Royal Holloway
University of London

Semi-Autonomous Link Layer Vulnerability Discovery and Mitigation Dissemination

Ziyad Al-Salloum and Stephen D. Wolthusen, PhD
Information Security Group

15 Sept. 2009

PROBLEM

Risk and vulnerability management is a critical task in maintaining any nontrivial network, but made increasingly difficult due to:

- The dynamic nature of inter-networking.
- The transient connectivity
- The use of virtual machines that are connected intermittently.

Vulnerability Announcement and Worm Appearance

Name	Vulnerability Announced	Worm Found	Interval
SQLSnake	Nov. 27, 2001	May 22, 2002	176
Code Red	June 19, 2001	July 19, 2001	30
Nimda	May 15, 2001	Sept. 18, 2001	126
•	August 6, 2001		42
	April 3, 2001		168
Slapper	July 30, 2002	Sept. 14, 2002	45
Zotob	August 9, 2005	August 16, 2005	7

Why a self-replicating approach

Self-replication approaches provide:

- Short probing distance.
- Ability to detect intermittent nodes.
- Traversing the network without high regard to network architecture.
- Distribute the workload among targets within the network.

RELATED WORK

- Random scanning as found e.g. in the Code Red I and Slammer (Moore *et al*).
- Flash worms all vulnerable nodes are already known (Staniford *et al*).
- Hitlist worm propagate to some known vulnerable nodes before switching to random scanning (Staniford *et al*).
- SQLSnake use encoded numbers to generate network space likely to contain vulnerable nodes (Nazario)

Related Work Cont.

- Routing worm which uses information provided by Border Gateway Protocol to determine target regions (Zou *et al*).
- Divide and Conquer scanning
  strategy passes half of the scanning space to the target and continues scanning the other half of its original space (Zou *et al*).
- Island Hopping
  (Code Red II) hosts closer to the infected target are scanned with higher probability than those farther away (Nazario)

Semi-autonomous link layer vulnerability discovery and mitigation dissemination

An agent -based (vulnerability) detection mechanism using semi-autonomous propagation strategies similar to those found in worms (self-replication), but which utilizes information from the link layer (layer 2 in the OSI model) to reconstruct topology information found through the Link Layer Discovery Protocol to detect neighboring nodes and propagate gradually until total coverage of an enterprise network is reached.

LLDP

Link Layer Discovery Protocol (LLDP) is a "media independent protocol intended to be run on all IEEE 802 LAN stations and to allow an LLDP agent to learn the connectivity and management information from adjacent stations." (*IEEE Std 802.1 AB 2005*)

PROPAGATION ALGORITHM

Assuming a node that have received the agent from another node as *nreceiver* and the node that sent the agent as *nsender*. The algorithm can be summarized as follows:

1. Install agent at the starting host *n0* in a subnet.
2. Unless *nreceiver* = *n0*, if *nreceiver* is already probed then stop, else continue.
3. Agent reads *nreceiver* LLDP Management Information Base objects to extract adjacent neighbors *A*.

