Journal of Analysis and Computation (JAC)
(An International Peer Reviewed Journal), www.ijaconline.com, ISSN 0973-2861
Volume XVII, Issue II, July-Dec 2023

TRACING IP ADDRESSES BEHIND VPN/PROXY SERVERS

Ms. Deepika Bansal[1], Praveen Sharma[2], Maidini Gautam[3]

[1]Assistant Professor, Jaipur Engineering College and Research Centre, Jaipur, India
[2] Jaipur Engineering College and Research Centre, Jaipur, India
[3]Jaipur Engineering College and Research Centre, Jaipur, India

ABSTRACT

Nowadays, detecting unauthorized users can be problematic for techniques that are available at present and if the nefarious actors are using identity hiding tools such as anonymizing proxies or Virtual Private Networks (VPNs). This work presents computational models to address the limitations currently experienced in detecting VPN traffic. A model to detect usage of virtual private networks (VPNs) was developed with a Multi layered perceptron neural network that was trained using flow statistics data found in the Transmission Control Protocol (TCP) header of captured network packets. Validation testing showed that the presented models are capable of classifying network traffic in a binary manner as direct (originating directly from a user's own device) or indirect (makes use of identity and location hiding features of VPNs) with high degrees of accuracy.

Keywords – Virtual Private Network(VPNs), Anonymising proxies, Transmission Control Protocol, Trained models.

[1] INTRODUCTION

Virtual Private Networks (VPNs) are a common method for criminals and other bad actors to disguise their online activities [1,2]. This is helped along by the increase in ease of use of VPNs; they are no longer just a tool for remotely accessing enterprise resources when travelling for work or when working from home. In fact, this could be a use-case for a criminal. If they wish to remotely access an enterprise network in order to steal company and trade secrets, they can use a VPN to hide their own location or to make it appear as if someone else

was infiltrating the network [3]. There have been a few notable cases of this happening in recent years, such as the Sony Pictures incident from 2014, where confidential data including personal information about employees was stolen [4, 5]. Other attacks of note are the various data breaches which have been occurring for the last number of years, such as the LinkedIn breach [6]. Approximately 167 million account details including emails and passwords were stolen. It is not known whether the attacker(s) were using a VPN service to hide their location.

We are proposing to devise and present a proxy detection methodology to protect businesses, as well as their end users, against electronic commerce (e-commerce) fraud. Knowledge gained from currently available detection methods, underlying technology, and methods of experimentation were all thoroughly considered and utilized. This proposed proxy detection methodology checks for cyber criminals who access the website and/or web application for the possibility of proxy usage and perform necessary action before cyber criminals carry out fraudulent activities.

This paper identifies different proxy connectivity methods, in order to develop a multi-tiered proxy detection module, and evaluate the implementation of the module in terms of cost andeffectiveness. Tests are completed using different types of devices and platforms, such as desktops, laptops (Windows), and mobile devices (Android). We also test the module using computers connected through home networks, work networks, and mobile networks. The results of the experiments indicate that the proxy detection module improves business security by successfully identifying proxy users.

A. Current Status of Research

Improve Caliper Solution Search Method Currently, Caliper uses a rudimentary grid search in order to find an optimal candidate solution. Future work could be conducted to research more sophisticated, and more efficient, methods of optimization. These methods could then be used on statistical geolocation algorithms developed in the future. 7.2.2 Use More Servers in Fast-Response Stage Future researchers looking to implement similar systems might consider using a larger set of Fast Response monitors in order to limit the number of RIPE Atlas credits needed. It isn't clear exactly.

RIPE stage cache entries are sane using the cheaper Fast-Response stage of the system. The RIPE stage entry should be within the region found by the Fast-Response stage – if it isn't, the servers will be in a different physical location since the cache entry was created, and the cache entry should be refreshed. 7.2.4 Trust Reputable Hosters Future researchers could consider filtering out IP addresses in netblocks owned by legitimate entities (such as Google, Digital Ocean and Amazon), who will not be using fraudulent geolocation, in order to reduce the number of lookups their system needs to do. 7.2.5 Use BGP & Routing Information BGP information – information about how packets are routed between Autonomous Systems (ASs) on the internet – could be used as additional information to use while performing geolocation fraud analysis.

B..Limitation of Current Detection Techniques

These testing methods cannot prevent an end user from performing modifications to their computer or network traffic with the intention of bypassing a configured detection method. Not all businesses will have the resources to manage and maintain secure access to all of their systems. This is especially the case when portions of the company are outsourced.Any tests that are heavily dependent on RBL databases might be prone to higher amounts of false positive results.

