

Network Working Group
Internet-Draft
Intended status: Standards Track
Expires: April 02, 2014

R. Penno
T. Reddy
Cisco Systems, Inc.
M. Boucadair
France Telecom
D. Wing
Cisco
S. Vinapamula
Juniper Networks, Inc.
September 29, 2013

Application Enabled SDN (A-SDN)
draft-penno-pcp-asdn-00

Abstract

To allow traversal of firewalls or provide additional network services such as QoS or supplemental bandwidth, it is necessary to deploy application-aware network elements. Such network elements are costly to create, deploy, and are unable to adequately cope with changes to the application itself, stifling innovation.

This document describes a different approach, where the application explicitly signals its needs to the network.

Requirements Language

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 [RFC 2119](#) [[RFC2119](#)].

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 April 02, 2014.

Internet-Draft

A-SDN

September 2013

Copyright Notice

Copyright (c) 2013 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.	Problem Statement	2
2.	Scope	3
3.	Proposed Approach	4
4.	Protocols	6
5.	A-SDN Flows	6
5.1.	Signaling Prior to Flow Creation	7
5.2.	Signaling After Flow Creation	8
5.3.	Flow Removal Event	8
5.4.	Flow Modification	8
6.	Use Cases	8
6.1.	Flow Prioritization	8
6.2.	Flow High availability	9
6.3.	On-demand Bandwidth	9
6.4.	Analytics and Reporting	9
7.	Security Considerations	9
8.	IANA Considerations	9
9.	Acknowledgments	9
10.	References	10
10.1.	Normative References	10
10.2.	Informative References	10
	Authors' Addresses	11

[1.](#) Problem Statement

In the context of ongoing efforts to add more automation and promote means to dynamically interact with network resources (e.g., SDN-

labeled efforts) [[I-D.sin-sdnrg-sdn-approach](#)], various proposals are made to accommodate the needs of Network Providers to program the network with flow information and its associated metadata in order to apply policies such as traffic prioritization.

Usually this programming is driven by a (centralized) controller that gather flow-related information and associated metadata through an army of probes, receiving a copy of the first packets of the flow, or even having to be on-path for the first few packets of the flow but not necessarily subsequent packets. But most of observed flows in current usages are dynamic, time-bound (short lived for some of them), possibly encrypted, peer-to-peer, possibly asymmetric, and might have different priorities depending on network conditions, direction, time of the day, and other factors.

This means that hairpinning of packets through a controller, deep packet inspection, and other similar static methods such as portals cannot be employed successfully to glean flow and metadata information, and subsequently program the network. Therefore new methods must be devised.

Unlike network-centric techniques, this document proposed an approach which involved hosts and applications.

[2.](#) Scope

Considerations related to dynamic network provisioning negotiation are out of scope. The reader can refer to [[I-D.boucadair-connectivity-provisioning-protocol](#)] for more details.

The proposed architecture is not a replacement to existing legacy techniques. It is an enhancement to existing network infrastructure and service infrastructure than can be empowered by new features to better accommodate application-specific needs while network and services resources are also optimized and better partitioned.

This document does not propose to update all existing/future applications to signal their network resources requirements; only a subset of applications having specific connectivity requirements and which require differentiated treatment at the network side are expected to be updated to support the framework defined in this

document.

This document does not require an end-to-end signalling before actual invocation of a service.

This document does not make any assumption on how differentiated connectivity is delivered to end users. It is up to each administrative entity managing a network to enforce its own engineering policies, techniques and protocols. Note, differentiated connectivity services can be provided by one or a combination of several dimension (forwarding, routing, resources management). It is out of scope of this document to elaborate on such aspects.

