[1]

Examining the Anonymity, Privacy, and Security of Scalable Anonymous Disposable Desktops (SADD)

Desmond Jackson, CS428, dajackson4@crimson.ua.edu,
December, 2019

***Abstract*—This report will be about how SADD provides Anonymity, Privacy, and Security to its users. The purpose of the report is to review, deduce possible vulnerabilities, and ultimately test how SADD's Anonymity, Privacy, and Security could be exploited or otherwise made null and void. Upon completion of this report, I was able to thoroughly explain how SADD protects its users and how a user could compromise oneself through simple mistakes. I was able to spin up a SADD desktop in the browser, use it, and test whether or not the claims of Anonymity, Privacy, and Security were true.**

I. Introduction

WHEN taking a look at the SADD website, it is explained that this technology is simply just a web-based virtual machine, that is routed through the Tor network, and is forensically destroyed at the end of the user session.

[1] This technology appears to have quite a few possible use cases, but the most notable are:

1) Giving Ethical Hackers the resources to anonymously report vulnerabilities in the wild without risking identifying themselves.
2) Employees of companies can use SADD without risking the integrity of their business' security model.
3) Government Entities are now able to engage enemies through Cyber means without any form of identifiable responsibility or culpability.

Before a user can generate a desktop the user must first either email sadd.io support or purchase an access token. The access token must then be entered into the window shown below:

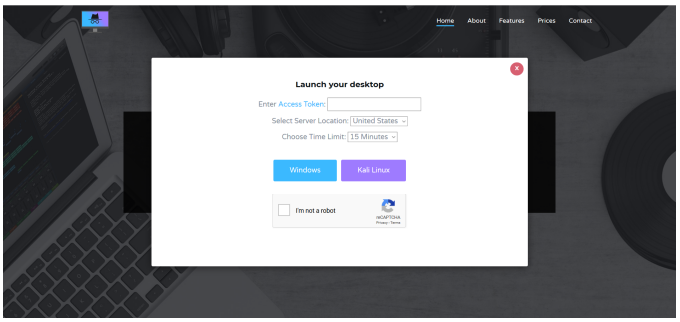

Fig.1 - SADD.IO Launch Panel.

The access token can be purchased via Paypal, Credit Card, or even cryptocurrency. Once the access token has been entered into the above box, the user is enabled to select Server Location, a Time-Limit up to 60 minutes, and select an Operating System between Windows and Kali Linux.

One the Operating System is selected the user is given a loading screen as shown below:

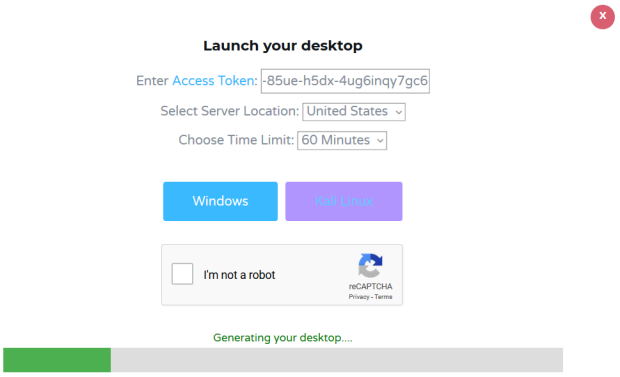

Fig. 2 - Launching a Kali Linux Desktop.

The user is finally redirected to a new screen where the user is able to remote control a securely generated virtual machine as shown in the following image:

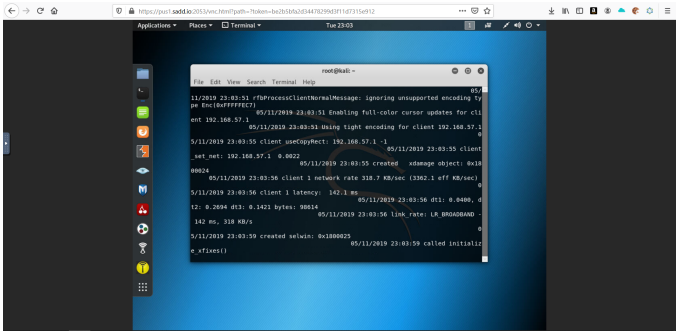

Fig. 3 A freshly generated Kali Linux Desktop.

II. ANONYMITY

A. *Review*

[2] An article that was published by The Hackers Online Club, discusses SADD in great detail.

It states that SADD utilizes the Tor network to maintain user anonymity. [1] This is also supported by the YouTube

[1]Manuscript received November 6, 2019. This work was supported in part by the patent information found on SADD.

video found on the sadd.io website. It is claimed that every network operation that is conducted by the virtual desktop is routed through Tor to maintain user anonymity.

