P2P Botnet Detection through Malicious Fast Flux Network Identification

David Zhao
Department of Electrical and Computer Engineering
University of Victoria
Victoria, BC, Canada
davidzhao@ieee.org

Issa Traore
Department of Electrical and Computer Engineering
University of Victoria
Victoria, BC, Canada
itraore@ece.uvic.ca

***Abstract*— A recent development in botnet technology is the adoption of P2P architecture as way to improve botnet resilience to disruption compared to the centralized architecture used by early botnets. Furthermore, in order to increase stealth and evade detection, many P2P botnets, such as storm, are employing fast flux service networks (FFSNs). We propose in this paper, a new P2P botnet detection approach by identifying malicious FFSNs. We define and compute a number of metrics from captured network flows which are analyzed using machine learning classification. For the proposed approach, we show experimentally that the presence of botnets may be detected with a high accuracy and identify its potential limitations.**

P2P Botnet Detection; Traffic Behavior Analysis; Network Flows; Fast Flux Networks;

I. INTRODUCTION

Botnets have been used in a variety of cybercrimes from click-fraud to DDOS attacks to the generation of spam, and represent currently one of the most predominant threats on the Internet. While early botnets used a centralized architecture based, for instance, on the Internet Relay Chat (IRC) or the HTTP protocols, which can easily be disrupted, a more recent development in botnet command & control (C&C) technology utilizes peer to peer (P2P) networks and protocols to form the communications network for bots. In P2P schemes, individual bots act as both client and server, producing a network architecture without a centralized point which may be incapacitated. The network is resilient in that when nodes are taken offline, these gaps may be closed automatically, allowing for the network to continue to operate under the attacker's control.

Increasingly, attempts are being made by network administrators to shutdown or disable botnets by removing or disabling C&C servers or individual nodes from the malicious network. Such shutdown attempts operate on the concept of denying availability to the malicious network. If servers associated with the malicious domains may be disabled or compromised, the reduction in availability could stop or severely restrict the potential for a botnet to inflict harm.

In response to these new anti-botnet tactics, botnets have responded by employing fast flux service networks (FFSNs). The goal of fast-flux is to improve availability through a strategy of associating a fully qualified domain name with hundreds or even thousands of individual IP addresses [5]. The domain swaps these IP addresses in extremely short time intervals through round robin or similar scheduling combined with a very short Time To Live (TTL) on the DNS record. In addition, the botmaster may employ load balancing mechanisms and other health checks to ensure that only healthy and available nodes are presented by the domain. Fast flux networks typically create a level of indirection for security and to increase stealth. The front end nodes are typically only redirectors to backend servers which actually serves requests. When some query is made to a malicious domain, the redirectors forward the request to the fast flux 'mothership' which then returns the actual response.

There are two primary categories of fast flux networks. 'Single Flux' networks are the more basic form of fast flux. In the single flux case, the DNS record for a malicious domain may rotate its front end node's IP address as often as once every three minutes. In this way, even if a redirector is shut down, others standing by can quickly be fluxed into the DNS record to maintain availability.

Double flux networks provide additional redundancy by not only fluxing the DNS A records, but also the authoritative name server (NS) records. In the double flux scheme, the authoritative name server for a fast flux domain is itself part of the fast flux network.

One challenge in detecting fast flux networks comes from the difficulty in distinguishing between malicious fast flux and benign fast flux networks. Benign domains may exhibit fast flux characteristics due to the use of content distribution networks (CDNs) which employ it as a technique to improve service availability for high traffic websites.

We examine the network behavior of a botnet at the level of the TCP/UDP flow, and generate a set of metrics for measuring fast flux networks and for determining whether a domain is a malicious fast flux network or part of a benign network. For final decision making, we use machine learning classification based on a decision tree classifier using reduced error pruning (REPTree).

Additionally, by polling a fast flux domain continuously over a period of a week, we generate a geographical distribution of its front facing nodes in order to gain a picture of the physical extent of the system.

