Internet Research Task Force
Internet-Draft
Intended status: Informational
Expires: May 1, 2018

K. Sivakumar
M. Chandramouli
Cisco Systems, Inc.

October 28, 2017

Concepts of Network Intent
draft-moulchan-nmrg-network-intent-concepts-00

Abstract

This document presents an overview of the concepts of Network Intent and provides definitions for some of the nomenclature. Some potential use cases are presented.

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 https://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 May 1, 2018.

Table of Contents

1. Introduction

Recently, there have been deployments of networks of Service Provider, enterprise and data centres in a very large scale. From a network management perspective, the manageability of networks of such scale poses new challenges. The increasing complexity of network configuration is an additional challenge for the network administrators. To an extent, for device-level configurations, there has been standardization efforts underway in technologies such as YANG [RFC6020], and NETCONF [RFC6241]. However, the challenge still remains at the network level configuration, orchestration and management. The complexity of the network can lead to potential mis-configurations and furthermore, it may be difficult to troubleshoot the network failure conditions.

From a management perspective, it is of paramount importance for the network administrator to reduce the complexity of the network management. There are several measures and approaches that have been under consideration towards that objective. One aspect that has gained attention is Network Programmability APIs in the management plane. Programmability allows the capabilities of network functionality to be modified or extended. Programmability promises to enable the development of a whole new wave of applications that provide additional management intelligence. Programmability enables the development of applications whose purpose is to make the networks easier to manage, and those applications can be embedded and tightly coupled with the network. The application developers can use the Network Programmability APIs that can allow them to add new features that can facilitate ease of network management, efficiency and the

effectiveness with which the network can be provisioned, administrated and managed. Programmability, as provided through SDN, provides exciting new opportunities to increase manageability by facilitating the development of corresponding applications. Software defined networking (SDN) is an umbrella term for a programmatic approach to managing network devices, using software controls to replace manual configuration. Initial motivations for SDN were to overcome the the lack of network programmability, and manageability in networks.

SDN technologies allow network-wide visibility and the possibility of feedback actions across the network. The desire to implement higher layers of management abstraction such as policy-based management, or the desire to extend an application's capabilities with application-specific pre-processing that can be delegated to the network.

Leveraging the Network Programmability APIs opens the possibility to introduce an abstraction for the network, which can be used to synthesise the overall system behaviour. In the networking parlance, there have been several concepts that have been have been considered to simplify the network management - Network Policy, Autonomic Networking, Service Models, and Network Configuration. We introduce the concept of Intent Based Networking, by which the network administrator can articulate a desired outcome to the network. The Network Intent is translated to appropriate network policies and/or network configurations. With this approach to Network Intent, the focus is more on "what" the network should do and less on "how" i.e., the intermediate steps that should be executed. This level of abstraction can be referred to as "Network Intent". The implicit assumption is that for "Network Intent" there might be some prerequisite steps that may need to be performed, such as the network elements are discovered and controlled, and device capabilities and features are identified.

While there has been investigations of Network Intent, there are some still ambiguities in terms of the terminology used. This initial proposal is an attempt to clarify some of the terms and provides a brief outline of the goals or the vision intended. Some use cases are presented to illustrate the concepts introduced in this document.

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. Hierarchy of Manageability

There is a certain non-physical, logical hierarchy in a network management environment, as described in the figure below. The user at the top of the hierarchy can be represented by a "real" user or a system that performs actions on behalf of a user, such as a management station.

The "user" establishes an "intent" to be taken on the network as a whole and pushes that intent to the second layer of the management hierarchy, which consists of the intent engine.

The next layer of the hierarchy consumes the "intent" and translates the intent to desired actions based on the meaning of the intent.

The bottom layer of the hierarchy consists of the devices on the network that consume the configurations and actions issued to them by the intent engine. These devices sit directly on the network and are responsible for traffic flowing through the network.

