e-ISSN: 2686-2573
DOI: 10.21512/emacsjournal.v6i1.10384

An Experiment to Prevent Malicious Actors from Compromising Private Digital Assets Over a Public Network

Feliks Hartanto[1*], Budiman Budiman[2], Eldwin Gwei[3], Alexander Agung Santoso[4], Ivan Sebastian Edbert[5]

[1-5] Computer Science Department, School of Computer Science,
Bina Nusantara University,
Jakarta, Indonesia 11480
feliks.hartanto@binus.ac.id; budiman003@binus.ac.id; eldwin.gwei@binus.ac.id;
aagung@binus.edu; ivan.edbert@binus.ac.id

*Correspondence: feliks.hartanto@binus.ac.id

Abstract *— In the current millennium, human society has immensely improved its ability to obtain and distribute information. This change on the other hand, has caused the majority of daily routines to actively involve the usage of computers and mobile devices, which in turn has made people rely heavily on the availability of internet access. This fact was taken advantage of, causing a massive increase in public networks by people or businesses to draw in customers or just as simple public service. This increase gives both ease and risks which this paper will address, specifically on the security measures in network devices that are nearby, and the solution proposed to provide complementary insight on securing the technologies. The authors of this paper supply the main point of the research through experimental efforts i.e., by testing the solution in a real-life scenario. The solution itself involves the configuration of a Raspberry Pi into a VPN server and rerouting all traffic into the Raspberry server so that it will be encrypted and safe from the dangers that will be mentioned in later parts of this paper. The result of the experiment shows that the proposed solution can successfully encrypt the targeted packet so it can't be read by malicious attackers. Although the solution works it can't be simply applied to every public network due to internet connection protocols and its inconvenience. Future research will involve the improvement or rework of the solution until the issues mentioned above are solved.*

Keywords: *Public Network; Digital Fafety; OpenVPN; Raspberry Pi; Traffic Encryption*

I. INTRODUCTION

In this current era of modernization, none can argue that the current lifestyle of the average human is bound to the internet, be it communicating with people, utilities, and entertainment services. As such, people have come to need reliable and easily available access points to be able to surf the internet. This is where public Wi-Fis comes into play on how they provide convenient internet access in certain public spaces such as cafes, restaurants, and learning institutes (Maimon et. al, 2017). However, using these public Wi-Fis exposes users to various potential risks that compromises their private digital assets.

Based on the reviews done on similar research, the current statistics show an increasing spread of public Wi-Fi worldwide i.e., 94 million in 2021 and is expected to rise to 549 million by the end of 2022. This dramatic increase in number is caused by the massive number of new computing devices each year (Lotfy et. al, 2021). Therefore, a change of perspective regarding mobile security can be expected in the following years as a result of combating the dangers of computer vulnerabilities that occur when utilizing public networks (Breitinger et. al, 2020).

Throughout the reviews done for this paper, a few possible threats over browsing requests done on public Wi-Fi has been enlisted, such as internet-based services, email, social networking, ecommerce site, and many more. Such dangers include exposing sensitive data like credit card information, credentials, activities, etc. Said potential attacks can occur due to either vulnerable access points or a disguised attacker network imitating the real public Wi-Fi (Choi et. al, 2021).

The 2nd section after the introduction consists of the literature review that focuses on the definition of networks and discovering possible dangers that come along with public networks. In the 3rd section i.e., methodology, this paper proposes a solution which involves configuring a Raspberry Pi as a VPN server and routing all traffic into it, which as a result encrypts and protects any sensitive data from being intercepted. Following the proposal of the solution, the methodology also explains what kind of experiment was used to prove the solutions' effectiveness and an explanation regarding what parts/tools were used in the experiment. The last 2 sections of this paper will respectively be about the results of the experiment and the conclusion of this paper.