PROPAGATION ALGORITHM

4. Agent then reads *nsource* LLDP MIB objects to extract adjacent neighbors *B*

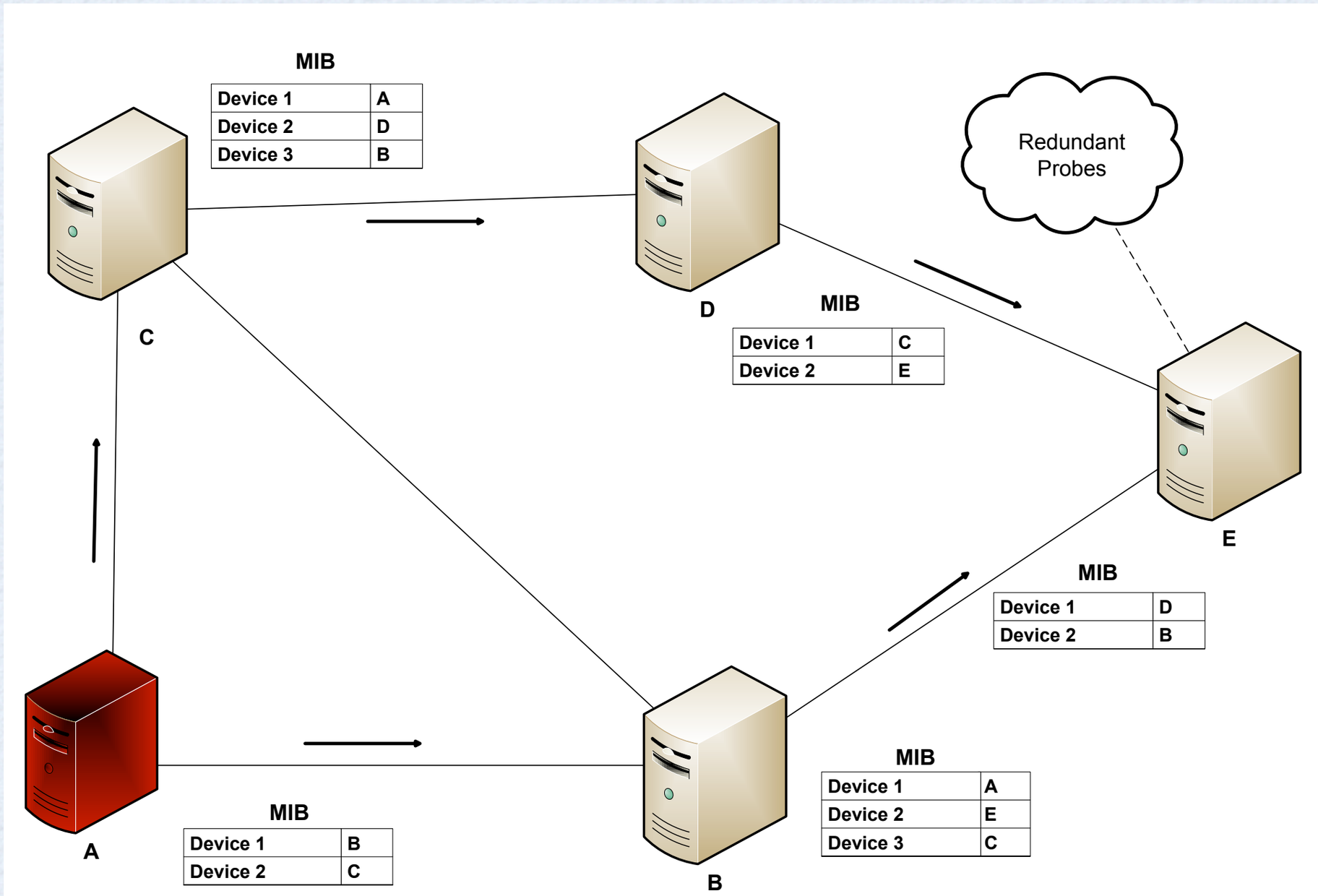
5. Remove *nreceiver* and *nsender* from the lists and compare them and self-replicate to non common neighbors, that is $\{A \setminus B\} - \{nsource\}$

6. Go to step 2

DESIGN COMPONENTS

- *Reconnaissance*: Hosts are discovered by looking up the neighboring nodes stored in the LLDP database.
- *Probe Component*: All hosts are vulnerable and it requires only one packet of the size 900 bytes to exploit.
- *Communication*: The scanning mechanism doesn't allow communication between agents at this stage, else for detecting redundant probes.

MECHANISM


MIB
Device 1 A
Device 2 D
Device 3 B
C
MIB
Device 1 B
Device 2 C
A
D
MIB
Device 1 C
Device 2 E
Redundant Probes
E
MIB
Device 1 D
Device 2 B
MIB
Device 1 A
Device 2 E
Device 3 C
B

NETWORK TOPOLOGY

- Hierarchical structure as is typically found in structured (enterprise) networks (produced using Tran sit-Stub model).
- Initially constructing a connected random graph then each node is replaced by another randomly connected graph representing the backbone of the network.

