



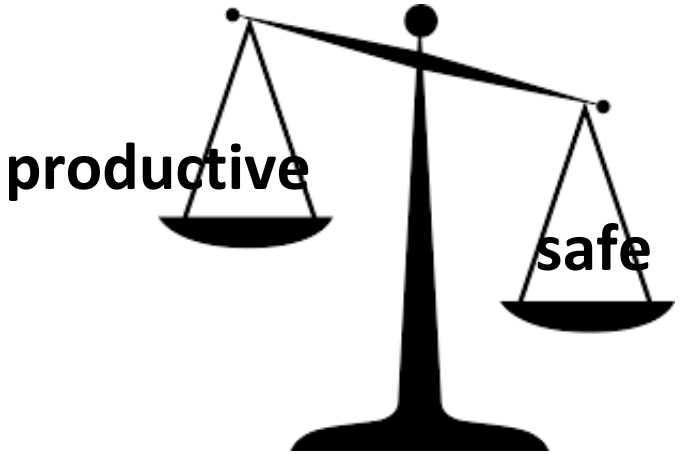


Understanding Malware's Network Behaviors using Fantasm

Xiyue Deng, Hao Shi, Jelena Mirkovic
University of Southern California
Information Sciences Institute
{xiyueden,shihao,mirkovic}@isi.edu

Motivation and Goal

- **Motivation:**
  - Malware is increasingly environment-sensitive and evasive
  - It needs to communicate, but that may be risky
- **Goal:** Support **safe** and **productive** malware analysis
  - Safe - no harm to other Internet hosts from experimentation.
  - Productive - malware continues operating in the experimental environment.



Background

- Malware is an increasing threat to cybersecurity.

Kaspersky Lab: 323,000 New Malware Samples Found Each Day

Credit it to mass-produced malware and better detection through machine learning.

Antivirus provider Kaspersky Lab has revealed that around 323,000 new malware files are being identified each day by its product as opposed to 70,000 files per day in 2011. This, it claims, is an increase by 13,000 per day when compared to last year.

Background (cont.)







- Malware is becoming stealthier and more sophisticated, resisting analysis
  - Anti-debugging
  - Anti-virtualization
  - Relying on network connectivity



- If we could understand malware communication patterns
  - We could decide what is **safe** to let out
  - We would know if some communication is **essential** to keep malware running (e.g., new exploit download or C&C)
  - We would know if we can fake some communications in a controlled environment

Related Work







- System trace and malware binary analysis (Ether, cwsandbox, etc)
  - Comprehensive study using static and dynamic analysis
  - Lacks network behavior analysis
- Sandnet (Rossow et al., 2011)
  - Protocol popularity
  - No understanding of communication purpose
- Morales, et al., 2010
  - Defined 7 activities based on heuristics and analyzed malware for presence of these
  - Limited insight into malware's purpose

