Phishing Infrastructure Fluxes All the Way

Fast flux aims to keep phishing and scam campaigns afloat by provisioning a fraudulent Web site's DNS records to make the site resolve to numerous, short-lived IP addresses. Although fast flux hurts take-down efforts, it's possible to detect and defend against it.

D. Kevin McGrath,
Andrew Kalafut, and
Minaxi Gupta
Indiana University, Bloomington

Spam, phishing, malware, and other forms of Internet fraud cost consumers and businesses billions of dollars each year. As efforts to prevent Internet fraud have intensified, so have innovations in how fraudsters' provision their infrastructure to resist detection and take-down. In fast flux, an increasingly popular innovation, miscreants provision a fraud Web site's DNS for extreme availability. A Web site exhibiting fast flux typically resolves to many IP addresses, each with a short validity. Successive site resolutions often lead to a new set of IP addresses, which increases availability. At the same time, the addresses' short validity ensures that the sites' operators can provide a new, up-to-date list of machines to host the sites. Combined, these fast-flux features help keep fraud campaigns afloat longer during take-down efforts.

Fast flux comes in three basic flavors.[1] In addition to standard fast flux, the DNS server that leads clients around the world to the Web server can exhibit flux. This *DNS flux* could continue all the way up the fraud-associated domain's DNS hierarchy. *Double flux* refers to cases in which fast flux and DNS flux occur together at a domain.

Although spam-connected Web sites' use of fast flux is well known, we know little about how phishing campaigns use it. Here, we examine fast flux, DNS flux, and double flux in the phishing context and how our mechanism identifies all three flux flavors using statistical models on real-world data. Our results show that phishing campaigns use all forms of flux, though they do so less often than spam-connected Web sites. We also found that double flux works better than fast flux alone to ensure phishing campaign longevity. Because detecting and defending against flux can help fight various types of cybercrime, we also examine the practicality of real-time flux detection and its effect on Web browsing performance.

Data Collection

To collect data to investigate flux, we used real-time feeds of phishing URLs from MarkMonitor (www.markmonitor.com), OpenDNS PhishTank (www.phishtank.com), and the Anti-Phishing Working Group (www.antiphishing.org). From the URLs, we extracted a list of phishing-associated Web servers and collected various information about each.

Target Information

First, we performed DNS lookups on each Web server name in our feed, similar to how a client obtains the server's IP address. In DNS terms, this amounts to requesting the Web server's corresponding `A` (address) records. Next, we looked up the `NS` (DNS server) records for each domain and subdomain contained in the Web server name. We then looked up the `A` records corresponding to each DNS server discovered in the `NS` lookups.

For example, if the host name was www.xyz.example.com, we requested `A` records for www.xyz.example.com, NS records for www.xyz.example.com, xyz.example.com, and example.com, and `A` records corresponding to each `NS` record. We performed each DNS resolution once every 15 minutes. We chose this interval because most fluxing records' caching duration

Table 1. Overview of combined data from all three feeds.

CATEGORY		TOTAL IDENTIFIED IN DATA	
Phishing URLs		53,154	
Top-level domains (TLDs)		132	
	Generic TLDs (gTLDs)		11
	Country code (ccTLDs)		121
Web server host names		30,450	
	.com		13,846
	other gTLDs		1,889
	ccTLDs		14,715

is less than 15 minutes. Our time limit thus let us avoid receiving answers to fluxing domains queries from the local cache. We continued attempting DNS resolutions for each host name until they failed for one full day.

To accurately infer the presence of flux, we collected additional information for each discovered IP address. First, we performed geo-location using IP2Location software (www.ip2location.com), which helped us infer which countries the phishing infrastructure was located in. Next, we determined the Border Gateway Protocol prefix and autonomous system number (ASN) for each IP address using BGP routing tables (www.routeviews.org).

Data Overview

We collected data for 31 days beginning on 1 August 2008. As Table 1 shows, the data contained 53 thousand URLs spanning 132 top-level domains (TLDs).