The diagram below shows how SADD routes the network connections:

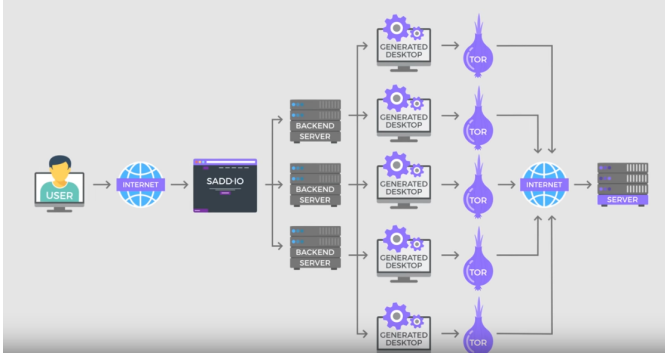

Fig. 4 - SADD.IO Youtube Video Diagram.

Images taken from the patent also show a similar diagram in more detail than the YouTube video:

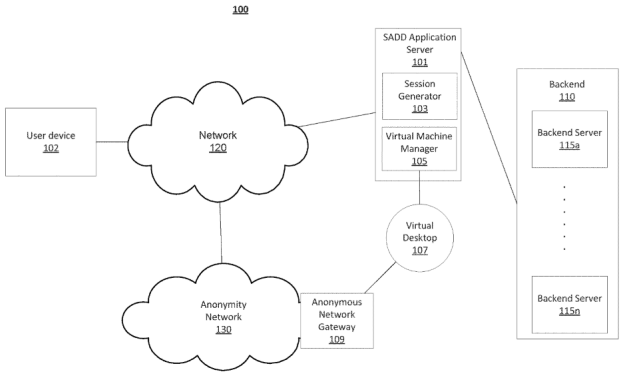

Fig. 5 - SADD Patent Diagram

[3] Both diagrams above show that the user device controls a virtual machine through the web. These virtual machines are generated on some backend servers. Each virtual machine is connected to *the anonymity network*. In this instance, the anonymity network is defined as Tor.

The most notable feature from the patent diagram is that the user never directly connects to the virtual machine. Meaning that the virtual machine is controlled by means of remote desktop over the browser.

Another aspect of SADD that provides anonymity, is the fact that users generate and access these desktops via an access token. This access token has no personally identifiable information (PII) corresponding with it from appearance. If the user is worried that using a credit card or PayPal could personally identify them, the user has the option to purchase an access token over Tor and through cryptocurrency simultaneously.

B. Deducing

The first point of deducing vulnerability to exploit the Anonymity of SADD is that the access token if purchased without the aid of an anonymity network or cryptocurrency could identify a person. Especially if the service was subpoenaed for all access tokens that correspond with PII. After all the access tokens are provided by a third party called Selly.

The second point of deducing possible vulnerability involves checking for vulnerabilities such as DNS leaks, IP Address leaks, and Operating System fingerprinting.

The third point of deducing possible vulnerability involves the exploitation of the Tor network via correlation and Man-In-The-Middle attacks.

The fourth and final point of deducing possible vulnerability involves the user performing personal identifying actions such as logging into Facebook. Or maybe the user may browse to a specific website and area at a specific time.

C. Testing

In order to test the first point of deducing possible vulnerability, I looked at the third party website that SADD used to sell and store access tokens, https://selly.io. Selly's privacy policy states that Selly collects PII including but not limited to the last 4 digits of credit card, physical address, IP Address, email addresses, and name. If this information is subpoenaed and the user did not take measures to use an anonymity network and cryptocurrency the user becomes easily identifiable.

To test the second point of deducing possible vulnerability I first checked to see if the virtual desktop was indeed connected to the Tor network. On the virtual machine's browser I navigated to https://check.torproject.org. As shown in the screenshot below, it is connected to the Tor network:

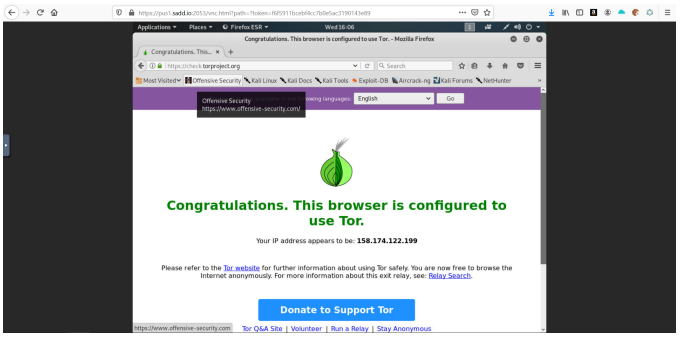

Fig. 6 - Showing the generated Desktop is routed through Tor.

The screenshot, however, shows that the browser is at least configured to connect to the Tor network. I then went to terminal and typed the following command to obtain the public IP address from this machine: curl https://ipinfo.io/ip

Once that command was executed the same IP address from the previous screenshot appears in the terminal:

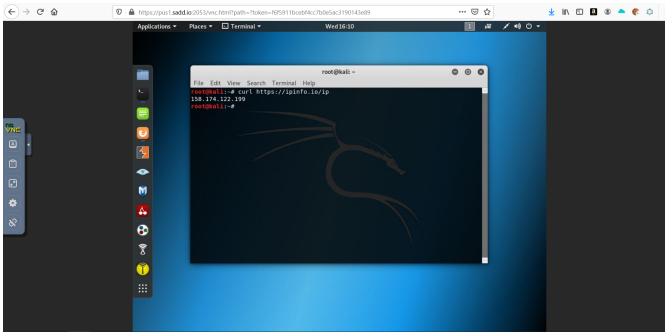

Fig. 7 - Obtaining the Public IP address of the Kali Linux Desktop.