Our Approach



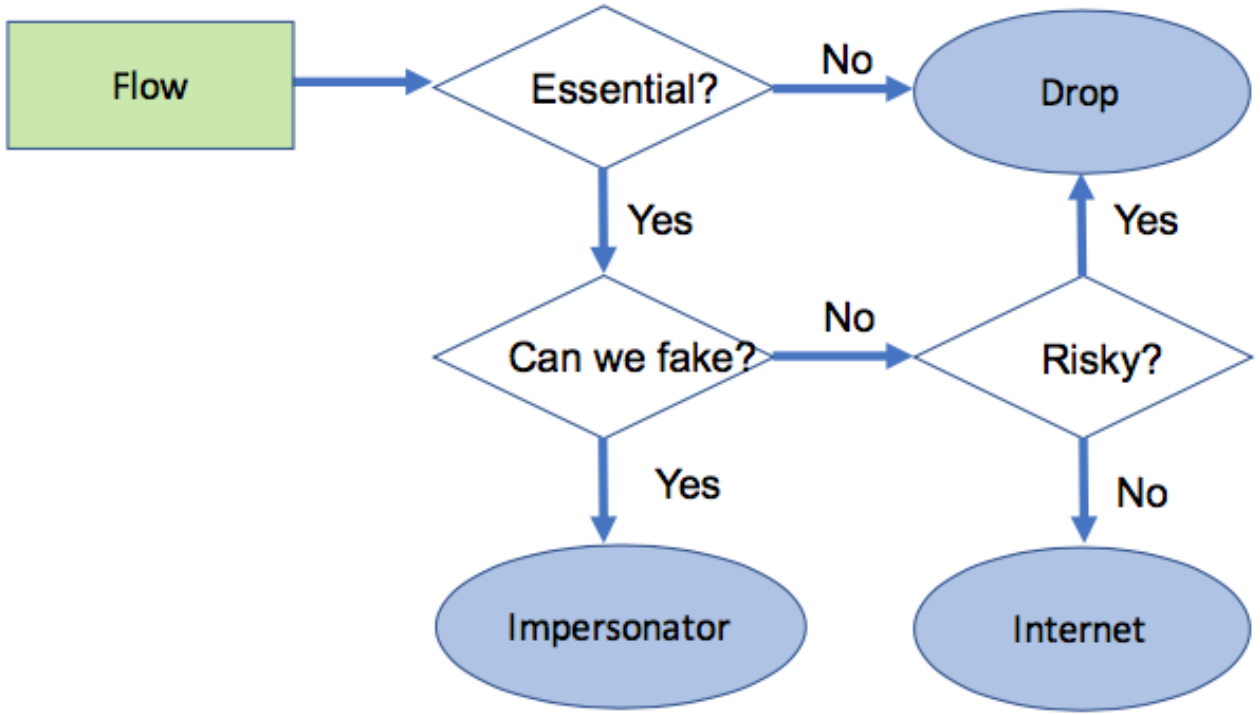




- Build a **partially** contained experimental environment
  - What is the risk of each outgoing flow?
  - How essential is this flow to malware's functioning?
  - How easy it is to fake replies to this flow in the analysis environment?
  - Decide what to do with the flow and, if let out, monitor for risky behavior
- Measure what percentage of malware samples relies on communication to function properly
- Analyze communication patterns
- Use communication patterns to understand malware purpose

Decision Flowchart

- Let out essential and non-risky flows, for which we cannot fake the replies



Decision Flowchart (cont.)







- Essential - malware will not run without it
  - We can identify these over multiple experiment runs
  - If we can fake replies, use impersonators
    - E.g. ICMP ECHO reply, SMTP message receipt
  - If we cannot fake and it is non-risky let it out with strict monitoring.
    - E.g. binary download, receive commands via C&C, resolve DNS names
    - Monitor for and stop DDoS and scanning - too many suspicious, one-sided flows
- Non-essential or risky
  - Drop

Fantasm Environment







- Bare metal machines
  - Physical machines, no debugger or VM
  - Change default interface/account naming to real names
- Clean state
  - Experiment will be reset to initial state after it is done
  - Avoid damaged system which may restrict activity of new malwares
  - Avoid interaction between malwares
- Deployed in DeterLab (http://isi.deterlab.net)
  - Automatic setup and OS reload
  - Customized network topology
  - Restricted/configurable Internet connection

Malware Dataset



Open Malware 13 days in Mar, 2017 — 29,319 — Has report from Virus Total — 28,495 — 50+% tags malicious — 19,007 — choose diverse samples — 2,994

Translate diverse AV tags into a few *concise* tags

Category	**Sample (#/%)**	**Category**	**Sample (#/%)**
Virus	6,126 / 32%	Riskware	409 / 2%
Trojan	6,040 / 32%	Backdoor	197 / 1%
Worm	4,227 / 22%	Bot	45 / <1%
Downloader	984 / 5%	Ransomware	17 / < %
Adware	962 / 5%	**Total**	**19,007**

