Entropy-Based Measurement of IP Address Inflation in the Waledac Botnet

Rhiannon Weaver[1] Chris Nunnery[2] Gautam Singaraju[2]
Brent ByungHoon Kang[3]

[1]CERT/SEI
[2]University of North Carolina
[3]George Mason University

January 11, 2011

Introduction

The Botnet Question: How “big” is it?

- Size relates to potential threat, adaptability
- Relative size can help us prioritize mitigation efforts

Currently research thinks about size in two ways (Rajab et. al.)

- Count of active individuals at any particular point in time
- Footprint count of all unique individuals across the entire history

What’s an “individual”?

- Often count and report IP addresses
- Often want to know the number of machines
- NAT, DHCP can inflate or deflate our estimates

What effect does IP vs. machine measurement have on a footprint count?

Title Deconstruction and Roadmap

This research:

- Extends Rajab's footprint count to a distribution that weights individuals by their level of activity
- Introduces a measurement of IP address inflation based on relative entropy of footprint distributions
- Shows how to use relative entropy to discover NAT/DHCP properties of sub-networks useful for prioritizing blacklisting and cleanup efforts
- Presents some results from applying these concepts to data (IP addresses and unique IDs) collected from the Waledac botnet

IP Address Inflation Rate (R)

The effect on a population estimate of counting IP addresses instead of machines

- $R > 1$ for a machine moving among a DHCP pool
- $R < 1$ for several machines using the same NAT address

We can study inflation rates directly in "visible" botnets (IPs and IDs available)

Network policy information can be transferrable to "hidden" botnets (IPs only are observable)

Inflation Rate of a Footprint Measurement

For a visible botnet, let

$$\begin{aligned} I &= \text{Set of observed IP addresses} \\ H &= \text{Set of observed machines} \end{aligned}$$

cumulative across the recorded active history.

A naive measurement of the footprint inflation rate is simply:

$$R_N(I, H) = \frac{|I|}{|H|}$$

Interpretation: breadth and spread
What is missing? relative popularity and visibility of IPs, individuals

An Activity-based Footprint Distribution

An individual j (IP address or machine) is observed over time due to its network activity a_j:

- Scan hits
- #Log-ins to C&C server
- #P2P clients contacted, etc.

For a population J, define the the *footprint distribution* $p_J(j)$:

$$p_J(j) = \frac{a_j}{\sum_{k \in J} a_k}$$

This distribution weights every individual by its associated activity (temporal or volumetric)

Entropy and Inflation

Shannon Entropy $S(p_J)$ of a footprint distribution p_J measures its uniformity:

$$S(p_J) = -\sum_{j \in J} p_J(j) \ln[p_J(j)]$$

For footprint distributions p_I and p_H, we define the Entropy-based IP Inflation Rate R_E as

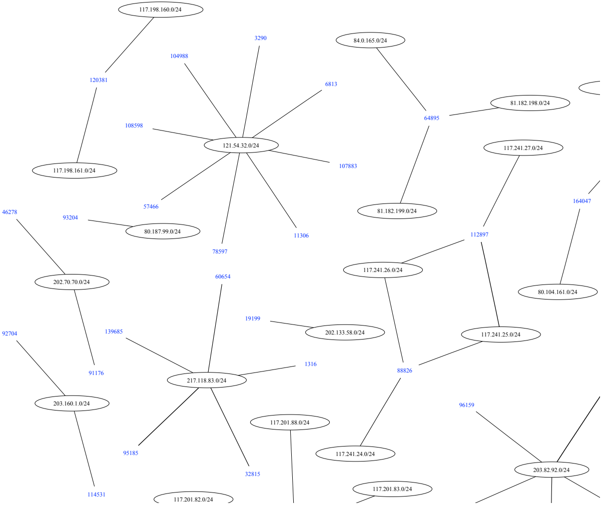
$$R_E(p_I, p_H) = \exp[S(p_I) - S(p_H)]$$

Note:

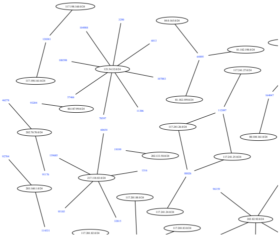
- Maximal (uniform) entropy among N items is equal to $\ln(N)$
- $R_E = R_N$ when p_I and p_H are uniform, but extends inflation to apply to unequal distributions.

