A Policy Management Framework for Flow Distribution on Multihomed End Nodes

Koshiro Mitsuya <mitsuya@sfc.wide.ad.jp>
Romain Kuntz <kuntz@lsiit.u-strasbg.fr>
Shinta Sugimoto <shinta@sfc.wide.ad.jp>
Ryuji Wakikawa <ryuji@sfc.wide.ad.jp>
Jun Murai <jun@wide.ad.jp>
Table of Contents

- Motivations
- Scenario
- Requirements
- Current Situation
- New Framework
- Conclusion
Motivations

Contemplated multihomed environment

- Node equipped with multiple (wireless) network accesses
- Especially true in mobile environment
- multiple communication paths,
- Maintained by one or multiple multihoming protocols (MIP6/MCoA, SHIM6, SCTP, etc.)
- Each has multiple goals and benefits,
- Ubiquitous access, fault tolerance, load sharing, ...
- **Flow Distribution:** distribute flow via multiple available paths
Motivations

How to achieve flow distribution

- User/Application creates its desired policies,
- Described in term of cost, bandwidth, delay, jitter, etc.
- The policy management framework confronts them to the characteristics of each available path
- Resulting the filter rules,
- Filter rules used as input to the OS-specific filtering framework,
- Filter rules exchanged among peer hosts.
Example Scenario

Legend:

- **node**
- functional entity

mpro = multihoming protocol stack
flow dist. = flow distribution mechanism
Requirements
for the policy management framework

R1: Policy description: language definition
   R1.1 Makes the relation between flow and path characteristics,
   R1.2 Multihoming protocol independent.
R2: Multiple requesters (local or remote) management
R3: Policy resolution to filter rules and error management
R4: Filter rules description,
R5: Filter rules transport,
R6: Multiple filter rules processing,
R7: Transport security
# Current Situation

## Summary of existing specifications

<table>
<thead>
<tr>
<th></th>
<th>Multihoming protocol principles</th>
<th>Multihoming Protocol specifications</th>
<th>Flow distribution in implementations</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Mobile-IPv6 based (MIPv6, NEMO)</strong></td>
<td>Multiple CoA binded to a single HoA</td>
<td>MCoA, Flow Binding, Flow distribution</td>
<td>Ipfilter, PF, Netfilter</td>
</tr>
<tr>
<td><strong>SHIM6 based</strong></td>
<td>Upper Layer ID (ULID) mapped to one or more locators</td>
<td>Policy DB in the SHIM6 IP sub-layer, Multihoming SHIM API</td>
<td>SHIM6 API, Netfilter</td>
</tr>
<tr>
<td><strong>HIP based</strong></td>
<td>Host ID (HI, public key) mapped to one or several IP addresses</td>
<td>Multihoming SHIM API</td>
<td>impossible</td>
</tr>
<tr>
<td><strong>SCTP based</strong></td>
<td>Transport layer protocol</td>
<td>SCTP socket API</td>
<td>Socket API, library</td>
</tr>
</tbody>
</table>
Current Situation

Main Principles

- **Identifier/locator separation** concept,

- Flow Distribution achieved by *choosing proper locator*,

- Locator set by *configuring filter rules* (via Socket API, OS-specific framework, etc.).
Current Situation

Main Issues

- Usually associates the flow to a system or protocol-oriented path ID (eg: BID for MCoA),

  - **Protocol-dependent**

- Usually **do not exchange filter rules** among hosts (HIP, SHIM6, SCTP).

  - **Cannot specify** e.g. round-trip path

- Flow distribution **tightly depends on the OS** on which the implementation is running

  - **Hard to define** a generic flow distribution architecture.
New Framework

Main Principles

➡ Unified Policy management framework on top of the various flow distribution mechanisms,

➡ Using the existing mechanisms given by the OS.
New Framework

Main Principles

- **Policy Data Set:**
  - Describes flow in terms of costs, bandwidth, delay, jitter, etc.

- **Policy Management Framework:**
  - Confront the policy data set against the interfaces’ characteristics,
  - Produce filter rules,
  - Install / send filter rules to peer hosts.
New Framework
The Policy Data Set

• Generic language to define a common policy data set whatever the multihoming protocol or OS is running on the node (R1.2),

• Set of **Policy Rules:**
  • Tells which policy (flow + action) to apply when some **conditions** are met (R1.1).
New Framework

The Policy Data Set

conditions = expected network characteristics
policy = selector + action
New Framework

Policy Management Framework

- Processes the Policy Data Sets from multiple sources (R2),
- Confront the user policies with the actual path’s characteristics to produce filter rules (R3, R4),
- Uses the existing filtering framework to install the rules on local host (R6),
- Send the filter rules for remote host (R5, R7)
Conclusion

- Defined requirements for a policy management framework,
- Draft framework working on top of several multihoming protocols,
- **Next steps:**
  - Grammar definition for the Policy Data Set,  
  - `draft-mitsuya-monami6-flow-distribution-policy`
  - Framework implementation and evaluation.
Thank you,

Any questions?

Romain Kuntz <kuntz@lsiit.u-strasbg.fr>