Before moving any further, there is a question that must be asked "what is a network?" Specifically, a computer network. The definition is that it's a distributed system that consists of multiple computers and other devices that are loosely coupled, which results in said devices gaining the ability to communicate with one another as long as they are following the set of rules or communication protocols configured in the network. Computer networks can be categorized based on their coverage or size (Kizza, 2020):

- Local Area Network (LAN)
- Metropolitan Area Network (MAN)
- Wide Area Network (WAN)

Wi-Fi is included in the LAN category, reason being LAN is a network that is confined to a small geographic region that has more than one computer or device, and their resources connected by a communication medium that's using the same communication protocols (Kizza, 2020). Then why is Wi-Fi included in LAN? Wi-Fi itself stands for Wireless Fidelity which is one of a kind in the family of network protocols authorized by the Institute of Electrical and Electronics Engineers (IEEE), specifically article 802.11 published in 1997. Wi-Fi has the capability to exchange information between devices across radio waves, thus eliminating the need to stretch long lines of cable from room to room, therefore its practicality being the reason it powers most of the LAN in modern computer practice. Creating a LAN subdivision where the communication medium uses high-frequency waves instead of cables called Wireless LAN (WLAN) (Fikriyadi et. al, 2020).

In the past 10 years, mobile devices have been growing rapidly at a speed unthought before. The advent of smartphones would not exactly be as great as they are if it were not for the invention of fast and reliable internet broadbands, thus much of the credit goes to Wi-Fi protocols and their improving implementation over the years (Jain et. al, 2019).

However, there are still dangers/risks that remain when using random public Wi-Fi. To understand how hotspots/public Wi-Fis expose consumers to a whole array of privacy risks, it is important to understand what defines a hotspot; "a hotspot is any location where [wireless internet access] is made publicly available" (Shahin, 2017).

According to a Symantec study, results show that 46% of the 15,532 users will immediately connect to a public Wi-Fi network and 60% of them believe that their data/personal information isn't at risk while using said network. Hotspot access is usually deployed in three forms, such as the following (Ali et. al, 2019):

- Captive portals
- Direct/open access (no captive portals)
- Password-protected networks.

Open access and password-protected networks are self-explanatory, Captive portals on the other hand are hotspots where the user would be directed to a webpage that usually contains the hotspot/Wi-Fi providers' privacy policy and/or their Term of Service (TOS), and a login page where users can register their personal data I.e., name and email, to be granted access to the internet services of the Wi-Fi (O'hanlon et. al, 2017).

The convenience of free internet is but a two-edged blade if one is not wary, the dangerous aspects include being prone to certain hacks that are usually done to collect personal information such as (Shahin, 2017):

1. Sniffing: Sniffing "allows hackers to passively intercept data sent between a web browser and web servers on the Internet."
2. Evil twins: "(a)n evil twin is a rogue Wi-Fi access point that appears to be legitimate but actually has been set up by a hacker to fool wireless users into connecting a laptop or mobile phone to a tainted hotspot." Typically, mobile devices will scan for available networks and connect automatically to certain Wi-Fi with known SSIDs that the device has connected to once before, even without human interactions. Thus, a rogue Wi-Fi could reuse common SSID names publicly available and so various information / requests will be transmitted over the malicious spoofer unknowingly. This often leads to leakage of private attributes such as banking, identity, affiliation, etc (Maimon et. al, 2017).
3. Man-in-the-middle-attacks: "intercept and modify" data going between the user and the hotspot server. Throughout the histories and origins of current networking technologies, there has yet been any protocol worthy of the achievement of being completely resistant to MITM attacks on its own i.e., no administrative action needed (Schofield, 2019). This is such a concern since every single company / institute will either route their network properly, not at all, or in between. Such inconsistencies are keys for perpetrators to take advantage of. Common MITM attacks include spoofing based, TSL/SSL based, BGP based, and FBS based (Bhushan et. al, 2017).
4. Side jacking: an attack that uses a program that can "intercept or log traffic passing over a digital network, to steal a session cookie containing usernames and passwords from a variety of websites, such as Facebook or LinkedIn."

