CARNet
HRVATSKA AKADEMSKA I ISTRAŽIVAČKA MREŽA
CROATIAN ACADEMIC AND RESEARCH NETWORK

CROFlux
Passive DNS Method for Detecting Fast-Flux Domains

NCERT-PUBDOC-2014-12-344

Nacionalni CERT

ACDC

Contents[1]

Abstract - In this paper we present our approach to fast flux detection called CROFlux that relies on the passive DNS replication method. The presented model can significantly reduce the number of false positive detections, and can detect other suspicious domains that are used for fast flux. This algorithm is used and implemented in Advanced Cyber Defense Centre – a European project co-funded by the European Commission.

[1] Ovaj članak prezentiran je na međunarodnom znanstvenom skupu MIPRO 2014 i dostupan je putem poveznice http://ieeexplore.ieee.org/xpl/articleDetails.jsp?arnumber=6859782

1 Introduction

Nowadays information systems are threatened by various types of attacks. If we look at the most popular threats in information security like distributed denial-of-service, malware distribution and spam, we can notice that they can be conducted using botnets. Botnets are networks of infected computers that are usually managed from a central point also called command and control center (C&C). These networks use various techniques for hiding their presence or amplifying the damage of their attacks. One commonly used technique is fast flux which abuses DNS (Domain Name System). DNS records of flux domains are frequently changed; in such a way that authoritative DNS returns different bots in various time intervals. Flux domains are usually used for hiding botnet command and controls servers, hosting malware delivery sites or for hosting phishing pages. Fast flux facilitates load balancing and proxy redirection that make malicious servers more resistant against detection or takedown attempts.

There are two different types of fast-flux: single-flux and double-flux. Single fast-flux is a technique where multiple bots, within the botnet, are registered and deregistered in a DNS A record for a single fully qualified domain. In combination with round robin DNS algorithm that has very short TTL values (e.g. 3 minutes), this technique produces a constantly changing list of destination IP addresses for the same domain. Resolved bots act as proxies for malware or phishing delivery sites, these sites are popularly called “mothership” nodes.

Double-flux technique, unlike the single flux, provides an additional layer of redundancy. Specifically, in double flux, both the DNS A record and authoritative name server (NS) record are changed. The used authoritative name server administers a fast-flux DNS zone, with all domains and subdomains in it.

A query that has been sent by client’s stub DNS resolver is received by DNS recursor (caching DNS server) which in turn is recursively sent downwards from the root server to the last authoritative name server. The last authoritative name server is a bot used for a double flux scheme. Periodically, other bots take a role of authoritative name servers, thus the corresponding NS records are changed.

After receiving the query the authoritative name server forwards it to the mothership node requesting required information (e.g. domain’s A record). The mothership node sends a response to a flux name server which forwards the response back to the DNS recursor. IP addresses in DNS responses belong to bots. Motherships are second layer C&C servers in a fast-flux botnet scheme, these nodes typically host both DNS and HTTP services to accomplish their single and double-flux scheme role.

2 Related work

Weimer [1] proposed the passive DNS replication (pDNSR) method whose purpose is to extract resource records from response packets of authoritative DNS servers. Using the paired query and response, pDNSR builds a replica of DNS zones that are available from the data. To preserve privacy pDNSR uses only queries between DNS servers. A similar approach was used by Zdrnja and his colleagues [2]; they deployed a custom solution for collecting domains. Analyzed data contained: misspelled domains, fast flux domains and spam related domains.
Holz et al [3] proposed a Flux score to differentiate malicious domains from benign ones. Flux score uses features derived from active DNS queries. These features are: number of unique A records in DNS lookups, number of name servers in lookups and number of unique autonomous numbers (ASN) from type A records. Their system sends several active queries to collect relevant data about domains. After the first query system waits for TTL to expire, and then it sends another query. Using popular spam domains they obtained an optimal hyper plane that separates flux domains from benign ones.
Fast flux domains are classified using Flux score as following:

$$1.32 \cdot \text{numIP} + 18.54 \cdot \text{numASN} + 0.0 \cdot \text{numNS} > 142.38 \quad (1)$$

