Internet Engineering Task Force
Internet-Draft
Intended status: Informational
Expires: September 14, 2017

C. Li
China Telecom
Y. Cheng
China Unicom
T. Peng
X. Song
J. Strassner
Huawei Technologies
March 13, 2017

Internet Classfication
draft-li-intent-classification-00

Abstract

Intent is an abstract high-level policy used to operate the network RFC 7575 [RFC7575]. Intent management system includes an interface for users to input requests and an engine to manage the requests. Up to now, there is no commonly agreed interface or model of intent. This document describes different ways to classify intent, and an associated taxonomy of this classification.

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 September 14, 2017.

Table of Contents

1. Introduction

Different SDOs (such as [ANIMA][ONF]) have proposed intent as a declarative interface for defining a set of network operations to execute. Although there is no common definition or model of intent which are agreed by all SDOs, there are several shared principles:

- intent should be declarative, using and depending on as few deployment details as possible
- intent should provide an easy-to-use interface, and use terminology and concepts familiar to its target audience
- intent should be vendor-independent and portable across platforms
- the intent framework should be able to detect and resolve conflicts between multiple intents

SDOs have different perspectives on what intent is, what set of actors it is intended to serve, and how it should be used. This document provides several dimensions to classify intents.

2. 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].

3. Acronyms

CLI: Command Line Interface

SDO: Standards Development Organisation

SUPA: Simplified Use of Policy Abstractions

VPN: Virtual Private Network

4. The Policy Continuum

The Policy Continuum defines the set of actors that will create, read, use, and manage policy. Each set of actors has their own terminology and concepts that they are familiar with. This captures the fact that business people do not want to use CLI, and network operations center personnel do not want to use non-technical languages.

5. Functional Characteristics and Behavior

Intent can be used to operate immediately on a target (much like issuing a command), or whenever it is appropriate (e.g., in response to an event). In either case, intent has a number of behaviors that serve to further organize its purpose, as described by the following subsections.

5.1. Persistence

Intents can be classified into transient/persistent intents.

If intent is transient, it has no lifecycle management. As soon as the specified operation is successfully carried out, the intent is finished, and can no longer affect the target object.

If the intent is persistent, it has lifecycle management. Once the intent is successfully activated and deployed, the system will keep all relevant intents active until they are deactivated or removed.

5.2. Abstracting Intent Operation

The modeling of Policies can be abstracting using the following three-tuple:

{Context, Capabilities, Constraints}

Context grounds the policy, and determines if it is relevant or not for the current situation. Capabilities describe the functionality that the policy can perform. Capabilities take different forms, depending on the expressivity of the policy as well as the programming paradigm(s) used. Constraints define any restrictions on the capabilities to be used for that particular context. Metadata can be optionally attached to each of the elements of the three-tuple, and may be used to describe how the policy should be used and how it operates, as well as prescribe any operational dependencies that must be taken into account. Put another way:

- Context selects policies based on applicability
- Capabilities describe the functionality provided by the policy
- Constraints restrict the capabilities offered and/or the behavior of the policy

Hence, the difference between imperative, declarative, and other types of policies lies in how the elements of this three-tuple are used according to that particular programming paradigm. This is how [SUPA] was designed: a Policy is a container that aggregates a set of statements .

5.3. Policy Subjects and Policy Targets

Policy subject is the actor that performs the action specified in the policy. It can be the intent management system which executes the policy. Policy target is a set of managed objects which may be affected in the policy enforcement.

5.4. Policy Scope

Policies used to manage the behavior of objects that they are applied to (e.g., the target of the policy). It is useful to differentiate between the following categories of targets:

- Policies defined for the Customer or End-User
- Policies defined for the management system to act on objects in the domain that the management system controls

- Policies defined for the management system to act on objects in one or more domains that the management system does not directly control

The different origins and views of these three categories of actors lead to the following important differences:

- Network Knowledge. This area is explored using three exemplary actors that have different knowledge of the network.