The rest of the paper is structured as follows. Section 2 summarizes and discusses related work on fast flux network detection. Section 3 describes general characteristics of fast flux network model. Section 4 introduces our detection model for malicious fast flux network. Section 5 presents the

experimental evaluation of our proposed detection model. Section 6 makes some concluding remarks.

II. Related Work

Fast flux techniques have been used by benign networks for many years as a way of distributing load for popular websites. Content management networks such as Akamai use techniques such as short TTL on DNS records, rotating IPs that are characteristics of fast flux behaviour. While a significant amount of research has been accomplished on botnet detection [1, 2, 3, 4, 6, 7, 8, 9], to our knowledge, only a limited number of proposals have been made on detecting botnet through malicious fast flux networks identification.

The usage of fast flux techniques on malicious networks have only become popular in the last few years, and have only come to wide attention since 2007 when the honeynet project released a report of their behaviour [5,10]. Since then, some researchers have explored ways to detect such networks and distinguish them from non-malicious fast flux networks [14, 15, 16, 17].

Nazario and Holz presented a way of identifying and qualifying a domain as fast flux using a series of rule based heuristics. They characterized malicious fast flux networks based on a series of metrics including TTL, the existence of more than 5 unique IP's in a single A record query, and the average distance between IP addresses within the queries. They marked a domain as 'fluxy' when it exhibited at least four of the nine metrics. While they did not directly measure the accuracy of their technique, they did discover some 900 qualified domains that may be qualified as fast flux using the system.

Caglayan and others developed a detection technique based on measuring the TTL and two other metrics known as the fast flux activity index and footprint index. The fast flux activity index measures the degree of DNS change for a particular domain, while the footprint index measures the geographical dispersion of IP addresses associated with a single domain [11]. These data points are fed into a Bayesian classifier to determine with high accuracy the probability that a domain is malicious fast flux.

Passerini et al. used a similar approach in their fast flux detection system called FluXor. They use the age of the domain, the number of distinct DNS A records, the time to live, and other metrics to distinguish between malicious fast flux networks with non-malicious networks. Ultimately, a naïve Bayesian network detector is used to make the final detection, resulting in a detection accuracy of over 90% on fast flux networks.

Our work builds upon this work and use flow metrics to create a decision tree based approach in detecting fast flux networks quickly as they exhibit fluxing behavior.

III. Fast Flux Network Model

The detection of FFSNs is motivated by the increasing prevalence of fast flux techniques within the botnet ecosystem. As malware authors explore new avenues to exploit their botnets, the concept of botnet based hosting has become an increasingly popular way for botnet authors to earn an income. The challenge facing these botnet based hosting services is similar to those of typical legitimate websites, namely, availability. Typical uses for botnet hosted sites include malware distribution, advertisement, spam and phishing attacks, and so aside from the normal challenges faced by a typical website, botnet hosted content are also frequently illegal in nature and so faces the risk of being actively disabled by enforcement agencies [12].

Malware authors therefore turn to fast flux networks, a strategy first employed by legitimate sites, in order to maintain availability in response to these pressures. Because fast flux behaviour exists for both malicious and non-malicious websites, it is therefore critical for any detection strategy to be able to distinguish between the two. Such a detector must be aware of the subtle but critical differences which distinguish the two types, as well as the inherent properties of a malicious fast flux network in order to avoid overlooking a potential malicious domain.

The Honeynet Project [5] identified two types of fast flux networks, single flux and double flux, where single flux consists of the fluxing of DNS 'A' records while double flux also fluxes the NS records for a domain. In either case, the average rate of change for the returned records associated with a fast flux domain is approximately 3 to 10 minutes, and in the double flux case a series of between five to ten A records and five NS records are returned in a round robin fashion. In a traditional phishing example, the shutting down of a single IP address will result in the shutdown of the hosting machine and therefore the scam site itself, but in the fast flux case, the shutdown of a single IP address correlates only to the shutdown of one of thousands of potential zombies which serve content redirected from a 'mothership', and is ineffective at halting the operation. In double flux, even if the DNS nameservers are in reality proxies for a command and control system which is protected behind the scenes, its content easily replicate to other IP addresses that have been infected.