Where the parameters are:

- *numIPs* – number of distinct IP addresses
- *numASN* – number of unique autonomous system numbers of IP addresses
- *numNS* – number of used name servers for the given domain (ignored because of the zero coefficient in the Equation 1)

Holz et al also realized that Fast-Flux domains have similar characteristics like content delivery networks (CDN) or round robin DNS's that are used to increase website availability. Thus, CDNs are usually classified from the aforementioned algorithm as false positive flux domains. A handful implementation of the algorithm can be found on GitHub [9].
Fluxy [4] uses a similar approach; their authors implemented an adapted equation to compute the Fast-Flux score. In addition to previously mentioned Holz et al method, Fluxy uses reverse DNS queries to detect dynamic IP addresses probably belonging to infected end-users.
EXSPOSURE [5] is a system based on a passive DNS replication. Used data was provided from Security Information Exchange (SIE), which contained response data from authoritative servers from North America and Europe. From this vast amount of data, approximately 4.8 million domains collected through 2.5 months, they extracted time based features of collected domains and analyzed them with a change point detection algorithm CUMSUM. CUMSUM is used to detect short lived domains and domains with repeating patterns. Their classification model was built using a C4.5 decision tree classifier and was trained with features of popular malicious domains. Beside the time related features, they used other groups of features based on: DNS answers, TTL values and domain name characteristics. An extensive list of features can be found in the paper.
Fluxbuster [6] is another system which uses passive DNS replication. Fluxbuster prunes the collected domains based on conservative criteria and outputs candidate domains; pruning procedure is based on data returned in responses (TTL values and diversity of servers' networks). After that, candidate domains are clustered using a hierarchical clustering algorithm. Classification of these domains is based on features collected by the passive analysis. They introduced some novel features like growth ratio of IP addresses and networks. Given clusters are classified using a decision tree classifier. The used training set was obtained using a semi-

manual process, where authors combined known malware domains and manual corrections in labeling.

3 CROFlux

3.1 System Architecture

The developed system CroFlux uses a Farsight's SIE framework [2]for DNS data collection. The framework is similar to the solution proposed by Weimar [1] which reconstructs a partial zone replica preserving users' privacy.
The SIE sensor, as shown in Fig. 1, captures DNS messages from the connected authoritative DNS server. Processing stage returns queried domains and IP addresses from a particular time interval. We can look at those results as aggregated responses to specific domains. SIE outputs the processed DNS messages in NMSG format. NMSG is a wire format optimized for storing and transmitting binaries over UDP [10]. NMSG stores the following information: domain name, DNS query type, name server that sent the query, authoritative name server that answered the query, IP addresses pointing to domain, record class and record type.

3.2 Fast Flux Detection

One limitation of designing a fast flux detection algorithm is the passive nature of DNS data collecting. So the classification process needs to rely on data gathered by completely unpredictable timing of DNS queries sent by various users. This posed a problem, since we need as many as possible resolved IP addresses of fast flux domains which enhance the detection result. We did not use any active DNS data checking like described in [3] and [4], because we wanted to avoid potential exposure to botnet operators. This limitation may become negligible if passive DNS replication method is installed on a big network with many users.
Another problem that may occur is detection of benign domains as flux (false positives). False negatives, i.e. fail to detect fast flux domain, are of a less pertinence. As the result of fast-flux detection algorithm is considered to be a part of public services and actions, false positive results may produce certain inconvenience, distortion of reputation or legal issues. Thus false positive elimination presents a problem of current fast flux detection solutions [8].

CROFlux algorithm runs through three phases. Namely, as shown on Figure 1:
(a) Prefiltering/pruning of collected domains
(b) Candidate domains clustering
(c) Detection of fast flux clusters

a) Prefiltering fast flux candidates

SIE outputs NMSG files that contain captured domains which are later filtered. This filtering stage is similar to the pre-filtering stage proposed by Perdisci [6]. On Figure 1 this step is called domain list pruning procedure.

