Study of Blacklisted Malicious Domains from a Microsoft Windows End-user Perspective: Is It Safe Behind the Wall?

Øyvind Jensen, Andrii Shalaginov, and Geir Olav Dyrkolbotn

Department of Information Security and Communication Technology,
Norwegian University of Science and Technology
oyvindjens02@gmail.com, {andrii.shalaginov,geir.dyrkolbotn}@ntnu.no

Abstract

The Internet is a dangerous place, filled with different cyber threats, including malware. To withstand this, blacklists have been utilized for a long time to block known infection and delivery sources. However, through blacklisting the domain names we are leaving a landscape of threats to be unknown and forgotten. In this paper, first, we investigate the current state-of-the-art in cyber threats available on such blacklists. Then, we study the corresponding malicious actors and reveal that those persistently appear since 2006. By shedding light on this part of the cyber threat landscape we target increased information security perception of the landscape from the perspective of the average end-user. Moreover, it is clear that the blacklisting the domains should not be one-way function and need to be regularly re-evaluated. Moreover, blacklisting might not be enforced by client applications in addition to outdated system software leaving real danger. For practical evaluation, we created a multi-focused experimental setup employing different MS Windows OS and browser versions. This allowed us to perform a thorough analysis of blacklisted domains from the perspective of the published information, content retrieved and possible malware distribution campaigns. We believe that this paper serves as a stepping stone in a re-evaluation of the once found and then blacklisted domains from the perspective of minimal security protection of a general user, who might not be equipped with a blacklisting mechanism.

1 Introduction

The world wide web has exploded in popularity the last two decades with the last numbers showing that there are 4.38 billion internet users[1]. With so many users there are lots of opportunity for profit, both legitimate and illegal. Internet companies are growing large, Google which started as a search engine is now one of the largest companies by stock valuation in the world[2], valued at 806.9 billion dollars. Facebook, the largest social network in the world with over 2.3 billion users[3] is valued at 528.9 billion dollars[4] on the stock market. With this many users and potentials for profit, new web portals and innovative applications are created all the time. Since the use of the internet is so widespread, the level of vigilance is not as high as when only specialized users used the internet. This makes the internet a good hunting ground for criminals that want to earn easy profits using social engineering attacks [9].

[1] https://www.internetworldstats.com/stats.htm, retrieved 25.5.19

[2] https://www.nasdaq.com/symbol/goog, retrieved 20.5.19

[3] https://www.statista.com/statistics/264810/number-of-monthly-active-facebook-users-worldwide/, retrieved 25.5.19

[4] https://www.nasdaq.com/symbol/fb, retrieved 20.5.19

Except for the dark web portals where the users need to use onion networks, other websites are considered to be publicly accessible. However, such websites that are either legitimate, but compromised websites or websites created with a malicious purpose are a serious problem to Internet cybersecurity and safety [16, 20, 7]. Given that these websites can be accessed by anyone means that people are at risk of being infected (by e.g. drive-by downloads) just by visiting websites in their (outdated) web browser. To combat this there have been developmental efforts towards a safer internet by building security features into operating systems, browsers and routers with e.g. certificates, blacklisting, sandboxing etc. Even with these developmental efforts towards a safer internet, users are still at risk if they are running old operating systems or by using old software that are missing protections against attacks. These older operating systems are typically not running the latest updates since Microsoft Windows XP, Vista, 7 and 8 are out of mainstream support[5]. This means that they will not get any more updates and other support and they can be severely outdated. Vulnerable users can be infected with malicious software which can then lead to these users being part of a botnet or cause other harm such as financial damages, privacy issues and other liability issues [30].

This paper targets multiple aspects of the overall field of publicly available websites that are blacklisted. Not every system has automated blacklists handling, therefore, leaving users susceptible to malicious actions. In particular, we will be looking at all possible aspects of the DNS-BH blacklist [6]. It is the initiative by RiskAnalytics to maintain the list of websites that are suspected of distributing or propagating malicious software. The focus of this paper is to understand what is the exact cyber threat landscape of such blacklisted websites. To comprehensively identify all affiliated risks, we will be looking at the following data: *Website contents*, *Software from websites*, *Domain name*, *Social engineering*, *Automated analysis*.

Our main contribution is an evaluation of the cyber threat landscape on one of the most famous blacklists, DNS-BH. Specifically, from how a user without security measures will experience visiting those websites. Blacklists are a useful, albeit old-fashioned and a static defence mechanism. It has certain limitations, such as it will not update itself, but the website addresses that are on the list will stay blocked for the users that employ the blacklist. The website addresses that are on the blacklists are on the lists because someone reported them as being malicious or spreading malware, or both. We are going to analyze these websites by looking at their content, software that is both being automatically downloaded and which you can download from the website, visiting links they have linked to, what servers they are using, etc. Besides, the focus will be on Microsoft Windows, known for being notoriously susceptible to malware attacks. To get this insight we will discover topics with topic modelling, identifying features that can be applicable for machine learning [31], gather intelligence from various sources and use these parts to create a holistic picture of the cyber threat landscape on blacklisted domains marked as malicious.

The work in this paper will help both the defenders and the users see what kind of cyber dangers that are present on the malicious internet. The paper identified that these domains are not running directly linked malware, which means that it's a vector that's less used, if at all. By exploring what is behind the blacklists we can raise awareness and knowledge of the threat landscape that is out there for users on malicious domains. The remainder of the paper is organized as follows. Section 2 gives an overview of the adversarial actions that can be affiliated with malicious websites. Section 3 presents evolution of threats and countermeasures in MS Windows. Suggested methodology is given in section 4 with experimental setup described in the section 5. Analysis of results and conclusions are presented in Sections 6 and 7 respectively.

[5] https://support.microsoft.com/en-us/help/13853/windows-lifecycle-fact-sheet, retrieved 3.1.19.

