Worm Detection, Early Warning and Response Based on Local Victim Information

Guofei Gu, Monirul Sharif, Xinzhou Qin, David Dagon, Wenke Lee, and George Riley

Georgia Institute of Technology

Outline

- Worm Local Detection using DSC
- Worm early warning using local victim information
- Local Response based on local victim information
- Conclusion

Worm Local Detection Using DSC

WORM Common Characteristics 1

- Many worms generate a substantial volume of identical or similar traffic to their targets
  - □ detecting known worms using their signatures
  - □ only well-known worms whose signatures are acquired.
- Helpless for
  - □ zero-day worms whose signatures are not known yet
  - □ polymorphic worms that do not have common signatures
- Honeycomb,AutoGraph,EarlyBird…

WORM Common Characteristics 2

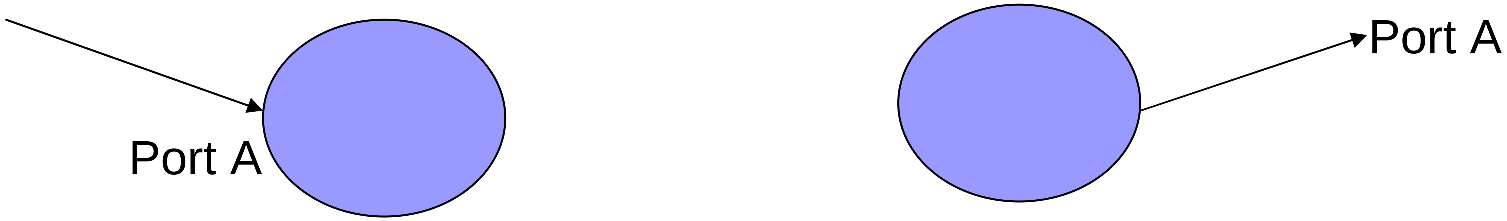
- Random scanning -> reach inactive IP addresses
- Observing abnormally quick increases in scans to inactive IP addresses
  - □ large monitored network (say, 2^{20} nodes)
  - □ Kalman filter, …
- But local networks find it more useful to know which machines are infected, and how the attack is progressing.

WORM Common Characteristics 2

- Random scan -> high failed connection ratio
- Observing abnormal failed connection ratio in local network
  - TRW,…
- Can hardly tell the difference of a worm from a scanner which also causes a high failed connection ratio.
- Cannot detect topological worms and flash worms that use lists of victim addresses.

WORM Common Characteristics 3

- Vulnerable hosts exhibit infection-like behavior when infected



Summary of Current Approaches

- “Symptom”-based
- Mainly depend on artifacts or “symptoms” of worm infections, i.e., based on detecting the scanning activity associated with scanning worms.

