SUPA   J. Bi
Network Working Group   Tsinghua University
Internet Draft   G. Karagiannis
Intended status: Informational   J. Strassner
Expires: October 7, 2016   Huawei Technologies
                                                          M. Klyus
                                                        NetCracker
                                                           Q. Sun
                                                     China Telecom
                                                 Luis M. Contreras
                                                         Telefonica
                                                     April 7, 2016

        Problem Statement for Simplified Use of Policy Abstractions (SUPA)
                    draft-bi-supa-problem-statement-00

Abstract

   Simplified Use of Policy Abstractions (SUPA) defines a set of rules
   that define how services are designed, delivered, and operated
   within an operator's environment independent of any one particular
   service or networking device. SUPA expresses policy rules using a
   generic policy information model, which serves as a unifying
   influence to enable different data model implementations to be
   simultaneously developed.

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), its areas, and its working groups. Note that
   other groups may also distribute working documents as Internet-
   Drafts.

   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 October 5, 2016.

Table of Contents

1. Introduction

The rapid growth in the variety and importance of traffic flowing over increasingly complex enterprise and service provider network architectures makes the task of network operations and management applications and deploying new services much more difficult. In addition, network operators want to deploy new services quickly and efficiently. Two possible mechanisms for dealing with this growing difficulty are the use of software abstractions to simplify the design and configuration of monitoring and control operations and the use of programmatic control over the configuration and operation of such networks. Policy-based management can be used to combine these two mechanisms into an extensible framework.

Policy rules can be used to express high-level network operator requirements directly, or from a set of management applications, to a network management or element system. The network management or element system can then control the configuration and/or monitoring of network elements and services.

Simplified Use of Policy Abstractions (SUPA) will define a generic policy information model (GPIM) [SUPA-info-model] for use in network operations and management applications. The GPIM defines concepts and terminology needed by policy management indepednent of the form and content of the policy rule. The ECA Policy Rule Information Model (EPRIM) [SUPA-info-model] extends the GPIM to define how to build policy rules according to the event-condition-action paradigm.

Both the GPIM and the EPRIM are targeted at controlling the configuration and monitoring of network elements throughout the service development and deployment lifecycle. The GPIM and the EPRIM will both be translated into corresponding YANG [RFC6020] modules that define policy concepts, terminology, and rules in a generic and interoperable manner; additional YANG modules may also be defined from the GPIM and/or EPRIM to manage specific functions.

The key benefit of policy management is that it enables different network elements and services to be instructed to behave the same way, even if they are programmed differently. Management applications will benefit from using policy rules that enable scalable and consistent programmatic control over the configuration and monitoring of network elements and services.

1.1. Problem Statement

Network operators must construct networks of increasing size and complexity in order to improve their availability and

quality, as more and more business services depend on them.

Currently, different technologies and network elements require
different forms of the same policy that governs the production of

network configuration snippets. The power of policy management is its applicability to many different types of systems, services, and networking devices. This provides significant improvements in configuration agility, error detection, and uptime for operators.

Many different types of actors can be identified that can use a policy management system, including applications, end-users, developers, network administrators, and operators. Each of these actors typically has different skills and uses different concepts and terminologies. For example, an operator may want to express that only Platinum and Gold users can use streaming and interactive multimedia applications. As a second example, an operator may want to define a more concrete policy rule that looks at the number of dropped packets. If, for example, this number exceeds a certain threshold value, then the applied queuing, dropping and scheduling algorithms could be changed in order to reduce the number of dropped packets. The power of SUPA is that both of these examples may be abstracted. For example, in the latter example, different thresholds and algorithms could be defined for different classes of service.

1.2. Proposed Solution

SUPA enables network operators to express policies to control network configuration and monitoring data models in a generic manner. The configuration and monitoring processes are independent of device, as well as domain or type of application, and result in configuration according to YANG data models.

