Internet Engineering Task Force
Internet-Draft
Intended status: Informational
Expires: December 6, 2015

W. Roome, Ed.
Bell Laboratories,
Alcatel-Lucent

June 4, 2015

Interoperability testing of the ALTO Protocol
draft-roome-interop-ietf93-00

Abstract

The Application-Layer Traffic Optimization (ALTO) protocol is designed to allow entities with knowledge about the network infrastructure to export such information to applications that need to choose one or more endpoints to connect to among large sets of logically equivalent ones. This document defines a data set that may be used to test the functionality and interoperability of ALTO clients and servers.

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 December 6, 2015.

Table of Contents

1. Overview

The Application-Layer Traffic Optimization (ALTO) protocol is designed to allow entities with knowledge about the network infrastructure to export such information to applications that need to choose one or more endpoints to connect to among large sets of logically equivalent ones.

This document defines procedures to test the functionality and interoperability of ALTO clients and servers.

This document is informational and is NOT NORMATIVE on any aspects of the ALTO protocol. The normative behavior of ALTO entities is prescribed in [RFC7285].

Section 2 defines the network maps, cost maps and other data necessary to provision an ALTO server. This ensures that all tested servers will return the same results, so a client may verify that a server is operating correctly. Section 3 defines the required and optional resources for an ALTO server to provide. Section 4 describes the actions expected from a client. Section 5 describes a set of invalid client requests, to verify that a server can respond correctly to client errors.

While every effort has been made to catalogue representative test cases, this document does not attempt to codify every test case that arises in ALTO. The aim of the document is to focus on areas that highlight the key offerings of the ALTO protocol.

2. Server Data

This section defines the data necessary to provision a tested ALTO server in a uniform manner. First it defines a default network map, and associated cost maps for the "routingcost" and "hopcount" metrics. Next it defines an optional alternate network map, along with "routingcost" and "hopcount" costs for that map. Finally it defines a set of optional endpoint properties.

Appendix A gives network and cost map data defined in this section formatted in JSON.

2.1. Default Network Map And Cost Maps

Every tested ALTO server MUST provide a default network map with the PIDs defined below:

PID	IP Address Block
mine	100.0.0.0/8
mine1	100.0.0.0/10
mine1a	100.0.1.0/24, 100.0.64.0/24, 100.0.192.0/24
mine2	100.64.0.0/10
mine3	100.128.0.0/10
peer1	128.0.0.0/16, 130.0.0.0/16, 2001:DB8:0000::/33
peer2	129.0.0.0/16, 131.0.0.0/16, 2001:DB8:8000::/33
tran1	132.0.0.0/16
tran2	135.0.0.0/16
default	0.0.0.0/0, ::0/0
loopback	127.0.0.0/8, ::1/128
linklocal	169.254.0.0/16, ff80::/10
private	10.0.0.0/8, 172.16.0.0/12, 192.168.0.0/16, fc00::/7

Figure 1: Default Network Map

Each ALTO server MUST provide a cost map for the "routingcost" metric. The following table presents the numerical values for those costs. If a server provides a numerical-mode cost map, it MUST use these values. If a server provides an ordinal-mode cost map, the server may use whatever values it wants, provided the ordinal values preserve the order of the numerical values.

	default	mine	mine1	mine1a	mine2	mine3	peer1	peer2	tran1	tran2
default	1.0	75.0	75.0	75.0	75.0	75.0	-	-	-	-
mine	75.0	1.0	15.0	15.0	15.0	15.0	30.0	30.0	50.0	50.0
mine1	75.0	15.0	1.0	2.5	5.0	7.0	20.0	25.0	40.0	45.0
mine1a	75.0	15.0	2.0	1.0	7.0	9.0	22.0	24.0	42.0	48.0
mine2	75.0	15.0	5.5	7.0	1.0	6.0	23.0	25.0	43.0	46.0
mine3	75.0	15.0	7.0	9.0	6.0	1.0	25.0	28.0	45.0	49.0
peer1	-	30.0	20.0	22.0	23.0	25.0	1.0	-	-	-
peer2	-	30.0	25.0	24.0	25.0	28.0	-	1.0	-	-
tran1	-	50.0	40.0	42.0	43.0	45.0	-	-	1.0	-
tran2	-	50.0	45.0	48.0	46.0	49.0	-	-	-	1.0

Figure 2: "routingcost" Numerical Cost Map

Note that this is a partial cost map, in that it does not define a cost for every source and destination PID.

