Hunting with DNS: Notos and Authoritative Query Mining

Manos Antonakakis

Georgia Institute of Technology
Information Security Center
Damballa, Inc.

DNS-OARC 2010

Overview

What I'll talk about?

- **DNS reputation**
  - How we manage to quantify DNS reputation?
  - Full academic paper is available on line
  - I'll give some insight on the core modules of Notos
  - I'll emphasize on things that make Notos applicable in real world scenarios
- **DNS Authoritative Query Mining**
  - Full academic paper is underway but not currently public
  - What we try to model and why?
  - Measurement results from authoritative DNS data
  - Which the learning steps we plan to use

Credits

Based on joint work with:

- Roberto Perdisci, Wenke Lee
- David Dagon, Nick Feamster

Special thanks to:

- SIE@ISC
  - Passive DNS data
  - Authoritative DNS data
- Sam Norris
- Robert Edmonds
  - Many useful comments

Credits

Based on joint work with:

- Roberto Perdisci, Wenke Lee
- David Dagon, Nick Feamster

Special thanks to:

- SIE@ISC
  - Passive DNS data
  - Authoritative DNS data
- Sam Norris
- Robert Edmonds
  - Many useful comments

Dynamic Reputation For DNS

Problem Description

- Malware families utilize large number of domains for discovering the “up-to-date” C&C address
- IP-based blocking technologies have well known limitation and are very hard to maintain
- DNSBL based technologies cannot keep up with the volume of new domain names used by botnet
  - Examples are Sinowal, Bobax and Conficker bots families which generate thousands on new C&C domains every day
- Detecting such type of **agile botnets** cannot be achieved by the current state of the art detection mechanisms

Problem Description

- Malware families utilize large number of domains for discovering the "up-to-date" C&C address
- IP-based blocking technologies have well known limitation and are very hard to maintain
- DNSBL based technologies cannot keep up with the volume of new domain names used by botnet
  - Examples are Sinowal, Bobax and Conficker bots families which generate thousands on new C&C domains every day
- Detecting such type of **agile botnets** cannot be achieved by the current state of the art detection mechanisms

Problem Description

- Malware families utilize large number of domains for discovering the "up-to-date" C&C address
- IP-based blocking technologies have well known limitation and are very hard to maintain
- DNSBL based technologies cannot keep up with the volume of new domain names used by botnet
  - Examples are Sinowal, Bobax and Conficker bots families which generate thousands on new C&C domains every day
- Detecting such type of **agile botnets** cannot be achieved by the current state of the art detection mechanisms

The Proposed Solution: Notos

- We designed Notos; a dynamic, comprehensive reputation system for DNS
- We constructed network and zone based statistical features that can capture the characteristics of domains
- These features enable Notos to learn the models of legitimate and malicious domains in order to compute reputation scores for new domains
- Notos can correctly classify new domains with a very low FP_{rate} (0.38) and high TP_{rate} (96.8), several days or even weeks before they appear on static blacklists

The Proposed Solution: Notos

- We designed Notos; a dynamic, comprehensive reputation system for DNS
- We constructed network and zone based statistical features that can capture the characteristics of domains
- These features enable Notos to learn the models of legitimate and malicious domains in order to compute reputation scores for new domains
- Notos can correctly classify new domains with a very low FP_{rate} (0.38) and high TP_{rate} (96.8), several days or even weeks before they appear on static blacklists

The Proposed Solution: Notos

- We designed Notos; a dynamic, comprehensive reputation system for DNS
- We constructed network and zone based statistical features that can capture the characteristics of domains
- These features enable Notos to learn the models of legitimate and malicious domains in order to compute reputation scores for new domains
- Notos can correctly classify new domains with a very low FP_{rate} (0.38) and high TP_{rate} (96.8), several days or even weeks before they appear on static blacklists