Both of the examples in section 1.1 can be referred to as "policy rules", but they take very different forms, since they are defined at different levels of abstraction and likely authored by different actors. The first example described a very abstract policy rule, and did not contain any technology-specific terms, while the second example included more concrete policy rules and likely used technical terms of a general (e.g., IP address range and port numbers) as well as vendor-specific nature (e.g., specific algorithms implemented in a particular device). Furthermore, these two policy rules could affect each other. For example, Gold and Platinum users might need different device configurations to give the proper QoS markings to their streaming multimedia traffic. This is very difficult to do if a common policy framework does not exist.

Note that SUPA is not limited to any one type of technology. While the above two policies could be considered "QoS" policies, other examples include:

  - network elements must not accept passwords for logins

- all SNMP agents in this network must drop all SNMP traffic
          unless it is originating from, or targeting, the
          management network

      - Periodically perform workload consolidation if average CPU
        utilization falls below X%

   The above three examples are not QoS related; this emphasizes the
   utility of the SUPA approach in being able to provide policies
   to control different types of network element configuration and/or
   monitoring snippets.

   There are many types of policies. SUPA differentiates between
   "management policies" and "embedded policies". Management
   policies are used to control the configuration of network
   elements. Management policies can be interpreted externally to
   network elements, and the interpretation typically results in
   configuration changes of collections of network elements. In
   contrast, "embedded policies" are policies that are embedded
   in the configuration of network elements, and are usually
   interpreted on network elements in isolation. Since embedded
   policies are interpreted in the network device, they are
   typically composed in a very specific fashion to run at
   near-realtime timescales.

1.3. Value of the SUPA Approach

   SUPA will achieve an optimization and reduction in the amount of
   work required to define and implement policy-based data models in
   the IETF. This is due to the generic and extensible framework
   provided by SUPA.

   SUPA defines policy independent of where it is located. Other
   WGs are working on embedding policy in the configuration of a
   network element; SUPA is working on defining policies that
   can be interpreted external to network elements (i.e., management
   policies). Hence, SUPA policies can be used to define the behavior
   of and interaction between embedded policies.

   Since the GPIM defines common policy terminology and concepts, it
   can be used to both define more specific policies as part of a
   data model as well as derive a (more abstract) information model
   from a (more specific) data model.

   This latter approach may be of use in discovering common structures
   that occur in data models that have been designed in isolation of
   each other.

   The SUPA policy framework defines a set of consistent, flexible,
   and scalable mechanisms for monitoring and controlling resources
   and services. It may be used to create a management and operations
   interface that can enable existing IETF data models, such as those
   from I2RS and L3SM, to be managed in a unified way that is

independent of application domain, technology and vendor. Resource
and service management become more effective, because policy

   defines the context that different operations, such as configuration
   and monitoring, are applied to.

[2]. **Application of Generic Policy-based Management**

   This section provides examples of how SUPA can be used to define
   different types of policies. Examples applied to various domains,
   including system management, operations management, access control,
   routing, and service function chaining, are also included.

[2.1]. **ECA Examples**

   ECA policies are rules that consist of an event clause, a condition
   clause, and an action clause.

      Network Service Management Example

         Event: too many interface alarms received from an
                    L3VPN service
         Condition: alarms resolve to the same interface within a
                    specified time period
         Action: if error rate exceeds x% then put L3VPN service
                    to Error State and migrate users to one or more
                    new L3VPNs

      Security Management Example

         Event: anomalous traffic detected in network
         Condition: determine the severity of the traffic
         Action: apply one or more actions to affected NEs based
                    on the type of the traffic detected (along with
                    other factors, such as the type of resource
                    being attacked if the traffic is determined to
                    be an attack)

      Traffic Management Examples

         Event: edge link close to being overloaded by
                    incoming traffic
         Condition: if link utilization exceeds Y% or if link
                    utilization average is increasing over a
                    specified time period
         Action: change routing configuration to other peers
                    that have better metrics

         Event: edge link close to be overloaded by
                    outgoing traffic
         Condition: if link utilization exceeds Z% or if link

utilization average is increasing over a
                    specified time period
         Action: reconfigure affected nodes to use source-based
                    routing to balance traffic across multiple links

     Service Management Examples

          Event: alarm received or periodic time period check
          Condition: CPU utilization level comparison
          Action: no violation: no action
                     violation:
                       1) determine workload profile in time interval
                       2) determine complementary workloads (e.g.,
                          whose peaks are at different times in day)
                       3) combine workloads (e.g., using integer
                          programming)

          Event: alarm received or periodic time check
          Condition: if DSCP == AFxy and
                     throughput < T% or packet loss > P%
          Action: no: no action
                     yes: remark to AFx'y'; reconfigure queuing;
                     configure shaping to S pps; ...