[6] https://www.malwaredomains.com/

2 Current State of the Art: Cyber Threat Landscape on the Public Internet

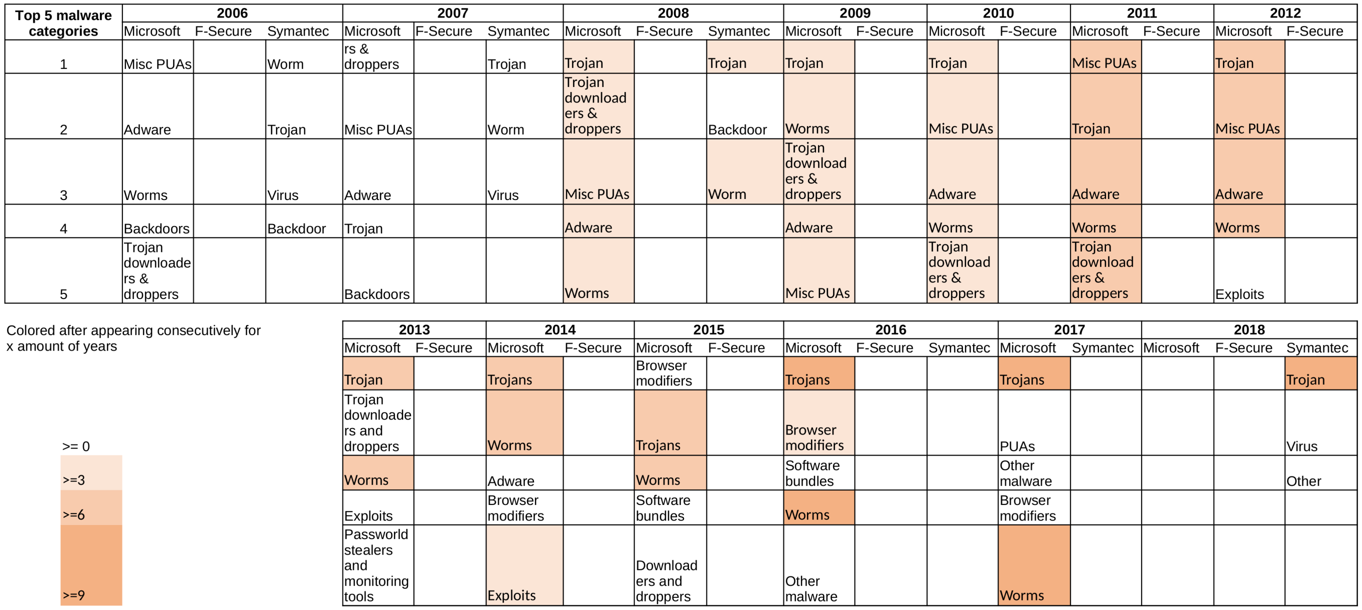
To the authors' knowledge, there has not been done a comprehensive review of the malware threats, vulnerabilities and risks that are focused solely on blacklisted malicious websites from an unprotected user's approach. Moreover, it is important to consider *social-technical* aspect of usage of malicious websites. By utilizing major anti-virus vendors' reports (Microsoft, Symantec, F-Secure) that look at the threat landscape through the view of the organization that has written them such as [14, 34, 6] we will build an understanding of the general threat landscape on non-blacklisted websites. The high-level summary of the security reports is shown in the Figure 1.

These reports identify, enumerate and explain the threats they see in their monitored systems. The attacked and infected users are customers of the companies that are creating these reports and thus much of the information they have is sensitive and confidential, even so, there is much information available in these reports. Furthermore, we can get a more "ground truth" perspective on cyber threats when we can see the landscape from *major* companies in the industry providing security services and one of the companies that are responsible for one of the operating systems used by most people in the world[7]. Combining the findings from the security reports with our experimental parts helps us to understand threats and the corresponding malicious content that can be found on the websites that are blacklisted and labelled as malicious by DNS-BH.

2.1 Social Engineering in Cyberspace: Human-related Aspects of Cyberthreats

One of the peculiar approaches used on malicious websites is social engineering. The persons using computer systems are an exploitable part of the computer ecosystem which is easier than targeting, e.g. the operating system itself. They are the ones setting up exploitable IoT devices that can be captured by criminals and used in botnets [1], they are also the ones that can be tricked into visiting malicious websites as seen in [8, 9]. When comparing non-expert and expert security practices [10] there were multiple interesting findings. The non-expert, the *average user*, were more inclined to follow the advice and more norm-like security practices that were popular around mid-2000s such as browsing known websites and using antivirus solutions. Not that these practices are necessarily bad, but what is a “known” website can vary extremely much from person to person. Additionally, antivirus solutions do not necessarily protect end-users from every threat. This is where the expert practices come into play since one of the most used practice was updating software. The software can quickly become outdated and some programs more than others, such as browsers and PDF-readers. By updating these, especially the browsers, the users can stay protected much more easily by e.g. getting the updates to blacklists and new features such as multithreaded support and sandboxing as mentioned in [29]. Another key aspect of personal security on the internet was the handling of passwords. A non-expert was more prone to often change passwords and instead of using password managers as the experts, they would try to remember passwords. On the other hand, the expert users had 2-factor authentication high on the list of important security measures, this is most likely because it is a much safer way to secure accounts. An attacker will have a much harder time getting access to both your computer and your phone.

[7] https://netmarketshare.com/operating-system-market-share.aspx, retrieved 17.5.19



Top 5 malware categories	2006			2007			2008			2009		2010		2011		2012	
	Microsoft	F-Secure	Symantec	Microsoft	F-Secure	Symantec	Microsoft	F-Secure	Symantec	Microsoft	F-Secure	Microsoft	F-Secure	Microsoft	F-Secure	Microsoft	F-Secure
1	Misc PUAs		Worm	rs & droppers		Trojan	Trojan		Trojan	Trojan		Trojan		Misc PUAs		Trojan	
2	Adware		Trojan	Misc PUAs		Worm	Trojan downloaders & droppers		Backdoor	Worms		Misc PUAs		Trojan		Misc PUAs	
3	Worms		Virus	Adware		Virus	Misc PUAs		Worm	Trojan downloaders & droppers		Adware		Adware		Adware	
4	Backdoors		Backdoor	Trojan			Adware			Adware		Worms		Worms		Worms	
5	Trojan downloaders & droppers			Backdoors			Worms			Misc PUAs		Trojan downloaders & droppers		Trojan downloaders & droppers		Exploits	