From the above screenshot it is feasible to assume the Operating System is indeed routed through Tor.

To test for DNS leaks, I navigated to http://dnsleak.com on the virtual machine's browser. The website returned information showing that whatever Tor node that the virtual machine was connected to used a Cloudflare based or protected DNS server:

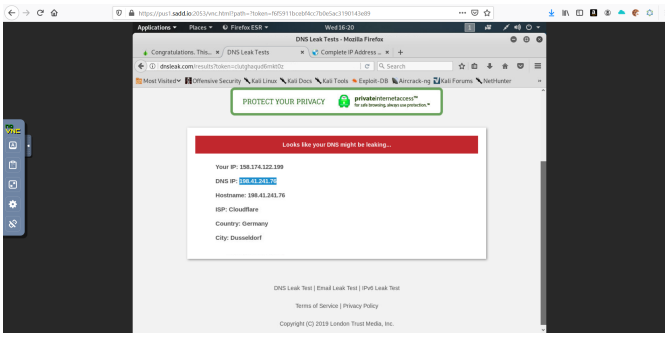

Fig. 8 - Testing dnsleak.com on the Kali Linux Desktop.

This could be either good or bad depending on who owns the DNS server. If an entity were able to gain access to the DNS server and monitor the entity could see what sites a user is visiting. This is a step in the right direction in the identification of a user, but it is not the ultimate way since the virtual machine does utilize the Tor network. A new node means a new DNS server. The virtual machine's public IP address changes quite often, as it should.

The last step in testing the second point of deducing possible vulnerability is looking at Operating System fingerprinting. Since SADD generates virtual machines on a clean slate, all virtual machines start out with the same fingerprint. [5] I ran the following command in the terminal to fingerprint the virtual machine: nmap -A localhost

The screenshot below shows the results:

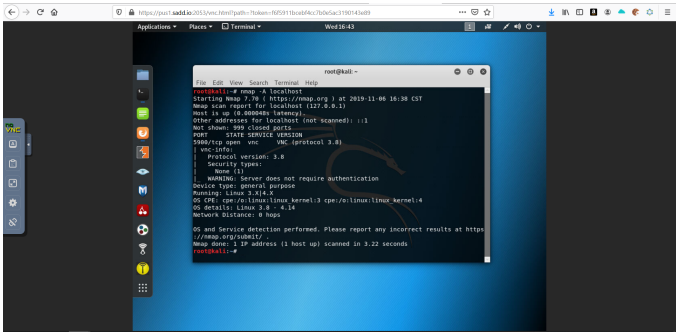

Fig. 9 - Using nmap on the Kali Linux Desktop.

On this particular VM we can see that it is running a VNC server. This would be hard to access through the Tor network since this service is not port forwarded on our Public IP address. It also says that the virtual machine is running a version of Linux between 3 - 4.14. If this version is not the most up-to-date, it could be a risk to a user if a Current Vulnerability and Exposure (CVE) exists for these Linux versions on https://cvedetails.com. The CVE details website will give instructions on how to execute the vulnerabilities if any exist. If the vulnerability can be executed remotely or via the user clicking a link or opening a file that would be detrimental to maintaining anonymity. However, if there exists a zero-day vulnerability that can be executed remotely or via the user clicking a link or opening a file, maintaining anonymity is still at risk. Browser Fingerprinting could also be used in conjunction with Operating System Fingerprinting. That is, whatever websites the user visits can capture the dimensions of the browser window, browser versions, and IP address information. By capturing this information websites could make the assumption that the user is utilizing a SADD desktop for access.

The third point of deducing possible vulnerability could be observed by learning how correlation attacks against the Tor network work and by understanding that Man-In-The-Middle attacks require some form of contact with the user, SADD servers, and/or affected machine(s). [4] A scholarly article titled *Users Get Routed*, explains that in correlation attacks the entities looking for a particular user has to be intercepting the traffic of the machine that is entering and leaving the Tor network and the traffic entering and leaving the affected machine. In the case of SADD would require an outside entity intercepting traffic between the user's machine and the SADD website, the SADD backend server(s) and the Tor network, and the traffic that is entering and leaving the affected machine. Since SADD utilized web based RDP through an open-source Javascript client called noVNC, the user is essentially clicking pixels which is not a lot of traffic. So unless the outside entity had access to the user's screen it would be difficult to tell what the user is accessing. If the outside had access to the server, it would have to keep track of which SADD desktop session corresponds with what user. Finally, if the outside entity only had access to the affected

machine, the entity would have to use the techniques discussed in the second point of deducing possible vulnerability to have reason to believe that a SADD desktop is responsible for the traffic. Otherwise, it could be in their eyes, another Tor user.

The fourth and final point of deducing possible vulnerability could be observed by taking a look at the user's actions while using a SADD desktop. If the user wishes to remain anonymous on the internet, the user should not be logging into any personally identifying websites. That is, do not log into social media, bank accounts, or websites that only a select few individuals access. This also means not performing actions that would identify a user such as viewing a favorite clothing wishlist at a specific time everyday or searching a user specific phrase on the Internet.