Notation and Terminology

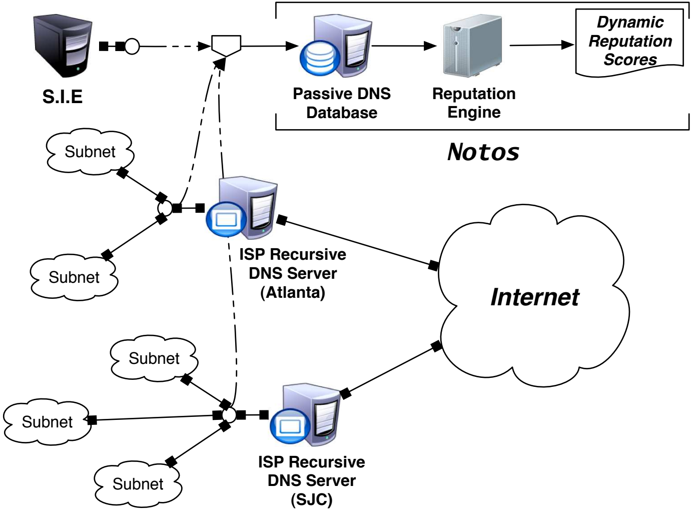
- What is a Resource Record (RR)?
  - `www.example.com 192.0.32.10`
- What is a 2^{nd} level domain (2LD) and 3^{rd} level domain (3LD)?
  - For the domain name `www.example.com`: 2LD is `example.com` and 3LD is `www.example.com`.
- What we define as Related Historic IPs (RHIPs)?
  - All "routable" IPs that historically have been mapped with the domain name in the RR, or any domain name under the 2LD and 3LD
- What we define as Related Historic Domains (RHDNs)?
  - All fully qualified domain names (FQDN) that historically have been linked with the IP in the RR, its corresponding CIDR and AS

Notation and Terminology

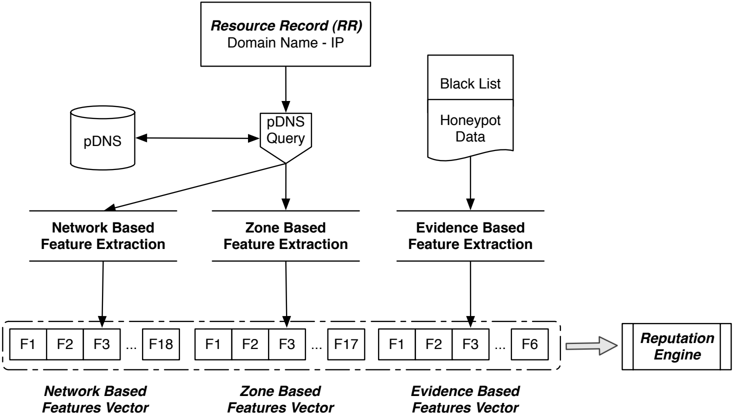
- What is a Resource Record (RR)?
  - `www.example.com 192.0.32.10`
- What is a 2^{nd} level domain (2LD) and 3^{rd} level domain (3LD)?
  - For the domain name `www.example.com`: 2LD is `example.com` and 3LD is `www.example.com`.
- What we define as Related Historic IPs (RHIPs)?
  - All "routable" IPs that historically have been mapped with the domain name in the RR, or any domain name under the 2LD and 3LD
- What we define as Related Historic Domains (RHDNs)?
  - All fully qualified domain names (FQDN) that historically have been linked with the IP in the RR, its corresponding CIDR and AS

Notation and Terminology

- What is a Resource Record (RR)?
  - `www.example.com 192.0.32.10`
- What is a 2*nd* level domain (2LD) and 3*rd* level domain (3LD)?
  - For the domain name `www.example.com`: 2LD is `example.com` and 3LD is `www.example.com`.
- What we define as Related Historic IPs (RHIPs)?
  - All “routable” IPs that historically have been mapped with the domain name in the RR, or any domain name under the 2LD and 3LD
- What we define as Related Historic Domains (RHDNs)?
  - All fully qualified domain names (FQDN) that historically have been linked with the IP in the RR, its corresponding CIDR and AS


Dynamic Reputation Scores
S.I.E
Passive DNS Database
Reputation Engine
Notos
Subnet
Subnet
ISP Recursive DNS Server (Atlanta)
Internet
Subnet
Subnet
Subnet
ISP Recursive DNS Server (SJC)

