Network Monitoring

- Attack traffic used to be well-defined and obvious, e.g.,
  - Payload contains exploit to a known vulnerability
  - Volume/rate suggests DoS, Spam, etc.

- Firewalls and network intrusion detection systems
  - Designed to identify attack traffic

Advanced Network Monitoring

Traditional firewalls/NIDS

- Are bypassed by mobile devices compromised while outside network perimeter

Attack traffic is now very subtle

- E.g., botnet HTTP-based command and control (C&C) traffic looks like normal web traffic

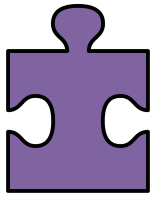
Need more advanced network monitoring

- Identify traffic beyond obvious exploit/attacks
- In particular, botnet detection systems

BOT Quiz

Fill in the blanks with the correct answers.

A Bot is often called a zombie because it is a compromised computer controlled by malware without the consent and knowledge of the user.



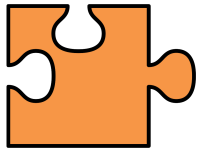
Botnet Quiz

Fill in the blanks with the correct answers.

A Botnet is a network of bots controlled by a Bot Master / *attacker*.

More precisely, a coordinated group of malware instances that are controlled via command and control (C&C) channels. C&C architectures: centralized (e.g., IRC, HTTP), distributed (e.g., P2P)

It is a key platform for fraud and other for-profit exploits.



Botnet Tasks Quiz

Select all the tasks that botnets commonly perform:

- [x] More than 95% of all spam
- [x] All distributed denial of service (DDoS) attacks
- [x] Click fraud
- [x] Phishing & pharming attacks
- [x] Key logging & data/identity theft
- [x] Distributing other malware, e.g., spyware
- [x] Anonymized terrorist & criminal communication

Why Traditional Security Measures Fail

Traditional Anti-Virus Tools

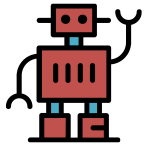
Traditional Anti-Virus Tools → Bots use packer, rootkit, frequent updating to easily defeat AV tools

Traditional IDS/IPS

- Look at only specific aspect
  - e.g. payload with exploit
- Do not have a big picture
  - Bots are for long-term use

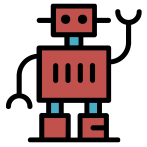
Honeypot

- Not scalable, mostly passively waiting
- Bots can detect/discover honeypot/honeynet
- Not a good botnet detection tool



Botnet Detection: Challenges

- Bots are stealthy on the infected machines
  - E.g., rootkit hides the malware
- 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



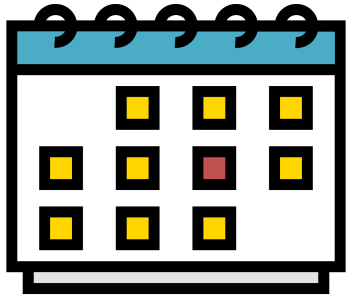
Botnet Detection: Guidelines

Distinguish botnet activities from normal network traffic

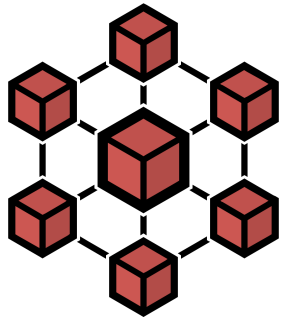
- Bot: non-human
- Net: bots are connected; activities are coordinated

Distinguish botnets from other (older) attacks

For profit (resources)

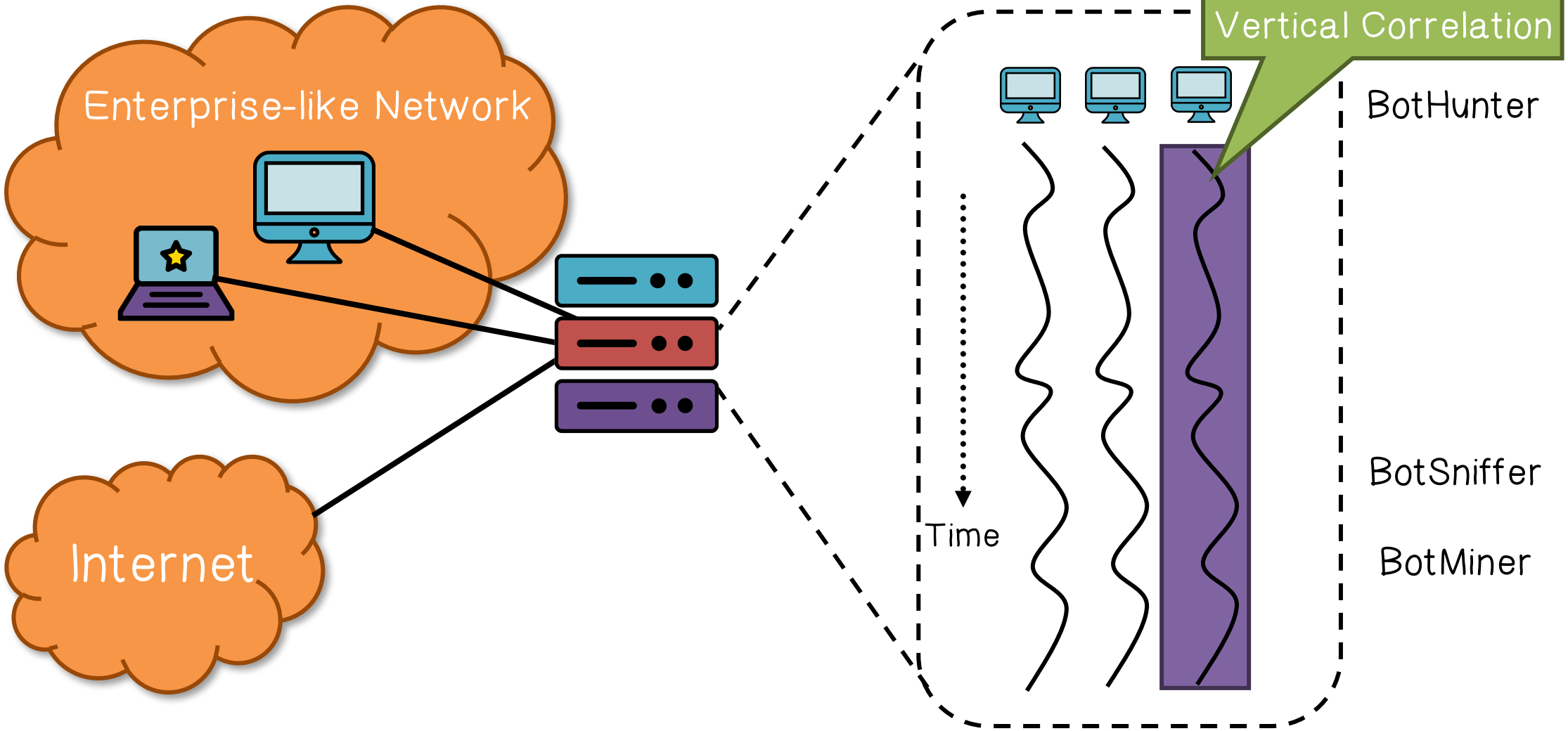


Long-term use (updates)

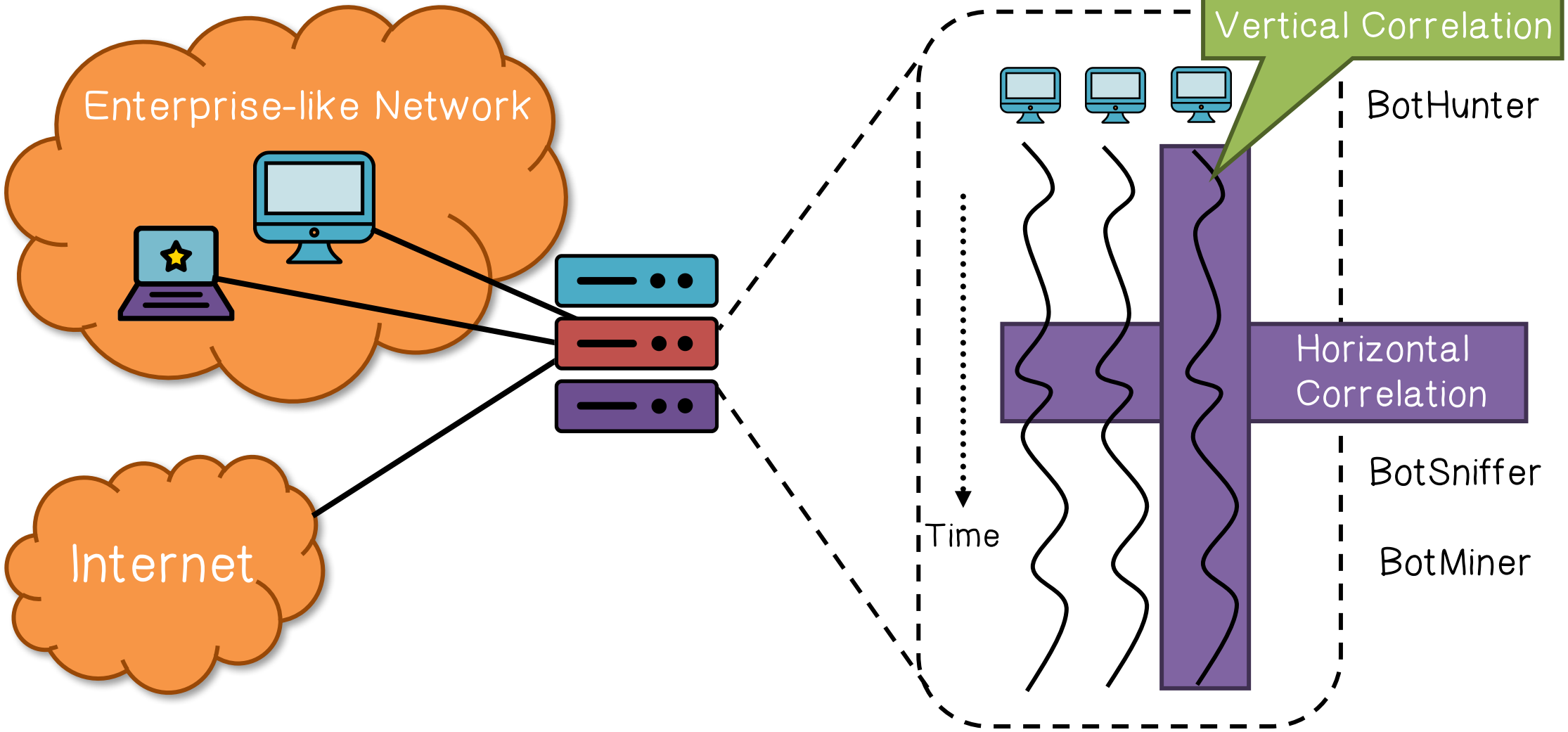


Net (coordination)

