Implementation and Evaluation of a NAT-Gateway for the General Internet Signaling Transport Protocol

Roland Bless and Martin Röhricht
Motivation

- **Signaling protocols** useful set of tools
  - Dynamically install, maintain, and manipulate state in network nodes
  - Create messaging associations between signaling peers

- **Network Address Translation (NAT) gateways**
  - Mitigate potential shortage of IPv4 addresses
  - Translate IP address and UDP/TCP port information

- **Signaling messages carry IP address information** in their payload
  - NAT gateway must be GIST-aware
    - Rewrite addressing information in signaling message’s payload

→ Create an application level gateway for the **General Internet Signaling Transport (GIST)** protocol
Next Steps in Signaling Framework

- IP-based signaling framework
- Two-layered approach
- General Internet Signaling Transport Protocol (GIST)
  - Routing and transport of signaling messages
    - Message Routing Information (MRI)
    - Network Layer Information (NLI)
    - Messaging Associations
  - 3-way handshake (QUERY, RESPONSE, CONFIRM) plus DATA
  - Supports delayed-state installation
    - Installation of routing state at Responding Node delayed until final CONFIRM arrives
Signaling Message’s Address Translation

- **Transparent translation**
  - Translate GIST header fields as is done with Layer 3 and 4
  - Not applicable if cryptographic protection is used

- **Non-transparent translation**
  - Use special [NAT Traversal Object (NTO)]
    - Must be included by NAT gateway into initial QUERY message
    - Echoed back by Responding Node

```
<table>
<thead>
<tr>
<th>MRI Length</th>
<th>Type Count</th>
<th>NAT Count</th>
<th>Reserved</th>
</tr>
</thead>
<tbody>
<tr>
<td>Original Message Routing Information</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>List of translated objects</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Length of opaque information</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Information replaced by NAT #1</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Length of opaque information</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Information replaced by NAT #N</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
```
GIST handshake with GIST-aware NAT-gateway

Querying Node
IP: 10.1.2.1

GIST-aware NAT gateway
IP$_1$: 10.1.2.2
IP$_2$: 10.2.3.2

Responding Node
IP: 10.2.3.3

Querying Node:
- Src IP: 10.1.2.1
- MRI (QN)
- NLI (QN)

GIST-aware NAT gateway:
- MRI, NLI, IP, and UDP headers are translated
- Only IP and UDP headers are translated
- MRI (QN)
- NLI (QN)

Responding Node:
- Src IP: 10.2.3.3
- MRI (QN)
- NLI (RN)

State installation MRI (QN)

MRI (QN)
NLI (QN)
CONFIRM
Src IP: 10.1.2.1

MRI (NAT)
NLI (NAT)
QUERY
Src IP: 10.2.3.2

MRI (QN)
MRI (QN)
NLI (RN)
RESPONSE
Src IP: 10.2.3.3

MRI (QN)
NLI (QN)
CONFIRM
Src IP: 10.2.3.2
Implementation

- **Kernel part**
  - Intercept and filter GIST packets
  - Use Linux *netfilter* framework
  - Communication to user-space via Linux *netlink* messaging system

- **User-space part**
  - Performs remaining packet translations
    - Translate IP and UDP header
    - Translate address information in MRI and NLI
    - Insert NAT Traversal Object
    - Serialize GIST PDU, re-calculate IP and UDP checksums
  - Based on existing NSIS-ka implementation ([http://nsis-ka.org](http://nsis-ka.org))
    - Not entire NSIS-ka suite (~40,692 lines of code) required

- Kernel module – 420 lines of C code
- GIST-aware NAT gateway – 680 lines of C++ code
Evaluation

- Evaluation in a real testbed environment
  - Four standard PCs (Pentium IV 2.8 GHz, 4 GB RAM, Gbit Ethernet)
  - Ubuntu 10.04 with Linux kernel 2.6.32
  - Latency intentionally kept small (~0.165 ms)

- Processing time of different GIST PDUs on first GIST-aware NAT gateway

- Processing time for complete GIST handshake and one subsequently sent DATA message
Evaluation – Processing time for initial QUERY message