Selected candidate domains must meet the following requirements:

- Domain's time to live (TTL) should be below 3 hours (TTL<=3600 s)
- Minimum number of resolved IP addresses for the following domain |RS|>= 3, or if RS requirement is not met take only domains with a small TTL<=30 seconds. The number

[2] Farsight SIE framework is available at https://archive.farsightsecurity.com/

of the resolved IP addresses is aggregated from various DNS queries and depends on cache timeout setup on the SIE sensor.

- Diversity of networks for a domain should be div(RS)>0.333. To be a part of the same network, resolved IP addresses must have the same /16 prefix. Diversity is calculated as entropy of /16 networks of resolved IPs for a given domain.

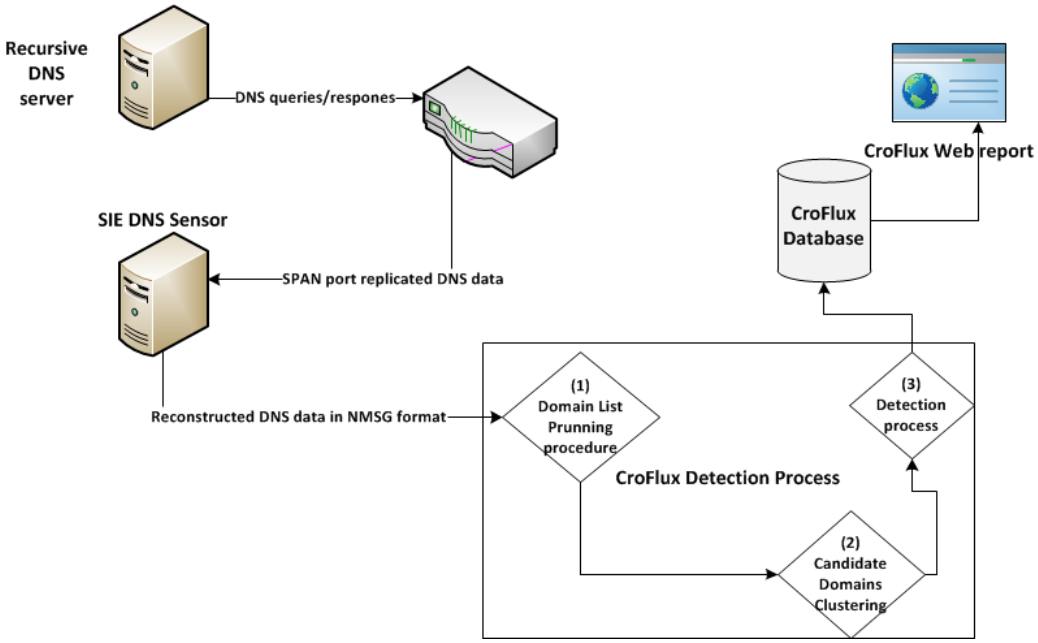

Figure 1. – Overview of CroFlux architecture

After the pruning procedure we have a dictionary with candidate flux domains, where domain names are keys and discovered IP addresses are values. Periodically, measured in time of few minutes, all dictionary records are added to a global dataset. Candidate domains from the global dataset are evaluated from: number of distinct IP addresses, number of distinct AS origins of those addresses and number of estimated dynamic IP addresses of those domains.
Behind flux domains are infected computers or bots which are spread across multiple ISP networks over the world, mostly linked with a broadband connection. The following method is used to mark domain as fast flux candidate [4]:

$$\text{Fluxy Score} = 1.32 * \text{numIPs} + 40.0 * \text{numSPLink} + 20.0 * \text{numASNs} \quad (2)$$

Where the parameters are:

- *numIPs* – number of distinct IP addresses
- *numSPLink* – number of dynamic IP addresses
- *numASNs* – number of distinct autonomous systems