Of the 30,450 Web servers in our feed, we resolved 15,547 (51.1 percent). The rest were likely taken down by the time they appeared in our feed. The active servers resolved to 15,230 IP addresses; although some Web servers are hosted on many IP addresses, many others share IP addresses. As a result, the Web server names are close in number to the IP addresses. This pattern is more startling in our DNS server results: the Web servers' 77,568 DNS server names corresponded to just 26,214 IP addresses.

Detecting Flux

Our goal is to identify the characteristics of flux in phishing data. Having collected the data described above, we now investigate how to detect fast flux and DNS flux in this data using a machine-learning algorithm.

Methodology

Given the IP addresses from one or more host name resolutions, we first identify whether fast flux, DNS flux, or double flux is occurring. To this end, we use Support Vector Machines (www.csie.ntu.edu.tw/~cjlin/libsvm), a class of machine-learning algorithms belonging to the generalized linear-classifier family. Training an SVM on a given data set produces a model that separates the data into two classes using a partition that minimizes error and maximizes the margin between the classes. We use this model to rapidly classify new data points.

As we describe later, we use several parameters to train two SVMs—one to determine fast flux and the other to determine DNS flux. Once these two are detected, determining double flux is straightforward. To train the SVM, we randomly select 10 percent of the data. Training includes a 10-fold cross validation to ensure a consistent model, unbiased by any particular data subset. We also need to preclassify data points. To do this, we apply a heuristic to the 10-percent data set based on the observation that fast flux host names have an ever-increasing number of IP addresses returned on successive resolutions. Over time, the number of IP addresses greatly outstrips all cases where flux isn't present.[2]

Once we've finished the preclassification, we ensure that the training data has a good mix of fluxing and nonfluxing hosts. We must be careful, however, about misclassifying content distribution networks; like fast-flux hosts, they can return a relatively large number of IP addresses corresponding to a host name, each with a short validity. However, CDNs generally don't return as many IP addresses as flux networks do, and the number of addresses generally doesn't increase as quickly with the number of resolutions performed. To ensure that our models handle the CDN case correctly, we added to the training data 13 popular, legitimate Internet sites that likely use CDNs. We derived these sites from the Alexa Web Information Service's top 1,000 most popular Internet sites in 16 categories (http://aws.amazon.com/awis).

The SVM training outcome is a model that partitions data into two classes: flux and not flux. Training a DNS flux model follows an identical process, but the heuristic checks whether the DNS servers' build up an ever-increasing number of IP addresses from successive resolutions.

Training Parameters

To accurately determine each flux type's occurrence, we explore a range of parameters. Some of the parameters are common to other related work,[3-6] but—as the "Related Work on Fast Flux" sidebar notes—such research explores them only in a fast-flux context, and doesn't investigate DNS or double flux. There are two other major differences in our model compared to these works. First, we use only parameters that can be easily and accurately derived. Thus, we avoid those based on the Internet whois database, which contains owner and other related domain information, but is sometimes unreliable. Second, we strive to derive the smallest set of parameters required for inferring each flux type with highest accuracy. Specifically, we consider the following six parameters.

Number of IP addresses (n_{IP}). The biggest flux indicator is the total number of IP addresses obtained from resolving a host name. However, nonfluxing servers can have numerous IP addresses for other reasons, so we must consider other parameters.

Number of associated ASNs (n_{ASN}). Generally, all IP addresses corresponding to a host name coexist in the same ASN because a single administrative entity maintains the host. Even for CDNs, results from a single vantage point exhibit the same behavior. However, this isn't true for fluxing hosts, simply because the bot armies of compromised host machines typically belong to multiple ASNs. Thus, if the resolved IP addresses belong to many ASNs, we take this to indicate flux.

Number of associated prefixes (n_P). We examine IP addresses' IP prefixes to explore for flux for similar reasons that we examine the number of ASNs. The IP addresses corresponding to legitimate hosts usually belong to a few BGP prefixes per host name; this is unlikely to be true for networks exhibiting flux. Thus, if a host name belongs to many BGP prefixes, we take it to indicate flux.

Number of associated countries (n_C). Even though many domains are hosted in multiple countries, individual hosts typically reside in a single country. In fact, hosts belonging to a particular country code TLD (ccTLD) are typically located on IP addresses physically residing within that country. Given this, if a host name belongs to multiple countries, we take it to indicate flux.