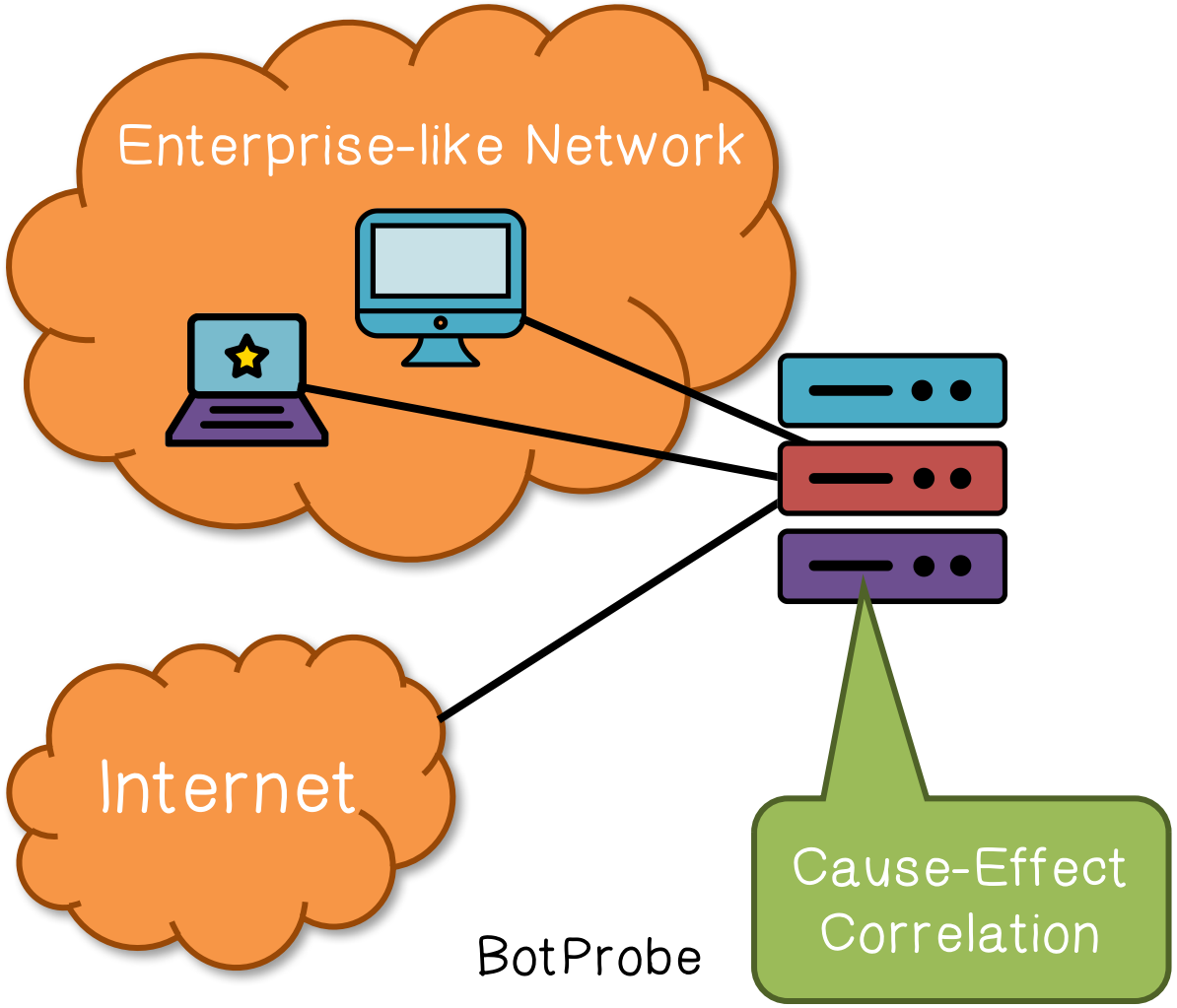
Botnet Detection: Enterprise Networks



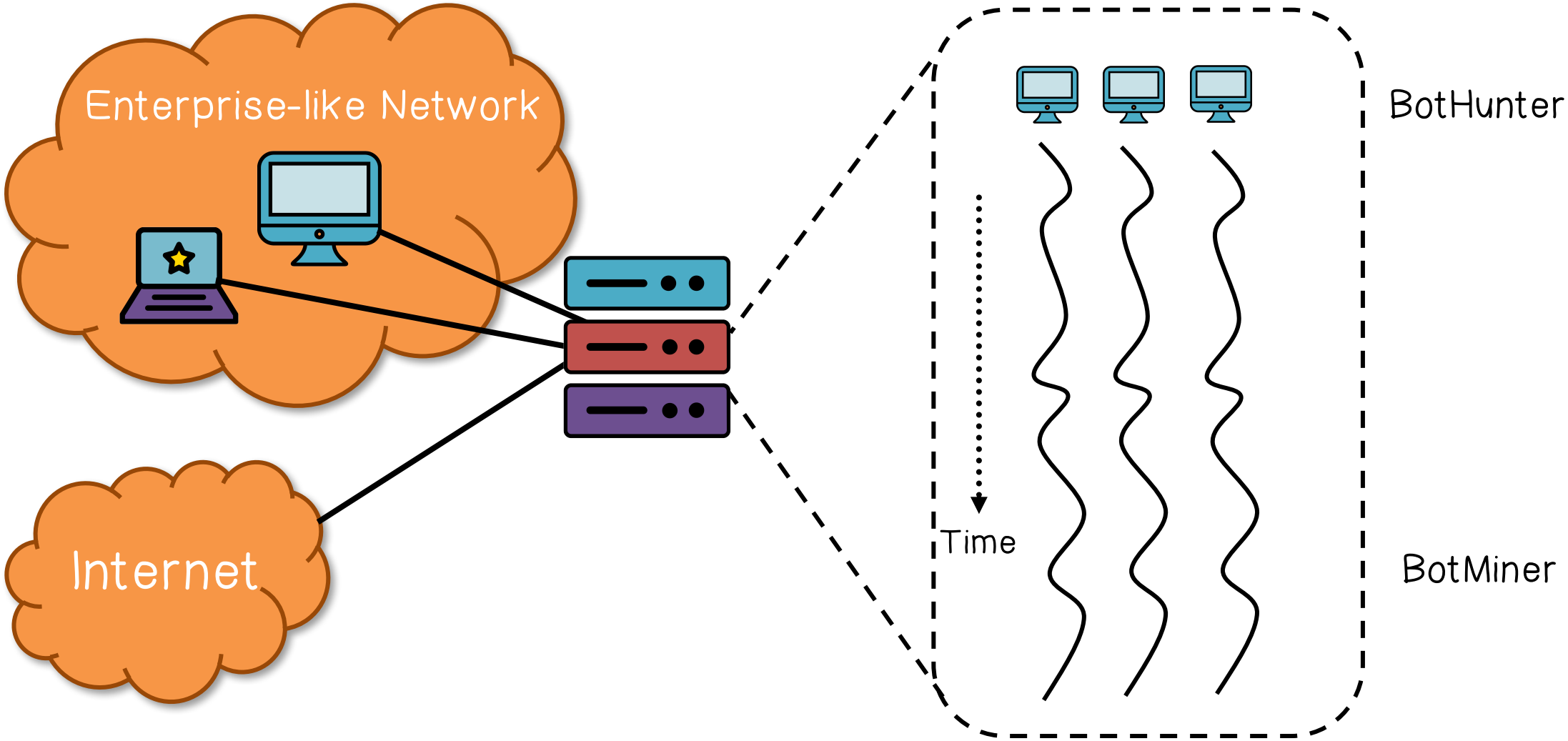
Botnet Detection: Enterprise Networks



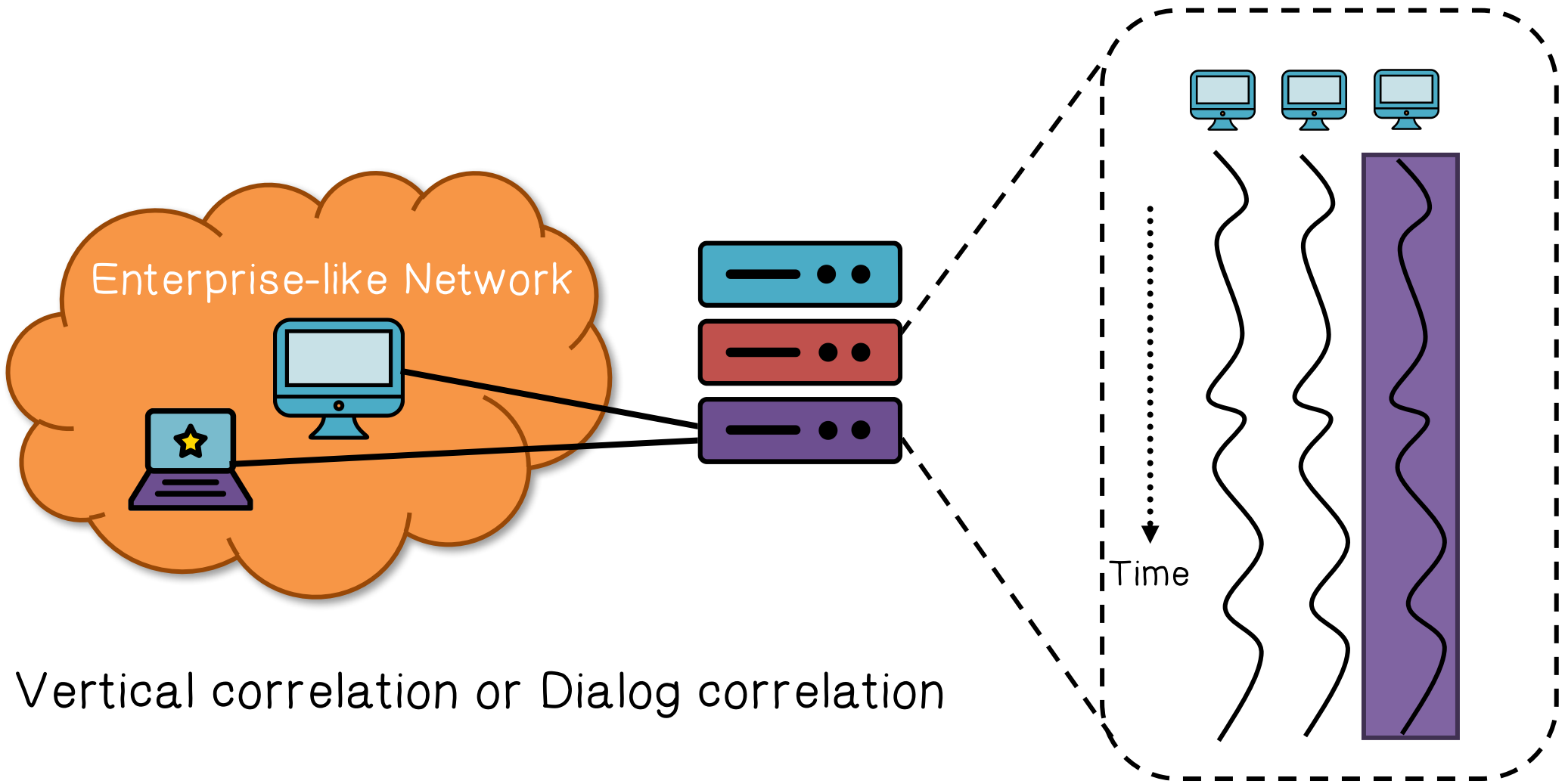
Botnet Detection: Enterprise Networks



Botnet Detection: Enterprise Networks

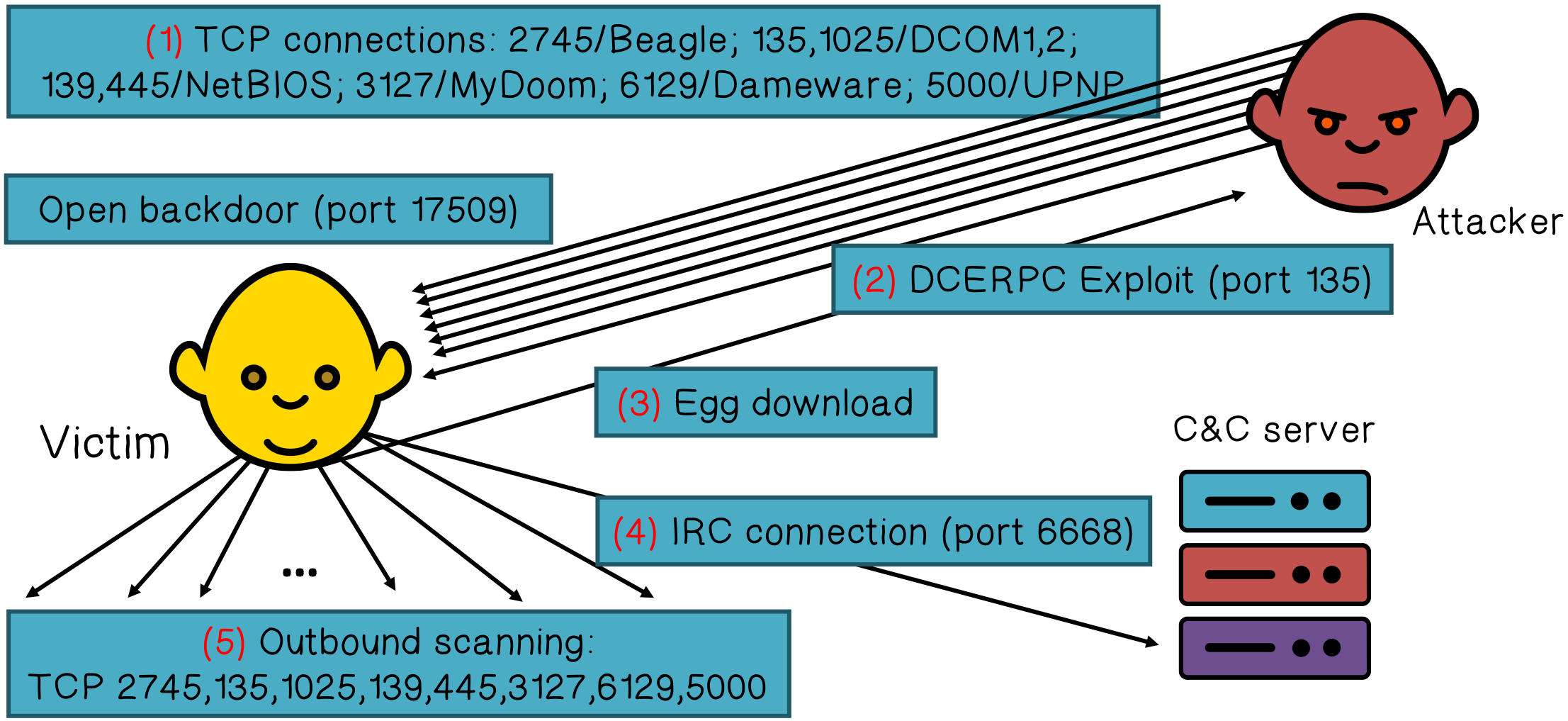


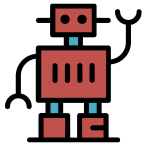
BotHunter: Vertical Dialog Correlation



Vertical correlation or Dialog correlation

BotHunter: PhatBot





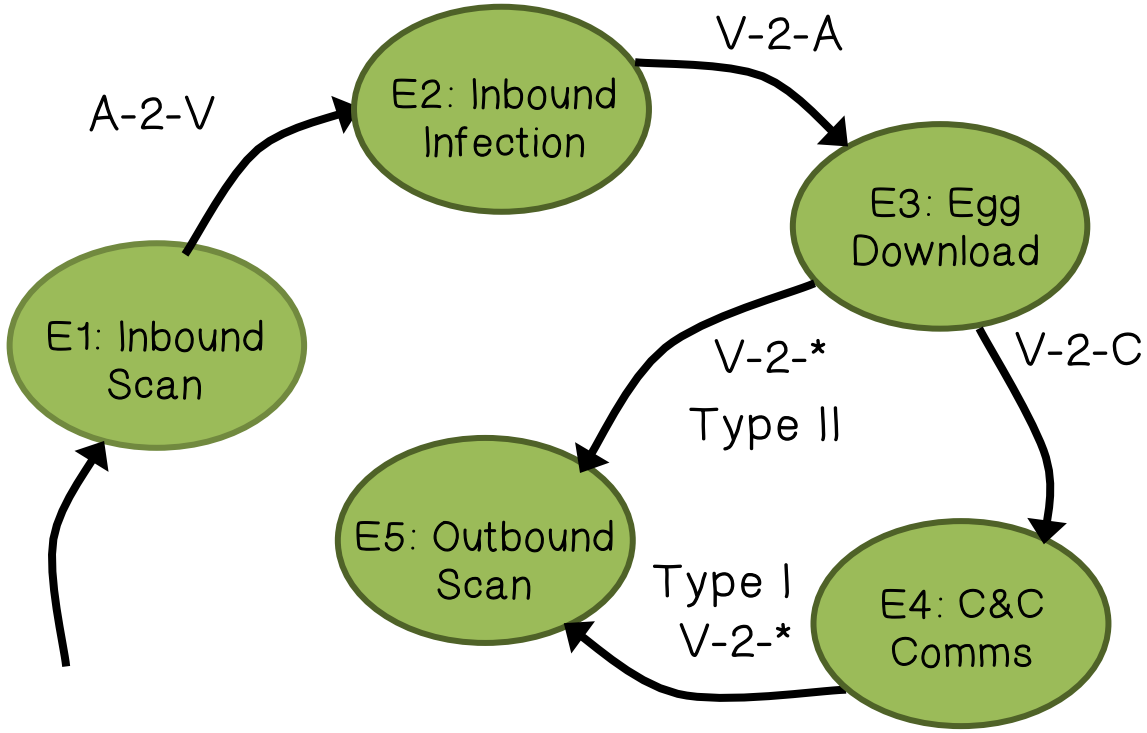
BotHunter

- An IDS-Driven Dialog Correlation Approach
- Monitors two-way communication flows between internal networks and the Internet for signs of bot and other malware
- Correlates dialog trail of inbound intrusion alarms with outbound communication patterns
- Produces a comprehensive ‘bot’ profile

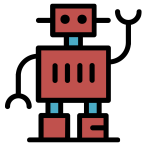
BotHunter: Dialog-based Correlation

- Egress point (internal – external)
- Search for duplex communication sequences that map to infection lifecycle model
- Stimulus does not require strict ordering, but does require temporal locality

Infection Lifecycle Model



A: Attacker, V: Victim, C: C&C server, Ei: events



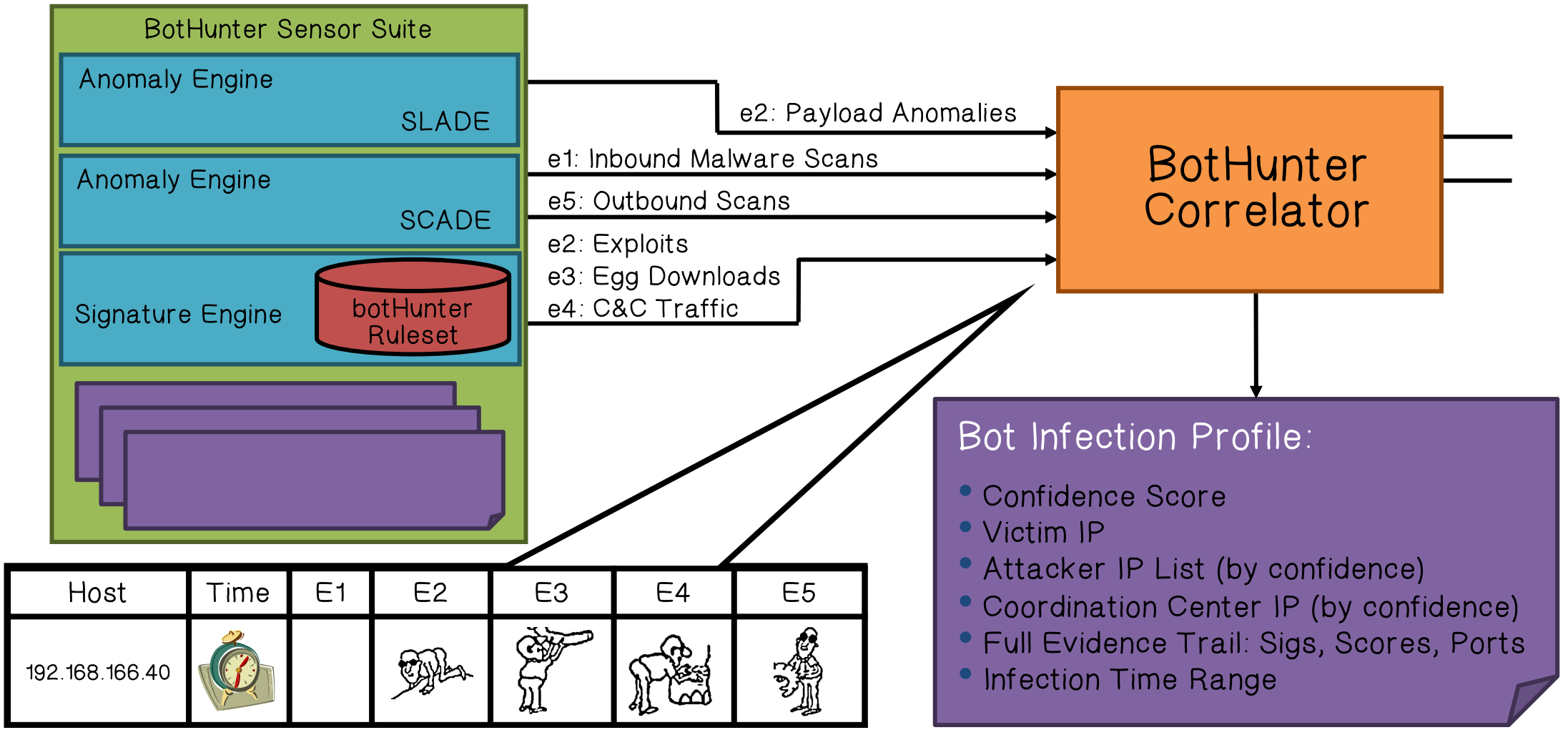
BotHunter: Dialog-based Correlation

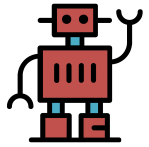
Characteristics of Bot Declarations

- External stimulus alone cannot trigger bot alert
- 2 x internal bot behavior triggers bot alert

Int. Host	Timer	E1 ◷	E2	E3	E4	E5
192.168.12.1	◷	$A_a \ldots A_b$				
192.168.10.45	◔		$A_c \ldots A_d$		$A_e \ldots A_f$	
192.168.10.66	◔		A_g			
192.168.12.46	◔				$A_h \ldots A_i$	$A_j \ldots A_k$
⋮						
192.168.11.123	◷ ◔	A_l	$A_m \ldots A_n$	A_o		

BotHunter Architecture





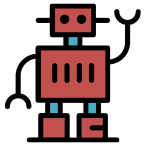
BotHunter Architecture: SCADE

SCADE: Statistical Scan Anomaly Detection Engine

- Custom malware specific weighted scan detection system for inbound and outbound sources
- Bounded memory usage to the number of inside hosts, less vulnerable to DoS attacks

Inbound *(E1: Initial Scan Phase)*:

- suspicious port scan detection using weighted score
- failed connection to vulnerable port = high weight
- failed connection to other port = low weight



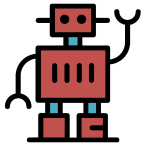