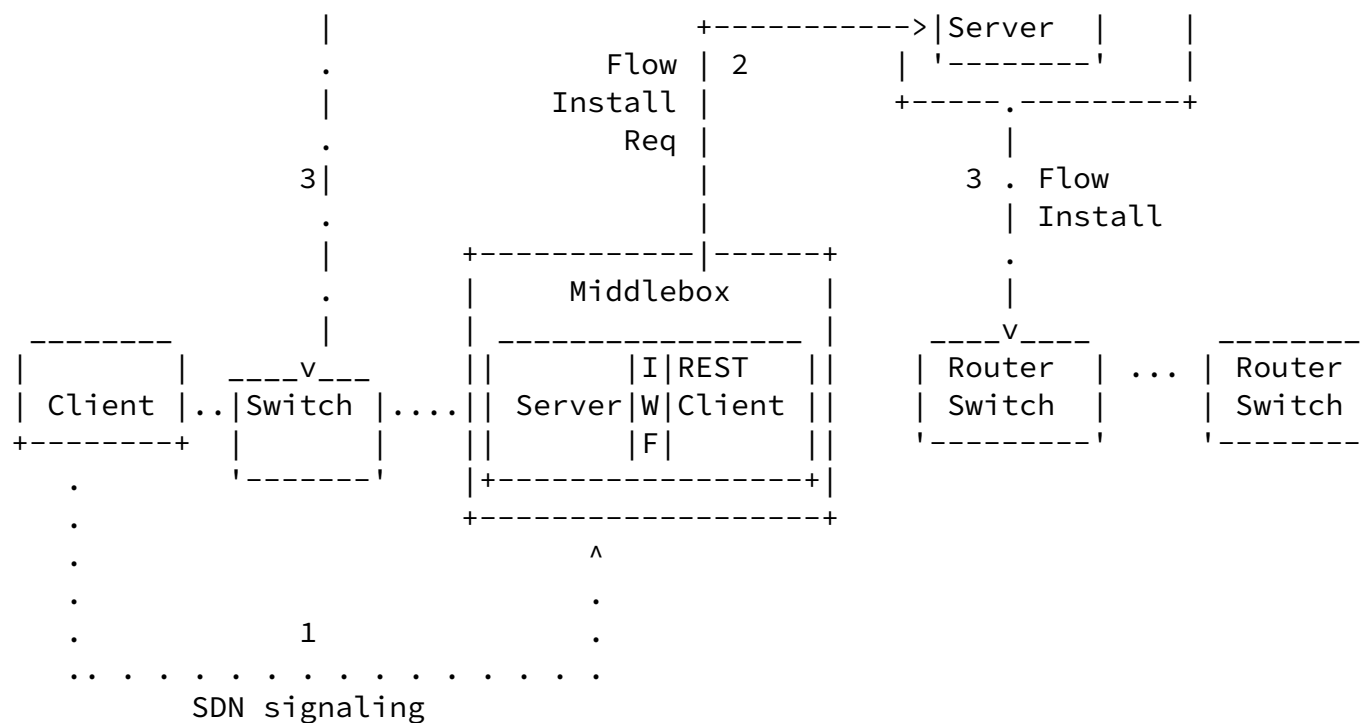
3. Proposed Approach

In order to offer more automation and dynamicity in resource usage and invocation, this document proposes an architecture that is composed of three parts:

1. Applications running on the end points (UEs, Server at a Data Centers, CPE routers) must communicate or install flow and associated metadata on network Elements. Means to discover such Network Elements may be supported.
2. On the network side, a PDP (Policy Decision Point, [[RFC2753](#)]) is responsible for orchestrating resources, generating policies and trigger provisioning-related operations.
3. The PDP configures the on-path devices to accommodate the signaled flow (e.g., open pinhole in the firewall, provide prioritized network services for the flow).

The diagram below depicts the general architecture and message flow for the Application-Enabled SDN (A-SDN).





Request (Flow + Metadata)

- e.g., PCP
- e.g., REST
- .-. e.g., COPS-PR, Netconf, Openflow

A middlebox could be a CPE router, edge router, switch, wireless access LAN controller, mobile gateway in 3GPP networks [[RFC6459](#)], or any other flow-aware device.

