LOW-RATE, FLOW-LEVEL PERIODICITY DETECTION

Genevieve Bartlett¹, John Heidemann¹, Christos Papadopoulos²

¹USC/Information Sciences Institute Marina del Rey, CA
²Colorado State University, Ft. Collins, CO
MOTIVATION

It’s 10pm, do you know what your computer’s doing??

- Automatic computer initiated communication
- More complex systems = more computer initiated communication
Low-Rate and Periodic Connections

- Subset of computer initiated: periodic connections
- Find periodic series in aggregate traffic with signal processing
- Flow-level
  - Event = connection start
  - Our methods could apply to many other events
- Low-Rate: 2s to several hours (Days? Weeks?)
Applies to Many Applications

Many applications are low-rate periodic:

- User services (30-120 mins)
  - WeatherEye
  - MacOS Dashboard apps
  - Clock applet in Gnome (Linux)
- RSS News Feeds (30-60 mins)
- Web Counters (5-30 mins)
  - http refresh
- Peer-to-Peer (~20-30 mins)
- Adware (minutes to hours)
- Spyware
- Botnet Command & Control
CONTRIBUTIONS

- Low-rate periodicity as a phenomenon of interest
- Low-rate periodicity prevalent in real-world traffic
- Novel method for detection
- Demonstration of applications
  - Self-surveillance (GI paper)
  - Pre-filtering for detection triage
CONTRIBUTIONS

- Low-rate periodicity as a phenomenon of interest
- Low-rate periodicity prevalent in real-world traffic
- Novel method for detection
- Demonstration of applications
  - Self-surveillance (GI paper)
  - Pre-filtering for detection triage
CONTRIBUTIONS

- Low-rate periodicity as a phenomenon of interest
- Low-rate periodicity prevalent in real-world traffic
- Novel method for detection
- Demonstration of applications
  - Self-surveillance (GI paper)
  - Pre-filtering for detection triage
ARE PERIODIC APPLICATIONS PREVALENT?

- Pick an interesting application
  - Malware!
- How do we confirm periodic malware exists at USC?
  - No payload
  - Blacklisted sites
  - Aggregate traffic (groups of ~20)
  - Determine which groups show periodic communication
**HOW PREVALENT IS PERIODIC COMMUNICATION?**

<table>
<thead>
<tr>
<th>Group</th>
<th>Blacklisted Destinations</th>
<th>Unique IPs (users)</th>
</tr>
</thead>
<tbody>
<tr>
<td>active to anywhere</td>
<td></td>
<td>128,614 [100%]</td>
</tr>
<tr>
<td>active to blacklisted</td>
<td>181 (100%)</td>
<td></td>
</tr>
<tr>
<td>Non-periodic</td>
<td>120 (66%)</td>
<td>n/a n/a</td>
</tr>
<tr>
<td>Periodic</td>
<td>61 (44%)</td>
<td>n/a n/a</td>
</tr>
<tr>
<td>User Services</td>
<td>5 (3%)</td>
<td>22 [0%]</td>
</tr>
<tr>
<td>Web Counters</td>
<td>15 (8%)</td>
<td>16,405 [13%]</td>
</tr>
<tr>
<td>Ad Servers</td>
<td>36 (20%)</td>
<td>31,277 [24%]</td>
</tr>
<tr>
<td>Other</td>
<td>5 (3%)</td>
<td>6 [0%]</td>
</tr>
</tbody>
</table>

Nearly a third show periodic behavior!

∴ We can find 1/3 blacklisted servers on our network looking at periodic behavior as a first pass.
CONTRIBUTIONS

- Low-rate periodicity as a phenomenon of interest
- Low-rate periodicity prevalent in real-world traffic
- **Novel method for detection**
- Demonstration of applications
  - Self-surveillance (GI paper)
  - Pre-filtering for detection triage
**Typical Approach to Finding Periodic Events**

Network events > time series > FFT > analysis

2007-02-06 09:00:22.611315 IP 68.181.195.4 59790 > 121.97.1.237 64393: . 1460:2920(1460) ack 1 win 5840
2007-02-06 09:00:22.611329 IP 209.191.84.225 36554 > 128.125.253.79.25: . 93440:94900(1460) ack 1 win 65535
2007-02-06 09:00:22.611334 IP 209.73.189.144.80 > 68.181.253.104.2943: P 37960:38165(205) ack 1 win 64409
2007-02-06 09:00:22.611343 IP 209.191.84.225.36554 > 128.125.253.79.25: . 94900:96360(1460) ack 1 win 65535
2007-02-06 09:00:22.611358 IP 209.191.84.225.36554 > 128.125.253.79.25: . 96360:97820(1460) ack 1 win 65535

Time → FFT → Frequency
WHAT ARE WE LOOKING FOR?