III. PRIVACY

A. *Review*

[2] The article published by The Hacker Online Club, also states that in order to maintain user Privacy, SADD destroys all virtual machines and middleware pertaining to them such as Tor gateways and sessions, forensically. That is, it goes to the memory locations of the files on the server harddrive (HDD) and rewrites the binary ones and zeros as zeros. [6] An article on How-To Geek called *Learn How To Securely Delete Files in Windows*, lists a couple of programs that could be used to do this. Some of these programs can also be used on other operating systems to accomplish, what it calls, secure file deletion. The SADD patent states that it may use secure-delete in Linux to securely destroy the Virtual Machines. There is no way to tell what parameters are passed to secure-delete to destroy the files.

Looking at the SADD.IO website and YouTube video, it is stated that there are no logs or history of user action recorded. Meaning that if SADD were to be subpoenaed, there would be no data that SADD could actually give to the requesting entity. Compared to Virtual Private Networks (VPNs), this same statement is usually advertised. [7] An article written by Rob Mardisalu, talks about a 100+ VPNs and their logging policies. SADD does not have a logging policy and their patent does show the destruction of all data connected with the SADD desktop. Also, the requesting party cannot see when the access tokens are used, there is no real way to say who is operating a SADD desktop at a specific time. This makes the desktop sessions private.

B. *Deducing*

The initial point of deducing possible vulnerability involving the exploitation of SADD's privacy is by taking a look at the persistence of the generated desktops. If it could be proven that files and programs do persist after the end of the user session, then the integrity of SADD's privacy would be in jeopardy.

The second point of deducing possible vulnerability involves proving that there are no logs kept by the SADD desktops. If a user could prove that there is some record of the time a generated desktop was used or what was done on that desktop, then it could be easily concluded that there is no user privacy on SADD.

The third and final point of deducing possible vulnerabilities involving SADD's privacy, is once again looking at Man-In-The-Middle attacks. If some outside entity had access to the SADD backend servers the outside entity could take measures to pinpoint who is using the generated desktops and record what is being done on these desktops.

C. *Testing*

To test the first point of deducing possible vulnerability of desktop persistence, I used https://webchat.freenode.net, which is an Internet Relay Chat server. Freenode allows users to connect and speak with one another via text. I used https://kiwiirc.com, which is an Internet Relay Chat client. I then used the Kiwi IRC client to create a channel called "sadd"on the Rizon server and connected to it as shown below:

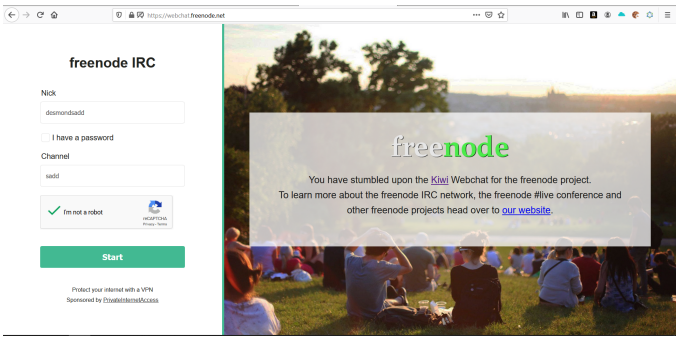

Fig. 10 - Connecting to Freenode IRC server with Kiwi IRC client.

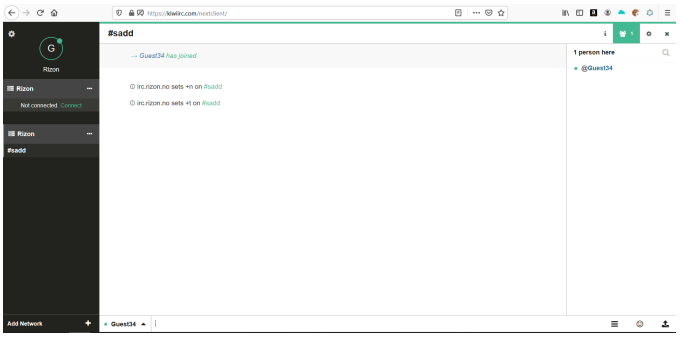

Fig. 11 - Joining the "sadd" channel on the Freenode IRC server.

I completed the same process above in our generated SADD desktop. The screenshot below shows the connection to the "sadd" channel on the generated SADD desktop:

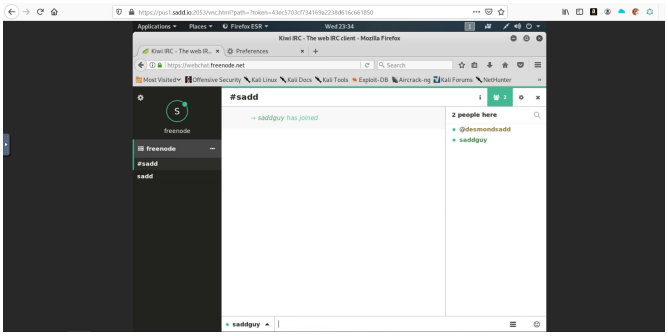

Fig. 12 - Joining the "sadd" channel on the Freenode IRC server through the generated SADD desktop.

