A Self-Learning Worm Using Importance Scanning

Zesheng Chen and Chuanyi Ji

Communication Networks and Machine Learning Group
School of Electrical and Computer Engineering
Georgia Institute of Technology, Atlanta, GA 30332

A Self-Learning Worm

Worm Scanning Methods

- Topological scanning
  - Rely on information contained in the victim host
  - Morris worm
- Random scanning
  - Select target IPv4 addresses at random
  - Code Red v2 and Slammer worms
- Localized scanning
  - Preferentially search for targets on "local" address space
  - Code Red II and Nimda worms

Advanced Worm Scanning Methods

- Hitlist scanning [SPW02]
  - Collect a list of vulnerable hosts in advance
  - Flash worm
- Routable scanning [WVGK04,ZTGC05]
  - Exploit the information provided by BGP routing table
- Importance scanning [CJ05]
  - Use the knowledge of vulnerable-host group distribution

<u>The use of additional information by an attacker can help a worm speed up the propagation</u>

Problem

- Information on vulnerable hosts may not be easy to collect before a worm is released
- What information can a worm learn?
- How to learn while a worm is propagating?
- How virulent is the resulting worm?
- How can we defend?

Outline

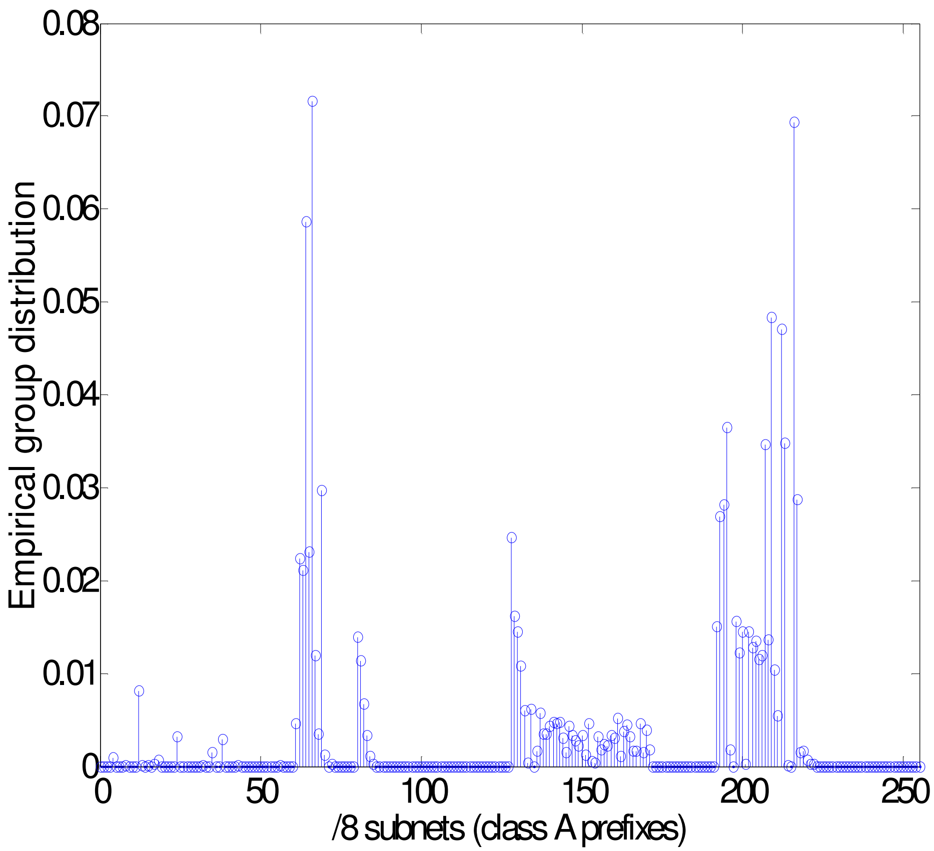
- Importance-scanning worm
  - Non-uniform vulnerable-host distribution
- A self-learning worm
  - Learning stage
  - Importance-scanning stage
- Performance evaluation
- Defense