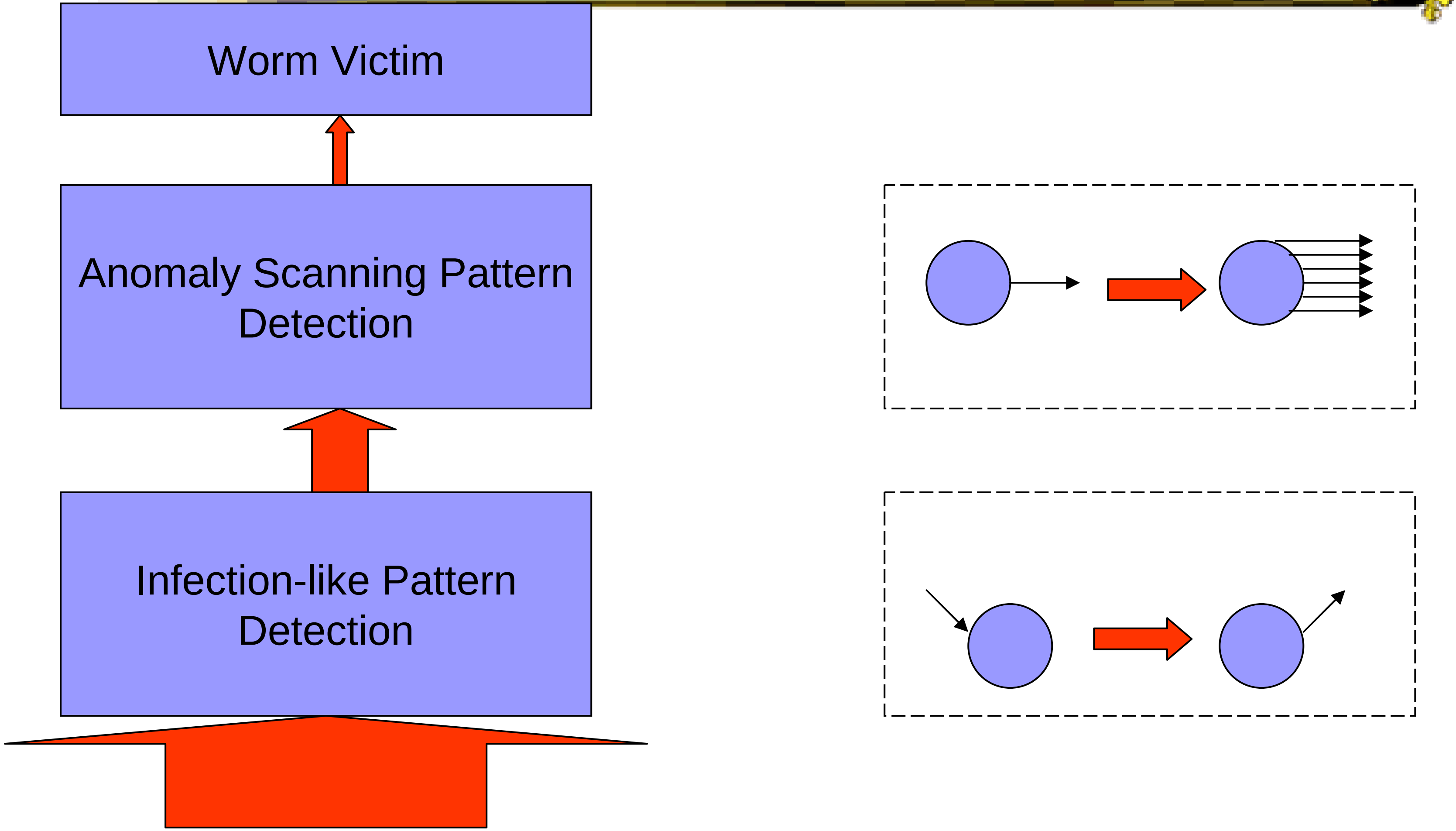
Shift Emphasis

- Global strategies -> Local detection (the benefit of local detection is clear)
    - Local networks may not share all necessary information to global strategies
        - Privacy, cost, sharing, security
    - Local networks can detect victims and respond much faster
    - The local victim information is more useful
- Symptoms of worm ->”Behavior” of worm
    - Consider both scanning pattern and infection pattern
        - our DSC (Destination-Source Correlation) algorithm

Basic Idea of DSC

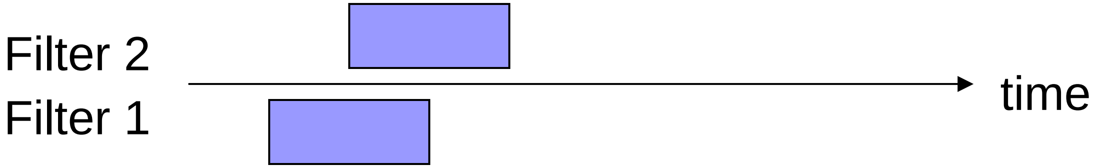
- Aims to detect scan-based, *fast* spreading worms.
- Two phases
  - □ Find infection-like pattern
  - □ Anomaly scan rate detection of systems with infection-like behavior
  - □ Local response (can be considered as third phase)

DSC-Two Phase Worm Detection


Worm Victim
Anomaly Scanning Pattern Detection
Infection-like Pattern Detection

DSC Implementation

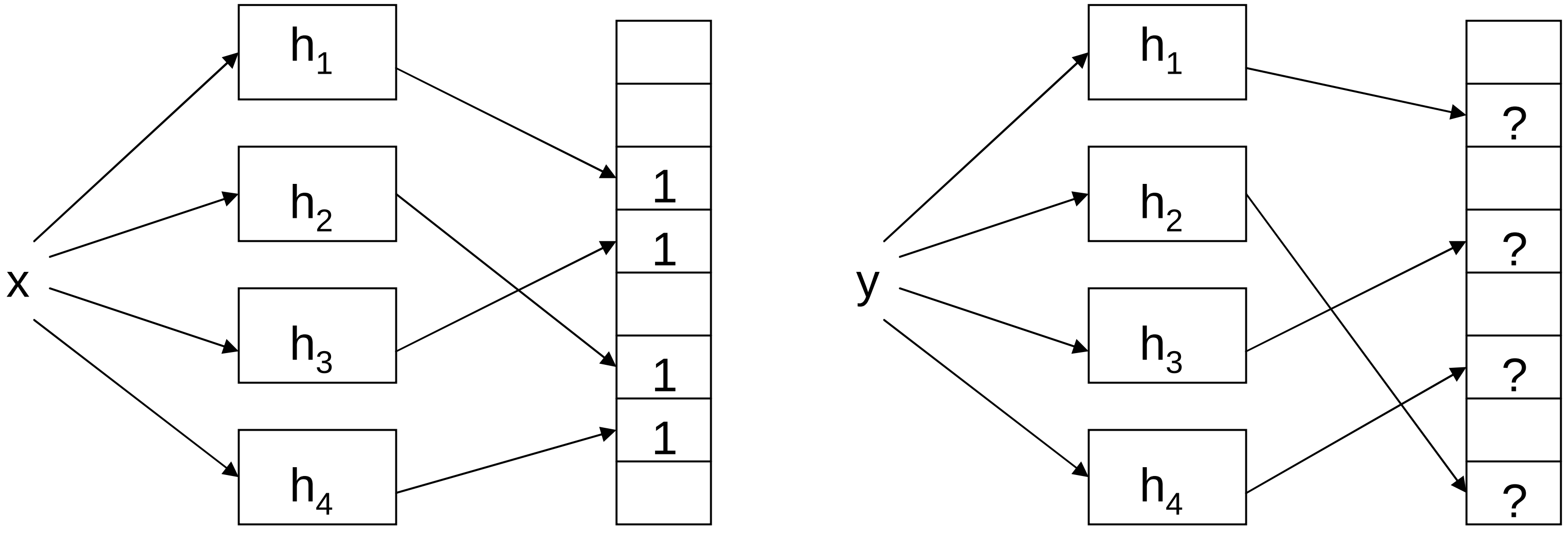
- Infection Discovery
  - Sliding window to record destination addresses. Check whether outgoing source in this window.
  - Bloom filter version: use two Bloom filters to roughly simulate a sliding window



- Anomaly Scanning Detection:
  - TRW can work. We use simple heuristics here
  - Normal profile
  - Chebyshev's inequality $p(|x-\mu| > t) < \frac{\sigma^2}{t^2}$

Bloom Filter

- Using Bloom filter to keep state and query.