At the end of the desktop session I was disconnected. It was also noticed that the second user in the IRC chat was disconnected. This proves that the desktop was at least shut down. There is no way to test if it was actually destroyed at the end of the user session.

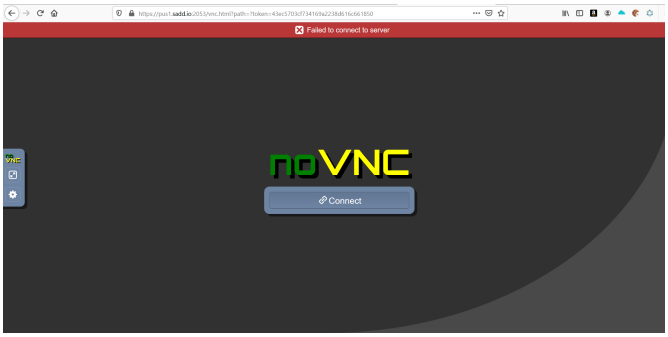

Fig. 13 - The generated SADD desktop session ending.

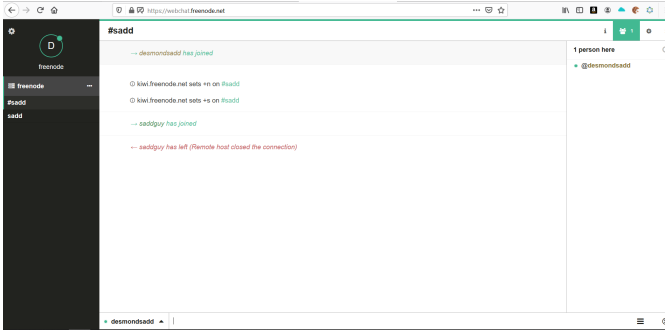

Fig. 14 - The user of the generated SADD desktop being disconnected.

As for the second point of deducing possible vulnerability, there is no real way to test if any logs or recordings of the user sessions had taken place. However, the privacy policy found on the SADD website (https://sadd.io/privacypolicy.pdf) states that no logs are kept and all user actions are forensically destroyed. If the desktops are forensically destroyed as stated by SADD, then there would be no record of any activity of what occurred on any of the generated desktops. It is also fair to state that each generated desktop is around 40 GB in memory. As of 12:05 AM CST on November 28th, 2019, SADD.IO has generated 72,773 desktops. This means that SADD would have been logging nearly two quadrillion bytes of Hard Disk data. Therefore, SADD storing the data of the desktops is not too feasible to believe. Not to mention that if SADD had a recording mechanism the amount of data stored would be at a much greater volume than two quadrillion bytes.

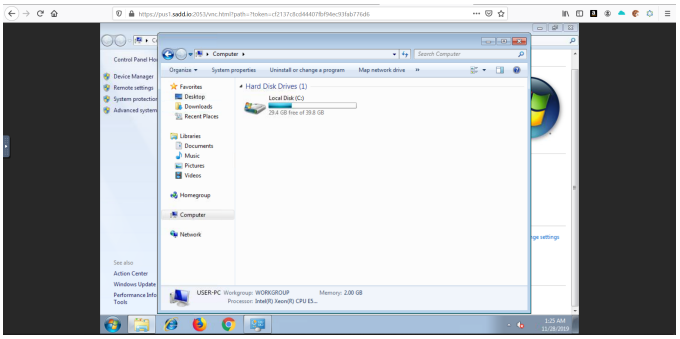

Fig. 15 - Generated Windows desktop showing 40 GB of HDD space.

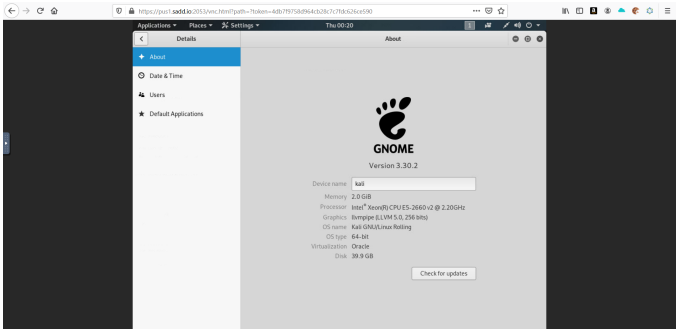

Fig. 16 - Generated Linux desktop showing 40 GB of HDD space.

Testing the third point of deducing possible vulnerability for Privacy is trivial, because when the third point of deducing possible vulnerability of Anonymity was tested, we showed that if it were possible, then Anonymity, Privacy, and Security would all be compromised. However, with Privacy, users should be more concerned about what settings that have on the devices that the user is using to access SADD. For example, users should be very mindful of what DNS servers and Internet Service Providers (ISPs) trust. It is recommended that the user uses honest and trustworthy ISPs and DNS servers that have a zero logging privacy policy. [8] The Content Delivery Network company, Cloudflare, provides a "privacy-first consumer DNS service". It holds no identity of the user making DNS requests. This aids in both Anonymity and Privacy for the user accessing SADD.