II. METHODS

For this section, the qualitative method will be used to analyze the effectiveness of the proposed solution by comparing the packets that are sent to the router when using a public Wi-Fi with and without the solution. The proposed solution involves utilizing a lightweight, network capable device such as the Raspberry PI, specifically the Raspberry PI 3 model B as a redirecting medium that provides full encryption of traffic coming in and out of the local network space while the interface socket connected to the public Wi-Fi router is only utilized to retain connectivity and routing.

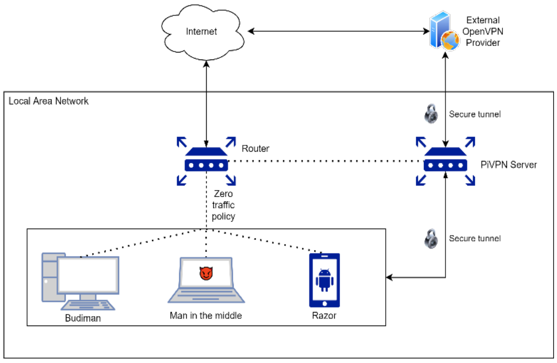


Figure 1. Mitigation flow chart

The experiment will be done on a website used by residents of a certain apartment, said website is used to find information related to their stay such as laundry quotas, whether a package has arrived, etc. To use this website, the residents are required to login using their IDs and password which will be the target/key of this experiment. We will be comparing the visibility of the target while using a public network that's applying the solution and a network that isn't, the results will be recorded with a packet capture of the network using a network analyzer tool called Wireshark.

The following will be the explanation of the parts used for the solution:

2.1 Server

For this experiment, the Raspberry Pi 3 Model B was used. A raspberry Pi is a small-single board computer commonly used for general-purpose computer development. The Raspberry Pi 3 Model B is the latest version of the Raspberry Pi computer and the most powerful among the other models of Raspberry Pi (Taib et. al, 2019) this technology discloses the user to the security threat. Barely users are sensitive that their data are being monitored by Internet Service Providers (ISPs).

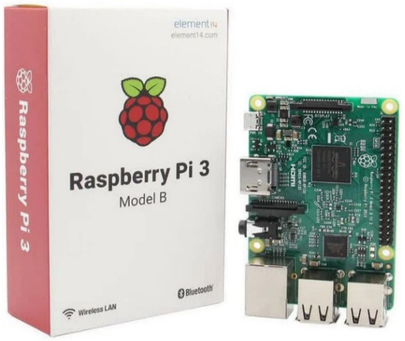


Figure 2. Raspberry Pi 3 Model B

2.2 Traffic encryption or virtual private networks (VPN)

The virtual private network (VPN) be it the service or client that were used for this experiment are as follows (Karaymeh et. al, 2019), (Iqbal & Riadi, 2019):

2.2.1 OpenVPN

Runs as a network service on the medium device to enable encrypted communications between devices. In this specific demonstration, the supported version for the Raspberry Pi is reasonably named PiVPN. The PiVPN was implemented with a self-triggered basis on each startup so that it continuously serves connectivity to the local devices while actively routing public connections in and out through trusted VPN services. With this setup, any destined network packet sent either way will be resistant against prying actors whether locally or throughout the globe.

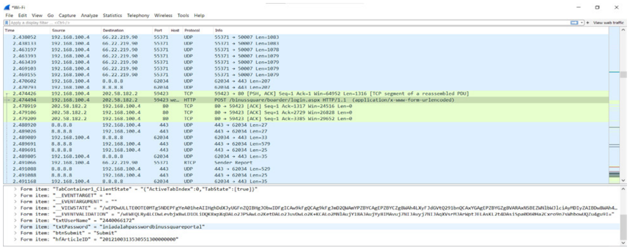
Figure 3. OpenVPN

2.2.2 OpenVPN client

As opposed to the previously mentioned PiVPN, the client-side service of OpenVPN compliments the protocol by enabling any devices to easily load-and-go connections with any configuration files generated by the server. This enables any daily kits like the smartphone, laptops, TVs, etc. to be a member of the PiVPN network with a simple tap on the 'connect' button after the initial configuration. Once connected to the secure tunnel, any packet sent and received by the particular device should not be readable by none other than the source / destination peer as the right encryption key is required to reconstruct the integrity.