Insert x into filter

Test is y in filter?

- Keep the history of destination to local addresses (for large and heavy-traffic network)

Actual Detection Results 1-WAND

- 65GB (compressed) trace sample (six and one half week trace between February and April 2001) at the University of Auckland
- Some representative TCP ports, i.e., 21, 22, 23, 25, 80, 139, 445 and UDP ports, i.e., 53, 1434.
- We did not find any *infection-like behaviors* on all selected ports except port 80
- 25 infection-like behaviors total on port 80 with very low scan rate: μ= 0.1, σ^2 = 0
- False positive rate: 0

Actual Detection Results 2 - GTTrace

- Infection-like behaviors # and normal scanning profile

Port	# of infection pattern	μ	σ^2
22	19	0.1263	0.002
23	8	0.3625	0.0512
80	451	0.7978	1.812
6884(BitTorrent)	8	1.9375	1.0455

- False positive rate=0
- Use DSC in honeynet data during the breakout of SQL Slammer, successfully identify all three victims. (TP=100%)
- We didn’t get real worm data to do further experiments.

Limitation of DSC

- Very slow worm
  - Combining failed connection check may help
- Some infection-like normal traffic like P2P
  - Blacklisting might help
  - Combining failed connection check may help
- Bipartite/dual worm: infect and propagate using different vectors

Worm Early Warning Using Local Victim Information

New Analytical Model

- To understand worm propagation behavior and evaluate detection and response strategies
- Existed models assume the vulnerable hosts uniformly distributed in the entire IPv4 addresses
  - The size of allocated IP space is only about 10^9 (about 1/4 of 2^{32})
- We use a more realistic assumption that vulnerable hosts are uniformly distributed in these parts of the address space
- Our simple discrete time-based model is based on adapted AAWP (Analytical Active Worm Propagation [Chen_InfoCom_2003])

Basic Model (Random Scan)

- T: victim space (all victims are located in this space)
- Ω: whole IPv4 space
- N: # of vulnerable hosts on Internet
- s: scan rate (per time tick)
- n_i is the number of infected hosts at time tick i

$$n_{i+1} = n_i + [N - n_i]\left(1 - (1 - \frac{1}{T})^{s n_i \frac{T}{\Omega}}\right)$$

Relation to Epidemic Model

- In fact because T is big, we have

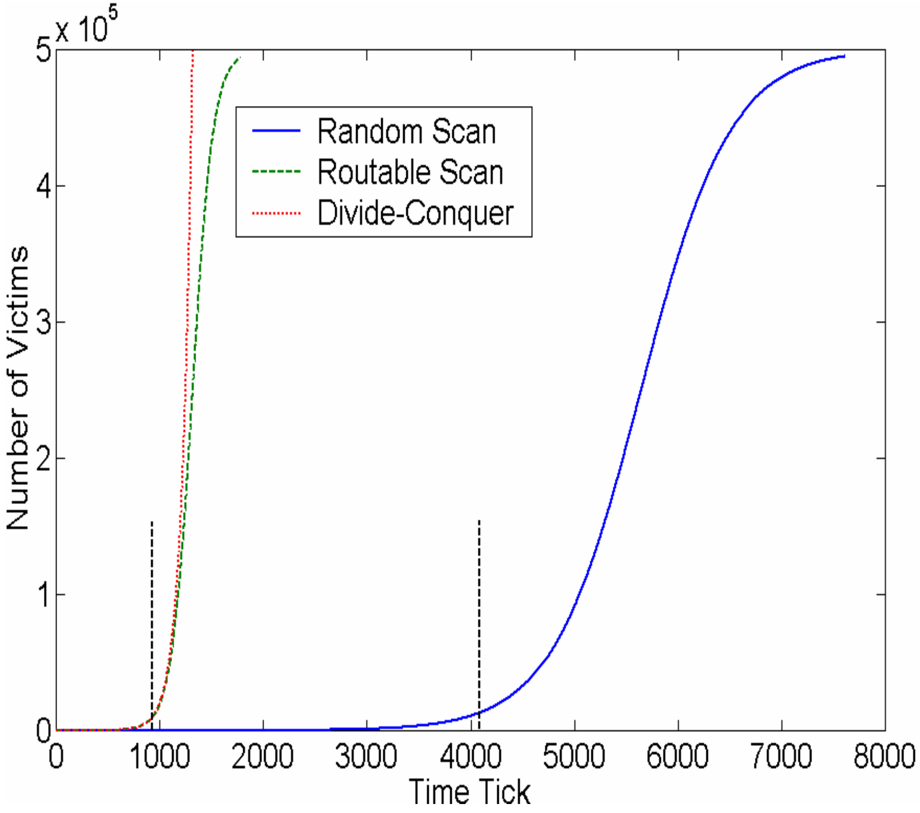
$$\frac{n_{i+\Delta t} - n_i}{\Delta t} = [N - n_i]\left(1 - (1 - \frac{1}{T})^{sn_i\frac{T}{\Omega}}\right) \doteq (N - n_i)\frac{1}{T}sn_i\frac{T}{\Omega}$$