We used a conservative threshold score of 450 for marking domain as a fast flux candidate. For comparison, Fluxy authors and Holz and colleagues used a lower threshold score 142.38 [3] [4].

b) Clustering

Domain clustering is a process in which we group domains that are operated in the same network. As we may expect, hosting providers, content delivery networks (CDN) and botnets that use fast flux are those types of networks. In this step we try to target more closely fast flux domains. Number of common IP addresses is used as clustering criteria. This number of overlapping IP addresses is empirically tuned. Every week clusters are expanded or new ones are created, based on newly collected DNS data**.**

c) Determining fast flux clusters

The final step is to determine fast flux clusters from all candidate clusters. Fast flux domains are often used for malware delivery or hosting phishing domains so we can say that they act as proxies. The way to determine fast flux cluster is to compare candidates' clusters against publicly available and private malware lists collected by internally used tools. Based on the number of malicious domains in cluster we label that cluster as fast flux cluster. Otherwise, cluster is labeled as some other type (CDN, hosting providers etc.). We use a minimum number of malicious domains to classify a candidate cluster as fast flux. This number is tunable and is found empirically. Aforementioned blacklists are available via multiple online services [11] [12] [13] [14] [15] [16] [17]

4 Results

DNS data has been replicated from August 2013 till January 2014, using a CARNet recursor. CroFlux has collected 427.502 de-duplicated second level domains. From these domains we have near 265 malware domains with flux characteristics. In Figure 2 we can see the distribution of top level domains with flux characteristics. Collected flux domains are primarily from Russia (.ru) and Soviet Union (.su) domain which is still used.
In Figure 3 are presented calculated Flux scores of CroFlux's detected flux domains. Scores are shown on logarithmic axes, and as we can see collected flux domains mostly have higher Flux scores. The bright dashed line represents the Fluxy's threshold for flux domains. Also, we can notice that detected flux domains tend to have higher flux scores.
Fluxy score, similarly to Holz score, generates many false positive results [9], i.e. benign domains labeled wrongly as flux. It is also useful to note that given classification hyperplane weights (calculated in [3] and [4]) are trained on elder flux domains - from circa 2008.
Many collected legitimate sites have a high score. They also have similar characteristics to flux domains like: short TTL, many IP addresses scattered through geographically distant autonomous systems etc. Examples of these sites are: content delivery networks like Akamai, cloud services like Amazon AWS, sites used for time synchronization NTP and video streaming services like Netflix and Hulu.

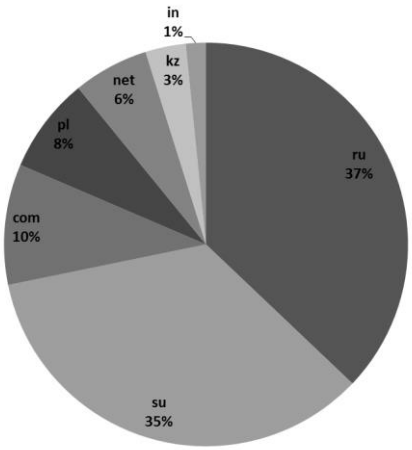

Figure 2 – TLD Distribution of detected flux domains

Similarly, distributed peer to peer networks like Bitcoin use fallback nodes hardcoded in Bitcoin miner applications that use a short TTL (60 seconds) and contain many peer nodes with dynamic IP addresses.
Using our data we recalculated the Fluxy separating hyperplane. Because of the difference between active querying and passive replication we used the last two inserts (of the specific domain) for calculating feature weights.
On a dataset with 9300 benign domains and 100 flux domains we performed a 10-fold cross validation, like in [3]. We obtained feature weights using the Support Vector Machine method with a linear kernel.
This resulted with an optimal hyperplane for flux domains:

$$-0.2853 \cdot \text{numIPs} + 0.4076 \cdot \text{numASNs} + 0.1837 \cdot \text{numSPLink} > 1.5704 \quad (3)$$

