SUPA
Network Working Group
Internet Draft
Intended status: Informational
Expires: January 18, 2017

J. Bi
Tsinghua University
G. Karagiannis
J. Strassner
Huawei Technologies
D. Romascanu
Avaya
M. Klyus
NetCracker
Q. Sun
China Telecom
Luis M. Contreras
Telefonica
July 18, 2016

Problem Statement for Simplified Use of Policy Abstractions (SUPA)
draft-bi-supa-problem-statement-02

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

reference material or to cite them other than as "work in progress."

This Internet-Draft will expire on January 18, 2017.

Copyright Notice

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. Terminology

This section lists the terminology used in this document.

Action: a set of purposeful activities that have a set of associated behavior.

Condition: a set of attributes, features, and/or values that are to be compared with a set of known attributes, features, and/or values in order to make a decision. A Condition, when used in the context of a Policy Rule, is used to determine whether or not the set of Actions in that Policy Rul can be executed or not.

Data Model: a data model is a representation of concepts of interest to an environment in a form that is dependent on data repository, data definition language, query language, implementation language, and protocol (typically one or more of these).

ECA: Event - Condition - Action policy.

Event: an Event is defined as any important occurrence in time of a change in the system being managed, and/or in the environment of the system being managed. An Event, when used in the context of a Policy Rule, is used to determine whether the condition clause of an imperative Policy Rule can be evaluated or not.

Information Model: an information model is a representation of concepts of interest to an environment in a form that is independent of data repository, data definition language, query language, implementation language, and protocol.

Metadata: is data that provides descriptive and/or prescriptive information about the object(s) to which it is attached.

Policy Rule: A Policy Rule is a set of rules that are used to manage and control the changing or maintaining of the state of one or more managed objects.

3. 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.

Note that typical use cases and the applicability of SUPA policy models are provided in [SUPA-Applicability].

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.

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
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

[SUPA-info-model] J. Strassner, J. Halpern, "Generic Policy Information Model for Simplified Use of Policy Abstractions (SUPA)", IETF Internet draft, draft-ietf-supa-generic-policy-info-model-00, June 2016

[SUPA-Applicability] N. Vadrevu, D. Zhang, S. Zhu, Y. Cheng, "Applicability of SUPA", IETF Internet draft, draft-vadrevu-supa-applicability-06, 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

Dan Romascanu
Avaya
Azrieli Center Holon
26, HaRokhmim Str., Bldg. D
Holon, 5885849
Israel
Phone: +972-3-645-8414
Email: dromasca@avaya.com