BotHunter Architecture: SCADE

SCADE: Statistical Scan Anomaly Detection Engine

- Custom malware specific weighted scan detection system for inbound and outbound sources
- Bounded memory usage to the number of inside hosts, less vulnerable to DoS attacks

Outbound *(E5: Victim Outbound Scan)*:

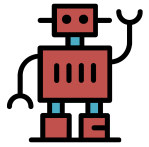
- S1 – Scan rate of V over time t
- S2 – Scan failed connection rate *(weighted)* of V over t
- S3 – Scan target entropy *(low revisit rate implies bot search)* over t
- Combine model assessments: Or, Majority voting, AND scheme



BotHunter Architecture: SLADE

SLADE: Statistical pay Load Anomaly Detection Engine

- Suspicious payload detection: new "lossy" n-gram byte distribution analysis over a limited set of network services
- Implements a lossy data structure to capture 4-gram hash space: default vector size = 2048 *(Versus n=4, $256^4 = 2^{32} \approx 4Gb$)*
- Comparable accuracy as full n-gram scheme: low FP and FN
- General performance comparable to PAYL (Wang2004): to detect all 18 attacks, the false positive of PAYL is 4.02%, SLADE is 0.3601%
- Ke Wang, Salvatore J. Stolfo. "Anomalous Payload-based Network Intrusion Detection", RAID'04



Sensor Suite: Signature Engine

Signature Set

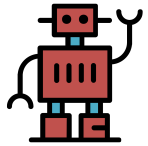
- Replaces all standard snort rules with five custom rulesets: e[1-5].rules"

Scope: Dialog content

- Known worm/bot exploit signatures, shell/code/script exploits, malware update/download, C&C command exchanges, outbound scans"

Rule sources

- Bleeding Edge malware rule sets
- Snort community rules
- Cyber-TA custom bot-specific rules



Example BotHunter Infection Profile

Score: 1.95 (>= 0.8)
Infected Target: 192.168.166.40
Infector List: 192.168.166.20
C & C List: 192.168.166.10 (27)
Observed Start: 01/19/2007 17:15:27.60 EST
Report End: 01/19/2007 17:18:26.22 EST
Gen. Time: 01/19/2007 17:18:26.22 EST

Example VMWare RBot Experiment
Initial Bot Infector: 192.168.166.20
Victim System: 192.168.166.40
Coordination Center: 192.168.166.10

INBOUND SCAN

EXPLOIT
192.168.166.20 (2) (17:15:27.60 EST)
E2[rb] SHELLCODE x86 0x90 unicode NOOP

EXPLOIT (slade)
192.168.166.20 (2) (17:15:27.60 EST)
E2[sl] Slade detected suspicious payload exploit with anomaly score 2312.725576.