This architecture provides several advantages such as:

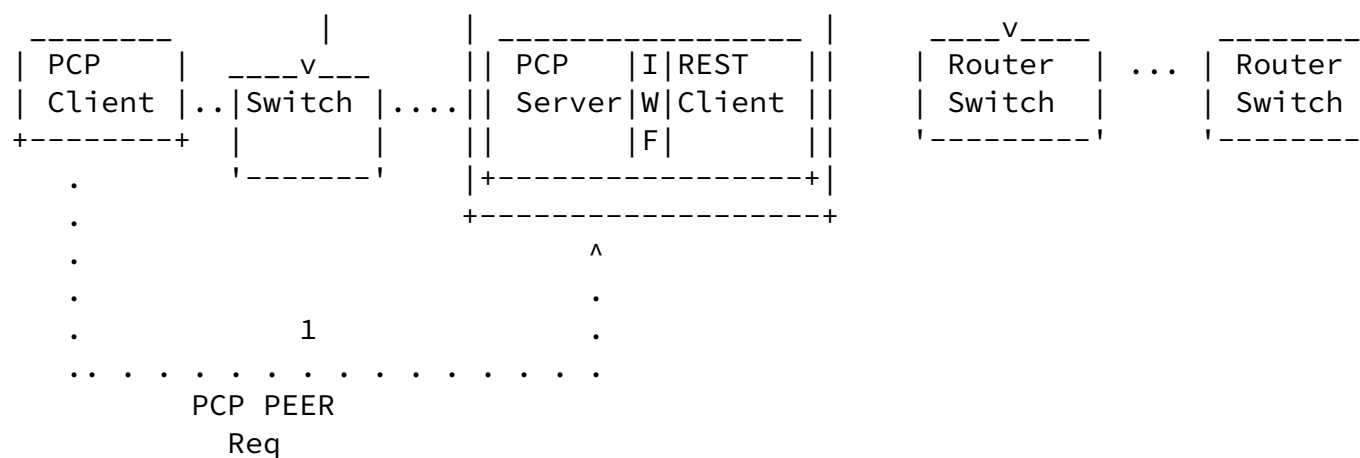
- o Host driven: The host (or application) is responsible for requesting proper flows and associated metadata based on each individual application needs. These needs may be time variant, and driven by processes only understood by the applications (or their users). The end host is the only entity in the system that has all of the information required to make the correct service request . This approach is compliant with requirements specific to encrypted and multi-party flows.

- o Network Authorization: If network access control is required, then the host could also get authorization from the Application Server trusted by the network in order to install flows and associated actions (e.g., policies). The Application Server could be deployed in a third party network. This is important for networks which do not trust the host.
- o Immediate incremental value for endpoints and applications: If, for example, a CPE router that supports this architecture is installed, applications could signal flow characteristics to the network on both directions, traffic prioritization, firewall pinholes and other services without changing the rest of the network. Meaning, although steps 2 and 3 of the picture above provide important end-to-end additional value they are not necessary for end-to-edge.
- o Access agnostic: An application should not care if it is on an ADSL, Cable, Wi-Fi, 3G, Ethernet or other network type.
- o Works across administrative domains: Home Network -> ISP1. Home Network communicates with ISP1 using PCP.
- o NAT and firewall aware: The flow information fed into the PDP will have pre and post NAT information, allowing provisioning using scoped IP addresses.

- o Extensible: Client protocol can be extended to provide a wide range of flow associated metadata.
- o Multi-interface support: Based on network conditions clients can switch from a Wi-Fi to a 3G interface, or install flows over certain paths

4. Protocols

The first element of this architecture could be met by using the Port Control Protocol (PCP) [[RFC6887](#)]. Indeed, PCP Flow Extension [[I-D.wing-pcp-flowdata](#)] allows a PCP Client, usually a host, to signal flow characteristics to the network, and the network to signal its ability to accommodate that flow back to the host.