Each ALTO server MAY provide a cost map for the "hopcount" metric. The following table gives the numerical values. As with

"routingcost", a numerical-mode cost map MUST use these values, and an ordinal-mode cost map may use any values consistent with this ordering.

	default	mine	mine1	mine1a	mine2	mine3	peer1	peer2	tran1	tran2
default	1.0	10.0	10.0	10.0	10.0	10.0	-	-	-	-
mine	10.0	1.0	3.0	3.0	3.0	3.0	5.0	6.0	8.0	8.0
mine1	10.0	3.0	1.0	2.0	2.0	2.0	4.0	5.0	6.0	7.0
mine1a	10.0	3.0	2.0	1.0	2.0	3.0	5.0	6.0	7.0	8.0
mine2	10.0	3.0	2.0	2.0	1.0	2.0	4.0	5.0	6.0	7.0
mine3	10.0	3.0	2.0	3.0	2.0	1.0	4.0	5.0	6.0	7.0
peer1	-	5.0	4.0	5.0	4.0	4.0	1.0	-	-	-
peer2	-	6.0	5.0	6.0	5.0	5.0	-	1.0	-	-
tran1	-	8.0	6.0	7.0	6.0	6.0	-	-	1.0	-
tran2	-	8.0	7.0	8.0	7.0	7.0	-	-	-	1.0

Figure 3: "hopcount" Numerical Cost Map

2.2. Alternate Network Map And Cost Maps

Every tested ALTO server MAY provide an alternate, or secondary, network map with the PIDs defined below:

PID	IP Address Block
dc1	101.0.0.0/16
dc2	102.0.0.0/16
dc3	103.0.0.0/16
dc4	104.0.0.0/16
user1	201.0.0.0/16
user2	202.0.0.0/16
user3	203.0.0.0/16
user4	204.0.0.0/16
default	0.0.0.0/0, ::0/0
loopback	127.0.0.0/8, ::1/128
linklocal	169.254.0.0/16, ff80::/10
private	10.0.0.0/8, 172.16.0.0/12, 192.168.0.0/16, fc00::/7

Figure 4: Alternate Network Map

Each ALTO server MAY provide a cost map for the "routingcost" metric for the alternate network map. The following table presents the numerical values for those costs. If a server provides a numerical-mode cost map, it MUST use these values. If a server provides an ordinal-mode cost map, the server may use whatever values it wants,

provided the ordinal values preserve the order of the numerical values.

	dc1	dc2	dc3	dc4	default	user1	user2	user3	user4
dc1	0.0	5.0	5.0	5.0	50.0	10.0	20.0	30.0	40.0
dc2	5.0	0.0	5.0	5.0	50.0	20.0	10.0	20.0	30.0
dc3	5.0	5.0	0.0	5.0	50.0	30.0	20.0	10.0	20.0
dc4	5.0	5.0	5.0	0.0	50.0	40.0	30.0	20.0	10.0
default	50.0	50.0	50.0	50.0	0.0	-	-	-	-
user1	10.0	20.0	30.0	40.0	-	0.0	-	-	-
user2	20.0	10.0	20.0	30.0	-	-	0.0	-	-
user3	30.0	20.0	10.0	20.0	-	-	-	0.0	-
user4	40.0	30.0	20.0	10.0	-	-	-	-	0.0

Figure 5: "routingcost" Numerical Cost Map

Note that this is a partial cost map, in that it does not define a cost for every source and destination PID.