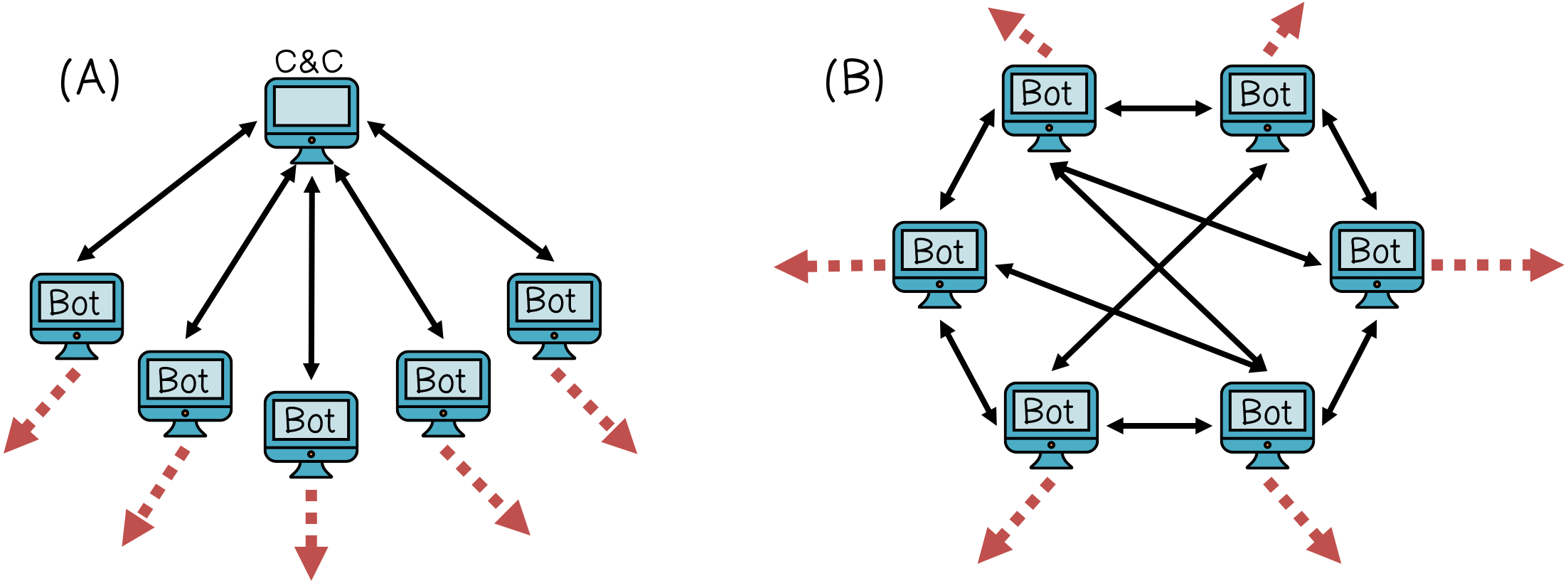
EGG DOWNLOAD
192.168.166.20 (2) (17:15:27.96 EST)
E3[rb] TFTP GET .exe from external source 1028->69 (17:15:27.96 EST)

C and C TRAFFIC
192.168.166.10 (27) (17:15:46.56 EST-17:18:26.22 EST)
E4[rb] BLEEDING-EDGE TROJAN IRC NICK command 1029->6668 (17:15:46.56 EST)
E4[rb] BLEEDING-EDGE TROJAN BOT - potential scan/exploit command
……

OUTBOUND SCAN
192.168.166.20 (17:16:42.18 EST)
E5[sc] scade detected suspicious scanner [192.168.166.40] scanning 30 IPs at ports [0 135 …]

BotMiner: Another Botnet Detection System?

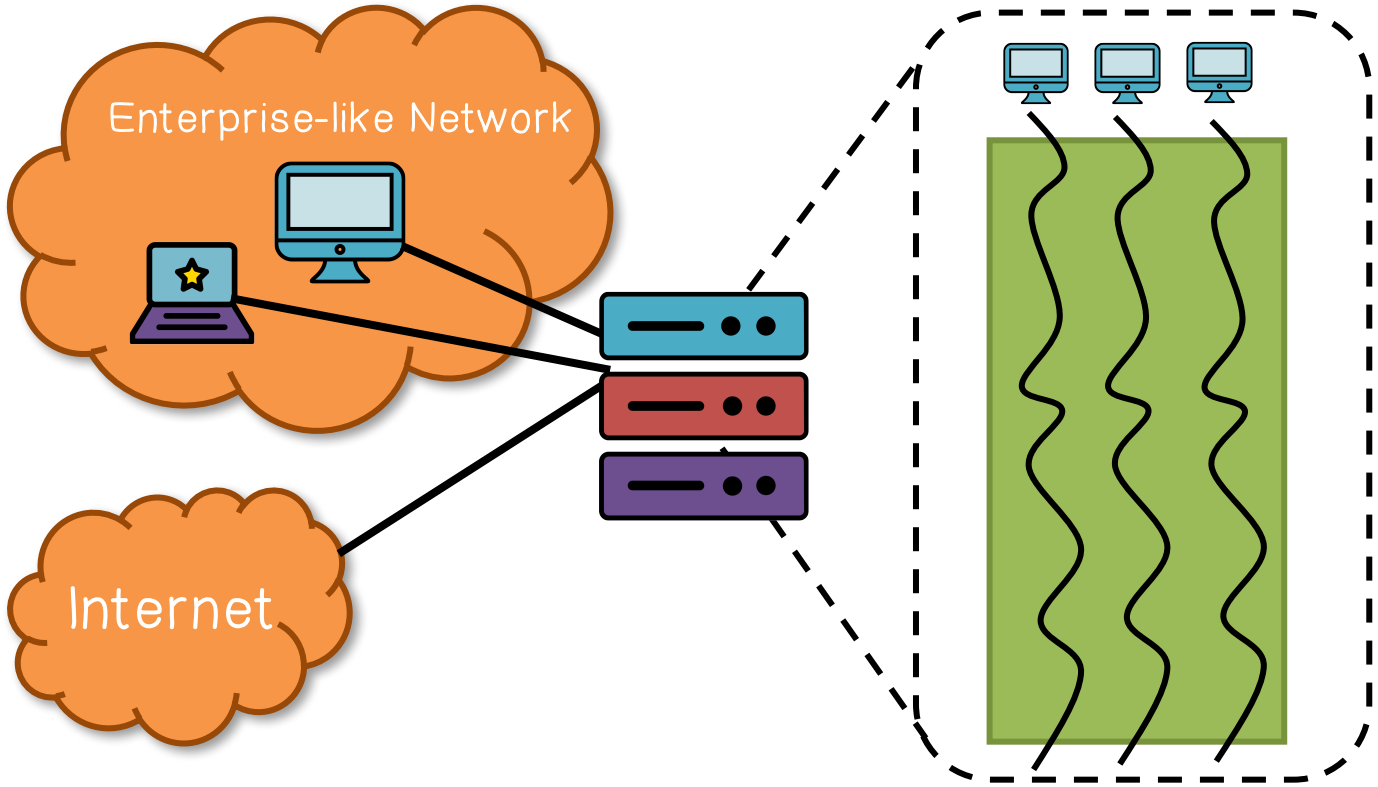
Botnets can change their C&C content *(encryption, etc.)*, protocols *(IRC, HTTP, etc.)*, structures *(P2P, etc.)*, C&C servers, infection models ...



Protocol & Structure Independent Detection

Both Vertical and Horizontal correlation

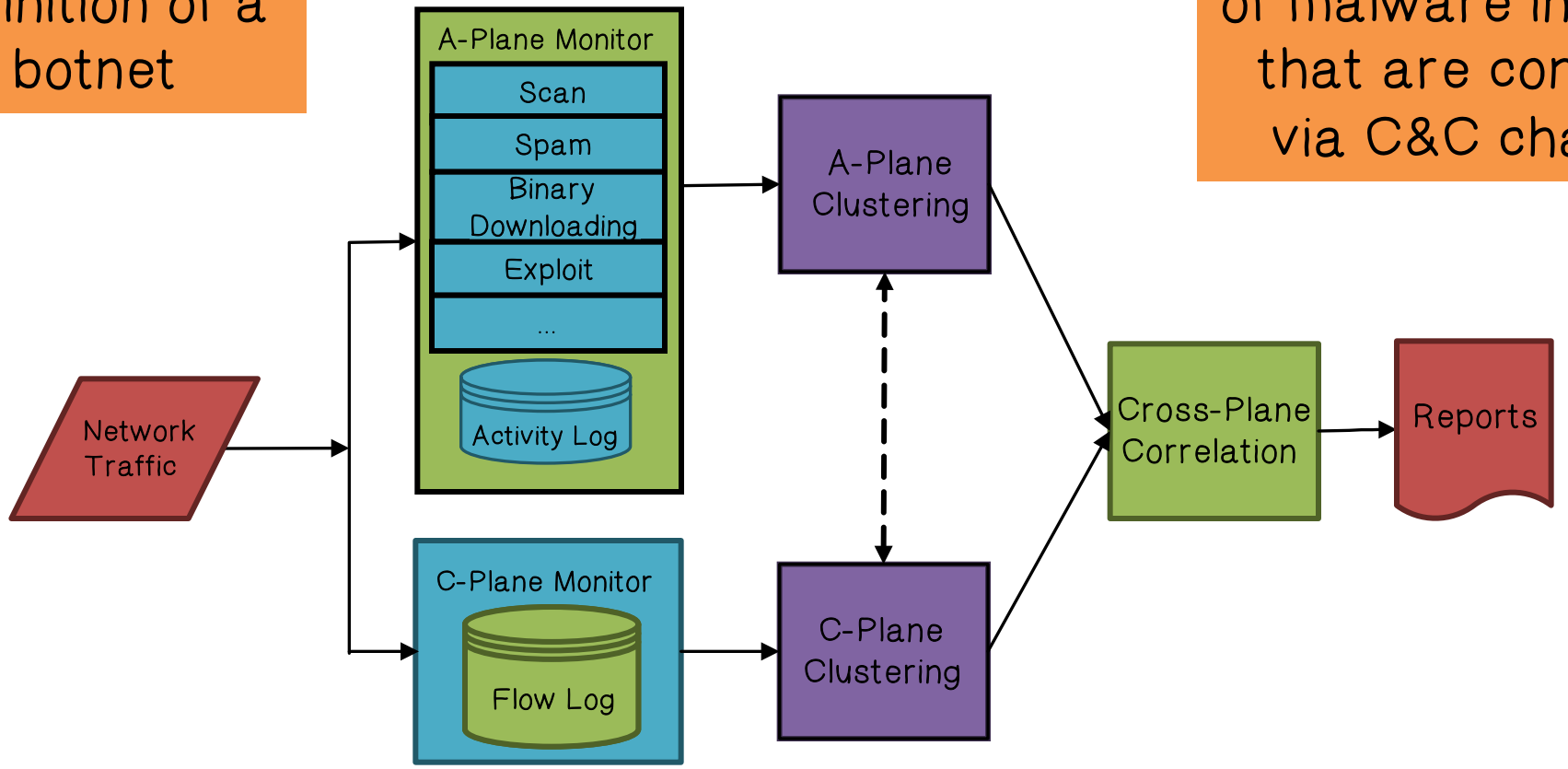
- Bots are for long-term use
- Botnet: communication and activities are coordinated/similar