High quality and quantity, while keeping operations affordable. Thus, it is crucial for a water utility to implement smart water solutions to collect and analyze data in a more efficient and coherent manner. By installing IoT devices and tools to aid in data analytics, such as the tools presented in this study, the water utilities can better understand their system and, thus, optimize the operational work and teams whilst guaranteeing the service. The outputs obtained from implementing operational tools such as flowise and meter wise can be integrated in a dynamic data analysis platform (waterwise; Figure 2) to calculate key performance indicators to assess performance at operational, commercial and financial levels. This will assist water utilities in the definition of operational actions and support decisions on if, when and how to invest in equipment and activities such as network metering and censoring, customer meters, pipe rehabilitation, active detection equipment and others. Ultimately, with the use of IoT and software tools, the water utilities can improve asset management, workforce transformation and reduce NRW in a more prompt manner which will improve customer service and reduce financial losses (Ramos et al. 2020).

The technology could solve this dilemma by providing integration, as well as autonomy at the same time. So the web services technology should be deployed for multiple system environments for enabling integration for top management without breaching the autonomy of departmental units. At the same time, guidelines and motivational activities to enrich organizational readiness for such a comprehensive change activity throughout the whole organization should be proposed

[2] LITERATURE SURVEY

we aim to build a detection methodology that functions similarly to the proxy detection demonstrated on WhatIsMyIPAddress [14]. This website uses a collection of six tests to determine if a user is behind a proxy or not. One of these tests is performed using a vast collection of internal testing data that has been formatted into an identification database. As this approach is out of the scope for this project, we will focus on the identification methods that can be completed without the need of database storage and access. The remaining five tests utilize packet header analysis, various scripting techniques, and routing analysis. We will analyze these tests, along with other known methods, to accomplish our goal.

A.Research Method

The primary ideology of this paper is hoping to introduce the readers into the world of e-commerce fraud and its related proxy-based operations. Hence, various references were chosen to deliver an adequate amount of knowledge to help readers to better understand the

relevance of fraud prevention via proxy detection. Since our target audiences are mostly small to medium sized enterprises, their needs and capabilities are also taken into consideration. In order to provide complete anonymity toour test subjects, we have sanitized all the IP addresses and personal information before publication

B. Data Gathering Method

For the purpose of data gathering, we have purchased a proxy service license throughTorGuard [9]. The services provided byTorGuard allow us to test five different proxy connectivity types from hundreds of servers across the globe. We also utilized configurations that are available through free proxy lists, and alternative connectivity types such as mobile data connections and VPN tunnels. Once the proxy configurations were completed, connection attempts were made to our pre-configured server, which contains a packet loggin.

C. *Design Detection and Prevention Method*

To identify a large number of configurable proxy connection types, several steps can be used. 1) Identify the public IP address of the target machine. 2) Implement a Flash element that runs client-side and quickly reports the true public IP. 3) If the target machine's IP and the retrieved public IP match, then this test will return a value representing that no proxy was detected. However, if the IP addresses do not match, we are able to confirm that a proxy is certainly in use. Utilizing this test, we are able to positively identify any simple proxy that has been configured through a browser, or users requesting access through a web-based proxy portal.

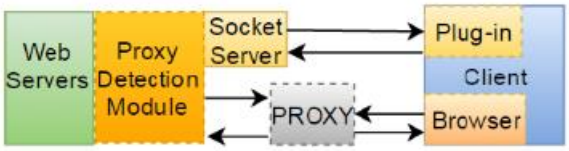

Fig. 1: Detection Architecture

Additional tests can also be made to identify stealth connections such as universal VPN services, though only to a certain degree. In order to target VPN services that our first test would not identify, we can implement further checks. 1) Reverse DNS test: Attempt to confirm the IP of a target machine through an Internet Control Message Protocol (ICMP) request, then using the resulting DNS name, verify that the connection path resolves to the same target machine and not to a local IP or a different system entirely. 2) TOR network discovery test: Identifying the majority of TOR (an anonymity network) users can be accomplished by parsing the list of publicly available TOR exit nodes, then comparing the target machine's public IP against the list. 3) RBL database test: Compare the target machine's public IP against RBL database. However, this test might not be as reliable, due to possible false positive results. In addition, all connection IP addresses are required to send to a third party service in order to use RBL database .

