Distributed Systems Unassessed Tutorial 6

Peer to Peer Networking

Lecturer: Morris Sloman

1) What are the main guarantees that users expect conventional servers to offer?

2) The guarantees offered by conventional servers may be violated by:
   a) physical damage to the host;
   b) errors or inconsistencies by system administrators or their managers;
   c) successful attacks on the security of the system software;
   d) hardware or software errors.

   Give two examples of possible incidents for each type of violation. Which of them could be described as a breach of trust or a criminal act? Would they be breaches of trust if they occurred on a personal computer that was contributing some resources to a peer-to-peer service? Why is this relevant for peer-to-peer systems?

3) Explain how the use of the secure hash of an object to identify and route messages to it ensures that it is tamper-proof. What properties are required of the hash function? How can integrity be maintained even if a substantial proportion of peer nodes are subverted?

4) It is often argued that peer-to-peer systems can offer anonymity for (a) clients accessing resources and (b) the hosts providing access to resources. Discuss each of these propositions. Suggest a way in which the resistance to attacks on anonymity might be improved.

Solution Distributed Systems Tutorial 6

Peer to Peer Networking

Lecturer: Morris Sloman

1) The main guarantees are:
   - to maintain a consistent state of the objects that they store;
   - to make their service continuously available.

2) a – power failure, act of sabotage
   b – accidental deletion of file, permission failure
   c – tampering of data, denial of service attack
   d – hard disk failures, program bugs

   The differences in what is 'trusted behaviour' for servers and PCs is relevant because peer-to-peer system must be designed to cope with the looser interpretation of trust for PCs.

3) If the routing mechanism is secure, then objects will only be contactable at an address that is derived from the secure hash. More importantly, even if the routing mechanism and some peer nodes are compromised, a client can request the content of the object and check its validity by computing the secure hash and comparing it with the GUID. The secure hash must be a one-way function for which it is computationally infeasible to generate two objects that hash to the same result. Else an attacker could store one value and then replace it with the other at a later date.

4) The general argument is that although TCP/IP messages contain the IP addresses of the source and destination nodes, when an application-level multi-hop routing overlay is used, only the previous and next node in the route can be discovered when packets are intercepted or logged somewhere in the network. A GUID does not by itself provide any information about the location of the node that hosts it. But if an attacker can gain knowledge of the contents of some of the routing tables, this property is compromised. Furthermore, an attacker with eavesdropping access at several points in the network could send 'probe' messages to specific GUIDs and observe the resulting IP traffic. This is likely to reveal quite a lot of information about the location of the GUID.
   So the proposition that clients and resource hosts can remain anonymous is only true for weak attackers with limited access to the network.
   This resistance to attacks might be improved by generating several outgoing messages for each incoming request at an intermediate node, all but one of the messages would be treated as a 'dummy' message and destroyed at the next node. This would incur a substantial additional cost in network traffic.