Modeling Populations of Large-Scale Internet Threats

Rhiannon Weaver

Joint Statistical Meetings, Vancouver BC
August 4, 2010

NO WARRANTY

THIS MATERIAL OF CARNEGIE MELLON UNIVERSITY AND ITS SOFTWARE ENGINEERING INSTITUTE IS FURNISHED ON AN “AS-IS" BASIS. CARNEGIE MELLON UNIVERSITY MAKES NO WARRANTIES OF ANY KIND, EITHER EXPRESSED OR IMPLIED, AS TO ANY MATTER INCLUDING, BUT NOT LIMITED TO, WARRANTY OF FITNESS FOR PURPOSE OR MERCHANTABILITY, EXCLUSIVITY, OR RESULTS OBTAINED FROM USE OF THE MATERIAL. CARNEGIE MELLON UNIVERSITY DOES NOT MAKE ANY WARRANTY OF ANY KIND WITH RESPECT TO FREEDOM FROM PATENT, TRADEMARK, OR COPYRIGHT INFRINGEMENT.

Use of any trademarks in this presentation is not intended in any way to infringe on the rights of the trademark holder.

This work was created in the performance of Federal Government Contract Number FA8721-05-C-0003 with Carnegie Mellon University for the operation of the Software Engineering Institute, a federally funded research and development center. The Government of the United States has a royalty-free government-purpose license to use, duplicate, or disclose the work, in whole or in part and in any manner, and to have or permit others to do so, for government purposes pursuant to the copyright license under the clause at 252.227-7013.

Scope of this talk

An overview of some interesting populations in network security

Some preliminary modeling ideas

Some cool data sets

Further details of technical approaches in references

Network Security

A young, emerging field:

- Rapidly changing landscape
- Massive data sets (alerts, infections, communications, malicious files)
- Operational focus
- Heuristic and exploratory analyses
- Raw reporting, little modeling or interpretation

CERT initiative: publish defensible security metrics

- Track Internet-wide growth and change in malicious populations
- Evaluate counter-measure policies
- Prioritize clean-up and intervention efforts

Good metrics are the basis of good risk management

Network Security as Risk Management

Priority number one: A system of security metrics:

“Within a decade, we must have a body of quantitative information risk management as sophisticated as the then existing body of financial risk management […] A clearinghouse review of what we know how to measure and how good what we know is at predicting the future would be a good start, as we do not even know what it is that we do not know.”

--Dan Geer, testimony to the US Subcommittee on Emerging Threats, Cybersecurity, and Science and Technology, April 2007

Three Malicious Populations

Three Malicious Populations-- Botnets

A network of thousands of compromised machines controlled by a single operator

Uses

- send spam email
- obfuscate malicious servers
- cripple communications with DDoS attacks

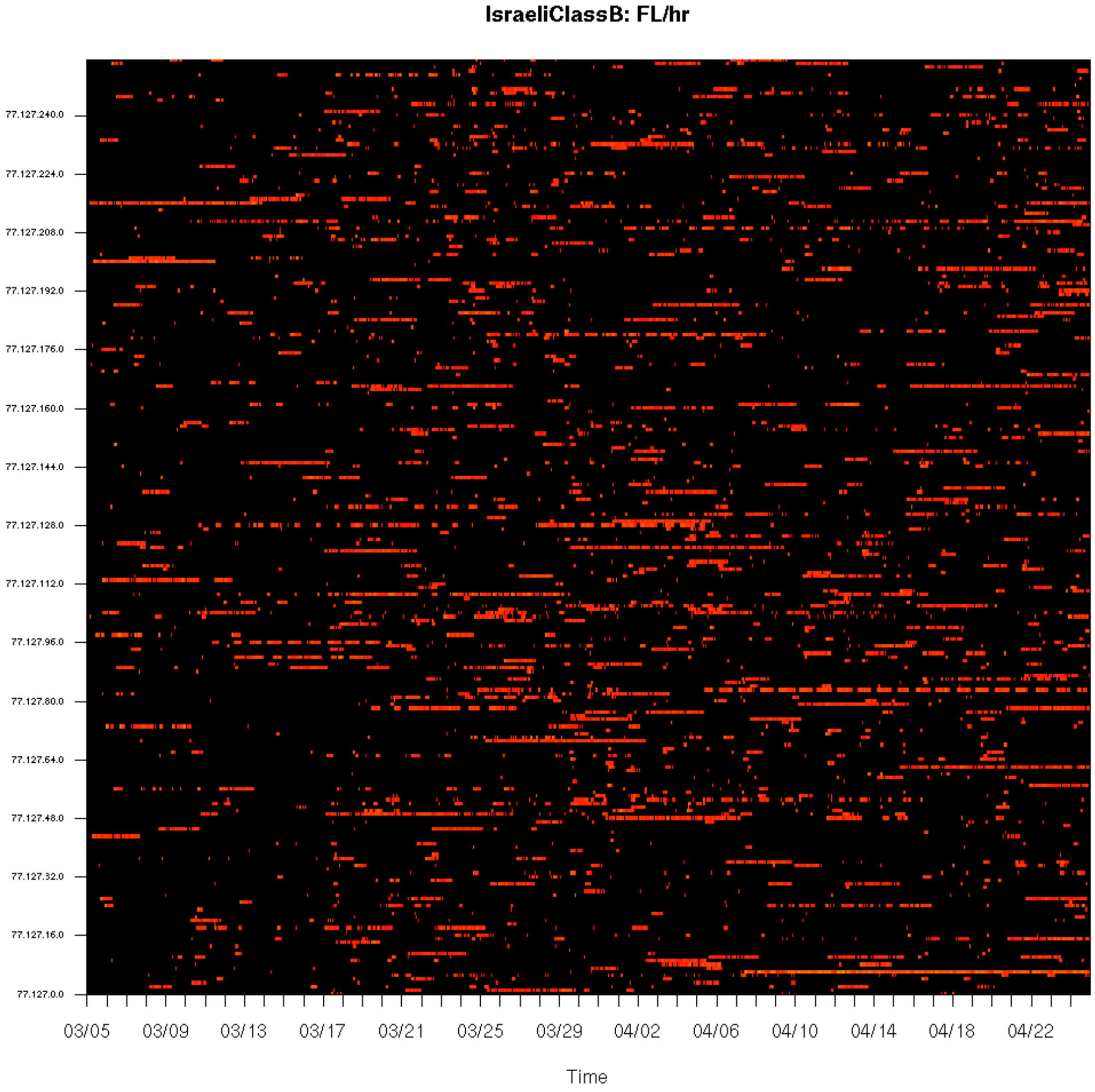
Botnets must communicate with each other, or with a central controller, to remain a co-ordinated threat

How big is it? How many infected computers?

Three Malicious Populations-- Botnets

How big is it?

How many infected computers?

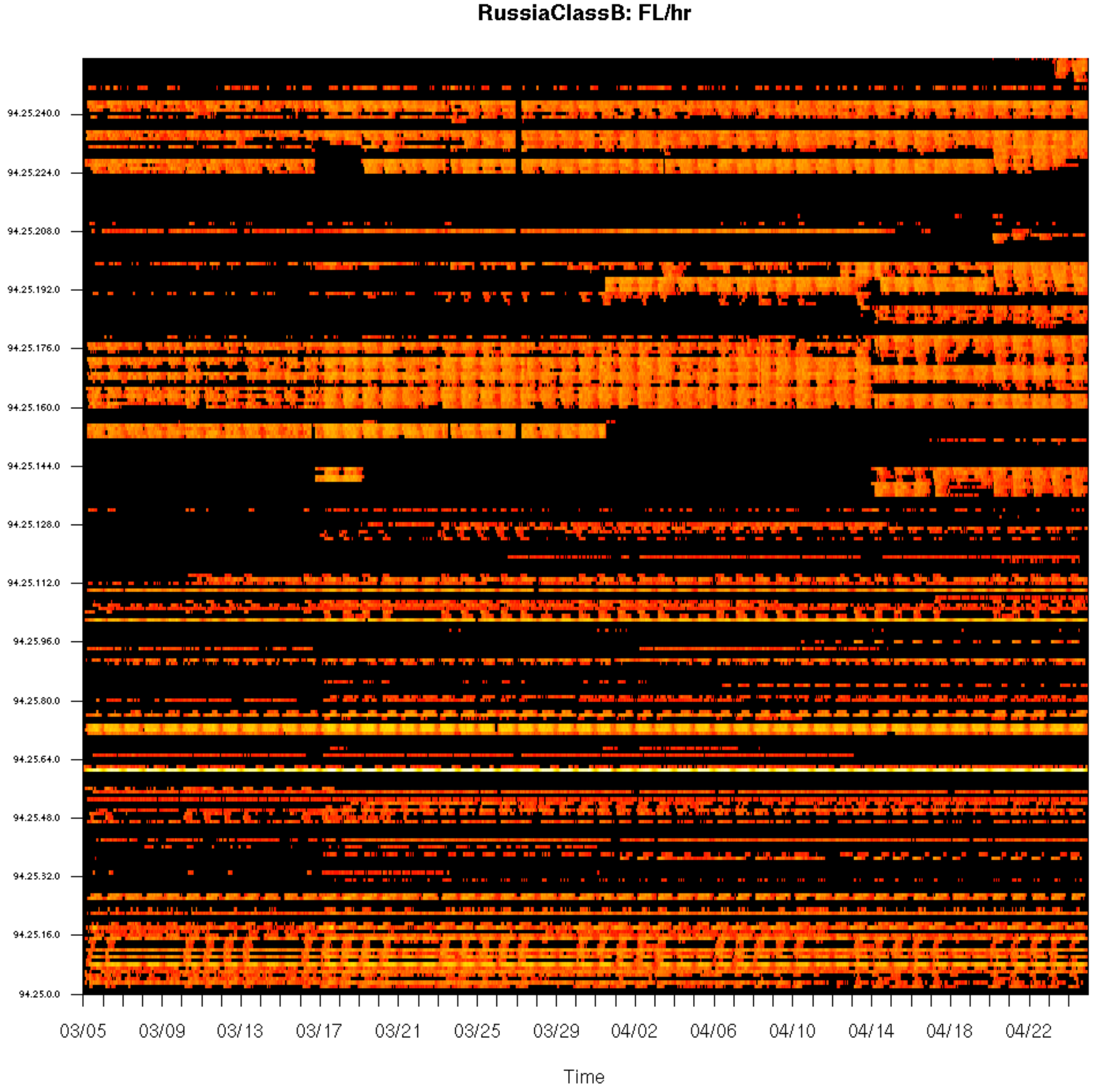


Indirect observation and heterogeneity

How big is it?

How many infected computers?

