An Implementation in OMNeT++ of Linux Rules for IP Routing

Alfonso Ariza, Eduardo Casilari, J. Hurtado-López

UNIVERSITY OF MÁLAGA, SPAIN
Desenzano, Italy - March 23th, 2012

Departamento de Tecnología Electrónica. University of Málaga
ETSI de Telecomunicación, Campus de Teatinos, 29071 – Málaga- Spain
E-mail: mjmoron@uma.es, ecasilari@uma.es
Typical IP routing merely based on the destination address

Need for **complex filtering & forwarding mechanisms** at the IP layer. Examples:

- Discard packets with specific characteristics
- Forwarding criteria can be founded on other parameters (e.g.: TCP/UDP ports, source address, etc.)

Linux: Alternative routing achieved with IP rules

Goal: to **emulate IP rules** in OMNeT++
• Integrated within the Routing Tables of Inet
• Executed at the IP layer
• Rules applied upon the packet arrival or departure
• Rules allow packets to bypass the existing routing table
• Rules defined in the same configuration file of that used to characterize Linux IPTables
• New labels for rules: `<rules> </rules>`
• Rules specification imitates that of actual Linux rules
Example of an rule: delete all packets flowing from source address 192.168.1.1

```bash
iptables -A INPUT -s 192.168.1.1 -j DROP
```

**Table: Command Options**

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>-s</code></td>
<td>Source address to filter</td>
</tr>
<tr>
<td><code>-p</code></td>
<td>Destination Address to filter</td>
</tr>
<tr>
<td><code>-A</code></td>
<td>The filter is used at the input or at the output</td>
</tr>
<tr>
<td><code>-p (tcp:udp)</code></td>
<td>Protocol</td>
</tr>
<tr>
<td><code>-sport</code></td>
<td>Source port</td>
</tr>
<tr>
<td><code>-dport</code></td>
<td>Destination port</td>
</tr>
</tbody>
</table>
Future extensions

• Rules for advanced forwarding.
• Rules to emulate selective random packet loss
• Rules to change the admission conditions dynamically
• Rules to duplicate packets (aiming at simulating multicast transmissions).
• Mechanisms to generate statistical results of the network performance