MalFire: Malware Firewall for Malicious Content Detection and Protection

Wyatt Yost
Computer Science Department
Truman State University
Kirksville, United States
wey2383@truman.edu

Chetan Jaiswal
Computer Science Department
Truman State University
Kirksville, United States
cjaiswal@truman.edu

***Abstract*—The online portion of modern life is growing at an astonishing rate, with the consequence that more of the user's critical information is stored online. This poses an immediate threat to privacy and security of the user's data. This work will cover the increasing dangers and security risks of adware, adware injection, and malware injection. These programs increase in direct proportion to the number of users on the Internet. Each of these programs presents an imminent threat to a user's privacy and sensitive information, anytime they utilize the Internet. We will discuss how current ad blockers are not the actual solution to these threats, but rather a premise to our work. Current ad blocking tools can be discovered by the web servers which often requires suppression of the ad blocking tool. Suppressing the tool creates vulnerabilities in a user's system, but even when the tool is active their system is still susceptible to peril. It is possible, even when an ad blocking tool is functioning, for it to allow adware content through. Our solution to the contemporary threats is our tool, MalFire.**

Keyword—components; Malware, Content Injection, Adware, Privacy, Security.

I. INTRODUCTION

Over the last couple of decades, the Internet has become more accessible to people across the globe. In December 1995, there was an estimate of 16 million internet users [1] [2]. Now, the number of internet users has risen to 3.6 billion users [3]. People have learned to coexist with the Internet and now live all aspects of their lives online; from shopping on Amazon to chatting with an old friend on Facebook. However, this new lifestyle has a downside. As the number of users increase so does the critical infrastructure of the cyber community and subsequently, the amount and variety of cyber-attacks increase as well. Cyber-attacks happen incessantly due to the reliance and inclusion of the Internet in a person's normal daily life. The effects of a successful cyber-attack could seriously impact the socioeconomic and physical welfare of the citizens.

Just using the Internet poses such an immediate threat to personal privacy and information, that cyber security is an absolute necessity. While cyber security covers a range of services; the threat we will concentrate on is malicious content such as adware and malware. This type of malicious content plays a major role by creating vulnerabilities or stealing private information.

Historically there have been numerous successful cyber-attacks such as CryptoLocker which would encrypt user's files, but would release a key upon payment. The ILOVEYOU cyber-attack which took control of the user's computer to overwrite system files then would spread itself via email. [4]

The latest cyber-attack known as WannaCry was released earlier this year. According to [5]: "*It was spread via an operation that hunts down vulnerable public facing SMB ports and then uses the alleged NSA-leaked EternalBlue exploit to get on the network and then the (also NSA alleged) DoublePulsar exploit to establish persistence and allow for the installation of the WannaCry Ransomware*". After installation WannaCry would encrypt the user's files asking for a payment to decrypt the files. If a user does not pay the ransom, their files are permanently encrypted. Not just the average user was affected, WannaCry targeted healthcare facilities denying doctors access to their patient's files. Cyber-attacks at this level not only need to be stopped quickly, but it is equally important to make the user aware of the intrusion. If the user is not notified of such intrusion, private data could be compromised resulting in financial loss or identity theft.

Our work culminated in the development of a tool which is called MalFire. MalFire is a malware/adware detection system and a non-traditional firewall combined. The purpose of MalFire is to intercept malicious content before it reaches the browser thereby creating a more secure and private browsing experience.

The remainder of the paper is as follows: Section 2 outlines preliminary work, Section 3 discusses our motivation, Section 4 covers our methodology to the development of our tool, Section 5 reviews prior literature to our work, Section 6 covers our contribution along with future evaluation, and Section 7 is the conclusion to our work.

II. Preliminary Work

Our initial approach was to use a brute force method; go to a website analyze the code and then monitor the network traffic between our computer and the web server. The objective was to find adware or malware on the page. We found and obtained the packet associated with content of the ad. Adware was easy to find due to its nature of wanting to be noticed. Adware can be displayed as a banner, on the side panel, or even before a video.

Figure 1. Adware injection on Huffington Post

The Fig. 1 shows an adware injection occurring on the Huffington Post website denoted by the two arrows titled Advertisement 1 and Advertisement 2. We confirmed this injection by checking the source Internet Protocol (IP) address of the packet carrying this adware against the web server source IP address. We discovered that the packet carrying the ad was not from the same domain therefore; the ad was not intended to be on the webpage and an adware injection was occurring.