Number of DNS servers corresponding to Web servers (n_{NS}). Typical Web servers are associated with only a handful of DNS servers—usually two to three for most Internet Web servers.[3] Web servers exhibiting fast flux typically have many associated DNS servers—often up to 20. Thus, we assume that many DNS servers corresponding to a Web server indicates fast flux. (This parameter doesn't apply to DNS flux detection.)

Short time to live (TTL). Fluxing hosts typically use a shorter average IP address TTL than legitimate hosts because miscreants want to avoid being cached by client resolvers. The shorter the TTL, the faster a host can change its A records. We therefore use TTL as a

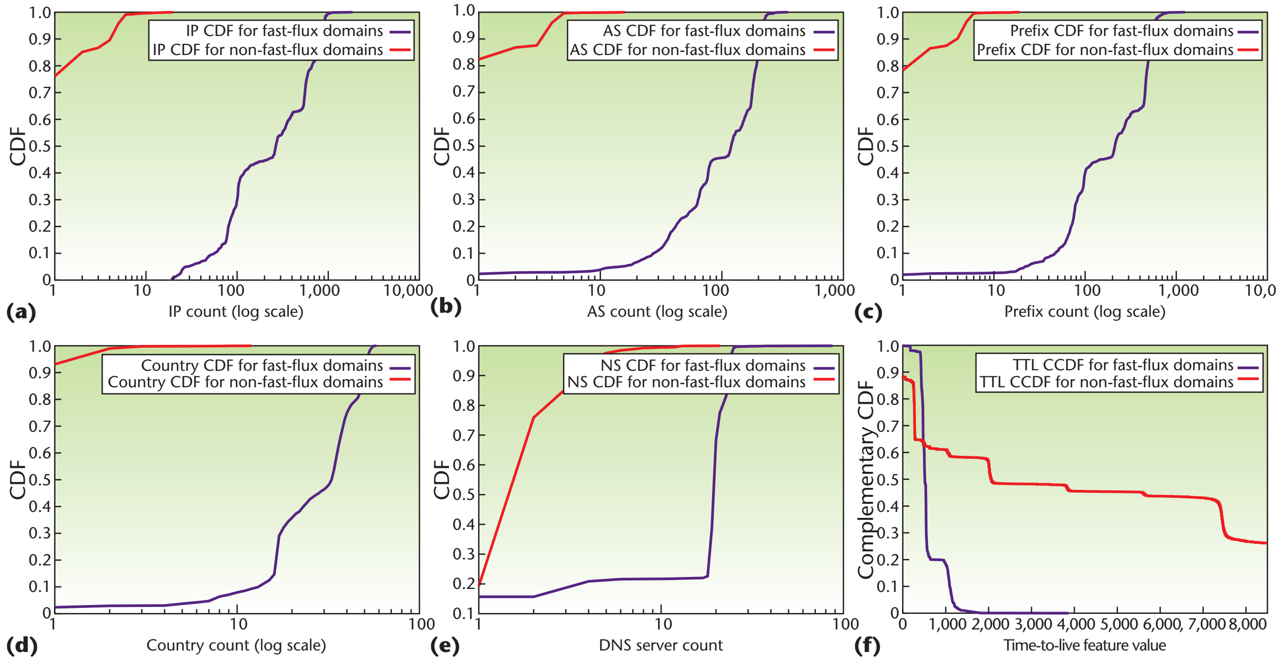
Related Work on Fast Flux

The Honeynet Project and Research Alliance was the first to recognize fast flux's prevalence in hosting malicious sites. In a white paper,[1] the alliance authors provided two real-world examples of campaigns exhibiting fast flux and double flux. In a recent similar study, Maria Konte and her colleagues examined fast fluxís role in hosting 21 online scams observed at a single spam trap.[2] Many fast-flux campaigns are hosted on botnets. Jose Nazario and Thorsten Holz investigated the behaviors of botnets behind fast flux's.[3]

Several researchers have also focused on fast-flux detection. Specifically, several flux-detection mechanisms have been proposed, including those by Thorsten Holz and his colleagues[4] and Emanuele Passerini and his colleagues.[5] Alper Caglayan and his colleagues also recently developed a fast-flux monitor to detect fast flux in real time.[6] Each of these works produced useful models, but focused only on fast flux. Our work also investigates double flux, as well as examining which parameters are most useful to detect fast flux and how much information is needed.

