BotMiner: Clustering Analysis of Network Traffic for Protocol- and Structure-Independent Botnet Detection

Guofei Gu[1,2], Roberto Perdisci[3], Junjie Zhang[1], and Wenke Lee[1]

[1]Georgia Tech [3]Damballa, Inc.

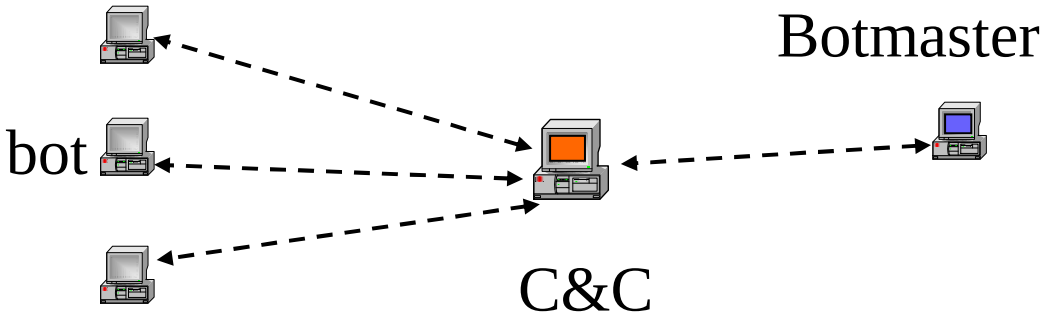
[2]Texas A&M University

Roadmap

- Introduction
  - Botnet problem
  - Challenges for botnet detection
  - Related work
- BotMiner
  - Motivation
  - Design
  - Evaluation
- Conclusion

What Is a Bot/Botnet?

- Bot
  - A malware instance that runs autonomously and automatically on a compromised computer (zombie) without owner's consent
  - Profit-driven, professionally written, widely propagated
- Botnet (Bot Army): network of bots controlled by criminals
  - Definition: "A coordinated group of malware instances that are controlled by a botmaster via some C&C channel"
  - Architecture: centralized (e.g., IRC,HTTP), distributed (e.g., P2P)
  - "25% of Internet PCs are part of a botnet!" (- Vint Cerf)



Botnets are used for …

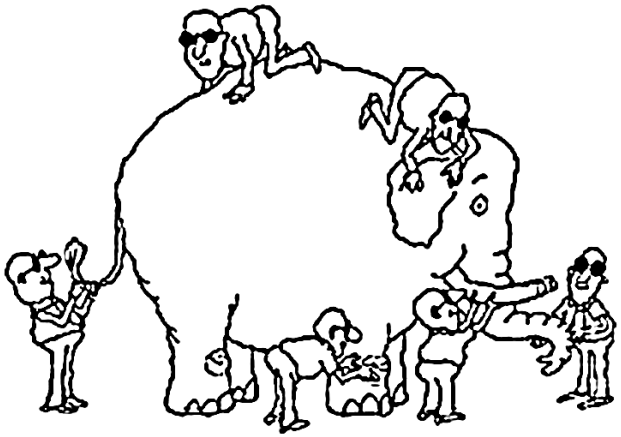
- All DDoS attacks
- Spam
- Click fraud
- Information theft
- Phishing attacks
- Distributing other malware, e.g., spyware

Challenges for Botnet Detection

- Bots are stealthy on the infected machines
  - We focus on a network-based solution
- Bot infection is usually a multi-faceted and multi-phased process
  - Only looking at one specific aspect likely to fail
- Bots are dynamically evolving
  - Static and signature-based approaches may not be effective
- Botnets can have very flexible design of C&C channels
  - A solution very specific to a botnet instance is not desirable

Why Existing Techniques Not Enough?

- Traditional AV tools
  - Bots use packer, rootkit, frequent updating to easily defeat AV tools
- Traditional IDS/IPS
  - Look at only specific aspect
  - Do not have a big picture



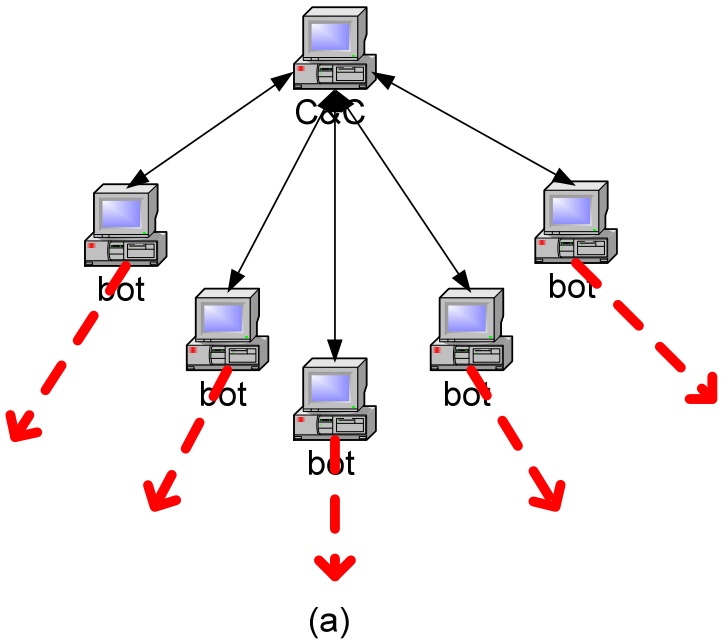
- Honeypot
  - Not a good botnet detection tool