   Note: it is possible to construct an ECA policy rule that is
   directly tied to configuration parameters.

3. Related Work

3.1. Related Work within the IETF

3.1.1. I2RS Working Group

   I2RS defines an interface that interacts with the routing
   system using a collection of protocol-based control or
   management interfaces. Users of I2RS interfaces are typically
   management applications and controllers. SUPA does not directly
   interface to the routing system. Rather, SUPA uses data
   produced by I2RS (e.g., topological information) to construct
   its policies.

3.1.2. L3SM Working Group

   L3SM defines an L3 VPN service model that can be used for
   communication between customers and network operators. This
   model enables an orchestration application or customers to
   request network services provided by L3 VPN technologies. The
   implementation of network services is often guided by specific
   policies, and SUPA provides a tool that can help with the
   mapping of L3 VPN service requests to L3 VPN configurations of
   network elements.

3.1.3. ALTO Working Group

The ALTO working group defined an architecture for exposing
topology information, more specifically the cost of paths
through an infrastructure, as defined in [RFC7285]. ALTO
services are able to provide network maps defined as groups of
endpoints, and can therefore represent any granularity of network,
from the physical to groups of networks following similar paths or
restraints. Although this model can represent different levels of
granularities, it is not clear if it could be adapted easily for
other purposes than providing cost maps in the context of ALTO.
The ALTO model is meant to be used outside of the trust domain of
an ISP by external clients.

SUPA does not generate data that is similar to ALTO. Rather,
SUPA could use ALTO data as part of its policies to configure
services and/or resources.

3.1.4. TEAS Working Group

The Traffic Engineering Architecture and Signaling (TEAS)
working group is responsible for defining MPLS- and GMPLS-based
Traffic Engineering architectures that enable operators to
control how specific traffic flows are treated within their
networks. It covers YANG models for a traffic engineering
database. In coordination with other working groups (I2RS)
providing YANG models for network topologies.

Both TEAS and SUPA use YANG data models. SUPA does not generate
traffic engineering (TE) data. However, SUPA could use TE data
as part of its policies for configuring resources and/or
services. SUPA could also define policies that define which
service, path, and link properties to use for a given customer,
and consequently, which protocol extensions to use. TEAS data
could also be used to enable operators to define how particular
traffic flows are treated in a more abstract (but still
consistent) manner.

3.1.5. BESS Working Group

The BGP Enabled Services (BESS) working group defines and
extends network services that are based on BGP. This includes
BGP/MPLS IP provider-provisioned L3VPNs, L2VPNs, BGP-enabled
VPN solutions for use in data center networking, and extensions
to BGP-enabled solutions to construct virtual topologies in
support of services such as Service Function Chaining. The
working group is also chartered to work on BGP extensions to
YANG models and data models for BGP-enabled services.
Both BESS and SUPA use YANG data models. SUPA could generate

BGP configurations by using data defined by BESS as part of
its policies for configuring resources and/or services.

   SUPA could also define policies that govern different aspects
   of services defined by BESS.

3.1.6. SFC Working Group

   The Service Function Chaining (SFC) working group defines a
   mechanism where traffic is classified; that classification is
   then use to select an ordered set of services to pass the
   traffic through.

   Both SFC and SUPA use YANG data models. SUPA could define
   policies that augment the functionality of SFC in several
   different ways, including: (1) path selection based on context,
   (2) which set of mechanisms to use to steer traffic through
   which set of service functions, (3) simplify the definition of
   dynamic service function chains (e.g., service paths that
   change based upon a set of data that is discovered at runtime),
   and (4) scalable mechanisms to monitor and control the
   configuration of SFC components.