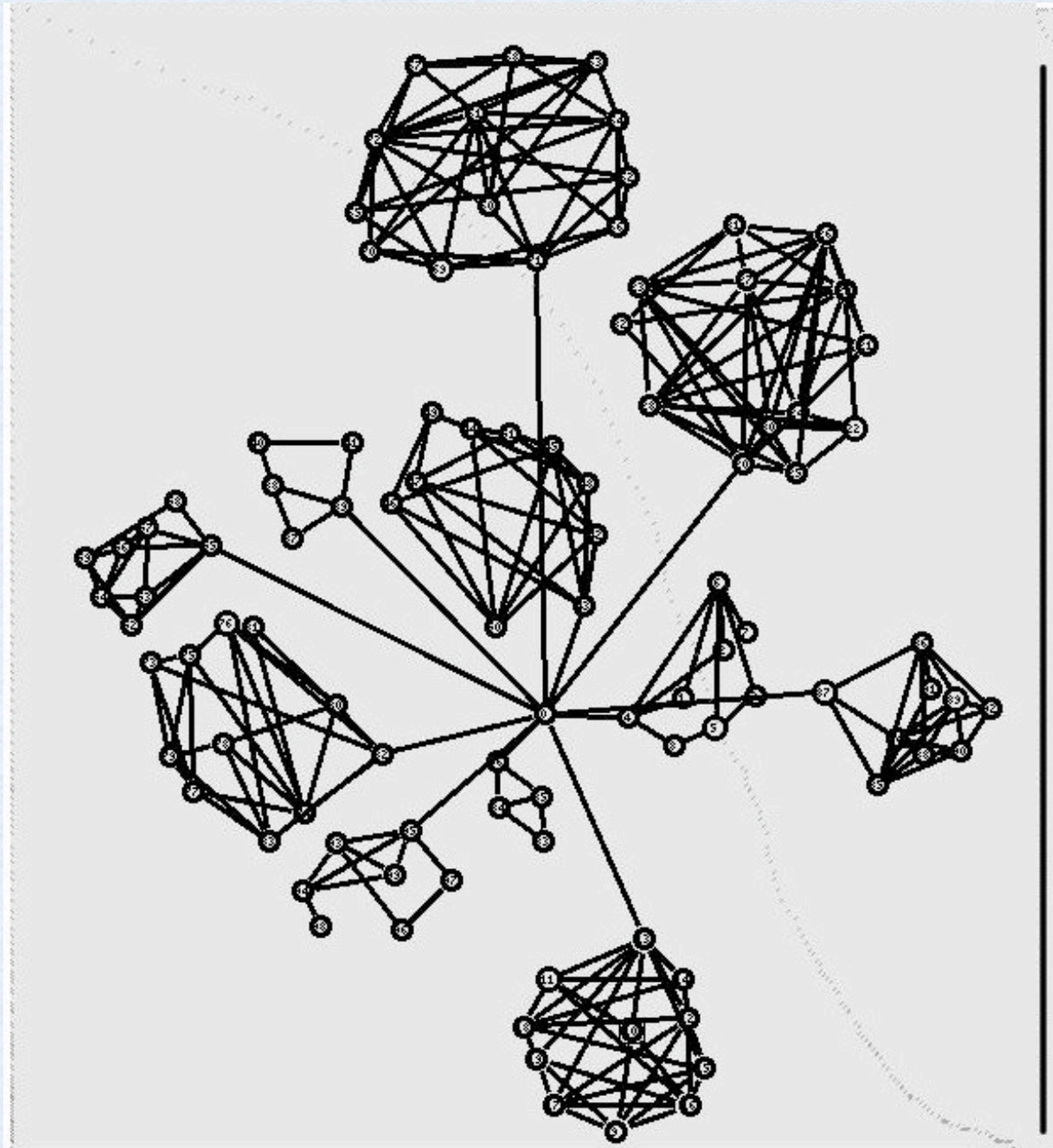
NETWORK TOPOLOGY CONT.

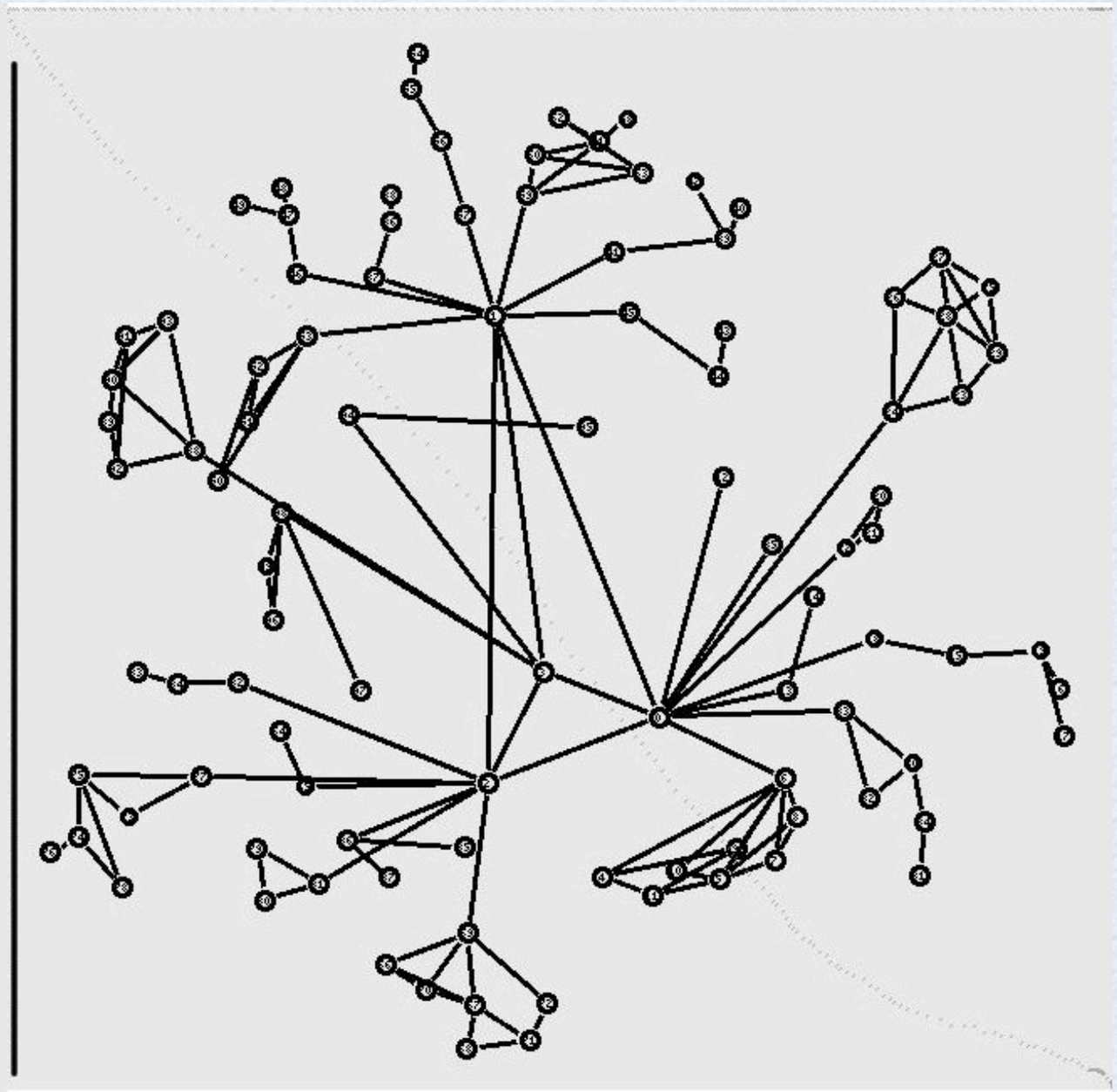
- Each node in the backbone is then replaced by a randomly connected graph to represent a LAN connected to a backbone node.
- Edges are then added with edge probability 0.5 within LANs and edge probability 0.8 between backbone nodes.
- Five different topologies were generated.