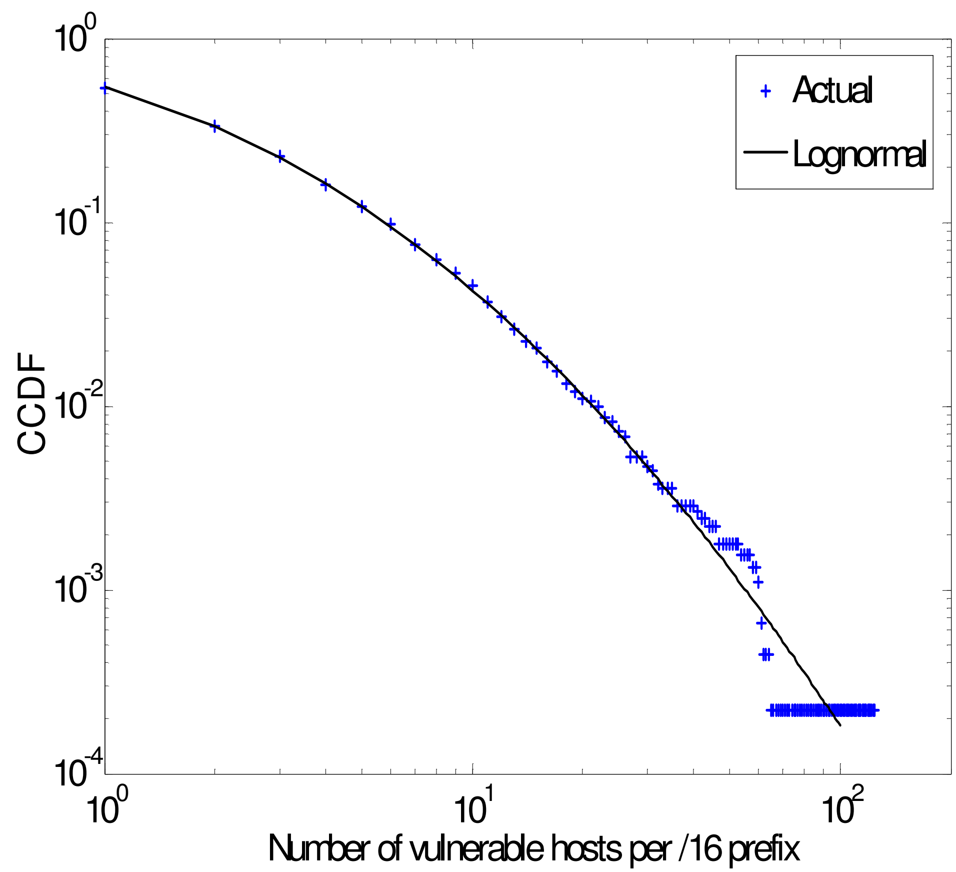
Importance Sampling

- Importance scanning is inspired by importance sampling in statistics
- Importance sampling is used to reduce the sample size for accurately estimating the probability of rare events
- Importance sampling biases the underlying sampling density

Key observation: *The vulnerable-host distribution is highly non-uniform*

Web-Server Distribution

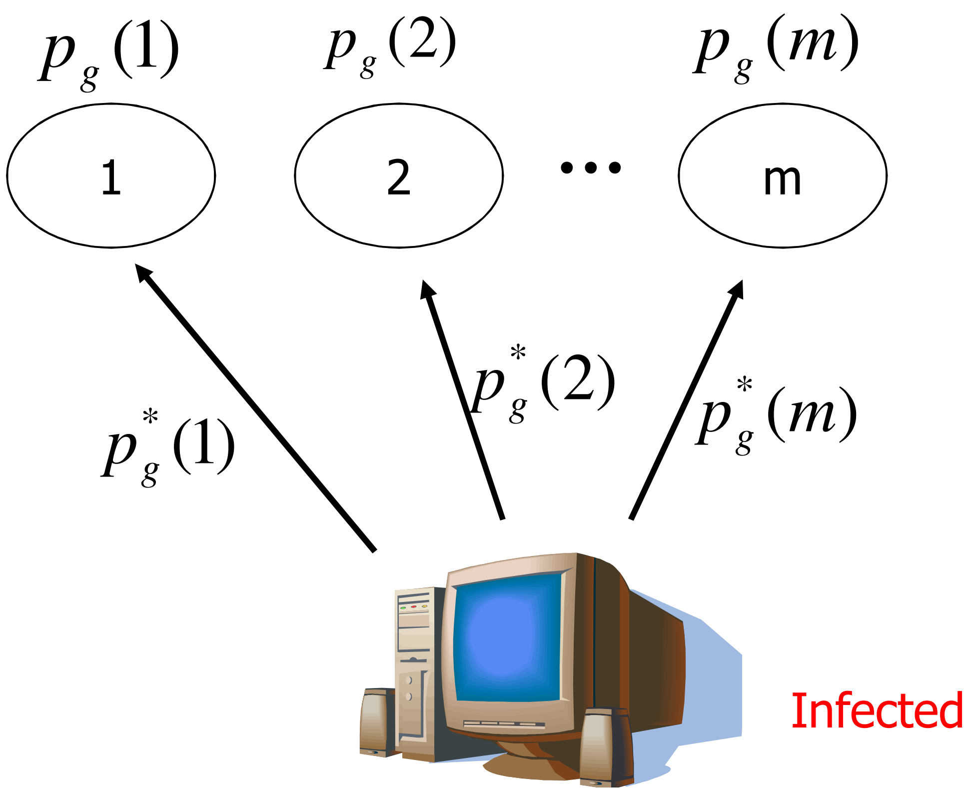




Importance Scanning

- Hitting a vulnerable host in a large population is a rare event
- Probing a target is equivalent to obtaining a sample in IP address space
- Sample the IP address space according to a given vulnerable-host distribution
- Reduce the number of scans needed for attacking a large number of vulnerable hosts