- Given network data:
  - Is there a periodic event?
  - If so, what is the period?
  - Location in time: Start/Stop of events
## Goals and Design

<table>
<thead>
<tr>
<th>Preserve time information</th>
<th>wavelets</th>
</tr>
</thead>
<tbody>
<tr>
<td>Simple representation and implementation</td>
<td>Haar wavelet basis: differencing/averaging match for sharp changes</td>
</tr>
<tr>
<td>Low-rate periods</td>
<td>Coarse time bins ~1min+</td>
</tr>
<tr>
<td>Large range of frequencies</td>
<td>Iterative filter-bank Full decomposition</td>
</tr>
</tbody>
</table>
Different paths give different frequency splits.
Can focus in on a frequency range, if we know which we want \textit{a priori}.
MULTIRESOLUTION ANALYSIS: FULL

- Full decomposition
- We examine multiple frequency ranges
- Level of decomp determined by length and sample rate of original data
VISUALIZATION

Original Time Series

Level of decomp
**VISUALIZATION**

- High time Res.
- Shorter periods (2s)
- Longer periods (30min)

Period in seconds: 16.0 8.0 5.3 4.0 3.2 2.7 2.3 2.0

Level of decomp: 0 1 2 3 4 5 6
### Visualization

<table>
<thead>
<tr>
<th>Period in seconds</th>
<th>16.0</th>
<th>8.0</th>
<th>5.3</th>
<th>4.0</th>
<th>3.2</th>
<th>2.7</th>
<th>2.3</th>
<th>2.0</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>2</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>3</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td>4</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>4</td>
</tr>
<tr>
<td>5</td>
<td>5</td>
<td>5</td>
<td>5</td>
<td>5</td>
<td>5</td>
<td>5</td>
<td>5</td>
<td>5</td>
</tr>
</tbody>
</table>

**Color scale:** Percent of total energy

- 0% (light red)
- 50% (orange)
- 100% (dark red)

- (30min) Longer periods
- Shorter periods (2s)

**High time Res.**

**High freq. Res.**
ARTIFICIAL EXAMPLE: 8S PERIOD

(30min) Longer periods          Shorter periods (2s)

High time Res.

High freq Res.

Level of decomp

Base

Harmonics

Color scale: Percent of total energy
**ARTIFICIAL EXAMPLE: 8S PERIOD**

- (30min) Longer periods
- Shorter periods (2s)

High time Res.

High freq. Res.

---

Artificial Signal, 8s

<table>
<thead>
<tr>
<th>Period in seconds</th>
<th>Level of decomp</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td>4</td>
<td>4</td>
</tr>
<tr>
<td>5</td>
<td>5</td>
</tr>
</tbody>
</table>

Color scale: Percent of total energy

- 0% base
- 50% harmonics
- 100%
**Visualization: Real-world Example**

BitTorrent client communicating with tracker

(hours) Longer periods

Shorter periods (128s)

300s update with BitTorrent Tracker
**AUTOMATIC DETECTION**

- Detection of period
  - Empirically derived threshold on energy
  - Threshold dependent on frequency range and decomposition level
    - Too few decompositions, not focused on frequency range
    - Too many decompositions, energy spreads out

- Detection of *when* a change occurs
  - Start and stop of a periodic series of events
  - Move backwards on levels of decomposition to get more time resolution
    - Details in techreport
CONTRIBUTIONS

- Low-rate periodicity as a phenomenon of interest
- Low-rate periodicity prevalent in real-world traffic
- Novel method for detection
- Demonstration of applications
  - Self-surveillance (GI paper)
  - Pre-filtering for detection triage
APPLICATIONS

- Self-surveillance
  - Desktop user
  - Changes indicate problems: stop in OS updates, addition of adware etc.

- Pre-filtering
  - Target apps with low-rate periodic com.
  - Reduce set of hosts to investigate
  - Eg. Target BitTorrent trackers
SELF-SURVEILLANCE DEMONSTRATION

- Detect start or stop of periodic communication
- Here we look at unwanted communication: installation of a keylogger
- Applies to stop of wanted periodic communication too!
- Detect install of Keyboard Guardian on Windows
  - Set to report every 3 hours
- 3 day monitoring
  - 1st day, no keylogger
  - 2nd day, install keylogger
Numerical Detection of Event

Automatic Detection
Identifies presence (at harmonic)
Correctly identifies installation time
(within a 9 hour window).
VISUAL DETECTION OF CHANGE

Before

After

Report every 3 hours
(every 10,800s)
SUMMARY OF SELF-SURVEILLANCE

- Automatic detection
  - Identifies a periodic series of events
  - Identifies changes in events and when those changes occur

- Demonstrated
  - Keylogger: Addition of a bad series of periodic communication
  - OS updates: Removal of a good series of periodic communication (techreport)
SENSITIVITY TO NOISE

- Signal-to-Noise ratio
  - 1 signal connection: 10-20 unrelated connections
  - Easily achievable with periods of user inactivity
  - Watch for a long enough window
**SUMMARY**

- Variety of applications show periodic behavior
- New wavelet based approach to finding periodic behavior in aggregate traffic
- Demonstrated use for self-surveillance
- Techreport & GI paper:
  - [http://www.isi.edu/~bartlett/pubs/Bartlett09a.html](http://www.isi.edu/~bartlett/pubs/Bartlett09a.html)
**How to Quantify Sensitivity?**

- **Why?**
  - Know when we work and when we won’t

- **Quantify sensitivity to noise**
  - Fixed amount of background traffic
  - Vary frequency
  - Study base frequency energy
    - With background/No background
SENSITIVITY TO NOISE

Need SNR of at least ~0.05-0.1
1 periodic connection for every
10-20 non-periodic connections
IS EVASION POSSIBLE?

- Yes: Jitter
- How much jitter is enough?
- Experiment: vary jitter, study detection
  - Artificial signal
  - Jitter varies by Gaussian random
EVALUATING JITTER FOR EVASION

Greater than 15% hides signal.
Not disruptive to operation:
1 hr period ± 10 mins