.... PCP Message
 ---- REST Messages
 -.-. Netconf, COPS, etc.

5.1. Signaling Prior to Flow Creation

When an end host installs a flow in the middlebox through a PCP message a REST API call is made to the PDP. This message will carry the following information:

- o Match condition: e.g., source/destination IP, source/destination port, L4 Protocol, Port, VLAN Id etc.
- o Metadata: e.g., metadata conveyed in PCP FLOWDATA option.
- o Lifetime: e.g., lifetime in PCP response will be mapped to idle_timeout and hard_timeout will be set to zero for the flow entry. (idle_timeout and hard_timeout are defined in OpenFlow switching protocol). This way PCP client is aware when the flow entry will be removed.

The PDP uses an appropriate protocol (e.g., netconf, COPS-PR, Openflow, etc.) to add/delete and modify flows and its metadata. For example Openflow controller using Openflow protocol version 1.3 [[OpenFlow](#)] would get the information of configured queues and

associated property of each queue. The Openflow controller will

either associate the flow with relevant queue or instruct the openflow-enabled network device to rewrite the DifServ CodePoint bits for the flow based on the metadata in REST message.

[5.2.](#) Signaling After Flow Creation

The application can create a implicit flow normally as with a TCP connection and later decide that it needs to modify it, for example, extending its lifetime or associating metadata such as bandwidth, delay, jitter, loss.

The mechanism is very similar to flow creation but does not require a pre-signaling step.

[5.3.](#) Flow Removal Event

When a application-driven flow times out or is explicitly deleted, a REST API call is generated in the case the controller wants to be notified. This allows the PDP to delete the flow from other devices in the network.

The PDP could also decide on its own to remove the installed flow. In this case a PCP unsolicited response will be sent to the PCP Client owner of such flow.

[5.4.](#) Flow Modification

After the PDP is notified of a flow creation, it can decide to modify its metadata. In order to do that the controller will send modify flow message through the appropriate protocol.

If the PDP succeeds in modifying a flow, a PCP unsolicited response will be sent to the PCP Client owner of such flow.

[6.](#) Use Cases

This section describes some use-cases in which A-SDNs can be beneficial.

[6.1.](#) Flow Prioritization

A video streaming client that wants to have a low loss, medium delay service signals these flow characteristics in PCP FLOWDATA option. PCP server would convey this metadata to a PDP which would in turn add flow entry with inbound DSCP AF32 on SDN-enabled network devices.

Packets matching this flow will be marked AF32 and internally put in an appropriate queue. More importantly, video packets should be marked as close as possible to the source.

[6.2.](#) Flow High availability

One of the ways for the PCP Server to determine that the flows are for business critical application is by using third party authorization. A PCP server for such flows will checkpoint all the state associated for such flows on the corresponding backup of active for high availability. At a high level, this authorization works by the PCP client first obtaining a cryptographic token from the authorizing network element (e.g., call controller) and includes that token in the PCP request. The PCP server in the network validates the token and grants access.

[6.3.](#) On-demand Bandwidth

In managed or unmanaged services deployments an enterprise many times needs more bandwidth for the entire link (all flows) or just some specific applications. Moreover, it does not need those permanently but just for a certain period of time. In this case the branch router can dynamically request this service from the network, streamlining service activation and modification.

[6.4.](#) Analytics and Reporting

Authorized applications within data centers and enterprises can attach metadata such as media-type, application-id and group to the flows which allows for ease and streamlined analytics and reporting without deep packet inspection.

[7.](#) Security Considerations

Security considerations in [[RFC6887](#)] and PCP Authentication [[I-D.ietf-pcp-authentication](#)] may need to be taken into account. For REST mutual authentication is required and TLS could be used for message integrity. Security-related consideration for the protocol enabled between the PDP and underlying nodes are discussed in [[RFC6241](#)] [[RFC3084](#)].

[8.](#) IANA Considerations

This document does not require any action from IANA.

