Monitoring And Early Warning For Internet Worms
Cliff C. Zou, Lixin Gao, Weibo Gong, Don Towsley (Univ. Massachusetts, Amherst)
Presented By: Xiaojin Niu

Outline
- Introduction and related work
- Worm propagation model
- Worm monitoring system
- Early warning scheme
- Experimental results
- Conclusion and discussion

Worm Threat Eg.
- Code Red, Nimda
  - Spread in 2001
  - Infected hundreds of thousands of computers
  - Caused millions of dollars of loss
- SQL Slammer
  - Spread on 1/25/2003
  - Doubled in size every 8.5 seconds
  - Infected more than 90% of vulnerable hosts within 10 minutes

What Is This Paper About?
- How to detect an unknown worm at its early stage?
- Monitoring:
  - Monitor and collect worm scan traffic
  - Observation data is very noisy
    - Old worms' scans
    - Port scans by hacking toolkits
- Detecting:
  - Anomaly detection for unknown worms
  - Traditional anomaly detection: threshold-based
    - Check traffic burst (short-term or long-term).
    - Difficulties: False alarm rate

A Different Approach
- Non-threshold-based worm detection method
  - "Trend Detection"
    - Detect traffic trend, not burst
- Trend: worm exponential growth trend at the beginning
- Detection: the exponential rate should be a positive, constant value

Fast Worm Spreading Pattern
- Fast worm has exponential growth pattern
  - Attacker's incentive: infect as many as possible before counteractions.
“Trend Detection”
— Detect traffic trend, not burst

Related Work
- “Network telescope”, D. Moore, 2002
- “Honeynet”
- “Enterprise early warning solution”, Symantec Corp.
- “Internet storm center”, SANS Institute
- “TRAFFEN”, Berk et al.

Assumption (1)
- Worms: uniformly scan the Internet
  - No prior knowledge of where vulnerable computers reside
- Worm scanning strategies
  - Hackers have tried various scanning strategies in their scan-based worms
    - Uniform scan — Code Red, Slammer
    - Local preference scan (hit-list) — Code Red II
    - Sequential scan — Blaster

Assumption (2)
- IP infrastructure: IPv4
  - IPv6 -> 2^2^3^2 IP address, hard for a worm to propagate through random IP scan

Worm Propagation Model
Simple Epidemic Model

\[ N = I(t) + S(t) \]
\[ \frac{dI(t)}{dt} = \beta I(t) S(t) \]

Worm infection rate: \( \alpha = \beta N \)

\( N \): # of hosts, \( S \): # of susceptible, \( I \): # of infectious, \( \beta \): Infectious ability, \( \alpha \): infection rate

Figure 1: Worm propagation model
**Why Use Simple Epidemic Model?**

- Can model most scan-based worms

![Graphs showing infection rate](Image)

**Figure from:**

**Discrete-time Model**

- Time is divided into intervals of length $\Delta$
  \[ I_t = (1 + \alpha)I_{t-1} - \beta I_{t-1}^2 \]
- $I_t$: the number of infected hosts at the real time $t$
- $\alpha = \beta N$
- $\alpha$ the infection rate
  - The average number of vulnerable hosts that can be infected per unit time by one infected host during the early stage of worm propagation

**Monitoring System**

- Provides comprehensive observation data on a worm’s activities for the early detection of the worm
- Consists of:
  - Malware Warning Center (MWC)
  - Distributed monitors
    - Ingress scan monitors
    - Egress scan monitors

**Monitoring System Architecture**

![Diagram of monitoring system architecture](Image)

**Ingress Scan Monitors**

- Monitor scan traffic coming into a local network by logging incoming traffic to unused IP addresses
- Located on gateways or borders of local networks

**Egress Scan Monitors**

- Monitor the outgoing traffic from a network to infer a potential worm’s scan behavior
- Located at the egress point of a local network
Data Collection

- Ingress monitors collect:
  - Number of scans received in $t$-th interval
  - IP address of infected hosts that have sent scans to the monitors by the time $t \Delta$
- Egress monitors collect:
  - Average worm scan rate
- Malware Warning Center (MWC) monitors:
  - Worm’s average scan rate: $\eta$
  - Total number of scans monitored: $Z_t$
  - Number of infected hosts observed: $C_t$