References

1. The Honeynet Project, *Know Your Enemy: Fast-Flux Service Networks*, July 2007, www.honeynet.org/papers/ff.
2. M. Konte, N. Feamster, and J. Jung, "Dynamics of Online Scam Hosting Infrastructure," *Proc. Int'l Conf. Passive and Active Measurement*, Springer-Verlag, 2009, pp. 219–228.
3. J. Nazario and T. Holz, "As the Net Churns: Fast-Flux Botnet Observations," *Proc. Int'l Conf. Malicious and Unwanted Software* (Malware), IEEE Press, 2008, pp. 24–31.
4. T. Holz et al., "Measuring and Detecting Fast-Flux Service Networks," *Proc. 16th Network & Distributed System Security Symp.* (NDSS), The Internet Society, 2008, www.isoc.org/isoc/conferences/ndss/08/papers/16_measuring_and_detecting.pdf.
5. E. Passerini et al., "Fluxor: Detecting and Monitoring Fast-Flux Service Networks," *Proc. Conf. Detection of Intrusions and Malware and Vulnerability Assessment* (DIMVA), LNCS 5137, Springer-Verlag, 2008, pp. 186–206.
6. A. Caglayan et al., "Real-Time Detection of Fast-Flux Service Networks," *Proc. Cybersecurity Applications and Technologies Conf. for Homeland Security (CATCH)*, IEEE CS Press, 2008, pp. 285–292.



CDF: Cumulative distribution function AS: Autonomous system count

Figure 1. The cumulative distribution function of each parameter for fluxing and nonfluxing Web servers. Each parameter exhibits a clear difference in the behavior of fluxing and nonfluxing hosts, indicating that all are promising for helping in flux detection. Although the number of IP addresses provides the biggest difference, no parameter offers enough difference to accurately distinguish flux on its own.

binary parameter in our model, using a threshold of 10 minutes: a TTL shorter than 10 minutes sets this parameter to 1.

Flux Prevalence in Phishing

Having established our parameters, we're now in a position to answer the question: How prevalent are fast flux, DNS flux, and double flux in phishing?

First, to detect fast flux, we trained an SVM on 10 percent of the data using our complete parameters set. We then applied the trained SVM to the remaining 90 percent of the data. We found that 11.4 percent of the active Web servers at the time of our measurements exhibited fast flux. Other research shows this percentage to be 30 percent for Web servers hosting scam sites pointed to in spam.[2] Interestingly, in our data, the 11.4 percent of phishing Web server names corresponded to 45.5 percent in the phishing IP addresses. Clearly, fluxing Web servers cycle through numerous IP addresses.

Next, as with fast flux, we trained a SVM classifier on 10 percent of the data to infer the presence of DNS flux in the remaining data. In this case, we used all the parameters except n_{NS}, which obviously doesn't apply to DNS flux. For Web server host names containing multiple subdomains, we looked for DNS flux at all levels up to the second-level domain name. We found that 61.7 percent of DNS servers exhibited DNS flux. Clearly, more DNS servers flux than Web servers, both in absolute numbers and in percentage. It's counter-intuitive that more DNS servers should flux than even the Web servers they point to. To gain a better insight, we looked at the IP addresses corresponding to the fluxing DNS servers. Surprisingly, there were only 904, indicating that while many DNS server names exhibit flux, they correspond to far fewer actual machines.

Finally, we looked for the presence of double flux in our data. We found that 77.6 percent of the fluxing Web servers were part of a double flux network. This implies that, typically, when phishing infrastructure fluxes, it fluxes all the way. In fact, in 98.8 percent of the double-flux cases, the associated DNS servers were fluxing at multiple levels of the DNS hierarchy. These percentages imply that most phishing campaigns are well provisioned against take-down efforts because they exhibit flux both at the Web server granularity and at various levels of DNS-server hierarchy.

