
Detecting Botnets Through Log Correlation

Yusof Al-Hammadi and Uwe Aickelin

**Automated Scheduling, Optimisation and Planning
(ASAP)**

**School of Computer Science
The University of Nottingham**



Outline

- Background
 - Botnet definition
 - Motivation
 - Botnet Lifecycle
- Objectives
- Data Collection and Design
- Results
- Conclusion and Future Work

Background

■ Botnet definition

- Collection of compromised machines (Bots) that are connected to a single IRC channel
- Respond simultaneously to various instructions generated by attacker
- Examples: spybot, sdbot, agobot, q8bot

■ Motivation

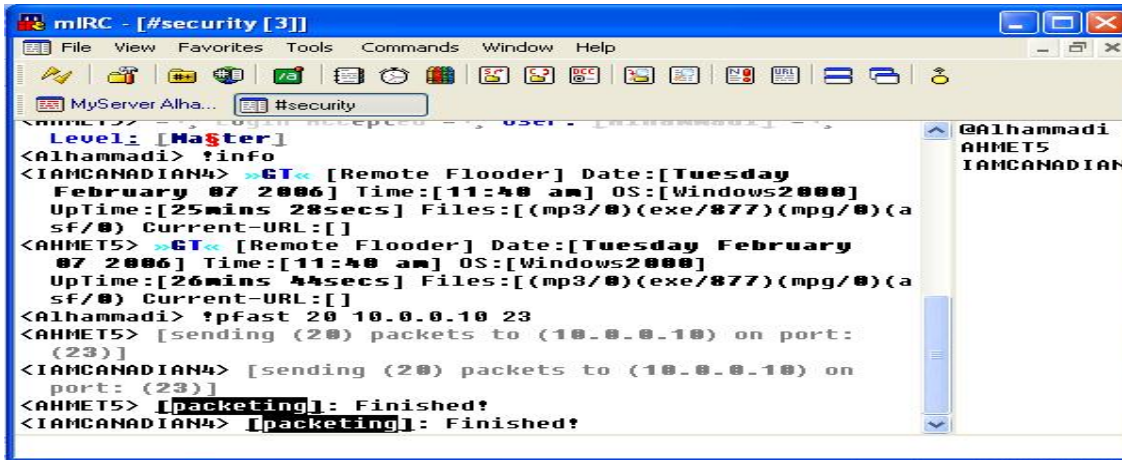
- Serious threats which includes:
 - DDoS
 - Email Spamming
 - Stealing sensitive information
 - Keystroke logging
 - Extortion



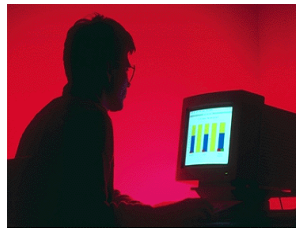
Bot Lifecycle

IRC Channel

Infection Source

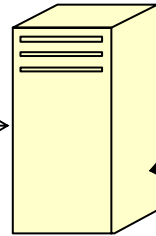


```
mIRC - [#security [3]]
File View Favorites Tools Commands Window Help
MyServer Alpha... #security
Level: [Master]
<Alhammadi> !info
<IAMCANADIAN4> >>GT<< [Remote Flooder] Date:[Tuesday
February 07 2006] Time:[11:40 am] OS:[Windows2000]
UpTime:[25mins 28secs] Files:[(mp3/0)(exe/877)(mpg/0)(a
sf/0) Current-URL:[
<AHMET5> >>GT<< [Remote Flooder] Date:[Tuesday February
07 2006] Time:[11:40 am] OS:[Windows2000]
UpTime:[26mins 44secs] Files:[(mp3/0)(exe/877)(mpg/0)(a
sf/0) Current-URL:[
<Alhammadi> !pfast 20 10.0.0.10 23
<AHMET5> [sending (20) packets to (10.0.0.10) on port:
(23)]
<IAMCANADIAN4> [sending (20) packets to (10.0.0.10) on
port: (23)]
<AHMET5> [packeting]: Finished!
<IAMCANADIAN4> [packeting]: Finished!
```