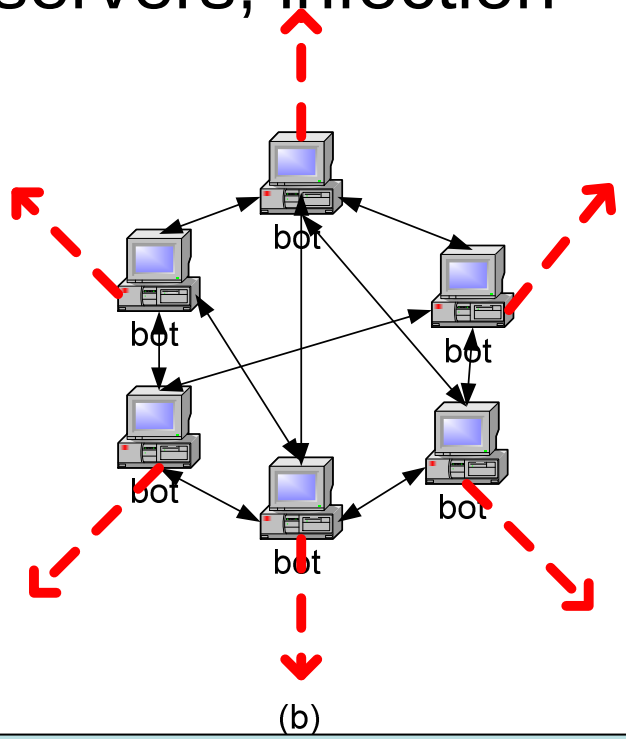
Existing Botnet Detection Work

- [Binkley,Singh 2006]: IRC-based bot detection combine IRC statistics and TCP work weight
- Rishi [Goebel, Holz 2007]: signature-based IRC bot nickname detection
- [Livadas et al. 2006, Karasaridis et al. 2007]: (BBN, AT&T) network flow level detection of IRC botnets (IRC botnet)
- BotHunter [Gu etal Security'07]: dialog correlation to detect bots based on an infection dialog model
- BotSniffer [Gu etal NDSS'08]: spatial-temporal correlation to detect centralized botnet C&C
- TAMD [Yen, Reiter 2008]: traffic aggregation to detect botnets that use a centralized C&C structure

Why BotMiner?

- Botnets can change their C&C content (encryption, etc.), protocols (IRC, HTTP, etc.), structures (P2P, etc.), C&C servers, infection models …