IV. SECURITY

A. Review

[2] The article published by The Hacker Online Club states that SADD provides security through various ways. The first being through isolation. This is because the generated desktop is actually a virtual machine spun up on a remote server. What this means is that if the virtual machine is corrupted, ruined, or destroyed in any way, it will not impact the user's host machine at all.

The virtual machine is remotely controlled by the user through HTML5 RDP. Simply put, the user controls the desktop by clicking the pictures of the rendered generated desktop through the SADD webpage. Wherever the user clicks is then sent to the server for the actual clicking to happen on the Virtual Machine. This same process is also used for typing.

The article also states that the same vulnerabilities that impact Tor browser do not impact SADD. [9] In a related article, *Warning: Critical Tor Browser Vulnerability Leaks Users' Real IP Address-Update Now* written by Mohit Kumar, elaborates on vulnerability that affect Tor Browser users by abusing the file:// protocol. If an outdated Tor Browser user were to click links using this protocol, the file:// protocol would target some file on the user's system instead of through Tor browser. However, since all SADD generated desktops are routed through the Tor network, this exploit would fail. This is because all network requests made on the generated desktops are made through Tor, not just the browser.

B. Deducing

First point of deducing possible vulnerabilities in exploiting the security of SADD is making attempts to discover the IP address(es) of the backend SADD server(s). If this information is made public to a malicious actor, the actor could make attempts to gain remote access to the server, compromising all users.

The second point of deducing possible vulnerability is looking at how SADD handles all requests from start to finish in generating a desktop. If SADD leaks any user information or has any unencrypted communications through the browser, this makes it easy for Man-In-The-Middle attacks.

The third point of deducing possible vulnerability involves the user escaping out of the generated desktop. If it is possible for the user to escape out of the generated desktop into the backend server, then all users will once again be compromised.

C. Testing

To test the first point of deducing possible vulnerability, I pinged both the main sadd.io domain and the pus1.sadd.io domain. The sadd.io domain is for the home page and the pus1 subdomain is where users are redirected to use their generated desktop. The IPv6 address, 2606:4700:30::6812:31c0, belonging to Cloudflare was returned for both. I checked the IPv6 address using the website using https://whatismyipaddress.com/ip/2606:4700:30::6812:31c0

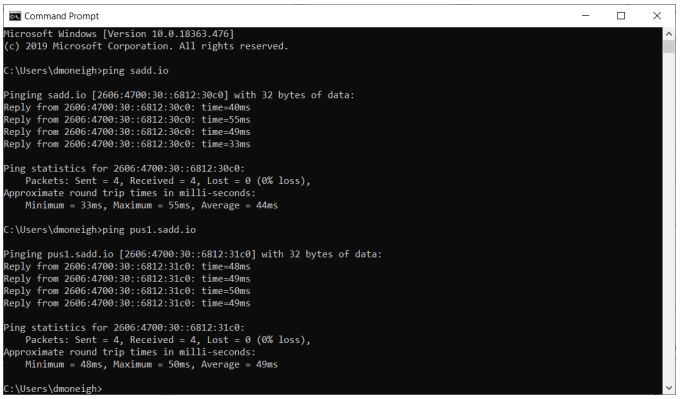

Fig. 17 - Pinging the sadd.io domain and subdomain.

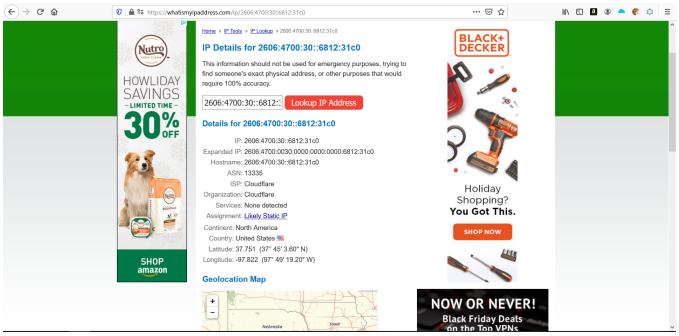

Fig. 18 - Checking the IPv6 address to determine it belonged to Cloudflare.

I then used https://dnsdumpster.com to see if any non-cloudflare IP addresses were exposed. Most subdomains returned with the IPv4 address of 104.18.48.192. This is also a Cloudflare IP address as shown below:

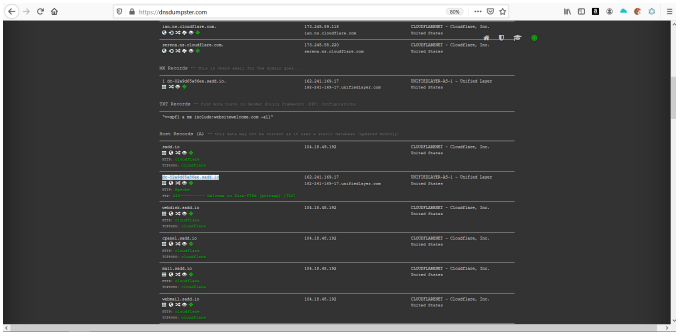

Fig. 19 - The DNS dumpster results for sadd.io.