3.1.7. NVO3 Working Group

   The NVO3 group proposes a way to virtualize the network edge
   for data centers in order to be able to move virtual instances
   without impacting their network configuration. This is realized
   through a centrally controlled overlay layer-3 network. The
   NVO3 work is not about defining policy information; rather, it
   uses policy information to perform some functions. Both NVO3 and
   SUPA use YANG data models. SUPA could define policies that define
   how the logically centralized network virtualization management
   entity (or entities) of NVO3 behave (e.g., the functions in the
   network virtualization control plane).

3.1.8. ACTN BoF (IETF-90)

   The ACTN proposed work, as described in [actn] framework, has
   two main goals, the abstraction of multiple optical transport
   domains into a single controller offering a common abstract
   topology, and the splitting of that topology into abstract
   client views that are usually a fraction of the complete
   network. The ACTN work is therefore about unification of
   several physical controllers into a virtual one, and also about
   the segmentation, isolation and sharing of network resources.
   The ACTN work is not about defining policy information. Both ACTN
   and SUPA use YANG data models. SUPA could define policies that
   define the behavior of the controller.

3.1.9. Previous IETF Policy Models

SUPA is technology-neutral, previous RFCs weren't. SUPA defines a
common structure from which ECA policies can be defined; this was
not possible in previous RFCs. Previous RFCs do NOT define metadata,

and do NOT enable policies to formally define obligation,
permission, and related concepts. Finally, SUPA uses software
patterns, which previous RFCs didn't.

A more complete analysis is in [Appendix A](#) of [[SUPA-info-model](#)].

[3.2](#). Related Work outside the IETF

[3.2.1](#). TM Forum

The TM Forum (a.k.a., the TeleManagement Forum) develops standards
and best practices, research, and collaborative programs focused on
digital business transformation. It consists of three major
programs:

    1) Agile Business and IT
    2) Customer Centricity (experience)
    3) Open Digital Ecosystem

Of these, the ZOOM (Zero-touch Orchestration, Operations, and
Management) project, located in the Agile Business and IT project,
is the main sub-project in this area that is of interest to SUPA.

Within ZOOM, the Foundational Studies project contains work on
an information model and management architecture that are
directly relevant to SUPA. The TMF Information Model, Policy,
and Security working groups are involved in this work.

The ZOOM information model updates the existing Shared
Information and Data (SID) information model to add support for
the management of physical and virtual infrastructure, event-
and data-driven systems, policy management (architecture and
model), metadata for describing and prescribing behavior that can
support changes at runtime, and access control. The policy
information model defines imperative (ECA), declarative (intent-
based), utility function, and promise policies. The work in
[[SUPA-info-model](#)] is based on, but extends and enhances, the
ZOOM ECA model and provides additional detail not currently
present in ZOOM.

There is currently no plan to use the utility function and
promise policies of ZOOM in SUPA. Finally, it should be noted
that the data model work planned for SUPA is not currently
planned for the ZOOM project.

[3.2.2](#). MEF

The MEF (originally named the Metro Ethernet Forum) develops
architecture, service and management specifications related to

Carrier Ethernet (CE). The CE architecture includes the definition
of several interfaces specific to CE like the User Network Interface
(UNI) and External Network Network Interface (ENNI). Specifications

developed in this space include the definitions of CE services, CE
service attributes, Ethernet Access Services, Class of Service, OAM
and Management interfaces, Service Activation and Test. The more
recent vision of the MEF is described as The Third Network, and
includes plans to develop Lifecycle Service Orchestration with APIs
for existing network, NFV, and SDN implementations enabling Agile,
Assured, and Orchestrated Services. This stage of the MEF activity
is now in early phases with focus on architectural work.

The MEF has developed a number of Information and Data Models, and
has recently started a project that uses YANG to model and manage
the services defined by the MEF. While the MEF has created rigorous
definitions of these services, they are specific to transport
technology, and they do not currently include and rely on policies.

3.2.3. Open Daylight

Open Daylight network controller implements a number of models
through its service abstraction Layer (MD-SAL) based on draft
IETF Yang models. Open Daylight is an open source project. Two of
these could be relevant to SUPA in the future (since they both are
focused on declarative policies, which are currently out of scope
for SUPA) and are described below.