Colored after appearing consecutively for x amount of years

>= 0

>=3

>=6

>=9

2013		2014		2015		2016			2017		2018		
Microsoft	F-Secure	Microsoft	F-Secure	Microsoft	F-Secure	Microsoft	F-Secure	Symantec	Microsoft	Symantec	Microsoft	F-Secure	Symantec
Trojan		Trojans		Browser modifiers		Trojans			Trojans				Trojan
Trojan downloaders and droppers		Worms		Trojans		Browser modifiers			PUAs				Virus
Worms		Adware		Worms		Software bundles			Other malware				Other
Exploits		Browser modifiers		Software bundles		Worms			Browser modifiers				
Password stealers and monitoring tools		Exploits		Downloaders and droppers		Other malware			Worms				

Figure 1: Summary of cybersecurity reports over last decade

The social engineering attacks that are most often seen are obfuscated URLs that can be spread via e.g. Twitter with its 280-character limit, phishing emails, drive-by downloads, spoofed websites and scareware [9]. Spoofed websites are often part of a phishing phase [8] in which a fake version of a known website is created, the URL to that fake website is distributed by e.g. mass mailing and users that access the website can thus be lured into thinking it is the actual website it is trying to imitate. When a user has opened the website a drive-by download can occur [5, 9, 33]. This is a successful social engineered attack where a user has been tricked into visiting this website and gotten malicious files downloaded to their computer.

Drive-by downloads are not the only way a user can get malicious files downloaded on their computer. Often a user will be enticed by a download button [28] or something similar in which the graphical user interface has been tailored to exploit the trust the user has to it [9]. A user could also download software deliberately from a suspicious source that is malicious without the user knowing it [2]. The downloaded software could be a variety of malware, but often it is *Trojans*. Attackers doing social engineering has a goal in mind and that is private information because that is how they make their salaries as explained in [13] and multiple Symantec reports, e.g. [40] and [41] in their *underground economy* sections.

2.2 Technical Aspects of the Threats on the Malicious Websites

The intertwinement of social and cyber threats is understandable when one needs to employ social engineering to attack a user. Therefore, there has been put many resources into identifying malicious activity and malicious websites so that the human factor is taken out of the equation. Browsers should have the defences to be able to stop the user from being exploited built-in by default, but that is a hard task given how browsers and rendering of websites have been developed over the years. Sandboxed browsers that can multi-thread has become standard in the recent years[8], but the users that are without this protection and malware that can override or circumvent the sandboxed environments are still threats. JavaScript and its integration in browsers as seen in [3, 11, 29] make it possible for malware to attack users through their browsers with JavaScript. Often, it is malicious code that only exploits particular browser versions, extension and plug-in combinations. This makes it hard to detect malicious websites causing them to go undetected and not being blocked by blacklists.

Even with the cloaking capabilities of JavaScript code, there are still ways to detect and mitigate malicious websites. Authors [4] look at what is the best way for search engines to intervene against malicious websites. They come up with a solution that makes a website lose relevancy as a form of punishment when it is detected as bad. This works as a carrot to quickly respond to the infection and makes it cheaper to detect. Another approach is to use host-based features and the URLs themselves to create machine learning models that automatically detect malicious websites [12]. Taking it further than using the basic contents and host-based features one can use multiple layers as seen in [44], where both the application- and the network-layer traffic were used to detect malicious websites. Some, [42], have taken it even further by utilizing more advanced features where they look at combining multiple machine learning methods to build a huge associative model to detect malicious websites. This approach was mentioned by [33] where they foresaw a hybrid approach utilizing new technology to build a better detection solution. It was mentioned in [5] that a combination approach was in the works, but that it was yet to see daylight.

Recently there has been built a detection model based on combining content on websites and the path clients take to reach a website. This model was described in [32] where they built it on

[8] https://chromium.googlesource.com/chromium/src/+/master/docs/design/sandbox.md

the features that can be gathered from redirection, HTML and JavaScript. One of the biggest problems with malicious websites is their evasive nature, where it is very hard to find malicious data because of environment checks that are being done by these websites. If the visitor does not have a fingerprint that matches the fingerprints supported by the exploit kit, the malicious website will hide their malicious data by, e.g. redirecting to a benign site. Their model is based on a honeyclient[9] that collects data and redirect paths on these websites. They designed it to detect both malicious redirect graphs with exploit URLs and evasion redirection graphs. Finally, Shibahara et al. [32] categorizes the most common systems for detecting malicious websites that are utilizing drive-by downloads [4, 12, 44, 33, 5, 32]: *Large-scale user traffic, System behavior and Web content and redirection.*

3 Evolution of Cybersecurity Threats and Countermeasures in Microsoft Windows

3.1 Noteworthy takeaways from industry reports

The noteworthy takeaways were identified when going through the reports and when we saw that in some way, they had a big impact on the cyber threat landscape. This enables us to identify trends and major changes for both the IT industry and the malware *industry*. Two key topics that we think of today as given, was fleshed out in 2006 and 2007 already. Both topics are mentioned already in 2006, the first being that the malware industry is shifting from caring about their reputation to caring about their coffers and how to fill them with gold instead of wasting their time for *fame*. The second topic is a web-based malicious activity, with XSS attacks being launched at the then famous MySpace.

3.2 MS Windows: operating system security measures

This section is based solely on Microsoft's Security Intelligence Reports. In these reports we are presented with insights into the data they are generating from all their users, every nook and cranny of the operating system is available for these authors. This makes them able to analyze malware and cyber incidents in a way no other organization like e.g. Symantec and F-Secure can. Since Microsoft are the ones developing the operating system and the tools that they include with it they can update and upgrade the different solutions they deliver and get instantaneous feedback on what is working and what is not.

Table 1: Evolution of MS Windows OS security measures: 2002-2019

Year	**Major event**	**Description**	**Report**
2002	/SafeSEH and /GH (compiler flags)	In Visual C++ .NET the compiler flags were introduced. These increases the application's resilience to stack-based buffer overruns.	V. 8 [16]
2003	Scheduled security updates	Microsoft started with regular security updates every second calendar Tuesday of every month. Additionally, they opened for out-of-band security updates in critical cases.	V. 6 [14]
2004	Windows XP SP2	A major update that introduced new features in Windows such as the Security Center, improved Windows Firewall, a pop-up blocker in IE and other configuration options that made the OS safer. DEP was one of them in addition to better heap protection through heap manager enhancements.	V. 7 [15], V. 8 [16]
2005	Malicious Software Removal Tool	Anti-malware software that Microsoft updates monthly through Windows Update and Microsoft Update for free to Windows users.	V. 7 [15]
2006	Windows Vista and Windows Server 2008	Introduced new features such as UAC and ASLR.	V. 7 [15]

[9] A client that is created with vulnerabilities so that it will be attacked when visiting malicious websites since it will seem like it can be exploited [26].