NETWORK TOPOLOGY CONT.

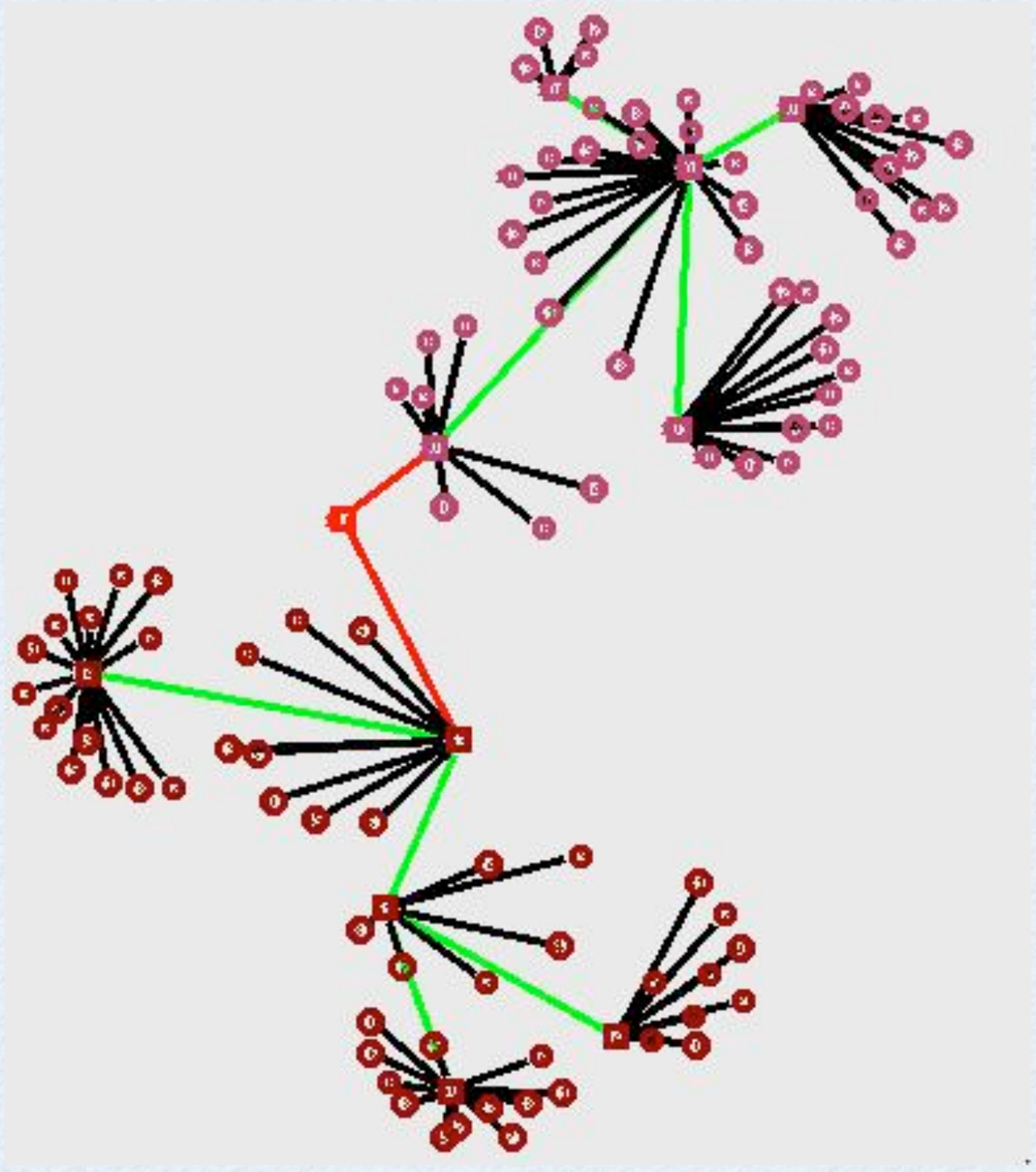
- For topologies that support STP, network nodes were picked randomly to be linked to a randomly chosen switch under a specified probability. Then is linked to the router in the backbone; such topologies are loop-free by definition.
- The Drop Tail queue management algorithm has been used as a queuing algorithm.
- Nodes are connected via a duplex-link where packets can flow in both directions.