Studying Sub-networks

Connections between IPs and Individuals form a graph G, that has inflation rate $R_E(G)$



The Graph Properties of IP Inflation



- $R_E(G_\ell)$ can be measured for any sub-graph $G_\ell \subset G$ with associated activity a_ℓ
- Equivalence classes are the only partitions of I or H that satisfy the rate-preserving equality:

$$R_E(G) = \sum_\ell \frac{a_\ell}{a_L} R_E(G_\ell)$$

Pruning within ASN to find sub-networks

We would like to interpret Equivalence Classes as independent networks, but they often traverse ASN or even country boundaries:

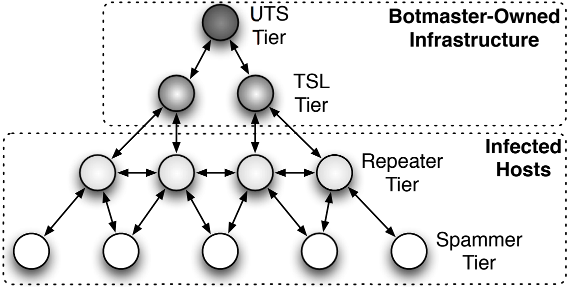
To obtain a more interpretable set of equivalence classes, create a sub-graph $G_R \subset G$:

- find the *modal ASN* M_h of each unique individual h
- Remove from G (set a_{hi} to 0) any edge (h, i) such that $i \notin M_h$

This restricts strong connected components in G_R to within-ASN clusters

The set of removed edges A has *weight* equal to $R_E(G)/R_E(G_R)$

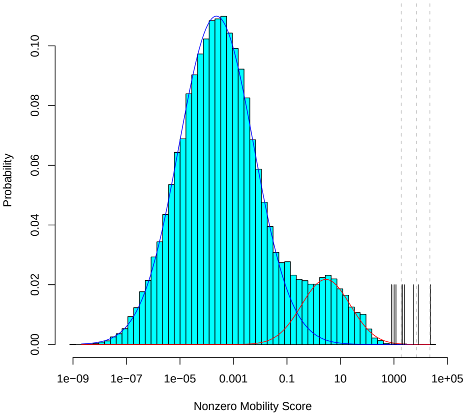
Application: Waledac Logs (12/04-22/2009)



Used SiLK to analyze 44 million log files over 3 different graphs

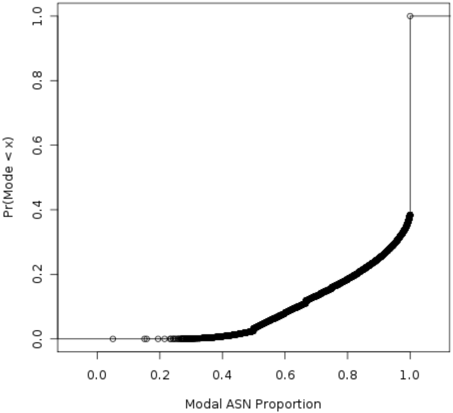
Graph	$\lvert I\rvert$	$\lvert H\rvert$	$\%a_\ell$	R_N	R_E
G	667033	172283	1.00	3.87	4.56
G_L					
G_R					

Removing Aliases to obtain G_L



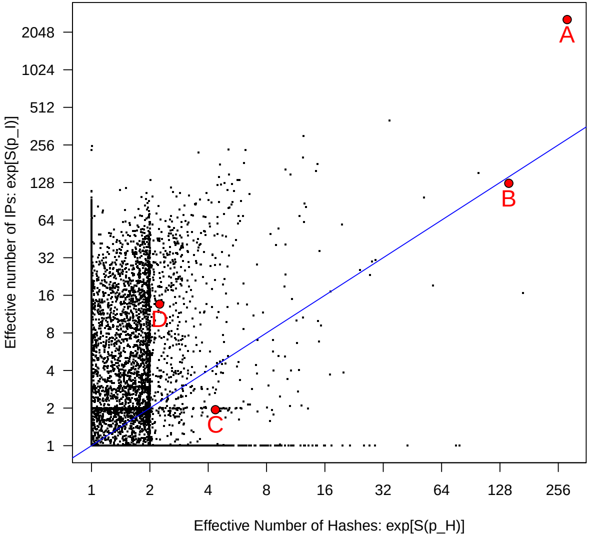
Graph	$\|I\|$	$\|H\|$	$\%a_\ell$	R_N	R_E
G	667033	172283	1.00	3.87	4.56
G_L	548997	172238	0.92	3.18	2.27
G_R					

Pruning within ASN to obtain G_R:



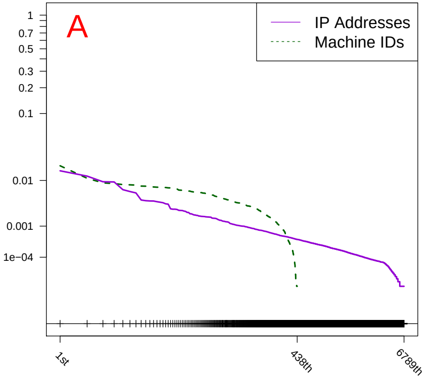
Graph	$\|I\|$	$\|H\|$	$\%a_\ell$	R_N	R_E
G	667033	172283	1.00	3.87	4.56
G_L	548997	172238	0.92	3.18	2.27
G_R	475665	172238	0.86	2.76	2.00