Importance-Scanning Worm



- $p_g(i)$: group distribution

$$p_g(i) = \frac{N_i}{N}$$

- $p_g^*(i)$: group scanning distribution

Importance Scanning

- Optimal dynamic importance scanning
  - $p_g^*(i)$'s vary with time → *not realistic*
  - All infected hosts scan the group containing the largest number of uninfected vulnerable hosts
  - Performance upper-bound for comparison
- Static importance scanning
  - $p_g^*(i)$'s are fixed at all time → *realistic*

Optimal Static Importance Scanning

- What are $p_g^*(i)$'s given $p_g(i)$'s
- A new metric: average number of worm scans required until the first scan hits a random-chosen vulnerable host
- Lagrangian optimization

$$\tilde{p}_g^*(i) = \frac{\sqrt{\Omega_i p_g(i)}}{\sum_{k=1}^{m} \sqrt{\Omega_k p_g(k)}}$$

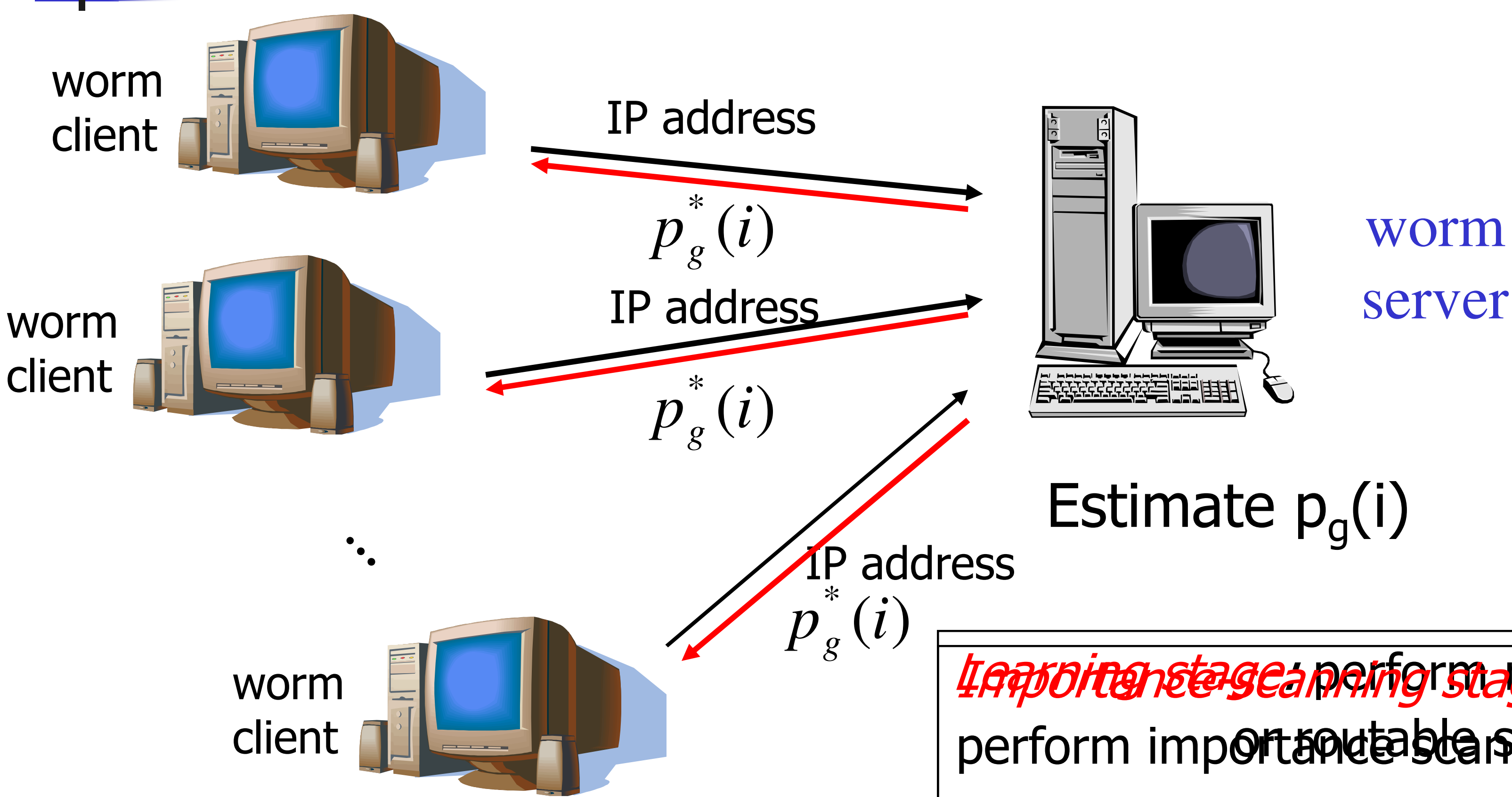
Outline

- Importance-scanning worm
  - Non-uniform vulnerable-host distribution
- A self-learning worm
  - Learning stage
  - Importance-scanning stage
- Performance evaluation
- Defense

A Self-Learning Worm

- Vulnerable-host distribution is unavailable before a worm is released
- Self-learn vulnerable-host distribution information while propagating
- *Learning* stage and *importance-scanning* stage

A Self-Learning Worm System



Learning stage: perform random or routable scanning

Importance-scanning stage: perform importance scanning

Estimating Group Distribution

$$\hat{p}_g(i) = \frac{L_i}{L}$$

- L: # of measurements (clients' IP addresses)
- L_i: # of clients' IP addresses from group i among all L addresses

- A simple proportion estimator
  - Unbiased
  - Maximum likelihood estimator
- Mean square error is bounded by $\frac{1}{L}$

Outline

- Importance-scanning worm
  - Non-uniform vulnerable-host distribution
- A self-learning worm
  - Learning stage
  - Importance-scanning stage
- Performance evaluation
- Defense