Malware Experiment







- Evaluate under full and partial containment
  - Each sample ran for 5 minutes
  - 1,737 (58%) samples exhibited some network activity
  - Other samples were dormant
    - maybe they needed a trigger
    - or are just slow acting
    - or need a different OS/app setup
    - or have expired

Fantasm Meets Its Goals







- **Productive** experimentation: 1,354 samples (78%) generated more flows under partial than under containment
- **Safe** experimentation: no complaints, not blacklisted (12 weeks)

Handling	Flow count	Percentage
Can fake replies using impersonator	9,270,831	99.64%
Had to let out (essential and non-risky)	2,295	0.02%
Dropped (non-essential or risky)	30,957	0.33%

Understanding Purpose

- Protocol popularity

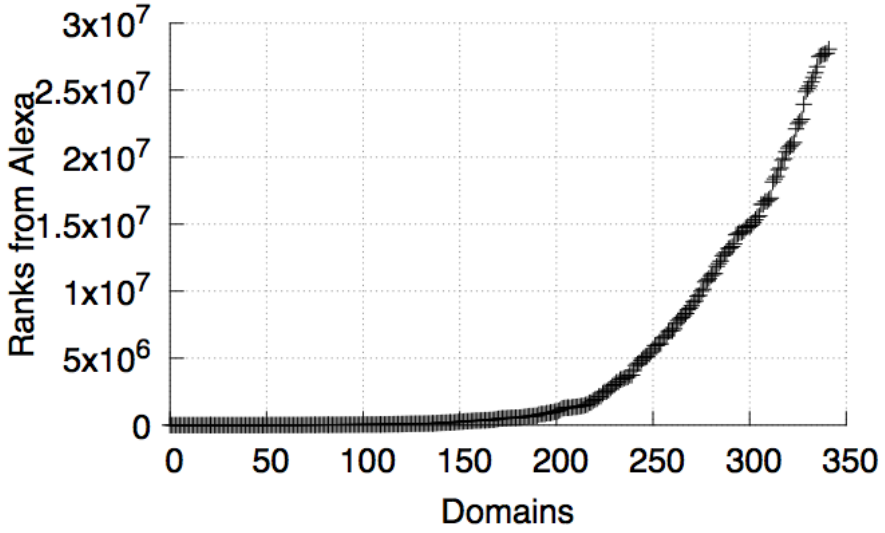
Protocol	Samples
DNS	1,081 (62%)
ICMP echo	818 (47%)
HTTP	600 (35%)
Port 65530	237 (14%)
HTTPS	173 (10%)
SMTP	74 (4%)

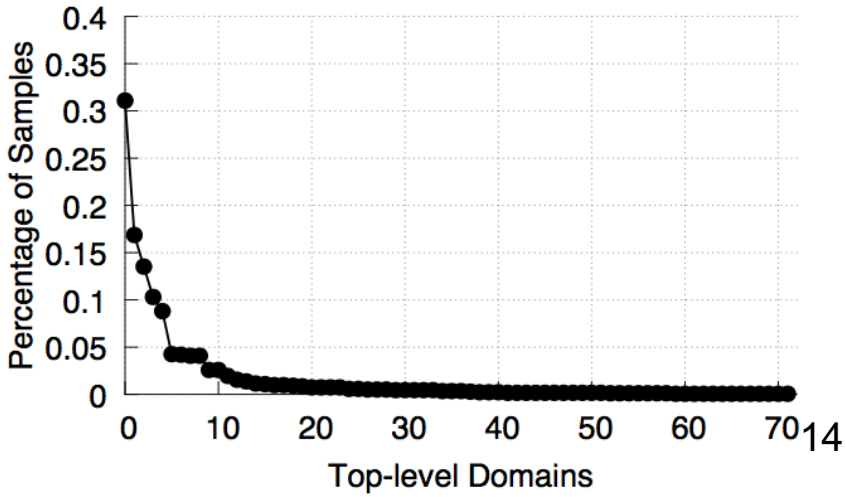
Protocol	Samples
port 1042	65 (4%)
port 799	33 (2%)
port 6892	25 (1%)
port 11110	17 (1%)
port 11180	17 (1%)
FTP	12 (1%)