III. RESULTS AND DISCUSSION

For the traffic analysis part of the experiment, Wireshark was used. Below, the authors showcase the difference between conventional routing of local traffic and the one implementing the proposed solution.

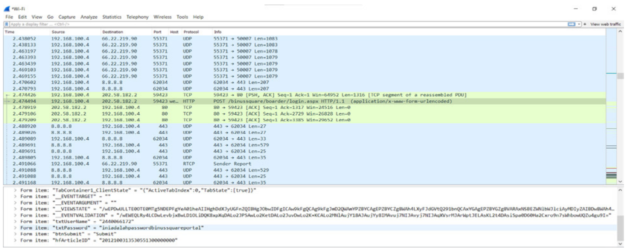

*Wi-Fi
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
Apply a display filter ... <Ctrl-/>
View web traffic

```
Time        Source           Destination      Port   Host  Protocol  Info
2.438052    192.168.100.4    66.22.219.90     55371        UDP       55371 → 50007 Len=1083
2.438133    192.168.100.4    66.22.219.90     55371        UDP       55371 → 50007 Len=1083
2.463197    192.168.100.4    66.22.219.90     55371        UDP       55371 → 50007 Len=1078
2.463393    192.168.100.4    66.22.219.90     55371        UDP       55371 → 50007 Len=1079
2.463439    192.168.100.4    66.22.219.90     55371        UDP       55371 → 50007 Len=1079
2.469103    192.168.100.4    66.22.219.90     55371        UDP       55371 → 50007 Len=1079
2.469155    192.168.100.4    66.22.219.90     55371        UDP       55371 → 50007 Len=1079
2.470602    192.168.100.4    8.8.8.8          62034        UDP       62034 → 443 Len=207
2.470793    192.168.100.4    8.8.8.8          62034        UDP       62034 → 443 Len=207
2.474426    192.168.100.4    202.58.182.2     59423        TCP       59423 → 80 [PSH, ACK] Seq=1 Ack=1 Win=64952 Len=1316 [TCP segment of a reassembled PDU]
2.474494    192.168.100.4    202.58.182.2     59423  we…   HTTP      POST /binussquare/boarder/login.aspx HTTP/1.1  (application/x-www-form-urlencoded)
2.478919    202.58.182.2     192.168.100.4       80        TCP       80 → 59423 [ACK] Seq=1 Ack=1317 Win=24516 Len=0
2.479106    202.58.182.2     192.168.100.4       80        TCP       80 → 59423 [ACK] Seq=1 Ack=2729 Win=26828 Len=0
2.479209    202.58.182.2     192.168.100.4       80        TCP       80 → 59423 [ACK] Seq=1 Ack=3385 Win=29652 Len=0
2.488920    8.8.8.8          192.168.100.4      443        UDP       443 → 62034 Len=27
2.489026    8.8.8.8          192.168.100.4      443        UDP       443 → 62034 Len=27
2.489089    192.168.100.4    8.8.8.8          62034        UDP       62034 → 443 Len=33
2.489691    8.8.8.8          192.168.100.4      443        UDP       443 → 62034 Len=579
2.489691    8.8.8.8          192.168.100.4      443        UDP       443 → 62034 Len=25
2.489805    192.168.100.4    8.8.8.8          62034        UDP       62034 → 443 Len=35
2.491066    192.168.100.4    66.22.219.90     55371        RTCP      Sender Report
2.491088    8.8.8.8          192.168.100.4      443        UDP       443 → 62034 Len=529
2.491088    8.8.8.8          192.168.100.4      443        UDP       443 → 62034 Len=25
2.491168    192.168.100.4    8.8.8.8          62034        UDP       62034 → 443 Len=35

> Form item: "TabContainer1_ClientState" = "{"ActiveTabIndex":0,"TabState":[true]}"
> Form item: "__EVENTTARGET" = ""
> Form item: "__EVENTARGUMENT" = ""
> Form item: "__VIEWSTATE" = "/wEPDwULLTE0OTE0MTg5NDEPFgYeA01heAIIHghDdXJyUGFnZQIBHgJObwIDFgICAw9kFgQCAg9kFgJmD2QWAmYPZBYCAgEPZBYCZg8WAh4LXyFJdGVtQ291bnQCAxYGAgEPZBYGZg8VARAxNSBEZWNlbWJlciAyMDIyZAIBDw8WAh4…
> Form item: "__EVENTVALIDATION" = "/wEWEQLRy4LCDwLevbjxBwLD1OLiDQK8xpXqDALo2JPSAwLo2KetDALo2JuvDwLo2K+KCALo2MNlAujY18AJAujYy8IMAvuj7NIJAvyj7NIJAqXVsrMJArWptJELAsKL2t4DAsiSpa0D60Ha2Cxro9n7sWhbowUQZu4gu9I="
> Form item: "txtUserName" = "2440066172"
> Form item: "txtPassword" = "iniadalahpasswordbinussquareportal"
> Form item: "btnSubmit" = "Submit"
> Form item: "hfArticleID" = "20121003135305513000000000"
```

