ALTO Extension: Path Vector

draft-ietf-alto-path-vector-11

Kai Gao, Young Lee, Sabine Randriamasy, Y. Richard Yang, Jensen Zhang

July 27, IETF 108@Virtual
Summary

- The Path Vector extension aims to provide abstracted network components along the path between a source and a destination
- Use cases: bandwidth reservation for large-scale data analytics, context-aware data transfers (capabilities of access network, etc.), CDN and service edge discovery
- Key abstractions:
  - **Abstract Network Element (ANE):** A network-agnostic abstraction of network components, associated with different properties
  - **Path Vector (PV):** An array of ANE names that represents the ANEs on the path between a <src, dst> pair
  - **ANE Property Map:** A Property Map that maps ANE names to the associated properties
- Key designs:
  - The Path Vector extension extends the Cost Map and Endpoint Cost Map to convey how each <src, dst> pair traverses the ANEs: **a new cost type is defined**
  - The Path Vector extension uses Property Map to convey the association between ANEs and their properties: **a new domain is defined, two initial properties are defined, entries are added to corresponding IANA registries following the procedures defined in the Unified Property draft**
  - For consistency and server security, the two pieces of information are provided together in **a multipart response.**
- Current status: Waiting to issue WGLC
The problem and proposed solution are better clarified

- We summarize 3 general additional requirements using the flow scheduling example, along other use cases in the broader context
- ... and present how the extension fulfills the requirements
- In particular, we discuss **ephemeral ANE** (returned by the Path Vector response) and **persistent ANE** (defined by another Unified Property Map resource) and how they are handled

*The network needs to expose more detailed routing information to show the shared bottlenecks.*

*The network needs to provide the necessary abstraction to hide the real topology information while providing enough information to applications.*

The path vector extension defined in this document propose a solution

In general, we can conclude that to support the multiple flow scheduling use case, the ALTO framework must be extended to satisfy the following additional requirements:

AR1: An ALTO server must provide essential information on intermediate network components on the path of a <source, destination> pair that are critical to the QoE of the overlay application.

AR2: An ALTO server must provide essential information on how the paths of different <source, destination> pairs share a common network component.

AR3: An ALTO server must provide essential information on the properties associated to the network components.

The Path Vector extension defined in this document propose a solution to provide these details.

**4.2. Recent Use Cases**

While the multiple flow scheduling problem is used to help identify the additional requirements, the Path Vector extension can be applied to a wide range of applications. This section highlights some real use cases that are recently reported. See [I-D.bernstein-alto-top] for a more comprehensive survey of use cases where extended network topology information is needed.
Updates in -11

Path Vector draft is now synchronized with Unified Property Map draft (version -12)

• ANE domain and ANE properties are defined and registered to IANA following the new procedure

• In particular, the defining resource of ANE is now Property Map (with media type `application/alto-propmap+json`) which apply to both ephemeral ANE and persistent ANE

• The entity identifiers used in the ANE Property Map and in the `persistent-entity-id` follow the naming convention of entity identifiers in the Unified Property Map draft
The specification is better clarified

- Fixed the inconsistent use of the "start" parameter in multipart messages
- Examples are added for request/response
- Fixed inappropriate/inconsistent letter cases (e.g., CAN -> can in a normal sentence, path vector -> Path Vector throughout the paper)

Old version is using “first” part and “second” part to refer to parts in a multipart message, which may be incorrect when “start” parameter is used
Conclusion

• A lot of clarification to make the document easier to follow
• PV-11 is fully synchronized with UP-12, and can move ahead as a bundle

• Issue WGLC?
Backup
Updates in -11

• The problem and proposed solution are better clarified
  • We summarize 3 general additional requirements using the flow scheduling example, along other use cases in the broader context
  • ... and present how the extension fulfills the requirements
  • In particular, we discuss ephemeral ANE (returned by the Path Vector response) and persistent ANE (defined by another Unified Property Map resource) and how they are handled

• Path Vector draft is now synchronized with Unified Property Map draft (version -12)
  • ANE domain and ANE properties are defined and registered to IANA following the new procedure
  • In particular, the defining resource of ANE is now Property Map (with media type application/alto-propmap+json) which apply to both ephemeral ANE and persistent ANE
  • The entity identifiers used in the ANE Property Map and in the persistent-entity-id follow the naming convention of entity identifiers in the Unified Property Map draft

• The specification is better clarified
  • Fixed the inconsistent use of the “start” parameter in multipart messages
  • Examples are added for request/response
  • Fixed inappropriate/inconsistent letter cases (e.g., CAN -> can in a normal sentence, path vector -> Path Vector throughout the paper)