Three Main Feature Vectors for Notos



Network, Zone and Evidence Vectors

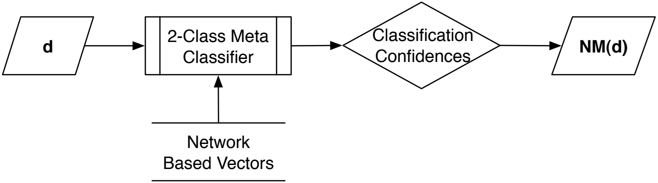
- Vectors for Clustering and Classification
  - **Network Based vector (18)**
    - M/M/STD of frequencies from the set of different networks properties in the list of RHIPs
  - **Zone Based vector (17)**
    - M/M/STD of frequencies from observation based on the zone structure of the domains in the list of RHDNs
- **Evidence vector** (used in the reputation function)
  - Various BLs (3 - IP/CIDR/AS) using public and private IP and DNS BLs
  - Malware Analysis (3 - IP/CIDR/AS) using domain names extracted from malware analysis

Network, Zone and Evidence Vectors

- Vectors for Clustering and Classification
  - **Network Based vector (18)**
    - M/M/STD of frequencies from the set of different networks properties in the list of RHIPs
  - **Zone Based vector (17)**
    - M/M/STD of frequencies from observation based on the zone structure of the domains in the list of RHDNs
- **Evidence vector** (used in the reputation function)
  - Various BLs (3 - IP/CIDR/AS) using public and private IP and DNS BLs
  - Malware Analysis (3 - IP/CIDR/AS) using domain names extracted from malware analysis

Network Profile Modeling

We train a Meta-Classifier based on the 5 anchor-classes.

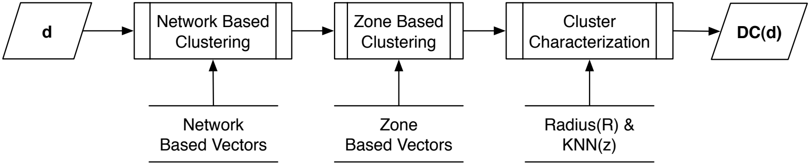


The network feature vector of a domain name *d* will be translated into the network modeling output (**NM(d)**) — the feature vector composed from the confidence scores for each different anchor-class.

The two clustering steps

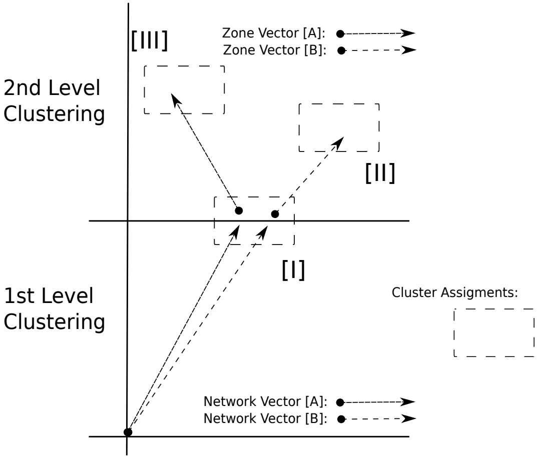
- 1^{st} **Level Clustering (using Network Feature Vectors):** Goal is to identify similarities in zones based upon their network profiles
- 2^{nd} **Level Clustering (using Zone Feature Vectors):** Goal is to further group domain names (within each 1^{st} level cluster) based upon their zone properties

Domain Clustering Flow



In this step we are able to **characterize** unknown domains within clusters based upon already labeled domains in close proximity. The **DC(d)** will assemble a 5 feature vector **characterizing the position of** d **in the** 2^{nd} **level sub-cluster**