All fast flux networks share certain characteristics which may be used to identify them, but the existence of non-malicious networks which share these characteristics complicates the problem of detection.

a) Low Time-to-live

One of the defining characteristics of all fast flux networks is a low TTL value on the A records. The TTL value in DNS specifies how long a nameserver should cache a particular record before it should query the authoritative nameserver to refresh the record. If a stub resolver queries the nameserver for a record before the cache is expired, the server simply returns the cached value with no change. A low time to live is therefore important for a fast flux network to ensure that any queries made to a particular domain always quickly reaches the latest and highest available nodes. Traditional web sites tend to exhibit a very long time to live (a common value is 24 hours, or 86400 seconds), but in a fast flux network this value may be as low as 300 seconds.

Low time to live value almost always identifies a network as fast flux, but due to the increasing load on popular websites today, the introduction of CDNs and fast

flux techniques have made it a metric that will typically catch malicious and non-malicious sites alike. Some popular non-malicious websites with very low TTL can be seen in Table I.

Table I: TTL of Popular Domains in seconds

Domain	TTL(s)
Google.com	300
Amazon.com	900
Microsoft.com	3600
Ebay.com	3600

b) Number of unique A records

While both malicious and non-malicious domains may exhibit a low TTL value, the IP addresses returned in a DNS query from a malicious domain can differ from a non-malicious domain in certain ways. One such difference is in the number of unique A records returned over time. Popular websites using content distribution networks typically have a small set of IP addresses serving some geographical area, with each IP corresponding to a powerful backend server capable of servicing many requests. On the other hand, the A records returned from malicious domains tend to be weaker, zombie machines of which there are tens of thousands. A DNS query made to a malicious fast flux domain therefore is likely to discover many more unique IP addresses over time, and this metric may be a powerful indicator of malicious activity.

c) IP Networks

In many non-malicious domains, the set of IP addresses returned correlates strongly with each other as they are often part of the same network clustered around a specific geographical location while malicious fast flux networks do not share this quality. In malicious fast flux, the geographic distribution of nodes tend to be more widespread and individual machines are not closely associated with each other as they tend to be personal computers on individual networks. IP address distance measures whether IP addresses returned from a DNS query are part of the same network or sub-network with the expectation that non-malicious networks will contain addresses belonging to a small number of distinct networks.

d) IP Geolocation

IP addresses in a non-malicious fast flux domain will tend to be geographically clustered to provide for the best service, this is not likely to be true for malicious fast flux domains as the geographical distribution depends on the locations of the infected machines.

e) Change consistency

Observations of malicious fast flux service networks in the wild indicates that they may lie 'dormant' for long periods of time in which no fast flux behaviour is exhibited. In these cases, fast flux behaviour is only observed when the network is used to perform some active malicious task, and these tasks may last a few weeks [12]. This behaviour is not typically observed in non-malicious fast flux networks which rotate its IP address set on a more consistent basis.

f) Domain lifetime

We may obtain additional information on a domain by querying its whois record for the lifetime of the domain name. Malicious domains typically do not last very long as they are frequently taken down by legal authorities or in other cases by the malware authors themselves as the domains become blocked on relevant anti-malware lists. On the other hand, non-malicious domains are generally much longer lasting, existing for years after creation. By comparing the time of domain registration in whois records, we can use this value as a discriminator in our detection algorithms.

IV. Detection Model

The generation of attributes for our fast flux detection framework is motivated by the characteristics identified in the previous section. Because malicious fast flux networks may have their fluxing behaviour turned on or off by the malware author depending on the situation, we cannot guarantee that monitoring a domain in a short time interval will fully capture fast flux behaviour. Because of this fact, our detector should be conscious of which domains should be subject to further monitoring, and which should be passed over for more suitable targets for monitoring.