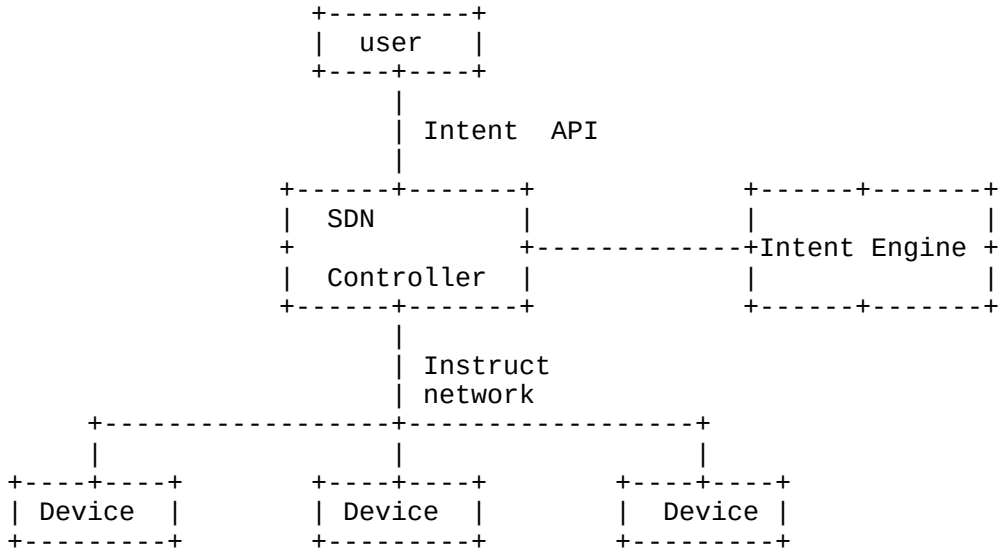

Figure 1

4. Network Configuration

Network configurations are the most basic atomic operations that can be performed on a network device. A particular feature of the network software can be enabled by one or many lines of network

configurations. Often the network devices are configured by experts with _domain expertise_ and based on the functionality the network device has to perform. Often, network configuration is performed on a device by device basis and this is a manual process. Automation of this process is very important step, which can save time and reduce the possible number of mis-configurations.

5. Network Policy

Policy based network management has been widely discussed in the literature [JNSM]. Several proposals for the semantics and structure for expressing network policy have been considered. There are some particular implementations and deployments of network policies such as Performance Forwarding, QoS profiles etc.

A network policy can be viewed as a set of rules a network administer can use to manage the network resources; for example to provide differential treatment for traffic. Policies can be at a network level and can provide a way of consistently managing multiple network devices. The administrator can define policies and specify how the network devices should deal with different types of traffic. Policies can be defined to be conditional, in the sense, if there is a condition A is observed, then a set a network policy can be implemented on some network devices. Policies can be a group of network configurations which perform a specific function that can be applied to network devices. In the SDN paradigm, network policies can be pushed to the network devices using NETCONF [RFC6020] and RESTCONF [RFC8040].

6. Network Intent

Network Intent can be considered as a declarative paradigm by which the network administrator articulates a desited outcome or the state of the network. In abstraction, the network enables a set of services that can be consumed. In particular, Network Intent is a desirable functionality that can be enabled from an SDN Controller. There are potential benefits of ease-of-use and operational simplicity and the capability of programming the entire network.

Network Intent need not be prescriptive or expressed explicitly in terms of specific actions. The following are the intended design considerations of network intent.

- First, there may be several alternative approaches to realise a specific Network Intent.

- Second, it is conceivable that it may not be possible to realise some of the Network Intents due to non-availability of network resources or the network may not have functionality.

- Third, some new Network Intent can be in conflict with the current state of the network or can disrupt the Network Intents expressed previously. It is assumed such a feedback regarding the conflicts is provided back to the administrator the originator of the Network Intent. Based on the feedback, it should be possible for the network administrator to refine the new Network Intent.

This proposal or definition of Network Intent can be viewed as analogous to the promise theory framework proposed [Promise]. In order to realise the Network Intent, it may be useful consider a logical functional block - the Intent Engine - that can resolve the network intent and render the Network Intent appropriately on to the network.

The simplistic method to realise network intent is to consider linear one-to-one mapping of Network Intents to actual network policies or network configurations. In a more general framework of Network Intent, it should be possible to consider a more general approach leveraging artificial intelligence based techniques so that the Network Intent can be accurately realised and appropriately rendered on the network. Translating the intent requests to rendering actions would require the modelling of network devices and the functionalities and configurations.

In order to realise a Network Intent eventually that should consist of network configuration blocks that can be implemented in one or more network devices.

There is a general confusion between policy based network management and intent based network management. An analogy can be drawn between intent based network management and the automotive industry. Though cliched, this analogy provides the closest match. Many cars, if not all, have cruise control as a function today. Cruise control is a very simplistic functionality that keeps the car going at a specific speed. It monitors the speed and adjusts it up or down. This can be considered as a policy, to keep the car driving at certain speed, until the operator disengages the policy manually.

An car that can handle intent would, on the other hand, accept a request such as "take me from San Francisco to Los Angeles within 6 hours," plot the appropriate path based on historical data on which roads are the best ones to take to achieve the constraint of reaching within 6 hours and plots the direction to go in. Then it would constantly monitor the traffic on the path and provide feedback to

the operator about whether the path chosen will still achieve the constraint. If the constraint cannot be achieved, then it either re-plots the path or lets the operator know that the constraint cannot be achieved and requests a new constraint. The operator is removed from making the decision about which exact path to take and is instead just providing the constraints that need to be achieved.

7. Use Cases

This section lists certain use cases that showcase the value of intent based network management. There are a variety of use cases where intent based network management is of value but the highest value is present in scenarios where a network needs to be reprogrammed in a significant manner in the shortest of time frames. Such a network reconfiguration should not result in misconfiguration that could result in the loss of communication capabilities for the users of the network.