3.2.3.1. Network Intent Composition (NIC)

The Network Intent Composition project aims at providing better
flexibility by using declarative policies. It does not cover
other types of policies, such as ECA policy rules. The intent-
based interface aims to provide a high level of abstraction,
primarily for use by an application developer. Its progress
has recently stalled.

3.2.3.2. Group Based Policy

The Group Based Policy project defines an application-centric
policy model for Open Daylight that separates information about
application connectivity requirements from information about
the underlying details of the network infrastructure. The model
is positioned as declarative, but uses a relational approach to
specifying policy. It does not cover other types of policies,
such as ECA policy rules.

3.2.4. Open Networking Foundation

The ONF created a group responsible of defining northbound
interfaces, but this hasn't lead to the publication of
standards in this area so far. A blog entry on the ONF web site
showed an interest in using the principle of intents at ONF,

but no details were provided on the status of this project. A
members-only whitepaper was recently published.

3.2.5. OpenStack

   OpenStack software controls large pools of compute, storage,
   and networking resources throughout a datacenter, managed
   through a dashboard or via the OpenStack API. OpenStack works
   with popular enterprise and open source technologies making it
   ideal for heterogeneous infrastructure. Few of the below
   mentioned OpenStack projects provides policy abstraction and
   better flexibility to the user.

3.2.5.1. Group-Based Policy

   The Group-Based Policy project for OpenStack Neutron is built
   around entities assembled in Endpoints Groups (EPG) that
   provide or consume Contracts. Such Contracts are hierarchical
   entities containing policy rules. A first version was released
   in January 2015, based on the Juno release. This type of
   approach is more relational than declarative, but could be used
   to describe a large amount of possible scenarios. It has the
   advantage of providing a relatively simple policy model that
   covers a large applicability. From an OpenStack point of view,
   the scope of Group-Based Policies is limited to networking
   within the Neutron module. Note that other types of policies, such
   as ECA policies, are not covered in Group-Based Policy.

3.2.5.2. Congress

   The Congress project within OpenStack provides a way to define
   complex policies using extensions to the Datalog language.
   Datalog is entirely declarative, and its evaluation is based on
   first-order logic with restrictions. This gives it interesting
   properties, such as providing the same result no matter the order
   in which the statements are made. The language allows for the
   definition of types and for active enforcement or verification
   of the policies. However, it does not cover ECA policies.

   There is a significant body of knowledge and experience relating
   to declarative languages and their implementation. Congress
   policies aim at manipulating objects exposed by multiple
   OpenStack modules, and is therefore larger in scope than network
   element policies.

3.2.6. The NEMO Project (not a BoF yet)

   The NEMO project is a research activity aimed at defining a
   simple framework for "intent-based" networking. This project
   concentrates on creating a domain-specific language and associated
   API, not a model or even a rigorous definition of what a policy

rule is. NEMO does not define ECA policies.

The NEMO syntax defines a very simple information model that has
three basic elements for network manipulation: nodes, links, and

flows. A policy rule is NOT defined in this model. Rather, policy
is defined as a command. The NEMO project has been successfully
demonstrated at IETF-91, along with a companion graphical user
interface.

NEMO declarative policies use a flatter, simpler object model with
fewer objects to represent targets of policy. NEMO uses a condition-
action paradigm to execute its declarative policies. In contrast,
SUPA uses a richer class model to represent ECA policies. However,
SUPA has not proposed a language.

3.2.7. The Floodlight Project

The Floodlight is an OpenFlow-enabled SDN controller. It uses
another open source project called Indigo to support OpenFlow
and manage southbound devices. The Indigo agent also supports
an abstraction layer to make it easy to integrate with physical
and virtual switches. It supports configuration of an abstraction
layer so that it can configure OpenFlow in hybrid mode.

3.2.8. The ONOS Project

The ONOS is an SDN controller design for Service Provider networks.
It uses a distributed architecture, and supports abstraction for
both southbound and northbound interfaces. Its modules are managed
as OSGi bundles. It is an open source project. ONOS announced an
"application-intent framework". However, no object model or
language has been defined yet.