Correction Of Biased Observation

- Limited number of monitors:
  - Each worm scan has a very small probability of being observed
  - Number of infected hosts monitored $C_t$ is not proportional to $I_t$
- How to estimate $I_t$ using $C_t$?
  - Bias correction formula: $\bar{I}_t = \frac{C_{t+1} - (1-p)^\eta C_t}{1-(1-p)^\eta}$

Validity of the Biased Correction Formula

Kalman Filter Estimation

- Equivalent to Recursive Least Square Estimator:
  - Give estimation at each discrete time.
  - Robust to noise.
- System: Discrete-time simple epidemic model
  \[ I_i = (1 + \alpha \Delta) I_{i-1} - \beta \Delta I_{i-1}^2 \quad i = 1, 2, \ldots, t/\Delta \]
- System state: $X = [1 + \alpha \Delta \beta \Delta]$.
- Worm infection rate $\alpha$. ($\alpha = \beta N$, exponential growth rate at beginning)
- Epidemic parameter $\beta$. (worm infectious ability)
- Measurement from monitors: $y_t = \delta I_t + \omega_t$
- $C_t$: cumulative # of observed infected, $Z_t$: # of scans at time $t$.

Kalman Filter Estimation

\[
\begin{align*}
X_t &= X_{t-1} \\
y_t &= H_t X_t + \nu_t
\end{align*}
\]

Kalman Filter for estimation of $X_t$:

\[
\begin{align*}
H_t &= [y_{t-1} \ y_{t-1}^2] \\
K_t &= P_{t-1} H_t^T / (H_t P_{t-1} H_t^T + R_t) \\
P_t &= (I - K_t H_t) P_{t-1} \\
X_t &= X_{t-1} + K_t (y_t - H_t X_{t-1})
\end{align*}
\]

Estimate Vulnerable Population

Direct from Kalman filter: $X_t = [1 + \alpha \Delta \beta]$

\[\alpha = \beta N \quad \Rightarrow \quad \hat{N} = \frac{\alpha}{\beta} \]

Alternative method:

$\eta$: A worm sends out $\eta$ scans per $\Delta$ time

\[\hat{N} = \frac{2^{2\Delta} \alpha}{\eta} \]

Estimation of population $N$
Overview Of Detecting Steps

- MWC collects and aggregates reports from distributed monitors
  - If scan traffic is over an alarm threshold for monitored illegitimated scan traffic $Z_t$ for several consecutive intervals, MWC activates the Kalman filter and begins to recode $C_t$ and $\eta$
  - MWC recursively estimates the infection rate $\alpha$
  - If estimated oscillates slightly around a positive, constant value, a worm is spreading

Code Red Simulation Experiments

| Population: N=160,000 | Infection rate: $\alpha = 1.8$/hour, |
| Scan rate: $\eta = N(358/min, 100^2)$ | Initially infected: $I_0 = 10$ |
| Monitored IP space $2^{20}$ | Monitoring interval: $\Delta = 1$ minute |

![Graph showing Code Red simulation results]

Before 2% (223 min): estimate is already stabilized and oscillating a little around a positive constant value

SQL Slammer Simulation Experiments

| Population: N=100,000 | Monitored IP space $2^{20}$, |
| Scan rate: $\eta = N(4000/sec, 2000^2)$ | Initially infected: $I_0 = 10$ |
| Monitoring interval: $\Delta = 1$ second | Consider background noise |

![Graph showing SQL Slammer simulation results]

Before 1% (45 sec): estimate is already stabilized and oscillating around a positive constant value

Summary

- Effective algorithms for early detection of the presence of a worm and the corresponding monitoring system
- Use a Kalman filter to detect a worm’s propagation at its early stage in real-time
  - Simple epidemic model
  - Observation data from the monitoring system

Discussion And Future Work

- Trend detection: non-threshold-based methodology
  - Principle: detect traffic trend, not burst
  - Pros: Robust to background noise → low false alarm rate
  - Cons: Rely on worm model, representation of observation data
- The simple epidemic model?
- Uniform scan?
- Monitoring interval?
Questions?
Comments?

Thank you!