications generate large amounts of Internet traffic and many of these produce traffic patterns that are network-oblivious, it is important to provide such applications with more information so that they can effectively utilize their communication flexibility to perform better-than-random peer selection, which can reduce network resource consumption and improve application performance.

Recently, multiple schemes have been proposed, including ALTO Info Export [Info-Export], P4P [P4P-framework], and PROXIDOR [PROXIDOR]. These schemes represent different points at the solution spectrum.

This document presents a unifying architecture of how ALTO can be integrated into large P2P applications such as those based on BitTorrent (e.g., BitTorrent and PPLive). This class of applications share many common features to allow a unified study of architecture.

The objective of this document is to complement the Problem Statement [ALTO-Problem] and the Requirements [ALTO-Reqs] documents to identify the essential functional components in the architecture. Note that the architecture presented in this document serves an informative purpose only. It provides a conceptual framework for more structured design and discussions.

2.2. Terminology

We use the following terms defined in [ALTO-Problem]:

2.3. Requirements

The architecture to be presented is guided by the following key design requirements:

- The architecture is extensible so that ALTO information can be used by both P2P Peers and P2P Application trackers (e.g., a BitTorrent tracker uses ALTO for initial peer selection). Tracker-based peer selection can be beneficial given the global knowledge of a tracker about a P2P Application. However, since a P2P tracker may manage a large number of Peers, ALTO should avoid substantial workload and redundancy on the tracker.

- Each ISP or its delegate can configure the ALTO Server that provides the information about the ISP’s network. Each ISP can determine the level of detail that it wishes to expose. It applies any desired aggregation and transformation mechanisms. Information from ALTO Servers can be "anonymized" or access-controlled to protect the ISP’s topology and business policies. Although multiple ISPs can interact and negotiate with respect to the information provided by their ALTO Servers, it is outside the scope of this architecture.

2.4. Comparisons with Alternatives

To be added.

3. Architectural Model

3.1. Basic Concepts

3.1.1. My-Internet View

The basic architecture we present is based on a simple model that each ALTO Server defines a "my-Internet" view, which consists of a set of network locations identifiable by Host Location Attributes, and generic costs among network locations. An ALTO Service provides its service based on its my-Internet view.

3.1.2. ALTO Cost Type

An ALTO server may allow multiple types of generic costs to be defined between each pair of network locations. Example types include hop-count, air-mile, and routing-cost. We refer to each type as an ALTO Cost Type.
3.1.3. Hosting ALTO Server

Given a Resource Consumer \( p \) and a given type of ALTO Cost, an ALTO Client identifies a Hosting ALTO Server using ALTO Service Discovery. Network efficiency for \( p \)'s behavior is considered from the my-Internet view of the Hosting ALTO Server of \( p \).

3.1.4. Location Grouping

In this architecture, we introduce Location Grouping in ALTO Queries and Responses for scalability and privacy.

Specifically, an ALTO Query specifies a set of locations on the my-Internet view representing Resource Consumers, and a set of locations on the my-Internet view representing potential Resource Providers. The ALTO Response reveals network information related with these network locations.

We distinguish two types of location grouping: Source (Resource Consumer) Grouping and Destination (Resource Provider) Grouping.

3.1.4.1. Source Grouping

Without Source Grouping, each ALTO Query is from the vantage point of a specific Resource Consumer on the my-Internet view. Consequently, the number of ALTO Queries to be handled (both by ALTO Servers and ALTO Clients) will grow with the number of Peers, which can be large. This can be particularly challenging for tracker-based systems, where a tracker manages a large number of peers and it is the tracker who uses ALTO to make initial peer selections.

One can identify settings where fine-grained queries for individual IP addresses are necessary, and thus should be supported.

On the other hand, for many ISPs and in particular in the setting of P2P, network information for each individual IP address may not be necessary or too fine-grained. Instead, a set of network locations may have similar costs to other network locations. Thus, an ISP, through its ALTO Server, may specify equivalent classes of network locations containing Resource Consumers. Each such class is called a Source Grouping.