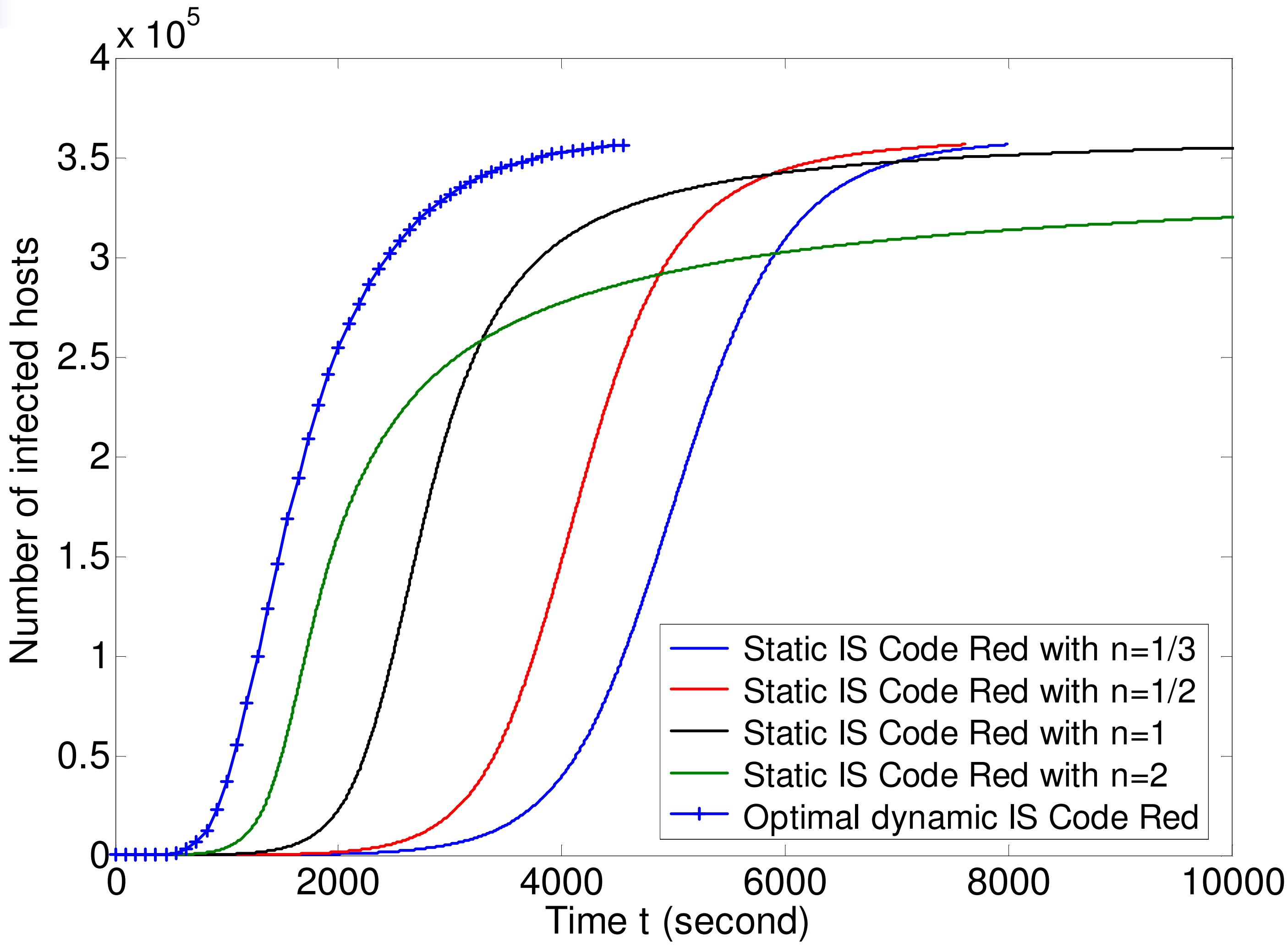
Performance Evaluation

- Parameters comparable to those of Code Red v2
  - N=360,000 and s=358 per minute
- Vulnerable host has the same distribution as web servers