Benign domains were classified with a precision of 99% and flux domains with a precision around 91%.
Detected flux domains contained many double flux domains which rotate name servers and IP addresses in responses.
An example of double flux is the following:

```
Ns1.nulled-db.com
Ns2.nulled-db.com
…
NsXX.nulled-db.com
```

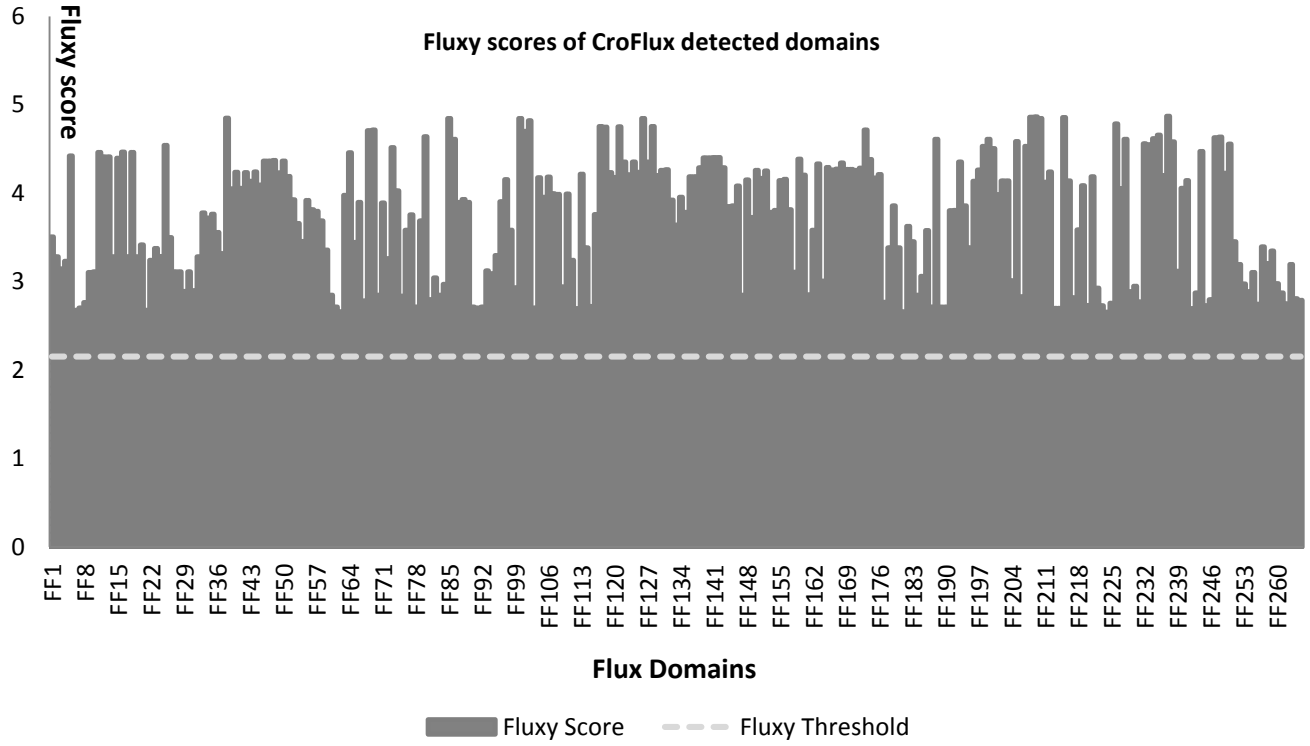

Figure 3 – Fluxy scores of collected Fast Flux domains

We also noted multiple level flux domains, like mentioned in [8]. These domains have multiple levels of name servers, where there is an overlap between the existing IP addresses.
An interesting type of domains are domains with a short TTL (less than 10 seconds), that rotate several IP addresses in every query. Usually, they return a single IP address. These domains tend to evade popular approaches to flux detection.

