Detecting Transparent Network Devices

-Wijaya
Supervisor: Dr Tim Moors

WHY detect?
Network faults can be difficult to localise, diagnose and recover from, particularly if they are occurring in a device that the end user is unaware of.

Thus we need some mechanism to determine the network devices that an end user is dependant upon when accessing network services.

This will be useful for:
• Network Fault diagnosis
• Interpreting Network Service Outage Notifications
• Generating Fault Reports
• Protocol Tuning

WHAT to detect?
• Bridges (Ethernet Switches)
• Web Caches
• Network Address Translators
• Firewalls

HOW to detect?
• Protocol Analysis … identification of specific packets, or modification of packet header fields
• Data Analysis … noticing a change in network performance

Detecting BRIDGES

We developed two different algorithms to detect bridges and the topology they are configured.

1. Layer 2 Ethernet Topology Discovery Using BPDUs

We can identify a bridge by capturing Bridge Protocol Data Units (BPDU) which are forwarded between bridges to maintain the Spanning Tree.

By collating these BPDUs amongst end stations, it is possible to determine the layer 2 Ethernet physical topology using the Root Path Cost.

2. Layer 2 Ethernet Topology Discovery by training and probing switches

Bridges learn which stations are located on which ports by observing the source address of the frames it receives on each port.

By training switches that certain stations are located on certain ports, and sending suitable probe frames and observing where they are delivered, we are able to determine the topology.

Detecting WEB CACHES

1. Resetting the Time To Live (TTL) field

It has been observed that web caches such as the UNSW Web Cache, reset the TTL of the IP packet header to a particular value for every website visited.

2. Performance Measurements
   • Elapsed Time
   • Inter-Arrival Time
   • Round Trip Time

By repeatedly accessing the same web object, it can be observed that performance metrics such as those mentioned above will decrease.

Detecting NETWORK ADDRESS TRANSLATORS

STUN: Simple Traversal of UDP Datagrams Through Network Address Translators

STUN is a protocol that allows an application on an end-system to discover the presence and learn the type of NATs between it and the public Internet.

Conclusion

By detecting these transparent network devices, we have managed to improve the network dependability for the end user.