Attacker

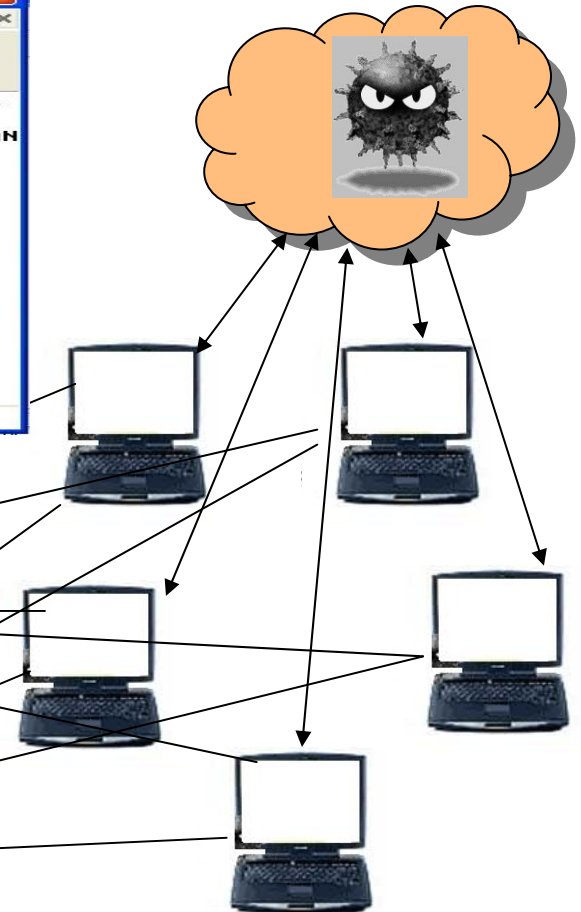
commands



IRC Server



Victim



Objectives

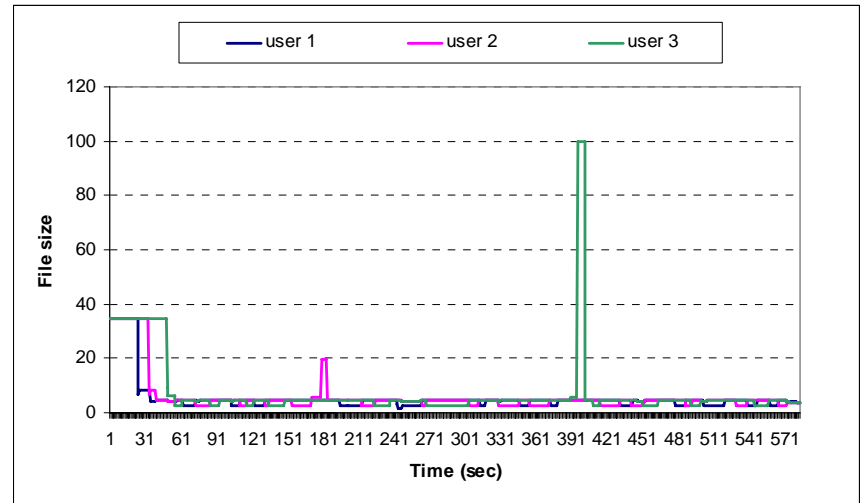
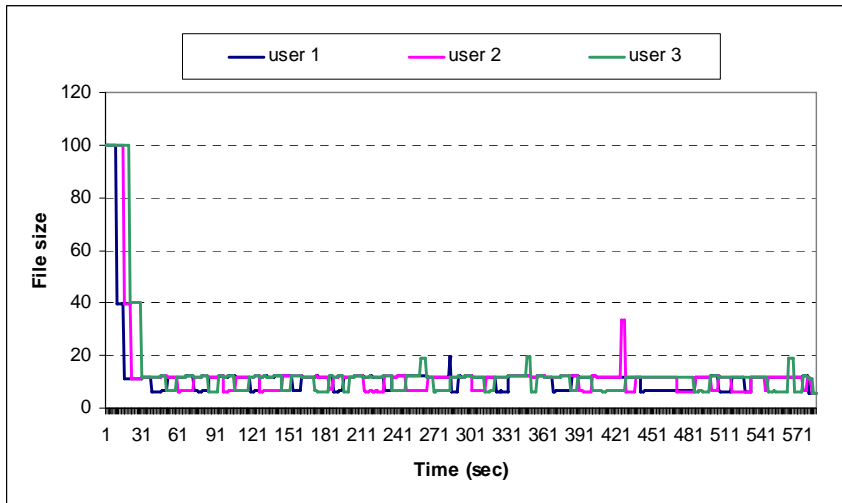
- **Problems of Previous Work (Honeypots):**
 - Honeypots need to receive activities directed against them
 - Bots can run on non-standard ports
 - No simple characteristics of communication channels can be used for detection