[9.](#) Acknowledgments

TODO.

Internet-Draft

A-SDN

September 2013

[10.](#) References

[10.1.](#) Normative References

- [I-D.wing-pcp-flowdata]
Wing, D., Penno, R., and T. Reddy, "PCP Flowdata Option", [draft-wing-pcp-flowdata-00](#) (work in progress), July 2013.
- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", [BCP 14](#), [RFC 2119](#), March 1997.
- [RFC6120] Saint-Andre, P., "Extensible Messaging and Presence Protocol (XMPP): Core", [RFC 6120](#), March 2011.
- [RFC6887] Wing, D., Cheshire, S., Boucadair, M., Penno, R., and P. Selkirk, "Port Control Protocol (PCP)", [RFC 6887](#), April 2013.

[10.2.](#) Informative References

- [I-D.boucadair-connectivity-provisioning-protocol]
Boucadair, M. and C. Jacquenet, "Connectivity Provisioning Negotiation Protocol (CPNP)", [draft-boucadair-connectivity-provisioning-protocol-00](#) (work in progress), May 2013.
- [I-D.ietf-pcp-authentication]
Wasserman, M., Hartman, S., and D. Zhang, "Port Control Protocol (PCP) Authentication Mechanism", [draft-ietf-pcp-authentication-01](#) (work in progress), October 2012.
- [I-D.sin-sdnrg-sdn-approach]
Boucadair, M. and C. Jacquenet, "Software-Defined Networking: A Service Provider's Perspective", [draft-sin-sdnrg-sdn-approach-03](#) (work in progress), June 2013.
- [OpenFlow]
OpenFlow, ., "OpenFlow Switch Specification", February 2011, <<http://www.openflow.org/documents/openflow->

[spec-v1.1.0.pdf](#)>.

- [RFC2753] Yavatkar, R., Pendarakis, D., and R. Guerin, "A Framework for Policy-based Admission Control", [RFC 2753](#), January 2000.

Penno, et al.

Expires April 02, 2014

[Page 10]

Internet-Draft

A-SDN

September 2013

- [RFC3084] Chan, K., Seligson, J., Durham, D., Gai, S., McCloghrie, K., Herzog, S., Reichmeyer, F., Yavatkar, R., and A. Smith, "COPS Usage for Policy Provisioning (COPS-PR)", [RFC 3084](#), March 2001.
- [RFC4594] Babiarez, J., Chan, K., and F. Baker, "Configuration Guidelines for DiffServ Service Classes", [RFC 4594](#), August 2006.
- [RFC6241] Enns, R., Bjorklund, M., Schoenwaelder, J., and A. Bierman, "Network Configuration Protocol (NETCONF)", [RFC 6241](#), June 2011.
- [RFC6459] Korhonen, J., Soininen, J., Patil, B., Savolainen, T., Bajko, G., and K. Iisakkila, "IPv6 in 3rd Generation Partnership Project (3GPP) Evolved Packet System (EPS)", [RFC 6459](#), January 2012.

Authors' Addresses

Reinaldo Penno
Cisco Systems, Inc.
170 West Tasman Drive
San Jose 95134
USA

Email: repenno@cisco.com

Tirumaleswar Reddy
Cisco Systems, Inc.
Cessna Business Park, Varthur Hobli

Sarjapur Marathalli Outer Ring Road
Bangalore, Karnataka 560103
India

Email: tiredddy@cisco.com

Mohamed Boucadair
France Telecom
Rennes 35000
France

Email: mohamed.boucadair@orange.com

Penno, et al.

Expires April 02, 2014

[Page 11]

Internet-Draft

A-SDN

September 2013

Dan Wing
Cisco Systems, Inc.
170 West Tasman Drive
San Jose, California 95134
USA

Email: dwing@cisco.com

Suresh Vinapamula
Juniper Networks, Inc.
1194 N Mathilda Ave
Sunnyvale, California 94089
USA

Email: sureshk@juniper.net