Quick Note on the 2^{nd} Clustering Step



2^{nd} Level Clustering Split Due to Zone Properties

[A]: ns6.b0e.ru 218.75.144.6

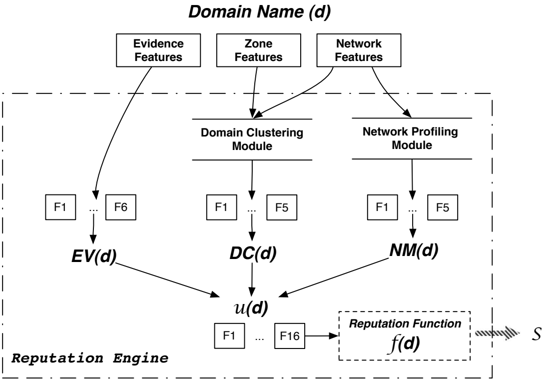
```
...
 188.240.164.122.dalfihom.cn  218.75.144.6
 0743f9.tvafifid.cn           218.75.144.6
 ns5.bg8.ru                   218.75.144.6
 097.groxedor.cn              218.75.144.6
 adelaide.zegsukip.cn         218.75.144.6
 07d2c.fpibucob.cn            218.75.144.6
 0c9.xyowijam.cn              218.75.144.6
 ns6.b0e.ru                   218.75.144.6
 0678fc.yxbocws.cn            218.75.144.6
 ns1.loverspillscalm.com      218.75.144.6
 09071.tjqsjfz.cn             218.75.144.6
 0de1f.wqutoyih.cn            218.75.144.6
 katnzvv.cn                   218.75.144.6
...
```

[B]: e752.p.akamaiedge.net 72.247.179.52

```
...
 e882.p.akamaiedge.net  72.247.179.182
 e707.g.akamaiedge.net  72.247.179.7
 e867.g.akamaiedge.net  72.247.179.167
 e747.p.akamaiedge.net  72.247.179.47
 e732.g.akamaiedge.net  72.247.179.32
 e932.g.akamaiedge.net  72.247.179.232
 e752.p.akamaiedge.net  72.247.179.52
 e729.g.akamaiedge.net  72.247.179.29
 e918.p.akamaiedge.net  72.247.179.218
 e831.p.akamaiedge.net  72.247.179.131
 e731.p.akamaiedge.net  72.247.179.31
...
```

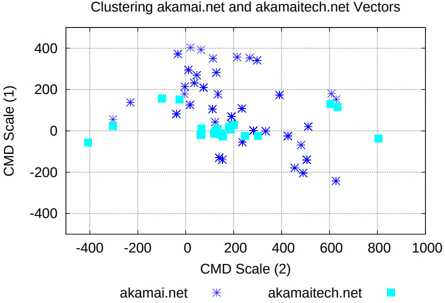
Reputation Function

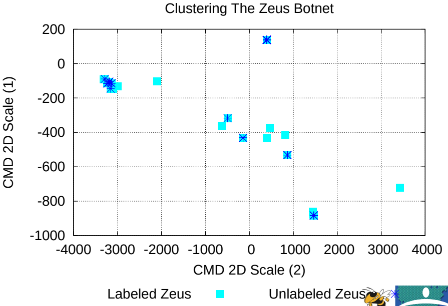
Each domain d will be transformed into 3 vectors $NM(d)$, $DC(d)$ and $EV(d)$ (or evidence vector) that is the final reputation vector $v(d)$.



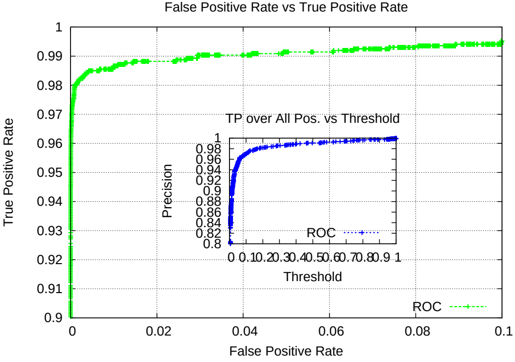
Akamaitech (unknown) VS Akamai (in knowledge base) domains