We want to count machines but we only see IP addresses

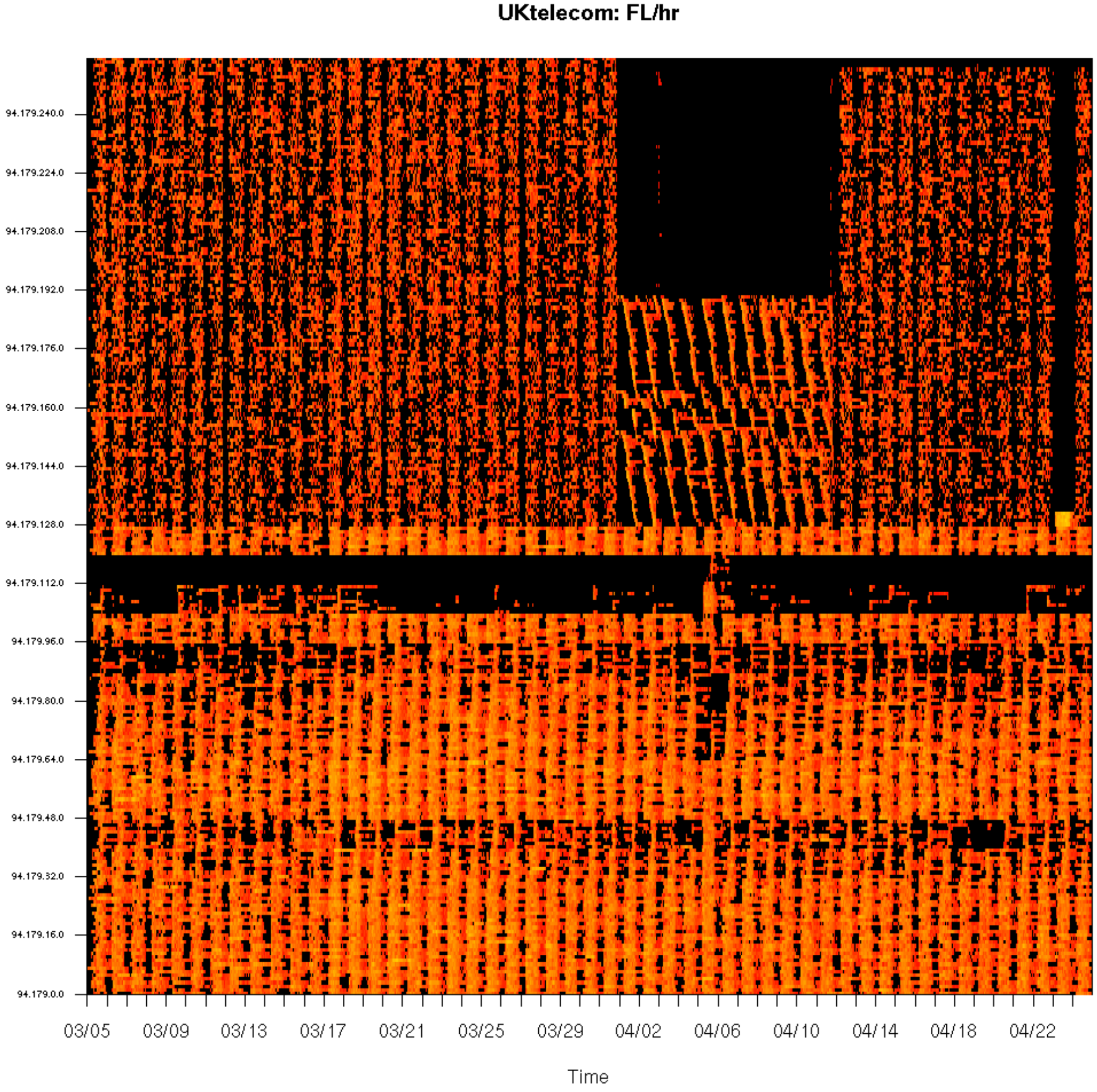


Indirect observation and heterogeneity

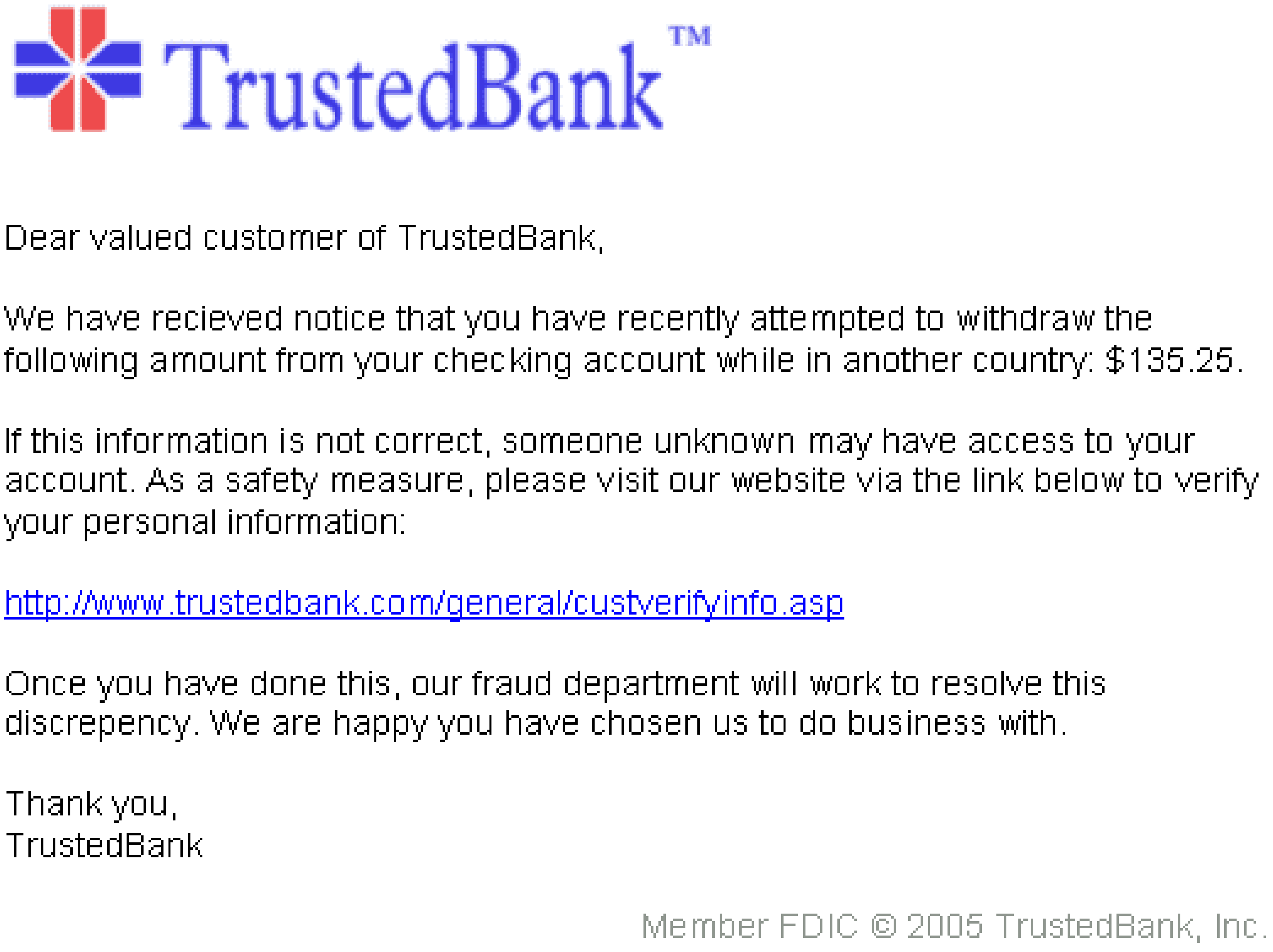
How big is it?

How many infected computers?

We want to count machines but we only see IP addresses



Three Malicious Populations-- Phishing



TrustedBank™

Dear valued customer of TrustedBank,

We have recieved notice that you have recently attempted to withdraw the following amount from your checking account while in another country: $135.25.

If this information is not correct, someone unknown may have access to your account. As a safety measure, please visit our website via the link below to verify your personal information:

http://www.trustedbank.com/general/custverifyinfo.asp

Once you have done this, our fraud department will work to resolve this discrepency. We are happy you have chosen us to do business with.

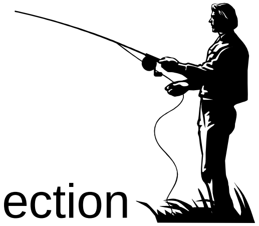
Thank you,
TrustedBank

Member FDIC © 2005 TrustedBank, Inc.

How many sites are there? How many perpetrators? How long do sites live?

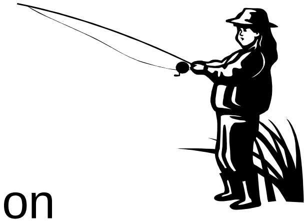
Imprecise Individuals (Jan-March '07)

Netcraft



- 36820 records
- commercial, toolbar collection
- individual targets, no further tracking

PIRT



- 20816 records
- volunteer collection
- aggregate targets, death dates

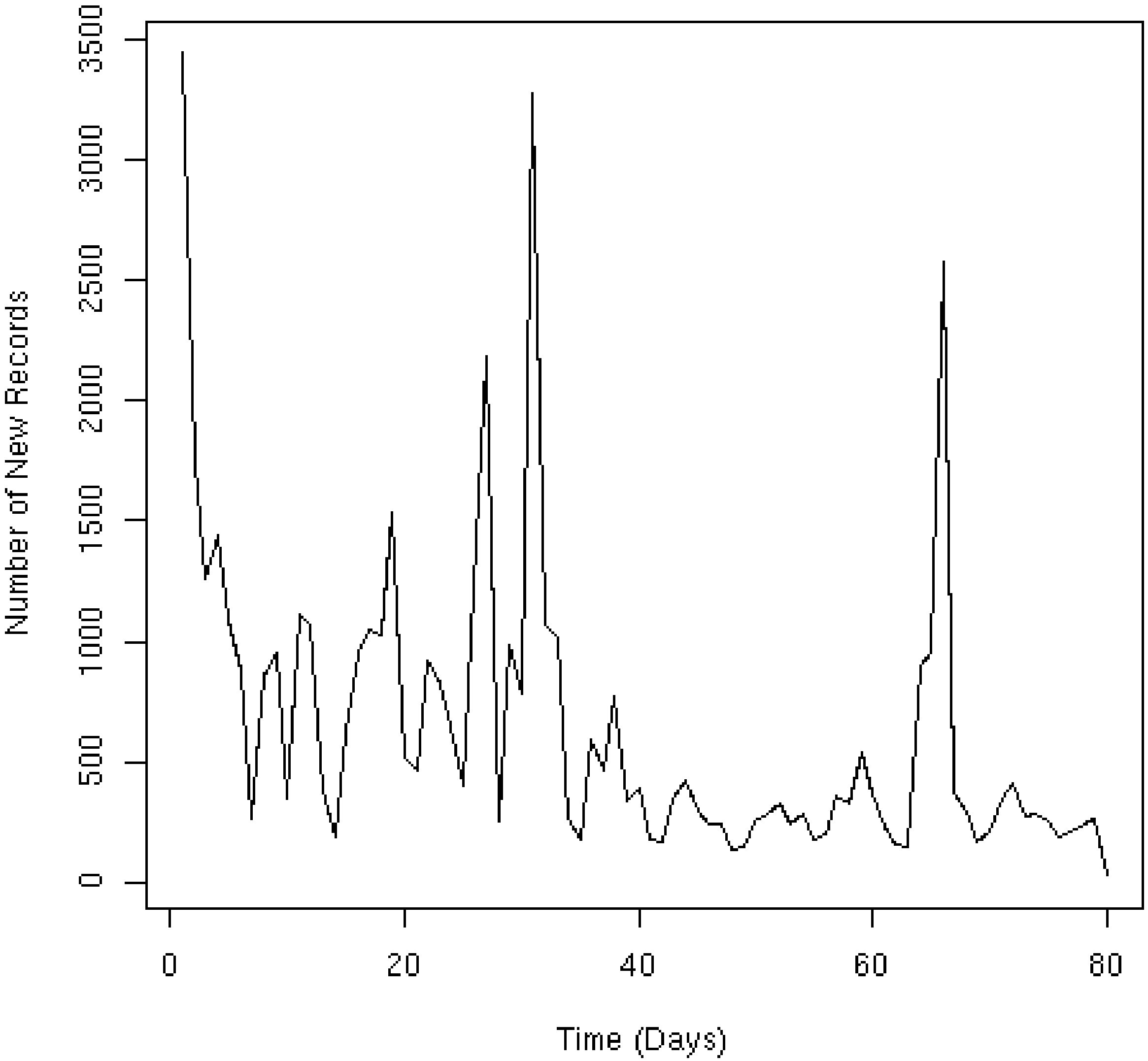
Each record contains:

- target (eg., eBay, Bank of America)
- URL
- IP where the URL resides
- date reported

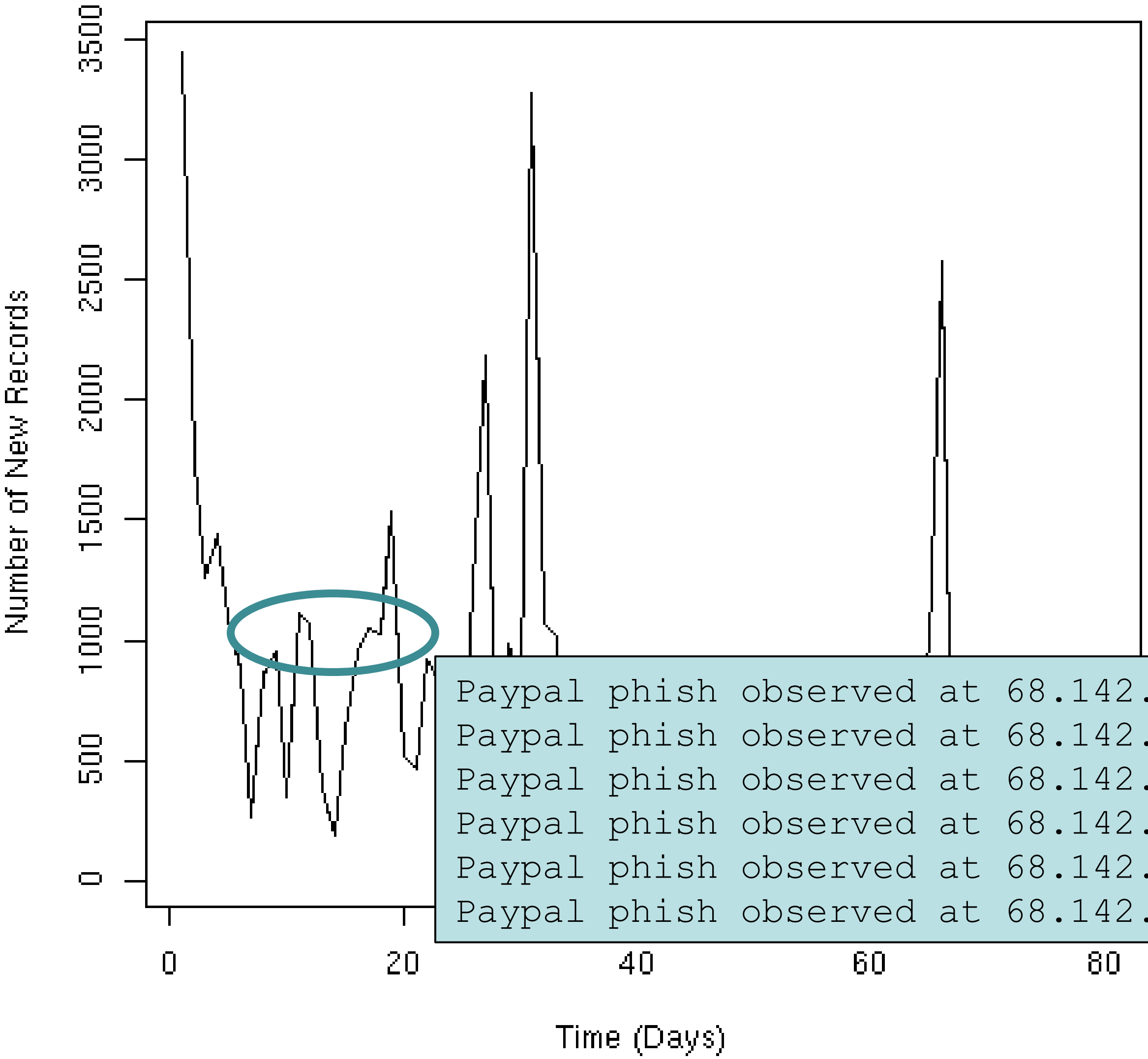
But each record isn't necessarily a single "phish":