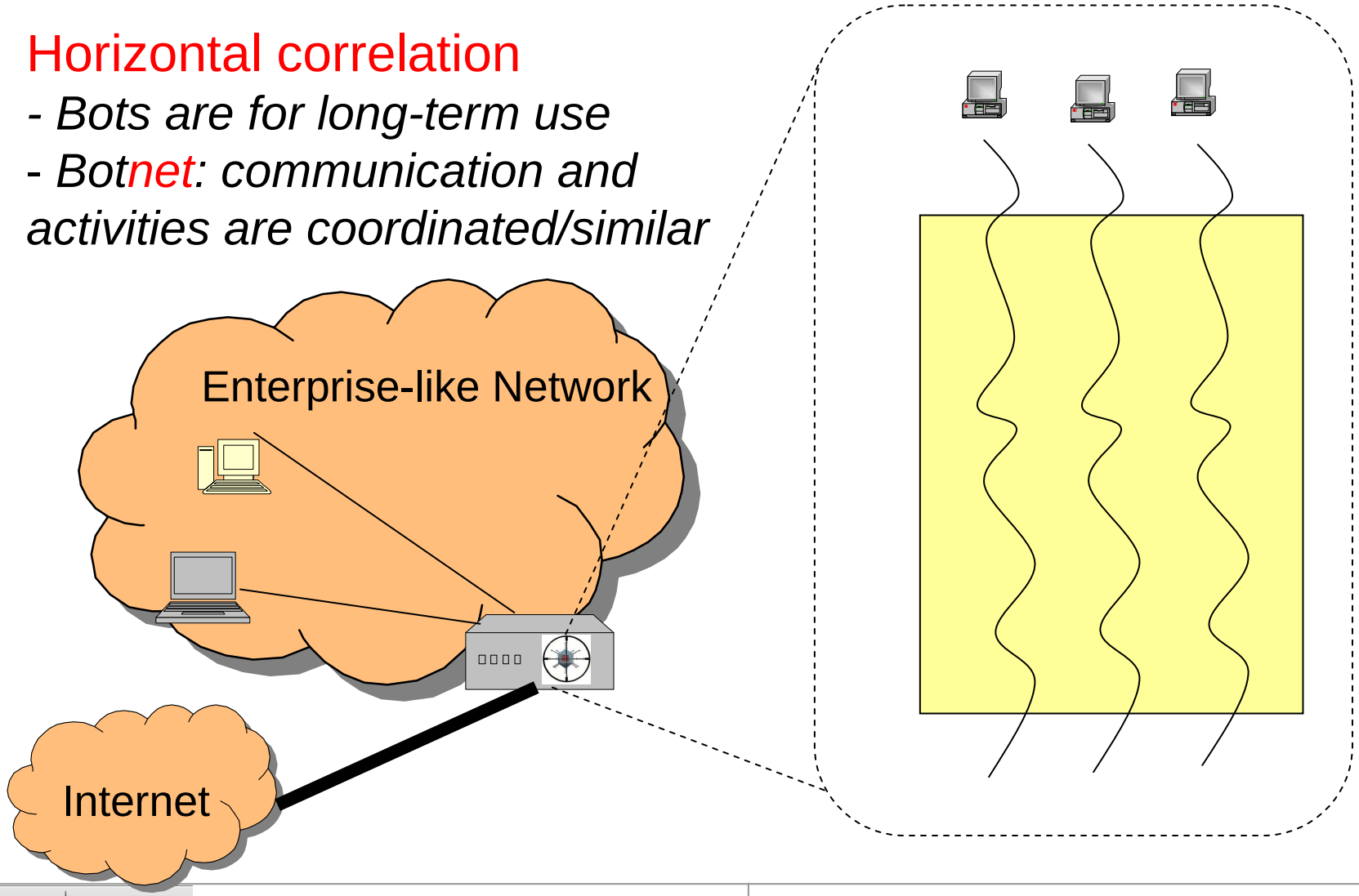
Example: Nugache, Storm, …

BotMiner: Protocol- and Structure-Independent Detection

Horizontal correlation

- Bots are for long-term use

- Botnet: communication and activities are coordinated/similar

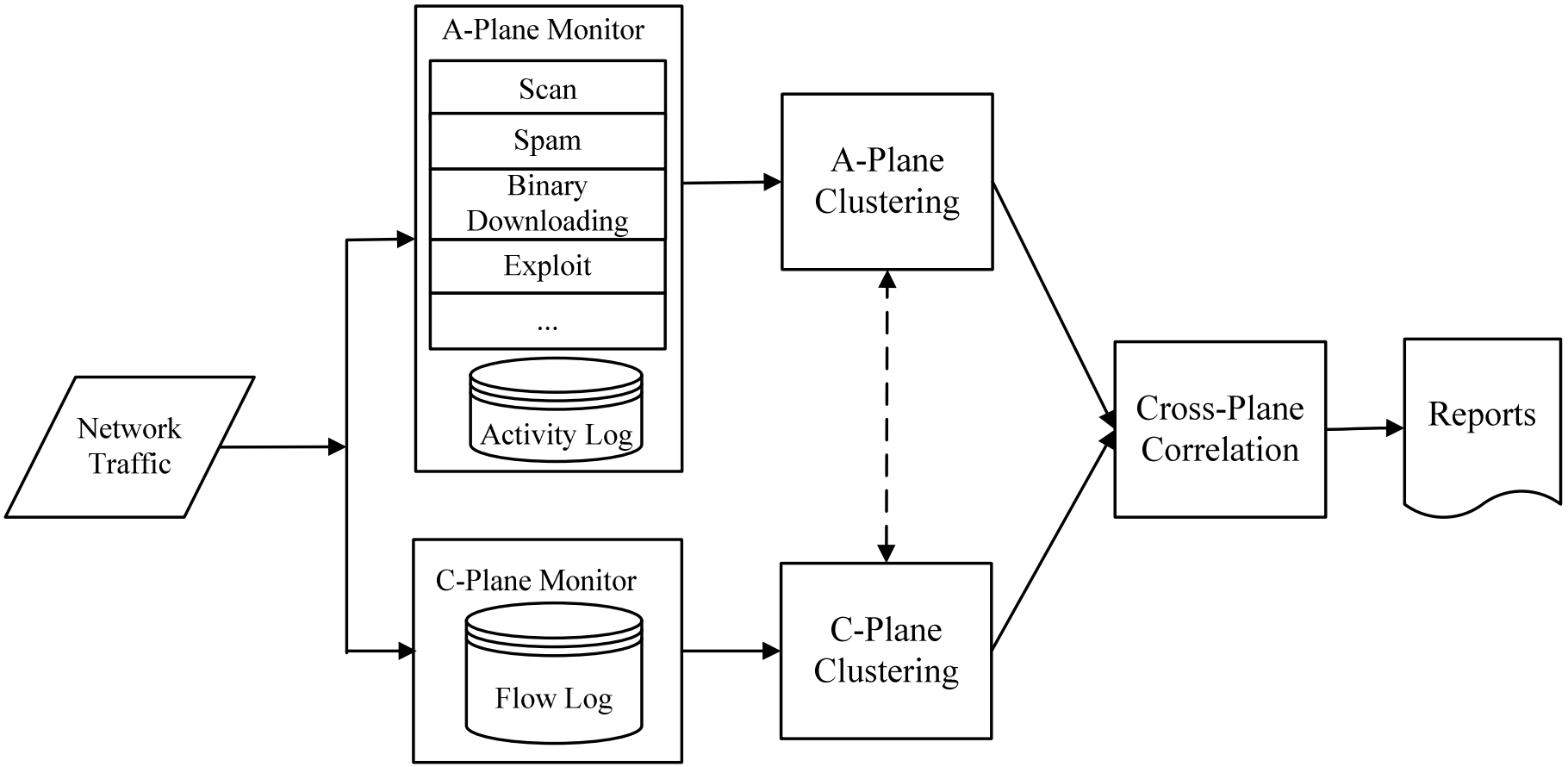


Revisit the Definition of a Botnet

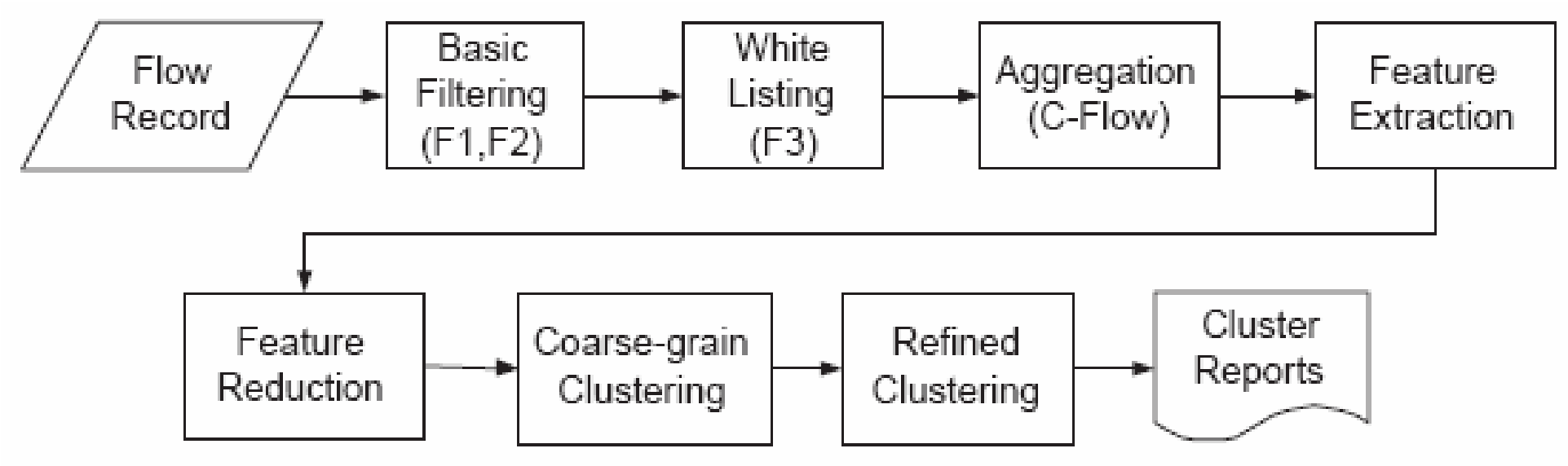
- "A coordinated group of malware instances that are controlled by a botmaster via some C&C channel"

- We need to monitor two planes
  - C-plane (C&C communication plane): "who is talking to whom"
  - A-plane (malicious activity plane): "who is doing what"

BotMiner Architecture



BotMiner C-plane Clustering



- What characterizes a communication flow (C-flow) between a local host and a remote service?
  - <protocol, srcIP, dstIP, dstPort>

How to Capture "Talking in What Kind of Patterns"?