- Extended Analytical Active Worm Propagation (AAWP) model for importance–scanning worms [CGK03,CJ05]

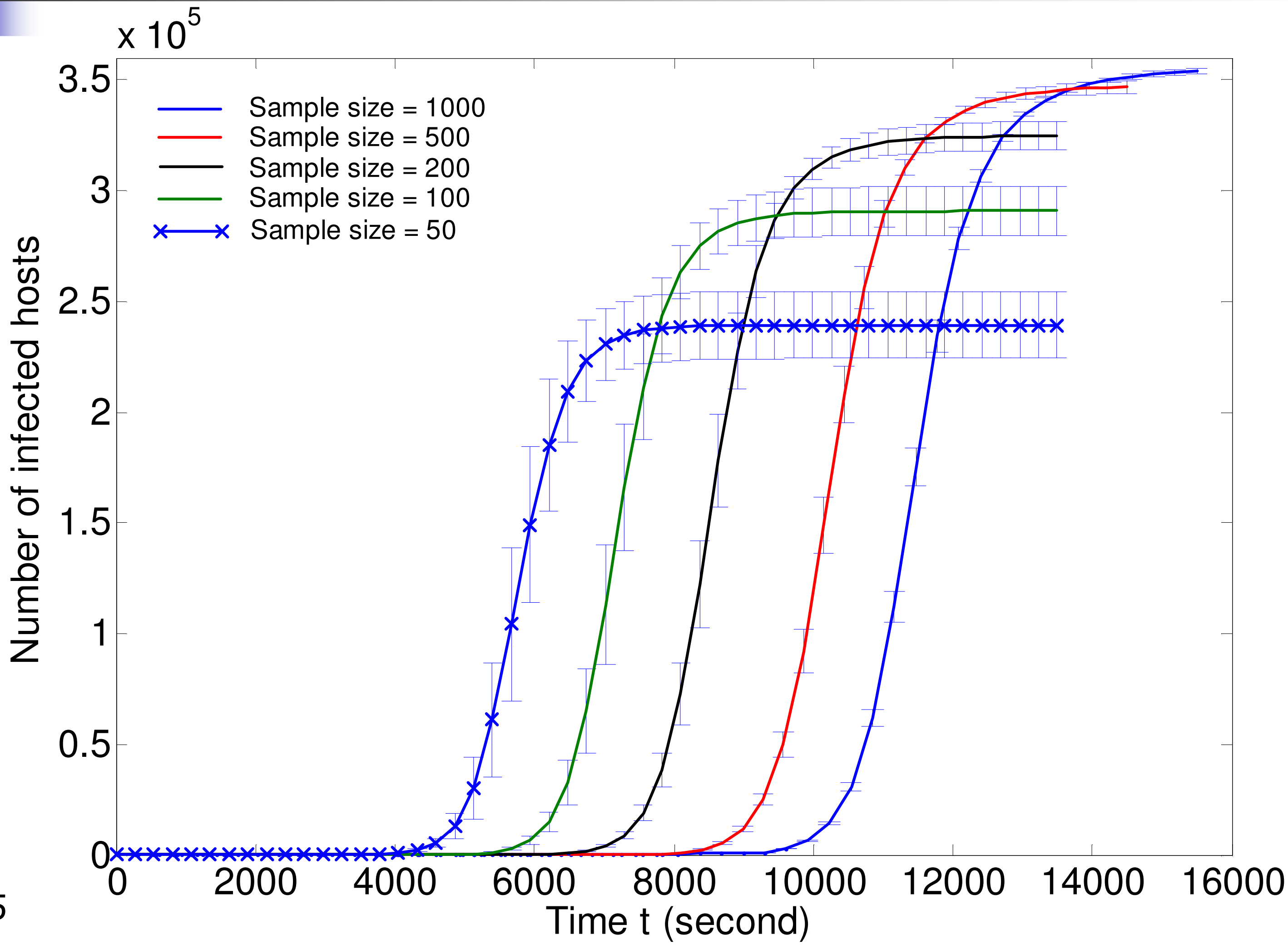
$$I_{t+1,i} = I_{t,i} + (N_i - I_{t,i})[1 - (1 - \frac{1}{\Omega_i})^{sI_t p_g^*(i)}]$$