This paper identifies different proxy connectivity methods, in order to develop a multi-tiered proxy detection module, and evaluate the implementation of the module in terms of cost and effectiveness. Tests are completed using different types of devices and platforms, such as desktops, laptops (Windows), and mobile devices (Android). We also test the module using computers connected through home networks, work networks, and mobile networks. The results of the experiments indicate that the proxy detection module improves business security by successfully identifying proxy users

Table 1 THE DISADVANTAGES OF VPN

Disadvantages	Features
Not for anonymity	Was not designed at all for anonymity, it is a single, supplemental tool instead of a privacy solution
Unsafe	Some VPN does not provide peer-to-peer sharing which might turn in usernames to a copyright holder if required
Don't offer robust protection from ad tracking	Some cheap/ free VPNs do not protect against malware and ad trackers
Could put us at risk	Poor configuration of VPN could give direct access to hackers to our personal LANs
Pre-shared keys	People know the pre-shared key for the VPN and control the Wi-Fi access point can quickly attack ou VPN
Use the obsolete PPTP VPN protocol	The usage of the old protocol might put us in troubl
Data retention/logging	Local data storage/ cache data/cookies on user's computer can lead to loss of user's anonymity
Leakage	Some of the outgoing packets might miss the VPN tunnel, which could compromise their privacy
Marketing Hype	Users who cloaked their IP addresses unwillingly became VPN exit nodes or endpoints.

[3] EXPERIMENT RESULTS AND EVALUATION

A.Proxy Operations Analysis

In the first part of this particular investigation, we have determined that when proxy connections are created, specific characteristics that are unique to the proxy become identifiable during transmission of information, or during connection attempts. The information needed to identify a proxy can sometimes be as simple as reading the packet header containing connection type details, or checking for a matching forward and reverse DNS records, or comparing the client's IP to a RBL database. Many methods exist to identify these traits and we intend to devise a detection logic that utilizes these tests with efficiency and accuracy.

B.Proxy Detection Test

Proxy detection tests were performed in each development phase. Each test signified advancement in our detection algorithm. During testing, it was determined that the easiest

method of integration is via PHP. Further testing was performed in an attempt to utilize HTML5; however, we were unable to create a non-PHP module that is capable of operating without requiring the user to install a plug-in or add-on extension. Through the utilization of both paid proxy services and manual configuration on several system platforms, we were able to positively identify proxy users of any manually configured proxy options, or web based portals. As previously described, VPN services were more difficult to identify in a meaningful way. These VPN services can potentially be detected using one of the following methods.

- Personal computer fingerprinting and analysis of data stored in a database.
- A client-side invasive application that monitors all web traffic and ensures a secure connection to the target site (this is used by a number of banks).
- Advanced hardware technology that performs detailed packet inspection used in combination with tracking packets.

C. *Experiment Results*

Below are some of the proxy users logged by the proxy detection module, which demonstrates the result of our experiment.

TABLE 1. SELECTED^ EXPERIMENT RESULTS

Time Stamp	Connection IP	Discovered IP	Proxy Detected	RDNS Failure	Tor Check
1/31/2016 7:11	10.190.147.234	10.190.22.176	Yes	Yes	No
1/31/2016 7:12	10.107.147.234	10.107.147.234	No	No	No
1/31/2016 15:41	10.150.208.18	10.150.208.18	No	Yes	No
1/31/2016 22:34	10.164.234.138	10.164.234.138	No	No	Yes

^ Out of 811 connection attempts from 50 devices.

1) Connection one is using a simple proxy configuration, as the initial IP address differs from the one identified through the proxy detection module. It also failed the reverse DNS test.

2) Connection two passes the test as both the public and detected IP are the same. There were no detections on the remaining two tests

3) Connection three is a partial match. It fails the reverse DNS check, which can mean that they are using a misconfigured stealth VPN service, but it can also indicate that they were connecting through a mobile data service, or have disabled any ICMP requests on their firewall.

4) Connection four indicates that a TOR network connection was detected.

A.*Result Evaluation*

The proxy detection module performed its function with efficiency and effectiveness. The detection process time per client is approximately one millisecond (1 ms) plus the latency between the client and the server. The detection rate for SOCKS proxy connections is 100%. On the other hand, the detection rate for HTTP proxy connections is 94%, due to some devices disabled flash and scripts. The module is relatively straightforward to integrate into existing systems. As long as we are able to enforce the use of the Flash object on the browser, the detection of any locally configured proxy will be positively identified. Unfortunately we were unable to create a database-free methodology of identifying users utilizing advanced VPN services. Since VPN services bind to a locally created network device, the proxy detection

module will find both the public IP and the discovered IP to be the same, which renders the proxy detection module ineffective. 78 out of 80 of the VPN services that we tested through TorGuard were positively identified with a reverse DNS test. However, the reverse DNS test is vulnerable to false positives. In order to filter out the false positives, we would need to create a complex mechanism to analyze the client machines' details. A detailed fingerprint can be created from any incoming connection containing information about the computer and location [18]. The fingerprint is used to identify information about the target machine, such as local machine's country codes, language options, and regional settings. This information can then be compared against the public IP address' country of origin.

