A Testbed-based Visualization of Misbehavior in Peer-to-Peer Systems

Christian Gottron, Daniel Seither, André König and Ralf Steinmetz
Multimedia Communications Lab (KOM), TU-Darmstadt, Darmstadt, Germany
{christian.gottron, daniel.seither, andre.koenig, ralf.steinmetz}@kom.tu-darmstadt.de

I. Introduction

The Pastry [2] Distributed Hastable (DHT) provides decentralized peer-to-peer services in a scalable and efficient manor. Peers do not have to provide links to each node in the network, but only to a small subset of nodes. Due to this, lookup requests for services offered in the peer-to-peer system mostly can not be sent to the destination directly, but have to be forwarded to a node that is logically located closer to the destination regarding the structure of the Peer-to-Peer overlay. This intermediate node again forwards the lookup request to an intermediate node closer to destination, unless the destination is reached. Even though this approach reduces the routing table size and, therefore, the overhead, each intermediate node has to behave benign. Yet, this can not be assumed in a realistic scenario. Therefore, mechanisms are required to increase the robustness of the routing algorithms.

To visualize the impact of misbehavior in Pastry, we introduce a tool that provides a graphical representation of the peer-to-peer structure of the overlay network and the routing behavior of benign and malicious nodes. By now, we implemented the *Incorrect Lookup Routing* attack and the *Redundant Routing* algorithm as countermeasure for this attack. Yet, further attacks and countermeasures can be implemented easily.

II. Background

A. Pastry

Pastry was introduced by Rowstron et al. as a hybrid DHT. Each node in Pastry is identified by a unique node ID of 128 bit length. In order to route requests, a recursive approach is used that is based on comparing the prefixes of the own node ID and the ID of the destination. For this, the main routing table of Pastry provides links to nodes with an ID that is closer to the ID of the destination node in terms of longest prefix match.

The FreePastry [2] implementation of Pastry is directly based on the works of Rowstron et al. and provides several applications as e.g. Past [3].

B. Incorrect Lookup Routing

The Incorrect Lookup Routing was initially presented by Sit and Morris [4] as a straight forward yet very efficient routing attack on DHTs. Malicious nodes drop received lookup requests instead of forwarding them correctly. This results in a denial of service attack as the destination node does not receive the request message. DHTs as Pastry are not inherently resistant against those attacks due to their recursive routing algorithm and the lack of influence on the selected route.

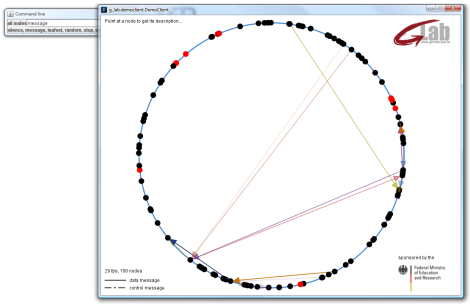

Fig. 1. The visualization tool

C. Redundant Routing

In order to increase the robustness of the recursive routing algorithm, Castro et al. [1] proposed a parallel *Redundant Routing* approach. Instead of sending a single route request, each request is sent to multiple nodes in order to increase the probability that at least one request reaches the destination. Though this results in an increased traffic, the countermeasure is reasonably efficient.

III. Visualization Tool

In order to visualize the impact of malicious nodes on Pastry, we implemented a graphical monitoring tool. The Pastry network is visualized by an overlay ring and each Pastry node is represented on the ring ordered by the Pastry ID. Although Pastry combines a hierarchical with a ring structure, we used the ring for visualizing the DHT as the hierarchical structure is not globally valid, but varies per peer.

With this tool, we are able to control the Pastry network directly, as we can select single nodes or the whole network and initiate a command. The tool can be extended easily by additional attacks and/or additional countermeasures.

References

[1] M. Castro et al. Secure routing for structured peer-to-peer overlay networks. *SIGOPS Oper. Syst. Rev.*, 36(SI):299–314, 2002.
[2] A. Rowstron et al. Pastry: Scalable, distributed object location and routing for large-scale peer-to-peer systems. In *Proc. of Middleware'01*, 2001.
[3] A. Rowstron et al. Storage management and caching in past, a large-scale, persistent peer-to-peer storage utility. In *Proc. of SOSP'01*, 2001.
[4] E. Sit et al. Security considerations for peer-to-peer distributed hash tables. In *Proc. of IPTPS '01*, pages 261–269, 2002. Springer-Verlag.