Clustering known with unknown domain names from Zeus botnet



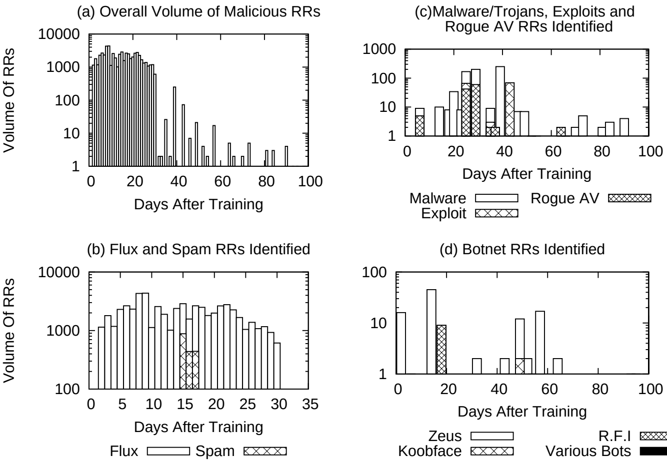


Results from the reputation function



- Results for 10-fold cross-validation, and detection threshold at 0.5, using different Alexa based White-lists:
  - (Top 500) FP_{rate} = 0.38 and TP_{rate} = 96.8 (ROC)
  - (Top 10K) FP_{rate} = 0.4 and TP_{rate} = 93.6
  - (Top 100K) FP_{rate} = 0.6 and TP_{rate} = 80.6

Early domain detections using Notos



Building an anomaly detection model
for authoritative DNS query traffic

Why should we care?

How do we currently discover domain names used for evil?

- Malware analysis: scaling problems
- DNS registration information: known bad users, stolen cards etc.
- Passive DNS — Notos-like dynamic reputation system: Passive DNS takes away key signal of the resolution plain

What we need next? Early warning system based on authority data

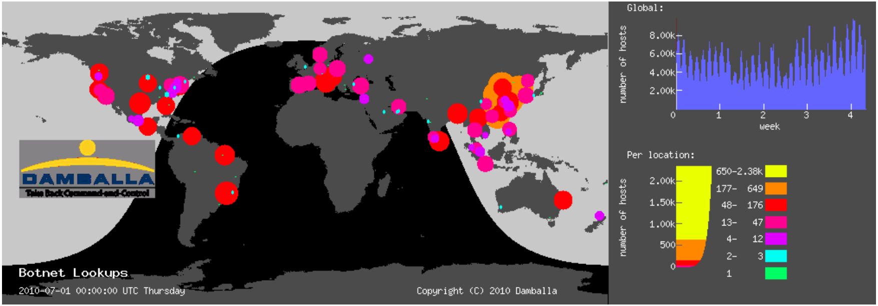
- What we collect and how we can scale?
- What we can reliable measure — and model?

Key: if you cannot reliable measure the traffic you cannot model any signal within it

A real world example please?

- Could there be a botnet out there that is fully active and nobody knows about?
- IMDDOS Botent:
  - Active since April 2010
  - Detected late July while evaluating diversity clustering results
  - Peak of traffic on July with more that 25K infected hosts
  - Sinkhole stats showed that there were 12 different malware agents we had no information (MD5 samples) about