$$p_g^*(i) = \frac{(p_g(i))^n}{\sum_{k=1}^{m} (p_g(k))^n} \propto (p_g(i))^n$$

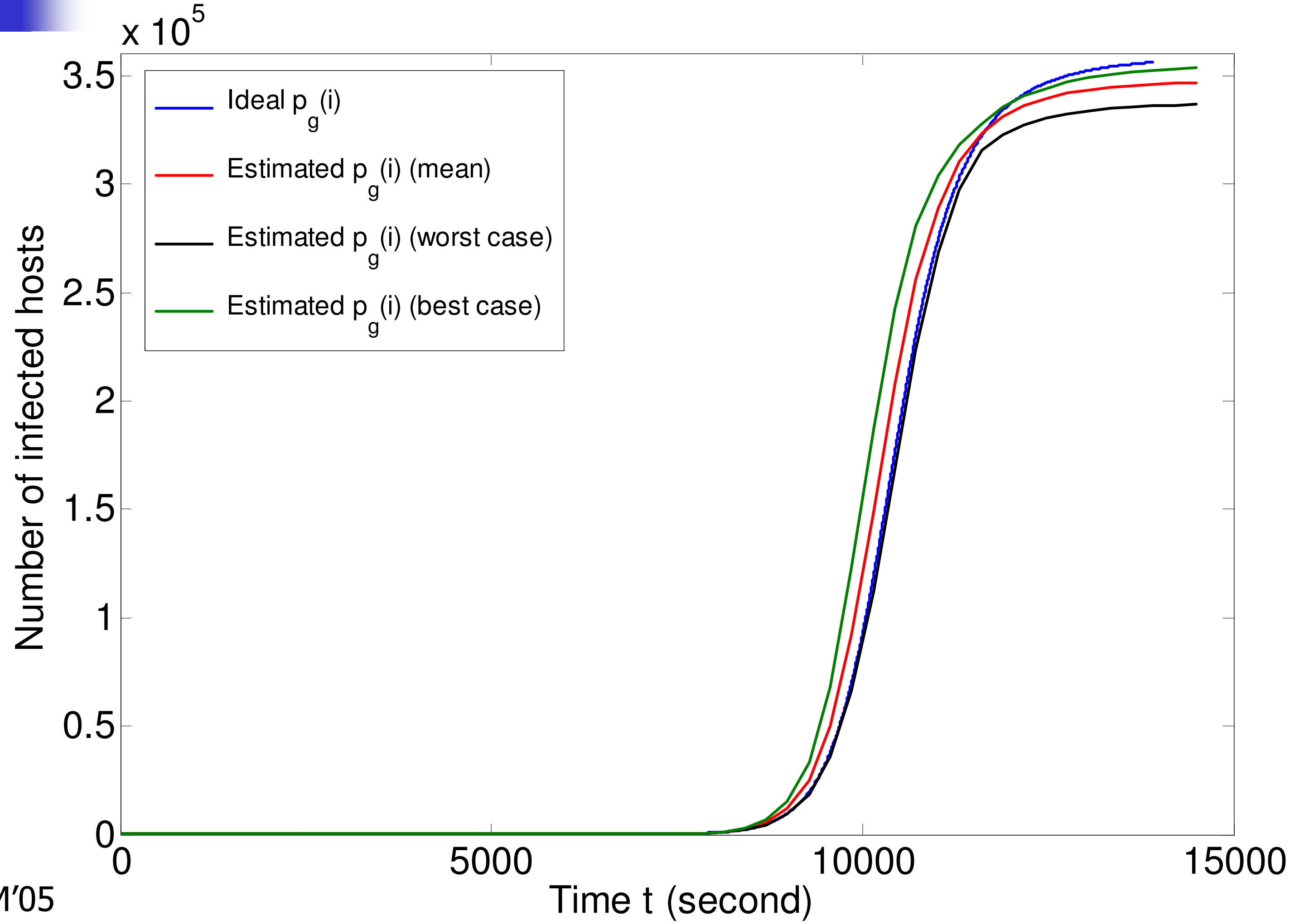