2008	Windows Vista SP1 and Windows Server 2008 RTM	Structured Exception Handler Overwrite Protection (SEHOP) was implemented to stop exception handler exploitation.	V. 8 [16]
2009	Windows 7 and Windows Server 2008 R2	Safe Unlinking in the kernel pool is an enhancement to kernel security so that malware cannot so easily exploit kernel pool overruns.	V. 8 [16]
2009	Enhanced Mitigation Experience Toolkit	The Enhanced Mitigation Experience Toolkit (EMET) was released in 2009 to be an extra safety layer for Windows XP, Vista, 7, Server 2003, Server 2008 and Server 2008 R2.	V. 12 [18] and [25]
2011	Change AutoRun feature in Windows XP and Windows Vista	Changed the AutoRun feature to behave like the default in Windows 7. Was pushed in an automatic update.	V. 10 [17]
2011	Infection rates for 64-bit Windows editions surpasses 32-bit Windows editions	The infection rates Windows Vista SP1 and SP2 64-bit versions were higher than the 32-bit versions.	V. 12 [18]
2012	Windows 8	Microsoft added real-time antimalware and antispyware to the default configuration of Windows 8.	V. 14 [20]
2013	Windows 8.1	Machines upgraded from Windows 8 to Windows 8.1 will have their default real-time security software changed to Windows Defender if their previous software was determined incompatible with Windows 8.1.	V. 17 [21]
2013	Internet Explorer 11	IExtensionValidation interface in IE11 introduced a new mechanism that enables security software to determine if a website is secure before allowing ActiveX controls to run, thus Java exploits cannot run on the machine.	V. 19 [22]
2014	Updates for Internet Explorer 8 to 11	Out-of-date ActiveX controls will be blocked, such as outdated versions of Java.	V. 19 [22]
2015	Windows 10 and Microsoft Edge	Microsoft Edge, the default browser in Windows 10, was released without support for Java or other ActiveX plugins.	V. 20 [23]
2015	Windows 10 — Windows Defender activation	Windows Defender is also automatically activated upon installation if no other real-time security product is detected. For Windows 8 and 8.1 Windows Defender also gets enabled automatically after a few days after installation if no other real-time security product is detected.	V. 20 [23]
2015	Windows 10 — Windows Defender cloud sample submission	If enabled in Windows Defender settings, Windows Defender will upload suspicious, but undetected files, to their cloud backend where the file will be analyzed with machine learning, heuristics and automated file analysis to determine if it is malicious or not.	V. 21 [24]
2019	Windows 10 — Windows Sandbox	Microsoft introduced a sandbox solution which creates a temporary version of Windows 10 in which you can install applications or visit websites which will be run isolated from the host.	[43]

Key security features introduced over the years:

- *ASLR - Address Space Layout Randomization*
- *DEP - Data Execution Prevention*
- *UAC - User Access Control*
- *ActiveX controls*
- *AutoRun in Windows 7*

4 Methodology for automated analysis of front-end generated content on blacklisted websites

This section presents a comprehensive methodology used to collect and analyse data from the blacklisted malicious websites as shown in the Figure 2.

- *Data Collection.* Collection of relevant information and files for the future investigation
- *Identification of acquired files.* A new round of data collection from VirusTotal is used to acquire: domain, hashes of the files that were collected from the domain, VirusTotal reports on the file hashes.
- *Preliminary automated analysis* will go through the collected data done by the crawler (The overview and functionality of which is shown in the Figure 3).

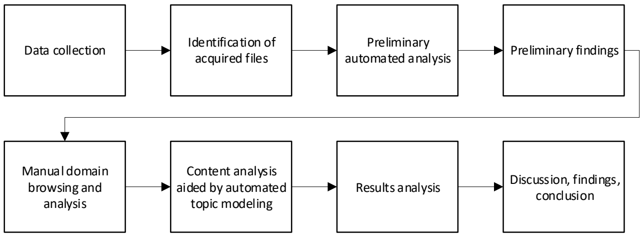

Data collection
Identification of acquired files
Preliminary automated analysis
Preliminary findings
Manual domain browsing and analysis
Content analysis aided by automated topic modeling
Results analysis
Discussion, findings, conclusion

Figure 2: Flowchart illustrating analysis stages and progression

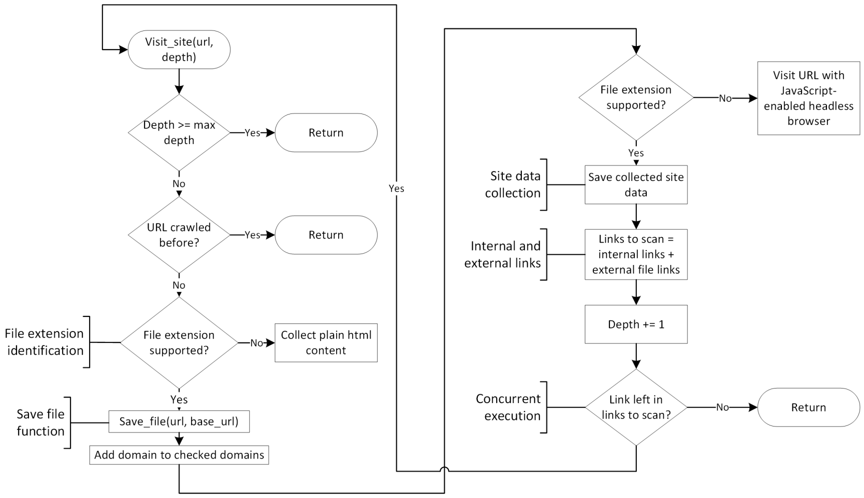

Visit_site(url, depth)
Depth >= max depth
Yes
Return
No
URL crawled before?
Yes
Return
No
File extension identification
File extension supported?
No
Collect plain html content
Yes
Save file function
Save_file(url, base_url)
Add domain to checked domains
Yes
File extension supported?
No
Visit URL with JavaScript-enabled headless browser
Yes
Site data collection
Save collected site data
Internal and external links
Links to scan = internal links + external file links
Depth += 1
Concurrent execution
Link left in links to scan?
No
Return

Figure 3: Automated crawler routine

- *Manual domain browsing analysis* helps to see what kind of domains that are malicious and how they look to a normal user. This was recently done in [2] where this gave additional information about the malware and the delivery process.
- *Content analysis aided by automated topic modelling.* Each website that is successfully crawled will most likely have some content in the body section of the HTML-document, this is the main content which is interesting for us to look at. Topic modelling can thus help us find commonalities between the websites when we feed all the bodies into a topic modelling algorithm.
- *Result analysis* will be used to make prevention guidelines, threat-, risk- and vulnerability assessments of the domains.