- Temporal related statistical distribution information in
  - BPS (bytes per second)
  - FPH (flow per hour)

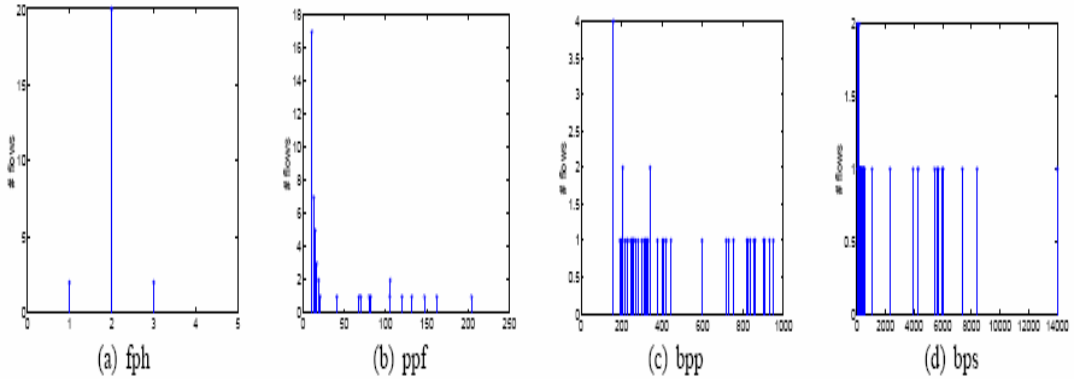


Figure 4: Visit pattern (shown in distribution) to Google from a randomly chosen normal client.

- Spatial related statistical distribution information in
  - BPP (bytes per packet)
  - PPF (packet per flow)

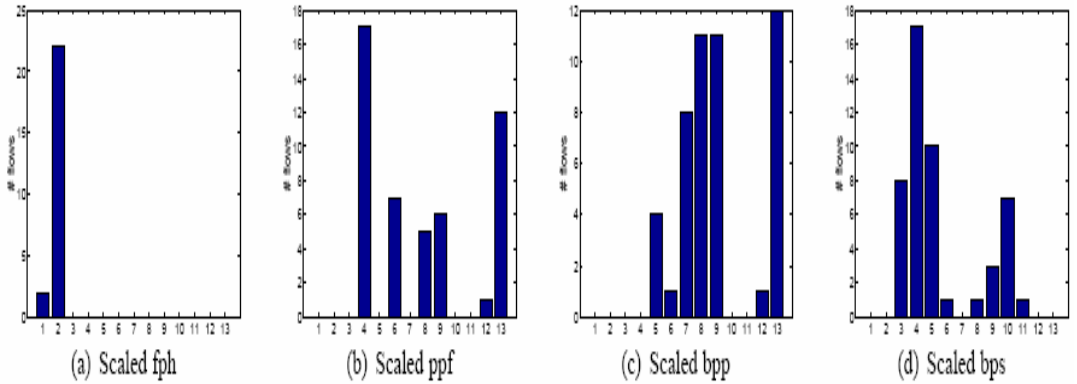
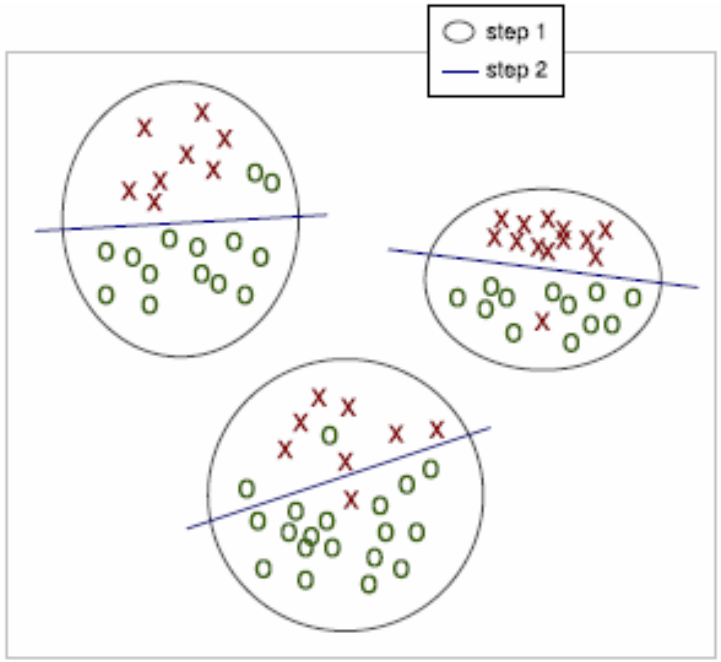


Figure 5: Scaled visit pattern (shown in distribution) to Google for the same client in Figure 4.