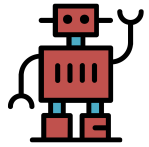


BotMiner

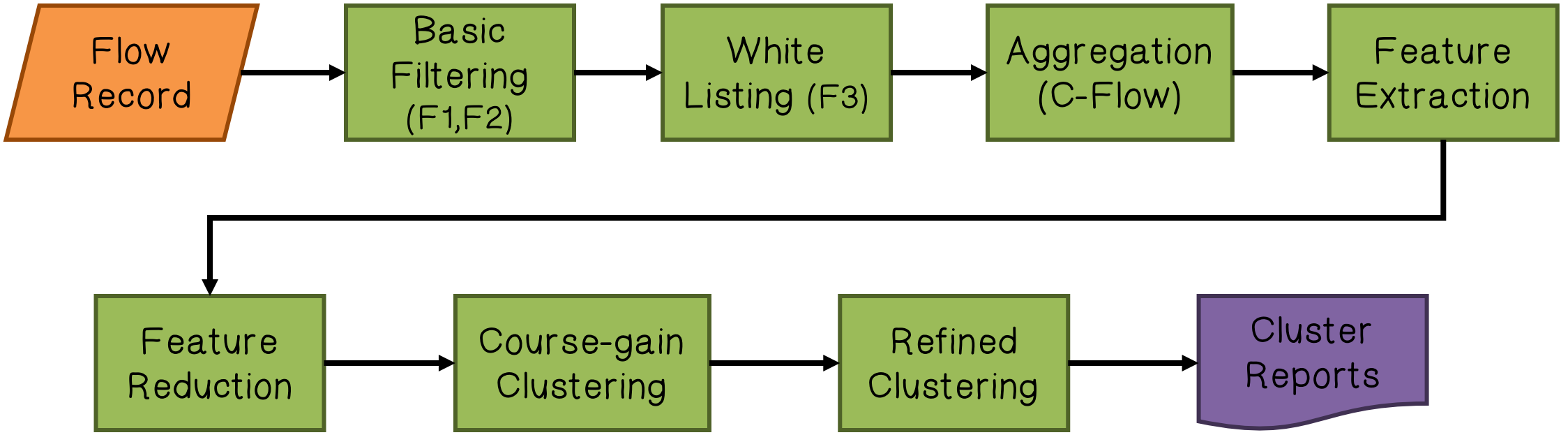
Revisit the definition of a botnet

A coordinated group of malware instances that are controlled via C&C channels

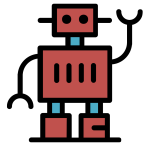




BotMiner: C-Plane Clustering



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



BotMiner: Related Statistical Distribution

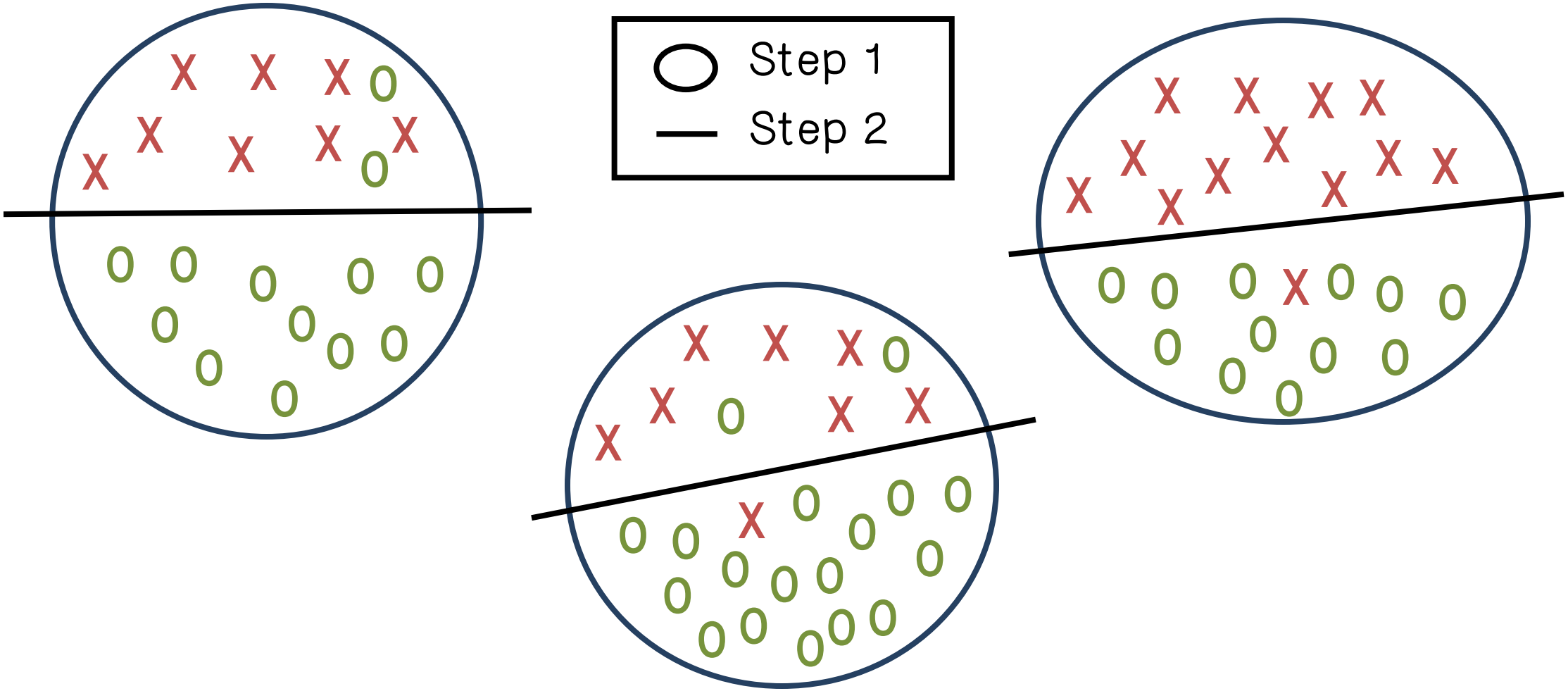
Temporal related statistical distribution information in:

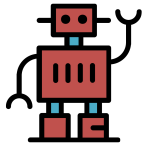
- BPS (bytes per second)
- FPH (flows per hour)

Spatial related statistical distribution information in

- BPP (bytes per packet)
- PPF (packets per flow)

BotMiner: Two-step Clustering of C-flows





BotMiner: Two-step Clustering of C-flows

Why multi-step?

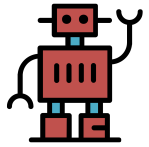
- Efficiency

Coarse-grained clustering

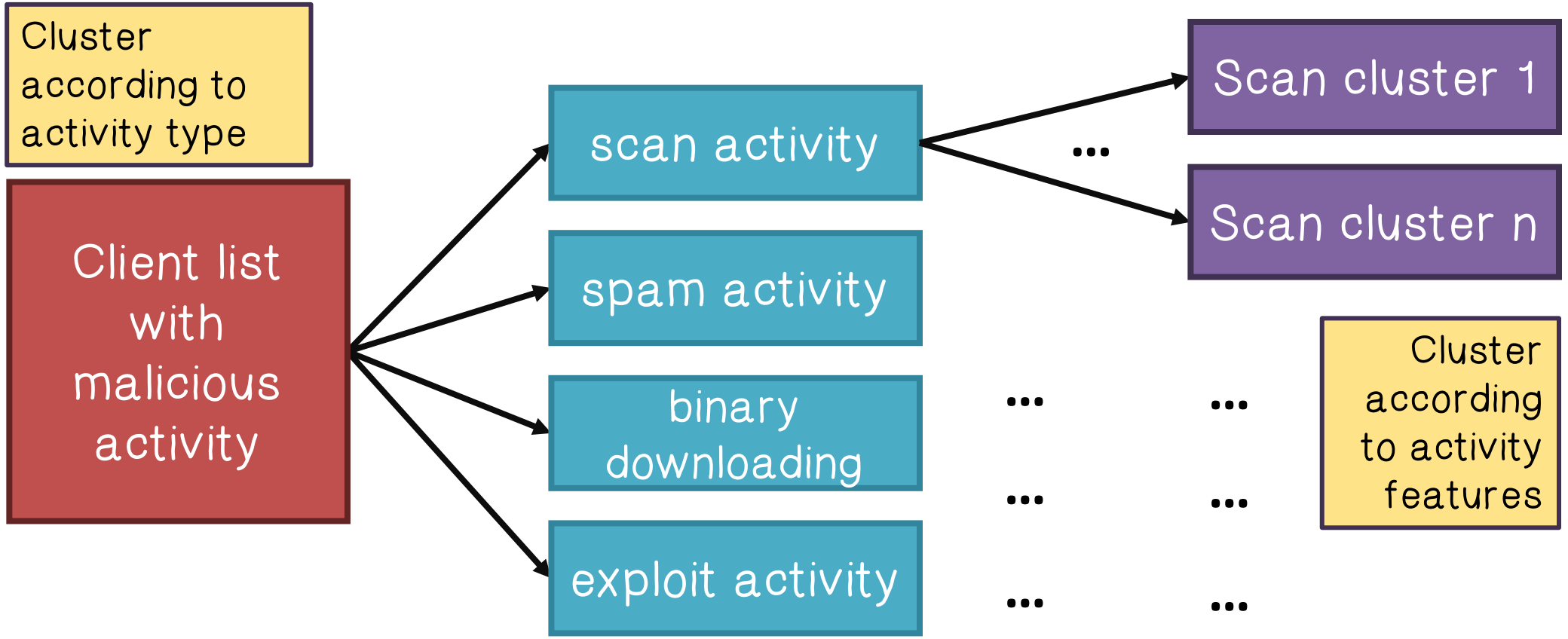
- Using reduced feature space: mean and variance of the distribution of FPH, PPF, BPP, BPS for each C-flow (2*4=8)
- Efficient clustering algorithm: X-means

Fine-grained clustering

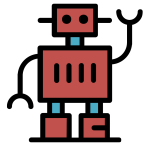
- Using full feature space (13*4=52)



BotMiner: A-plane Clustering



Capture "similar activity patterns"



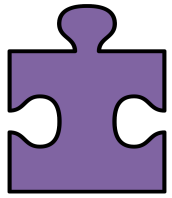
BotMiner: Cross-plane Correlation

Cross-check the clustering results of A-plane *(activities)* and C-plane *(communications)*

- Intersections provide stronger evidence that a host is in a botnet: similar malicious activities AND C&C patterns
- The more intersections a host falls into the stronger evidence that it is a bot

Clustering bots into the same botnet

- If two hosts appear in the same activity clusters and in at least one common C-cluster, they should be clustered together (as the same botnet)



Botnet Detection Quiz

Which of these behaviors are indicative of botnets?

- [x] Linking to an established C&C server
- [x] Generating Internet Relay Chat (IRC) traffic using a specific range of ports
- [ ] Generating DNS requests
- [x] Generating SMTP emails/traffic
- [x] Reducing w… level that it is

Generating DNS requests is not suspicious behavior. Generating SIMULTANEOUS IDENTICAL DNS requests is suspicious

BotMiner Limitations Quiz

What can botnets do to evade C-plane clustering?

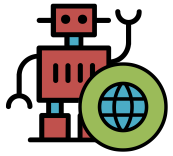
Manipulate communication patterns.

Introduce noise (in the form of random packets) to reduce similarity between C&C flows.

What can botnets do to evade A-plane monitoring?

Perform slow spamming

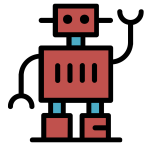
Use undetectable activities (spam sent with Gmail, download exe from HTTPS server)



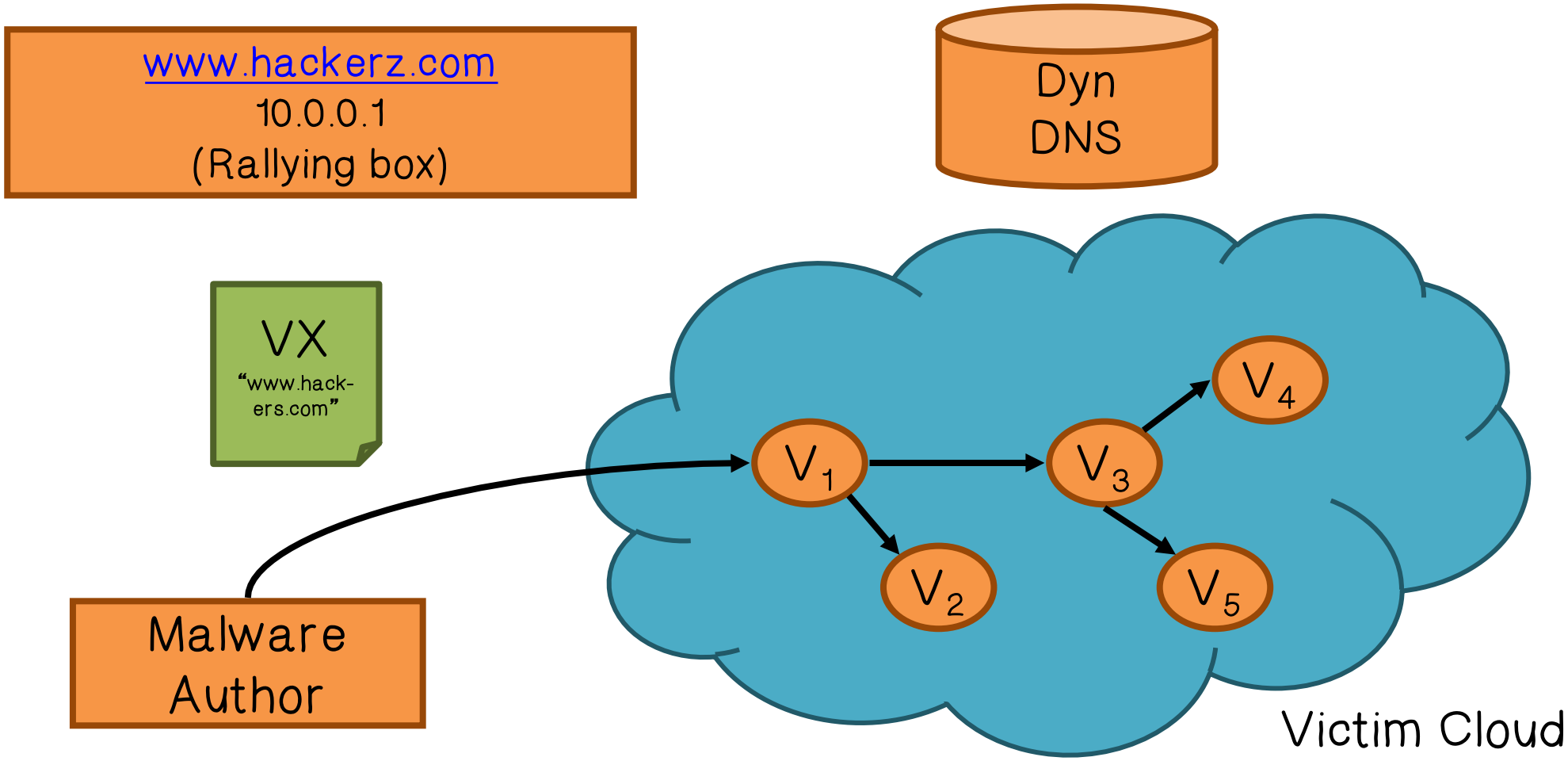
Botnet Detection on the Internet

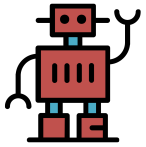
A botnet must use Internet protocols or services for efficiency, robustness, and stealth