Each ALTO server MAY provide a cost map for the "hopcount" metric. The following table gives the numerical values. As with "routingcost", a numerical-mode cost map MUST use these values, and an ordinal-mode cost map may use any values consistent with this ordering.

	dc1	dc2	dc3	dc4	default	user1	user2	user3	user4
dc1	0.0	1.0	1.0	1.0	8.0	3.0	4.0	5.0	6.0
dc2	1.0	0.0	1.0	1.0	8.0	4.0	3.0	4.0	5.0
dc3	1.0	1.0	0.0	1.0	8.0	5.0	4.0	3.0	4.0
dc4	1.0	1.0	1.0	0.0	8.0	6.0	5.0	4.0	3.0
default	8.0	8.0	8.0	8.0	0.0	-	-	-	-
user1	3.0	4.0	5.0	6.0	-	0.0	-	-	-
user2	4.0	3.0	4.0	5.0	-	-	0.0	-	-
user3	5.0	4.0	3.0	4.0	-	-	-	0.0	-
user4	6.0	5.0	4.0	3.0	-	-	-	-	0.0

Figure 6: "hopcount" Numerical Cost Map

2.3. Endpoint Properties

An ALTO server may provide the private endpoint property "priv:ietf-type" with the following values for endpoints in the indicated address blocks:

Value	IP Address Block
mine	100.0.0.0/8
peer	128.0.0.0/6, 2001:DB8::/32
transit	132.0.0.0/16, 135.0.0.0/16

Figure 7: Values for "priv:ietf-type" endpoint property

3. Server Resources and Configuration

An ALTO server MUST provide the following resources, as required by [RFC7285]:

- An Information Resource Directory (IRD) which describes all of the server's resources.
- A Network Map resource for the default network defined above.
- A Cost Map resource for the "routingcost" metric for the default network map. The mode may be either "numerical" or "ordinal". If "numerical", the values MUST be identical to those defined above. If "ordinal", the server can use whatever values it wants, but the ordering MUST be consistent with the ordering of the "numerical" values.
- An Endpoint Property Service for the "pid" property for the default network map.

A server MAY provide whatever additional resources it desires, as long as they are consistent with the network maps, cost maps and endpoint properties defined in Section 2. In particular, a server may provide:

- An additional Network Map resource, using the PIDs and address prefixes for the alternate network map defined above.
- Cost Map resources for the "routingcost" and/or "hopcount" metrics, in either "numerical" or "ordinal" modes, using the values defined above.
- Filtered Network Map resources for either or both network maps.
- Filtered Cost Map resources for any combination of "routingcost" and "hopcount" metrics, in either "numerical" and "ordinal" modes, for either or both network maps. The resources may or may not accept constraint tests.

- Endpoint Cost Service(s) or any combination of "routingcost" and "hopcount" metrics, in either "numerical" and "ordinal" modes. The cost values MUST be consistent with those for the default network map. The resources may or may not accept constraint tests.

- Endpoint Property Service(s) for the custom endpoint properties defined above.

However, a server MUST NOT provide more than the two network maps defined in this document. This restriction simplifies testing, because it allows a client to automatically identify the alternate network map (e.g., any network map which is not the default must be the alternate network). If servers could offer three or more network maps, a client would have to be provisioned with the resource id of the alternate network map.

Note that if a server provides a Network Map resource for the alternate network map, [RFC7285] requires the server to also provide a Cost Map resource for the "routingcost" metric, in either "numerical" or "ordinal" mode, and an Endpoint Property Service for that network map's "pid" property.

A server MAY structure the IRD however it wants. In particular, a server may

- Use secondary IRDs which the root IRD references.

- Use arbitrary resource IDs and cost type names.

- Use arbitrary URIs, in any recognized URI format.

