

**coverage:**

# A Cooperative Immunization System for an Untrusting Internet

Kostas Anagnostakis, Michael Greenwald

University of Pennsylvania

Angelos Keromytis

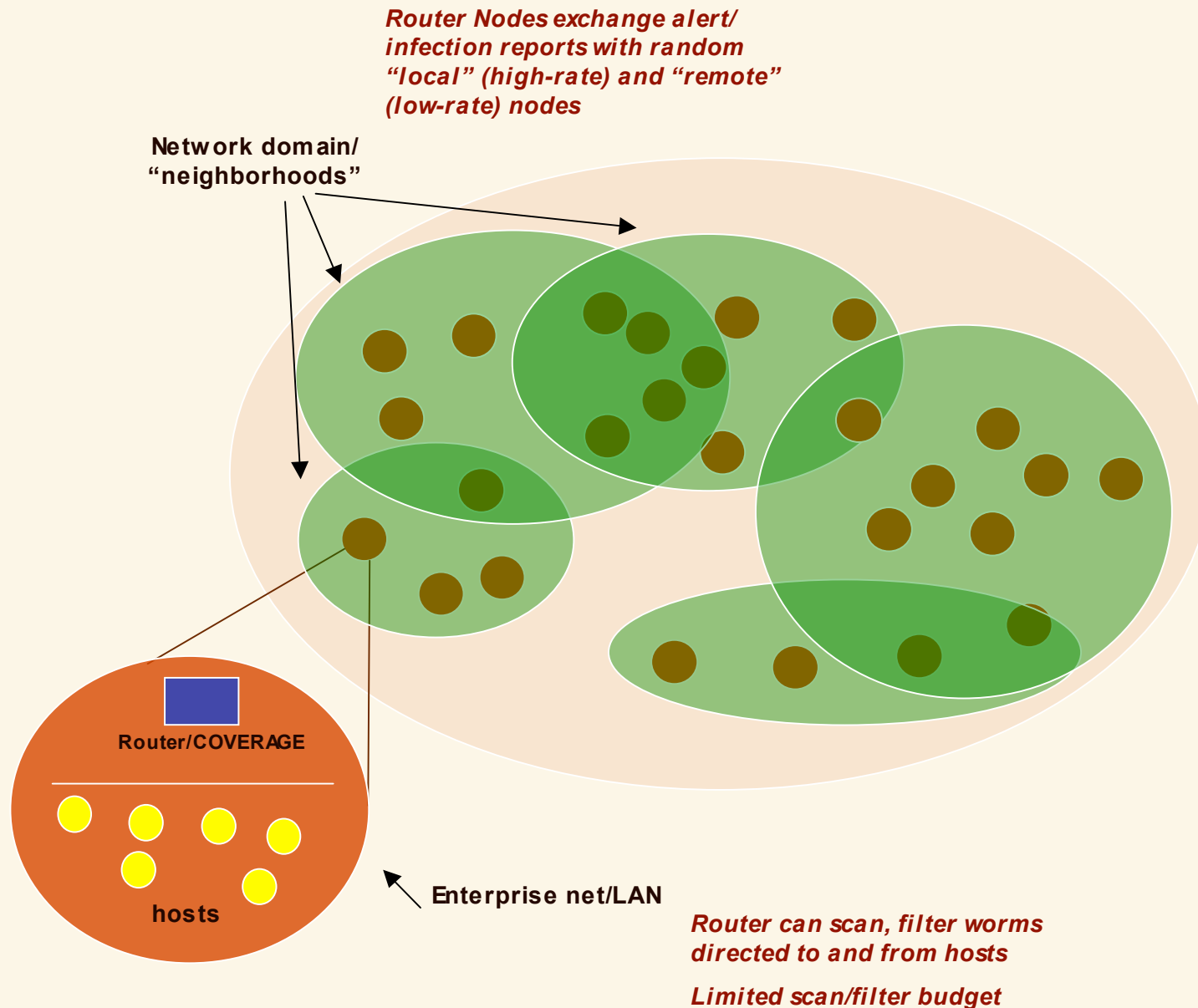
Columbia University

Partially supported by CIP/SW URI TIME DC  
<http://www.cis.upenn.edu/~timedc/>




# Overview

- **Goal:** detect and react to large-scale worm attacks
  - All sorts of worms including "day-zero"
  - Minimize infection and cost of false alarms
- **Challenges:**
  - **Detection uncertainty:** false positives vs. false negatives
  - **Reaction cost:** quarantine infrastructure, service disruption
  - **Limited resources:** detection & quarantine
- **Basic idea:**
  - Cooperative distributed system of sensors, sharing alert and infection information
  - Distributed algorithm: "**coverage**"
  - Emphasis in dealing with **faulty** and **malicious** sensors

# System model



# COVERAGE: communication

- Nodes periodically exchange reports:
