Autograph
Toward Automated, Distributed Worm Signature Detection

Hyang-Ah Kim
Carnegie Mellon University

Brad Karp
Intel Research &
Carnegie Mellon University

Usenix Security 2004
Internet Worm Quarantine

- Internet Worm Quarantine Techniques
  - Destination port blocking
  - Infected source host IP blocking
  - Content-based blocking [Moore et al., 2003]

- Worm Signature

Signature for CodeRed II

Signature: A Payload Content String Specific To A Worm
Content-based Blocking

Signature for CodeRed II

- Can be used by Bro, Snort, Cisco’s NBAR, ...
Signature derivation is too slow

- **Current Signature Derivation Process**
  - New worm outbreak
  - Report of anomalies from people via phone/email/newsgrp
  - Worm trace is captured
  - Manual analysis by security experts
  - Signature generation

  ⇒ Labor-intensive, Human-mediated
Goal

Automatically generate signatures of previously unknown Internet worms

- as accurately as possible
  ⇒ Content-Based Analysis

- as quickly as possible
  ⇒ Automation, Distributed Monitoring
Assumptions

- We focus on TCP worms that propagate via scanning

  Actually, any transport
  - in which spoofed sources cannot communicate successfully
  - in which transport framing is known to monitor

- Worm’s payloads share a common substring
  - Vulnerability exploit part is not easily mutable
    - Not polymorphic
Outline

- Problem and Motivation
- Automated Signature Detection
  - Desiderata
  - Technique
  - Evaluation
- Distributed Signature Detection
  - Tattler
  - Evaluation
- Related Work
- Conclusion
Desiderata

- Automation: Minimal manual intervention

- Signature quality: Sensitive & specific
  - Sensitive: match all worms $\Rightarrow$ low false negative rate
  - Specific: match only worms $\Rightarrow$ low false positive rate

- Timeliness: Early detection

- Application neutrality
  - Broad applicability
Automated Signature Generation

- Step 1: Select suspicious flows using heuristics
- Step 2: Generate signature using content-prevalence analysis
S1: Suspicious Flow Selection

Reduce the work by filtering out vast amount of innocuous flows

- **Heuristic:** Flows from scanners are suspicious
  - Focus on the successful flows from IPs who made unsuccessful connections to more than $s$ destinations for last 24 hours
  - Suitable heuristic for TCP worm that scans network

**Autograph ($s = 2$)**

- Non-existent
- Non-existent

This flow will be selected
S1: Suspicious Flow Selection

Reduce the work by filtering out vast amount of innocuous flows

- **Heuristic:** Flows from scanners are suspicious
  - Focus on the successful flows from IPs who made unsuccessful connections to more than $S$ destinations for last 24 hours
  - Suitable heuristic for TCP worm that scans network

- **Suspicious Flow Pool**
  - Holds reassembled, suspicious flows captured during the last time period $t$
  - Triggers signature generation if there are more than $\theta$ flows
S2: Signature Generation

Use the most frequent byte sequences across suspicious flows as signatures

All instances of a worm have a common byte pattern specific to the worm

Rationale
- Worms propagate by duplicating themselves
- Worms propagate using vulnerability of a service

How to find the most frequent byte sequences?
Worm-specific Pattern Detection

- Use the entire payload
  - Brittle to byte insertion, deletion, reordering

<table>
<thead>
<tr>
<th>Flow 1</th>
<th>GARBAGEEABCDEFGHIJKLMNOPQRSTUVWXYZ</th>
</tr>
</thead>
<tbody>
<tr>
<td>Flow 2</td>
<td>GARBAGEABCDEFGHIJKLMNOPQRSTUVWXYZ</td>
</tr>
</tbody>
</table>
Worm-specific Pattern Detection

Partition flows into non-overlapping small blocks and count the number of occurrences

- Fixed-length Partition
  - Still brittle to byte insertion, deletion, reordering

<table>
<thead>
<tr>
<th>Flow 1</th>
<th>Flow 2</th>
</tr>
</thead>
<tbody>
<tr>
<td>GARBAGEEABCDEFGH</td>
<td>GARBAGEA</td>
</tr>
<tr>
<td>IJKABCDXXXX</td>
<td>ABCDEFGHIJKABCD</td>
</tr>
<tr>
<td>XXX</td>
<td>XXX</td>
</tr>
</tbody>
</table>
Worm-specific Pattern Detection

