Presentation at HAW Hamburg

An Experiment Tutorial for the NorNet Core Testbed at HAW Hamburg

Thomas Dreibholz
Simula Research Laboratory

22 March 2016
Contents

- Preparations
- Getting an Overview of the Testbed
- Using a Slice
- A Practical Example
- Conclusion
Overview:
Preparations

- Preparations
- Getting an Overview of the Testbed
- Using a Slice
- A Practical Example
- Conclusion
Tutorial Accounts

- You should have received an account
  - Username
  - Password
- Valid for:
  - SSH login server
  - PLC server

Do you have an account? If not, ask!
Initial Tasks

- **Account for our SSH login server** gatekeeper.nntb.no:
  - Server is gateway into NorNet Core network
  - `ssh <username>@gatekeeper.nntb.no`
  - Use port forwarding to access PLC and Monitor servers:
    - Forwards TCP port 2000 to PLC server's HTTPS port
    - Forwards TCP port 2001 to Monitor server's HTTP port
  - **Account for the PLC server** plc.simula.nornet (inside NorNet Core only):
    - Login: `<username>@haw.nornet`
Access to PLC and Monitor

• Via port forwarding:

• Inside NorNet Core network:
  – Monitor: http://monitor.simula.nornet
  – PLC: https://plc.simula.nornet

Is everybody able to log in?
Overview:
Getting an Overview of the Testbed
See http://monitor.simula.nornet within NorNet Core!
PLC User Interface: Sites View

See https://plc.simula.nornet within NorNet Core!
PLC User Interface: Nodes View

Node state: should be “boot”
PLC User Interface: Account View

- Upload your SSH public key here!
- Public keys get distributed to all nodes (may take up to 1 hour!)

For this tutorial: This has already been done!
Overview:
Using a Slice

- Preparations
- Getting an Overview of the Testbed
- Using a Slice
- A Practical Example
- Conclusion
The Test Slice *haw_tutorial*

- A test slice has already been created:
  - Name: *haw_tutorial*
  - Special NorNet Core properties:
    - Own IP addresses on each node
    - IPv4 and IPv6
    - Multiple ISPs (at sites with several ISPs)
  - The slice is instantiated on all nodes by a sliver (LXC container)
  - Your account is mapped as user to *haw_tutorial*
Logging In

- From the login server:
  
  - `ssh -i <your private key> <slice name>@<node name>`

- Examples (private key is in `~/.ssh/id_rsa`, slice is `haw_tutorial`):
  
  - `ssh -i ~/.ssh/id_rsa haw_tutorial`
  - `ssh -i ~/.ssh/id_rsa haw_tutorial`
  - `ssh -i ~/.ssh/id_rsa haw_tutorial`
  - `ssh -i ~/.ssh/id_rsa haw_tutorial`
  - `ssh -i ~/.ssh/id_rsa haw_tutorial`

- Note: login is via node's SSH server to sliver on the node!

**Use PLC to find other nodes. There are more than 100 nodes!**
Note the Different Entities: Server, Node, Sliver

**Server** (physical)

**Node** (virtual)
- Sliver hu_multipath
- Sliver haw_tutorial
- Sliver ntnu_test
- Sliver due_rserpool
- Sliver uib_mptcp
- ...

**Slice:**
- User list
- Node list

Sliver = an instance of a slice on a node

ssh <Sliver>@<Node>

Forwarding to sliver!
Inside a Sliver

- Each sliver contains a Fedora Core 18 environment

**Obtain root access:**
- `su`
- `sudo bash`

**Install custom software:**
- `yum install <package> ...`
  - **Example:** `yum install netperfmeter`

**Show IP addresses and routes:**
- `ip -4 addr show ; ip -4 route show`
- `ip -6 addr show ; ip -6 route show`

---

**Remember:** slivers have their own addresses!
Nodes, Slivers and Addresses

- **Node:**
  - The node itself, e.g. altenessen.ude.nornet
  - Used for SSH login