5 Experimental Setup

Because possible malicious websites are going to be analysed, several dedicated Virtual Machines (VM) were created to address also possible different security measures. The main idea is to simulate an average end-user environment that might not be updated or patched together with average user experience and cybersecurity awareness.

5.1 Data collection

The main data source for this study is a list of websites from DNS-BH blacklist. After analysis of the list, there was found to be 1,907 relevant to this research domains: 1,139 are labelled as *Malware* and 768 are labelled as *Malicious*. The subsequent analysis includes processing of all relevant content fetched from the websites and meta-information about the domain name.

5.2 Overall experimental architecture

There have been used two machines as host systems for VMs. One was mostly used when creating the Cuckoo Host VM while setting it up and configuring the necessary parts. When the Cuckoo Host VM was ready it was transferred over to the host machine that was going to run the analysis part. It was imported into VMware where it was configured as a copied VM and set to work on the new host as shown in the Figure 3.

- *Host 1 - Workstation* Detailed specifications. *OS:* Windows 10, Version 1803 (OS Build 17134), *CPU:* AMD Ryzen 5 2600X @3.6 GHz, Turbo: 4.2 GHz, 6 cores, 12 threads, *RAM:* Corsair Vengeance LPX DDR4 16GB 3000 MHz C16, *3 Disk(s)*.
- *Host 2 - Dell Precision M4600* Detailed specifications. *OS:* Linux Mint 19.1, *CPU:* Intel i7-2760QM @2.4 GHz, Turbo: 3.5 GHz, 4 cores, 8 threads, *RAM:* OEM supplied, DDR3 SDRAM 24GB 1333 MHz, *3 Disk(s)*

The main difference from the Host 2 and Host 1 is that the manual analysis VM is not present. Additionally, the Cuckoo host VM had some small configuration changes made when running it, some of them to correct issues and some for performance enhancements (related to several processors and memory, since Host 1 and Host 2 had different quantities of both). On this system as with Host 1, all VMs did not run at the same time. Instead, the crawler VM ran till it was finished, then the Cuckoo Host VM ran.

5.3 Linux Guest VMs: initial crawling and analysis

- *Manual Analysis VM* Used to manually analyze a selection of domains: Linux Mint 19.1, Firefox 66.0.1 64-bit; RAM: 8GB, CPU: 6 cores, HDD 20GB.
- *Crawler VM* Used for the crawler: Linux Mint 19.1, RAM: 19.5GB, CPU: 4 cores, HDD 40GB.

5.4 Windows Guest VMs: subsequent sandboxing analysis

- *Windows XP SP3 OS specifications:* Windows XP, Version: 5.1, Build: 2600, Service Pack: 3, Browser: Internet Explorer 8.0 (Final). Note that the TLS support for Windows XP is limited to 1.0 [19] so that websites that are utilizing encryption higher than this will not work if they do not have backwards compatibility added for Windows XP. *Hardware specifications:* (RAM) 2GB, 1 CPU, 127Gbytes HDD
- *Windows 7 OS specifications:* Windows 7 Enterprise, Build: 7601, Service Pack: 1, Browser version: Internet Explorer 11.0. *Hardware specifications:* RAM 4 GB, 2 CPU, HDD 40 GB.
- *Windows 10 OS specifications:* Windows 10, Version: 1803, Build: 17134, Browser version: Microsoft Edge 42.17134. *Hardware specifications:* 4GB RAM, 2 CPU, 40 GB HDD

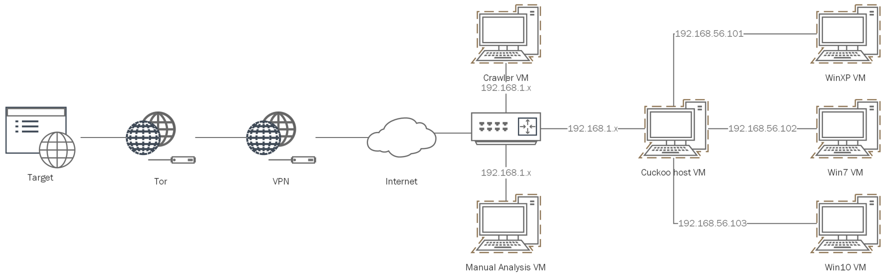


Figure 4: A network diagram of the internet-connected systems in this paper

Domain name	Count
www.calabriasportfishing.com	95
www.vishwaweighingsystem.com	24
hkitforce.com	17
webdesigning.name	17
podstrigis.com	16

Table 2: Top 5 resolved domains with their respective count number

File type	Count
HTML document	7,064
UTF-8 Unicode text	250
JPEG image data	247
data	166
XML 1.0 document	76

Table 3: Top 5 file types of all downloaded files by crawler

6 Evaluation of acquired data and results

This section contains an analysis of the following data:

- *Crawler:* File type distribution of files downloaded, Number of domains crawled, Number of domains from blacklist, Content analysis and topic modeling
- *Domain data collection:* GeoIP - Countries involved, Whois - Distributions, URLAbuse - BGPRanking
- *Cuckoo:* Scans statistics

6.1 File and links analysis

The website crawling and data collection resulted in 947 domains, where 668 were unique and in the blacklist. Some domains had multiple different URLs stored. Listing 2 represents top 5 most resolved domains.