By defining Source Grouping, we improve both scalability and privacy:

- Scalability: The number of ALTO Queries grow with the number of Source Groupings. The state at a tracker who uses ALTO to make initial peer selection grows with the number of Source Groupings instead of the number of Peers. It also makes it possible for
3.1.4.2. Destination Grouping

The network information to a set of network locations as Resource Providers may be similar. For example, although there are tens of thousands of Autonomous Systems (ASes), the routing costs of an ISP to these ASes may be quite coarse-grained (e.g., according to the business relationship of the first hop BGP peer). Thus, an ISP, through its ALTO Server, may specify equivalent classes of destination network locations. Each such class is a Destination Grouping. Destination Grouping can reduce redundancy, improve scalability, and can help to protect Peer privacy.

3.1.4.3. Grouping Levels

The level of grouping can span a whole spectrum: individual IP address, a subnet, a set of subnets, a point of presence (PoP), an Internet exchange point, an autonomous system (AS), a set of autonomous systems (e.g., those with the same next-hop AS, those reachable through provider ASes, or those with the same BGP AS_PATH length), or some other set defined by a common set of properties.

Although some of the preceding groupings can be naturally expressed using the current Internet addressing scheme (IP prefix, ASN), others (e.g., PoP, those reachable through provider ASes) are not.

In this architecture, we adopt a generic approach. Each grouping is identified by a Network Grouping Identifier to allow flexibility, reuse and reducing redundancy. A Grouping Identifier can be considered as a Host Location Attribute.

3.1.4.4. Grouping Examples

Table 1 illustrates how the three approaches aforementioned in the Introduction use Location Grouping. The column "Source Grouping" denotes the levels of Source Groupings supported (i.e., how an ALTO Client specifies a Resource Consumer), the column "Destination Query" denotes how an ALTO Client specifies the Resource Providers (candidate Peers), and the column "Destination Grouping" denotes the levels of Destination Groupings supported.
Table 1: Grouping Levels Used in ALTO Info Export, P4P, and PROXIDOR.

### 3.2. Basic Functional Components

Given the preceding key concepts, we now introduce the basic functional components. Figure 1 shows the main functional components in the basic architecture:

```
<table>
<thead>
<tr>
<th>ALTO Server</th>
<th>ALTO Query/Response</th>
<th>ALTO Client</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>ALTO SD Query/Response /</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>: ALTO Service :</td>
<td></td>
</tr>
<tr>
<td></td>
<td>: Discovery :</td>
<td></td>
</tr>
<tr>
<td></td>
<td>'...............'</td>
<td></td>
</tr>
</tbody>
</table>
```

Figure 1: Basic ALTO Architecture.
o ALTO Query/Response: We refer to an ALTO Query and its corresponding ALTO Response as an ALTO Transaction. The information conveyed in ALTO Transactions is referred to as ALTO Information.

o ALTO Service Discovery: Although it is possible to use manual configuration to avoid this component, for large-scale deployment, service discovery is necessary.

3.2.1. ALTO Query/Response Protocol

Specifically, different types of ALTO Queries can be constructed depending on the information needed by the client. In its most generic form, an ALTO query specifies (1) a set of Host Location Attributes (Network Grouping Identifiers) representing Resource Consumers, (2) a set of Host Location Attributes (Network Grouping Identifiers) representing potential Resource Providers, and (3) a desirable ALTO Cost Type.

An ALTO Response includes the values of the costs from the Resource Consumers to the Resource Providers.

The returned information may also be in a transformed format. For example, instead of the exact values, they may be rankings derived from the cost values. The response will be used for making peer selection.

3.2.2. ALTO Service Discovery

3.3. Extensions

The Basic Architecture provides interoperability, but lacks components for (1) improving scalability, (2) using multiple information sources, and (3) handling issues that can arise when networks and P2P applications operate autonomously. Although the ALTO Working Group may not pursue these components initially, it is important to identify the related issues when defining the core components defined in the Basic Architecture.