- Kit, reporter, URL, IP make records dependent in capture probability
- We want to construct the conditionally independent (if heterogenous) "individual"

Fast flux and domain flux

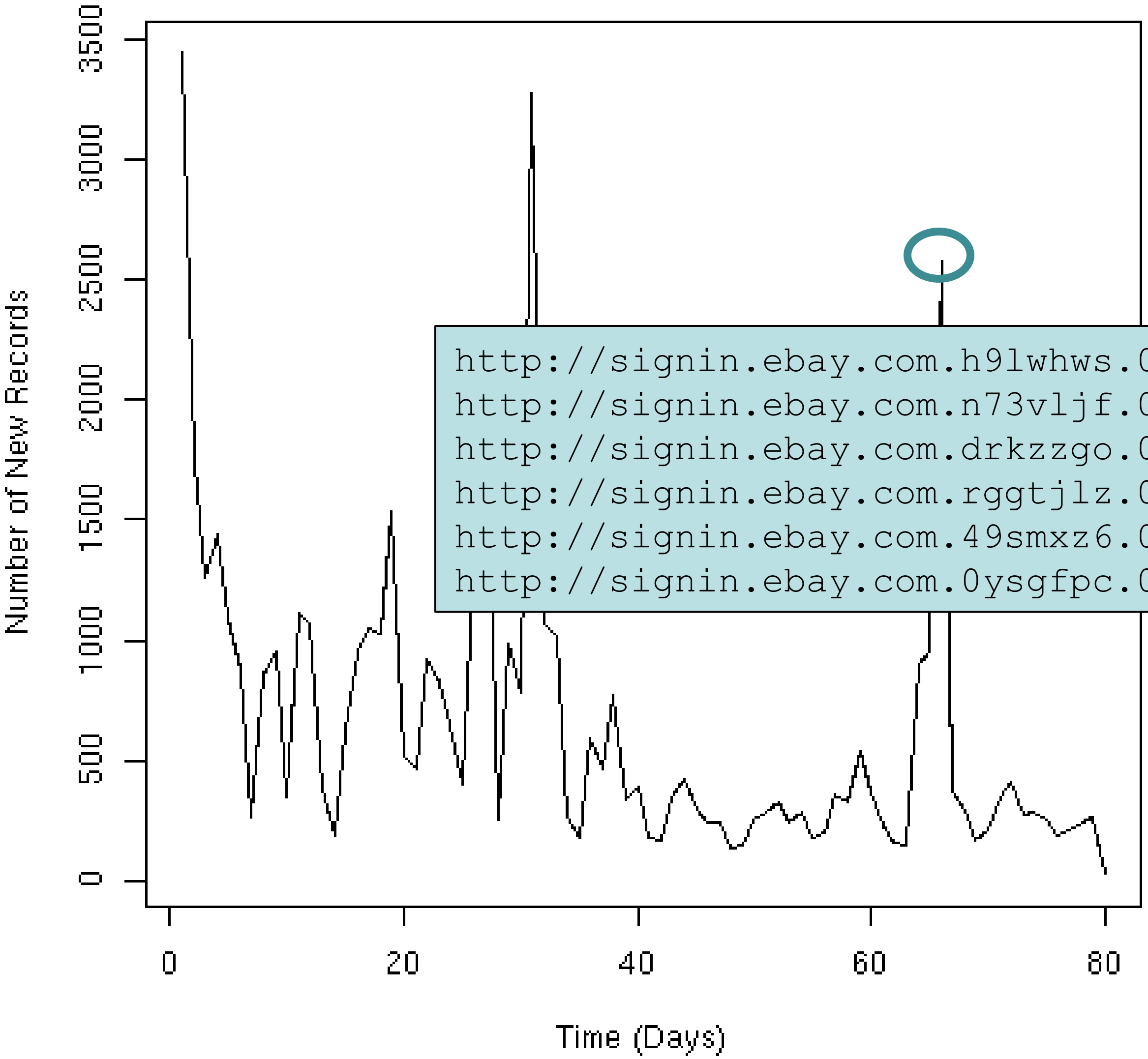


Fast flux and domain flux



```
Paypal phish observed at 68.142.212.23
Paypal phish observed at 68.142.212.18
Paypal phish observed at 68.142.212.20
Paypal phish observed at 68.142.212.19
Paypal phish observed at 68.142.212.22
Paypal phish observed at 68.142.212.23
```

Fast flux and domain flux



```
http://signin.ebay.com.h9lwhws.03iana.com/.../eBayISAPI.php
http://signin.ebay.com.n73vljf.03iana.com/.../eBayISAPI.php
http://signin.ebay.com.drkzzgo.03iana.com/.../eBayISAPI.php
http://signin.ebay.com.rggtjlz.03iana.com/.../eBayISAPI.php
http://signin.ebay.com.49smxz6.03iana.com/.../eBayISAPI.php
http://signin.ebay.com.0ysgfpc.03iana.com/.../eBayISAPI.php
```

Three Malicious Populations-- Malware

Malicious code:

- Viruses
- Trojans
- Spyware

CERT has a large artifact catalogue (~8 million unique files), but viruses often can have thousands of variants

AV-vendor IDs

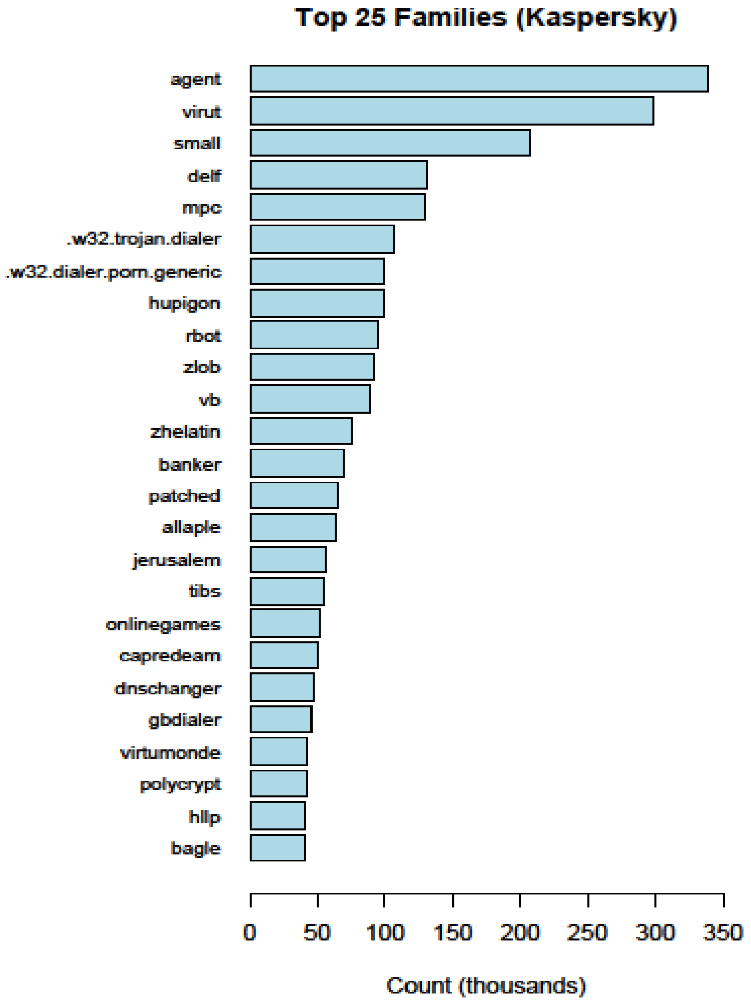
```
   Unique ID    ;     VendorA      ; [...] ;     VendorZ
‘‘00afd123000’’ ; ‘‘BAT/virut.ZZ’’ ; [...] ; ‘‘Virut-2G-A.x’’
‘‘00bd34289ac’’ ;      ‘‘-’’       ; [...] : ‘‘Foo (suspicious)’’
```

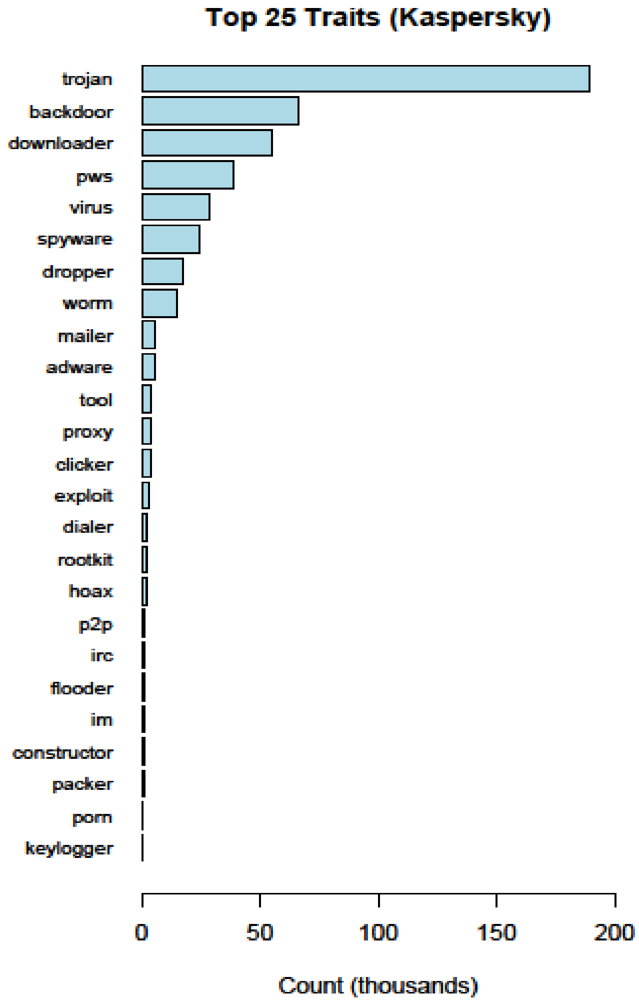
In-house analysis/classification into families and variants:

- Functional extraction
- Entry point similarity

How many unique families are there? What types are easy to detect?

Polymorphism, lists, and obfuscation





Creating defensible population metrics

Measurement Methods

- Signature detection
- Publicly available watch lists/blacklists
- Network communications monitoring and honeypots

Raw counts are not always stable or interpretable

- Dynamic, ephemeral locations on the Internet
- Short life cycles
- Polymorphic “individuals”
- Indirect observation

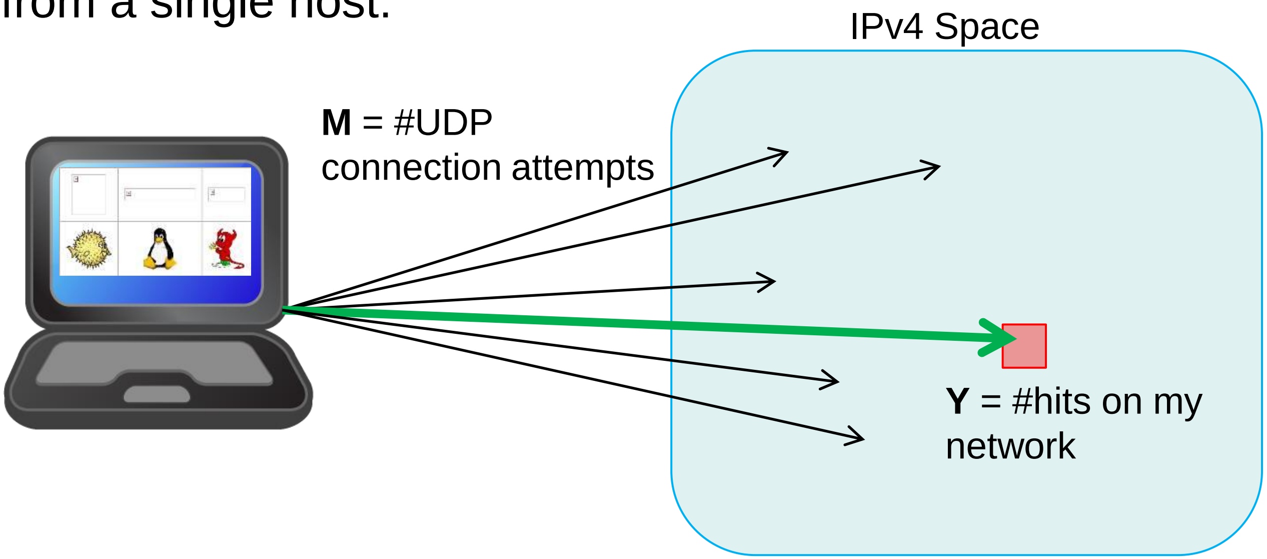
GLM-based mark-recapture models must be adapted

- Lots of literature:
  - open populations
  - observable heterogeneity
- Less methodology:
  - indirect observation of imprecise individuals
  - multiple lists and multiple samples

Preliminary adaptations

Botnets: Exploit common software base

Model activity from a single host:



M ~ Poisson(λ)
Y|M ,p ~ Binomial(M, p)
Y| λ, p ~ Poisson(λ p)

S = $Y_1 + Y_2 + \ldots + Y_H$ is approximately Normal(mean=H*mean(Y), var=H*var(Y))
Central Limit Theorem 95% CI for H is: $S/(\lambda p)$ +/- $1.96(S^{1/2})/(\lambda p)$

