Universal Plug and Play (UPnP) Internet Gateway Device (IGD)-Port Control Protocol (PCP) Interworking Function

draft-bpw-pcp-upnp-igd-interworking-02
IETF 80-Prague, March 2011

M. Boucadair, R. Penno, D. Wing, F. Dupont
Agenda

• Architecture Model
• Rationale
• Open questions
• First experiment results
Architecture Model

**UPnP Control Point**
- H1: 192.168.1.2
- H2: 192.168.1.1

**IGD**

**PCP is used to control this NAT**

**UPnP IGD/PCP IWF**

**IGD:1 or IGD:2 may be used in the LAN side**
UPnP IGD-PCP IWF: Rationale

• IWF: Variables, Methods and Errors
• Control of a local firewall is not managed by the IWF but a means to control it is required
• Both the cases where a NAT is co-located or not with the IWF are covered
• When AddPortMapping() is used, a PREFER.FAILURE option is included in the PCP request
  – The IWF checks first if the requested port is in use locally or not
• No PREFER.FAILURE option is inserted in the corresponding PCP message when AddAnyPortMapping() is received for the UPnP CP
• A Local Mapping Table is maintained by the IWF
• When GetExternalIPAddress is received, a corresponding MAP4 is generated by the IWF with a short lifetime
### Some IGD:1 Implementations

#### Behaviour

<table>
<thead>
<tr>
<th>Software</th>
<th>Behaviour</th>
</tr>
</thead>
<tbody>
<tr>
<td>Microtorrent (uTorrent) v2.2</td>
<td>Calls GetSpecificPortMapping until it finds an external available port, and then calls AddPortMapping()</td>
</tr>
<tr>
<td>Azureus v4.6.0.2</td>
<td>Calls AddPortMapping, after it finds the external port is not available, then it tries the same port 5 more times by calling AddPortMapping, then it returns an error</td>
</tr>
<tr>
<td>Emule v0.50a</td>
<td>Calls AddPortMapping, after it finds the external port is not available, then it returns an error</td>
</tr>
</tbody>
</table>
Open Question # 1

• To list active mappings, an UPnP Control Point may issue GetGenericPortMappingEntry(), GetSpecificPortMappingEntry() or GetListOfPortMappings()

• In the current version of the I-D, these methods are not relayed to the PCP Server but are handled according to the content of the local mapping table
  – The answer may not be accurate since the IWF has only a local knowledge of the global mapping table
  – Some applications issue first Get* to check whether a port is in use or not before sending Add*
  – Open question: Do we allow relaying Get* to PCP MAP4?
    • No such method is currently defined in the base PCP
    • GET is only an extension
      – GET is defined in I-D.boucadair-PCP-failure
Open Question #2

• When DeletePortMappingRange() is used
  – The IGD-PCP IWF undertakes a lookup on its local mapping table to retrieve individual mappings instantiated by the requested Control Point and matching the signalled port range
  – If no mapping is found, "730 PortMappingNotFound" error code is sent to the UPnP Control Point
  – If a set of mappings are found, the IGD-PCP IWF generates individual PCP MAP4 delete requests corresponding to these mappings

• Question (optimization)
  – Do we allow the IWF to send a positive answer to the requesting UPnP Control Point without waiting to receive all answers from the PCP Server?
    • It is unlikely to encounter a problem in the PCP leg because the IWF has verified authorization rights and also the presence of the mapping in the local table
Open Question #3

• In case of loss of synchronization (reboot, power outage), the IWF sync its state with the PCP Server using GET/NEXT
  – Per-subscriber quota may be exhausted due to stale mappings

• Question
  – This procedure focuses on the scenario where no PCP Client is in the LAN side
    • Things may be complex when there are PCP Proxy/IWFs
  – GET/NEXT is not defined in the base PCP
    • GET/NEXT is defined in I-D.boucdair-pcp-failure
    • What to do with that I-D?
Open Question #4

• **Scenario**
  – UPnP Control Point creates a mapping with a 5 day lifetime
  – IWF creates the corresponding pinhole with one day lifetime, schedules a renewal each 24 hours and returns success
  – Two days after the renewal fails

• **What to do?**
  – State this is a rare failure case and we can do nothing?
  – Else?
Testing Activities: First Results
(Conducted by X. Deng)

- UPnP-PCP interworking function (based on -05, working on an update to be compatible with -07)
  - Plain IPv6 mode is used
- PCP server: Embedded with “Vendor A” DS-Lite AFTR
- UPnP-PCP IWF: based on OpenWrt OS and customized to support DS-Lite B4
- UPnP Client: utorrent, Emule, Bitcomet, Azureus v4.6.0.2, Shareaza v2.5.4
Testing Activities: First Results

• Add a pinhole
  – PCP-UPnP IWF transmits the request of the UPnP CP to the PCP Server, returns the result to the CP and updates the IWF mapping list

• Delete a pinhole
  – PCP-UPnP IWF transmits the request of UPnP CP to PCP Server, returns the result to CP and updates the IWF mapping list

• Get CPE's list of instructed mappings
  – On reboot, PCP-UPnP IWF acquires all instructed mappings related to the CPE by GET/GETNEXT requests from PCP server and update PCP proxy mapping list

• Refresh PCP pinhole according to lifetime
  – PCP-UPnP IWF, check lifetimes in mapping list and refresh specific PCP pinhole on PCP server

• Supported PCP Options
  – PREFER_FAILURE Option
  – Description PCP Option to associate a text with a PCP pinhole
  – Check whether the PCP Server is able to preserve port parity

• Misc
  – Check the PCP Server holds the external IP and port pair of a deleted port mapping for 120 seconds