- Look-up services (e.g., DNS, P2P DHT)
  - Find C&C servers and/or peers
- Hosting services (Web servers and proxies)
  - Storage and distribution/exchange of attack-related data, malware download
- Transport (e.g., BGP)
  - Route (or hide) attack from bots to victims
- Identify the abnormal use of Internet services that suggests botnet activities
- Let's focus on DNS
  - Used by most bots for locating C&C and hosting sites



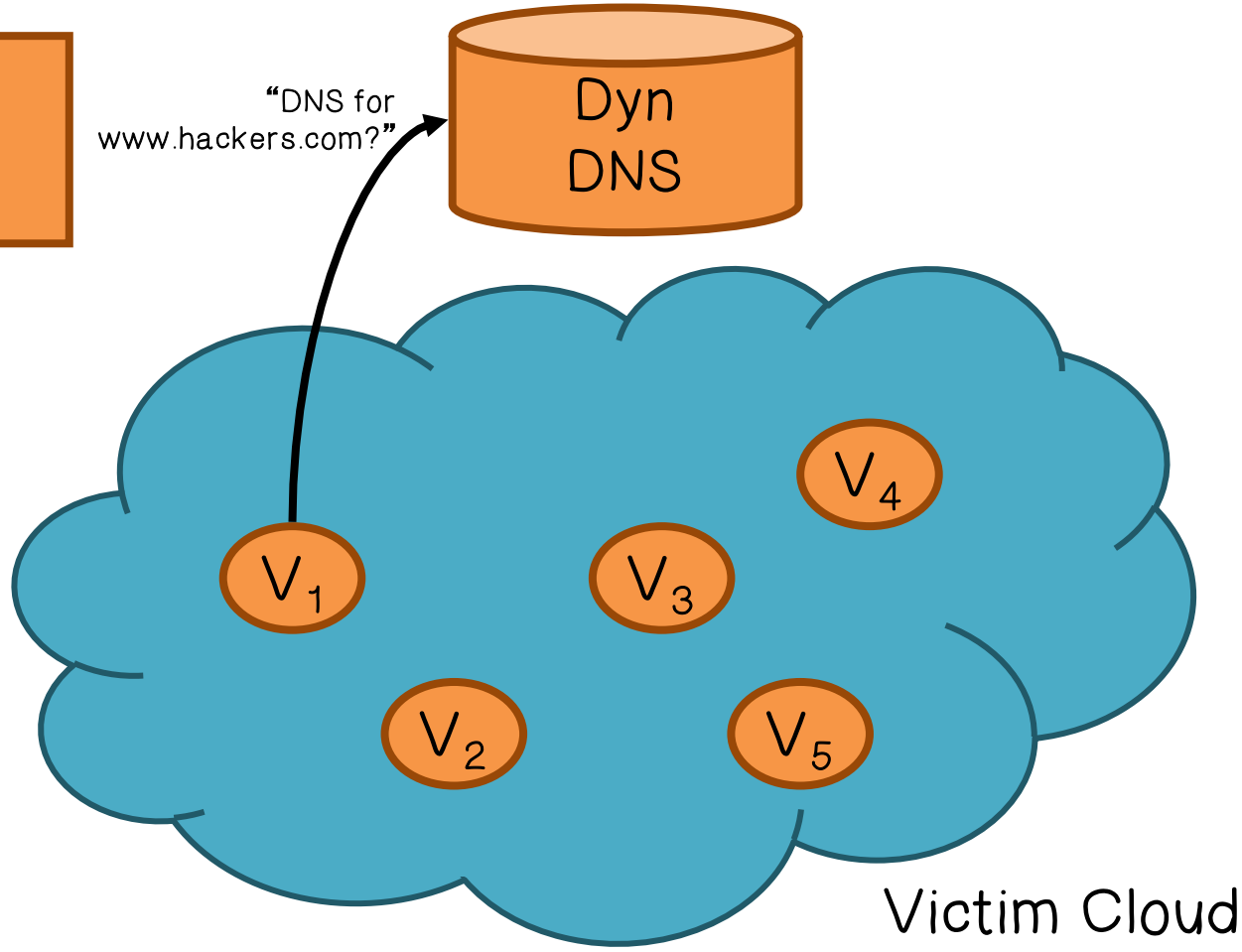
Botnet Use of Dynamic DNS Services

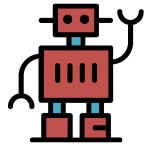




Botnet Use of Dynamic DNS Services

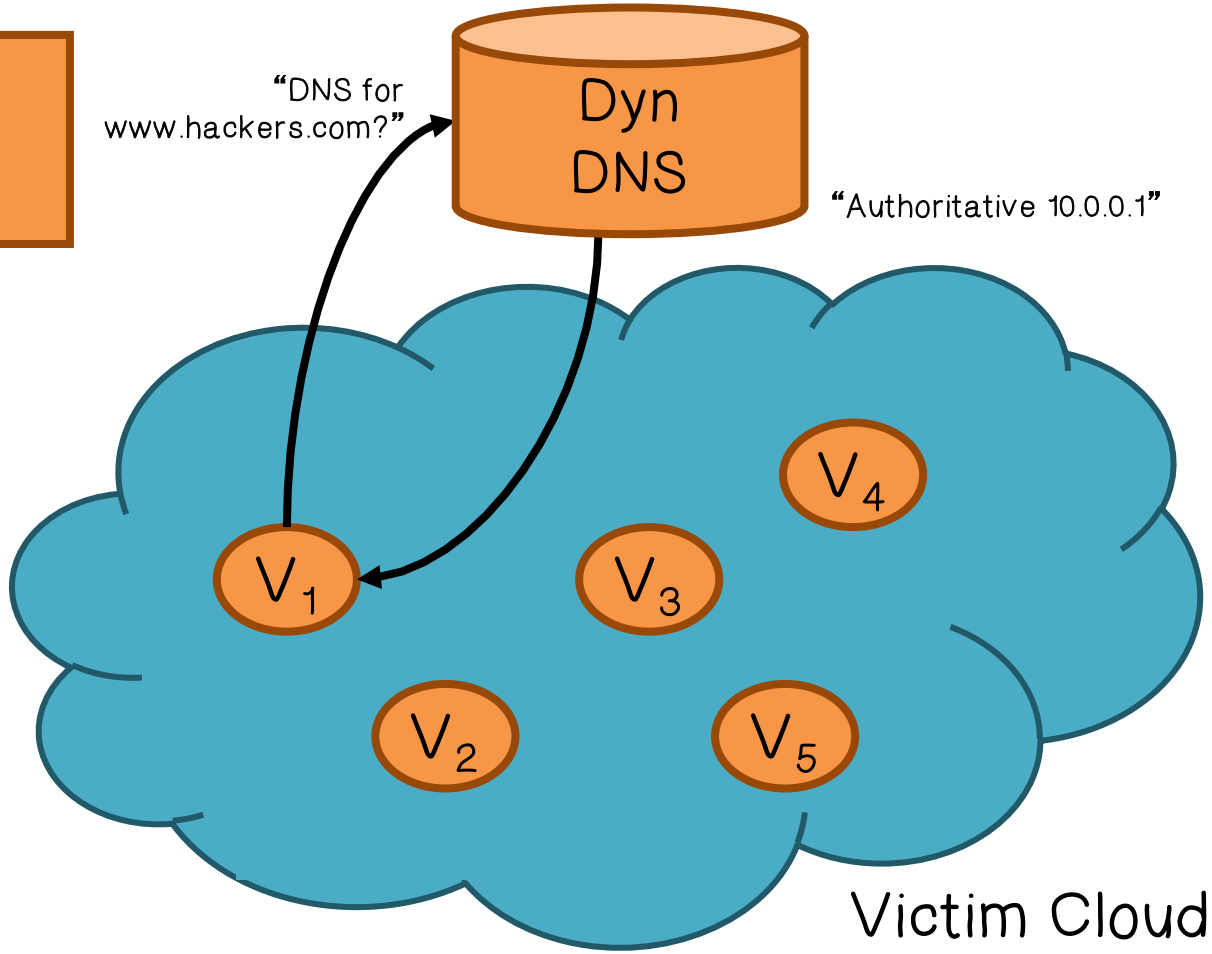
www.hackerz.com
10.0.0.1
(Rallying box)

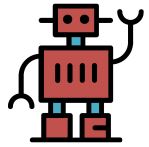




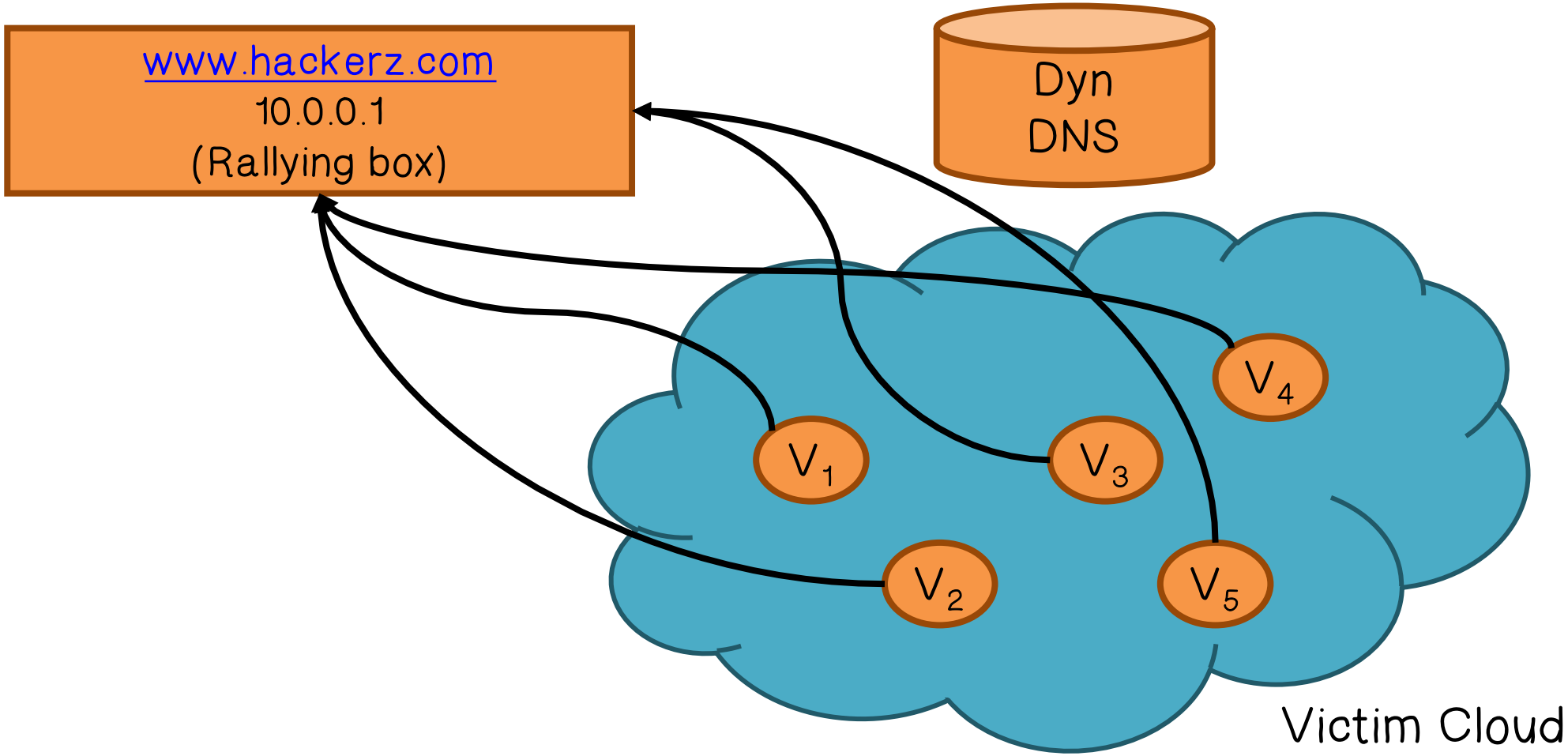
Botnet Use of Dynamic DNS Services

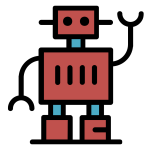
www.hackerz.com
10.0.0.1
(Rallying box)