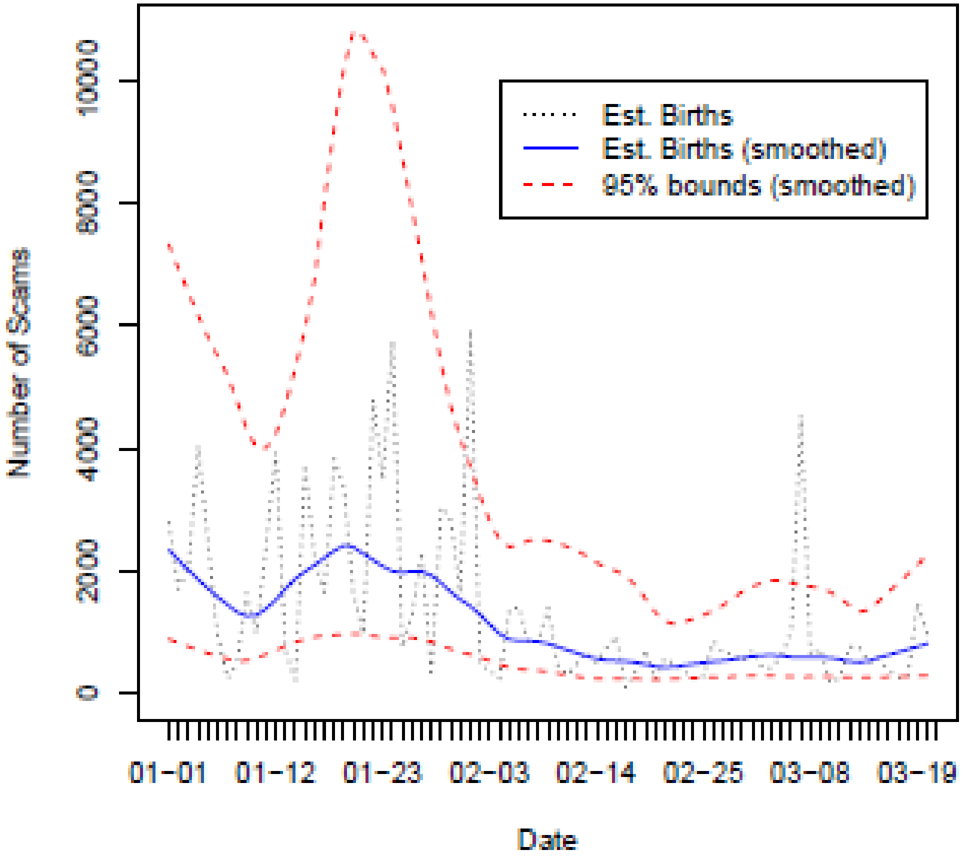
Weaver (2010). “A Probabilistic Population Study of the Conficker-C Botnet”. *PAM 2010* proceedings, Zurich, Switzerland

Phishing: Heuristic Clustering

Creating “scams”:

- same target
- similar URLs (levenshtein dist.< 20)
- nearby IP addresses

Match scams across lists for simple capture-recapture model

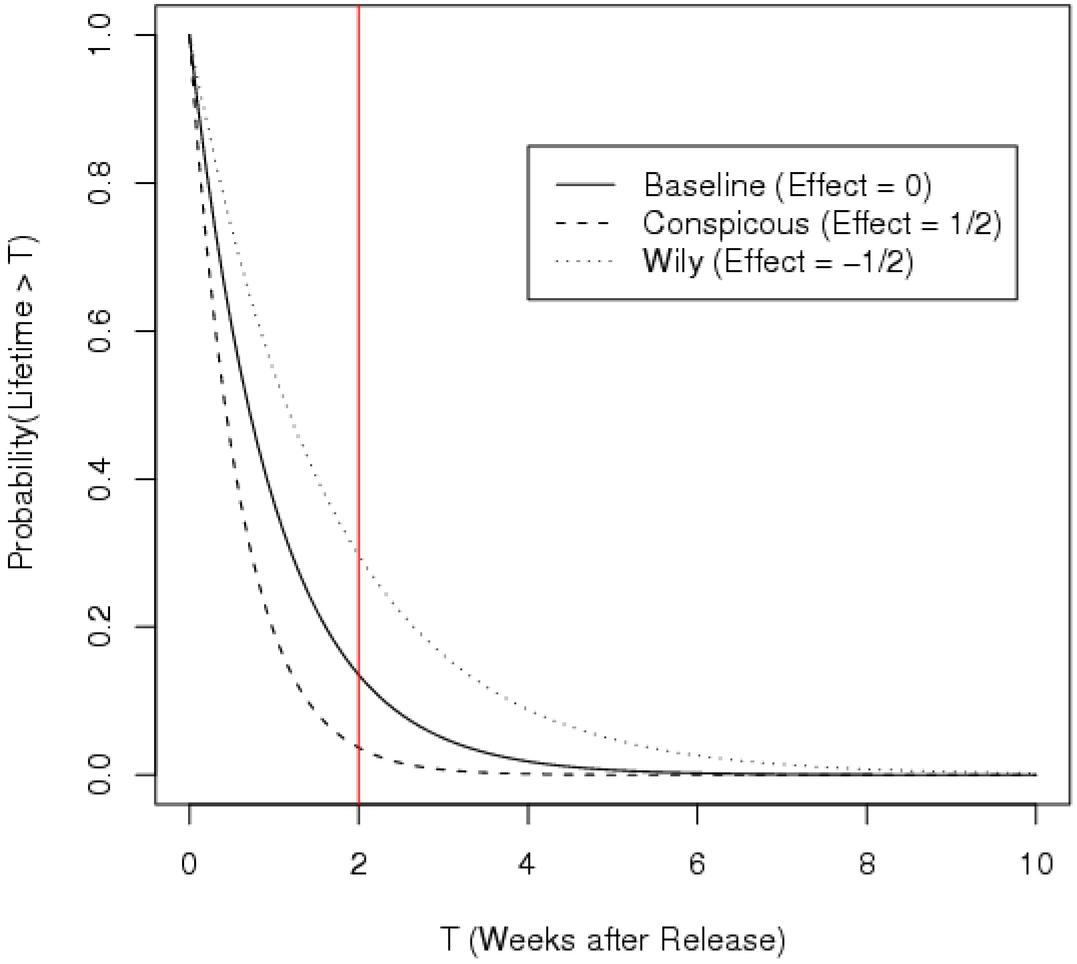


Weaver and Collins (2007). “Fishing for phishes: applying capture-recapture methods to estimate phishing populations” APWG eCrime researcher summit, October 4-5, 2007, Pittsburgh, PA

Malware: Continuous time list capture

Measure time from release until detection

A “wily” file takes longer to be noticed than a conspicuous one



Malware: Continuous time list capture

List heterogeneity:

$$Pr(\text{Lifetime}_j > T) = \frac{1}{e^{\left({}_{Te}\sum \text{trait effects}_j\right)^{\text{rate}_j}}}$$

Assuming lists act independently, overall survival rate:

$$Pr(\text{Lifetime} > T) = \prod_{j=1}^{J} Pr(\text{Lifetime}_j > T)$$

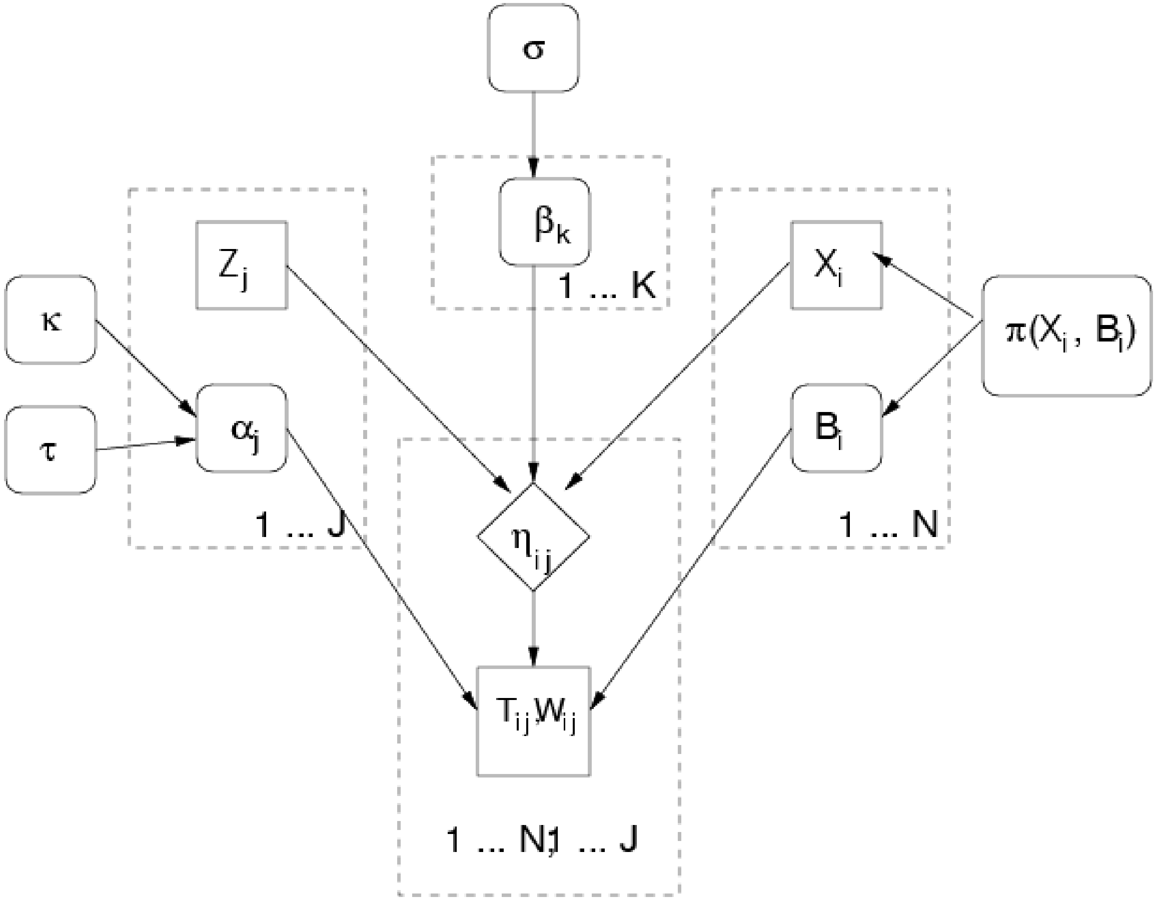
A population estimate for each malware strata at time t_0:

$$N_{\text{observed}} = N_{\text{total}} * [1 - Pr(\text{Lifetime} > t_0)]$$

Malware: Continuous time list capture

Related work: survival analysis (staying uncaught) with competing risks (caught by different lists)

Simulated data currently shows non-identifiability in list rates, covariates



Weaver (2010). A continuous time list capture model for Internet threats. In *JSM Proceedings*, Risk Analysis section. Vancouver, BC, Canada: American Statistical Association.

Summary

The data we record for security metrics is getting more ephemeral:

- IP addresses (especially in light of IPv6 and clouds)
- Fast flux, domain flux, what next?
- Polymorphism in unique files
- Multiple lists, shorter lifetimes

Adversaries are more intelligent than nature, actively adapting in unforseen ways:

- Affect stability of raw counts
- Inference/smoothing is needed to track “useful” populations

There is a lot of room for extending/adapting mark-recapture models to fully Bayesian hierarchical models.

We have lots of data. We’re looking for more and for collaborations!

CERT

Thank you!

rweaver@cert.org

Extra Slides

Botnets

The Big Question: “How big is it?”

- Many reports on blogs, media
- Hard to tell methodology and biases
- Observe and report IPs but want to know the number of machines

Two ways to think about botnet “size”

- Active size at any particular time slice
- “Footprint” size, accounting for dormancies, new infections, cleanup

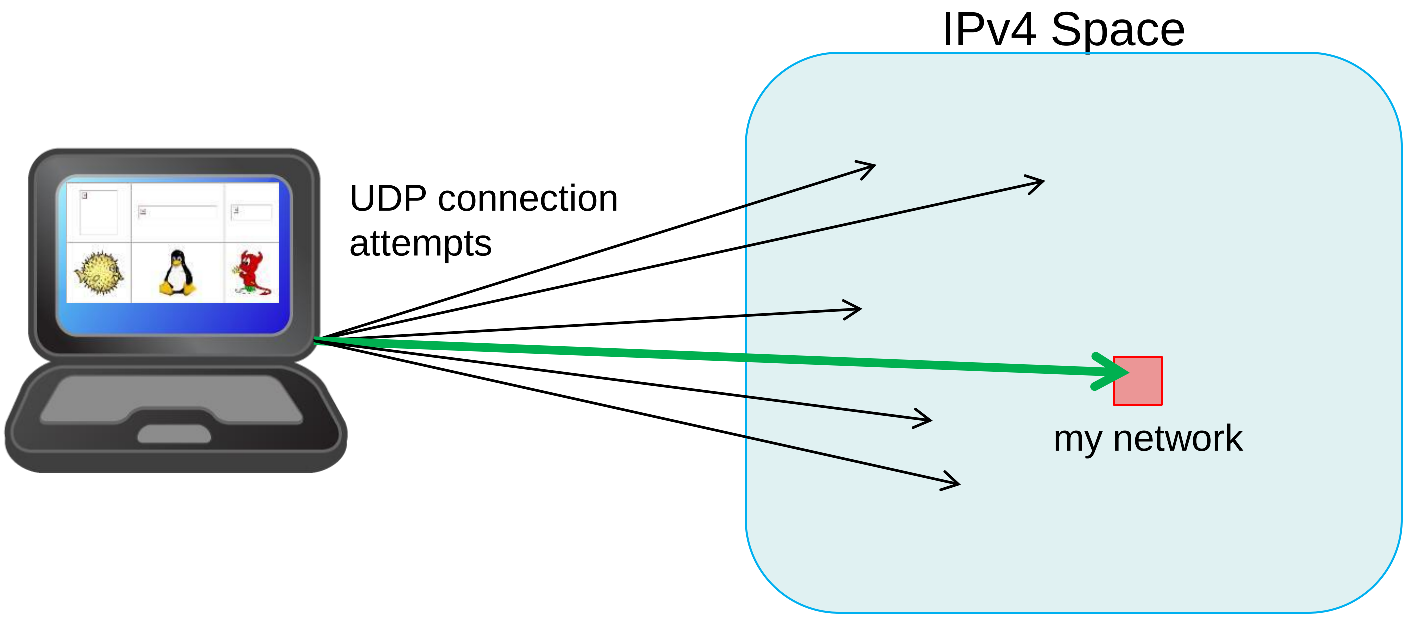
Kaleidoscope view of machines

- Passive scan detection from multiple honeynets
- Count machines, but view only IP addresses
- One-to-many (DHCP) and Many-to-one (NAT) relationships exist