- Let Δt→0, then we get the traditional epidemic model

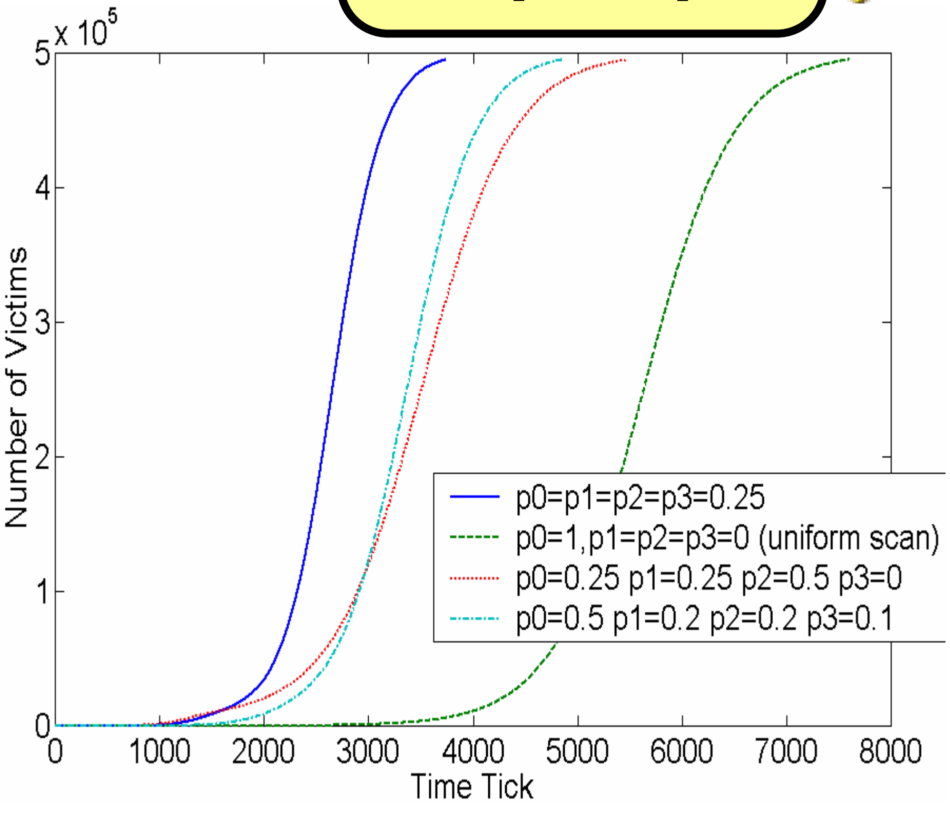
$$\frac{d(n_t)}{dt} = \frac{s}{\Omega}n_t(N - n_t)$$

Analytical Model

Local Preference is faster than random scan, faster than shown in [Chen03]