Figure 2 shows additional functional components. In particular, ultimately, P2P applications make peer selection decisions based on multiple sources of information, including not only ALTO information, but also application state (e.g., distribution of super nodes, seeds, and down loaders), and other sources of information such as probed network information or shared underlay information.
### 3.3.1. Redistribution and Caching of ALTO Info

Redistribution and Caching of ALTO Information: Large-scale deployment of P2P applications may generate a large number of ALTO Transactions. Thus, mechanisms for the caching (e.g., leveraging HTTP caches \[P4P-spec\]) and redistribution (e.g., using P2P) of ALTO Information are necessary to avoid ALTO becoming a system bottleneck and to reduce ISP deployment costs.

In particular, if redistribution is allowed, then the ALTO Request/Response protocol may include mechanism to allow ease of redistribution and validation of redistributed ALTO Information.

### 3.3.2. ALTO Effectiveness Monitoring

ALTO Effectiveness Monitoring: Applications evaluate effectiveness of ALTO information, and the effectiveness is used in Peer Selection. It is possible that networks may also want to monitor the effects of its provided ALTO information.

### 4. Example Functional Components Instantiations

The preceding section gives basic functional components. In this section, we give instantiations of the architecture for multiple application scenarios.
4.1. Tracker-based Peer Selection using ALTO

In this setting, the application tracker (Resource Directory) keeps track of Peers in a torrent. There is an ALTO Client embedded in the application tracker. The ALTO Client queries the ALTO Servers of the Peers in the torrent to obtain their my-Internet views. Then the application tracker conducts peer selection.

There can be an alternative instantiation, in which a third party provides an ALTO Client (e.g., application optimization engine [P4P-SIGCOMM08], [P4P-framework]). The application tracker sends application state to the third party, who issues ALTO Queries and computes peer selection guidance for the application tracker.

4.2. Client Peer Selection using ALTO

In this setting, a Peer uses ALTO when conducting peer selection (i.e., selecting among the known peers those that it preferentially connects to). There are several settings that this may happen:

- There does not exist a tracker. The Peers discover each other using mechanisms such as DHT.
- There is a tracker but the tracker does not support ALTO (yet).
- Even though there is a tracker and the tracker applies ALTO when conducting initial peer selection, the Peer applies ALTO when selecting peers as it learns additional peers such as those from Peer Exchange.

There are at least two types of instantiations of ALTO Clients in this setting: In the first case, an ALTO Client is embedded in the BitTorrent Client; in the second case, the ALTO Client is implemented as part of the operating system.

In either case, the ALTO Client discovers the ALTO Server of the Internet Service Provider of the Peer. Then the ALTO Client obtains the my-Internet view of the ISP to help with peer selection.

5. P2P Application ALTO Integration Guidelines

5.1. Peer Selection Guidelines

Multiple examples of using ALTO Information have been proposed (e.g., the usage examples in [Info-Export], the Application Optimization Engine in [P4P-framework], and the example usages in [PROXIDOR]). Although it is unlikely that we can enforce Applications behaviors, development of guidelines for Application peer selection can be
beneficial, in an organization such as the P4P Working Group.

5.2. Interoperability with Non-ALTO Clients

To be added.

6. ALTO Service Discovery Guidelines

Key issues in ALTO Service Discovery are the following:

6.1. Delegation

6.2. NAT

6.3. Load Balancing and Robustness

7. Security Considerations

There are security considerations from the perspectives of both ISPs and P2P applications. See [ALTO-Problem] and [ALTO-Reqs] for discussions.

8. References

8.1. Normative References


8.2. Informative References


Appendix A. Contributors

Additional contributors:

- Richard Alimi, Yale
- Satish Raghunath, Juniper
- Richard Woundy, Comcast
- Yu-Shun Wang, Microsoft

Appendix B. Acknowledgments

The authors would like to thank the members of the p4p/alto/p2pi mailing lists for their insights. The concept of my-Internet view is from Emilio Sepulveda of Telefonica. The authors benefited from multiple discussions with Stefano Previdi and Anja Feldmann.

Authors’ Addresses

Y. Richard Yang (editor)
Yale University
EMail: yry@cs.yale.edu

D. Pasko
Verizon
EMail: pasko@verizon.com
L. Popkin  
Pando Networks  
EMail: laird@pando.com

Reinaldo Penno  
Juniper  
EMail: rpenno@juniper.net

Stanislav Shalunov  
BitTorrent  
EMail: shalunov@bittorrent.com