Botnet: Conficker-C

Released March 2009:

- Installed on Conficker-A, Conficker-B infected hosts
- Sinkhole domain monitoring (not talking about this today)
- P2P bootstrap by random scanning (talking about this today)
- Ports determined by IP and start time = Accurate flow-based signature



Indirect Observation

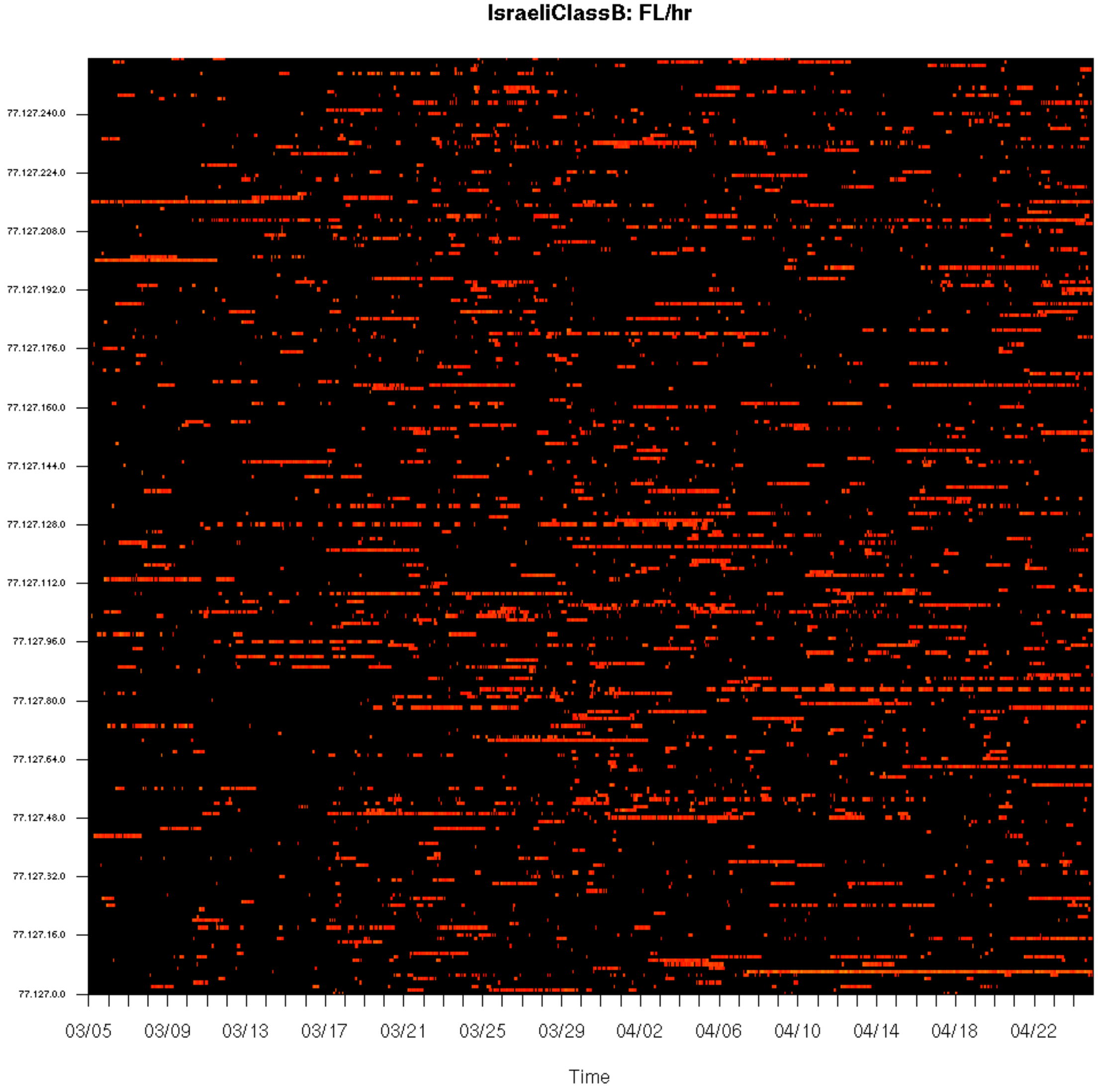
Network telescope into infections:

“my network” has about 21,000 class C net blocks

I count the number of UDP requests I see coming from net blocks all over the outside world

33 mil. IPs, 1.1 mil. blocks

They sometimes look like this →



Indirect Observation

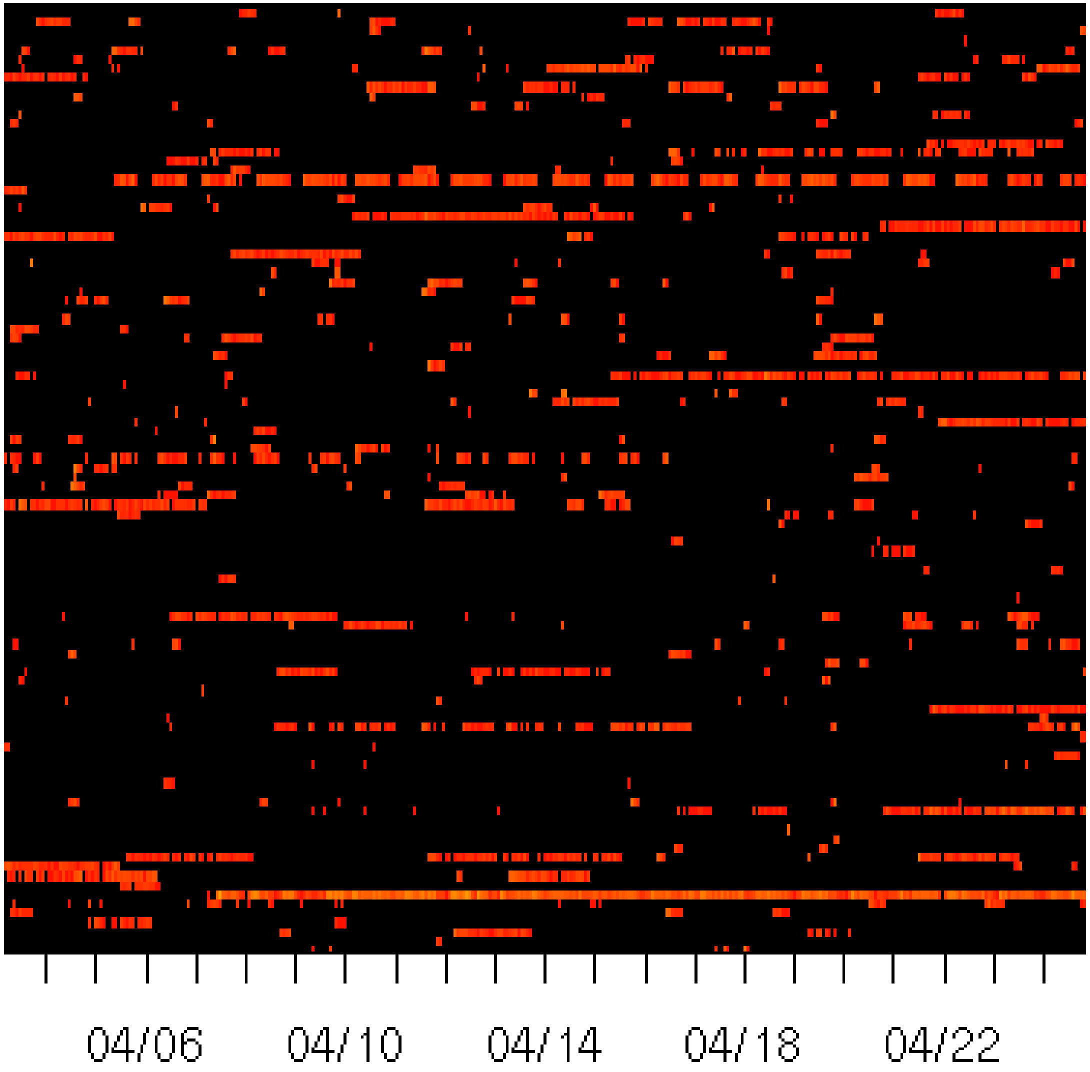
Network telescope into infections:

“my network” has about 21,000 class C net blocks

I count the number of UDP requests I see coming from net blocks all over the outside world

33 mil. IPs, 1.1 mil. blocks

They sometimes look like this →



Indirect Observation

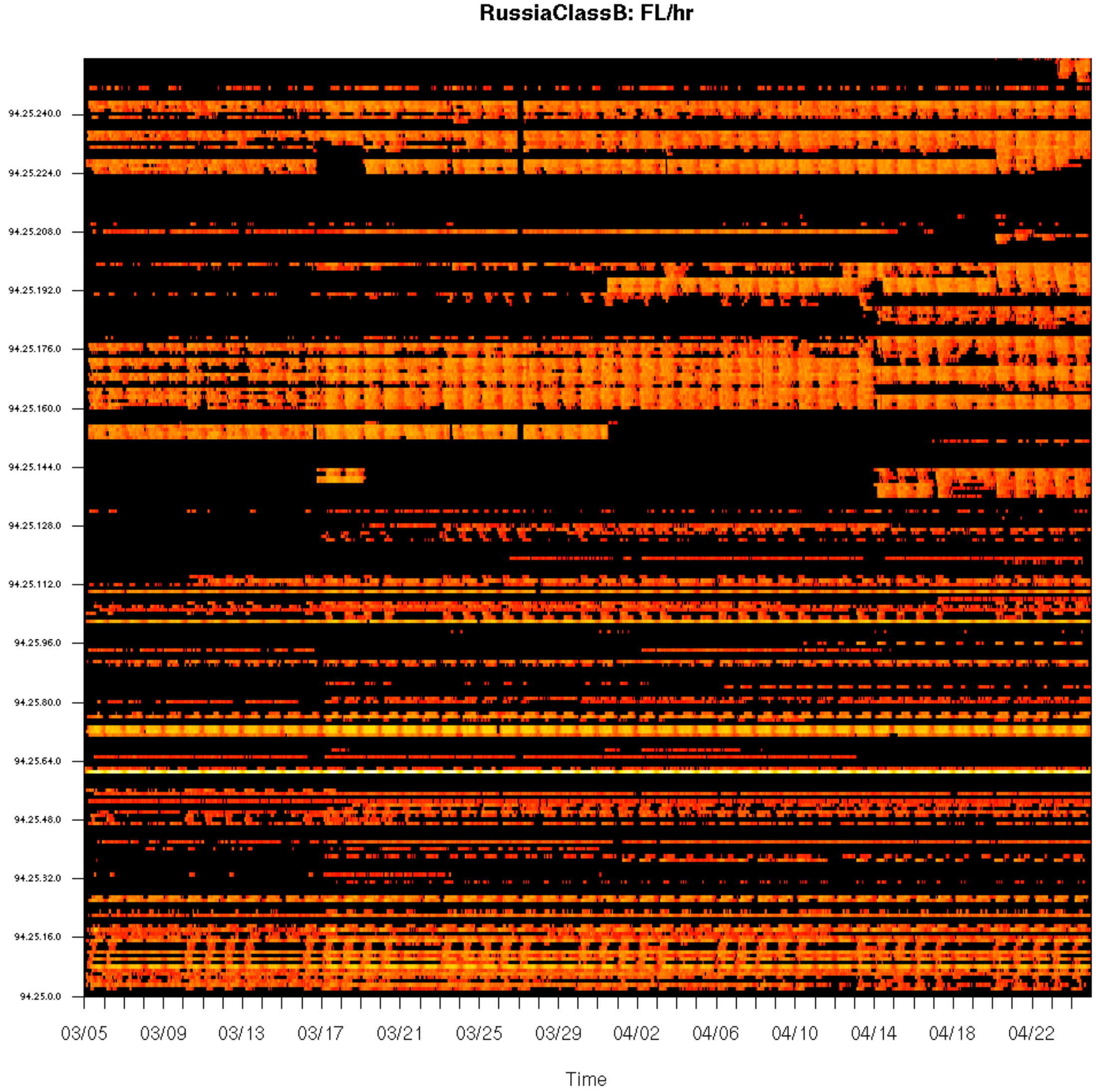
Network telescope into infections:

“my network” has about 21,000 class C net blocks

I count the number of UDP requests I see coming from net blocks all over the outside world