5 Conclusion

In this paper we introduced CROFlux, a method for detecting fast flux domains in the wild. Our approach leverages on publicly available knowledge about malicious delivery sites.
Our system can continuously detect unknown fast flux domains with an advantage that reduces false positives, relaying on detected domains with flux characteristics which are usually used for sharing malware.
With our conservative approach we get a list of real flux domains, and we avoid reporting benign domains with similar characteristics.
We also have similar lists of malicious domains that are collected using other systems developed in house and used in an EU funded project - Advanced Cyber Defence Centre (ACDC). CroFlux has been developed within the ACDC project, and temporally is tested on one DNS recursor hosted in our network. We have plans to expand the number of monitored DNS servers, in order to gain more flux domains. At the moment information about detected flux domains is sent to a centralized ACDC database called central clearing house. Further actions like domain takedown will be done in collaboration with law enforcement agencies and domain registrars with the aim of reducing botnets' damage.
As we can see from the results, botnet operators constantly work on new techniques to avoid detection and takedown of malicious domains. In order to have an updated classification, we updated the classifier using newly collected domains. Edge cases like small TTL domains with one IP address in response cannot be detected using an active querying approach, but passive replication helps on aggregating resolved IP addresses if this type of domains is queried enough times.

6 References

1. Weimer, Florian. Passive DNS replication., FIRST Conference on Computer Security Incident. 2005.
2. Zdrnja, Bojan, Nevil Brownlee, and Duane Wessels. Passive monitoring of dns anomalies, Detection of Intrusions and Malware, and Vulnerability Assessment. Springer Berlin Heidelberg, 2007. 129-139.
3. Thorsten Holz, Christian Gorecki, Konrad Rieck, Felix C Freiling Detection and mitigation of fast-flux service networks, Proceedings of the 15th Annual Network and Distributed System Security Symposium (NDSS'08). 2008.
4. Wegrzynowicz Patrycja, Jantura Jaroslaw, Detection of Fast Flux Botnets, 21th CENTR Technical Workshop,2009, Lisbon, Portugal
5. Bilge, L., Kirda, E., Kruegel, C., & Balduzzi, M. EXPOSURE: Finding Malicious Domains Using Passive DNS Analysis, NDSS 2011.
6. Perdisci, Roberto, Igino Corona, and Giorgio Giacinto. Early Detection of Malicious Flux Networks via Large-Scale Passive DNS Traffic Analysis, Dependable and Secure Computing, IEEE Transactions on 9.5 (2012): 714-726.
7. Farsight SIE, https://archive.farsightsecurity.com/, accessed 29.1.2014.
8. Wei Xu. Xinran Wang, Huagang Xie, New Trends in FastFlux Networks, Black Hat USA 2013
9. Github, PFFDetect - https://github.com/z0mbiehunt3r/pffdetect, accessed 6.2.2014.
10. Robert Edmonds – ISC Passive DNS Architecture, Internet Systems consortium, 2012
11. Zeus Tracker, https://zeustracker.abuse.ch/. Accessed 2.4.2014.
12. SpyEye Tracker, https://spyeyetracker.abuse.ch/, Accessed 2.4.2014.
13. Palevo Tracker, https://palevotracker.abuse.ch/, Accessed 2.4.2014.
14. SpamHaus, http://www.spamhaus.org/, Accessed 2.4.2014.
15. Feodo Tracker, https://feodotracker.abuse.ch/, Accessed 2.4.2014.
16. Malware Domain List, http://www.malwaredomainlist.com/, Accessed 2.4.2014.
17. Arbor Atlas, http://atlas.arbor.net/, Accessed 2.4.2014.

7 Acknowledgment

Writing of this paper is funded by the European Commission. The views and opinions expressed in this paper are those of the authors and do not necessarily reflect the opinions and views of the European Union. Also, we would like to thank anonymous reviewers whose comments helped us to improve this article.