Understanding Purpose

- Popularity of DNS names
  - Alexa has record for 341 (6%)
  - 1% have rank < 10K
  - 5% have rank [10K, 1M]
  - 94% are not on Alexa

Top-level	Samples	Second-level	Samples
.com	540/31%	google.com	187/35%
		msrl.com	73/14%
		ide.com	73/14%
.pl	293/17%	zief.pl	244/83%
		brenz.pl	26/9%
		ircgalaxy.pl	22/8%
.net	235/14%	secureserver.net	73/31%
		surf.net	68/29%
		aol.net	65/28%





NetDigest

- Concise and human-readable representation of malware communication
- Digest field excerpt

Protocol	Attribute	Value Type
All	LocalPort	integer
	NumPktSent	integer
	NumPktRecv	integer
	PktSentTS	float_list
	PktRecvTs	float_list
	PayloadSize	integer_list

Protocol	Attribute	Value Type
HTTP/FTP	Server	IP_address
	Proactive	boolean
	GotResponse	boolean
	Code	integer
	Download	file_type
	Upload	file_type

NetDigest Example








```
1488068895.052901: DNS - [Server: 10.1.1.3], [A: ic-dc.deliverydlcenter.com],
                    [CNAME: N/A], [A: 52.85.83.81, 52.85.83.112, 52.85.83.132,
                    52.85.83.4, 52.85.83.96, 52.85.83.56, 52.85.83.32,
                   52.85.83.37]
1488068895.154335: HTTP - [Server: 52.85.83.81], [Proactive: True],
                    [GotResponse: True], [Download: blob], [Download: .png],
                    [Download: blob]
1488068895.948346: HTTP - [Server: 52.85.83.81], [Proactive: True],
                   [GotResponse: True]
1488068896.767094: DNS - [Server: 10.1.1.3], [A: www.1-1ads.com], [CNAME:
                    n135adserv.com], [A: 212.124.124.178]
...
```

Malware Classification

- Binary code is polymorphic but communication is harder to morph
- 83 features from NetDigest (excerpt below)

Categories	Subgroups	Features (83 in total)
Packet	Header	Distinct no. of: IPs, countries, continents, local ports, etc.
	Payload	Total size (bytes); Sent/received: total #, min, max, mean, stdev
	Statistics	Packets: total #, rate; Time interval: min, max, mean, stdev
Session	Direction	Proactive or passive
	Result	Succeeded or failed
	Statistics	Total SYN #; Session# per IP: min, max, mean, stdev
...	...	...