- **How to find sliver addresses of a node?**
  - Look inside the sliver itself (login to sliver → ip addr show)
  - Ask the DNS server:
    - Use “dig” (part of bind-utils package for Fedora Core)
    - `dig <slice name>.<node name>.<site name>.nornet`
    - But replace “_” by “–” in slice name!
    - Examples for haw_tutorial slice:
      - `dig haw-tutorial.altenessen.ude.nornet any` to obtain primary provider (it is in the CNAME, here: “dfn”)
      - `dig haw-tutorial.altenessen.all.ude.nornet any` to obtain all providers' addresses
      - `dig haw-tutorial.solvang.all.simula.nornet` without “any” → gets only A RRs (i.e. IPv4 addresses)
A *dig* Example

```
olal@nordberg:~$ dig haw-tutorial.solvang.all.simula.nornet any
;; >> DiG 9.9.2-P1 <<>> haw-tutorial.solvang.all.simula.nornet any
...
;; ANSWER SECTION:
haw-tutorial.solvang.all.simula.nornet. 86400 IN A 10.2.1.130
haw-tutorial.solvang.all.simula.nornet. 86400 IN A 10.1.1.130
haw-tutorial.solvang.all.simula.nornet. 86400 IN AAAA 2001:700:4100:101::82:69
haw-tutorial.solvang.all.simula.nornet. 86400 IN AAAA 2001:700:4100:201::82:69
haw-tutorial.solvang.all.simula.nornet. 86400 IN HINFO "Amiga 5000" "Slice haw_tutorial"
haw-tutorial.solvang.all.simula.nornet. 86400 IN LOC 59 53 45.240 N 10 37 39.360 E 15.00m

;; AUTHORITY SECTION:
simula.nornet. 86400 IN NS ns.ntnu.nornet.
...
```
Overview:
A Practical Example

- Preparations
- Getting an Overview of the Testbed
- Using a Slice
- A Practical Example
- Conclusion
A Multi-Path Routing Test

- Select two nodes at different sites
  - List: [https://www.nntb.no/pub/nornet-configuration/NorNetCore-Sites.html](https://www.nntb.no/pub/nornet-configuration/NorNetCore-Sites.html)
  - Login to *haw_tutorial* sliver: `ssh haw_tutorial@<node name>`
  - Check IP addresses: `ip -4 addr show dev eth0`
  - Example:
    - *haw-tutorial.kettwig.ude.nornet*: 10.30.42.122 10.31.42.122
      - ISPs: 30=DFN, 31=Versatel (an ADSL connection)
    - *haw-tutorial.frogner.simula.nornet*: 10.1.1.131 10.2.1.131 10.4.1.131 10.9.1.131
      - ISPs: 1=UNINETT, 2=Hafslund, 4=Telenor, 9=PowerTech
  - Try ping/traceroute:
    - `ping [-f] [-s <size>] [-c <count>] <dest IP> -I <src IP>`
    - `traceroute <dest IP> -s <src IP>`
    - Look at the second and third hop (and their reverse DNS lookups)!
    - What do you see?
Some Flood Ping Results

haw_tutorial@kettwig.ude.nornet # ping -c 1000 -s 1400 -f 10.1.1.129 -l 10.30.42.122
PING 10.1.1.129 (10.1.1.129) from 10.30.42.122 : 1400(1428) bytes of data. 1000 packets transmitted, 1000 received, 0% packet loss, time 14591ms rtt min/avg/max/mdev = 70.115/108.064/177.958/26.870 ms

haw_tutorial@kettwig.ude.nornet # ping -c 1000 -s 1400 -f 10.2.1.129 -l 10.30.42.122
PING 10.2.1.129 (10.2.1.129) from 10.30.42.122 : 1400(1428) bytes of data. 1000 packets transmitted, 1000 received, 0% packet loss, time 14783ms rtt min/avg/max/mdev = 31.009/76.446/136.024/27.666 ms

haw_tutorial@kettwig.ude.nornet # ping -c 1000 -s 1400 -f 10.1.1.129 -l 10.31.42.122
PING 10.1.1.129 (10.1.1.129) from 10.31.42.122 : 1400(1428) bytes of data. 1000 packets transmitted, 999 received, 0% packet loss, time 14412ms rtt min/avg/max/mdev = 121.153/175.432/252.685/28.585 ms

haw_tutorial@kettwig.ude.nornet # ping -c 1000 -s 1400 -f 10.2.1.129 -l 10.31.42.122
PING 10.2.1.129 (10.2.1.129) from 10.31.42.122 : 1400(1428) bytes of data. 1000 packets transmitted, 999 received, 0% packet loss, time 14182ms rtt min/avg/max/mdev = 78.643/124.496/207.773/26.729 ms

RTT differences among provider combinations; higher ADSL delay (Versatel)
### Some Traceroute Results