- **Content-based Payload Partitioning (COPP)**
  - Partition if Rabin fingerprint of a sliding window matches Breakmark → **Content Blocks**
  - Configurable parameters: content block size (minimum, average, maximum), breakmark, sliding window

![Diagram showing content blocks and breakmark with examples of Flow 1 and Flow 2]

Flow 1: AGEEABCD
Flow 2: GARBAGEABCD

Breakmark = last 8 bits of fingerprint (ABCD)
Why Prevalence?

Prevalence Distribution in Suspicious Flow Pool
- From 24-hr http traffic trace

- Worm flows dominate in the suspicious flow pool
- Content-blocks from worms are highly ranked
Select Most Frequent Content Block

f0: C F
f1: C D G
f2: A B D
f3: A C E
f4: A B E
f5: A B D
f6: H I J
f7: I H J
f8: G I J
Select Most Frequent Content Block

<p>| | | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>f0</td>
<td>C</td>
<td>F</td>
<td></td>
<td></td>
</tr>
<tr>
<td>f1</td>
<td>C</td>
<td>D</td>
<td>G</td>
<td></td>
</tr>
<tr>
<td>f2</td>
<td>A</td>
<td>B</td>
<td>D</td>
<td></td>
</tr>
<tr>
<td>f3</td>
<td>A</td>
<td>C</td>
<td>E</td>
<td></td>
</tr>
<tr>
<td>f4</td>
<td>A</td>
<td>B</td>
<td>E</td>
<td></td>
</tr>
<tr>
<td></td>
<td>A</td>
<td>B</td>
<td>D</td>
<td></td>
</tr>
<tr>
<td>f6</td>
<td>B</td>
<td>H</td>
<td>I</td>
<td>J</td>
</tr>
<tr>
<td>f7</td>
<td>B</td>
<td>I</td>
<td>H</td>
<td>J</td>
</tr>
<tr>
<td>f8</td>
<td>B</td>
<td>G</td>
<td>I</td>
<td>J</td>
</tr>
</tbody>
</table>

|   |    |    |    |
|---|----|----|
| f0 | C  | F  |
| f1 | C  | D  | G  |
| f2 | A  | B  | D  |
| f3 | A  | C  | E  |
| f4 | A  | B  | E  |
| f5 | A  | B  | D  |
| f6 | H  | I  | J  |
| f7 | I  | H  | J  |
| f8 | G  | I  | J  |
Select Most Frequent Content Block

Signature:

W: target coverage in suspicious flow pool
P: minimum occurrence to be selected

f0  C F
f1  C D G
f2  A B D
f3  A C E
f4  A B E
f5  A B D
f6  H I J
f7  I H J
f8  G I J

W≥90%
P≥3
Select Most Frequent Content Block

Signature: A

W: target coverage in suspicious flow pool
P: minimum occurrence to be selected

W ≥ 90%
P ≥ 3

<table>
<thead>
<tr>
<th>f0</th>
<th>C F</th>
</tr>
</thead>
<tbody>
<tr>
<td>f1</td>
<td>C D G</td>
</tr>
<tr>
<td>f2</td>
<td>A B D</td>
</tr>
<tr>
<td>f3</td>
<td>A C E</td>
</tr>
<tr>
<td>f4</td>
<td>A B E</td>
</tr>
<tr>
<td>f5</td>
<td>A B D</td>
</tr>
<tr>
<td>f6</td>
<td>H I J</td>
</tr>
<tr>
<td>f7</td>
<td>I H J</td>
</tr>
<tr>
<td>f8</td>
<td>G I J</td>
</tr>
</tbody>
</table>