Three Uniform Scanning Worm



Local Preference Scanning Worm

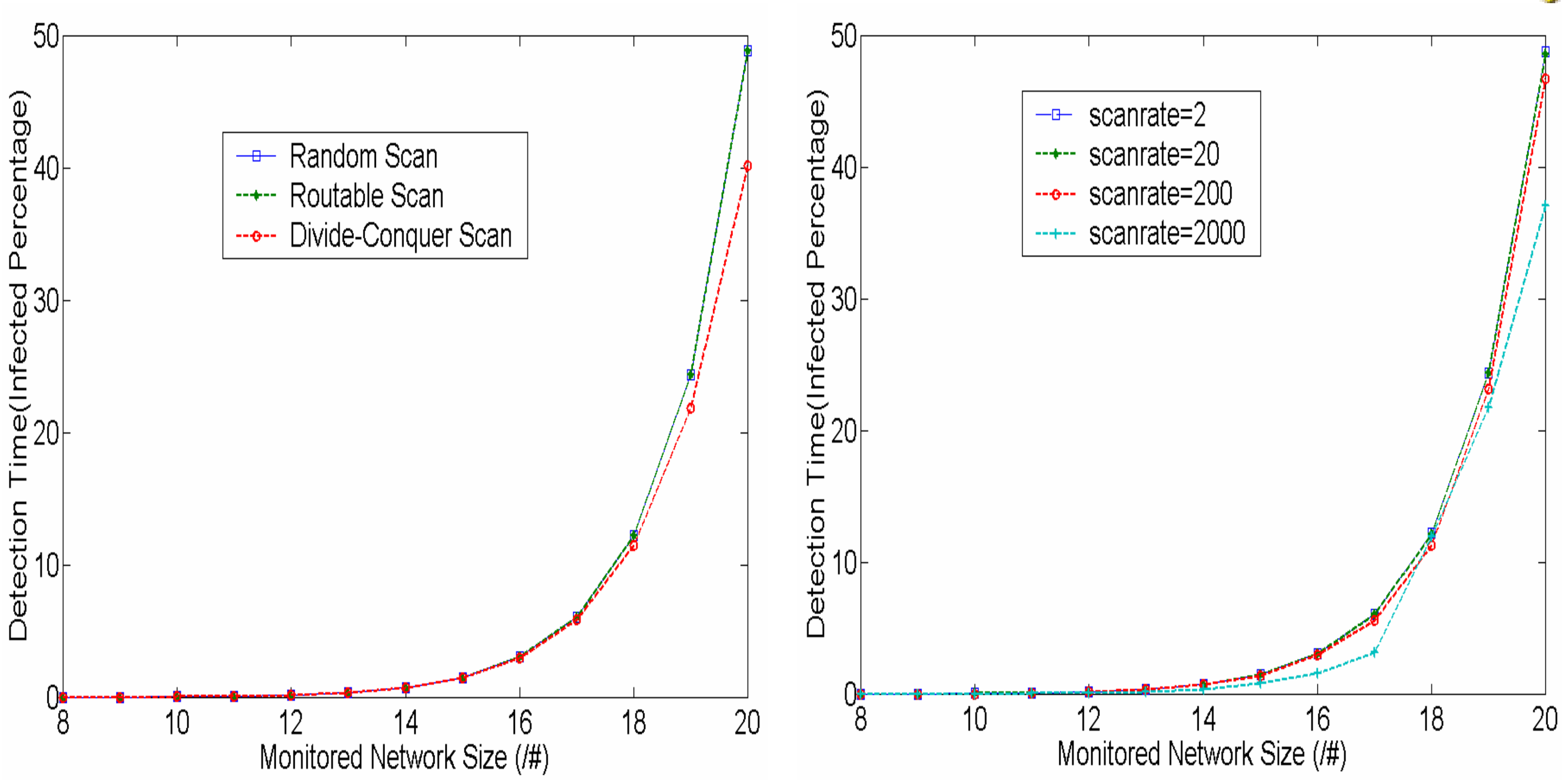
Worm Early Warning

- With a good local victim detection algorithm is available, what’s the effectiveness of worm early warning?
- We evaluate the detection time in terms of infected percentage of the whole Internet's vulnerable hosts when at least one infected victim in our monitored network is identified, i.e., the time when $v_i \geq 1$.

Analytical Results

- Worm warning occurs with 0.19% infection of all vulnerable hosts on Internet when using a /12 monitored network or 3.05% infection using a /16 monitor.
- For all kinds of scanning methods (random, routable, divide-conquer, local preference, sequential scan), the performance is almost the same.

Scan Methods or Rates Don’t Matter



Network Simulator Experiments

- Using a packet level worm simulator (based on GTNetS—a network simulator) to validate our local warning system and the results of our analytical models in an Internet-like setting.
- We used a hybrid network topology with cluster-ring backbone and hierarchical sub-networks.
- We use $\Omega=2^{16}$,T= Ω 3/4,N=32000,Hitlist=1
- Experiment results well matched the output of the analytical model.