NETWORK TOPOLOGY EXAMPLE





EXAMPLE STP SUPPORTIVE



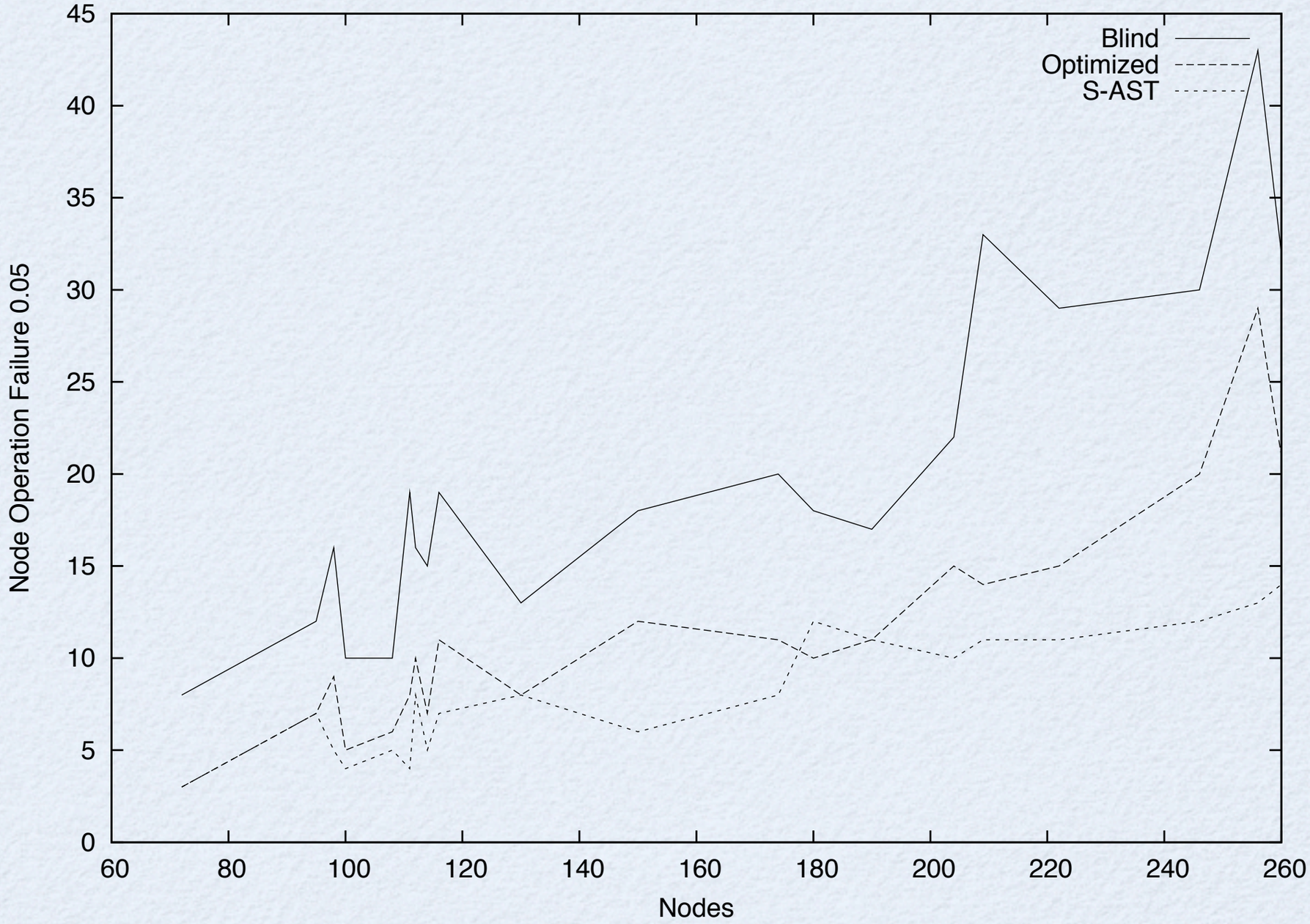
SIMULATION

- All simulations have been performed using the Network Simulator 2 (NS-2) a discrete event simulator mainly used for research activities.
- In total, there were 300 hierarchical network simulations, grouped into 5 groups each group consisted of different quantity of nodes that varied from 72 to 260.

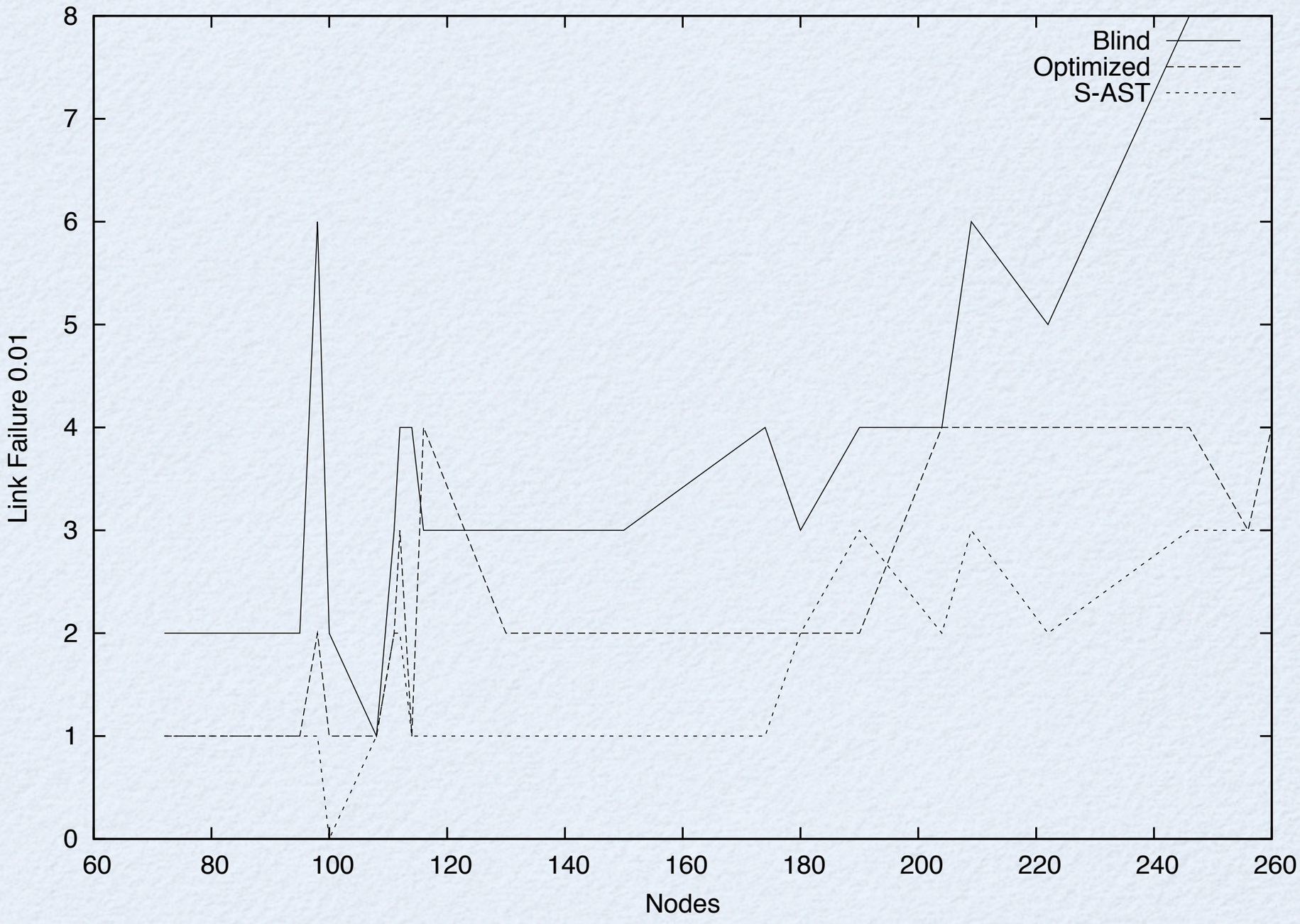
SIMULATION GATHERED

- The number of link failures under the probability of 0.01.
- The number of node operations failures under the probability of 0.05. Node operation failures are failures caused by the node it self (e.g. system is busy or in different state due to restating).
- The number of redundant probes issued by each mechanism. Redundant probes are probes received by a node more than once.
- The actual number of missed nodes during the process of vulnerability discovery.