Figure 4. Wireshark capture without the solution

By regular means, it is possible for a third party to sniff packets within a local network as they could spoof the network to believe that all the traffics are meant for a particular device, in this case, by poisoning the ARP cache. Thus, a malicious perpetrator would be able to monitor the entire pool. In this case, as seen from Figure 4 it is possible to find a certain packet from the targeted network that contains the POST request sent to login.aspx.

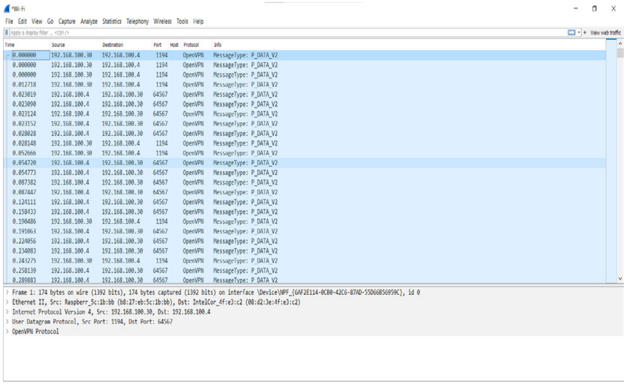

*Wi-Fi
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
Apply a display filter ... <Ctrl-/>
View web traffic

```
Time        Source          Destination     Port      Host  Protocol  Info
2.438052    192.168.100.4   66.22.219.90    55371           UDP       55371 → 50007 Len=1083
2.438133    192.168.100.4   66.22.219.90    55371           UDP       55371 → 50007 Len=1083
2.463197    192.168.100.4   66.22.219.90    55371           UDP       55371 → 50007 Len=1078
2.463393    192.168.100.4   66.22.219.90    55371           UDP       55371 → 50007 Len=1079
2.463439    192.168.100.4   66.22.219.90    55371           UDP       55371 → 50007 Len=1079
2.469103    192.168.100.4   66.22.219.90    55371           UDP       55371 → 50007 Len=1079
2.469155    192.168.100.4   66.22.219.90    55371           UDP       55371 → 50007 Len=1079
2.470602    192.168.100.4   8.8.8.8         62034           UDP       62034 → 443 Len=207
2.470793    192.168.100.4   8.8.8.8         62034           UDP       62034 → 443 Len=207
2.474426    192.168.100.4   202.58.182.2    59423           TCP       59423 → 80 [PSH, ACK] Seq=1 Ack=1 Win=64952 Len=1316 [TCP segment of a reassembled PDU]
2.474494    192.168.100.4   202.58.182.2    59423 we…       HTTP      POST /binussquare/boarder/login.aspx HTTP/1.1  (application/x-www-form-urlencoded)
2.478919    202.58.182.2    192.168.100.4      80           TCP       80 → 59423 [ACK] Seq=1 Ack=1317 Win=24516 Len=0
2.479106    202.58.182.2    192.168.100.4      80           TCP       80 → 59423 [ACK] Seq=1 Ack=2729 Win=26828 Len=0
2.479209    202.58.182.2    192.168.100.4      80           TCP       80 → 59423 [ACK] Seq=1 Ack=3385 Win=29652 Len=0
2.488920    8.8.8.8         192.168.100.4     443           UDP       443 → 62034 Len=27