Impact of Parameter Set

We created our initial SVM model for fast flux using all six parameters, and the DNS flux model using five. Our next step was to see which parameters offered the best accuracy without being redundant.

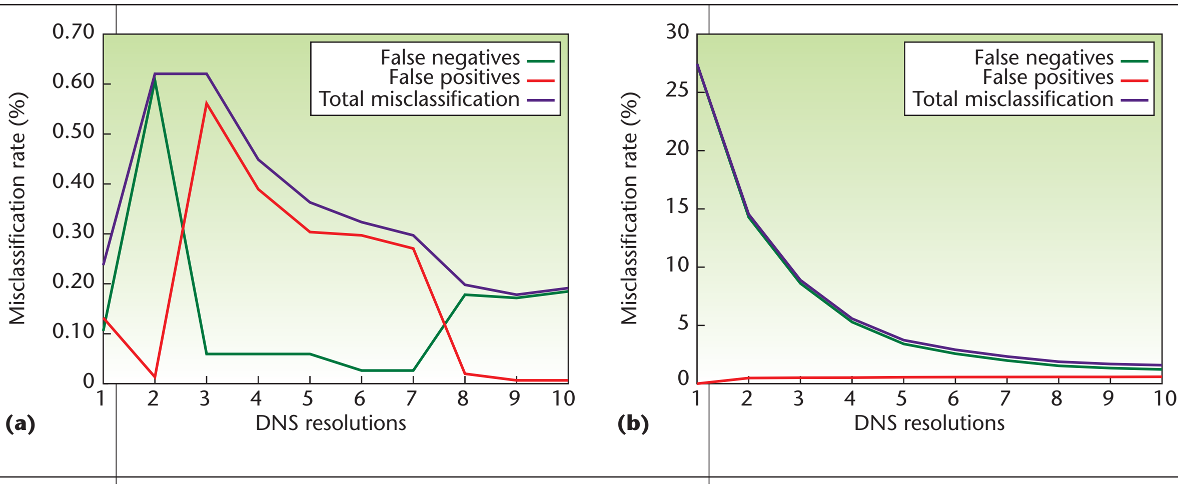


Figure 2. Effect of various DNS resolutions on identification accuracy. (a) Fast flux's misclassifaction rate is erratic, yet reaches only 0.64 percent at its maximum. (b) DNS flux's misclassification rates were considerably higher, requiring more DNS resolutions to detect accurately. This implies that detecting DNS flux will be more difficult and take longer compared to fast flux.

Fast-flux parameters. We began by examining how Web servers with and without fast flux fared for each of the six parameters. Figure 1 shows the cumulative distribution function (CDF) of fast fluxing and nonfluxing Web servers for each parameter (except for TTL, for which we show a complementary CDF (CCDF). As the figure shows, each parameter clearly distinguishes between hosts with fast flux and those without it. Approximately 80 percent of nonfluxing Web servers have a single IP address and belong to a single BGP prefix and a single ASN, while 90 percent belong to a single country. Additionally, 85 percent have at most two DNS servers, and more than 60 percent have an average TTL of more than 600 seconds. In contrast, only about 2 percent of the Web servers exhibiting fast flux are in a single ASN, prefix, or country; less than 20 percent have two or fewer DNS servers; and less than 20 percent have average TTLs of more than 600 seconds. Clearly, each parameter offers promise for helping detect fast flux.

Our next step was to investigate which parameters identify the most fluxing servers (low false negatives) without misclassifying nonfluxers (low false positives). To do this, we created SVMs to classify phishing Web servers with all the parameters and sequentially peeled off one parameter at a time. If there was no change between the parameter's presence and absence, we deemed it to be of no additional discriminatory value. We were thus able to determine the minimum set of parameters necessary for classification with good accuracy.

We found that combining four parameters—n_{IP}, n_{ASN}, n_P, n_C—produced the best results. Although a few other combinations of five and two parameters came close, this combination produced the least false positives and negatives. We therefore conclude that we don't need all six parameters to detect fast flux.