- **Our Approach:**
 - Monitoring the change of behaviour in log file sizes
 - Find correlation between these changes
 - Benefits
 - No IRC traffic processing
 - No decryption technique is needed

Data Collection

- Monitoring API socket function calls
- Design
 - Topology
 - Centralized network (IRC)
 - Peer to Peer (P2P) network
 - Algorithm
 - Monitor changes in log file sizes
 - File(1), File(2), File(3), ..., File(n)
 - True, True, True, ... ← Correlated log files
 - False, False, False, ... ← Correlated log files
 - True, False, True, True, False ← Uncorrelated log files

Results – Normal Behaviour



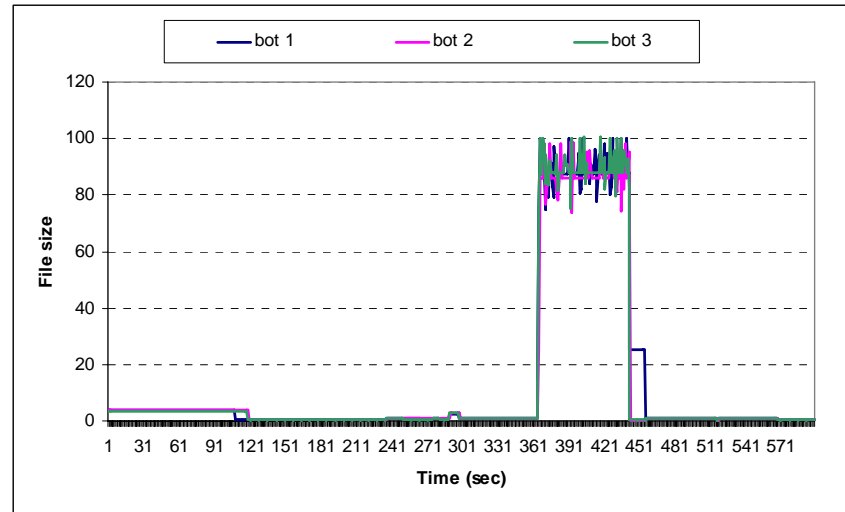
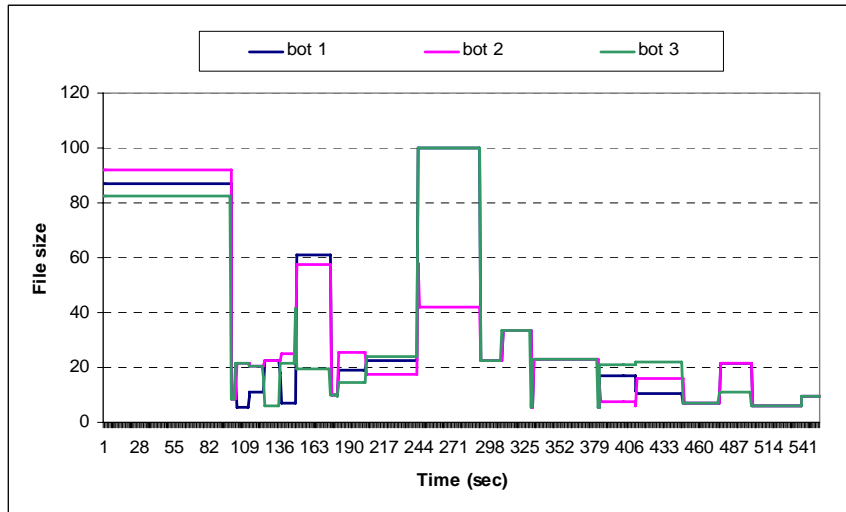
■ Case 1:

- Normal conversation
- Some IRC commands were used
- Result:
 - Low correlation between users

■ Case 2:

- One user send a file to another user
- Some IRC commands were used
- Result:
 - Low correlation between users

Results – Attack Behaviour



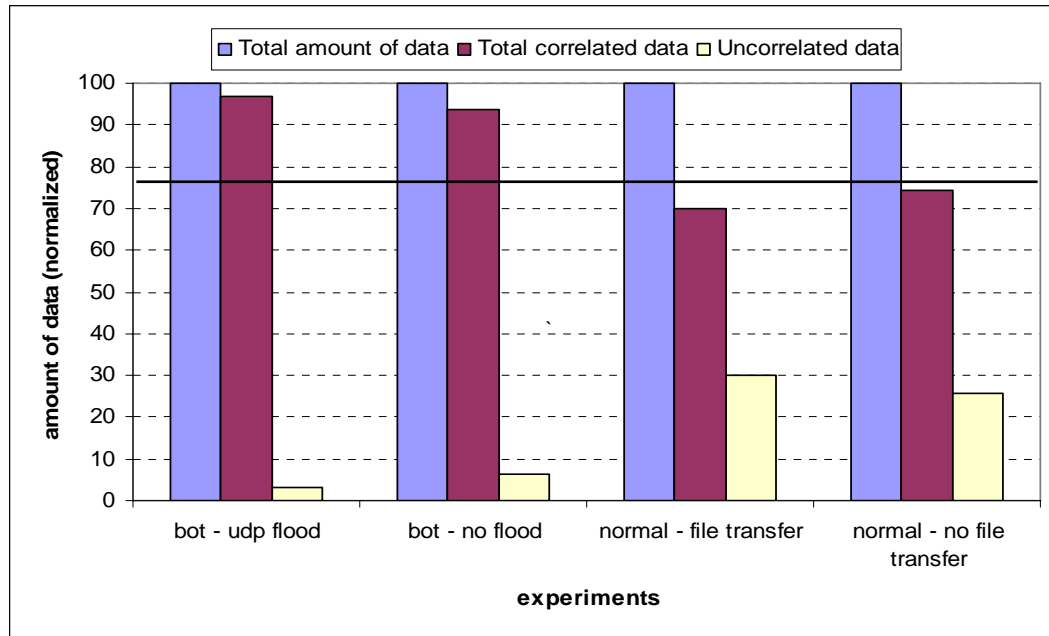
■ Case 1:

- Bots are idle for some time
- No flood commands (UDP, PING)
- Result:
 - Bots respond simultaneously
 - High correlation between bots

■ Case 2:

- Bots are idle for some time
- UDP Flood is used
- Result:
 - Bots respond simultaneously
 - High correlation between bots

Results – Comparison



- Detect Botnet:
 - High number of correlated events that exceed threshold
 - Threshold on average 70% of file size

Conclusion and Future Work

- Successfully detect botnet by monitoring the changes in log file sizes
- Future Work
 - Detect a single bot in a machine by monitoring and classifying API function calls
 - Detect botnets in Peer to Peer (P2P) networks

Thank You

- Any Question!