33 mil. IPs, 1.1 mil. blocks

…but they also look like this →



Indirect Observation

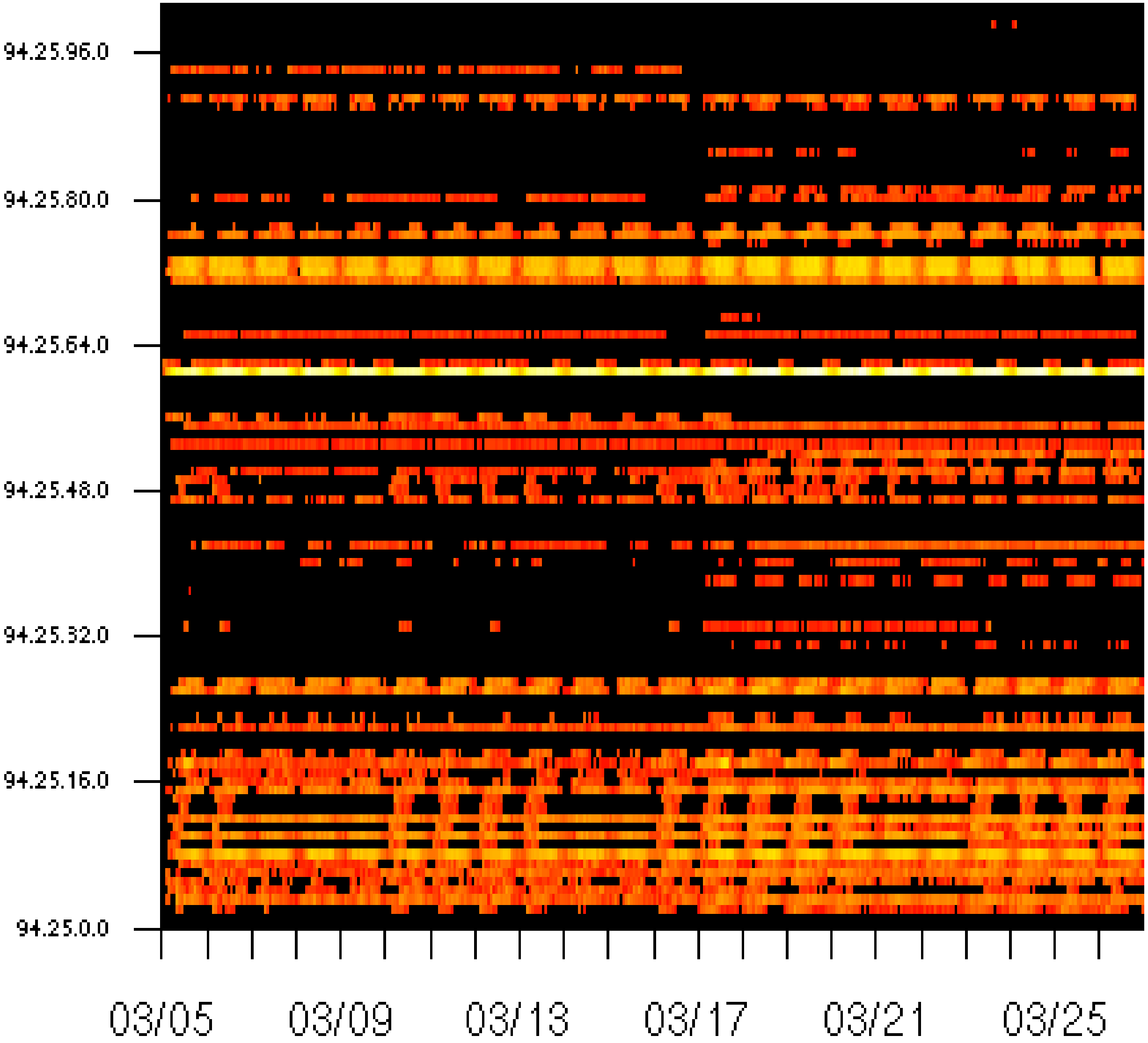
Network telescope into infections:

“my network” has about 21,000 class C net blocks

I count the number of UDP requests I see coming from net blocks all over the outside world

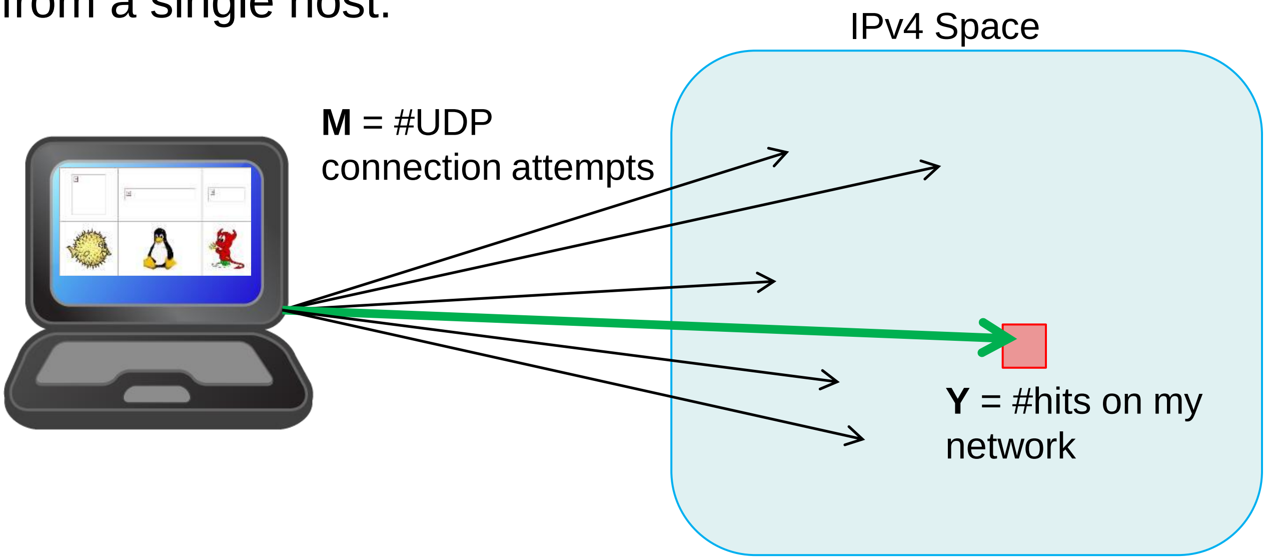
33 mil. IPs, 1.1 mil. blocks

…but they also look like this →



Exploit common software base

Model activity from a single host:



M ~ Poisson(λ)
Y|M ,p ~ Binomial(M, p)
Y| λ, p ~ Poisson(λ p)

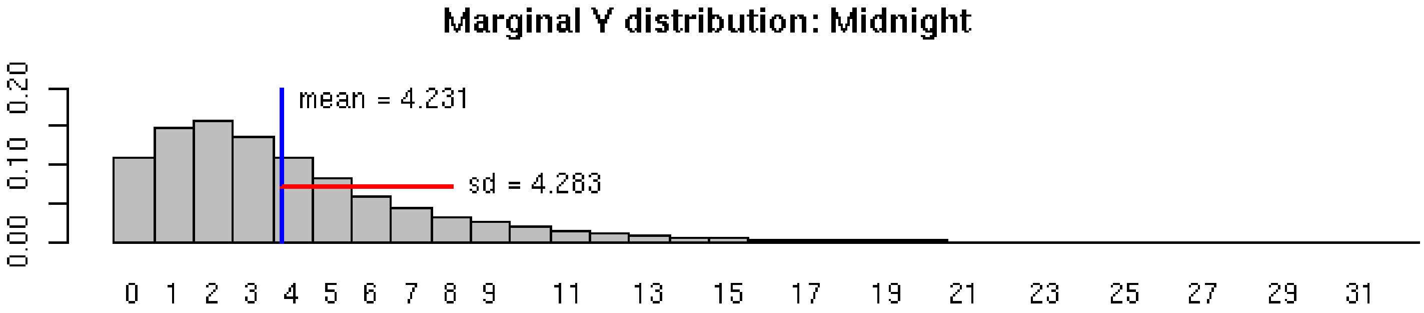
S = $Y_1 + Y_2 + \ldots + Y_H$ is approximately Normal(mean=H*mean(Y), var=H*var(Y))

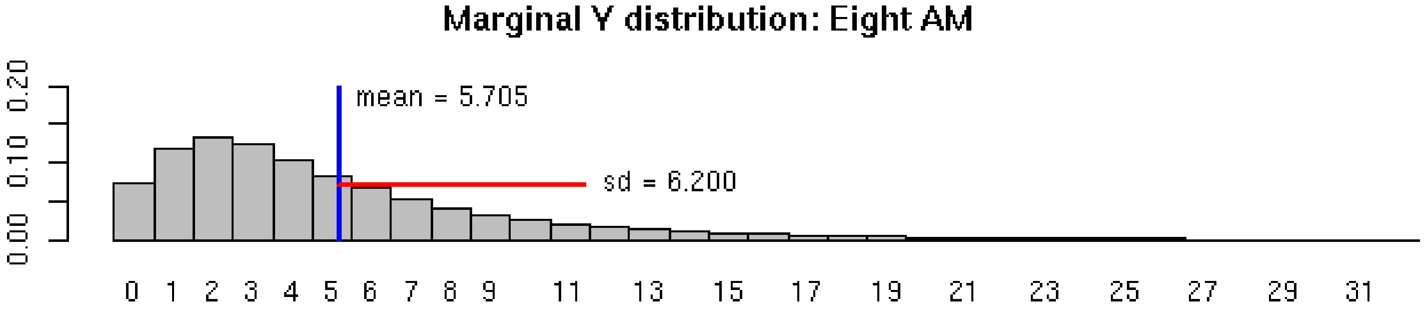
So a Central Limit Theorem 95% CI for H is: S/(λp) +/- 1.96($S^{1/2}$)/(λp)

Weaver (2010). "A Probabilistic Population Study of the Conficker-C Botnet". PAM 2010

Scans per Hour for One Host

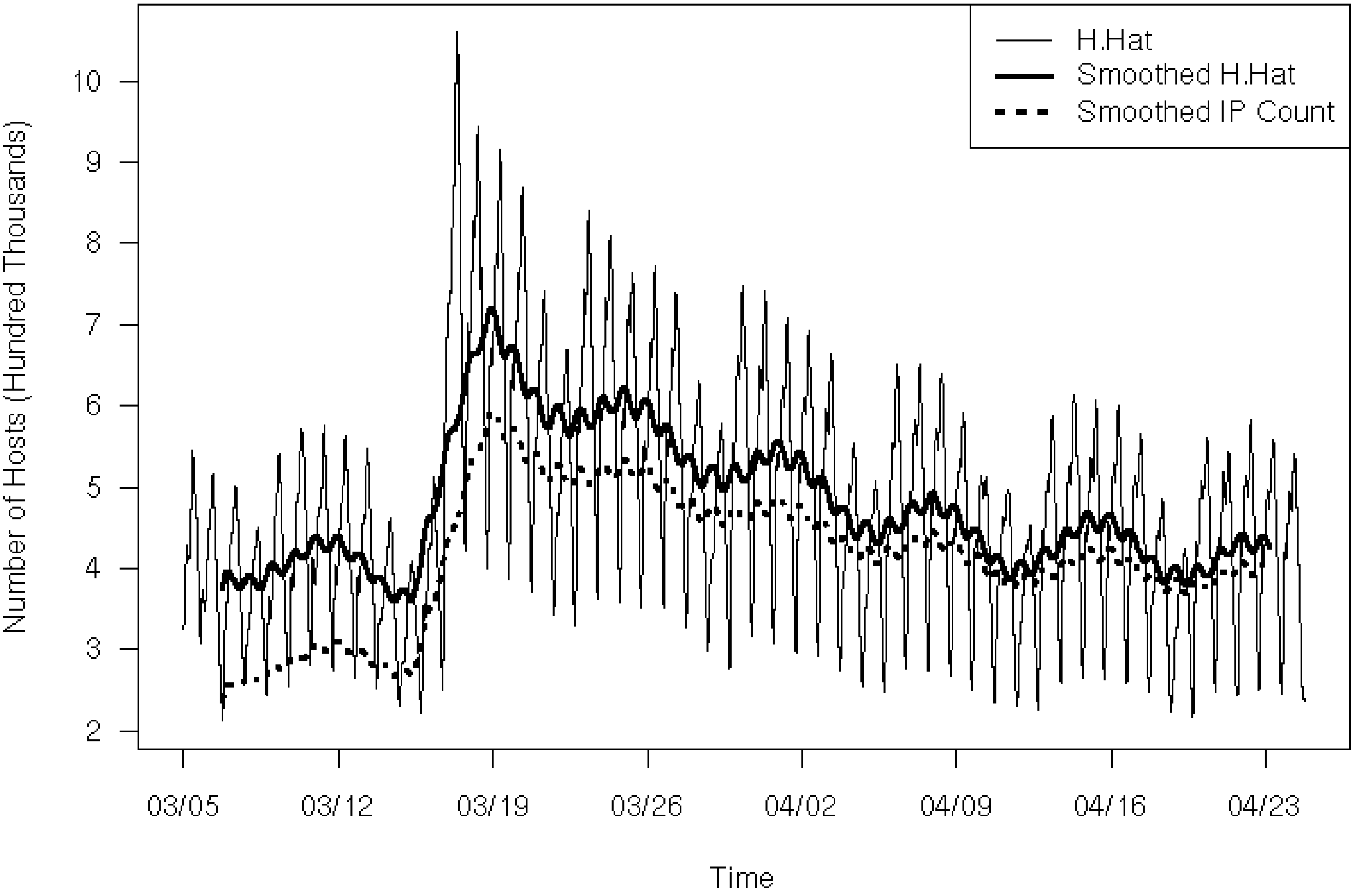
Simulations from the model





The Central Limit Estimator still holds, using the marginal mean and sd (as above) instead of mean=sd=λp

Active Size Estimates (Preliminary)



Open Questions

Central Limit estimator is straightforward, but measuring hyperparameters is not:

- Mine are not that great, yet
- More can be done to get better information
  - Sandbox experiments on a single host
  - Iterative extraction of “one-host” data from the existing data
- Plan to implement a fully Bayesian model to update priors to posteriors given data
- Formal sensitivity analysis

Toward a footprint model:

- Active size estimate does not use information from hour to hour
- λ_t likely depends on λ_{t-1} within individual hosts
- Some “clean-ups” or “births” are merely jumps within network IP space

Phishing Sites: Netcraft and PIRT watch lists

Phishing Scams

Phishing scams on the large scale:

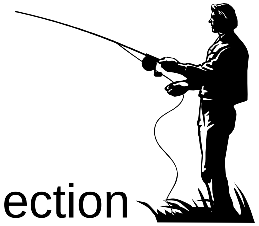
- Black market phishing “kits” are readily available
- Organized groups run dedicated businesses, using spam, botnets and “money mules”
- Groups use DDoS attacks against institutions trying to shut them down
- Revenue estimated at US $150 million for one gang in 2006, billions more in damages[1]

Operationalizing phishing data under uncleanliness:

- Build IP watch lists for phishing activity, find where to look for other threats
- How can we use information from multiple list sources?
- When can we stop collecting data?