  - Low-rate exchanges with random "remote" sensors
  - High-rate exchanges with "local" sensors
- Reports contain (for each worm):
  - Is node infected?  "local"
  - Is node scanning/filtering?  "direct"
  - # infected nodes seen  "remote"
  - # scanning nodes seen
  - # infected nodes others reported
  - # scanning nodes others reported
- Update local state based on reports:
  - Trust "pull" more than "push"

# COVERAGE: estimation

- Virus model:

- At timesteps  $n, n+1, \dots$  :

$$p_n, p_{n+1} = p_n(1+a), p_{n+2} = p_n(1+a)^2, \dots$$

- Infection growth rate  $a$
- Fraction of infected hosts  $p$

- For each virus, **coverage** maintains history of:

- Fraction of scanning nodes  $N$
- Fraction of infected hosts  $p$

- History is used to compute:

- Infection growth  $a$  based on  $p$
- Virulence  $v$  based on  $p$  and  $a$

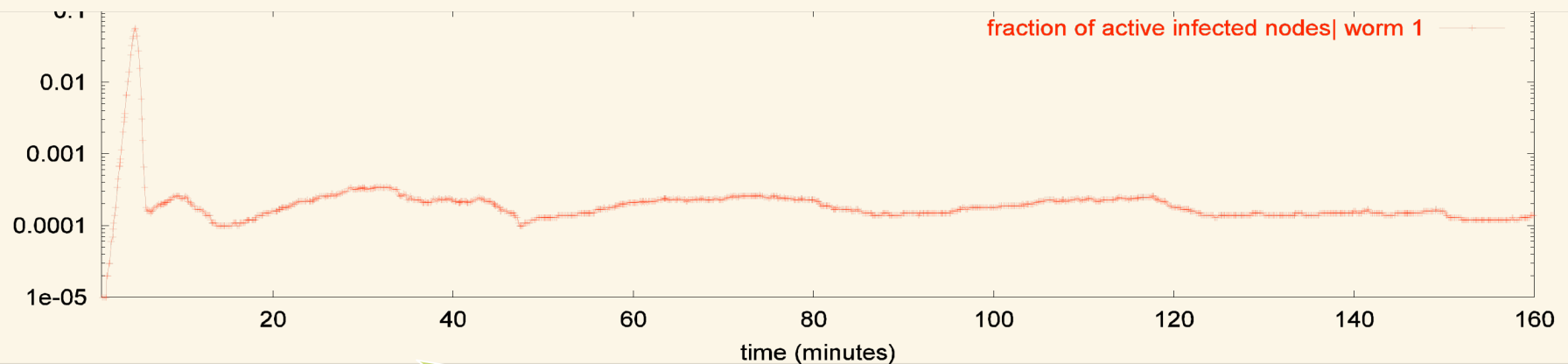
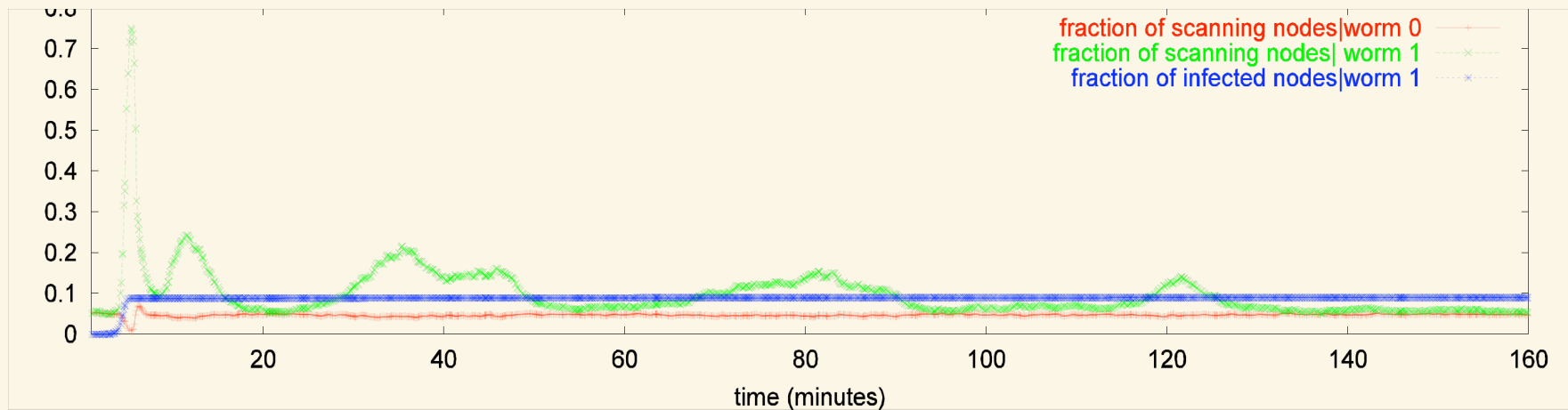
# COVERAGE: reaction