- Provide multiple versions of POST-mode resources. For example, if a server provides the secondary network map, it must provide an Endpoint Property Service for the "pid" properties for both maps. A server may provide one EPS for both properties, or a separate EPS for each property.

4. Client Actions

When given the URI for an ALTO server's IRD, an ALTO client should read the IRD, and for each resource that it recognizes, verify that the server returns the correct response. Note that most of the data the server returns is determined by the network maps, cost maps and property values specified in Section 2, and hence can be verified by a client. Some data cannot be determined a priori (e.g., resource id and tag of a network map), but a client can verify their consistency

(e.g., a cost map's dependent-vtag field should match the vtag field of the associated network map).

It is expected that not every client will be able recognize and verify every possible resource. However, each client MUST be able to verify the default network map and the associated "routingcost" cost map. In particular, although clients are not required to recognize the alternate network map, if presented with an IRD with two network maps, every client MUST be able to distinguish the default network map, and its associated cost map, from the alternate network map.

Ideally clients should be scripted. That is, when given the URI for a server, an ideal client would verify the server automatically, without further operator intervention. A client should log the resources it tested, and clearly highlight any response the client considered incorrect.

The HTTP GET-mode resources (Network Map and Cost Map) do not require client input, and hence testing is straight-forward: the client sends the appropriate HTTP GET request, and verifies the response.

However, POST-mode resources, such as Filtered Cost Maps and Endpoint Property Services, require client input. The following sections present recommended input parameters for various resources, and clients SHOULD implement as many of these tests as possible. Clients MAY add additional tests, and are encouraged to do so.

4.1. Filtered Network Map Tests

- Empty "pid" array, omitted or empty "address-types" array. The server should return the entire network map.
- Empty "pid" array, "address-types" array containing just "ipv6". The server should return PIDs with ipv6 addresses, and only those PIDs.
- "pid" array with one or more non-existent PID names, such as "not-a-pid". The server should return an empty network map.
- "pid" array with a set of valid PID names (client's choice), plus one or more non-existent PID names. The server should return the valid PIDs and ignore the invalid ones.

4.2. Filtered Cost Map Tests

All tests require an appropriate "cost-type" parameter. At a minimum, clients should run these tests for the "routingcost" metric for the default network map. If possible, clients should also run

these tests for the "hopcount" metric and the alternate network map.

Clients should remember that when testing "ordinal" costs, any values are acceptable as long as they are consistent with the order of the "numerical" costs defined in Section 2. Clients are also reminded that ordinal values are only comparable to other values in the same request, and a server may recalculate ordinal values for each request. Hence the same cost point may have ordinal value "6" in a full cost map, but have value "1" in a filtered cost map.

- Empty "srcs" and "dsts" arrays. The server should return the entire cost map.

- Empty "srcs" array, "dsts" array with one or more valid PIDs. The server should return costs from all PIDs to the specified destination PIDs.

- Empty "dsts" array, "srcs" array with one or more valid PIDs. The server should return costs from the specified source PIDs to all destination PIDs.

- "srcs" and "dsts" arrays with only non-existent PID names. The server should return an empty cost map.

- "srcs" and "dsts" arrays with a set of valid PID names (client's choice), plus one or more non-existent PID names in one or the arrays. The server should return costs for the valid PIDs and ignore the non-existent ones.

- The two-element constraint test "ge 20", "le 30" for the numerical "routingcost" for the default network map, with empty "srcs" and "dsts" arrays (assuming that resource allows constraints, of course). The server should return the all costs in the range, namely:

	mine	mine1	mine1a	mine2	mine3	peer1	peer2
mine	-	-	-	-	-	30.0	30.0
mine1	-	-	-	-	-	20.0	25.0
mine1a	-	-	-	-	-	22.0	24.0
mine2	-	-	-	-	-	23.0	25.0
mine3	-	-	-	-	-	25.0	28.0
peer1	30.0	20.0	22.0	23.0	25.0	-	-
peer2	30.0	25.0	24.0	25.0	28.0	-	-

Figure 8: Filtered Cost Map Constraint Test

4.3. Endpoint Property Service Tests

Every client should verify that the server's EPS resource for the default network's "pid" property returns the correct PID name for a representative set of endpoint addresses. If possible, clients should also verify the alternate network's "pid" property and the "priv:ietf-type" property.