IMDDOS in action

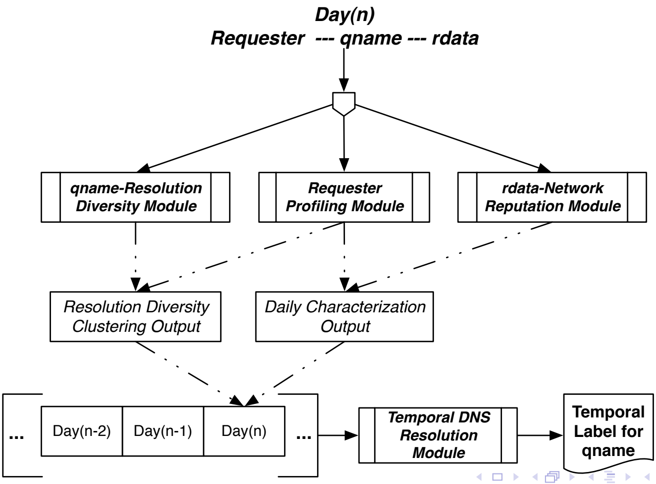


Our ongoing work

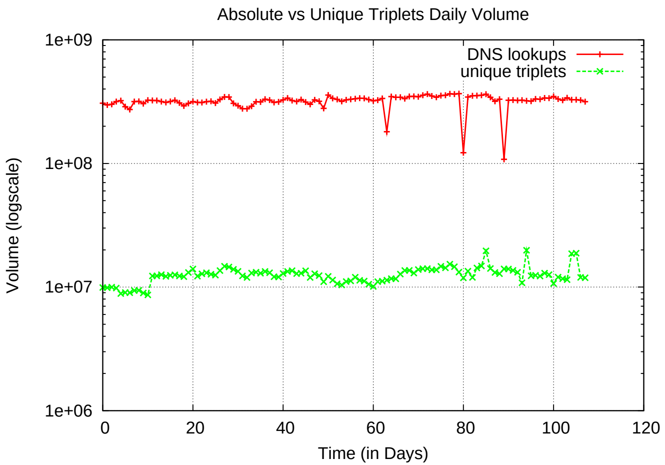
How we can model authoritative DNS query traffic?

- You cannot keep up with absolute DNS lookup traffic: need for data abstraction
- Our technique is based on daily collected triplets: `requester -- qname -- rdata`
- Our modeling efforts are threefold:
  - Model the requester diversity (`unsupervised`)
  - Model the rdata information based on CIDR/AS reputation (`supervised`)
  - Model the last two events over time (`time series analysis`)

System overview

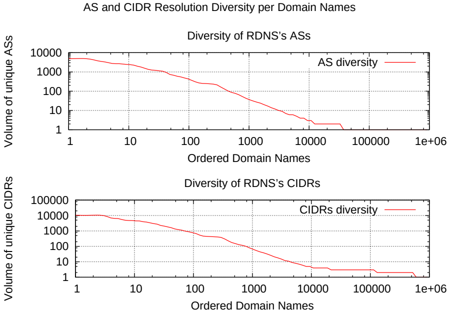


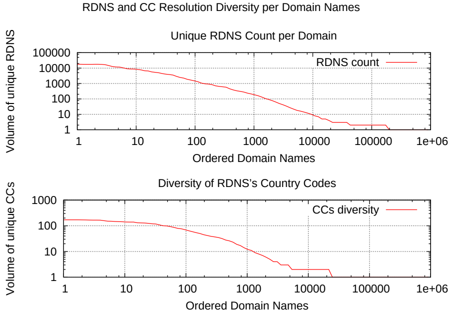
Absolute VS triplet observation volume



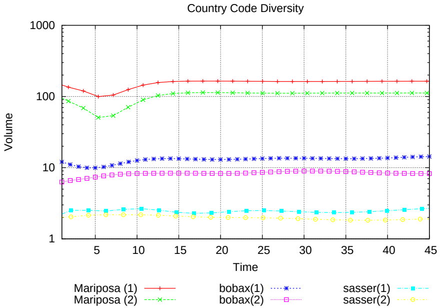
AS and CIDR diversities

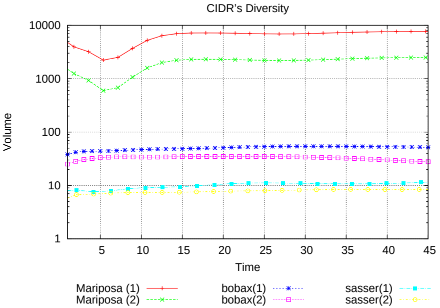
CC and RDNSs diversities





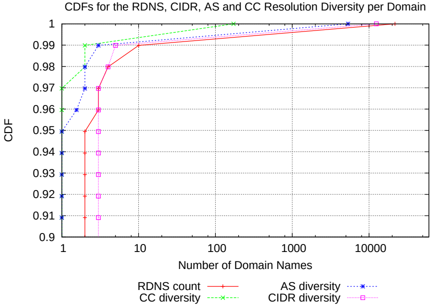
Why modeling the requesters is so important?