Parameters for detecting DNS flux. We repeated the same process to investigate which parameters were most useful for DNS flux. We found that several two-, three-, and four-parameter models do well. Incidentally, they all have one parameter in common: the number of IP addresses, n_{IP}. In fact, an SVM using only n_{IP} is almost as good as any model. We therefore conclude that n_{IP} alone is sufficient to detect DNS flux.

How Many DNS Resolutions Are Enough?

An identifying characteristic of both fast flux and DNS flux is that each resolution might return a new set of IP addresses. Typically, as the number of resolutions increase, so do the total number of IP addresses. This implies that having multiple DNS resolutions' results would be useful in accurately identifying flux. We therefore investigated just how many DNS resolutions are sufficient. The answer to this question has important implications for our technique's applicability: requiring fewer (possibly one) DNS resolutions to infer flux could help the client's DNS resolver protect the client from visiting a malicious site.

Resolutions for inferring fast flux. To determine the marginal returns on successive lookups, we constructed SVM models using the best (minimal) parameter set for fast flux (n_{IP}, n_{ASN}, n_P, n_C), and data obtained from 10 different DNS lookups. We compared each model's output to our earlier model, which used all resolutions for each Web server with all six parameters. Specifically, we looked at false positive and false negative rates for each model.

As Figure 2 shows, using multiple resolutions offers only a slight benefit. While misclassification seems to be highly erratic, the maximum misclassification rate is approximately 0.64 percent. Among the few misclassifications we found for a single DNS resolution, there were more false negatives than false positives. This is desirable to avoid penalizing good hosts. We therefore conclude that single DNS resolution is sufficient to classify a given Web server name as fast flux or not within an accuracy of close to half a percent. This further implies that administrators could easily integrate our method into antiphishing filters at either the client itself or the client's DNS resolver.

Resolutions for inferring DNS flux. We next investigated whether multiple resolutions are required to identify DNS flux. The outcome here was quite different than that for fast flux. Specifically, as Figure 2b shows, with only one DNS resolution on DNS servers, we saw more than 25 percent misclassifications, which is far higher than the maximum for fast flux. Even with 10 DNS resolutions, we still saw 2 percent misclassification, which is an order of magnitude more than with fast flux. Fortunately, most of the misclassifications were false negatives, not false positives, implying that good DNS servers were rarely flagged as fluxing while fluxing DNS servers were missed more often than desired. We therefore conclude that identifying DNS flux with fewer resolutions is significantly more difficult than identifying fast flux.

The obvious question is why DNS flux is so much harder to identify than fast flux. We found a couple of reasons for this. A DNS server name in our data was much more likely than a Web server name to have multiple IP addresses in the same country, averaging two IP addresses per country instead of one. It was also somewhat more likely to have multiple IP addresses in the same ASN and prefix. Additionally, while we saw a large difference in the median number of IP addresses returned for nonfast-flux Web servers and fast-flux Web servers—one versus 14—the difference wasn't present for fluxing and nonfluxing DNS servers. Instead, the median number of IP addresses that records returned for both of these was one, indicating that the fluxing DNS servers were only fluxing sometimes, not every time we looked them up. This difference was likely the major cause of the difficulty in identifying DNS flux domains with only a few lookups.

Flux and Fraud Longevity

Given that some phishing campaigns set up their infrastructures with fluxing hosts, an obvious question is: *Does flux help with the longevity of fraud campaigns?* To gain insight into this question, we looked at the length of phishing campaigns with and without flux.

Figure 3 compares the lifetimes of phishing-associated domains. We focused on domains that lasted for less than 10 days. Although this cut-off time is somewhat arbitrary, we didn't want to be biased by domains that lasted longer and simply remained undiscovered for a long time. The phishing Web servers in those domains would continue to operate irrespective of whether they exhibited any form of flux. In Figure 3, we divided the domains into three categories: those with servers exhibiting no flux; those with servers exhibiting fast flux, but no DNS flux; and those that exhibited double flux.