2.489026    8.8.8.8         192.168.100.4     443           UDP       443 → 62034 Len=27
2.489089    192.168.100.4   8.8.8.8         62034           UDP       62034 → 443 Len=33
2.489691    8.8.8.8         192.168.100.4     443           UDP       443 → 62034 Len=579
2.489691    8.8.8.8         192.168.100.4     443           UDP       443 → 62034 Len=25
2.489805    192.168.100.4   8.8.8.8         62034           UDP       62034 → 443 Len=35
2.491066    192.168.100.4   66.22.219.90    55371           RTCP      Sender Report
2.491088    8.8.8.8         192.168.100.4     443           UDP       443 → 62034 Len=529
2.491088    8.8.8.8         192.168.100.4     443           UDP       443 → 62034 Len=25
2.491168    192.168.100.4   8.8.8.8         62034           UDP       62034 → 443 Len=35

> Form item: "TabContainer1_ClientState" = "{"ActiveTabIndex":0,"TabState":[true]}"
> Form item: "__EVENTTARGET" = ""
> Form item: "__EVENTARGUMENT" = ""
> Form item: "__VIEWSTATE" = "/wEPDwULLTE0OTE0MTg5NDEPFgYeA01heAIIHghDdXJyUGFnZQIBHgJObwIDFgICAw9kFgQCAg9kFgJmD2QWAmYPZBYCAgEPZBYCZg8WAh4LXyFJdGVtQ291bnQCAxYGAgEPZBYGZg8VARAxNSBEZWNlbWJlciAyMDIyZAIBDw8WAh4…
> Form item: "__EVENTVALIDATION" = "/wEWEQLRy4LCDwLevbjxBwLD1OLiDQK8xpXqDALo2JPSAwLo2KetDALo2JuvDwLo2K+KCALo2MNlAujY18AJAujYy8IMAvuj7NIJAvyj7NIJAqXVsrMJArWptJELAsKL2t4DAsiSpa0D60Ha2Cxro9n7sWhbowUQZu4gu9I="
> Form item: "txtUserName" = "2440066172"
> Form item: "txtPassword" = "iniadalahpasswordbinussquareportal"
> Form item: "btnSubmit" = "Submit"
> Form item: "hfArticleID" = "20121003135305513000000000"
```

Figure 5. Sensitive information in intercepted packets

Although some packet transmission protocol comes with its own measure in terms of security, some services still address their availability by primitive methods, such as HTTP. As seen in Figure 5, following is the content of a particular request coming from a local device. The request appears to be a login probe to a web server, specifically a resident web platform to administer information regarding their stay. Into the request body, the victim's username and password can be clearly seen in simple plain text. Thus, this goes to show how an open public network could leak their users' sensitive personal data.