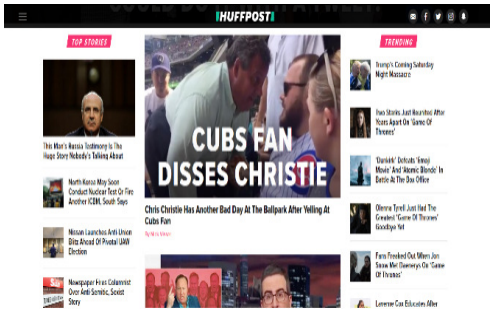


Figure 2. No adware injection

The image in Fig. 2 shows how the website should have displayed before the adware was injected. In Fig.2 the website is more secure than in Fig. 1 since the adware packet did not reach the end user. To that end the browsing experience has become safer.

On this occasion, the malicious content blocked was strictly adware, but that is not always the case. It could have very well been a wrapper program; which in this context refers to an adware that is embedded with malicious code like Malware, Virus, Worms or Trojans. This type of software is designed to inflict harm upon the computer it resides on or to obtain user information. Damage to the system can range from stolen data to complete encryption/loss of all data and files. Eliminating adware packets from the stream will lessen the risk of the end user facing private/sensitive information loss or identity theft.

III. Motivation

In this section, we review the negative impacts of adware and current ad blockers that motivated us to work on this problem. Adware on websites is difficult to ignore especially when the adware is crafted by social engineering to appeal specifically to the user's desires. This is a construct produced by data collected from a user's browsing patterns, website return rate, as well as geological location. Adware follows a user to different websites advertising a product that social engineering indicates an end user might be interested in. To demonstrate this more clearly, we did a search about St. Louis and things to do in St. Louis. The outcome is shown in Fig 3.

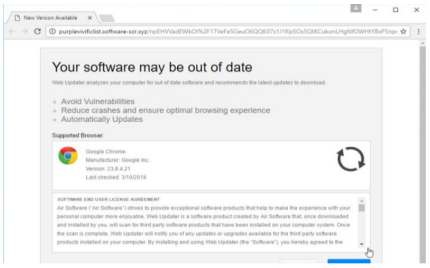
Figure 3. Banner ad crafted after recent searches

The results show that our data was tracked and subsequently an advertisement about flights to St. Louis from our local airport was crafted. But not all adware is designed to create interest in a product, another technique intends to scare a user into downloading a malicious product.

Malicious adware sites, as displayed in Fig 4a, will attempt to pressure or deceive a user into giving personal data such as credit card details, their social security number or banking information [6]. This scare tactic relies on intimidation by making false claims about or to the user. These claims will mislead the user into sending their banking information in an effort to rescue their computer from a critical situation. Websites of this nature also attempt to dupe users into believing that their software is out dated and needs to perform an update. This action often

leads to the download of malware; an example of which is displayed in Fig. 4b [8].

a. FBI Ransomware [6]



b. False Claiming Webpage [7]

Figure 4. Malicious Websites

A study was conducted and the results revealed 47% of participants were unsure of the origin of the programs and 97% of the participants have no recollection of an end user license agreement [9]. When a user clicks on these malicious ads; more happens behind the scenes. According to [10]:"*The programs are rarely installed from a conspicuous website, but rather through social engineering banner ads, drive-by-downloads, and through peer-to-peer networks with misleading filenames*". The users are tricked and unaware that their computer is being invaded. This is one of the many negative effects of adware.

With the understanding of adware along with the above statistics; we claim that not only is the adware paradigm robust, but also the threat it presents is all too real. The direct negative impact of such programs are not restricted to the security risk, but also causes an overhead. Overhead in this context refers to the excessive time and resources required to perform a certain task. When an advertisement runs, it requires more data transfer which causes an overhead. According to [11]: "*All ad-blockers except Ghostery give around 25-34% savings in the amount of data transferred (on average). This is a bit higher than the 18% saving reported by* [13] *and 13-34% reported by* [15]". Data transfer reduction will decrease load time, but each of these ad blockers carries their own overhead. Therefore, the data transfer reduction is nearly cancelled out by the overhead of the exhaustively sizeable libraries [11].

Current ad blockers do protect the users to a certain extent. Ad blocking companies have developed an acceptable ads program which is called whitelisting. Whitelisting will allow the display of pre-approved ads on websites. As mentioned in [12], "*Ads that meet its criteria for things like placement, size, and distinction, are 'whitelisted'—that is, if the company displaying the ads is willing to split the revenue gained by whitelisting with Adblock Plus*". This poses a risk in the security and a false sense of trust to the end user. A user believing the ad blocker is protecting them could trust ads that are displayed on the website not realizing they could contain malware injected by a third-party.