Usenix Security 2004
Select Most Frequent Content Block

Signature: A

W: target coverage in suspicious flow pool
P: minimum occurrence to be selected

W ≥ 90%

P ≥ 3

<p>| | | | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>f0</td>
<td>C</td>
<td>F</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>f1</td>
<td>C</td>
<td>D</td>
<td>G</td>
<td></td>
<td></td>
</tr>
<tr>
<td>f2</td>
<td>A</td>
<td>B</td>
<td>D</td>
<td></td>
<td></td>
</tr>
<tr>
<td>f3</td>
<td>A</td>
<td>C</td>
<td>E</td>
<td></td>
<td></td>
</tr>
<tr>
<td>f4</td>
<td>A</td>
<td>B</td>
<td>E</td>
<td></td>
<td></td>
</tr>
<tr>
<td>f5</td>
<td>A</td>
<td>B</td>
<td>D</td>
<td></td>
<td></td>
</tr>
<tr>
<td>f6</td>
<td>H</td>
<td>I</td>
<td>J</td>
<td></td>
<td></td>
</tr>
<tr>
<td>f7</td>
<td>I</td>
<td>H</td>
<td>J</td>
<td></td>
<td></td>
</tr>
<tr>
<td>f8</td>
<td>G</td>
<td>I</td>
<td>J</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Select Most Frequent Content Block

Signature: \( A \quad I \)

\[ W \geq 90\% \]

\( W \): target coverage in suspicious flow pool

\( P \): minimum occurrence to be selected

<table>
<thead>
<tr>
<th>( f_0 )</th>
<th>C F</th>
</tr>
</thead>
<tbody>
<tr>
<td>( f_1 )</td>
<td>C D G</td>
</tr>
<tr>
<td>( f_2 )</td>
<td>A B D</td>
</tr>
<tr>
<td>( f_3 )</td>
<td>A C E</td>
</tr>
<tr>
<td>( f_4 )</td>
<td>A B E</td>
</tr>
<tr>
<td>( f_5 )</td>
<td>A B D</td>
</tr>
<tr>
<td>( f_6 )</td>
<td>H I J</td>
</tr>
<tr>
<td>( f_7 )</td>
<td>T H I</td>
</tr>
<tr>
<td>( f_8 )</td>
<td>G I T</td>
</tr>
</tbody>
</table>

\( P \geq 3 \)
Select Most Frequent Content Block

Signature:  A  I

W: target coverage in suspicious flow pool
P: minimum occurrence to be selected

<table>
<thead>
<tr>
<th></th>
<th>f0</th>
<th>f1</th>
<th>f2</th>
<th>f3</th>
<th>f4</th>
<th>f5</th>
<th>f6</th>
<th>f7</th>
<th>f8</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>C</td>
<td>C</td>
<td>D</td>
<td>A</td>
<td>A</td>
<td>A</td>
<td>H</td>
<td>I</td>
<td>G</td>
</tr>
<tr>
<td>B</td>
<td>F</td>
<td>D</td>
<td>E</td>
<td>C</td>
<td>B</td>
<td>B</td>
<td>I</td>
<td>J</td>
<td>I</td>
</tr>
<tr>
<td>C</td>
<td>G</td>
<td>F</td>
<td>D</td>
<td>E</td>
<td>E</td>
<td>J</td>
<td>J</td>
<td>T</td>
<td>T</td>
</tr>
<tr>
<td>D</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

W ≥ 90%
P ≥ 3
Outline

- Problem and Motivation
- Automated Signature Detection
  - Desiderata
  - Technique
  - Evaluation
- Distributed Signature Detection
  - Tattler
  - Evaluation
- Related Work
- Conclusion
Behavior of Signature Generation

- Objectives
  - Effect of COPP parameters on signature quality

- Metrics
  - Sensitivity = # of true alarms / total # of worm flows $\Rightarrow$ false negatives
  - Efficiency = # of true alarms / # of alarms $\Rightarrow$ false positives

- Trace
  - Contains 24-hour http traffic
  - Includes 17 different types of worm payloads
Signature Quality

- Larger block sizes generate more specific signatures
- A range of w (90-95%, workload dependent) produces a good signature
Outline

- Problem and Motivation
- Automated Signature Detection
  - Desiderata
  - Technique
  - Evaluation
- Distributed Signature Detection
  - Tattler
  - Evaluation
- Related Work
- Conclusion
Signature Generation Speed

- Bounded by worm payload accumulation speed
  - Aggressiveness of scanner detection heuristic
    - \( s \): # of failed connection peers to detect a scanner
  - # of payloads enough for content analysis
    - \( \theta \): suspicious flow pool size to trigger signature generation