Static Importance Scanning Strategies



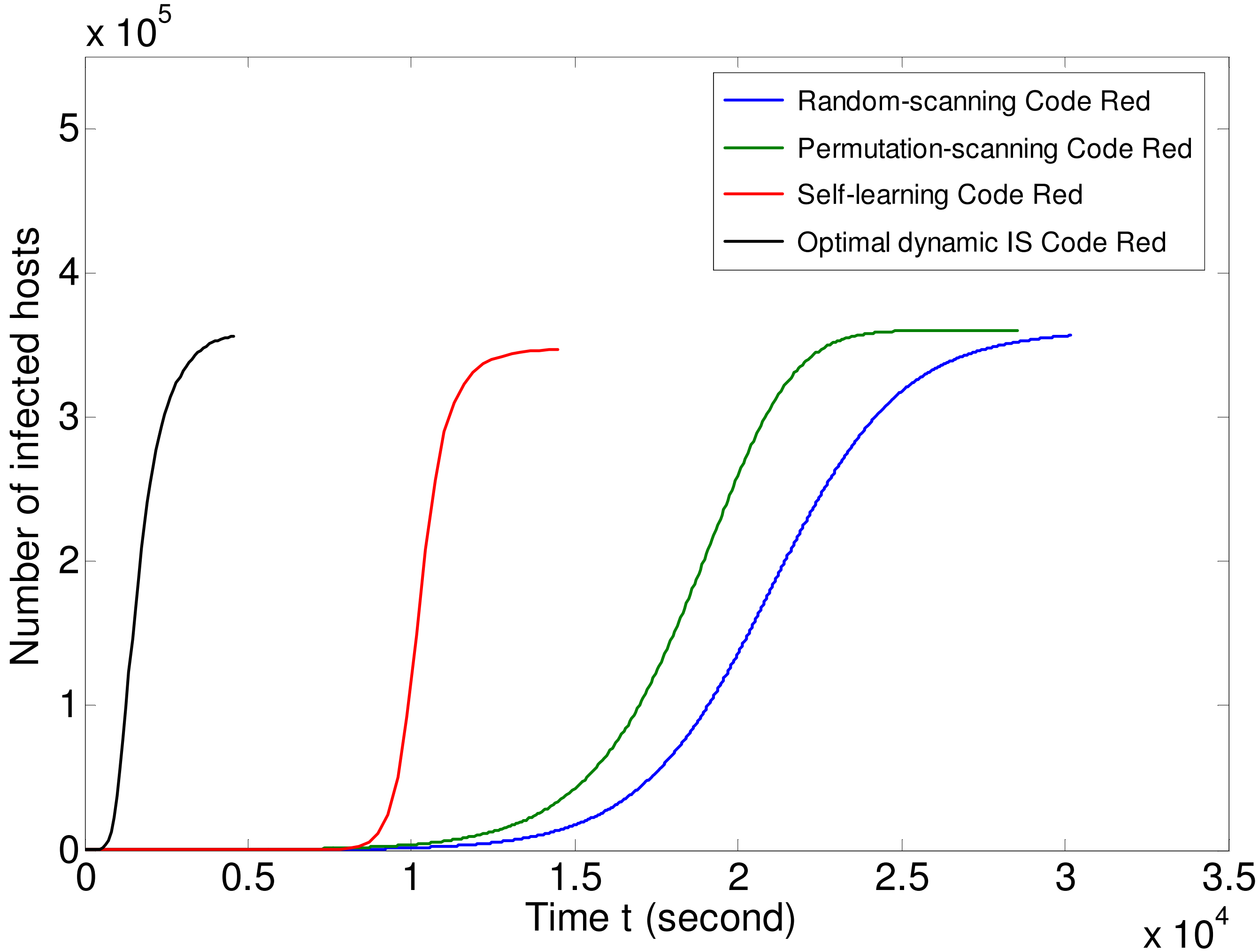
Sample Size (L)



