Protocol for Forwarding Policy Configuration (FPC) in DMM
draft-ietf-dmm-fpc-cpdp-10
L. Bertz, S. Matsushima, M. Liebsch, S. Gundavelli, D. Moses, C. Perkins
IETF 101, London
2018-03-18
Agenda

• Updates Since Version 09
  • Templates
  • Policy
  • Topology & Service
  • Mobility Context
  • RPCs
• Next Steps
What is this work about..?

• Enable the separation of a mobility network‘s Control-Plane function from its Data-Plane function

• Enable distributed deployment of Control- and Data-Plane functions by abstracted Data-plane model and protocol messages

• Support multi-tenancy on a single real deployed D-plane network and multiple domains within a tenant
Changes since Version 09

Migration to a Template based framework. This affects all elements. The framework has a template definition language.

Topology

DPN Peer Groups and DPN Groups are now PeerServiceGroup and ServiceGroup.

The Topology system supports selection information by ServiceGroup or ServiceEndpoint.

Policy

Rule was inside Policy and now is Rule-Template and stands as a peer structure to Policy.

Types, e.g. Descriptor Types, Action Types, etc., are now templates that have no values filled in.

Mobility Aspects

The embedded rule from version 09 has been replaced by a template that has no predefined variables. All rules, pre-configured or embedded, are realized as Policy instantiations.

Mobility Context is migrated to a Template Design.

The Unassigned DPN is used to track requests vs. those that are installed, i.e. Agent assignment of Policy is supported.

Operation(s)

Bulk Configuration and Configuration now follow a style similar to YANG Patch. Agents MAY respond back with edits it made to complete the Client edit request.

All operations have a common error format.

Misc

Basename is split into two aspects. The first is version which applies to Templates. The second is checkpointing which applies to specific sections only.

RFC 5777 Classifiers have been added.
Templates

Simplify development and maintenance of the needed policies and other objects

+[-[Template] <U-Key, Name> (M) <Set>]
  +[-[Attributes] <Set> (M)]
  +-[Extensible ~ FALSE]  
  +-[Entity-State ~ Initial]  
  +-[Version]

Definition Notation

'[Att-Name: ]'  Mandatory Attribute is defined, but template does not provide any configured value.

'[Att-Name: Att-Value]'  Mandatory Attribute is defined, and has a statically configured value.

'[Att-Name: ~ Att-Value]'  Mandatory Attribute is defined, and has a default value.

'[Att-Name]'  Non-mandatory Attribute may be included but template does not provide any configured value.

'[Att-Name = Att-Value]'  Non-mandatory Attribute may be included and has a statically configured value.

'[Att-Name ~ Att-Value]'  Non-mandatory Attribute may be included and has a default value.
Configuring Policy Templates

A Policy Template may be configured in several stages by configuring default or missing values for Attributes that do not already have statically configured values.

Entity Configuration Blocks:

- Domain-Policy-Configuration
- DPN-Policy-Configuration
- Descriptor-Configuration
- Action-Configuration
- MN-Policy-Configuration
- Flow-Policy-Configuration

Pattern

[Entity Configuration Block]
  |    +-[Entity-Key] (M)
  |    +-[Attribute-Expression] <Set> (M)

DPN Example

+-[DPN] <G-Key>, <Name> (O) <Set>
  |    +-[Extensible: FALSE]
  |    +-[Interface] <L-Key> <Set>
  |      |    +-[Role] <U-Key>
  |      |    +-[Protocol] <Set>
  |      |    +-[Settings] (O)
  |    +-[Domain-Key]
  |    +-[Service-Group-Key] <Set> (O)
  |    +-[DPN-Policy-Configuration] <List> (M)
  |    +-[DPN-Resource-Mapping-Reference] (O)
Rule was inside Policy and now is Rule-Template and stands as a peer structure to Policy.

Types, e.g. Descriptor Types, Action Types, etc., are now templates that have no values filled in.
Policy Lifecycle

Design - Entity Status = Initial, PartiallyConfigured or Configured

DPN Policy Installation & Customization
Entity Status = Initial, PartiallyConfigured or Configured

Mobility Sessions Entity Status = Initial, PartiallyConfigured, Configured or Active
Topology & Service

DPN Groups => ServiceGroup
Definition: A collection of DPN interfaces serving some data-plane purpose.

DPN Peer Groups => PeerServiceGroup

ServiceEndpoint
Definition: is the collection of all services provided by DPN interfaces in the network.

Domain
Definition: represents a group of heterogeneous Topology resources typically sharing a common administrative authority.

DPN
Access-Technology is gone as it is assumed in Role (which is now a U-Key)

+-[DPN] <G-Key>, <Name> (O) <Set>
  +-[Extensible: FALSE]
  +-[Interface] <L-Key> <Set>
    |  +-[Role] <U-Key>
    |  +-[Protocol] <Set>
    |  +-[Settings] (O)
  +-[Domain-Key]
  +-[Service-Group-Key] <Set> (O)
  +-[DPN-Policy-Configuration] <List> (M)
  +-[DPN-Resource-Mapping-Reference] (O)

DPN NOW holds Policy

Installed-Policy Concepts from v09 are gone
Mobility Context

The embedded rule from version 09 has been replaced by a template that has no predefined variables. All rules, pre-configured or embedded, are realized as Policy instantiations.

Mobility Context is migrated to a Template Design.

The “Unassigned” DPN (reserved Key) is used to track requests vs. those that are installed, i.e. Agent assignment of Policy is supported.
RPCs

• Migrated to YANG-Patch Style (not quite compliant)
• Eliminated CONF_BUNDLES as a result
• Update now has more options including move, merge, etc.
• Also migrated ALL error responses to a common format (based also on YANG Patch)
Configure Output

• Unlike YANG-Patch we also permit updates (subsequent-edits) to complete an edit.

• This is the equivalent of Agent changes in responses used in v09
Next Steps

• This week
  • Improve or eliminate ServiceEndpoints
  • Add 1 more example to doc for topology selection
• Feedback - Tell us what is not intuitive!
• Edits, edits, edits
• More Examples
  • How many more to add (maybe a separate work)
• Reviews