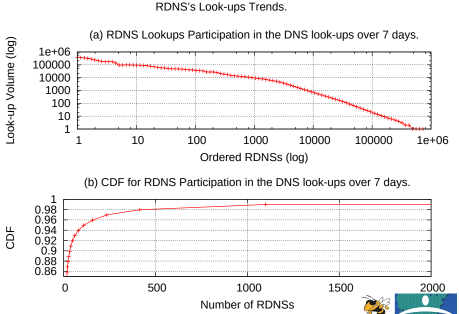




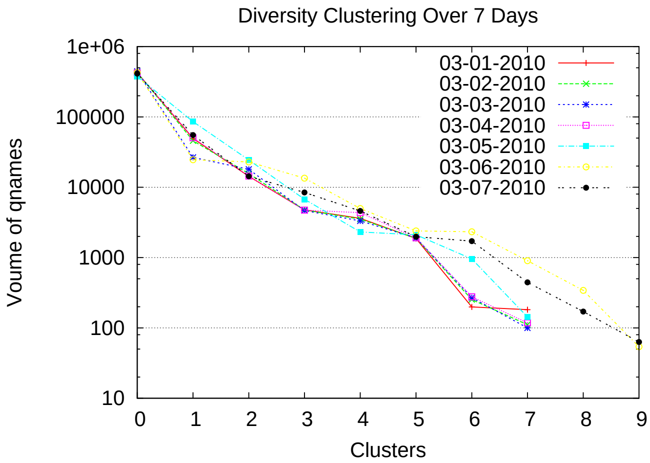
CDF for all diversities

Measuring the requester lookup volume





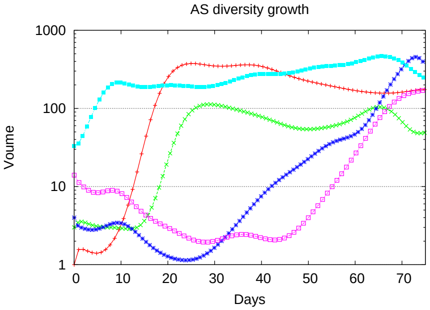
Unsupervised learning — weekly clustering

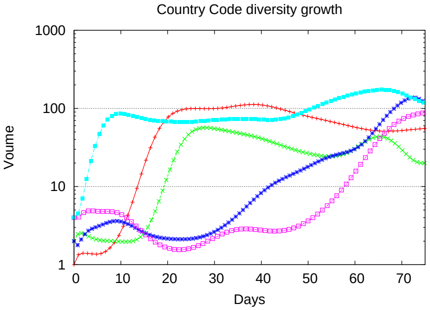


CIDR and AS reputation

- A supervised learning “daily” step
- It characterizes qnames every day
- For TLD traffic we passively reconstruct the rdata
- We try to characterize rdata based on BL data, malware, pDNS, SBL and DNSWL
- Basic balding block for our time series analysis

Time series analysis: threat trends





Conclusions

- We successfully used supervised and unsupervised learning on pDNS to quantify a "dynamic reputation system for DNS"
- We try to do the same for authoritative DNS query data from various different authorities
- The preliminary results are processing — IMDDOS
- The task is significantly harder than handling data at the recursive level
  - The volume of the query traffic mandates abstraction
  - Hard to establish ground truth for sequential observations
- Always looking for data form other TLDs or large authorities to evaluate our methods and demonstrate operational merit

Conclusions

- We successfully used supervised and unsupervised learning on pDNS to quantify a "dynamic reputation system for DNS"
- We try to do the same for authoritative DNS query data from various different authorities
- The preliminary results are processing — IMDDOS
- The task is significantly harder than handling data at the recursive level
  - The volume of the query traffic mandates abstraction
  - Hard to establish ground truth for sequential observations
- Always looking for data form other TLDs or large authorities to evaluate our methods and demonstrate operational merit

Thanks!