Local Response based on Local Victim Information

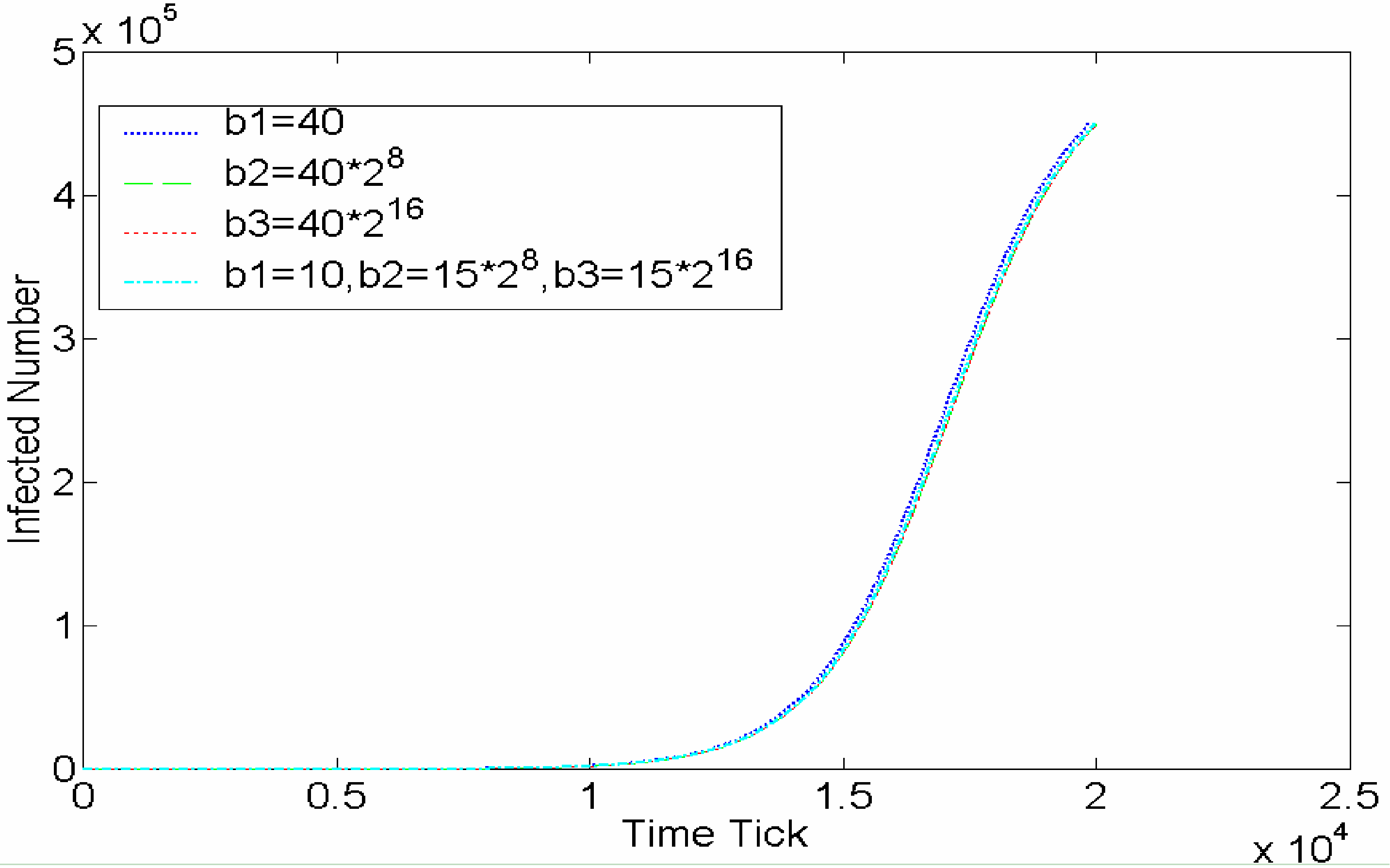
“Global” Response Vs. Local Response

- “Global” response: require complex and time consuming coordination between CDC-like authority and Internet routers.
- With the victim information provided by a local victim detection algorithm (e.g., DSC), we can automatically take immediate and accurate responses that block victims so as to effectively stop the outgoing propagation.

Rate Limit Vs. Accurate Quarantine

- Deploy rate limit on every host and limit connections all the time are expensive
- Local response can be more effective, since local administrators know details about the victim machines and take more accurate action to block (not rate limit) the outgoing connections of victims (not all hosts) at that port (not all ports).
- Host level or Network level local response?

Random Scan: Different deployment almost no effect

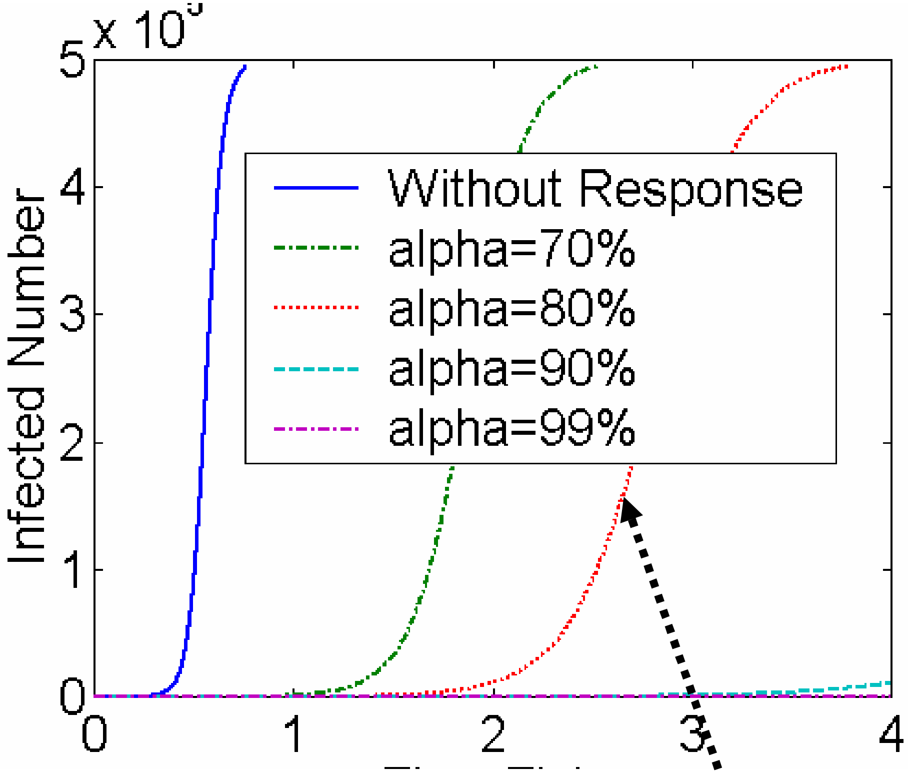


Random Scan

- Deploy rate of local response (quarantine rate): α=D/T
  - □ In every time tick, on average there will be α percent of scans blocked by local response.
- For random scan, we have

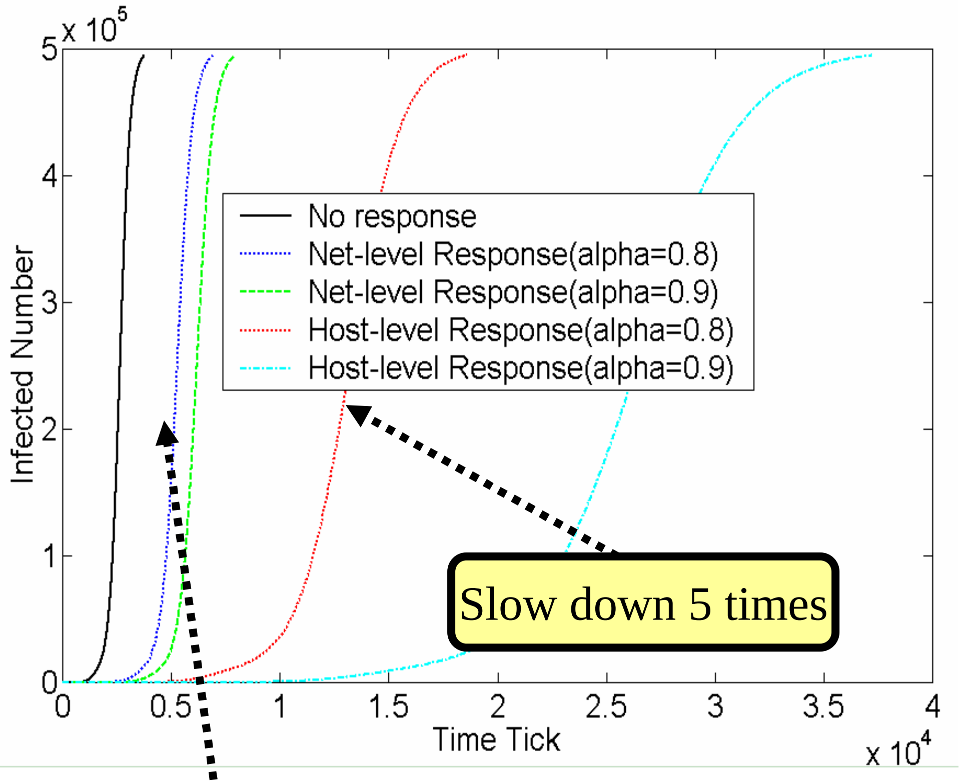
$$n_{i+1} = n_i + (N - n_i)(1 - (1 - 1/T)^{sn_i \frac{T}{\Omega}(1-\alpha)})$$