The table below gives the expected values for a set of addresses. Clients are encouraged to test other addresses as well.

Address	def.pid	alt.pid	priv:ietf-type
ipv4:0.0.0.1	default	default	-
ipv4:10.1.2.3	private	private	-
ipv4:100.0.0.1	mine1	default	mine
ipv4:100.0.1.1	mine1a	default	mine
ipv4:100.0.192.1	mine1a	default	mine
ipv4:100.0.64.1	mine1a	default	mine
ipv4:100.130.0.1	mine3	default	mine
ipv4:100.200.0.1	mine	default	mine
ipv4:100.75.0.1	mine2	default	mine
ipv4:101.0.0.1	default	dc1	-
ipv4:101.1.0.1	default	default	-
ipv4:102.0.0.1	default	dc2	-
ipv4:103.0.0.1	default	dc3	-
ipv4:104.0.0.1	default	dc4	-
ipv4:127.0.0.1	loopback	loopback	-
ipv4:127.255.255.255	loopback	loopback	-
ipv4:128.0.0.1	peer1	default	peer
ipv4:129.0.0.1	peer2	default	peer
ipv4:130.0.0.1	peer1	default	peer
ipv4:131.0.0.1	peer2	default	peer
ipv4:132.0.0.1	tran1	default	transit
ipv4:135.0.0.1	tran2	default	transit
ipv4:169.254.1.2	linklocal	linklocal	-
ipv4:201.0.0.1	default	user1	-
ipv4:201.1.2.3	default	default	-
ipv4:202.0.0.1	default	user2	-
ipv4:203.0.0.1	default	user3	-
ipv4:204.0.0.1	default	user4	-
ipv4:99.0.0.1	default	default	-
ipv6:::1	loopback	loopback	-
ipv6:::2	default	default	-
ipv6:2001:db8::	peer1	default	peer
ipv6:2001:db8:8000::1	peer2	default	peer
ipv6:fc00:1::	private	private	-
ipv6:ff80:1:2::	linklocal	linklocal	-

Figure 9: EPS Test Addresses And Property Values

4.4. Endpoint Cost Service Tests

TBD!!!

5. Error Tests

TBD!!!

6. Security considerations

This document does not present any new security considerations above and beyond what is documented in the ALTO protocol [RFC7285].

7. IANA considerations

This document does not require any action from IANA.

8. Normative References

[RFC7159] Bray, T., "The JavaScript Object Notation (JSON) Data Interchange Format", RFC 7159, March 2014.

[RFC7285] Almi, R., Penno, R., Yang, Y., Kiesel, S., Previdi, S., Roome, W., Shalunov, S., and R. Woundy, "Application-Layer Traffic Optimization (ALTO) Protocol", RFC 7285, September 2014.

Appendix A. Appendix: JSON Network And Cost Maps

This section presents the network and cost maps defined in Section 2 formatted as JSON ([RFC7159]) objects.