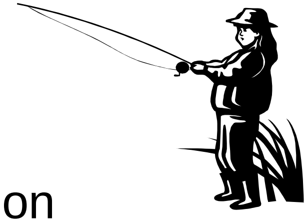
Imprecise Individuals (Jan-March ’07)

Netcraft



- 36820 records
- commercial, toolbar collection
- individual targets, no further tracking

PIRT



- 20816 records
- volunteer collection
- aggregate targets, death dates

Each record contains:

- target (eg., eBay, Bank of America)
- URL
- IP where the URL resides
- date reported

But each record isn’t necessarily a single “phish”:

- Kit, reporter, URL, IP make records dependent in capture probability
- We want to construct the conditionally independent (if heterogenous) “individual”

Imprecise Individuals

Clustering records into “scams”:

- group records with the same target, similar URLs (levenshtein distance < 20), nearby IP addresses (within /24, distance < 5)
- For Netcraft, construct approximate death date
- For PIRT, break up aggregated records by target

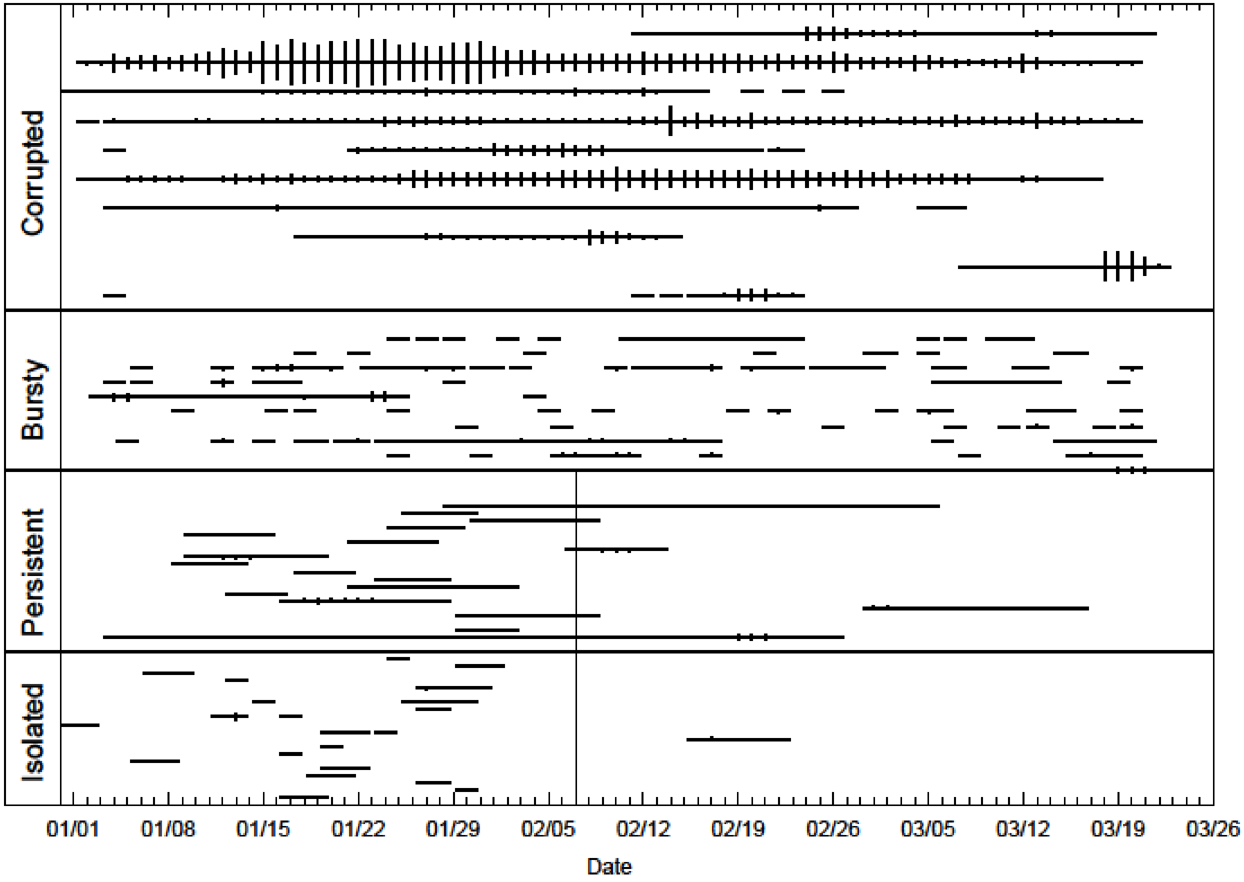
Matching scams across lists:

- Exact match of any URL within a scam
- Merge scams within each list for “many-many” relations

So far, it is an exploratory method;

- Empirical methods for finding “independent” clusters?
- List bias?
- Effect of email address distribution lists
- More experimentation with honeypots

Heterogeneity in Networks



Open Questions

So far, it is an exploratory method;

- Empirical methods for finding “independent” clusters?
- List bias?
- Effect of email address distribution lists
- More experimentation with honeypots

Malware Example: Continuous Time List Capture

Malware

Goal: Quantify a “Big Picture” view of Malware on the Internet

- MD5 files, variants, families
- Growth rates and trends
- Propagation

What we are trying to do

- ”Color the pond” by categorizing malware fitness traits
- Find the directions that sources are casting their nets
- Get a population estimate accounting for these intricacies
- Evaluate how well we are doing

Sample Questions

- How much harder is it to find a key logger than a worm?
- Who is good at finding browser exploits?
- What percentage of existing foo-variants have all my sources found?

Multiple Lists, Competing Risks

Release → Propagation → Discovery → Mediation

A “wily” file takes longer to be noticed than a conspicuous one.

Population estimates depend on the probability of surviving *unnoticed* up to the present.

What is “death”?

- Signature
- Patch
- Platform demise
- Objective completed

In our analysis “survival” is the time from release until discovery by a list.

A file can “die” as many times as there are lists to notice it.

Survival analysis with competing risks

Terminology

Malware: a unique MD5 file

- i = 1, …, N files observed

List: a source recording individual malware files

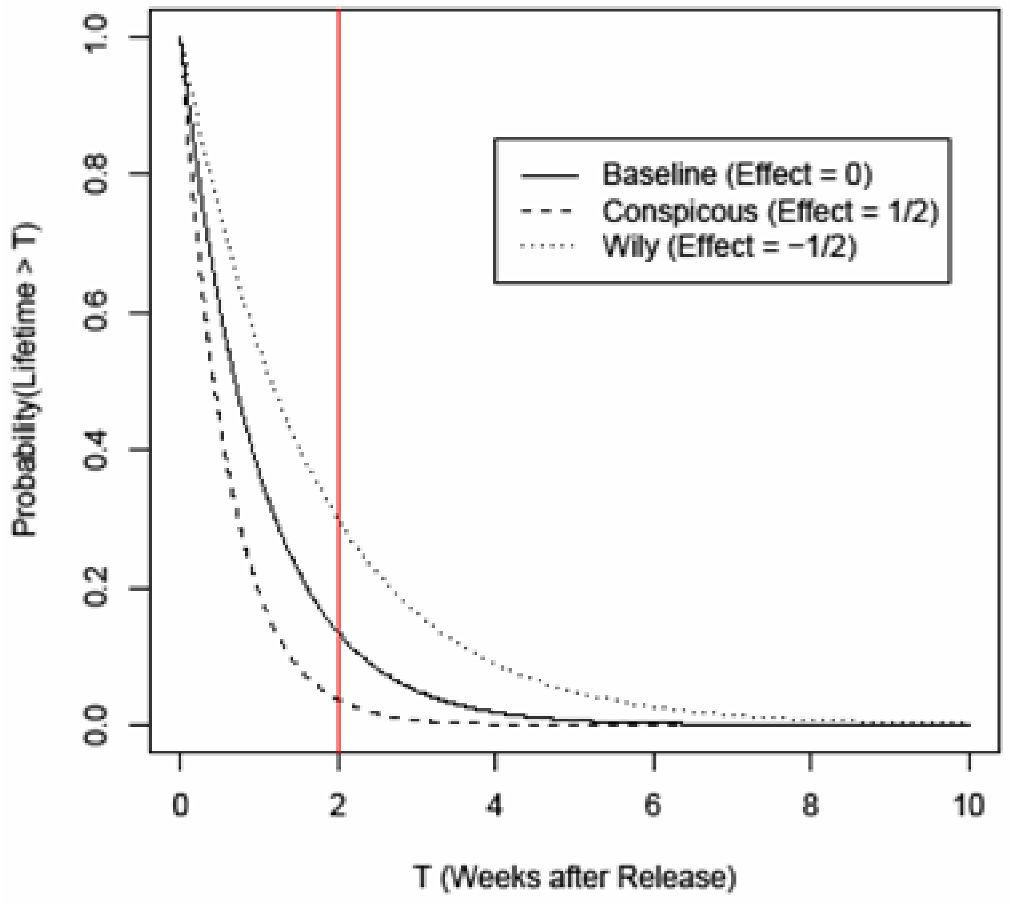
- j = 1, … , J lists available

Lifetime: Time from a file's release on the Internet to appearance on a list

Trait: file feature (Y/N question). The set of unique trait profiles *stratifies* the malware population

Trait effects

$$Pr(\text{Lifetime} > T) = \frac{1}{e^{Te^{\sum \text{trait effects}}}}$$



Multiple lists

List heterogeneity:

$$Pr(\text{Lifetime}_j > T) = \frac{1}{e^{\left(Te^{\sum \text{trait effects}_j}\right)^{\text{rate}_j}}}$$

Assuming lists act independently, overall survival rate:

$$Pr(\text{Lifetime} > T) = \prod_{j=1}^{J} Pr(\text{Lifetime}_j > T)$$

A population estimate for each malware strata at time t_0:

$$N_{\text{observed}} = N_{\text{total}} * [1 - Pr(\text{Lifetime} > t_0)]$$

Simulated data

Malware:

- 832 files released between 0 and 90 weeks
- Releases simulated according to a “clumpy” poisson process
- Traits:
  - 15 “families”
  - IsKeylogger (Y/N)
  - IsTrojan (Y/N)

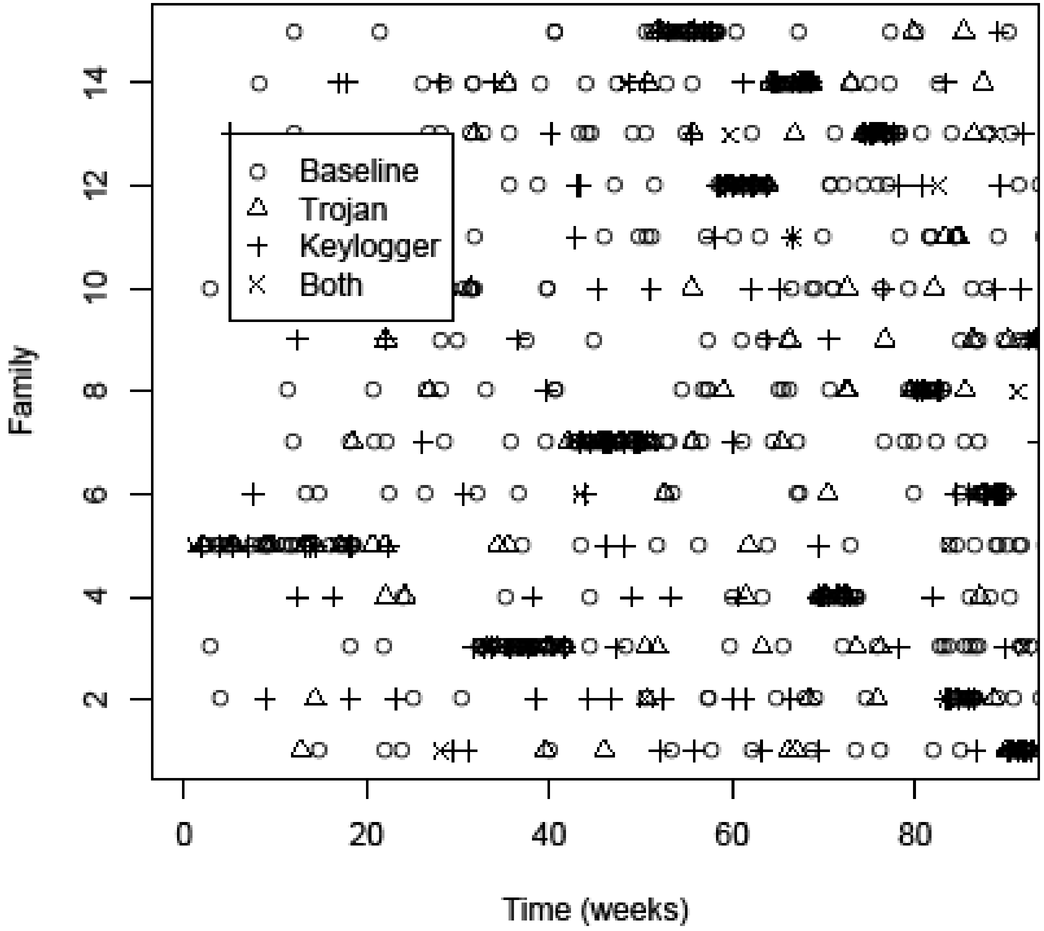
Lists:

- 3 independent lists: rates= (1.5, 2, 2.5)
- Trait effects constant across lists (no list heterogeneity in effects)
- Varying baseline trait effect for each family
- IsKeylogger effect = -0.5 (all families, all lists)
- IsTrojan effect = 0.5 (all families, all lists)

Families and traits

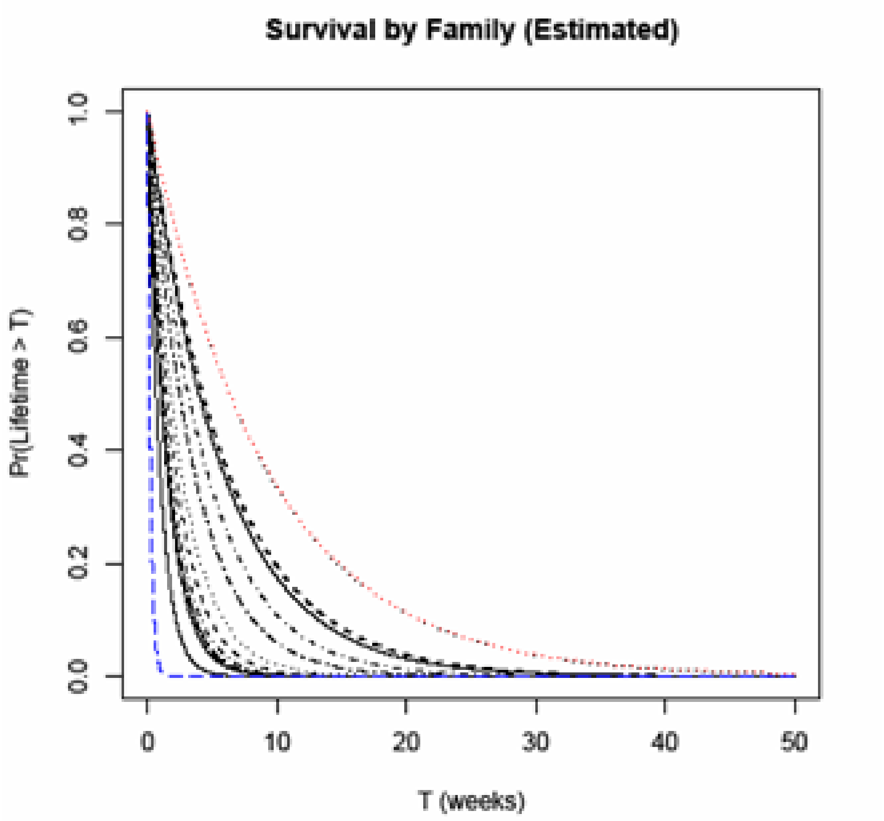
Family Trait Effects

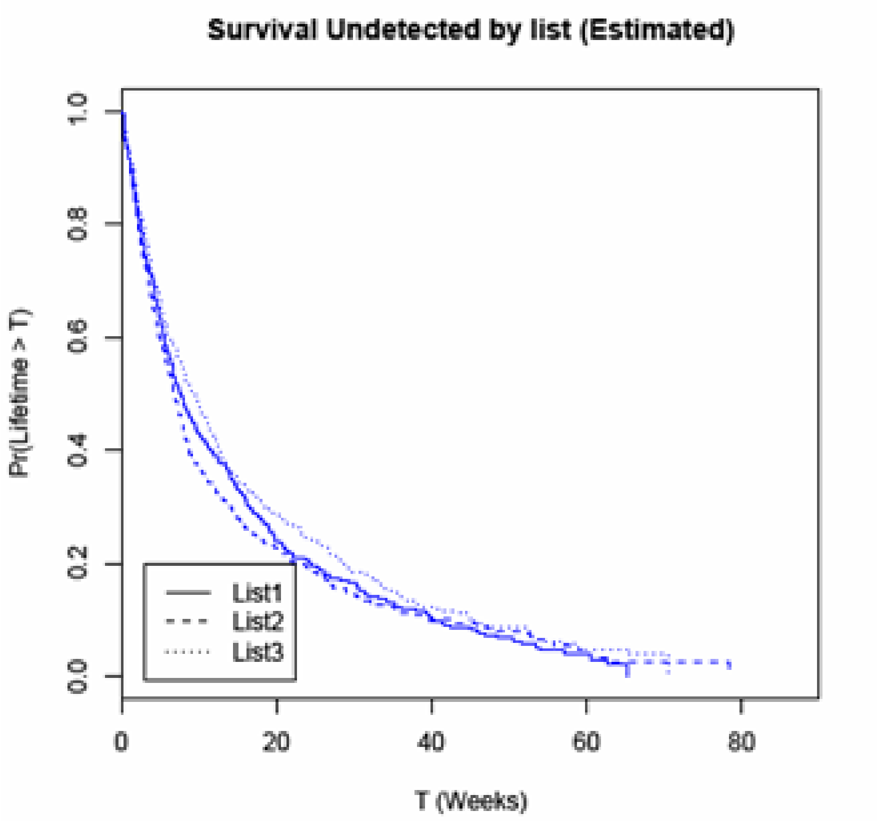
1	2.24	
2	3.91	
3	3.81	
4	2.43	
5	0.21	← most wily
6	2.51	
7	1.72	
8	2.48	
9	3.13	
10	3.23	
11	2.31	
12	2.89	
13	3.74	
14	3.55	
15	4.00	← most conspicuous



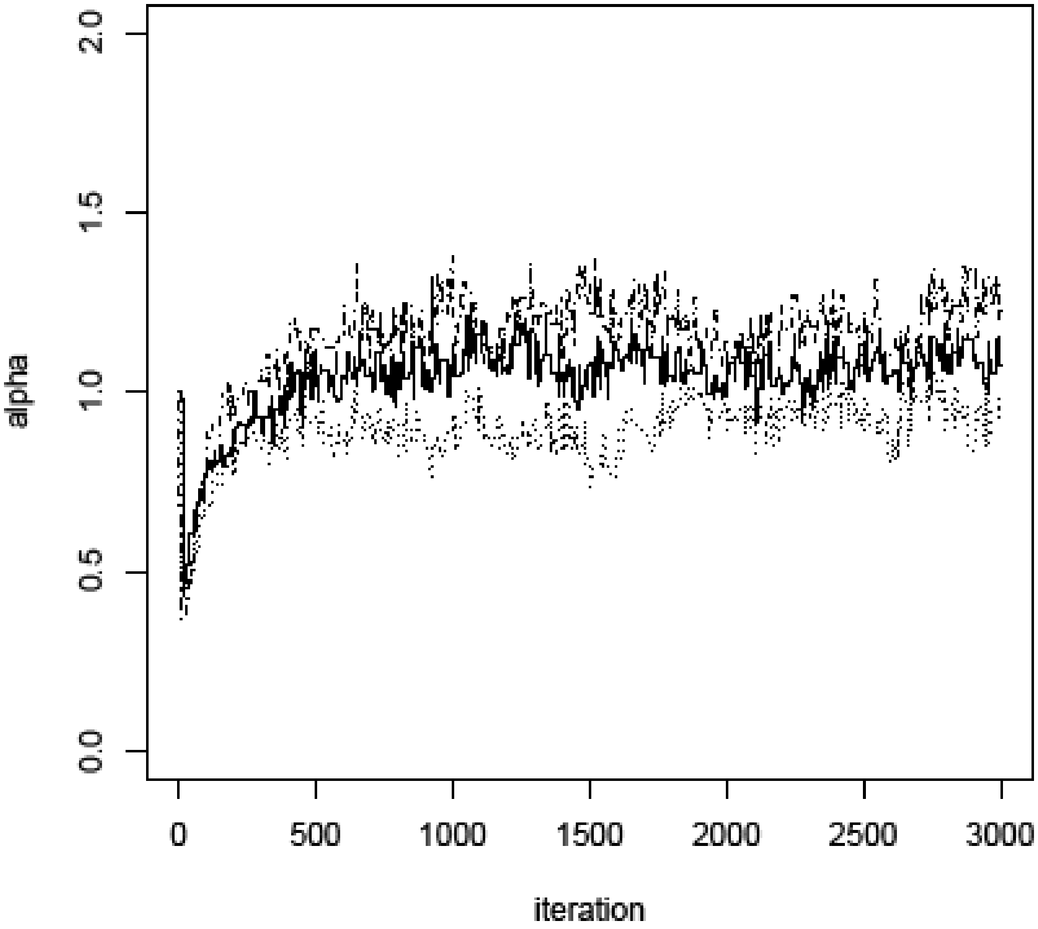
Estimates

Quantity	Point Estimate	Interval
Keylogger Effect	-0.68	(-0.89, -0.57)
Trojan Effect	0.35	(0.24, 0.49)
Difference	1.03	(0.88, 1.28)

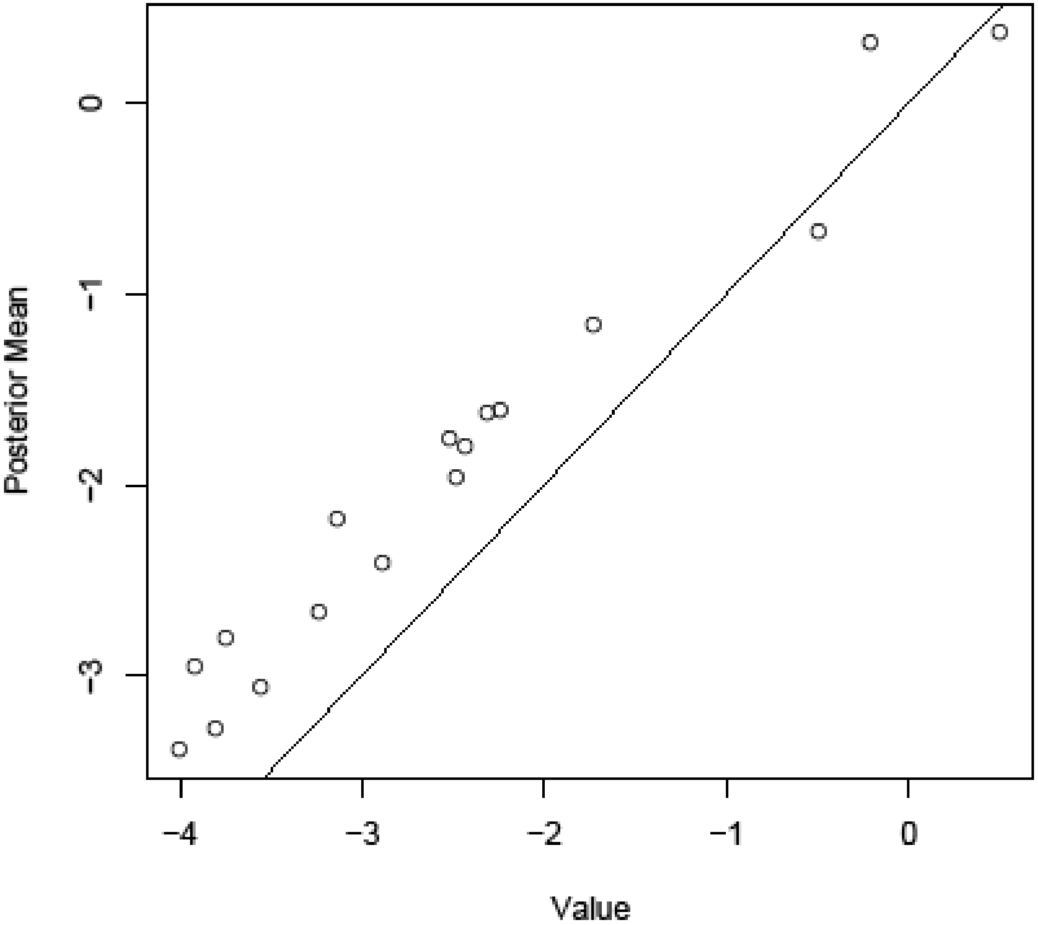




Underestimating Rates



Overestimating Trait effects



Open Questions

Model Identifiability

Effect of Signature publication

What about when births cannot be observed?

Summary

Mark-recapture needs some more “layers” for malicious Internet populations:

- Indirect observation
- Imprecise individuals

Three threats, three different approaches to modeling

- Extensions away from strict capture-recapture toward GLM
- Account for indirect observation with hierarchical models
- Account for imprecise individuals with clustering

How much is measurable?

- Identifiability
- List dependence
- Prior information