Local Response: Analytical Results



Random Scanning Worm

Slow down 5 times



Slow down 5 times

Local Preference Scanning Worm

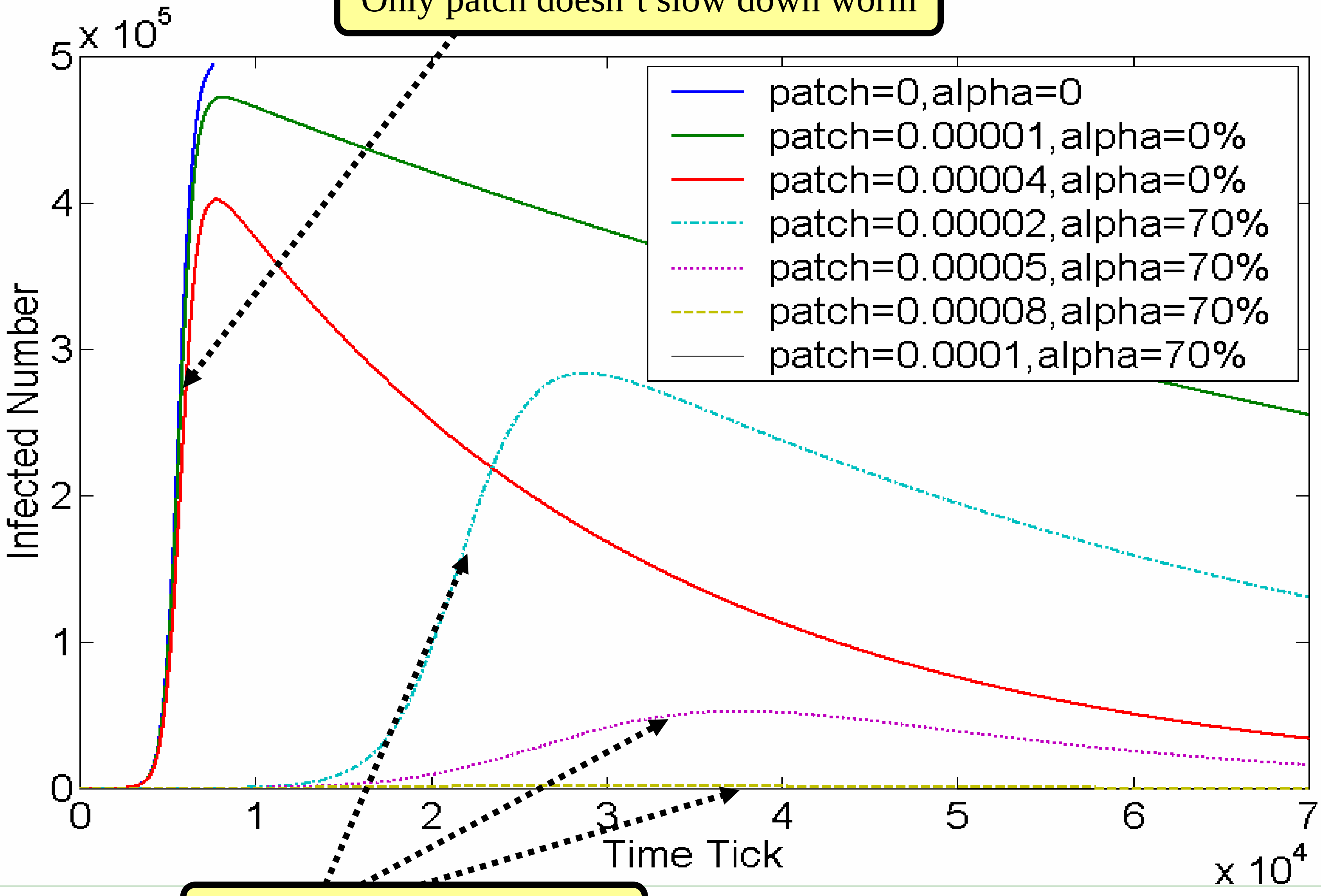
Slow down 2 times

Together with Patching

- Because we know the detail local victim information, we can easily take more aggressive and focused actions to immunize the victims.
- So in addition to a quarantine rate we also consider the effect of a delayed patching rate p, applied after using worm detection and analysis, with the response time t_r, the delay before people learn about the need to patch.
- When i> t_r , we have