A.1. Default Network Map

```
   "network-map": {
      "default": {
         "ipv4": ["0.0.0.0/0"],
         "ipv6": ["::/0"] },
      "linklocal": {
         "ipv4": ["169.254.0.0/16"],
         "ipv6": ["FF80::/10"] },
      "loopback": {
         "ipv4": ["127.0.0.0/8"],
         "ipv6": ["::1/128"] },
      "mine": {
         "ipv4": ["100.0.0.0/8"] },
      "mine1": {
         "ipv4": ["100.0.0.0/10"] },
      "mine1a": {
         "ipv4": ["100.0.64.0/24", "100.0.192.0/24", "100.0.1.0/24"] },
      "mine2": {
         "ipv4": ["100.64.0.0/10"] },
      "mine3": {
         "ipv4": ["100.128.0.0/10"] },
      "peer1": {
         "ipv4": ["130.0.0.0/16", "128.0.0.0/16"],
         "ipv6": ["2001:DB8::/33"] },
      "peer2": {
         "ipv4": ["131.0.0.0/16", "129.0.0.0/16"],
         "ipv6": ["2001:DB8:8000::/33"] },
      "private": {
         "ipv4": ["10.0.0.0/8", "172.16.0.0/12", "192.168.0.0/16"],
         "ipv6": ["FC00::/7"] },
      "tran1": {
         "ipv4": ["132.0.0.0/16"] },
      "tran2": {
         "ipv4": ["135.0.0.0/16"] }
   }
```

Figure 10: Default Network Map, in JSON

A.2. Default "routingcost" Cost Map

```
   "cost-map": {
      "default": {
         "default": 1.0,  "mine": 75.0,  "mine1": 75.0,  "mine1a": 75.0,
         "mine2": 75.0,  "mine3": 75.0 },
      "mine": {
         "default": 75.0,  "mine": 1.0,  "mine1": 15.0,  "mine1a": 15.0,
         "mine2": 15.0,  "mine3": 15.0,  "peer1": 30.0,  "peer2": 30.0,
         "tran1": 50.0,  "tran2": 50.0 },
      "mine1": {
         "default": 75.0,  "mine": 15.0,  "mine1": 1.0,  "mine1a": 2.5,
         "mine2": 5.0,  "mine3": 7.0,  "peer1": 20.0,  "peer2": 25.0,
         "tran1": 40.0,  "tran2": 45.0 },
      "mine1a": {
         "default": 75.0,  "mine": 15.0,  "mine1": 2.0,  "mine1a": 1.0,
         "mine2": 7.0,  "mine3": 9.0,  "peer1": 22.0,  "peer2": 24.0,
         "tran1": 42.0,  "tran2": 48.0 },
      "mine2": {
         "default": 75.0,  "mine": 15.0,  "mine1": 5.5,  "mine1a": 7.0,
         "mine2": 1.0,  "mine3": 6.0,  "peer1": 23.0,  "peer2": 25.0,
         "tran1": 43.0,  "tran2": 46.0 },
      "mine3": {
         "default": 75.0,  "mine": 15.0,  "mine1": 7.0,  "mine1a": 9.0,
         "mine2": 6.0,  "mine3": 1.0,  "peer1": 25.0,  "peer2": 28.0,
         "tran1": 45.0,  "tran2": 49.0 },
      "peer1": {
         "mine": 30.0,  "mine1": 20.0,  "mine1a": 22.0,  "mine2": 23.0,
         "mine3": 25.0,  "peer1": 1.0 },
      "peer2": {
         "mine": 30.0,  "mine1": 25.0,  "mine1a": 24.0,  "mine2": 25.0,
         "mine3": 28.0,  "peer2": 1.0 },
      "tran1": {
         "mine": 50.0,  "mine1": 40.0,  "mine1a": 42.0,  "mine2": 43.0,
         "mine3": 45.0,  "tran1": 1.0 },
      "tran2": {
         "mine": 50.0,  "mine1": 45.0,  "mine1a": 48.0,  "mine2": 46.0,
         "mine3": 49.0,  "tran2": 1.0 }
      }
```