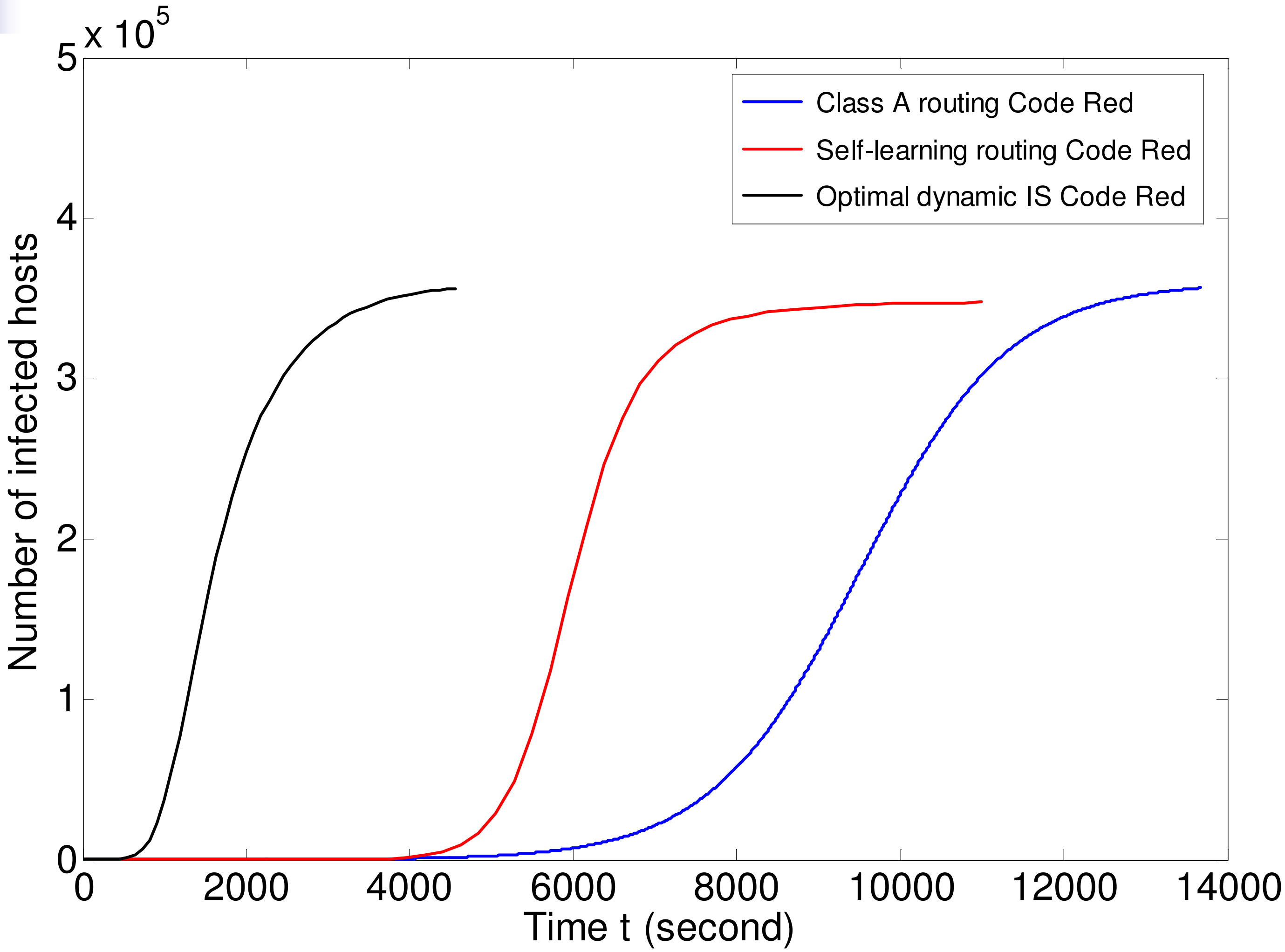
Sample Size of 500



Performance of Self-Learning Worm



Performance of Self-Learning Worm (2)



Outline

- Importance-scanning worm
  - Non-uniform vulnerable-host distribution
- A self-learning worm
  - Learning stage
  - Importance-scanning stage
- Performance evaluation
- Defense

Defending Against Self-Learning Worms

- Attackers control $p_g^*(i)$
- Defenders customize $p_g(i)$
- A game exists between attackers and defenders
- It shows that best strategy for defenders is to scatter applications uniformly in the entire IP-address space from the view of game theory

Conclusions

- A self-learning worm
  - *Learning stage:* Learn /8 subnet distribution well using a proportion estimator and as few as 500 samples
  - *Importance-scanning stage:* Use optimal static importance scanning method
- Game between attackers and defenders
  - Applications need to be uniformly distributed in the whole IPv4 address space

Reference

- [SPW02] S. Staniford, V. Paxson, and N. Weaver, "How to 0wn the Internet in Your Spare Time," in *Proc. of the 11th USENIX Security Symposium (Security '02),* 2002.
- [WVGK04] J. Wu, S. Vangala, L. Gao, and K. Kwiat, "An Effective Architecture and Algorithm for Detecting Worms with Various Scan Techniques," in *Network and Distributed System Security Symposium*, 2004.
- [ZTGC05] C. C. Zou, D. Towsley, W. Gong, and S. Cai, "Routing Worm: A Fast, Selective Attack Worm based on IP Address Information," *19th ACM/IEEE/SCS Workshop on Principles of Advanced and Distributed Simulation (PADS'05),* 2005.
- [CJ05] Z. Chen and C. Ji, "Importance-Scanning Worm Using Vulnerable-Host Distribution," in *Proc. of IEEE Globecom 2005*, 2005.
- [CGK03] Z. Chen, L. Gao, and K. Kwiat, "Modeling the Spread of Active Worms," in *Proc. of INFOCOM 2003*, 2003.

Q/A