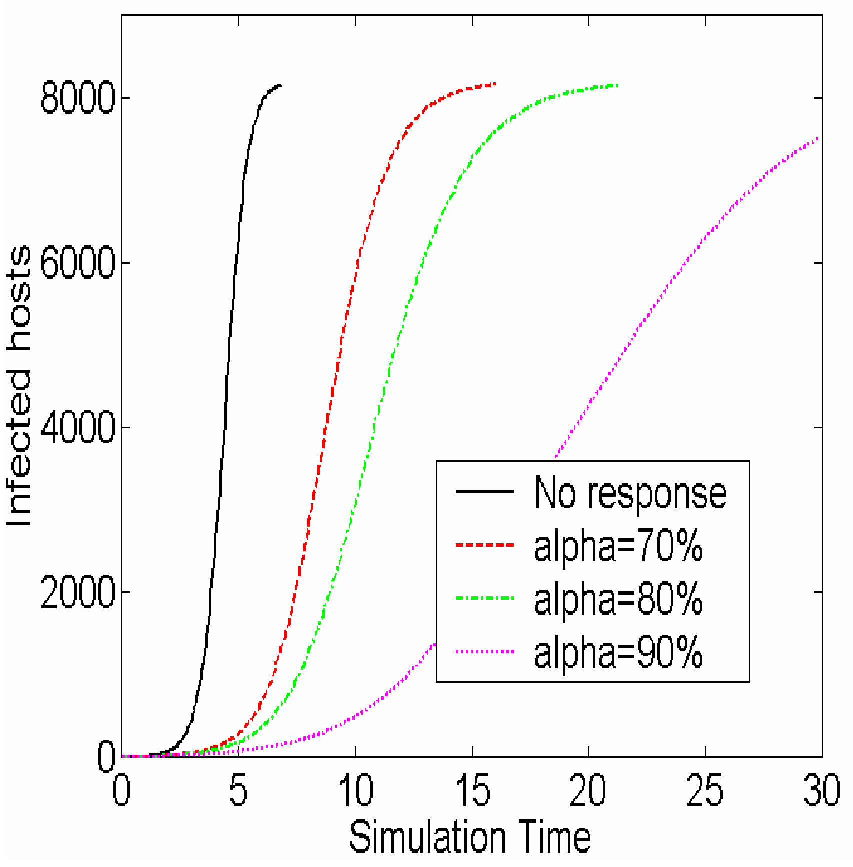
$$n_{i+1} = (1-p)n_i+[((1-p)^{i-t_r}N-n_i)(1-(1-1/T)^{sn_i\frac{T}{\Omega}(1-\alpha)})]$$

Only patch doesn’t slow down worm

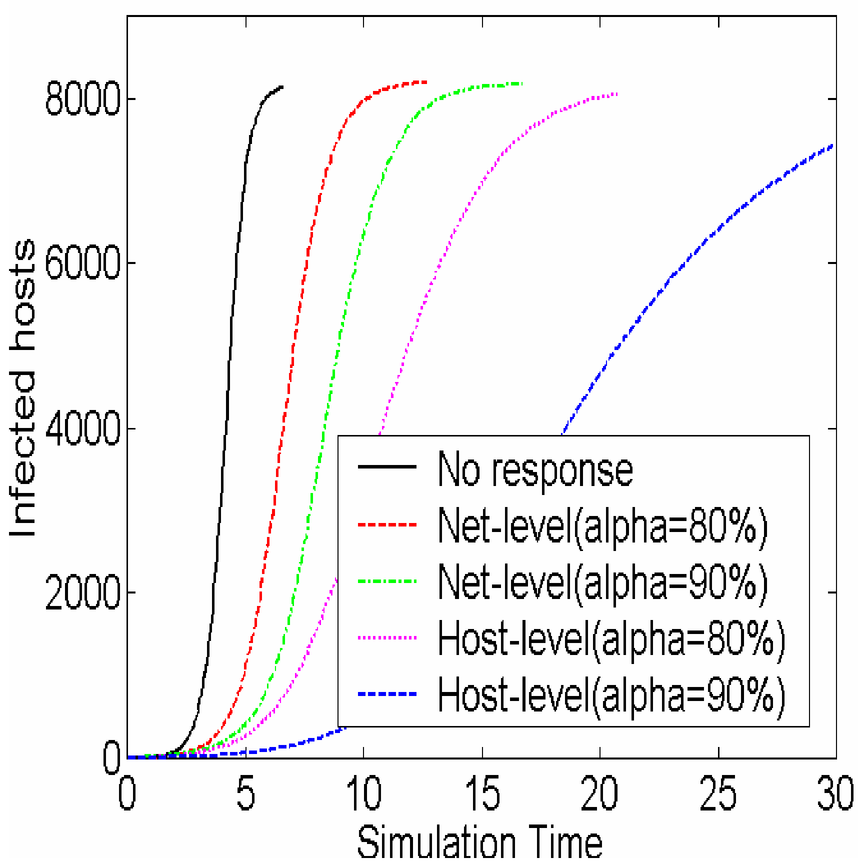


First slow down, then stop

Local Response Simulation using worm simulator



Random Scanning Worm



Local Preference Scanning Worm

Conclusion

- DSC: a full worm behavior based local victim detection algorithm
- New analytical model to analyze different scanning worms
- Worm early warning using local victim information is effective
- Based on accurate local victim detection, an automatic, real-time local response can greatly slow down the Internet worm propagation, and can stop it together using patching

Q &A

Thank you!