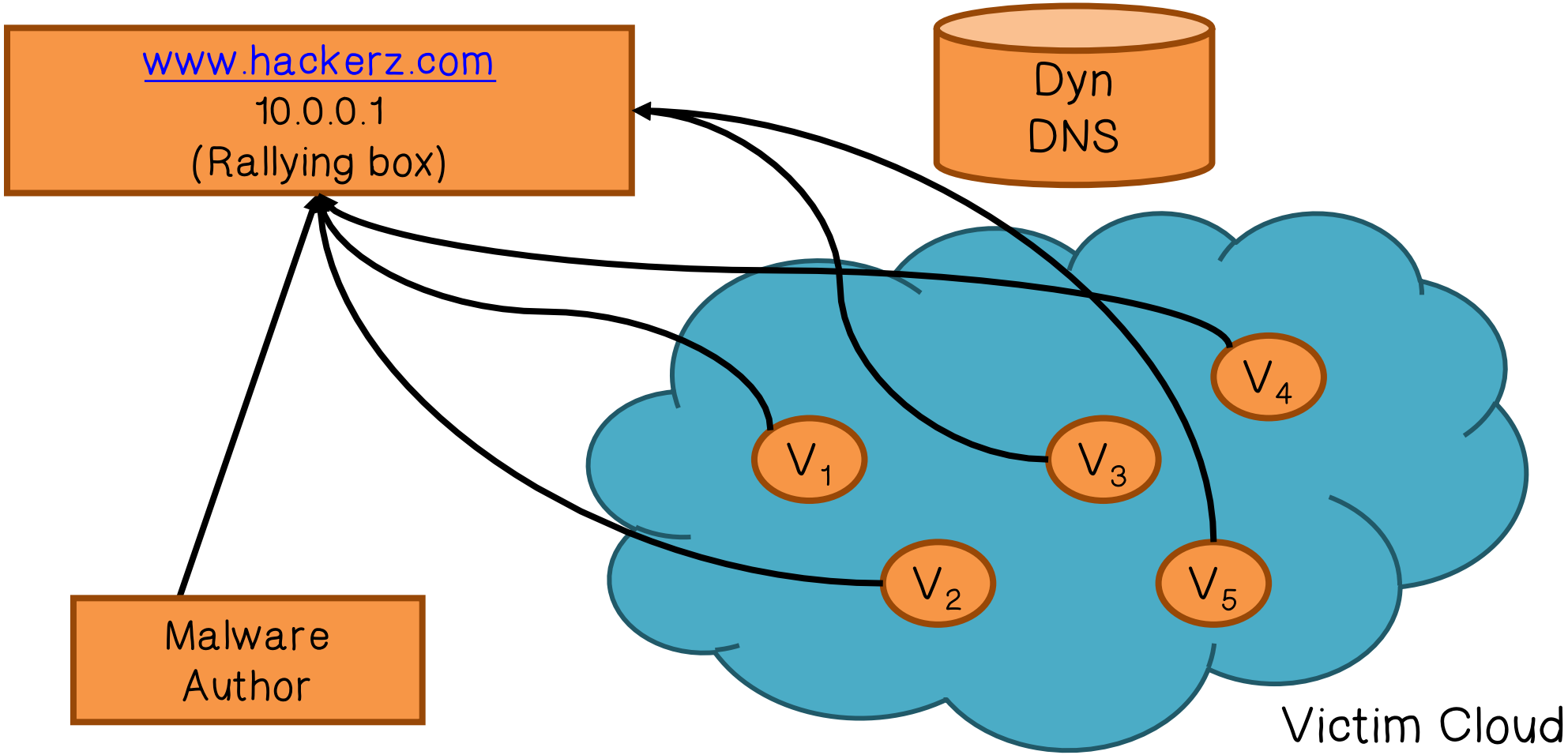


Botnet Use of Dynamic DNS Services

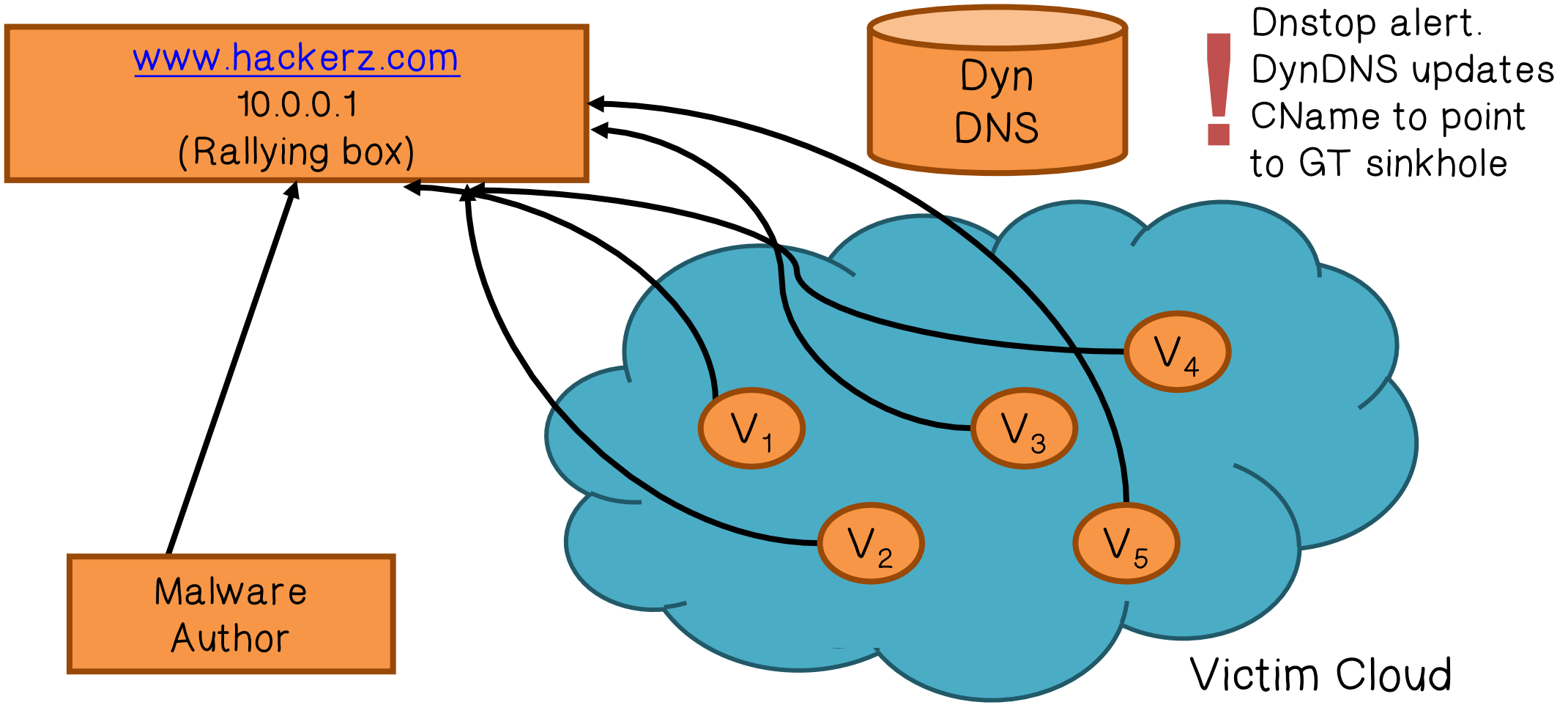


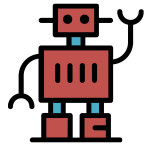


Botnet Use of Dynamic DNS Services



Botnet Use of Dynamic DNS Services





Botnet Use of Dynamic DNS Services

www.hackerz.com
10.0.0.1
(Rallying box)

???

Malware Author

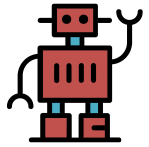
Dyn DNS

! Dnstop alert. DynDNS updates CName to point to GT sinkhole

V_1 V_2 V_3 V_4 V_5

GT Sinkhole

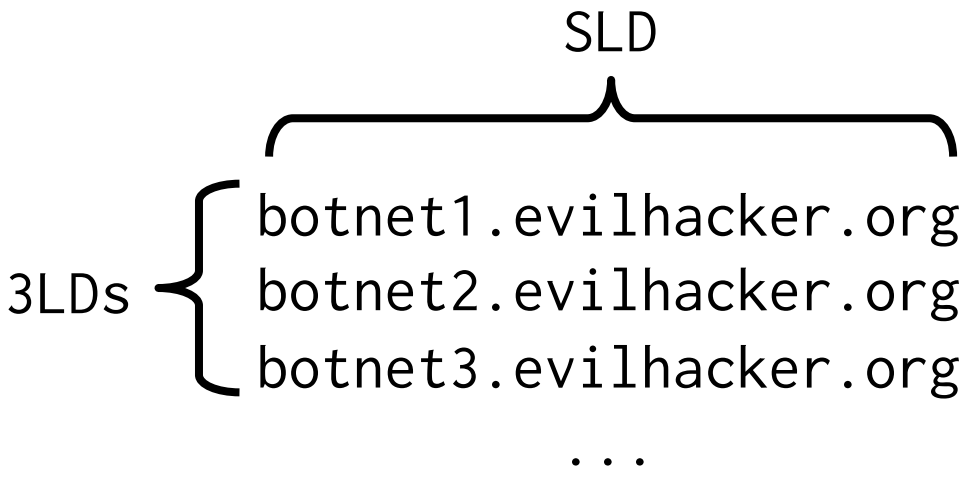
Victim Cloud

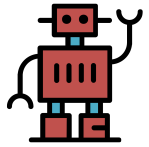


Botnet Use of Dynamic DNS Services

Observation 1: hard-coded C&C domain (string)

- Domain name purchases use traceable financial information. Multiple 3LDs can use DDNS service with one package deal
- Thus: financial and stealthy motives for botnet authors to "reuse" SLD with numerous similar/clustered 3LDs