Customers and end-users do not necessarily know the functional and operational details of the network that they are using. Furthermore, most of the actors in this category lack skills to understand such details; in fact, such knowledge is typically not relevant to their job. In addition, the network may not expose these details to its users. This class of actor focuses on the applications that they run, and uses services offered by the network. Hence, they want to specify policies that provide consistent behavior according to their business needs. They do not have to worry about how the policies are deployed onto the underlying network, and especially, whether the policies need to be translated to different forms to enable network elements to understand them.

Application developers work in a set of abstractions defined by their application and programming environment(s). For example, many application developers think in terms of objects (for example, a VPN). While this makes sense to the application developer, most network devices do not have a VPN object per se; rather, the VPN is formed through a set of configuration statements for that device in concert with configuration statements for the other devices that together make up the VPN. Hence, the view of application developers matches the services provided by the network, but may not directly correspond to other views of other actors.

Management personnel, such as network Administrators, have complete knowledge of the underlying network. However, they may not understand the details of the applications and services of Customers and End-Users.

- Automation. In theory, intents from both end-user and management system can be automated. In practice, most intents from end-user are created manually according to business request. End-users do not create or alter intents unless there is change in business. Intents from management systems can be created or altered to reflect with network policy change. For example, end-users create intents to set up paths between hosts, while the management system creates an intent to set a global link utilization limit.

6. Acknowledgements

The authors would like to thank Will (Shucheng) Liu for his comments to this document.

7. IANA Considerations

This document includes no request to IANA.

8. Security Considerations

This document does not have any Security Considerations.

9. References

9.1. Normative References

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, DOI 10.17487/RFC2119, March 1997, <http://www.rfc-editor.org/info/rfc2119>.

[RFC7575] Behringer, M., Pritikin, M., Bjarnason, S., Clemm, A., Carpenter, B., Jiang, S., and L. Ciavaglia, "Autonomic Networking: Definitions and Design Goals", RFC 7575, DOI 10.17487/RFC7575, June 2015, <http://www.rfc-editor.org/info/rfc7575>.

[SUPA] Strassner, J., "Simplified Use of Policy Abstractions", 2017, <https://datatracker.ietf.org/doc/draft-ietf-supa-generic-policy-info-model/?include_text=1>.

9.2. Informative References

[ANIMA] Du, Z., "ANIMA Intent Policy and Format", 2017, <https://datatracker.ietf.org/doc/draft-du-anima-an-intent/>.

[ONF] ONF, "Intent Definition Principles", 2017, <https://www.opennetworking.org/images/stories/downloads/sdn-resources/technical-reports/TR-523_Intent_Definition_Principles.pdf>.

[ONOS] ONOS, "ONOS Intent Framework", 2017, <https://wiki.onosproject.org/display/ONOS/Intent+Framework/>.

[RFC3198] Westerinen, A., Schnizlein, J., Strassner, J., Scherling, M., Quinn, B., Herzog, S., Huynh, A., Carlson, M., Perry, J., and S. Waldbusser, "Terminology for Policy-Based Management", RFC 3198, DOI 10.17487/RFC3198, November 2001, <http://www.rfc-editor.org/info/rfc3198>.

Authors' Addresses

Chen Li
China Telecom
No.118 Xizhimennei street, Xicheng District
Beijing 100035
P.R. China

Email: lichen.bri@chinatelecom.cn

Ying Cheng
China Unicom
No.21 Financial Street, XiCheng District
Beijing 100033
P.R. China

Email: chengying10@chinaunicom.cn

Tao Peng
Huawei Technologies
Bantian
Shenzhen, Longgang District 518129
P.R. China

Email: dr.pengtao@huawei.com

Xiaolin Song
Huawei Technologies
Bantian
Shenzhen 518129
P.R. China

Email: sxlin@huawei.com

John Strassner
Huawei Technologies
2330 Central Expressway
Santa Clara 95138
P.R. China

Email: john.sc.strassner@huawei.com