4. Conclusions: the Value of SUPA

SUPA can be used to define high-level, possibly network-wide
policies to create interoperable network element configuration
snippets. SUPA expresses policies and associated concepts using a
generic policy information model, and produces generic policy YANG
data modules. SUPA focuses on management policies that control the
configuration of network elements. Management policies can be
interpreted outside of network elements, and the interpretation
typically results in configuration changes to collections of
network elements.

Policies embedded in the configuration of network elements are not
in the scope of SUPA. In contrast to policies targeted by SUPA,
embedded policies are usually interpreted on network elements in
isolation, and often at timescales that require the representation
of embedded policies to be optimized for a specific purpose.

The SUPA information model generalizes common concepts from multiple
technology-specific data models, and makes it reusable.

Conceptually, SUPA can be used to interface and manage existing and
future data models produced by other IETF working groups. In

addition, by defining an object-oriented information model with
metdata, the characteristics and behavior of data models can be
better defined.

5. Security Considerations

Security is a key aspect of any protocol that allows state
installation and extracting detailed configuration states of network
elements. This places additional security requirements on SUPA (e.g.,
authorization, and authentication of network services) that needs
further investigation. Moreover, policy interpretation can lead to
corner cases and side effects that should be carefully examined,
e.g., in case policy rules are conflicting with each other.

6. IANA Considerations

This document has no actions for IANA.

7. Contributors

The following people all contributed to creating this document,
listed in alphabetical order:

    Parviz Yegani, Juniper Networks
    Vikram Choudhary, Huawei Technologies
    Diego Lopez, Telefonica I+D
    Dan Romascanu, Avaya
    J. Schoenwaelder, Jacobs University, Germany
    Will(Shucheng) Liu, Huawei Technologies
    Tina Tsou
    Jean-Francois Tremblay

8. Acknowledgments

This document has benefited from reviews, suggestions, comments
and proposed text provided by the following members: H. Rafiee, J.
Saperia, C. Zhou, Spencer Dawkins, Xing Li, Chongfeng Xie,
Benoit Claise, Ian Farrer, Marc Blancet, Zhen Cao, Hosnieh Rafiee,
Mehmet Ersue, Simon Perreault, Fernando Gont, Jose Saldana,
Tom Taylor, Kostas Pentikousis, Juergen Schoenwaelder, Eric Voit,
Scott O. Bradner, Marco Liebsch, Scott Cadzow, Marie-Jose Montpetit.

9. References

[9.1](#). Informative References

   [RFC6020] Bjorklund, M., "YANG - A Data Modeling Language for the
   Network Configuration Protocol (NETCONF)", RFC 6020, October 2010.

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

   [SUPA-info-model] J. Strassner, J. Halpern, J. Coleman, "Generic
   Policy Information Model for Simplified Use of Policy Abstractions
   (SUPA)", IETF Internet draft, draft-strassner-supa-generic-policy-
   info-model-05, March 2016

Authors' Addresses

   Georgios Karagiannis
   Huawei Technologies
   Hansaallee 205,
   40549 Dusseldorf,
   Germany
   Email: Georgios.Karagiannis@huawei.com

   Maxim Klyus, Ed.
   NetCracker
   Kozhevnicheskaya str.,7 Bldg. #1
   Moscow, Russia
   E-mail: klyus@netcracker.com

   Qiong Sun
   China Telecom
   No.118 Xizhimennei street, Xicheng District
   Beijing 100035
   P.R. China
   Email: sunqiong@ctbri.com.cn

   Luis M. Contreras
   Telefonica I+D
   Ronda de la Comunicacion, Sur-3 building, 3rd floor
   Madrid 28050
   Spain
   Email: luismiguel.contrerasmurillo@telefonica.com
   URI: http://people.tid.es/LuisM.Contreras/

   John Strassner
   Huawei Technologies
   2330 Central Expressway
   Santa Clara, CA 95138 USA
   Email: john.sc.strassner@huawei.com

Jun Bi
Tsinghua University
Network Research Center, Tsinghua University
Beijing 100084
China
EMail: junbi@tsinghua.edu.cn