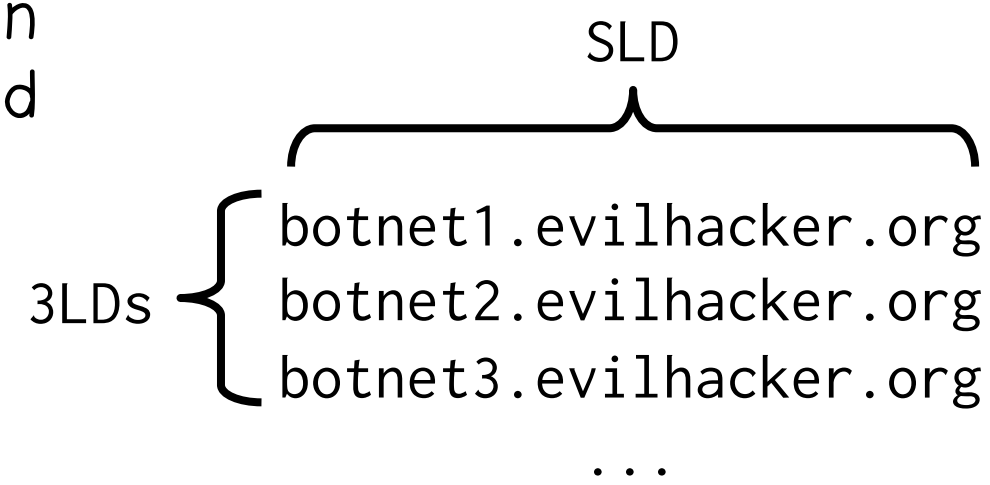




Botnet Use of Dynamic DNS Services

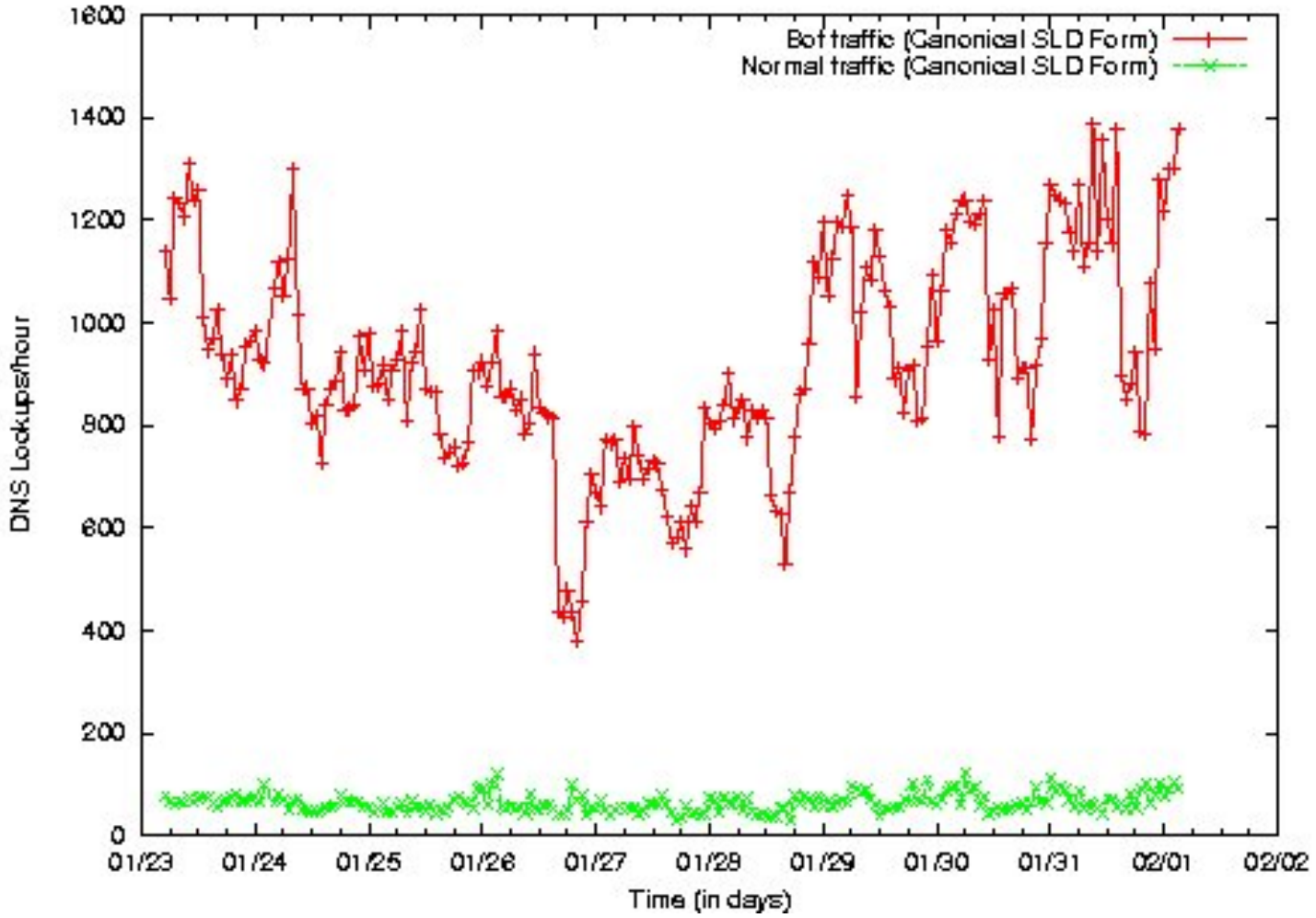
Clustered 3LD Look-ups

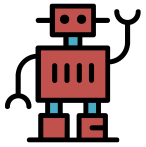
- Cluster the 3LDs under a SLD based on their similarities on names, and subnets of resolved IPs.
- Sum up the look-ups to all domains within a Cluster



Botnet Use of Dynamic DNS Services

Clustered
3LD Look-ups

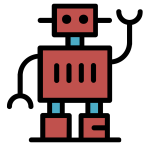




Botnet Use of Dynamic DNS Services

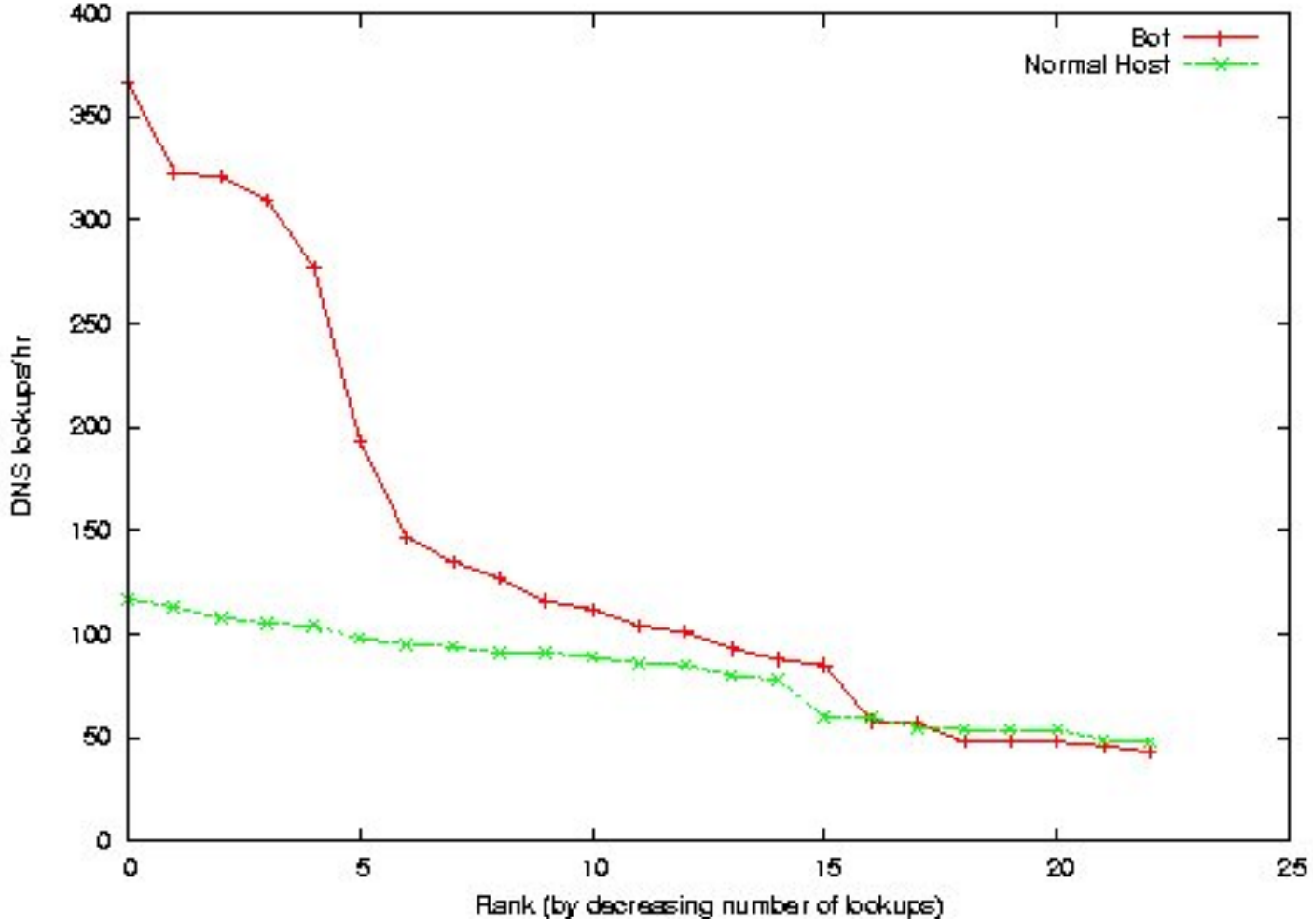
Observation 2: DNS look-up behavior of botnets

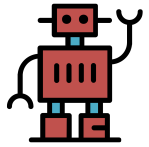
- After boot, bots immediately resolve their C&C
  - Exponential arrival (spike) of bot DNS requests, because of time zones, 9 a.m./5 p.m. schedules, etc.
- Normal DNS lookup behavior is a lot smoother
  - Human users don't all immediately check the same server right after boot



Botnet Use of Dynamic DNS Services

Look-up Arrival Rate (cont'd)





Botnet Use of Dynamic DNS Services

Other Observations/Features

Source IP dispersion in DNS look-ups

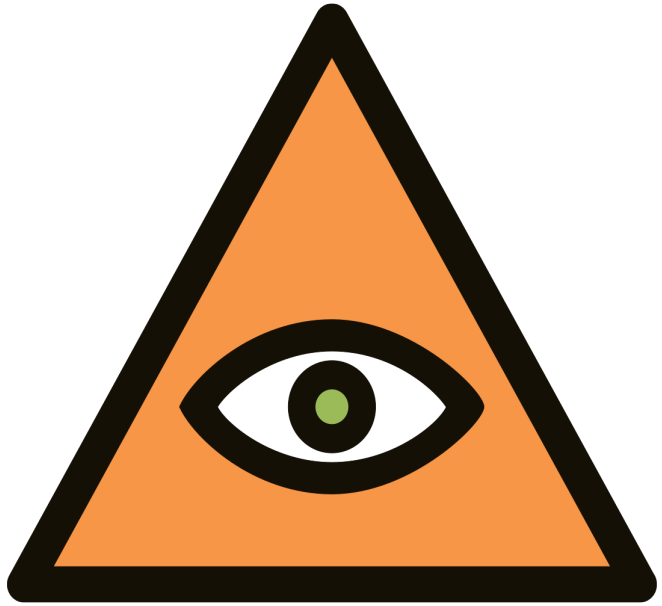
- Local or global popularity of the domain

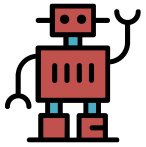
Resolved IP dispersion

- Distributed in many different networks?

Number of times resolved IP changed

...





Botnet Use of Dynamic DNS Services

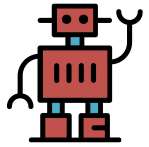
Recursive DNS Monitoring at ISP

Analyze DNS traffic from internal hosts to a recursive DNS server(s) of the network

Detect abnormal patterns/growth of "popularity" of a domain name

- Identify botnet C&C domain and bots

Common means of botnet propagation: (worm-like) exploit-based, email-based, and dry-by egg download



Botnet Use of Dynamic DNS Services

Recursive DNS Monitoring at ISP

Exploit-based propagation:

- The number of infected machines grows exponentially in the initial phase

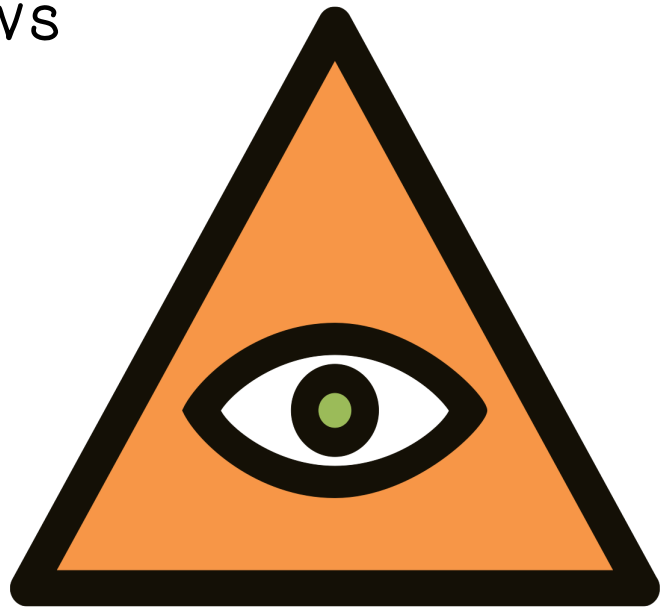
Email-based propagation:

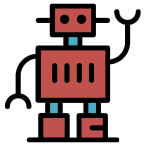
- Exponential or linear

Dry-by egg download:

- Likely sublinear

Studies showed:





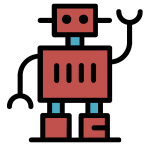
Botnet Use of Dynamic DNS Services

Anomalous Domain Names

Botnet-related domains usually contain random-looking (sub) strings

Many/most sensible domain names have been registered (for legitimate use)

In particular, botnet domain name 3LD often looks completely random, and the domain name tends to be very long E.g. wbghid.1dumb.com, 00b24yqc.ac84562.com



Botnet Use of Dynamic DNS Services

Popularity Growth of the Suspicious Names

Monitor for "new and suspicious" domain names that enjoy exponential or linear growth of interests/look-ups

- Train a Bloom filter for N days to record domain names being looked-up, and a Markov model of all the domain name strings
  - On the N+1 day, consider a domain "new" if it is not in the Bloom filter; and if it does not fit the Markov model, it is also "suspicious"
- Treat the sequence of look-ups to each new and suspicious domain (on the N+1 day) as a time series
- Apply linear and exponential regression techniques to analyze the growth of number of look-ups
  - A match of the growth patterns suggests botnet domain

Detection of Targeted & Advanced Threats

- Zero-day exploits, custom-built malware
- Low-and-slow
- Lateral movement
- …

- Need multi-faceted monitoring and analysis
  - Malware analysis
  - Host-based monitoring, forensics, recovery
  - Network monitoring
  - Internet monitoring, threat analysis, attribution

APT Quiz

Which of the information should be considered in order to identify the source (perpetrator) of an APT attack?

- [ ] Source IP address of TCP-based attack packets
- [ ] Coding style of malware
- [ ] Inclusion of special libraries with known authors
- [ ] Motives of the attack
- [ ] Language encoding
- [x] All of the above