We then gathered the file type information using the *file* command in Linux which was gathered by a script written in Python that would call a subprocess and collect the output for each. By trimming them down we were able to more easily see the file distribution. By analyzing the *data*, *UTF-8* and *ASCII* files further we were able to determine that 9 of the data-files were binary files, but most likely corrupted files where the crawler must have crashed or timed out during the creation of the files were originally garbled. The rest of the data-files were HTML-files with encoding issues. Out of the UTF-8 files, 2 were HTML-files the rest were logfiles created by the crawler. The same was the case with the ASCII files, none of them being HTML-files, all being logfiles.

Since we found no executables, scripts or similar interesting files in the files directly downloaded by our crawler, we checked the office files on VirusTotal, but no engines detected anything suspicious. We then used the ClamAV[10] antivirus engine to do a recursive scan on all folders within the crawler's download folder, this includes the files downloaded by the *pywebcopy*

[10] https://www.clamav.net/, retrieved 26.05.2019

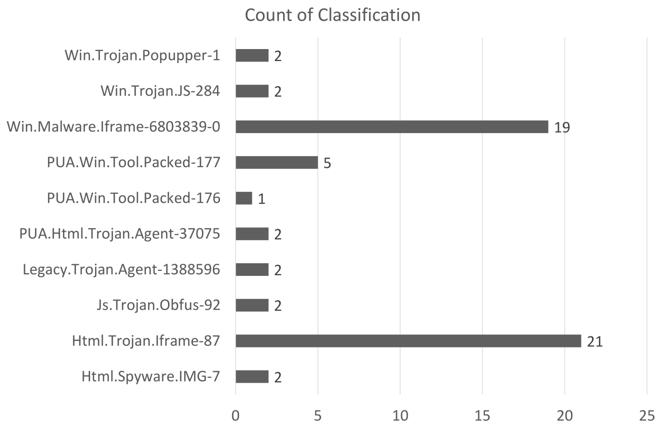


Figure 5: ClamAV classification distribution of all downloaded files from the crawler

module that is creating its own folder inside the crawler's download folder. ClamAV found 58 possible threats after scanning 26,278 files where the classification distribution is shown in figure 5. Distribution of threats is as following: Trojan - 54%, Win.Malware - 33%, PUA - 10% and Spyware - 3%. Two domains stood out: Domain A - 19 detections of *Html.Trojan.Iframe-87* and Domain B - 17 detections of *Win.Malware.Iframe-6803839-0*.

These threats are typical delivery malware, by being embedded on websites their task is to infect users with malware, often trojans. Iframes are easily embedded as images you cannot see with the size specified to 0 and their position rendered in the negatives - outside the browsing area.

6.2 Textual content analysis on the webpages

Our content analysis started with cleaning our documents then we looked at words used in the body of all websites. This gives us an indication of the complexity, e.g. if every site was just a template with "Error, the server did not respond" when an unsupported fingerprint is detected.

- *Without JS rendering enabled* Number of non-words (that are occurring more than 100 times): 51. A number of (any) words in the body text that are occurring more than 100 times: 199. A total number of (any) words in body text: 63,806. A total number of unique words: 131. A number of sites with content: 918.
- *With JS rendering enabled* A number of non-words (that are occurring more than 100 times): 69. A number of (any) words in the body text that are occurring more than 100 times: 288. A total number of (any) words in body text: 80,019. A total number of unique words: 188. A number of sites with content: 898.

We further removed the following: duplicate bodies in our data frame, words without (Latin) characters a-z, A-Z and words with length less than 2. We did not see a clear difference in the basic word analysis between the pre-duplication removal and the post-duplication removal.

6.3 GeoIP analysis

From the GeoIP, we made a list of the top 5 most represented countries seen in listing 4. Majority of the websites were hosted in the United States, which can be attributed to the fact that major cloud providers for shared hosting and dedicated servers are located there.

Country	Count
United States	341
Ireland	96
Germany	86
Netherlands	37
China	31

Table 4: Top 5 most represented countries

Name	Count
Private Person	23
Redacted for Privacy Purposes	7
REDACTED FOR PRIVACY	3
Chen Jianjun	2
SAKURA Internet Domain Registration	2

Table 5: Top 5 names from WHOIS

Classification label	Count
malicious site	4,890
malware site	3,477
phishing site	1,136

Table 6: Classification of blacklisted domains in VirusTotal with their respective count

Anti-Virus Vendor	Count
Malware Domain Blocklist	1,887
AutoShun	1,340
G-Data	1,308
Fortinet	1,048
BitDefender	836

Table 7: Top 5 detections by Anti-Virus engines

6.4 WHOIS analysis

Since WHOIS data is not reliable, not much weight will be given, but we did do some frequency analysis on the results from that too, seen in listing 5.

6.5 VirusTotal analysis

VirusTotal had records for all the domains, except 1 in our selection from the blacklist. This is not surprising given that they are blacklisted for malicious activity. In listing 6 is the classifications that these domains have in the VirusTotal database from the different vendors that have data on these domains. Listing 7 shows the top 5 antivirus engines (vendors) that have data on the domains in question.

6.6 Manual analysis of websites

To select which domains to analyze in our manual analysis we decided to use measures that can easily be selected and confirmed by others doing similar experiments. The selected measures were; Cuckoo scores above 4. 5 and VirusTotal reports with more than 4 positive results on the particular domain. This produced a list of 92 unique domains. To analyze these the *Manual Analysis* VM from section 5.3 were used. When browsing these domains, the web console and network inspector in Firefox were used. The network tab allowed us to see the connections as they were made, and the inspector tab allowed us to see the source code of the website directly. This allows us to see and follow network requests and inspect to see if there are iframes or other hidden parts on the websites. To save our analysis we had an Excel spreadsheet open with the following columns: domains, descriptions, type, social engineering techniques, private information requests, reloaded, warning from Google Safe Browsing, links/redirect landings. Of these columns, the most relevant became the domains, description, type and Google Safe Browsing. The "type" column is the determined type of the domain by us when analyzing and the distribution is shown in figure 6. Additionally, Google Safe Browsing gave following warnings: "Yes, deceptive site" - 11, "Yes, attack site" - 1, "Yes" (without threat specification) - 14, "No" - 67.