Monetary success for ad blocking companies does not only come from white listing, but also from selling the user's data as mentioned in [12], "*The company (Ghostery), however, makes money by collecting anonymized data on what those trackers pick up. It repackages that data and resells it to publishers, websites, and other companies. The company claims it can use the information to help improve the speed, privacy, and performance of their sites.*". The users are not aware that their data is distributed to ad companies crafting improved adware. We suspect if the data is being sold to the company, then the chances are that ad service will be white listed. Once more this raises the chances of user's attention being drawn to the ad, which leads us to suspect the chances of a malicious injection will increase. This creates a vicious cycle driven by the attraction of the ad and trust a user has in their software.

IV. Methodology

The approach to the development of MalFire was to identify the current state of adware blocking tools. The first issue found was that websites could identify an active ad blocking tool. The stipulation to access the website is for a user to disable the ad blocker. Disabling an ad blocker increases the risk of being phished by injected content or a wrapper program.

Similarly, this is the same risk incurred with whitelisted ads, but because of the trust factor, whitelisting is a greater danger. Whitelisted ads will display on a website as if no ad blocking tool is present. The possibility of clicking on a wrapper program is likely to increase as the user relies on the functionality of their ad blocking tool. This is not the only security risk; the users data profile is being sold to adware companies. Data profiles are used to create more targeted ads that could be added to the whitelist.

Another cyber-hazard is edge internet service provider injection. Network operators do inject forged packets which carries the false content. The end goal of edge ISP is to gain revenue, but there

have been findings where malicious content has been detected [17]. When there is no monitoring agent between Edge ISP and the end user false content injection occurs.

The last thing ad blockers are doing is causing an overhead to the user. The overhead causes slower performance in website display and in load time. This does not exploit or create a security risk, but the goal is to reduce the overhead that is currently happening with ad blockers. Now, we present our different dimensions of MalFire.

A. *Privacy of the User*

The privacy of the user is our top priority. When a user requests a website, each packet is analyzed which has fragments of the website and possible user sensitive information. Valid segments from the website pass through the tool and invalid segments will be discarded and ignored.

B. *Packet Sniffer*

For MalFire to work we had to create a network sniffing tool. By capturing the packets with the network sniffing tool, MalFire can dissect them to get at and analyze the data inside each packet.

To obtain the packets Jnetpcap application program interface was implemented (API). With this API, it was possible to get all packet data necessary for our tool.

C. *Packet Validation and Identification*

The four packets payloads of interest are as follows, TCP (Transmission Control Protocol), IP (Internet Protocol), HTTP (Hypertext Transfer Protocol) and HTML (Hypertext Markup Language packets). These packets carry vital information to validate each packet.

Inside the TCP packet is the port number and the web server used to connect with the end user. When a potential valid packet is received from that web server the port number should always match. But if packets resemble the web server, but is received on different ports then there's a probability of the packet being forged. Performing this check will decrease the risk of third party injection.

The IP packet checking is comparable to the TCP packet checking, but instead of port number MalFire is checking the destination IP address of the web server. If the IP address does not match the original IP address connection then the packet is originating from a different source and should not be in the network stream.

A HTTP packet holds the cookie and the web server name. This is a quick identification method that allows MalFire to look and identify if the server that the packet is from is an ad server and look if the cookie is being set for an ad.

Finally, the HTML packet allows MalFire to obtain the website code. The validation process looks over the code in the packet looking for ad signatures that are added and maintained by EasyList and EasyPrivacy. Both databases are used by most ad blocking software. If any of the keywords match then the packet will be stored for review and further assessment.

D. *Packet Blocking Algorithm*

In our packet blocking algorithm we identify if a packet is valid or contaminated. Pseudo code of the algorithm goes as follows.

```
Input: Packet, GlueList
 If header is found
   Get payload
   For GlueList Size
     If packet contains GlueList[i]
   Modify and log IP
```

The GlueList is used to store all the filters used in MalFire. These filters will be used to check against the packet content to identify if there is adware or malware contained in the packet. Next, we obtained the payload of each packet which would be compared against all the filters inside the GlueList. If the packet does not contain an adware signature then the packet is not logged. If the packet does have adware signature the packet is to be logged and will store all the payload information. Once the packet has been logged we reviewed to check if it came from an ad server or is a valid ad from the current domain a user is accessing. The packets that originate from an ad sever, their IP address are will be added to the iptables to block future connections.