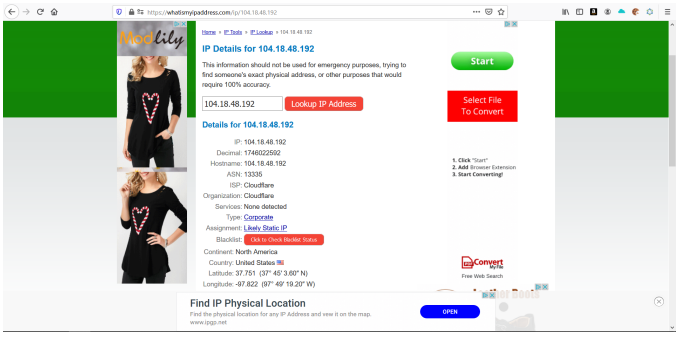

Fig. 20 - Checking the IPv4 address to determine it belonged to Cloudflare.

However, there was one subdomain whose IP address did not belong to Cloudflare. The subdomain was dc-02a9d65a86ee.sadd.io. The IPv4 tied to that subdomain was 162.241.169.17. It belongs to a Virtualization Hosting company called United Layer. There is no guarantee that this service is tied to the sadd.io website or its generated desktop service but it is definitely a good finding.

To test the second point of deducing possible vulnerability I looked at how traffic was passed on both the sadd.io domain and pus1.sadd.io subdomain. I used Wireshark and Firefox built in Network Monitor to analyze all traffic. When connecting to sadd.io all traffic is immediately encrypted with TLS 1.3.

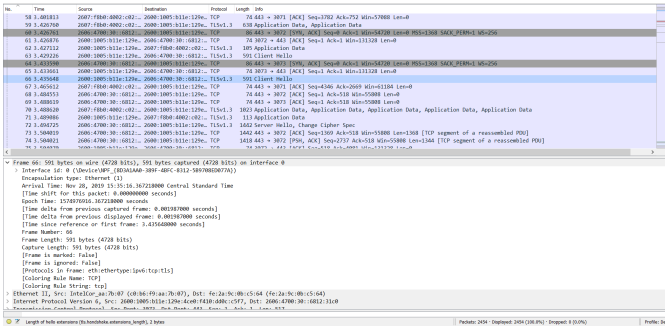

Fig. 21- Wireshark showing the client hello and TLS 1.3 handshake occurring with sadd.io.

I proceeded to generate a desktop of choice and paid close attention to how the parameters of generation were passed. There were four parameters: type, order, location, time, and g-captcha-response. There parameters were posted to the site securely as shown below:

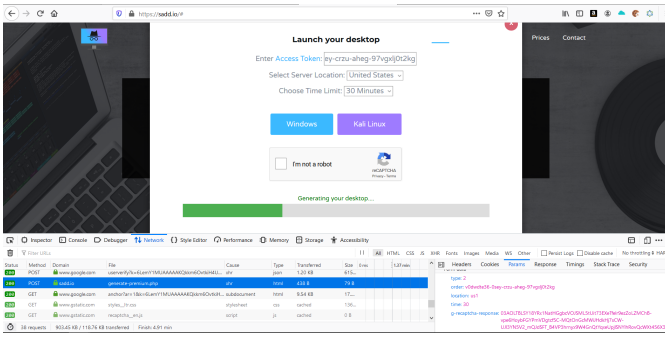

Fig. 22 - POST parameters being securely posted to the backend server.

SADD.IO is utilizing Google Captcha when posting parameters to the backend server. This mitigates most efforts of Cross-Site-Request-Forgery.

When connecting to the generated desktop, it was noted from Wireshark that the streaming of the generated desktop or HTML5 RDP was actually using version 1.2 of TLS.

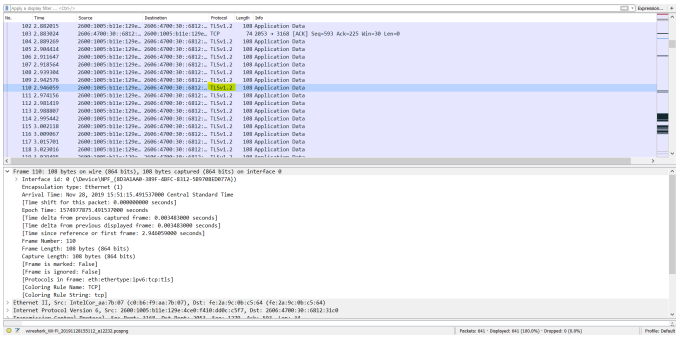

Fig. 23 - Wireshark showing the generated desktops are using TLS 1.2 to encrypt traffic.

To test the third point of deducing possible vulnerability I checked the network settings on multiple different generated desktops. These desktops each had two Local Area Network (LAN) IP addresses. One IP address that starts with the octets of 192.168.xx.101. The other IP address always points towards 10.152.152.10.