In order to provide this behaviour, we develop our detection system with two phases. In the first phase a fast rule based detector observes several attributes of the domain and makes a decision on whether further monitoring is needed. If the characteristics of the domain do not suggest malicious fast flux behaviour, the domain is not scheduled for extended monitoring. If, on the other hand, the initial rule based detector finds signs that a domain does exhibit characteristics of malicious fast flux, it is monitored extensively over a long period of several days in order to more closely observe its behaviour.

A. Filtering stage

In order to determine if a domain should be monitored for an extended period, we must first discover if it is immediately obvious that the domain is fast flux or not. To do this, we first run our attribute collection framework on the domain and collect information for a period of 1 week. If our confidence of fast flux during this monitoring period is low, or if we simply did not detect any fast flux behavior, we use the domain creation date as a judge of whether the domain is worth further monitoring. Domains created a year ago or older are discarded, while domains created within the past year are flagged for further monitoring. The monitoring phase serves to eliminate obvious non-malicious domains from our detector list so we can utilize these resources to monitor more likely targets.

B. Data capture stage

Once a set of domains to monitor have been finalized, the detector moves onto the data capture stage. During this stage, a DNS record is polled continuously at a rate of ½ it's A record's TTL, and the responses captured. The valuable datasets captured during this process include the A and NS recorded returned and their individual TTL's. The data captured through this process are then transformed into a set

of attributes which we may feed into our classification model. The attributes we generate from this dataset consists of the following:

1) Mean Reported TTL (MTTL)

It is possible that during the polling period we witness a change in TTL value. This metric computes the average of the TTL values returned over the polling period. If the TTL does not change, MTTL = TTL. We use this attribute to capture the 'low TTL' behavior of fast flux networks.

2) Actual Mean TTL (ATTL)

There have been observations that in several malicious fast flux networks, the actual rate of flux is even lower than the given TTL value for a DNS record [11]. Typically, non-malicious networks respects the TTL values and do not deviate heavily from it. The ATTL attribute records the mean time of actual change of IP addresses for a given DNS record. As we are limited in resolution by our polling interval, the lowest value ATTL can achieve is half the MTTL. Difference between ATTL and MTTL may suggest malicious behavior, and a low ATTL suggests fast flux behavior.

3) Total Unique A Records (ARCRD)

Motivated by the number of A records characteristic, this attribute is a count of the number of unique A records witnessed during the polling period. A high ARCRD may suggest fast flux behavior, while a very high ARCRD record may suggest malicious behavior.

4) A Record Change Variance (ARCRDV)

This attribute captures the variance in the number of A record changes witnessed over the polling period and is motivated by the change inconsistency characteristic of malicious fast flux networks. For most non-malicious networks, we expect the variance to be low, as records will tend to change on a consistent basis, while a malicious domain may exhibit a larger variance as the operators turn the fast flux behavior on and off depending on activity.

5) A Record IP Stability (ARCRDS)

We capture the network differences in the IP addresses returned by measuring its 'stability'. This is done by measuring the difference in the octets for each IP address returned in a record. We expect non-malicious domains to exhibit a smaller difference per record than a malicious domain (most may share higher order bits as they belong to the same network).

6) Domain name confidence (DCONF)

The domain name confidence rating is computed from the age of the domain and its domain registrar. This value ranges from 0 – 1.0, where 0 specifies a low confidence in the domain and 1.0 specifies a high confidence. We use two simple characteristics to compute this value: the age of the domain name, and whether the registrar of the domain name exists on a blacklist. If the age of the domain exceeds 1 year, and the domain does not exist on our blacklist, the value is 1. Otherwise, the value is 0 or 0.5 depending on whether one or none of the criteria mentioned above is fulfilled. The DCONF attribute allows us to attach an additional discriminator on our attributes that can increase or decrease our suspicions regarding a particular domain name.