E. *Iptables*

Iptables is a Linux based firewall that can be modified to create an agile system by inputting an IP address to drop packets. The purpose of Iptables in our system is to block malicious and adware servers. If malicious content or adware reaches the end user, the tool will update the Iptables to block the connection.

V. LITERATURE REVIEW

In this section, we discuss previous works that cover ad blockers, malware injection, anti-ad-blockers, and adware.

The Anti-ad blocker is the focus of the work presented in [18]. When the user loads the webpage the anti-ad-blocker analyzes the Document Object Model for modifications and can scan extension resources for ad blocking tools. This backs our claim that an ad blocker is identifiable at application layer. MalFire's advantage is that it is not at the application layer and subsequently is invisible to anti-ad

blockers. This work provides a noticeable distinction between current ad blocking tools and the tool developed in our work.

The work presented in [17] covers the false content injection by Network Operators to their own end users network stream. It was found the predominate threat originated from the two largest Network Operator in China; however, an additional 14 groups were found to inject adware and malicious content. The end goal of the Network Operator is increase revenue or transmit malicious content. Network Operators can inject false content because of the lack of a monitoring agent. Our work resulted in a tool that is a monitoring agent and filter for the end user.

The effectiveness of ad blockers at their default setting is covered in the work detailed in [15]. Adblocker Plus and AdGuard give minimal protection and Ghostery gives no protection when first added to the browser. AdBlocker Plus currently filters 58% of third-party domains, but 11% of the domains are whitelisted. An end user is subject to adware exposure as the whitelist grows. In addition, an end user may witness adware due to a browser leak allowing it to request the third-party domain even when blocked by the ad blocking tool. Our tool does not authorize whitelisted ad and no there is risk of browser leakage.

The work done in [19] covers the impact and perception of whitelisted ads. The acceptable ads program only had 9 ads whitelisted in 2011. This number grew to 5,900 by the Spring of 2015 and is still growing at a rate of 11.4 ads whitelisted per 1.5 days. Our work has a zero tolerance to whitelisting ads because whitelisted ads create a false sense of security and posing an immediate threat to the end user.

VI. Our Contribution

In this section, we cover our solution to the current threats stated throughout our work. Conventionally, firewalls work up to the transport layer, however it is extremely difficult if not impossible, to detect and purge ad content without reading and analyzing the application layer. This is why our name is MalFire; which means it's is not a conventional firewall; rather a high-level firewall for the specific purposes of ad blocking as well as malware detection.

MalFire filters unwanted packets that contain adware that could harm an end user. When a packet does not pass validation, MalFire utilizes Iptables creating an auto-updating firewall that will drop packets or will attempt to separate the ad. Once identified the course of action will be decided.

Implementation is one of the key differences between current ad blockers and MalFire. Comparatively, when a user uses a typical ad blocker an error message might occur.

Figure 5. Ad block tool detection notification [20]

The error message will be triggered when the web server identifies an ad blocking tool at the application layer or by running a fake ad script. When an error message like Fig. 5 is displayed then the content from the web server is blocked. A user must choose to leave the site or disable ad blocking software.

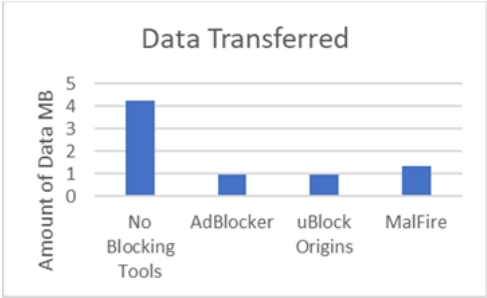


Figure 6. Data transfer with and without ad blocking tools

Currently MalFire is slightly behind in Data Transferring shown in Fig 6. AdBlocker has a lower excess data transfer barely beating out uBlock Origins. Overall all three tools prevent the occurrence of excess data transfer which can be caused by malicious content from the web servers. When malicious content is displayed it must load the image or gifs (graphic interchange format) from the ad server or from the current web server. Users who are on data plans suffer due to excess data and could face an overage fee. Once these tools remove excess data, the load speed of a website will increase causing it to display faster; benefitting users.

To that end MalFire has a faster webpage load speed than Adblocker, but uBlock Origins load speed is even faster shown in Fig 7. The utilization of Iptables by MalFire causes a reduction in the overhead. The IPs contained within these lists such as Pete Lowe's List and Malware Domain List are automatically added to the iptables and blocking these domains.