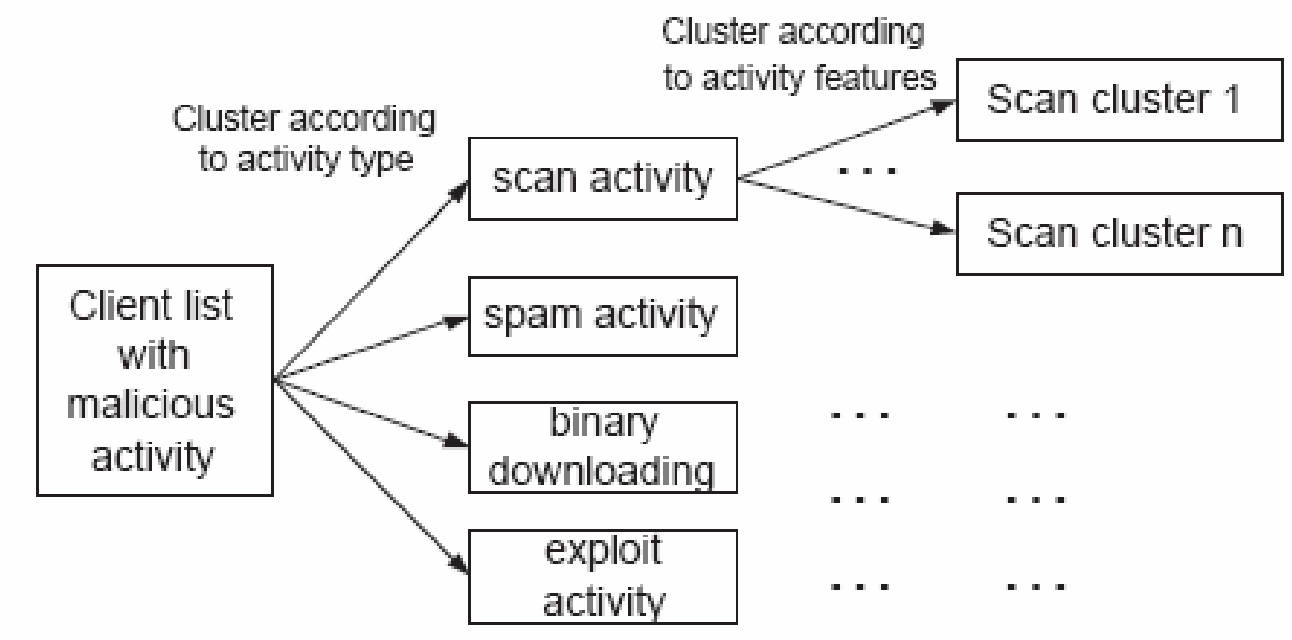
Two-step Clustering of C-flows

- Why multi-step?
- How?
  - Coarse-grained clustering
    - Using reduced feature space: mean and variance of the distribution of FPH, PPF, BPP, BPS for each C-flow (2*4=8)
    - Efficient clustering algorithm: X-means
  - Fine-grained clustering
    - Using full feature space (13*4=52)



- What's left?

A-plane Clustering

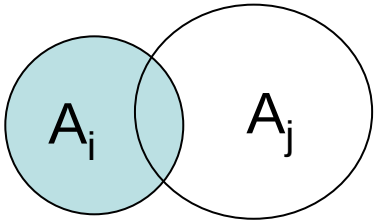


- Capture “activities in what kind of patterns”

Cross-plane Correlation

- Botnet score *s(h)* for every host *h*

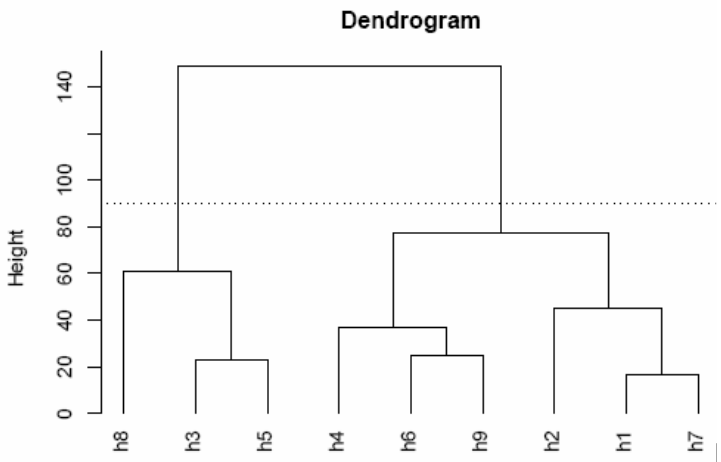
$$s(h) = \sum_{\substack{i,j \\ j>i \\ t(A_i)\neq t(A_j)}} w(A_i)w(A_j)\frac{|A_i \cap A_j|}{|A_i \cup A_j|} + \sum_{i,k} w(A_i)\frac{|A_i \cap C_k|}{|A_i \cup C_k|}$$