*Wi-Fi
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
Apply a display filter ... <Ctrl-/>
View web traffic

```
Time        Source          Destination     Port   Host  Protocol  Info
0.000000    192.168.100.30  192.168.100.4    1194        OpenVPN   MessageType: P_DATA_V2
0.000000    192.168.100.30  192.168.100.4    1194        OpenVPN   MessageType: P_DATA_V2
0.000000    192.168.100.30  192.168.100.4    1194        OpenVPN   MessageType: P_DATA_V2
0.012718    192.168.100.30  192.168.100.4    1194        OpenVPN   MessageType: P_DATA_V2
0.023019    192.168.100.4   192.168.100.30  64567        OpenVPN   MessageType: P_DATA_V2
0.023090    192.168.100.4   192.168.100.30  64567        OpenVPN   MessageType: P_DATA_V2
0.023124    192.168.100.4   192.168.100.30  64567        OpenVPN   MessageType: P_DATA_V2
0.023152    192.168.100.4   192.168.100.30  64567        OpenVPN   MessageType: P_DATA_V2
0.028028    192.168.100.4   192.168.100.30  64567        OpenVPN   MessageType: P_DATA_V2
0.028148    192.168.100.30  192.168.100.4    1194        OpenVPN   MessageType: P_DATA_V2
0.052666    192.168.100.30  192.168.100.4    1194        OpenVPN   MessageType: P_DATA_V2
0.054720    192.168.100.4   192.168.100.30  64567        OpenVPN   MessageType: P_DATA_V2
0.054773    192.168.100.4   192.168.100.30  64567        OpenVPN   MessageType: P_DATA_V2
0.087382    192.168.100.4   192.168.100.30  64567        OpenVPN   MessageType: P_DATA_V2
0.087447    192.168.100.4   192.168.100.30  64567        OpenVPN   MessageType: P_DATA_V2
0.124111    192.168.100.4   192.168.100.30  64567        OpenVPN   MessageType: P_DATA_V2
0.158433    192.168.100.4   192.168.100.30  64567        OpenVPN   MessageType: P_DATA_V2
0.190486    192.168.100.30  192.168.100.4    1194        OpenVPN   MessageType: P_DATA_V2
0.191063    192.168.100.4   192.168.100.30  64567        OpenVPN   MessageType: P_DATA_V2
0.224056    192.168.100.4   192.168.100.30  64567        OpenVPN   MessageType: P_DATA_V2
0.234083    192.168.100.4   192.168.100.30  64567        OpenVPN   MessageType: P_DATA_V2
0.243275    192.168.100.30  192.168.100.4    1194        OpenVPN   MessageType: P_DATA_V2
0.258139    192.168.100.4   192.168.100.30  64567        OpenVPN   MessageType: P_DATA_V2
0.289883    192.168.100.4   192.168.100.30  64567        OpenVPN   MessageType: P_DATA_V2

> Frame 1: 174 bytes on wire (1392 bits), 174 bytes captured (1392 bits) on interface \Device\NPF_{6AF2E114-0CB0-42C6-87AD-55D66B56959C}, id 0
> Ethernet II, Src: Raspberr_5c:1b:bb (b8:27:eb:5c:1b:bb), Dst: IntelCor_4f:e3:c2 (08:d2:3e:4f:e3:c2)
> Internet Protocol Version 4, Src: 192.168.100.30, Dst: 192.168.100.4
> User Datagram Protocol, Src Port: 1194, Dst Port: 64567
> OpenVPN Protocol
```

Figure 6. Wireshark capture with the solution

However, with the presented solution the packet capture that can be seen from Figure 6, shows that the captured packets are encrypted and as a result the attacker will be unable to read the contents of the traffic. Thus, the solution can be seen as successful as it eliminates the potential risk when using a public network or to at least minimize it as an option for those who requires it.

IV. CONCLUSION

From the results, the proposed solution works as intended, however the current internet connectivity protocol doesn't easily allow a straightforward application of the proposed solution, such that it will nevertheless require a network user to install the OpenVPN client and import its configuration manually on the first connection. Therefore, the solution itself can't be imposed by default, which might be a matter of convenience.

For future approach to the same problem, the authors suggest user activity as the first in consideration, as it could greatly affect how quickly the researchers could identify a convenient protocol, i.e., one that is natively available across platforms so to integrate the solution seamlessly, eliminating user's obligation to follow specific instructions which is of no inherent cognitive memory they have built alongside the long-term usage of their mobile technologies so far. On the other hand, it could be very challenging nevertheless as known protocols are persistent to their standards and are not expected to easily approve of new patches, especially in short term.