E. *Fingerprinting method*

The possibility of remotely inferring characteristics of devices has been known for some time. E.g. Kohno et al. [6] use TCP clock skew to remotely fingerprint devices. Eckersley [5] was the first to draw significant attention to the problem of fingerprinting web browsers. His fingerprinting algorithm returned a unique fingerprint for 83.6% of the browsers considered. Subsequent papers established how to fingerprint other aspects such as the user-agent string and IP address [17], the HTML5 “canvas” element [10], the used Javascript engine [9,11]), the fonts present [3], etc. Other work has suggested approaches to combine several fingerprintable attributes to arrive at unique fingerprints (e.g. [3,17]). Using such techniques only on one site does not constitute a large invasion of privacy. How-ever, as Mayer and Mitchel [8] and Roosendaal [15] have pointed out, social plugins are embedded on many pages, which allows the social network to track users across the Internet. Thanks to fingerprinting, such tracking doesn't even require an account – anyone with a unique fingerprint can be traced.

CONCLUSIONS

Proxy connections have many types and protocols, and with different software and technique configurations, it can be difficult to uncover a proxy connection. Although there are many existing methods to detect a proxy connection, all methods have their limitations. It is our goal to create a module that is capable of identifying as many proxy types as Possible. In this paper, we have investigated and tested different detection techniques, used the knowledge attained to design a multi-tiered proxy detection module, and explained how to implement the module in a business environment. With the overall detection rate of 97% and low integration cost, our proxy detection module is an effective and efficient solution for businesses to prevent fraudulent transactions from non-VPN proxy connections.

The experiments conducted to classify OpenVPN usage found that the Neural Network was able to correctly identify the VPN traffic with an overall accuracy of 93.71%. The further work done to classify StunnelOpenVPN usage found that the Neural Network was able to correctly identify VPN traffic with an overall accuracy of 97.82% accuracy when using 10-fold cross validation. This final experiment also provided an observation of 3 different validation techniques and the different accuracy results obtained. Upon successful

experiments conducted for the detection of Anonymising Proxy traffic, the focus was extended to include VPN traffic. The VPN technology OpenVPN was chosen as the focus for the experiments, which in turn found that the Neural Network was capable of classifying network traffic as either VPN traffic or as non-VPN traffic. This led to a further set of experiments which attempted to classify a form of OpenVPN traffic that made use of Stunnel to provide encryption. These found that a Neural Network trained on the StunnelOpenVPN data could classify network traffic as either VPN traffic or non-VPN traffic. Again, the experiments were conducted in such as fashion as to eliminate bias where possible. This included keeping a portion of the captured dataset away from the training and tuning phases, so it could be used to simulate real world data that the model had never seen before.

REFERENCES

[1] R.-M. Lin, Y.-C. Chou and K.-T. Chen, "Stepping Stone Detection at The Server Side," in 2011 IEEE Conference, Shangai, 2011, pp. 964-969.
[2] D. Stuttard and M. Pinto, "The Web Application Hacker's Handbook:
[3] Finding and Exploiting Security Flaws," John Wiley & Sons, 2011, pp.
[4] [3] D. Gourley and B. Totty, "Http: The Definitive Guide," O'Reilly Media, 2002, pp. 131-137.
[5] M. Ligh, S. Adair, B. Hartstein and M. Richard, "Malware Analyst's
[6] Cookbook and DVD-: Tools and Techniques for Fighting Malicious Code," John Wiley & Sons, 2010, pp. 11-15.
[7] V. Rawat, R. Tio, S. Nanji and R. Verma, "Layer Two Tunneling Protocol (L2TP) over Frame Relay," February 2001, pp. 1-3. [Online].Protocol_L2TP_over_Frame_Relay.
[8] Z. Hou, M. Xu, L. Zhu, L. Peng and B. Hu, "The Design and Realization of the Test Scheme OpenVPN, Based on Message Simulation," November 2013. [Online]. Available:
[9] K. Hamzeh, G. Pall, W. Verthein, J. Taarud, W. Little and G. Zorn, "Point-to-point tunneling protocol (PPTP)," December 1998. [Online].
[10] G. Trinder, "How SSTP based VPN connection works," Microsoft, January 2007. Available:https://blogs.technet.microsoft.com/rrasblog/2007/01/10/how-sstp-based-vpn connection-works.
[11] P. Ružicka, "Deployment of Cisco IronPort Web Security Appliance," Cisco Expo, 2009, pp. 27-31.
[12] R. Fielding, J. Gett S, J. Mogul, H. F. Nielsen, L. Masinter, P. J. Leach and T. Berners-lee, "RFC 2616: Hypertext Transfer Protocol - HTTP/1.1," December 1998, pp. 145-169.
[13] TorGuard.net, "Anonymous VPN, Proxy& Anonymous Proxy Services," 2016. [Online]. Available: https://torguard.net.
[14] [Online].Available:https://www.researchgate.net/publication/242418693_RFC_2616_Hypertext_Transfer_Protocol_-_HTTP11.