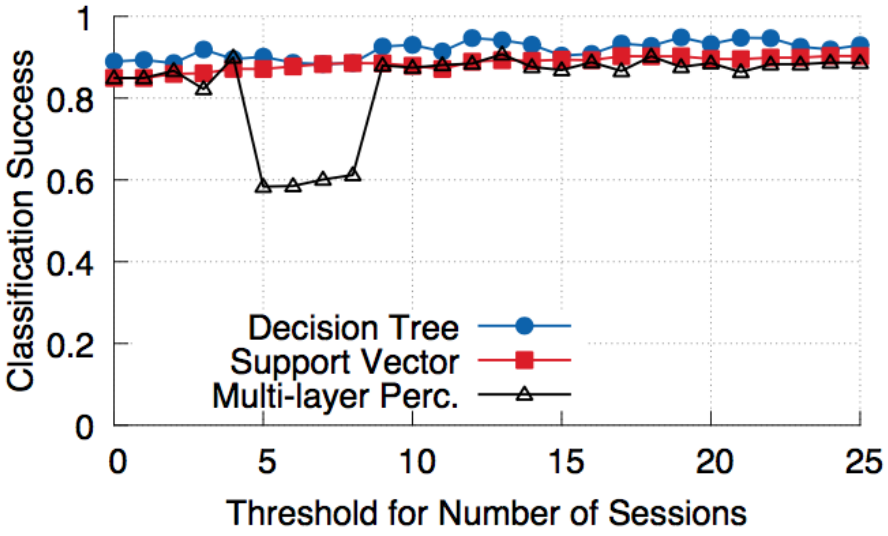
Malware Classification

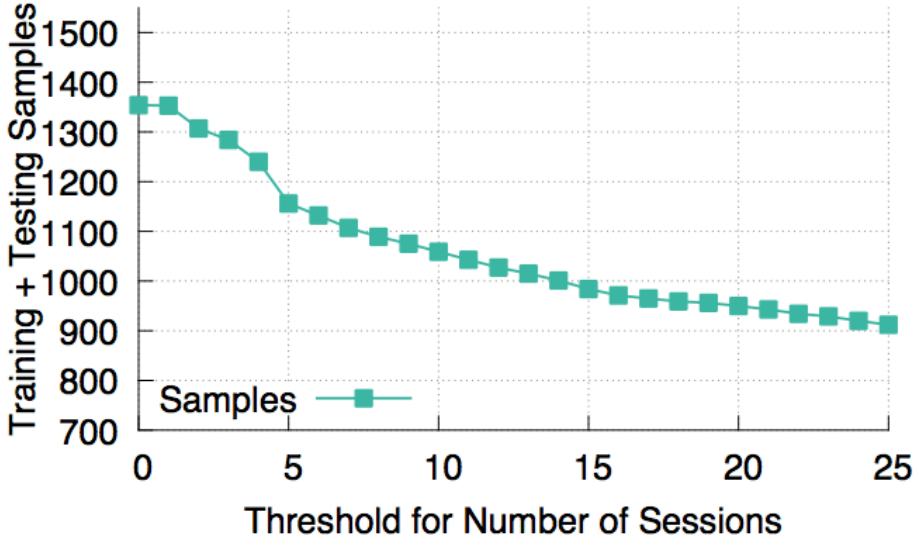
- Apply ML techniques to extracted features, learn concise labels
  - 80% as training data, 20% for testing
  - Apply 3 popular classification methods
    - Decision trees, SVM, multi-layer perceptron
  - Same sample may legitimately match multiple labels if it exhibits multiple behaviors

Algorithms	Top label matched	Top 2 labels matched	Top 3 labels matched
Decision Trees	242 / 89%	257 / 94%	259 / 95%
SVM	231 / 85%	259 / 95%	265 / 97%
Multi-layer Perceptron	231 / 85%	257 / 94%	262 / 96%

Malware Classification (cont.)

- Classification accuracy and sessions
  - Gradually drop samples with fewer than N sessions, N = [1, 25]
  - Same 80% training / 20% testing split
  - Slight increase of accuracy (88% -> 93%)





Conclusion







- Investigated how essential Internet connectivity is for malware activity
  - 58% of samples showed network activity in first 5 minutes of the experiment
  - 78% will become dormant in full containment
- Built an experiment environment with partial containment for safe and productive malware analysis
- Investigated popular communication patterns
- Showed how we can achieve accurate malware classification by leveraging communication patterns

For More Information







Xiyue Deng <xiyueden@isi.edu>
Hao Shi <shihao@isi.edu>
Jelena Mirkovic <mirkovic@isi.edu>

Project page: http://steel.isi.edu/Projects/malwareExp/


STEEL
Security Research Lab@ISI

ISI
Information Sciences Institute

USC
Viterbi
Information
Sciences Institute

Thank you!