Figure 11: Default "routingcost" Cost Map, in JSON

A.3. Default "hopcount" Cost Map

```
   "cost-map": {
      "default": {
         "default": 1.0,  "mine": 10.0,  "mine1": 10.0,  "mine1a": 10.0,
         "mine2": 10.0,  "mine3": 10.0 },
      "mine": {
         "default": 10.0,  "mine": 1.0,  "mine1": 3.0,  "mine1a": 3.0,
         "mine2": 3.0,  "mine3": 3.0,  "peer1": 5.0,  "peer2": 6.0,
         "tran1": 8.0,  "tran2": 8.0 },
      "mine1": {
         "default": 10.0,  "mine": 3.0,  "mine1": 1.0,  "mine1a": 2.0,
         "mine2": 2.0,  "mine3": 2.0,  "peer1": 4.0,  "peer2": 5.0,
         "tran1": 6.0,  "tran2": 7.0 },
      "mine1a": {
         "default": 10.0,  "mine": 3.0,  "mine1": 2.0,  "mine1a": 1.0,
         "mine2": 2.0,  "mine3": 3.0,  "peer1": 5.0,  "peer2": 6.0,
         "tran1": 7.0,  "tran2": 8.0 },
      "mine2": {
         "default": 10.0,  "mine": 3.0,  "mine1": 2.0,  "mine1a": 2.0,
         "mine2": 1.0,  "mine3": 2.0,  "peer1": 4.0,  "peer2": 5.0,
         "tran1": 6.0,  "tran2": 7.0 },
      "mine3": {
         "default": 10.0,  "mine": 3.0,  "mine1": 2.0,  "mine1a": 3.0,
         "mine2": 2.0,  "mine3": 1.0,  "peer1": 4.0,  "peer2": 5.0,
         "tran1": 6.0,  "tran2": 7.0 },
      "peer1": {
         "mine": 5.0,  "mine1": 4.0,  "mine1a": 5.0,  "mine2": 4.0,
         "mine3": 4.0,  "peer1": 1.0 },
      "peer2": {
         "mine": 6.0,  "mine1": 5.0,  "mine1a": 6.0,  "mine2": 5.0,
         "mine3": 5.0,  "peer2": 1.0 },
      "tran1": {
         "mine": 8.0,  "mine1": 6.0,  "mine1a": 7.0,  "mine2": 6.0,
         "mine3": 6.0,  "tran1": 1.0 },
      "tran2": {
         "mine": 8.0,  "mine1": 7.0,  "mine1a": 8.0,  "mine2": 7.0,
         "mine3": 7.0,  "tran2": 1.0 }
      }
```

Figure 12: Default "hopcount" Cost Map, in JSON

A.4. Alternate Network Map

```
    "network-map": {
       "dc1": {
          "ipv4": ["101.0.0.0/16"] },
       "dc2": {
          "ipv4": ["102.0.0.0/16"] },
       "dc3": {
          "ipv4": ["103.0.0.0/16"] },
       "dc4": {
          "ipv4": ["104.0.0.0/16"] },
       "default": {
          "ipv4": ["0.0.0.0/0"],
          "ipv6": ["::/0"] },
       "linklocal": {
          "ipv4": ["169.254.0.0/16"],
          "ipv6": ["FF80::/10"] },
       "loopback": {
          "ipv4": ["127.0.0.0/8"],
          "ipv6": ["::1/128"] },
       "private": {
          "ipv4": ["10.0.0.0/8", "172.16.0.0/12", "192.168.0.0/16"],
          "ipv6": ["FC00::/7"] },
       "user1": {
          "ipv4": ["201.0.0.0/16"] },
       "user2": {
          "ipv4": ["202.0.0.0/16"] },
       "user3": {
          "ipv4": ["203.0.0.0/16"] },
       "user4": {
          "ipv4": ["204.0.0.0/16"] }
    }
```