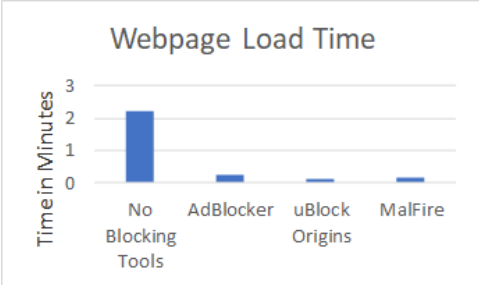


Figure 7. Webpage load page with and without ad blocking tool

VII. Acknowledgment

This work is partially supported by the TruScholars research program at Truman State University.

VIII. Conclusion And Future Work

In conclusion, our work discusses the harmful effects of adware and malicious content on the web and the security risk they pose to an end user. We backed our hypothesis claiming adware does in fact track your previous searches shown in Fig 3. Another merit is that MalFire has not been detected by any anti ad blocking tool hence supporting our claim that MalFire is undetectable by a web server. We also discussed the performance of websites on how it suffers performance loss when loading excessive data such as adware. MalFire falls behind AdBlocker and uBlock Origins in reducing excessive data transfer. In contrast, the difference when reducing data transfer without ad blocking tool is drastic. This will be further analyzed to identify what is causing the excessive data transfer. We also noticed that websites had better performance with our tool, though our data transfer was higher against other ad blockers, we were still able to achieve faster load speed against AdBlocker. Our results show that advertisements can influence website's performance. MalFire's future implementations will include a full intrusion detection system that scans for Virus, Trojans, Worms and Malware, using malware signature databases. The other future addition would be an outbound firewall that will look for unauthorized traffic, more investigation to follow.

IX. References

[1] M. M. Group, "Internet Growth Statistics Today's road to e-Commerce and Global Trade Internet Technology Reports," 2017.

[2] http://www.idc.com.

[3] http://www.internetlivestats.com/internet-users/#trend, "Internet Users," 2017.

[4] Norton Team, "The 8 Most Famous Computer Viruses of All Time," Norton, 2016.

[5] A. McNeil, "How did the WannaCry ransomworm spread?," MalwareBytes Labs, 2017.

[6] J. Segura, "FBI Ransomware Now Targeting Apple's Mac OS X Users," Malwarebytes, 2013.

[7] S. Pilici, "Remove "Your software may be out of date" pop-up virus (Removal Guide)," Malwaretips, 2017.

[8] Google, "Social Engineering (Phishing and Deceptive Sites)," 2017.

[9] P. Institute, "Spyware Study," 2005.

[10] E. Chien, "Techniques of Adware and Spyware," in *B2005 Conference*, 2017.

[11] G. Kiran, K. Orestis and M. Michael, "Ad-blocking: Study on Performance, Privacy and Counter-measures," *CoRR,* 2017.

[12] Julia Greenberg, "Ad Blockers Are Making Money Off Ads (And Tracking, Too)," *Wired,* vol. Business, 2016.

[13] E. Pujol, O. Hohlfeld and A. Feldmann, "Annyoed Usrs: Ads and Ad-Block Usage in the Wild," in *IMC '15 Proceedings of the 2015 Internet Measurement Conference* , Tokyo, 2015.

[14] https://adblockplus.org, 2017.

[15] C. E. W. a. D. C. Uzunoglu, "What Ad Blockers Are (and Are Not) Doing," Worcester, 2016.

[16] https://www.eff.org/privacybadger.

[17] G. Nakibly, J. Schcolnik and Y. Rubin, "Website-Targeted False Content Injection by Network Operators," in *USENIX Security Symposium*, Austin, 2016.

[18] M. H. Mughees, Z. Qian, Z. Shafiq and P. Hui, "Detecting Anti Ad-blockers in the Wild," in *Proceedings on Privacy Enhancing Technologies*, 2017.

[19] R. J. Walls, E. D. Kilmer, N. Lageman and P. D. McDaniel, "Measuring the Impact and Perception of," in *Internet Measurement Conference*, Tokyo, 2015.

[20] https://getadblock.com/, 2017.

[21] Tech Target, "Secure Software Quality," 2017.

[22] OSWAP, "Content Injection," 2017.

[23] https://www.ublock.org/, 2017.

[24] P. Kumar, "How to disable Opera inbuilt adblocker for a specific site.," Pcmobitech, 2017.