Preventing Network Instability Caused by Propagation of Control Plane Poison Messages

X. Du / M.A. Shayman (UMD), R. Skoog (Telcordia Technologies)

1. Motivation

•There have been a number of incidents where commercial data and telecommunication networks have collapsed due to their entering an unstable mode of operation.
•The events were caused by unintentional triggers activating underlying system defects (e.g., software 'bugs', design flaws, etc.) that create the propagation mechanism for instability.
•These system defects are generally not known to the network providers, and new defects are constantly introduced.
•These points of vulnerability can be easily triggered through malicious attack!

2. Poison Message Failure Propagation

•Several propagation mechanisms can cause instability.
•One such is "poison message failure propagation".
•Some or all of the network elements have a software or protocol 'bug' activated on receipt of particular network control/management message (the poison message). Activated 'bug' causes node to fail with some probability.
•If network control or management is such that this message is persistently passed among network nodes, and if node failure probability is sufficiently large, large-scale instability can result.
•Such an instability occurred in the AT&T switched network in 1990 and affected 98% of the 4ESS switches.

3. Exploitable Problem

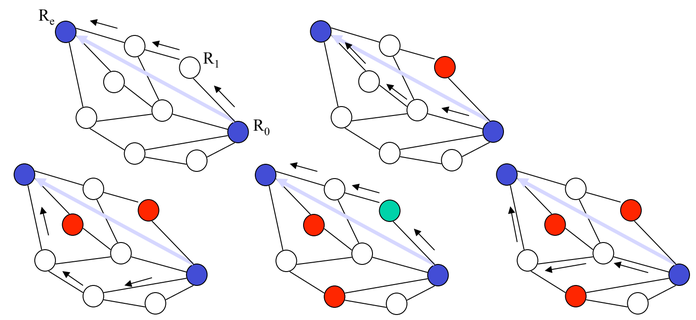
•A protocol may have a characteristic pattern of propagation.
- OSPF uses flooding.
- RSVP path messages follow a shortest path.
•Pattern recognition techniques can be used to infer protocol responsible for carrying poison message.
•Filtering may be used to isolate responsible protocol.

4. Passive and Active Diagnosis

• Combining passive diagnosis and active diagnosis. Centralized network management is used.
• Assume a **link box** can be put at each link of the network that can be configured to block certain message types or all messages belonging to a protocol. The link box can also record the recent communication history (messages exchanged).
• **Passive diagnosis**:- **Finite State Machine Method:** Analyze protocol events at failed node.
- **Correlating Messages:** Correlate protocol events across multiple failed nodes.
- **Using Node Failure Pattern:** Classify pattern of failure propagation by neural network .
• **Active diagnosis**:-Use protocol or message type filtering. Filtering can block propagation, but it can also disable important network control functions.
-Tradeoff analysis required. Filter configuration formulated as **sequential decision problem**.
• Passive diagnosis generates a probability distribution **P** over the suspect protocols, and this probability distribution is used in the sequential decision problem.
• If current passive and active diagnosis does not find the responsible protocol, new observations are used to update **P,** filter configurations are updated, and a new round of active diagnosis is started.

5. MPLS Example

• Ingress router R_0 tries to set up an LSP to egress router R_e.
• A poison label request message (LRM) is sent to next router R_1, and with some probability R_1 fails.
• If R_1 does not fail, then R_1 sends poison LRM to next router R_2, and the process repeats at R_2.
• If any node along the LSP fails, the LSP set-up fails and R_0 tries to find another path to R_e, and the process continues.
• Failed node recovery may initiate LSP reroutes, and rolling instability results.



• Blue: edge router; Red: failed router; Green: rebooted router.

6. OPNET Simulation

• To validate this approach, we have implemented OPNET testbed to simulate an MPLS network.
• Poison messages can be carried by OSPF, BGP or LDP (Label Distribution Protocol).
• Passive diagnosis (FSM method and message correlation method) has been implemented.
• The results indicate that passive diagnosis can provide a good probability distribution about the poison message.

7. Current Research

* Set up OPNET simulations for active diagnosis, e.g., configure filters to block certain message types or protocols.
* Implement neural network for node failure pattern recognition. * Find near optimal solutions for sequential decision problem.

Research Supported by DARPA-ITO Fault Tolerant Networks Program, Laboratory for Telecommunications Sciences and ARDA