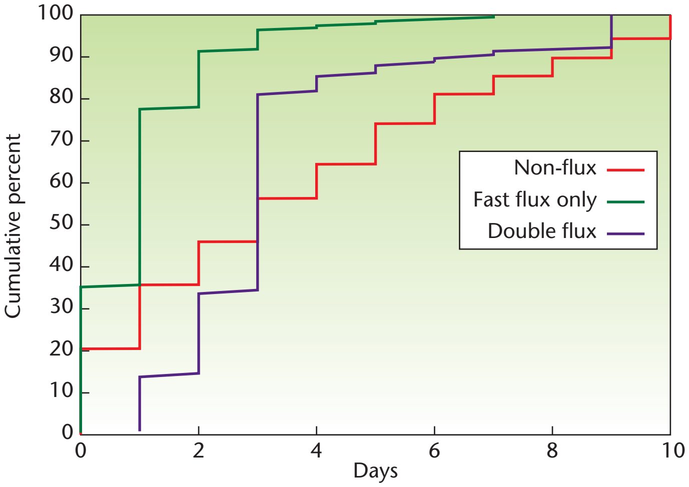


Figure 3. The lifetimes of double flux, fast flux only, and nonfluxing phishing domains lasting 10 days or less. Domains with fast flux alone were less enduring than those without flux.

Figure 3's curves are somewhat surprising: domains with just fast flux were less enduring than those without any form of flux. Specifically, while 65 percent of the phishing domains without flux lasted longer than a day, only 20 percent of domains with just fast flux managed to live more than a day. On the other hand, double flux seemed to help fraud longevity: 83 percent of domains with double flux survived more than a day. Clearly, when deciding to provision the phishing infrastructure with flux, it's better to provision it with double flux than fast flux alone.

In examining why fast-flux domains fare worse than those without any form of flux, we considered the recent attention that fast flux has received from commercial enterprises.[4,5] This could be leading to faster detection and take-down of domains that exhibit fast-flux characteristics. In contrast, although researchers have described DNS flux,[1] this article marks the first time it's been studied. This situation might help people using DNS flux. Also, given that a popular take-down methodology is to blacklist fast-flux Web servers or remove their DNS servers, the large number of DNS servers that frequently change and aren't under the same administrative control make it harder to eliminate DNS flux.

Figure 3 shows another interesting aspect: While double flux provides benefits to phishing-campaign longevity in the short term, it seems detrimental in the long run. Specifically, only 20 percent of phishing domains with double flux lasted for more than three days, while close to half of those without flux lasted more than three days. So, why does double flux become detrimental in the long run? We conjecture that the difficulty of taking down double flux domains keeps them up in the short term. However, over time, their behavior might attract more attention, leading to take-downs by other methods, such as removing the domain through the registrar.

Fighting Flux

Given that flux is now making its presence felt in various fraud infrastructures, defending against it is becoming important. There are two options for fighting against flux. One is to modify the DNS system to make flux difficult. The other is to detect fluxing behavior.

DNS Modification

A recent Internet draft suggested ways to modify the DNS to make flux difficult,[6] including that

- domain registrars limit changes to authoritative DNS servers to once every 72 hours,
- DNS servers not allow caching times of less than 24 hours for DNS server records, and
- DNS clients not accept records with a caching time of less than 12 hours.

Unfortunately, this proposal will be difficult to implement, as it requires changes to DNS clients on every computer. Further, the server changes would be ineffective because miscreants would run their own (unchanged) DNS servers. Additionally, the proposed changes could harm legitimate records, such as those resulting from CDNs.

Flux Detection

Compared to DNS modification, detecting flux is a more practical approach. There are two clear choices for where to run a flux-detection system: at the email server and at the DNS server. Each has its benefits and drawbacks. Because our focus here is on phishing, the email server seems the obvious choice: the system could simply extract URLs from phishing email to see if they used flux. The advantage here is that the system would see all incoming email and check them before users received them. The disadvantage is that it couldn't identify fluxing sites arrived at through the Web or any route other than email. Also, because email servers can't determine which links users will actually visit, it could do a lot of unnecessary checking.

Alternately, flux detection could run at the DNS resolver when a client contacts it to perform the resolution. This is a benefit because the system would see all the server names people actually visit, instead of just those derived from email. It also avoids unnecessary DNS resolutions because it uses information from actual DNS requests. The disadvantage is that it would see only URLs actually visited, so it might not have as much data to retrain with as it would at the mail server.