To further analyze the domains, the 30 first were checked manually in VirusTotal where there is an option to get a more detailed view of domains when using their website. This

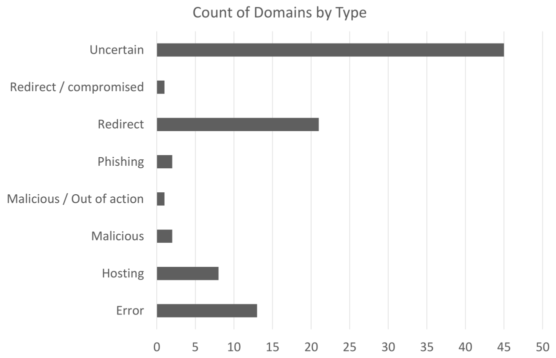


Figure 6: The types of domains visited during the analysis

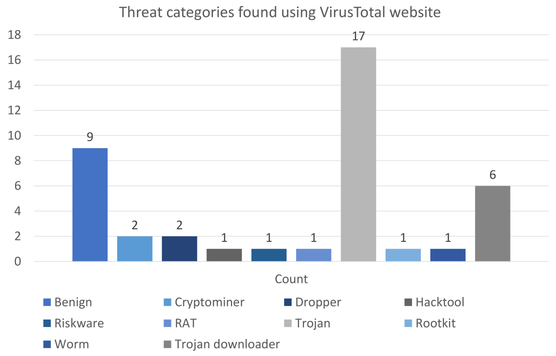


Figure 7: Threat categories according to VirusTotal advanced domain information

gives us the following additional information about domains[11]; categories it has been classified as, passive DNS replication to get the IP the address resolves to, WHOIS lookup, observed subdomains, URLs associated with the site (e.g. download links), downloaded files and lastly the communicating files which communicates with the domain on execution or opening. For the 30 analyzed domains, we saw that 27 of the domains had files in subfolders. This means that when visiting the domain, you are not seeing the subfolder if it has not been linked to. One of the most interesting things to get from doing this was the threat categories present on the analyzed domains as shown in the Figure 7 since these corresponded well with the categories found in our threat report analysis seen in table 1.

6.7 Final remarks

The collection phase did not result in any dangerous malicious executables one could expect from malicious domains that are labelled as spreading malware and being malicious. On the other hand, it lines up with what the landscape has evolved to over the years. A shift that already started in 2006 towards the internet and the possibilities for exploitation and the increased attack surface. The notion that you should trust a website was still a thing in 2008 since Microsoft recommended to ”only browse sites you trust” [14], this notion is abolished today for

[11] In this example the following domain was used:https://www.virustotal.com/#/domain/shzwnsarin.com, retrieved 26.5.19

many reasons, but an obvious one being the advertising networks that are on the majority of sites you should *trust*. To get a better understanding of this content we can look at the topic modelling results which possibly can shine more light on the actual contents. Woocommerce is one of the most popular commerce plugins for WordPress and the associated words, posted and contact are regularly seen on webshops. Many of the websites in our dataset are most likely compromised WordPress sites which have been a target of malware campaigns for a while.

Some malicious detections were made by ClamAV, most of these being iframe-based malware as seen in figure 5 with the categories being aligned with the categories that the overall landscape have produced the last 12 years. When we then manually analyzed these domains, we saw that most of the malware one could download from them was unavailable for direct download since they resided in subfolders, where 27 of 30 domains are utilizing subfolders for delivery of malware. There could be multiple reasons for this, they could be malware hosts, or they could be showing different websites, benign-looking ones, when browsing them manually or with a crawler. The threats that were found on the detailed VirusTotal domain information again shows that the blacklisted domains are delivering malware that aligns with the general cyber threat landscape with trojans being the most popular malware type to deliver. Additionally, PUAs, hack tools are detected by the tools which are also seen by us on the blacklisted domains.

From the OS usage perspective, Windows XP is not shown in Microsoft reports from 2014 and onward since it had passed the end of life support in 2014 as mentioned in the introduction of this paper. Even though it is not supported anymore it proved in our small sample size to be more secure on average than Windows 7 which is a much newer operating system and used by many more. It could also be that some security features that are mentioned in the Microsoft Windows operating system and security measures evolution has not been enabled since by default they are opt-in [16]. Further, without many of these enabled and a bigger focus on Windows 7 by malware developers, Windows XP could skirt by, even though that is unlikely since exploits for Windows XP are probably included in the exploit kits by default. Our Windows XP VM was installed with just the default applications that come with Windows and updated to IE 8.0 so it might be that the attack surface is limited as recommended by [27] so that exploits would not exploit it as easily.

Further, the threats we have seen downloaded from our crawler and executed on our VMs correlates well with other information that Symantec has seen over the years. In their 2018 internet security threat report [41] they say that 1 in 10 URLs are malicious. This is an interesting fact when setting in perspective with their internet security threat report from 2012 [38] where only 1 in 532 of websites was found to be infected with malware. What this means is that malware since 2006, have turned towards web-based malware and attacks since that is the avenue that is their opening. The reasons behind this are that the Windows operating system has been hardened over many iterations and the newest version has real-time antivirus, security features such as UAC, ASLR, DEP, kernel unlinking and so on. Even with increased defences, the attackers have not been resting either since these developmental changes have created an arms race on both sides. Currently, many malware relies on client applications and human factor as a way into the system. Symantec has statistics for the malware volume since 2002 to 2018, where a clear shift in malware sample numbers between 2009 and 2010 are shown due to polymorphism, obfuscation, encryption and the use of simple droppers [35, 36, 37, 39, 40, 41].

7 Conclusions and Discussions

In this paper, we have researched the cyber threat landscape on blacklisted malicious domains. It was shown that this landscape is not substantially different from what is considered the general cyber threat landscape by the industry. Even with a limited dataset with many non-resolving domains we have gathered enough data to see that the focus of malicious domains is on exploit kits and not on directly spreading malware via direct links to malicious files. Further, we have shown the broadness of malicious activities on the sandboxed test VMs when visiting malicious websites. The discovered threats are mainly exploiting kits used by malicious domains and the most commonly delivered malware are trojans. This means that the most notorious threats are the ones that are let in by users, often unknowingly since it happens in the background. By demystifying the landscape, it was further presented that there is not necessarily a need for users to use any special software other than a solid real-time protection software that is receiving updates at a regular pace. Besides, by having the most recent and updated OS version, updated browser with multithreading, sandboxing and a modern API that a user reduces the risk of being affected. Finally, there is a need for proper cybersecurity awareness campaigns to be able to reduce such risk even further.