REFERENCES

Ali, S., Osman, T., Mannan, M., & Youssef, A. (2019). On privacy risks of public wifi captive portals. In Data Privacy Management, Cryptocurrencies and Blockchain Technology: ESORICS 2019 International Workshops, DPM 2019 and CBT 2019, Luxembourg, September 26–27, 2019, Proceedings 14 (pp. 80-98). Springer International Publishing. doi: 10.1007/978-3-030-31500-9_6.

Bhushan, B., Sahoo, G., & Rai, A. K. (2017, September). Man-in-the-middle attack in wireless and computer networking—A review. In 2017 3rd International Conference on Advances in Computing, Communication & Automation (ICACCA) (Fall) (pp. 1-6). IEEE. doi: 10.1109/ICACCAF.2017.8344724.

Breitinger, F., Tully-Doyle, R., & Hassenfeldt, C. (2020). A survey on smartphone user's security choices, awareness and education. Computers & Security, 88, 101647. doi: 10.1016/j.cose.2019.101647.

Choi, H. S., Carpenter, D., & Ko, M. S. (2021). Risk taking behaviors using public Wi-fi™. Information Systems Frontiers, 1-18. doi: 10.1007/s10796-021-10119-7.

Fikriyadi, F., Ritzkal, R., & Prakosa, B. A. (2020). Security Analysis of Wireless Local Area Network (WLAN) Network with the Penetration Testing Method. Jurnal Mantik, 4(3), 1658-1662. [Online]. Available: https://iocscience.org/ejournal/index.php/mantik

Iqbal, M., & Riadi, I. (2019). Analysis of security virtual private network (VPN) using openVPN. International Journal of Cyber-Security and Digital Forensics, 8(1), 58-65. doi: 10.17781/p002557.

Jain, S., Bensaid, E., & de Montjoye, Y. A. (2019, May). UNVEIL: capture and visualise WiFi data leakages. In The World Wide Web Conference (pp. 3550-3554). doi: 10.1145/3308558.3314143.

Karaymeh, A., Ababneh, M., Qasaimeh, M., & Al-Fayoumi, M. (2019, October). Enhancing data protection provided by VPN connections over open WiFi networks. In 2019 2nd International

Conference on new Trends in Computing Sciences (ICTCS) (pp. 1-6). IEEE. doi: 10.1109/ICTCS.2019.8923104.

Kizza, J. Migga. (2020). Guide to Computer Network Security (Texts in Computer Science) 5th ed. 2020 Edition. Publisher: Springer. ISBN: 978-3030381400.

Lotfy, A. Y., Zaki, A. M., Abd-El-Hafeez, T., & Mahmoud, T. M. (2021, May). Privacy Issues of Public Wi-Fi Networks. In The International Conference on Artificial Intelligence and Computer Vision (pp. 656-665). Cham: Springer International Publishing. doi: 10.1007/978-3-030-76346-6_58.

Maimon, D., Becker, M., Patil, S., & Katz, J. (2017). {Self-Protective} Behaviors Over Public {WiFi} Networks. In The LASER Workshop: Learning from Authoritative Security Experiment Results (LASER 2017) (pp. 69-76).

O'hanlon, P., Borgaonkar, R., & Hirschi, L. (2017, May). Mobile subscriber wifi privacy. In 2017 IEEE Security and Privacy Workshops (SPW) (pp. 169-178). IEEE. doi: 10.1109/SPW.2017.14.

Schofield, G. (2019). Has your wifi left you wide open to cybercrime?. Network Security, 2019(3), 13-14. doi: 10.1016/S1353-4858(19)30036-4.

Shahin, E. (2017). Is Wifi Worth It: The Hidden Dangers Of Public Wifi. Catholic University Journal Of Law And Technology, 25(1), 7. [Online]. Available: http://scholarship.law.edu/cgi/viewcontent.cgi?article=1023&context=jlt

Taib, A. M., Ishak, M. F. H., & Kamarudin, N. K. (2020). Securing network using raspberry Pi by implementing VPN, Pi-hole, and IPS (VPiSec). International Journal, 9(1.3). doi: 10.30534/ijatcse/2020/7291.32020.