- Similarity score between host h_i and h_j

$$sim(h_i, h_j) = \sum_{k=1}^{m_B} I(b_k^{(i)} = b_k^{(j)}) + I(\sum_{k=m_B+1}^{m_B+n_B} I(b_k^{(i)} = b_k^{(j)}) \geq 1)$$

Two hosts in the same A-clusters and in at least one common C-cluster are clustered together



- Hierarchical clustering

Evaluation Traces

Trace	Pkts	Flows	Filtered by F1	Filtered by F2	Filtered by F3	Flows after filtering	C-flows (TCP/UDP)
Day-1	5,178,375,514	23,407,743	20,727,588	939,723	40,257	1,700,175	66,981 / 132,333
Day-2	7,131,674,165	29,632,407	27,861,853	533,666	25,758	1,211,130	34,691 / 96,261
Day-3	9,701,255,613	30,192,645	28,491,442	513,164	24,329	1,163,710	39,744 / 94,081
Day-4	14,713,667,172	35,590,583	33,434,985	600,901	33,958	1,520,739	73,021 / 167,146
Day-5	11,177,174,133	56,235,380	52,795,168	1,323,475	40,016	2,076,721	57,664 / 167,175
Day-6	9,950,803,423	75,037,684	71,397,138	1,464,571	51,931	2,124,044	59,383 / 176,210
Day-7	10,039,871,506	109,549,192	105,530,316	1,614,158	56,688	2,348,030	55,023 / 150,211
Day-8	11,174,937,812	96,364,123	92,413,010	1,578,215	60,768	2,312,130	56,246 / 179,838
Day-9	9,504,436,063	62,550,060	56,516,281	3,163,645	30,581	2,839,553	25,557 / 164,986
Day-10	11,071,701,564	83,433,368	77,601,188	2,964,948	27,837	2,839,395	25,436 / 154,294

Trace	Size	Duration	Pkt	TCP/UDP flows	Botnet clients	C&C server
Botnet-IRC-rbot	169MB	24h	1,175,083	180,988	4	1
Botnet-IRC-sdbot	66KB	9m	474	19	4	1
Botnet-IRC-spybot	15MB	32m	180,822	147,945	4	1
Botnet-IRC-N	6.4MB	7m	65,111	5635	259	1
Botnet-HTTP-1	6MB	3.6h	65,695	2,647	4	1
Botnet-HTTP-2	37MB	19h	395,990	9,716	4	1
Botnet-P2P-Storm	1.2G	24h	59,322,490	5,495,223	13	P2P
Botnet-P2P-Nugache	1.2G	24h	59,322,490	5,495,223	82	P2P

Evaluation Results: False Positives

Trace	Step-1 C-clusters	Step-2 C-clusters	A-plane logs	A-clusters	False Positive Clusters	FP Rate
TCP/UDP						
Day-1	1,374	4,958	1,671	1	0	0 (0/878)
Day-2	904	2,897	5,434	1	1	0.003 (2/638)
Day-3	1,128	2,480	4,324	1	1	0.003 (2/692)
Day-4	1,528	4,089	5,483	4	4	0.01 (9/871)
Day-5	1,051	3,377	6,461	5	2	0.0048 (4/838)
TCP only						
Day-6	1,163	3,469	6,960	3	2	0.008 (7/877)
Day-7	954	3,257	6,452	5	2	0.006 (5/835)
Day-8	1,170	3,226	8,270	4	2	0.0091 (8/877)
Day-9	742	1,763	7,687	2	0	0 (0/714)
Day-10	712	1,673	7,524	0	0	0 (0/689)