- In regular intervals, a node adapts its behavior as follows. For every virus, in decreasing order of virulence  $v$  :
  - Activate scanning/filtering if  $N < 2.p$   
Deactivate if  $N > p$   
Keep COVERAGE "ahead" of the infection  
Avoid overreacting to false or malicious alerts
  - Activate if  $N < \text{min}$  with probability  $(\text{min}-N)$   
Deactivate if  $N > \text{min}$  with probability  $(N-\text{min})$   
Always keep a minimum fraction of nodes active
  - Adapt remote polling rate to  $v$   
If there is some indication of a virus outbreak, but not sufficient evidence to start scanning, poll remote nodes more aggressively.

# Simulation study

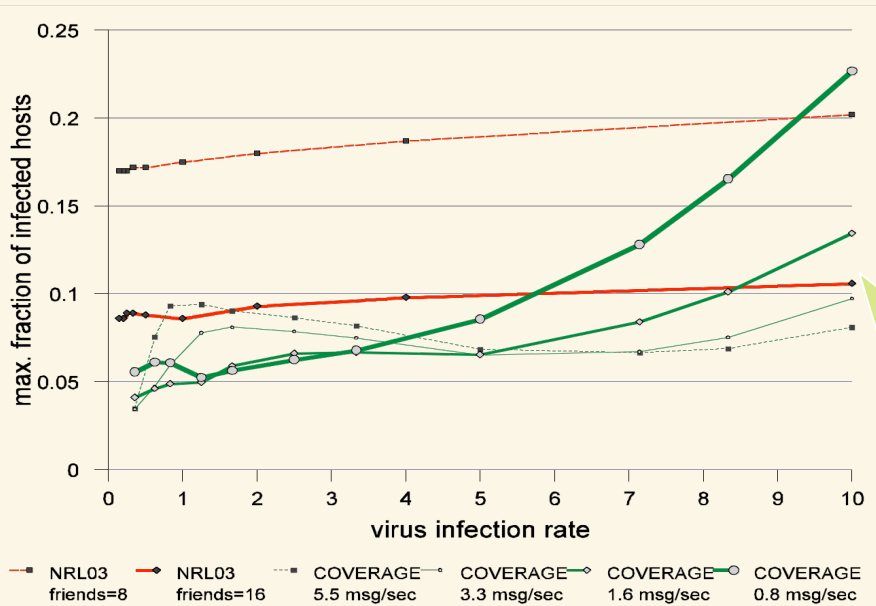
- Network model:
  - Full mesh network topology
  - 100,000 routers - 8 nodes on LAN
  - 2000 "domains" with 50 routers each
  - Single worm
- Metrics of interest:
  - Max. infection, false alarm cost, comm. cost
- Comparison with naive alert broadcast algorithm [NRL03]