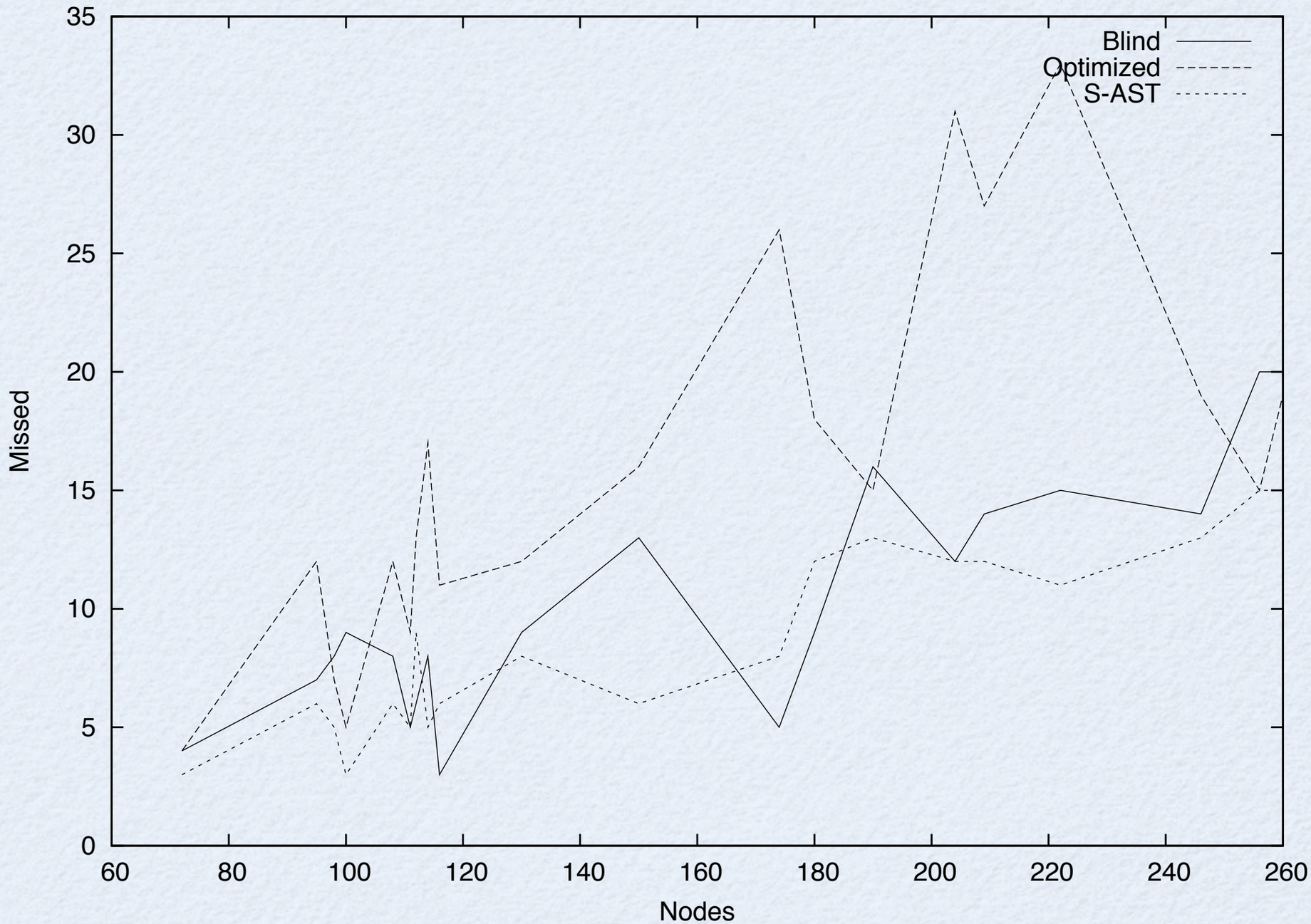
NODE OPERATION FAILURE PROBABILITY



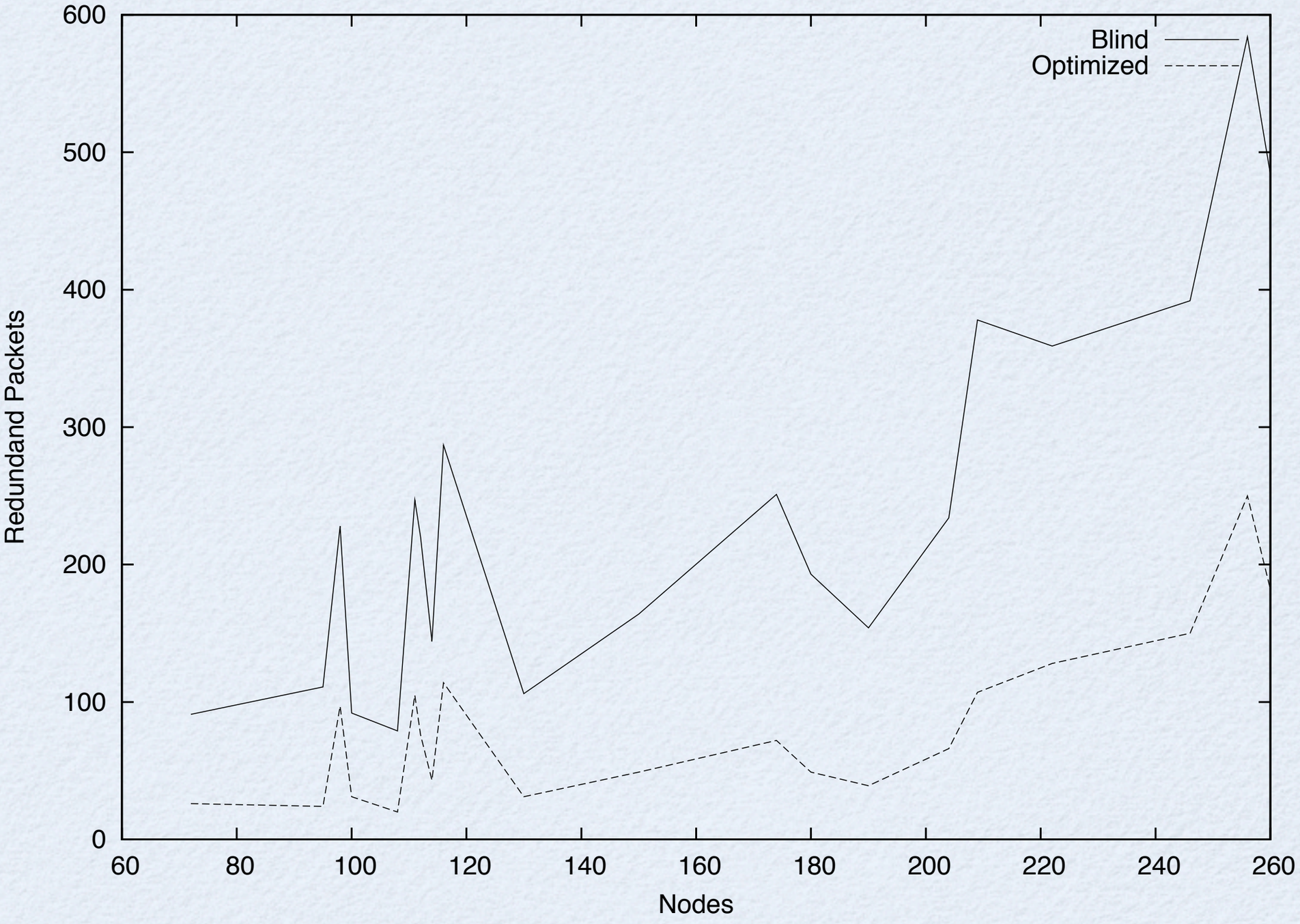
LINK FAILURE PROBABILITY



ACTUAL NUMBER OF MISSED NODES



REDUNDANT PROBES



Self-replication approach sensitivity to network topology

Blind vulnerability discovery simulation resulted in 63 redundant probes on the 100 nodes topology. The second simulation on a topology with the same number of nodes have resulted in 277 redundant probes. Running our mechanism on the same topologies have resulted in 20 and 116 redundant probes respectively.

VULNERABILITY MITIGATION

One packet with payload to achieve three tasks.

- First, exploit the vulnerability to gain the necessary privilege to apply temporally remediation.
- Second, apply vulnerability remediation to eliminate the security exposure of the vulnerable machine.
    - Disabling a port, installing a wrapper script, or uninstalling the vulnerable application.
- Third, trigger the agent for further propagation to cover other vulnerable nodes.

RISKS AND THREATS

- Compromise of the LLDP database stored in a system.
    - Hide the compromised node by stopping the node from advertising its identity.
- Compromise the mechanism agent.
    - Stop the mechanism propagation by deleting the LLDP database providing no further hosts to scan.

THANK YOU

- Questions