C. Classification Model

Many machine learning (ML) classification techniques exist which all attempt to cluster and classify data based on attribute sets. In our work, we use decision tree based classifiers, which are a well-known classification technique exhibiting desirably low computational complexity. In a decision tree, interior nodes represent input attributes with edges extending from them which correspond to possible values of the attributes. These edges eventually lead to a leaf node which represents an output variable (in our case, whether a flow is malicious or non-malicious). Classification of an attribute vector simply consists of traversing the path from root to leaf by following edges which correspond to the appropriate values of the attributes in the attribute vector. Decision trees are learned via a partitioning process where the source attribute set is split into subsets based on a value test. This partitioning halts after a user-defined stopping criteria has been reached. For our evaluation, we select a decision tree using the Reduced Error Pruning algorithm (REPTree). This algorithm helps improve the detection accuracy of a decision tree with respect to noisy data, and reduces the size of the tree to decrease the complexity of classification.

V. Experimental Evaluation

A. Dataset

We evaluate the effectiveness of our algorithm at detecting fast flux domains by using a list of malicious domain names provided by the DNS-BH project [13]. The project provides a constantly updated list of over 70,000 domains known to serve malware or known to be associated with malicious activities. Additionally, we use a community created malware domain list containing 80,000 active and inactive malicious domains discovered between 2009 and 2012. This list contains domains from the popular Zeus botnet as well as Waledac bots.

A custom build application is used to monitor a random sampling of these domains over a period of 3 days to collect data and observe for possible fast flux activity. Amongst the domains in our dataset, approximately 200 have been confirmed by a third party to have exhibited fast flux behavior; for fast flux behavior observed on other domains, we manually verify the accuracy of detection.

We also include in our sample data a selection of 300 known non-malicious domains pulled from Alexa's top 500 Internet sites. All sites on this list are high traffic, high demand websites and typically use a variety of techniques for load balancing, including the use of CDN's and fast flux behavior. We use these sites to check for the false positive rates of our detector and fine tune its behavior to address any false positives we may find.

B. Results

Malicious fast flux networks frequently exhibit transient behavior. As our detection algorithm inspects networks for fast flux behavior and then makes a decision on whether that network is malicious, we must first observe fast flux behavior before we can classify the network.

For purposes of evaluation, we cease monitoring a network after 7 days if we noticed no fast flux behavior and our confidence in the network's maliciousness is low. The algorithm is configured to examine roughly 2,000 domains simultaneously, and multiple machines were used, resulting in an evaluation period of roughly 2 months. Because verification of fast flux behavior must be done manually, or by label, we use only a small subset of 500 domains as a test for our algorithm's accuracy. For the remainder, we simply list the results reported by the algorithm without hand verification. 2 gives the detection accuracy for our small sample. All 100% of domains labeled and verified as malicious fast flux exhibited fast flux behavior and were detected as malicious. Table II shows the detection rate of our algorithm on our test dataset.

99.0% of domains labeled non-malicious were detected as fast flux networks and selected as being non-malicious, with 0.5% (1 site) that was mistakenly identified as malicious. The misidentified site (imageshack.us) presented interesting behavior that resembled those of malicious networks, namely, the domain resolved to a large number of IP addresses (over 100), and exhibited very low TTL. In future improvements to the detector, a more discriminatory domain name confidence rating may be used to eliminate such errors. In the case of imageshack.us, the age of the domain registrar and the fact that all IP's belonged to the same network could have been used to increase the confidence rating and assist in it being identified correctly.

Table II: Detection rate of malicious fast flux on sample of 500 domains

	True positive	**False positive**
Malicious	100.0%	0.5%
Non-Malicious	99.5%	0.0%

Table III: Count of domains detected as fast flux

	Count
Malicious fast flux	10,312
Non-malicious fast flux	9870
Non fast flux or dead	69,688

Table III gives the count of fast flux and non-fast flux behavior detected by our algorithm. Of the 80,000 malicious domains tested, 10,312 were found to be fast flux networks or have exhibited some fast flux behavior. The remainder either did not exhibit any fast flux behavior over the evaluation period, or were moved / dead.