- Single Autograph
  - Worm payload accumulation is slow

- Distributed Autograph
  - Share scanner IP list
  - Tattler: limit bandwidth consumption within a predefined cap
Benefit from tattler

- Worm payload accumulation (time to catch 5 worms)

<table>
<thead>
<tr>
<th>Info Sharing</th>
<th>Autograph Monitor</th>
<th>Aggressive $s=1$</th>
<th>Conservative $s=4$</th>
</tr>
</thead>
<tbody>
<tr>
<td>None</td>
<td>Luckiest</td>
<td>2%</td>
<td>60%</td>
</tr>
<tr>
<td></td>
<td>Median</td>
<td>25%</td>
<td>--</td>
</tr>
<tr>
<td>Tattler</td>
<td>All</td>
<td>&lt;1%</td>
<td>15%</td>
</tr>
</tbody>
</table>

- Signature generation
  - More aggressive scanner detection ($s$) and signature generation trigger ($\theta$) \(\Rightarrow\) faster signature generation, more false positives
  - With $s=2$ and $\theta=15$, Autograph generates the good worm signature before < 2% hosts get infected

Many innocuous misclassified flows

Conservative $(s = 4)$

Aggressive $(s = 1)$

Fraction of Infected Hosts

Monitor Info Sharing

Many innocuous misclassified flows
Related Work

- **Automated Worm Signature Detection**

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Signature Generation</strong></td>
<td>Content prevalence \rightarrow Address Dispersion</td>
<td>Honeypot + Pairwise LCS</td>
<td>Suspicious flow selection \rightarrow Content prevalence</td>
</tr>
<tr>
<td><strong>Deployment</strong></td>
<td>Network</td>
<td>Host</td>
<td>Network</td>
</tr>
<tr>
<td><strong>Flow Reassembly</strong></td>
<td>No</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td><strong>Distributed Monitoring</strong></td>
<td>No</td>
<td>No</td>
<td>Yes</td>
</tr>
</tbody>
</table>

- **Distributed Monitoring**
  - Honeyd [Provos 2003], DOMINO [Yegneswaran et al. 2004]
  - Corroborate faster accumulation of worm payloads/scanner IPs
Future Work

- **Attacks**
  - Overload Autograph
  - Abuse Autograph for DoS attacks

- **Online evaluation with diverse traces & deployment on distributed sites**

- **Broader set of suspicious flow selection heuristics**
  - Non-scanning worms (ex. hit-list worms, topological worms, email worms)
  - UDP worms

- **Egress detection**

- **Distributed agreement for signature quality testing**
  - Trusted aggregation
Conclusion

- Stopping spread of novel worms requires early generation of signatures
- Autograph: automated signature detection system
  - Automated suspicious flow selection → Automated content prevalence analysis
  - COPP: robustness against payload variability
  - Distributed monitoring: faster signature generation
- Autograph finds sensitive & specific signatures early in real network traces
For more information, visit
http://www.cs.cmu.edu/~hakim/autograph
Attacks

- Overload due to flow reassembly
  
  Solutions
  ⇒ Multiple instances of Autograph on separate HW (port-disjoint)
  ⇒ Suspicious flow sampling under heavy load

- Abuse Autograph for DoS: pollute suspicious flow pool
  
  o Port scan and then send innocuous traffic
    Solution
    ⇒ Distributed verification of signatures at many monitors
  
  o Source-address-spoofed port scan
    Solution
    ⇒ Reply with SYN/ACK on behalf of non-existent hosts/services
Number of Signatures

- Smaller block sizes generate small # of signatures
A modified RTCP (RTP Control Protocol)

Limit the total bandwidth of announcements sent to the group within a predetermined cap

---

**tattler**
About 340,000 vulnerable hosts from about 6400 ASes

Service deployment
- 50% of address space within the vulnerable ASes is reachable
- 25% of reachable hosts run web server
- 340,000 vulnerable hosts are randomly placed.

Scanning
- 10 probes per second
- Scanning the entire non-class-D IP address space

Network/processing delays
- Randomly chosen in [0.5, 1.5] seconds