Equivalence Classes in G_R



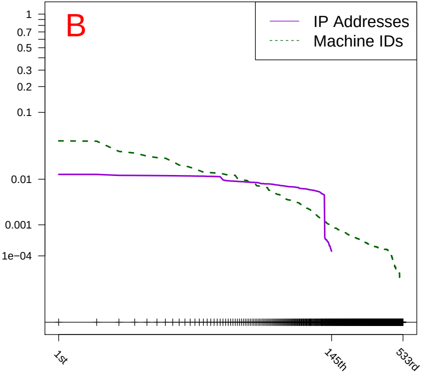
A Tale of Four Networks

Graph	$\lvert I \rvert$	$\lvert H \rvert$	a_ℓ	R_N	R_E
A	6789	438	317435	15.50	9.08
B	145	533	119684	0.27	0.89
C	5	5	296	1.00	0.45
D	16	16	1746	1.00	6.06



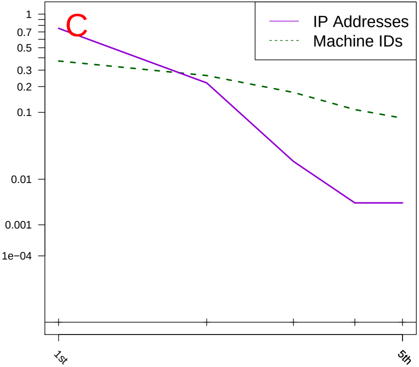
A Tale of Four Networks

Graph	$\|I\|$	$\|H\|$	a_ℓ	R_N	R_E
A	6789	438	317435	15.50	9.08
B	145	533	119684	0.27	0.89
C	5	5	296	1.00	0.45
D	16	16	1746	1.00	6.06



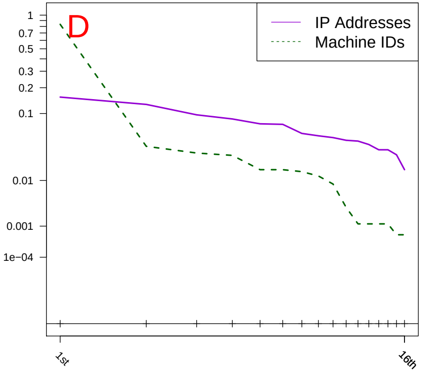
A Tale of Four Networks

Graph	$\lvert I \rvert$	$\lvert H \rvert$	a_ℓ	R_N	R_E
A	6789	438	317435	15.50	9.08
B	145	533	119684	0.27	0.89
C	5	5	296	1.00	0.45
D	16	16	1746	1.00	6.06



A Tale of Four Networks

Graph	$\lvert I\rvert$	$\lvert H\rvert$	a_ℓ	R_N	R_E
A	6789	438	317435	15.50	9.08
B	145	533	119684	0.27	0.89
C	5	5	296	1.00	0.45
D	16	16	1746	1.00	6.06



Summary and Future work

With this method and data, we are trying to answer a larger question:

Can we learn about individuals in a hidden botnet by studying a visible one?

- Find specific static regions of NAT or DHCP pools across the world and transfer this information to hidden botnets
- Create a tool/method that adjusts raw IP address counts for network structure
- Learn how to find a set of "most likely" Equivalence Classes when IPs only are visible

We are currently looking into learning about Conficker from this study of Waledac

Extra Slides

Subversive uses of SiLK

- Each Hash (eg “55530ea22bfee564631490025e”) assigned a unique integer ID (eg “10345”)

- Each Hash marked as Repeater (R) or Spammer (S) level

- Each Login stored as a SiLK record using `rwtuc`:

```
sIP          |    dIP |              sTime | tcpflags
111.222.33.4 |  10345 | 2009/12/20T00:14:12|        S
222.33.44.5  |  10345 | 2009/12/22T00:03:55|        S

...

rwtuc UTS-formatted.txt --output-file=UTSlogs.rw
```

Subversive uses of SiLK

- Inter-ASN network created with a tuple file:

```
sIP           |   dIP |
111.222.33.4 |  25667|
223.156.255.4|  25667|

...

rwfilter UTSlogs.rw --tuple-file=EdgesToRemove.txt --pass=InterASNlogs.rw

--fail=IntraASNlogs.rw
```

- Equivalence Class IDs and ASNs stored as P-maps:

```
rwfilter UTSlogs.rw --pmap-file=EQCLASS:Eqclasses.pmap --pmap-src=EQ2100 --pass=stdout |

rwstats --sip --threshold=1 > EQ2100-IP-distribution.txt
```

- Summary tables created using `rwuniq`:

```
rwuniq IntraASNlogs.rw --pmap-file=EQCLASS:Eqclasses.pmap --pmap-file=ASN:ASNs.pmap

--fields=src-EQCLASS,src-ASN --flows --sip-distinct --dip-distinct --stime

src-EQCLASS|                   src-ASN|Records|     sTime-Earliest|sIP-Distin|dIP-Distin|
        EQ0|"AS5089 NTL Group Limited"|    596|2009/12/12T21:14:45|         1|         1|
        EQ1|    "AS4766 Korea Telecom"|     45|2009/12/05T10:41:33|         1|         1|
        EQ3|  "AS1221 Telstra Pty Ltd"|     55|2009/12/08T04:43:00|        10|         1|
        EQ4|            "AS17858 KRNIC"|    628|2009/12/04T12:42:34|         2|         1|
```