GTS Support in jFed

Thijs Walcarius
thijs.walcarius@intec.ugent.be
Contents of this presentation

- What is jFed?
- Designing an experiment
- Using GTS resources
What is jFed?

- jFed is a Java-based framework for testbed federation
- Developed to support all resources in the Fed4FIRE-project

Fed4FIRE = Federation for Future Internet Research and Experimentation
Fed4FIRE assets - facilities

![Map of Fed4FIRE assets](image)

- Wired
- Wireless
- Open Flow
- Cloud
- Other
Fed4FIRE facilities

- Diverse technologies
- Diverse implementation stacks

→ Abstract this for the user
Goals of federation

- Make it easy for experimenters to use multiple testbeds
  - Single account
  - Single (or small number) of tools, choice of tools

- Multiple testbeds
  - To scale up
  - To use/combine special resources (e.g. wireless robots)
  - Redundancy (e.g. testbed in maintenance)
  - To re-use experiments (class exercises, scientifically, …)
  - To compare environments (e.g. wireless, openflow hardware, …)
Design principles

Multiple identity provider

Multiple tools

Testbeds trust IdPs in federation

Multiple testbeds

All of them can appear and disappear!
International federation and connectivity
International federation and connectivity

Compatible initiatives in:

- South-Korea
- Canada
- Japan
- Brazil
- China
What is jFed?

• Three components:
  ◦ jFed Probe
  ◦ jFed Automated Tester
  ◦ jFed Experimenter
jFed Probe

- Assists developers in testing API implementations
- Supports GTS API and more than 10 other API’s!
jFed Automated Tester

- Performs extensive full-automated tests
- Simulates complete workflow of an experiment
jFed Experimenter

- End-user tool to design, provision and manage experiments

<geantTestbedType xmlns="http://jfed.iminds.be/">
    <![CDATA[UntitledGts {}}]]>
    <dsl><![CDATA[UntitledGts {}}]]>
        host {
            id = "host0"
            port {
                id = "if0"
            }
        }

        host {
            id = "host1"
            port {
                id = "if0"
            }
        }

        link {
            id = "link0"
            port {
                id = "src"
                id = "dst"
            }
        }

        adjacency host0.if0, link0.src
        adjacency host1.if0, link0.dst
    </dsl>
</geantTestbedType>

<node client_id="node0" exclusive="true" component_map="" sliver_type="raw-pc"/>
<location xmlns="http://jfed.iminds.be/rspec/ext/jfedBonfire"></location>
</node>
</rspec>
## Slivers

<table>
<thead>
<tr>
<th>Sliver Testbed</th>
<th>Sliver ID</th>
<th>Expiration Time</th>
<th>Status</th>
</tr>
</thead>
<tbody>
<tr>
<td>[GTS] TwoLinkedHosts</td>
<td>55</td>
<td>2526-01-31 23:59:59</td>
<td>UNINITIALISED</td>
</tr>
</tbody>
</table>

### Node login information

<table>
<thead>
<tr>
<th>Node name</th>
<th>Hostname</th>
<th>Port</th>
<th>Username</th>
<th>Login</th>
</tr>
</thead>
<tbody>
<tr>
<td>[GTS TwoLinkedHosts] host1</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>[GTS TwoLinkedHosts] host0</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Task progress:
- Creating GTS Project: Created project 'jfeddemo2'
- Starting GTS Resource 'TwoLinkedHosts'
- Fetching GTS resources description of project jfeddemo2
- Waiting for activation of GTS reservation '55'

Checked status 4 times. Next check in 10 seconds.
Thank you!

http://jfed.iminds.be