C. Geomapping

With the results in *B.*, we may take the resolved IP addresses of the malicious fast flux networks and obtain geolocation data on the IP's. Fig. 1 shows this data plotted on a map of the world, with each pin indicating a malicious IP address that was found to be a part of a malicious fast flux network.

For the networks discovered in our evaluation, most IP addresses originated from the United States, Europe, or China, with a few from other countries such as India and South Africa.

As we would expect, there is very little correlation between the domain of a site and the fast flux agents servicing that domain. Many domains in our list uses the top level domain name for Russia (.ru) but most flux agents servicing these domains were found outside of that country.

VI. CONCLUSION

In this paper we proposed a system for detecting bot activity by identifying malicious fast flux networks. We have demonstrated that malicious and non-malicious fast flux behavior may be classified through a machine learning process with very high accuracy. Some non-malicious sites present some problems for classification as they exhibit some behavior typically seen in malicious networks. We recommend an improvement over our existing system by adding additional confidence discriminators. One discriminator that may be of interest is the homogeneity of IP addresses within a fast flux network. As malicious networks tend to be more distributed, measuring the number of subnets members of a fast flux network belongs to could be a very good way of eliminating false positives.

For our future work, we note that in order to create a framework that is truly robust, we must additionally produce a system which allows for the evolution of classifiers while it is still operating, without requiring an offline training process. Such a system would ideally also be capable of identifying new threats without training on existing datasets.

REFERENCES

[1] Rajab, Moheeb Abu, et al., "A Multifaceted Approach to Understanding the Botnet Phenomenon." New York : ACM, 2006. Proceedings of the 6th ACM SIGCOMM conference on Internet measurement. 1-59593-561-4.

[2] Grizzard, Julian B, et al., "Peer-to-Peer Botnets: Overview and Case Study." Berkeley : USENIX Association, 2007. Proceedings of the first conference on First Workshop on Hot Topics in Understanding Botnets.

[3] Holz, Thorsten, et al., "Measurements and Mitigation of Peer-to-Peer-based Botnets: A Case Study on Storm Worm." Berkeley : USENIX Association, 2008. Proceedings of the 1st Usenix Workshop on Large-Scale Exploits and Emergent Threats.

[4] Faily, Maryam, Shahrestani, Alireza and Ramadass, Sureswaran., "A Survey of Botnet and Botnet Detection." s.l. : Third International Conference on Emerging Security Information, Systems and Technologies, 2009.

[5] The Honeynet Project., "HOW FAST-FLUX SERVICE NETWORKS WORK." *The Honeynet Project*. [Online] August 16, 2008. [Cited: October 10, 2011.] http://www.honeynet.org/node/132.

[6] Gu, Guofei, et al., "BotHunter: Deteting Malware Infection Through IDS-Driven Dialog Correlation." 2007. Proceedings of the 16th USENIX Security Symposium. pp. 167-182.

[7] Gu, Guofei, et al., "BotMiner: clustering analysis of network traffic for protocol- and structure-independent botnet detection." San Jose : Proceedings of the 17th conference on Security symposium, 2008.

[8] Yu, Xiaocong, et al., "Online Botnet Detection Based on Incremental Discrete Fourier Transform." s.l. : Journal of Networks, 2010, Issue 5, Vol. 5.

[9] Livadas, Carl, et al., "Using Machine Learning Techniques to Identify Botnet Traffic." 2006. 2nd IEEE LCN Workshop on Network Security. pp. 967-974.

[10] Jiayan, Wu, et al., "A Comparative Study for Fast-Flux Service Networks Detection." Seoul : IEEE Computer Society, 2010. INC

2010: International Conference on Networked Computing, pp. 346-350.

[11] Caglayan, Alper, et al., "Behavioral Patterns of Fast Flux Service Networks." 2010. Proceedings of the 43rd Hawaii International Conference on System Sciences. pp. 1 - 9. 978-1-4244-5509-6 .