Performance Issues

Because flux detection running at the DNS resolver is more beneficial to the clients, we assumed it was the selected approach when investigating performance issues. We analyzed performance on a computer with a 2.4 GHz Intel Core 2 Duo processor and 4 Gbytes of RAM. We compiled the code with the gcc compiler using high optimization levels. We conducted timings on 4,770 host names and didn't use caching to improve performance for IP address overlap. Given the sheer amount of overlapping IP addresses among hosts, such a cache should improve performance.

In our tests, the classification time alone was 15.09 microseconds per classification. We needed more time to build the data point to classify, which involved IP to ASN mapping and geo-location. When we included the data-point build time, the average time rose to 1.286 ms per host name. We then conducted DNS resolutions to compare the extra overhead that flux-detection calculations would incur on median DNS resolution time. We found the median DNS resolution time to be 164 ms over 2.9 million unique DNS resolutions. We therefore conclude that checking for flux on each DNS resolution would typically add only 0.78 percent overhead.

Another important practical consideration is SVM retraining. Because we analyzed only a month of data, we never retrained our SVM. In a practical system, retraining would be periodically required to keep up with changes in fast-flux behavior—especially given the inevitable attempts to avoid triggering our system. Such retraining would likely be required only periodically, and would add little overhead: administrators could simply generate a new model offline and then copy it to the DNS resolvers for use.

Although it's important to ensure that flux detection doesn't incur false positives, experience shows that it can be incorporated as a practical test in identifying malicious infrastructures. Flux offers compelling advantages to miscreants in terms of fraud infrastructure longevity, so it'll likely appear more often in all types of malicious campaigns in the near term. In the long run, however, miscreants might discard it in favor of other attractive alternatives—especially as efforts to detect it in real-time intensify. □

Acknowledgments

We are grateful to MarkMonitor, PhishTank, and the Anti-Phishing Working Group (APWG) for providing us with phishing URLs, and IP2Location for providing their geolocation software. We also thank Craig Shue for providing the DNS resolution times for our performance evaluation. Finally, we thank Rob Henderson for facilitating the DNS lookups necessary for this research.

References

1. The Honeynet Project, *Know Your Enemy: Fast-Flux Service Networks*, July 2007, www.honeynet.org/papers/ff.
2. T. Holz et al., "Measuring and Detecting Fast-Flux Service Networks," *Proc. 16th Network and Distributed System Security Symp.* (NDSS), The Internet Society, 2008, www.isoc.org/isoc/conferences/ndss/08/papers/16_measuring_and_detecting.pdf.
3. A. Kalafut, C. Shue, and M. Gupta, "Understanding Implications of DNS Zone Provisioning," *Proc. 8th ACM Sigcomm Internet Measurement Conf.* (IMC), ACM Press, 2008, pp. 211–216.
4. J. Nazario and T. Holz, "As the Net Churns: Fast-Flux Botnet Observations," *Proc. Int'l Conf. Malicious and Unwanted Software* (Malware), IEEE Press, 2008, pp. 24–31.
5. A. Caglayan et al., "Real-Time Detection of Fast-Flux Service Networks," *Proc. Cybersecurity Applications and Technologies Conf. for Homeland Security (CATCH)*, IEEE CS Press, 2008, pp. 285–292.
6. J. Bambenek, "Double Flux Defense in the DNS Protocol," IETF Internet draft, work in progress, Nov. 2008.

D. Kevin McGrath *is a graduate student in the School of Informatics and Computing at Indiana University, Bloomington. McGrath has an MS in computer science from Indiana University. Contact him at dmcgrath@cs.indiana.edu.*

Andrew Kalafut *is a PhD candidate in the School of Informatics and Computing at Indiana University, Bloomington. His research interests focus on developing signatures for identifying cybercrime infrastructures. Kalafut has an MS in computer science from Indiana University. Contact him at akalafut@cs.indiana.edu.*

Minaxi Gupta *is an assistant professor in the School of Informatics and Computing at Indiana University, Bloomington. Her research interests are in computer networks and security, with special focus on cybercrime infrastructures. Gupta has a PhD in computer science from Georgia Tech. Contact her at minaxi@cs.indiana.edu.*