Robust Forwarding in Structured Peer-to-Peer Overlay Networks

Wang-kee Poon Rocky K. C. Chang
Department of Computing, The Hong Kong Polytechnic University
{cswkpoon, csrcchang}@comp.polyu.edu.hk

Overview
We consider the problem of robust message forwarding in structured peer-to-peer overlay networks (e.g. CAN, Chord, Pastry, and Tapestry).

- We particular address forwarding faults includes message dropping, undesirable delay, and forwarding to undesirable nodes.
- This poster presents Fence, a fault diagnosis protocol, to detect, locate and isolate faulty nodes that are involved in injecting forwarding faults to lookup messages in a peer-to-peer overlay network.

Motivation and Related Work
- Vulnerability of the overlay networks: Because of openness to participating in message forwarding and a lack of facilities ensuring the forwarding integrity, adversaries can easily disrupt the normal message lookup forwarding.
- Severe damage caused by forwarding attacks: Even a small percentage of faulty nodes that inject forwarding faults can seriously degrade the performance of a P2P network. In the simulation study of Pastry, the probability of successful delivery of a lookup message is dropped to 65% in the presence of 10% faulty nodes in a network of 100,000 nodes [1].
- Lack of solutions with low communication complexity: Existing approaches mainly focus on providing resilience by forwarding each request via multiple paths concurrently. These approaches increase reliable at the cost of high communication complexity.
- Extension of robust forwarding protocols to the overlay networks: Existing robust forwarding protocols (reference in [2]) rely on source routing and continuous end-to-end communications, which are not suitable for the overlay networks.

We present a robust forwarding protocol using a fault diagnosis approach that works in overlay networks with fast response and low communication complexity.

Fence: How-to
Feedback from forwarding nodes:
- On receiving a lookup message from a node (either the source or another forwarding node), each forwarding node sends an acknowledgment to the node that sends the lookup message.
- On receiving an acknowledgment from its next hop, each forwarding node forwards the acknowledgment to the source as a proof of forwarding.

When a source receives a proof of forwarding, it knows
1. a particular forwarding node has forwarded the lookup message;
2. another particular node has received the message;
3. the period that the message has stayed at the forwarding node (Derivation is required)

Example: For a normal forwarding,

```
<table>
<thead>
<tr>
<th>Time</th>
<th>Type</th>
<th>Node</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>MSG</td>
<td>1</td>
</tr>
<tr>
<td>33</td>
<td>ACK</td>
<td>33</td>
</tr>
<tr>
<td>38</td>
<td>MSG</td>
<td>38</td>
</tr>
<tr>
<td>40</td>
<td>ACK</td>
<td>40</td>
</tr>
</tbody>
</table>
```

After receiving ACK33 from node 33, node 1 sets a timer that expires at the mark "Timeout" shown in the left figure. The timer expires if

```
Timeout < \Delta_{33} = 1, A33
```

Node 1 adjusts the value \Delta_{33} to control maximum forwarding delay where

```
\Delta_{33} = Timeout - \Delta_{33} = 1, A33
```

Message dropping is a special case of delays with an infinite delay period.

Recovery and Isolation
After location of a forwarding fault, a source can send the lookup message to the previous hop to recover forwarding.

- To isolate located faulty nodes, a source can attach a list of the nodes into it.

References

The further information of our work can be accessed via http://www.comp.polyu.edu.hk/~cswkpoon/fence