# Example: reaction to worm attack



*COVERAGE reacts fast, even for highly virulent "Warhol"-type worms. Secondary outbreak attempts are likely, but can be successfully contained.*

# Worm response performance

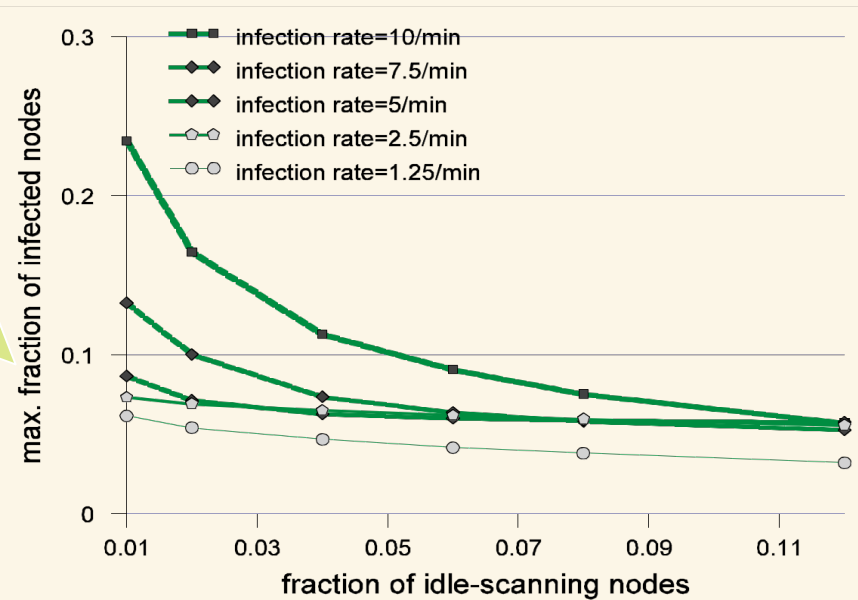


*Most worms can be contained to under 8% infection, but highly virulent worms may spread to a higher fraction of nodes. Need to increase communication rate to improve containment performance.*

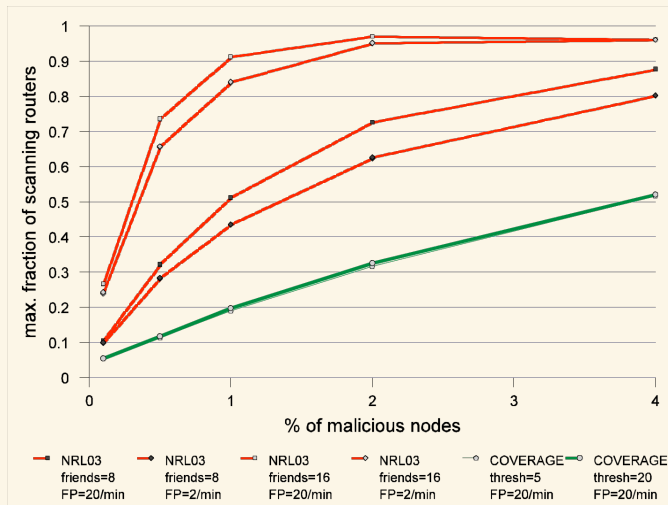
*Slower worms sometimes harder to contain especially as we increase communication cost - response is sometimes better when there are less accurate virulence estimates!*