Evaluation Results: Detection Rate

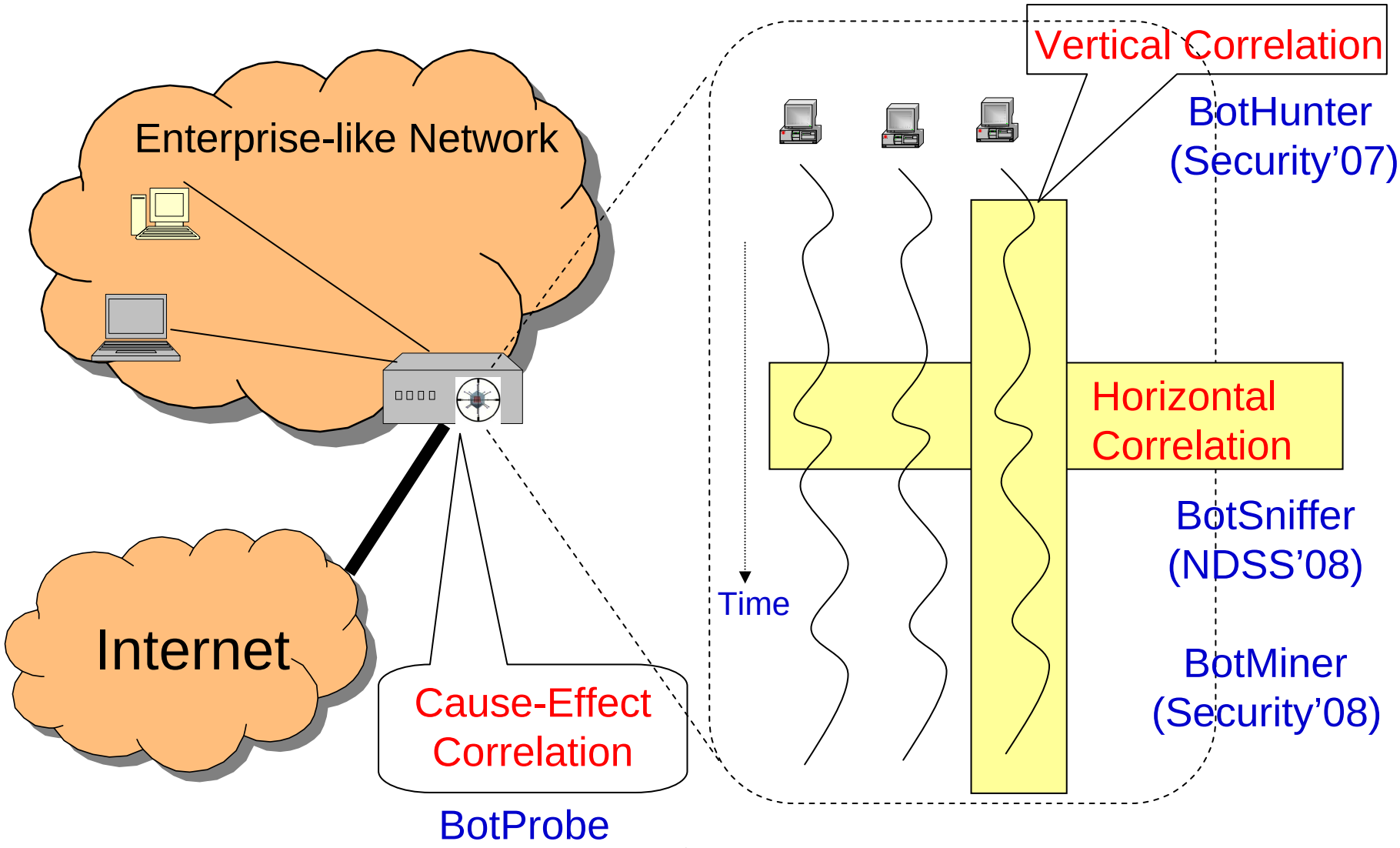
Botnet	Number of Bots	Detected?	Clustered Bots	Detection Rate	False Positive Clusters/Hosts	FP Rate
IRC-rbot	4	YES	4	100%	1/2	0.003
IRC-sdbot	4	YES	4	100%	1/2	0.003
IRC-spybot	4	YES	3	75%	1/2	0.003
IRC-N	259	YES	258	99.6%	0	0
HTTP-1	4	YES	4	100%	1/2	0.003
HTTP-2	4	YES	4	100%	1/2	0.003
P2P-Storm	13	YES	13	100%	0	0
P2P-Nugache	82	YES	82	100%	0	0

Summary and Future Work

- BotMiner
  - New botnet detection system based on Horizontal correlation
  - Independent of botnet C&C protocol and structure
  - Real-world evaluation shows promising results

- Future work
  - More efficient clustering, more robust features
  - New faster detection system using active techniques
    - BotMiner: offline correlation, and requires a relatively long time for detection
    - BotProbe: fast detection by observing at most one round of C&C
  - New real-time solution for very high speed and very large networks

Correlation-based Botnet Detection Framework



Limitation and Discussion

- Evading C-plane monitoring and clustering
  - Misuse whitelist
  - Manipulate communication patterns
- Evading A-plane monitoring and clustering
  - Very stealthy activity
  - Individualize bots' communication/activity
- Evading cross-plane analysis
  - Extremely delayed task