Figure 13: Alternate Network Map, in JSON

A.5. Alternate "routingcost" Cost Map

```
  "cost-map": {
     "dc1": {
        "dc1": 0.0,  "dc2": 5.0,  "dc3": 5.0,  "dc4": 5.0,
        "default": 50.0,  "user1": 10.0,  "user2": 20.0,  "user3": 30.0,
        "user4": 40.0 },
     "dc2": {
        "dc1": 5.0,  "dc2": 0.0,  "dc3": 5.0,  "dc4": 5.0,
        "default": 50.0,  "user1": 20.0,  "user2": 10.0,  "user3": 20.0,
        "user4": 30.0 },
     "dc3": {
        "dc1": 5.0,  "dc2": 5.0,  "dc3": 0.0,  "dc4": 5.0,
        "default": 50.0,  "user1": 30.0,  "user2": 20.0,  "user3": 10.0,
        "user4": 20.0 },
     "dc4": {
        "dc1": 5.0,  "dc2": 5.0,  "dc3": 5.0,  "dc4": 0.0,
        "default": 50.0,  "user1": 40.0,  "user2": 30.0,  "user3": 20.0,
        "user4": 10.0 },
     "default": {
        "dc1": 50.0,  "dc2": 50.0,  "dc3": 50.0,  "dc4": 50.0,
        "default": 0.0 },
     "user1": {
        "dc1": 10.0,  "dc2": 20.0,  "dc3": 30.0,  "dc4": 40.0,
        "user1": 0.0 },
     "user2": {
        "dc1": 20.0,  "dc2": 10.0,  "dc3": 20.0,  "dc4": 30.0,
        "user2": 0.0 },
     "user3": {
        "dc1": 30.0,  "dc2": 20.0,  "dc3": 10.0,  "dc4": 20.0,
        "user3": 0.0 },
     "user4": {
        "dc1": 40.0,  "dc2": 30.0,  "dc3": 20.0,  "dc4": 10.0,
        "user4": 0.0 }
     }
```

Figure 14: Alternate "routingcost" Cost Map, in JSON

A.6. Alternate "hopcount" Cost Map

```
    "cost-map": {
       "dc1": {
          "dc1": 0.0,  "dc2": 1.0,  "dc3": 1.0,  "dc4": 1.0,
          "default": 8.0,  "user1": 3.0,  "user2": 4.0,  "user3": 5.0,
          "user4": 6.0 },
       "dc2": {
          "dc1": 1.0,  "dc2": 0.0,  "dc3": 1.0,  "dc4": 1.0,
          "default": 8.0,  "user1": 4.0,  "user2": 3.0,  "user3": 4.0,
          "user4": 5.0 },
       "dc3": {
          "dc1": 1.0,  "dc2": 1.0,  "dc3": 0.0,  "dc4": 1.0,
          "default": 8.0,  "user1": 5.0,  "user2": 4.0,  "user3": 3.0,
          "user4": 4.0 },
       "dc4": {
          "dc1": 1.0,  "dc2": 1.0,  "dc3": 1.0,  "dc4": 0.0,
          "default": 8.0,  "user1": 6.0,  "user2": 5.0,  "user3": 4.0,
          "user4": 3.0 },
       "default": {
          "dc1": 8.0,  "dc2": 8.0,  "dc3": 8.0,  "dc4": 8.0,
          "default": 0.0 },
       "user1": {
          "dc1": 3.0,  "dc2": 4.0,  "dc3": 5.0,  "dc4": 6.0,
          "user1": 0.0 },
       "user2": {
          "dc1": 4.0,  "dc2": 3.0,  "dc3": 4.0,  "dc4": 5.0,
          "user2": 0.0 },
       "user3": {
          "dc1": 5.0,  "dc2": 4.0,  "dc3": 3.0,  "dc4": 4.0,
          "user3": 0.0 },
       "user4": {
          "dc1": 6.0,  "dc2": 5.0,  "dc3": 4.0,  "dc4": 3.0,
          "user4": 0.0 }
       },
```

Figure 15: Alternate "hopcount" Cost Map, in JSON

Appendix B. Acknowledgements

The editors will like to thank the ALTO working group participants for reviewing test cases. Richard Alimi and Mikio Hara contributed review cycles to the contents of this document.

Author's Address

Wendy Roome (editor)
Bell Laboratories, Alcatel-Lucent

Email: w.roome@alcatel-lucent.com