We provide two scenarios where such a reconfiguration of the network is required. There are obviously many more day-to-day scenarios where the intent of change or monitoring of a network can be of a much lower scale.

7.1. A simple example

The network administrator articulates the Network Intent, "Route traffic from Node A to Node B with minimum bandwidth of K mbps". The Intent Engine then resolves the intent. This step involves understanding the intent expressed and the second step to resolving that intent would require performing routing calculations between Node A and Node B. This is a key step involved in this proposal.

Once the intent has been resolved, routing calculations are well-known and there are standard techniques taking into account the network topology between Node A and Node B; the current utilisations with minimum guaranteed bandwidth of K Mbps between Node A and Node B. Once the path is determined, that routing and next hop configurations are communicated to the respective network nodes.

7.2. Disaster Management

Planning for disaster management and sudden reconfiguration of infrastructure is common in the "physical" world - ie roads, water supply, electricity, etc. Similar reconfigurations of communication networks also is important during a disaster. During a disaster management / recovery, it is important to ensure that emergency communication traffic (such as 911 in the USA, 999 in UK and similar in other countries) gets more bandwidth and resources than non-

emergency communication. It is also important to allow people to communicate with their family members inside and outside the disaster area, to help in recovery efforts. For this reason, voice communication, including VoIP, should be prioritized over streaming video services.

Such a disaster management is geographically bounded, therefore the network changes need to also be appropriately geographically bounded. This is very often hard to apply manually in a very large network at the moment that the change is needed. Intent based networks can provide an abstraction that use the underlying knowledge of the network and policies to achieve an action to provide this ability in a finer grained manner.

As the disaster scenario subsides the applied intent should automatically subside as well. This requires not only action to be taken based on policies, but also requires constant monitoring of the operational state network. Such monitoring presents significant amounts of data and it is quite hard to build rules and conditions to operate on such data while minimizing mistakes. Machine learning based monitoring can provide a mechanism to make applying an intent easier, especially in very large networks. Such machine learning based mechanisms can be integrated with physical world monitoring to identify when a disaster hits a certain geography and to automatically trigger a pre-set intent for that scenario. With such machine learning mechanisms and multiple pre-set intents, it would be possible for a management system to automatically trigger a specific intent when it detects a particular scenario. Similar combination of operational monitoring and intent based networking mechanism can be used to withdraw an intent when the disaster like scenario recedes.

8. Issues with Intent based networking

Intent based network management is about creating an abstraction to handle the management of a network. Naturally issues related to any abstraction mechanism applies here as well. Specifically, an abstraction like this removes the direct interaction of a user with the network for operations management. While the original creators of this intent, and the associated policies, would have understood the reasoning behind this intent, and more importantly the fine distinction between when to apply and when NOT to apply such an intent, later users of the system may not have that clear distinction and may apply this intent needlessly. This problem exists in any abstraction mechanism.

9. Security Considerations

This draft currently does not impose any security considerations.

10. IANA Considerations

This memo has no actions for IANA.

11. References

11.1. 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, <https://www.rfc-editor.org/info/rfc2119>.

[RFC6020] Bjorklund, M., Ed., "YANG - A Data Modeling Language for the Network Configuration Protocol (NETCONF)", RFC 6020, DOI 10.17487/RFC6020, October 2010, <https://www.rfc-editor.org/info/rfc6020>.

[RFC6241] Enns, R., Ed., Bjorklund, M., Ed., Schoenwaelder, J., Ed., and A. Bierman, Ed., "Network Configuration Protocol (NETCONF)", RFC 6241, DOI 10.17487/RFC6241, June 2011, <https://www.rfc-editor.org/info/rfc6241>.

[RFC8040] Bierman, A., Bjorklund, M., and K. Watsen, "RESTCONF Protocol", RFC 8040, DOI 10.17487/RFC8040, January 2017, <https://www.rfc-editor.org/info/rfc8040>.

11.2. Informative References

[JNSM] Boutaba, R. and I. Aib, "Policy-Based Management: A Historical perspective, Journal of Network and Systems Management 15 (4), 447-480", 2007.

[Promise] Borril, P., Burgess, M., Craw, T., and M. Dvorkin, "A Promise Theory Perspective on Data Networks, CoRR, abs/1405.2627", September 2014.

Authors' Addresses

Kaarthik Sivakumar
Cisco Systems, Inc.
Sarjapur Outer Ring Road
Bangalore 560103
India

Phone: +91 80 4429 2264
Email: kasivaku@cisco.com
URI: http://www.cisco.com/

Mouli Chandramouli
Cisco Systems, Inc.
Sarjapur Outer Ring Road
Bangalore 560103
India

Phone: +91 80 4429 2409
Email: moulchan@cisco.com
URI: http://www.cisco.com/