[12] Nazario, Jose and Holz, Thorsten., "As the net Churns: Fast-Flux Botnet Observations." s.l. : IEEE Press, 2008. International Conference on Malicious and Unwanted Software (Malware). pp. 24-31.

[13] [13] Glosser, David., "BlackHole DNS for Malware and Spyware." *DNS-BH - Malware Domain Blocklist.* [Online] November 27, 2007. [Cited: March 27, 2012.] http://www.malwaredomains.com/wordpress/?page_id=6.

[14] Yu, Sheng, Zhou, Shijie and Wang, Sha., "Fast-flux attack network identification based on agent lifespan." 2010. Wireless Communications, Networking and Information Security (WCNIS), 2010 IEEE International Conference on. pp. 658-662. 978-1-4244-5850-9.

[15] Wang, Xiao, et al., "Analyzing the availability of fast-flux based service network under countermeasures." 2010. Communications, Circuits and Systems (ICCCAS), 2010 International Conference on. pp. 259-264. 978-1-4244-8224-5.

[16] Passerini, Emanuele, et al., "FluXOR: Detecting and Monitoring Fast-Flux Service Networks." *Detection of Intrusions and Malware, and Vulnerability Assessment.* s.l. : Springer Berlin / Heidelberg, 2008, Vol. 5137, pp. 186-206.

[17] Caglayan, Alper, et al., "Real-Time Detection of Fast Flux Service Networks." 2009. Conference For Homeland Security, 2009. CATCH '09. Cybersecurity Applications & Technology . pp. 285-292.

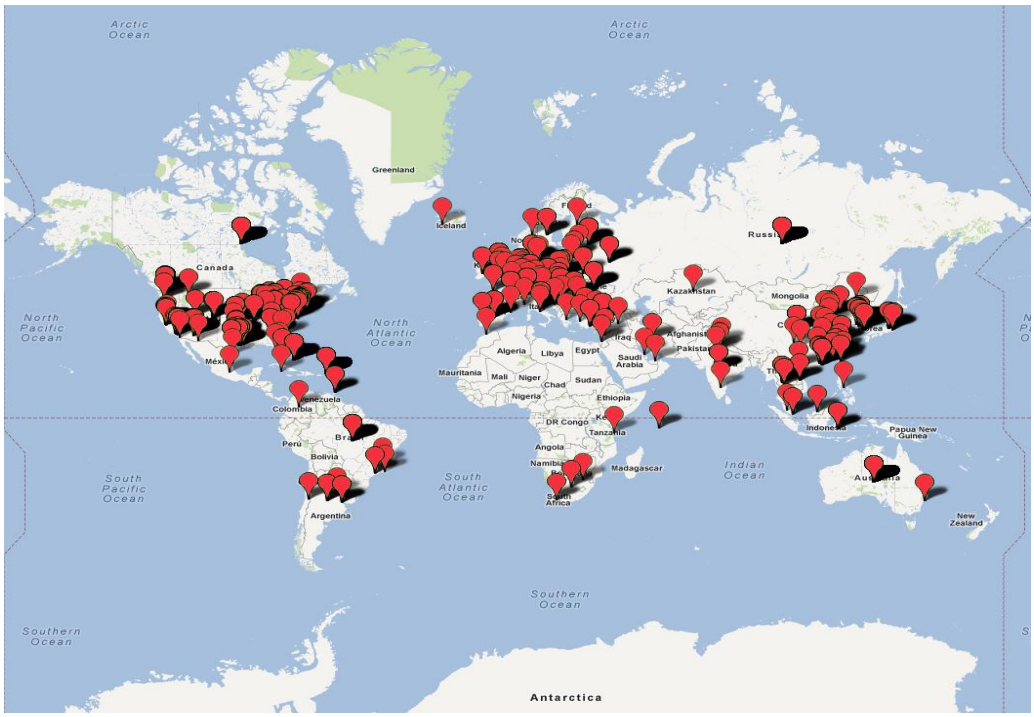

Figure 1. Map of 10,000 malicious fast flux IPs detected during evaluation