References

[1] Akamai. [state of the internet] / security q4 2017 report. Report, 2017.

[2] Francesca Bosco and Andrii Shalaginov. Identification and analysis of malware on selected suspected copyright-infringing websites. Report, EUIPO, 2018.

[3] Marco Cova, Christopher Kruegel, and Giovanni Vigna. Detection and analysis of drive-by-download attacks and malicious javascript code. In *Proceedings of the 19th international conference on World wide web*, pages 281–290. ACM, 2010.

[4] Benjamin Edwards, Tyler Moore, George Stelle, Steven Hofmeyr, and Stephanie Forrest. Beyond the blacklist: modeling malware spread and the effect of interventions. In *Proceedings of the 2012 New Security Paradigms Workshop*, pages 53–66. ACM, 2012.

[5] B. Eshete, A. Villafiorita, and K. Weldemariam. Malicious website detection: Effectiveness and efficiency issues. In *2011 First SysSec Workshop*, pages 123–126, 2011.

[6] F-Secure. Threat summaries volume 1, 2014.

[7] F-Secure. Threat report 2015, 2016.

[8] B. B. Gupta, Nalin Asanka Gamagedara Arachchilage, and Konstantinos E. Psannis. Defending against phishing attacks: Taxonomy of methods, current issues and future directions. 2017.

[9] Ryan Heartfield and George Loukas. A taxonomy of attacks and a survey of defence mechanisms for semantic social engineering attacks. *ACM Computing Surveys*, 48(3):37, 2016.

[10] Iulia Ion, Rob Reeder, and Sunny Consolvo. "... no one can hack my mind": Comparing expert and non-expert security practices. In *SOUPS*, volume 15, pages 1–20, 2015.

[11] Clemens Kolbitsch, Benjamin Livshits, Benjamin Zorn, and Christian Seifert. Rozzle: De-cloaking internet malware. In *Security and Privacy (SP), 2012 Symposium on*, pages 443–457. IEEE, 2012.

[12] Justin Ma, Lawrence K Saul, Stefan Savage, and Geoffrey M Voelker. Beyond blacklists: learning to detect malicious web sites from suspicious urls. In *Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 1245–1254. ACM, 2009.

[13] Michael McGuire. Into the web of profit. 2018.

[14] Microsoft. Microsoft security intelligence report volume 6: July - december 2008, 2008.

[15] Microsoft. Microsoft security intelligence report volume 7: January - june 2009, 2009.

[16] Microsoft. Microsoft security intelligence report volume 8: July - december 2009, 2009.

[17] Microsoft. Microsoft security intelligence report volume 10: July - december 2010, 2010.

[18] Microsoft. Microsoft security intelligence report volume 12: July - december 2011, 2011.

[19] Microsoft. Support for ssl/tls protocols on windows, 2011. [Last Accessed May 27th 2019].

[20] Microsoft. Microsoft security intelligence report volume 14: July - december 2012, 2012.

[21] Microsoft. Microsoft security intelligence report volume 17: January - june 2014, 2014.

[22] Microsoft. Microsoft security intelligence report volume 19: January - june 2015, 2015.

[23] Microsoft. Microsoft security intelligence report volume 20: July - december 2015, 2015.

[24] Microsoft. Microsoft security intelligence report volume 21: January - june 2016, 2016.

[25] Microsoft. Moving beyond emet, 2016. [Last Accessed: May 26th 2019].

[26] MITRE. Honeyclient. [Last Accessed May 29th 2019].

[27] Kartik Nayak, Daniel Marino, Petros Efstathopoulos, and Tudor Dumitras. Some vulnerabilities are different than others - studying vulnerabilities and attack surfaces in the wild. In *RAID*, 2014.

[28] Terry Nelms, Roberto Perdisci, Manos Antonakakis, and Mustaque Ahamad. Towards measuring and mitigating social engineering software download attacks. In *USENIX Security Symposium*, pages 773–789, 2016.

[29] Patil Shital Satish and RK Chavan. Web browser security: Different attacks detection and prevention techniques. *International Journal of Computer Applications*, 170(9), 2017.

[30] Andrii Shalaginov, Sergii Banin, Ali Dehghantanha, and Katrin Franke. Machine learning aided static malware analysis: A survey and tutorial. In *Cyber Threat Intelligence*, pages 7–45. Springer, Cham, 2018.

[31] Andrii Shalaginov, Lars Strande Grini, and Katrin Franke. Understanding neuro-fuzzy on a class of multinomial malware detection problems. In *International Joint Conference on Neural Networks (IJCNN) 2016*, pages 684–691. Research Publishing Services, 2016.

[32] Toshiki Shibahara, Yuta Takata, Mitsuaki Akiyama, Takeshi Yagi, and Takeshi Yada. Detecting malicious websites by integrating malicious, benign, and compromised redirection subgraph similarities. *2017 IEEE 41st Annual Computer Software and Applications Conference (COMPSAC)*, 01:655–664, 2017.

[33] A. K. Sood and S. Zeadally. Drive-by download attacks: A comparative study. *IT Professional*, 18(5):18–25, 2016.

[34] Symantec. Symantec internet security threat report volume x: Trends for january - june, 2006.

[35] Symantec. Symantec internet security threat report volume xiii: trends for july–december, 2008.

[36] Symantec. Symantec internet security threat report volume xiv: trends for 2008, 2009.

[37] Symantec. Symantec internet security threat report volume 17: trends for 2011, 2012.

[38] Symantec. Symantec internet security threat report volume 18: 2012 trends, 2013.

[39] Symantec. Symantec internet security threat report volume 21, 2016.

[40] Symantec. Symantec internet security threat report volume 22, 2017.

[41] Symantec. Symantec internet security threat report volume 24, 2019.

[42] Senhao Wen, Zhiyuan Zhao, and Hanbing Yan. Detecting malicious websites in depth through analyzing topics and web-pages. In *Proceedings of the 2nd International Conference on Cryptography, Security and Privacy*, pages 128–133. ACM, 2018.

[43] Windows. Windows sandbox, 2018. [Last Accessed May 28th 2019].

[44] Li Xu, Zhenxin Zhan, Shouhuai Xu, and Keying Ye. Cross-layer detection of malicious websites. In *Proceedings of the third ACM conference on Data and application security and privacy*, pages 141–152. ACM, 2013.