<table>
<thead>
<tr>
<th>Command</th>
<th>Target IP</th>
<th>Hops</th>
<th>Latency</th>
</tr>
</thead>
<tbody>
<tr>
<td><a href="mailto:haw_tutorial@kettwig.ude.nornet">haw_tutorial@kettwig.ude.nornet</a> # traceroute 10.1.1.129 -s 10.30.42.122</td>
<td>traceroute to 10.1.1.129 (10.1.1.129), 30 hops max, 60 byte packets</td>
<td>1</td>
<td>essen.dfn.ude.nornet (10.30.42.1) 1.190 ms 1.739 ms 1.031 ms</td>
</tr>
<tr>
<td></td>
<td></td>
<td>2</td>
<td>dfn.ude.uninett.simula.nornet (192.168.178.10) 56.972 ms 56.722 ms 56.853 ms</td>
</tr>
<tr>
<td></td>
<td></td>
<td>3</td>
<td>haw-tutorial.frogner.uninett.simula.nornet (10.1.1.129) 100.773 ms 99.513 ms 99.337 ms</td>
</tr>
<tr>
<td><a href="mailto:haw_tutorial@kettwig.ude.nornet">haw_tutorial@kettwig.ude.nornet</a> # traceroute 10.2.1.129 -s 10.30.42.122</td>
<td>traceroute to 10.2.1.129 (10.2.1.129), 30 hops max, 60 byte packets</td>
<td>1</td>
<td>essen.dfn.ude.nornet (10.30.42.1) 1.190 ms 1.739 ms 1.031 ms</td>
</tr>
<tr>
<td></td>
<td></td>
<td>2</td>
<td>dfn.ude.uninett.simula.nornet (192.168.178.10) 56.972 ms 56.722 ms 56.853 ms</td>
</tr>
<tr>
<td></td>
<td></td>
<td>3</td>
<td>haw-tutorial.frogner.hafslund.simula.nornet (10.2.1.129) 100.773 ms 99.513 ms 99.337 ms</td>
</tr>
<tr>
<td><a href="mailto:haw_tutorial@kettwig.ude.nornet">haw_tutorial@kettwig.ude.nornet</a> # traceroute 10.1.1.129 -s 10.31.42.122</td>
<td>traceroute to 10.1.1.129 (10.1.1.129), 30 hops max, 60 byte packets</td>
<td>1</td>
<td>essen.versatel.ude.nornet (10.31.42.1) 1.830 ms 2.633 ms 2.609 ms</td>
</tr>
<tr>
<td></td>
<td></td>
<td>2</td>
<td>versatel.ude.uninett.simula.nornet (192.168.133.222) 127.768 ms 127.954 ms 127.507 ms</td>
</tr>
<tr>
<td></td>
<td></td>
<td>3</td>
<td>haw-tutorial.frogner.uninett.simula.nornet (10.1.1.129) 182.544 ms 182.564 ms 182.269 ms</td>
</tr>
<tr>
<td><a href="mailto:haw_tutorial@kettwig.ude.nornet">haw_tutorial@kettwig.ude.nornet</a> # traceroute 10.2.1.129 -s 10.31.42.122</td>
<td>traceroute to 10.2.1.129 (10.2.1.129), 30 hops max, 60 byte packets</td>
<td>1</td>
<td>essen.versatel.ude.nornet (10.31.42.1) 1.830 ms 2.633 ms 2.609 ms</td>
</tr>
<tr>
<td></td>
<td></td>
<td>2</td>
<td>versatel.ude.uninett.simula.nornet (192.168.133.222) 127.768 ms 127.954 ms 127.507 ms</td>
</tr>
<tr>
<td></td>
<td></td>
<td>3</td>
<td>haw-tutorial.frogner.hafslund.simula.nornet (10.2.1.129) 79.603 ms 75.599 ms 69.910 ms</td>
</tr>
</tbody>
</table>

---

**Hop 2: Router's ICMP TTL Exceeded is sent back via Simula's primary ISP!**

---

[ simula . research laboratory ] - by thinking constantly about it
What Else To Do?

- Try the same with IPv6!
  - ping6 [-f] [-s <size>] [-c <count>] <dest IP> -I <src IP>
  - traceroute6 <dest IP> -s <src IP>

- Try NetPerfMeter!
  - Supports TCP including MPTCP, SCTP, UDP, DCCP
  - Server side: netperfmeter <port>
  - Client side: netperfmeter <server>:<port> <flow details> ...
    (see manpage for details!)

- Install custom software
  - But note: do not assume the slivers to be permanent storages
  - Write scripts to automatise installation
  - In case of problems, nodes may just be wiped and reinstalled

And, of course, try your own experiments in NorNet!
Overview:

Conclusion

- Preparations
- Getting an Overview of the Testbed
- Using a Slice
- A Practical Example
- Conclusion
Conclusion and Future Work

- **NorNet Core is ready for your ideas!**
  - Think about your experiments
  - Let them run in NorNet Core

- How to get permanent access?
  - **Talk to us!**
  - Provide some information on your project
    Let us **discuss the details** about running your experiment in NorNet Core!

**In case of questions, ask us!**
“NorNet wants to be a building block of the railroad to heaven” ...

... and not be another unused testbed that paves the road to hell!
Any Questions?

Visit https://www.nntb.no for further information!