*2-4% seems sufficient for most worms but very-high-virulence worms may need higher level of idle-scanning.*

*Min-level of scanning should be configured according to perceived risk*

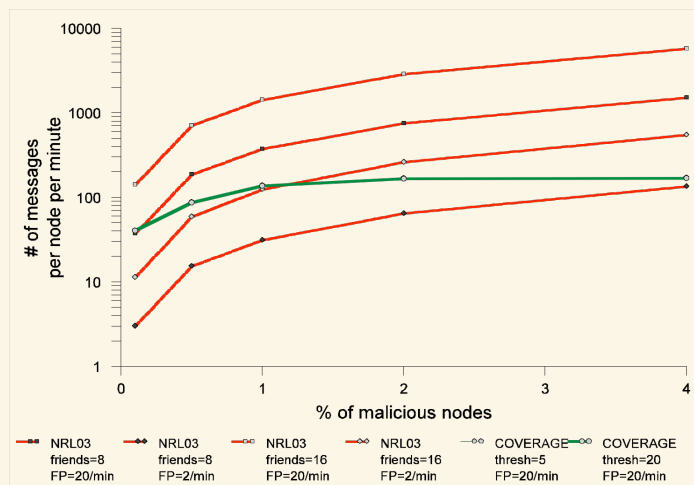
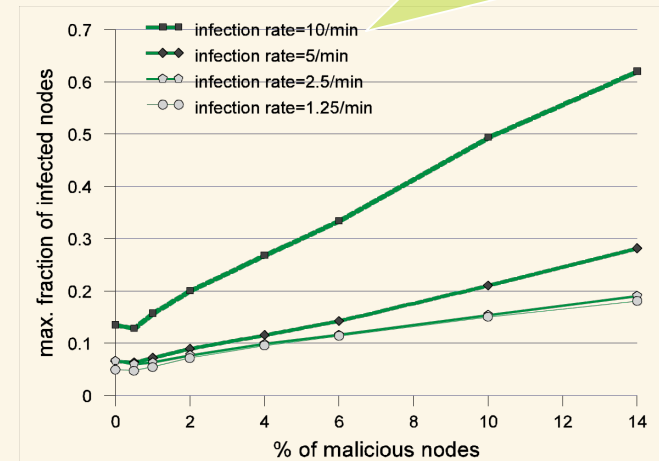


# Robustness to false alarms



*Naïve approach collapses for >1% malicious nodes – 90% of the nodes are scanning for the wrong worm!*

*Impact of false alarms reasonable for up to 3% malicious nodes and most worms, but VERY hard to deal with highly virulent worms as % of malicious nodes increases*



*COVERAGE communication cost peaks for 1% malicious nodes (unlike naïve approach where communication cost increases linearly with the % of malicious nodes \*and\* their alert generation rate)*

# Summary

- **COVERAGE** is a cooperative worm defense mechanism
  - Main design issue is to respond quickly to worm attacks and tolerate a fair number of faulty or malicious participants
  - Basic idea is to carefully sample global state to validate claims made by individual participants
- **Experimental results (preliminary):**
  - It is possible to detect and react to worm attacks while not over-reacting to false alarms
  - There is a three-way trade-off between communication, false scanning, and max. infection

# Ongoing work

- Detailed performance analysis:
  - More complex simulation scenarios with multiple worms, network vulnerability model, scanning vs. filtering, patching, etc.
- Tuning of algorithm:
  - Reduce communication cost
  - Pre-filter reports
  - Prelim. results show that better trade-off between reaction performance + resilience to attacks is possible
- Experiments with prototype (small-scale):
  - Use combination of signature-based IDS, anomaly detection schemes, honeypots, etc.