Dynamic IP Reputation from DNS

Manos Antonakakis, Roberto Perdisci, and Wenke Lee
Georgia Institute of Technology
MURI Project Background

• Goal: develop dynamic trust management systems for Internet principals and services
  – E.g., IP addresses, DNS domains/servers, BGP/AS, etc.
  – Avoid connections to/from malicious/fraudulent elements on the Internet

• Progress thus far
  – Help build an infrastructure, SIE, for collecting real-time Internet security information
    • Operational; data sources for dynamic trust management
  – Dynamic IP reputation using DNS data
Overview and Motivation

• Dynamic Domain Name reputation rating using passive DNS (pDNS)
  – Professional DNS hosting differs from non-professional
  – pDNS information is already present in our network
  – Static IP/DNS blacklists have limitations
  – Malicious users tend to reuse their infrastructure

• Contributions:
  – Zone and network based clustering of pDNS
  – A new method of assigning reputation on new RRSETs using limited {White/Grey/Black}-listing
  – A dynamic Domain Name reputation rating system
    • Always maintain fresh reputation knowledge based on pDNS
Passive DNS data

• 28 Sensors from ISPs, Banks and corporate networks
• Off-line analysis is possible due to pDNS data locality
• Computing Clustering and Classification Vectors
  – 15 features for the domain name based vector
  – 16 features for the network based vector
• For Labeling the dataset
  – Damballa botnet intelligent, honey-pot data, spam feeds, zeus tracker, do-not-route lists.
Clustering and Classification Vectors

Gray List

Black List

pDNS

RRSET
Domain Name - IP

pDNS Query

Set of IPs mapped with Domain Name

Set of DNs mapped with IP

Black/Gray IPs and Domains linked with DN/IP/CIDR/AS

F1 F2 F3 ... F16

F1 F2 F3 ... F15

F1 F2 F3 ... F6

Network Vector

DN-String Vector

G/B List Vector
Computing Vectors

• Computing Vectors for Clustering and Classification
  – Network Based vector [16]:
    • M/M/Std(#{IPs,CIDRs,ASNs,CC,RegDate,Owner,size(CIDR)})
  – Domain Based vector [15]:
    • M/M/Std(#{chars,TLDs,2LDs,3LDs,{2,3}-grams,Non-Com})

• Computing Vectors for Cluster Labeling
Dynamic Domain Name Reputation System

1. New RRSET
2. Compute Vector From RRSET
3. Passive DNS DB
4. Cluster Based Rating
5. Semi Supervized Rating
6. Classification Based Rating
7. Reputation Function
8. RRSET Reputation Assignment
9. Update Passive DNS DB
Goal: Group relevant, from the network behavior and DNS characteristics point of view, domain names in the same cluster.
Cluster based Rating: Details

- **1st Level Clustering (Network Vectors):**
  - Identify similarities in zones based solely in their network characteristics

- **2nd Level Clustering (Network and Domain Vectors):**
  - Further group vectors in each cluster to have domain name and network correlation
  - Why the network vectors are not good enough? Is it necessary to use a larger vector?
    - Yes, that is the ideal way to cluster RRsets with similar network and domain name characteristics.
There is some separation between the ideal clusters but the combination of most features are still too confused.
Using both vectors we can see that the cluster separation is more natural even between 2 features. The combination of all features gives us a better overall sub-cluster separation.
Take-away From Clustering

- It is very expensive and too noise to use both vectors in the 1\textsuperscript{st} level clustering
- Using only the network vector in the 1\textsuperscript{st} level cluster you get the initial domain name separation
- Finer Grain Analysis: Using both vectors in the 2\textsuperscript{nd} level clustering you gives us better sub-clusters with less distortion between “similar” RRsets
Goal: Utilize existing knowledge for special classes of domain names in order to increase confidence in the identification of RRSEst from these classes.

In other words, professional DNS hosting (i.e. legitimate, popular zones) should exhibit different network behavior than promiscuous DNS hosting.
Classification Based Rating: Details

- 2-classes: Very popular domains
  - pop: google, yahoo, amazon, ebay, facebook, msn
  - The rest top 100 Alexa zones labeled as “common”
- 2-classes: CDNs
  - Akamai
  - Limelight, coralcdn, cloudfront.com, footprint.net
- 1-class: Dynamic DNS:
  - DynDNS, no-ip

NOTE: We don’t try to identify all benign traffic; rather we measure the network properties for a given zone and build a reputation for it
Dynamic DNS Reputation metric

- The Meta Classification step will feed values (Label \([i]\), Confidence \([i]\)) for each vector.
- The clustering step will provide the average Euclidean distances from the \(k\) closest labeled vectors (Gray & Black).
- Final reputation score: Still *work-in-progress*
  - A neural network will “learn” in \((i+2/2)+1\) steps the reputation rating function from returned values of the supervised and unsupervised process and the labeled data.
  - Overall results … soon.
  - Per process results follows.
• The Confusion Matrix
  – Remind: Our goal is not assign labels to vectors based on information that we can easily collect
  – The label we used:
    • dynamic (noip, dyndns), akamai (akamai, akadns), pop (google, amazon, ebay, yahoo, msn), common[ !(pop) & in top 100 alexa.com domains] and CDN (limelight, footprint, cloudfront, coralcdn)

<table>
<thead>
<tr>
<th></th>
<th>dynamic</th>
<th>pop</th>
<th>common</th>
<th>akamai</th>
<th>CDN</th>
</tr>
</thead>
<tbody>
<tr>
<td>dynamic</td>
<td>933</td>
<td>3</td>
<td>3</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>pop</td>
<td>4</td>
<td>4969</td>
<td>17</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>common</td>
<td>2</td>
<td>77</td>
<td>2361</td>
<td>0</td>
<td>5</td>
</tr>
<tr>
<td>akamai</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1851</td>
<td>0</td>
</tr>
<tr>
<td>CDN</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>5000</td>
</tr>
</tbody>
</table>
Evaluating the Clustering process

• **1\textsuperscript{st} Level Clustering:**
  – Goal: get a preliminary separation between vectors based on network properties
  – We get many clusters:
    • Benign (0,3)
    • Malicious (6,17,15)
    • and mixed (i.e.14,7)

• **2\textsuperscript{nd} Level Clustering:**
  – Need for finer grain analysis. How cluster 14 would looks like after this step?
Intuition: The 2nd level clustering process is capable in many cases to differentiate the known benign and professionally operated zones from the rest, by using the combined network and domain name vector.
Conclusion and Future Work

• What we’ve learned
  – pDNS contain an interesting information signal
  – We identify the features that can harvest this signal from the pDNS DB
  – Classification works great & Clustering needs more tuning

• What’s the next step
  – Benchmark the reputation function
  – Utilize information from the zone authority (ANS) to assist in better RRset inter-cluster association
Beyond the Immediate Next Step

• Incentivize “good behaviors” from networks
  – E.g., do not host bad domains just for the money
  – If trust dynamic trust score of IP or Domain depends heavily on the trust score of the network service provider, the provider could lose legitimate domains if it hosts a few number of bad domains

• Ultimate goal:
  – An on-line dynamic trust/reputation service for IP/Domain
Credits and Acknowledgment

• Georgia Tech
  – David Dagon, Nick Feamster
• Damballa
  – Gunter Ollmann