- Processing time for initial QUERY messages
- NAT traversal objects are included
Evaluation – Processing Time of different GIST PDUs

- Measured on first GIST-aware NAT gateway
- Over TCP
- Over UDP
Evaluation – Complete GIST handshake

- Complete GIST handshake with one subsequently sent DATA message
- Measured on Querying Node using TCP
- Measured on Querying Node using UDP
Conclusion

- Design of a **NAT application level gateway** for the General Internet Signaling Transport protocol
- Implementation of a **NAT Traversal Object** as being specified
  - Works as expected
  - Use GIST Responder Cookie for delayed-state installation
→ Allows NSIS signaling messages to safely traverse such NAT gateways

- Evaluations show
  - Slight overhead for initial GIST QUERY messages
  - Almost no overhead for subsequent GIST messages
  - Only small impact on duration of complete GIST handshake
  - **Delayed-state installation** with no notable performance overhead
Thank you for your attention

Questions?
Institute of Telematics, Department of Comp. Science

03.09.2010 Implementation and Evaluation of a NAT-Gateway for the General Internet Signaling Transport Protocol
Evaluation Results – Different PDUs

- Processing time of different GIST PDUs on the first GIST-aware NAT gateway

<table>
<thead>
<tr>
<th>Processing time on the first GIST-aware NAT gateway</th>
<th>Avg [ms]</th>
<th>Median [ms]</th>
<th>StdDev [ms]</th>
</tr>
</thead>
<tbody>
<tr>
<td>UDP Query (with NTO)</td>
<td>2.153</td>
<td>2.161</td>
<td>0.152</td>
</tr>
<tr>
<td>TCP Response (with NTO)</td>
<td>0.012</td>
<td>0.011</td>
<td>0.004</td>
</tr>
<tr>
<td>UDP Response (with NTO)</td>
<td>0.026</td>
<td>0.026</td>
<td>0.002</td>
</tr>
<tr>
<td>TCP Confirm</td>
<td>0.010</td>
<td>0.009</td>
<td>0.003</td>
</tr>
<tr>
<td>UDP Confirm</td>
<td>0.013</td>
<td>0.012</td>
<td>0.002</td>
</tr>
<tr>
<td>TCP Data</td>
<td>0.009</td>
<td>0.009</td>
<td>0.001</td>
</tr>
<tr>
<td>UDP Data</td>
<td>0.008</td>
<td>0.007</td>
<td>0.001</td>
</tr>
</tbody>
</table>
Evaluation Results – Complete handshakes

Complete GIST handshake with one subsequently sent DATA message

<table>
<thead>
<tr>
<th>GIST handshake duration using TCP</th>
<th>Avg [ms]</th>
<th>Median [ms]</th>
<th>StdDev [ms]</th>
</tr>
</thead>
<tbody>
<tr>
<td>Using NATs, with DSI</td>
<td>6.843</td>
<td>6.820</td>
<td>0.178</td>
</tr>
<tr>
<td>Using NATs, without DSI</td>
<td>6.659</td>
<td>6.630</td>
<td>0.182</td>
</tr>
<tr>
<td>No NATs, with DSI</td>
<td>1.816</td>
<td>1.746</td>
<td>0.210</td>
</tr>
<tr>
<td>No NATs, without DSI</td>
<td>1.797</td>
<td>1.732</td>
<td>0.176</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>GIST handshake duration using UDP</th>
<th>Avg [ms]</th>
<th>Median [ms]</th>
<th>StdDev [ms]</th>
</tr>
</thead>
<tbody>
<tr>
<td>Using NATs, with DSI</td>
<td>5.737</td>
<td>5.722</td>
<td>0.127</td>
</tr>
<tr>
<td>Using NATs, without DSI</td>
<td>5.744</td>
<td>5.720</td>
<td>0.154</td>
</tr>
<tr>
<td>No NATs, with DSI</td>
<td>1.432</td>
<td>1.413</td>
<td>0.124</td>
</tr>
<tr>
<td>No NATs, without DSI</td>
<td>1.449</td>
<td>1.407</td>
<td>0.136</td>
</tr>
</tbody>
</table>