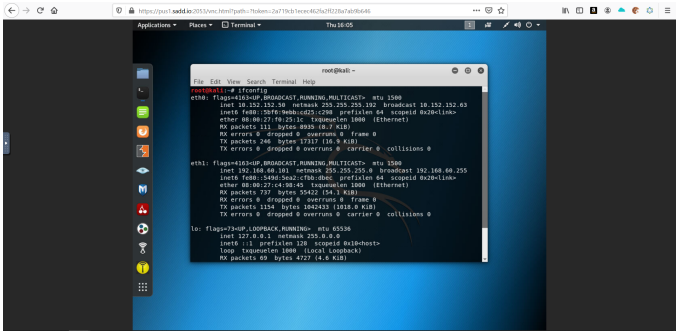

Fig. 24 - The Kali Linux Desktop showing the LAN IP addresses.

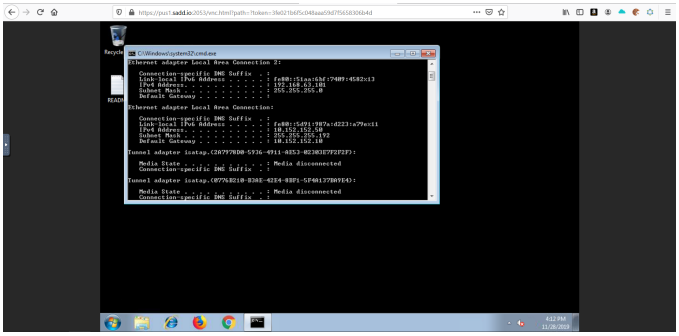

Fig. 25 - The Windows Desktop showing the LAN IP addresses.

The only IP address that changes on desktop generation is the 192.168.xx.101 IP address. This can help conclude that this IP address is unique to every machine on the server. The 10.152.152.10 IP address is obviously the connection to the Tor network. The SADD patent states that the 10.152.152.10 IP is a Whonix Virtual Machine. Whonix is an Operating System used to route all traffic through Tor. What this means is that every generated desktop has its own generated Tor router. [10] Whonix, the organization, posted an article that talks about all the Vulnerabilities that it protects its users from. Its website also has resources that explain how users are able to remain anonymous on the Internet.

With all the information known about what the generated desktop is connected to, I made an attempt to escape out of the Virtual Machine. Because it is known that these desktops use RDP, I generated two desktops, and tried using the command rdesktop on the Kali Linux terminal to connect to other desktop I generated. The IP address of the Windows machine I generated is 192.168.64.101. Remoting into it completely failed and pinging that IP address from the Kali Linux machine seemed useless as well.

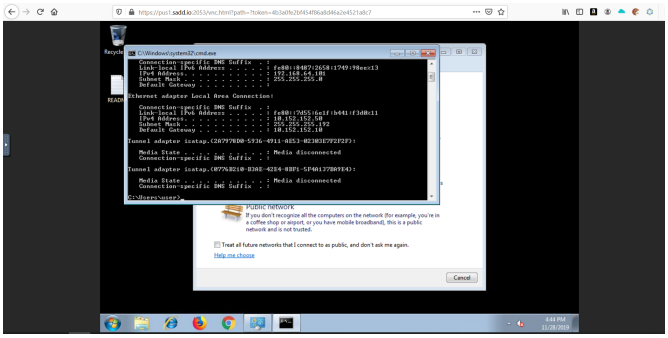

Fig. 26 - Obtaining the LAN IP address of the Windows Desktop.

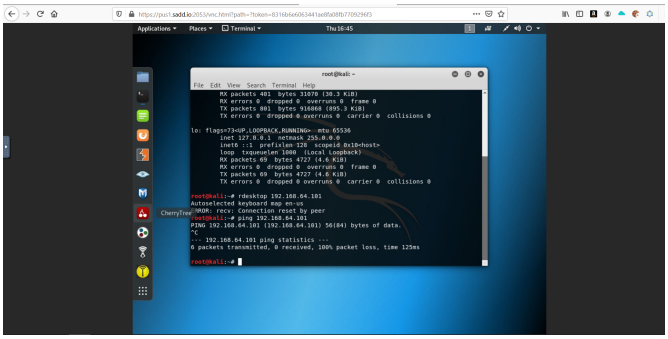

Fig. 27 - Attempting to ping and connect to the Windows machine via the rdesktop command.

I was unable to ping 10.152.152.10. So I decided to run an NMAP scan on it. This scan said that almost all ports were open. This is incorrect as I was not able to connect to the machine via the rdesktop or vncviewer command.

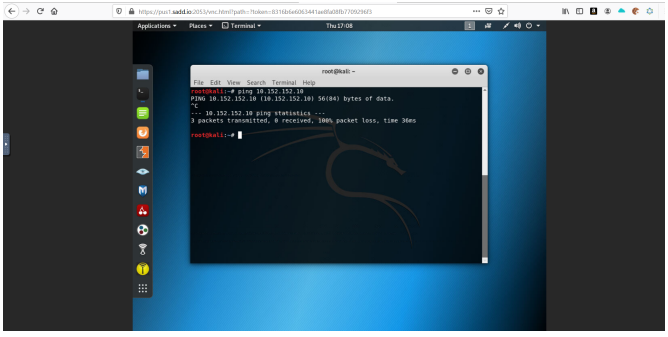

Fig. 28 - Pinging 10.152.152.10 and it fails.

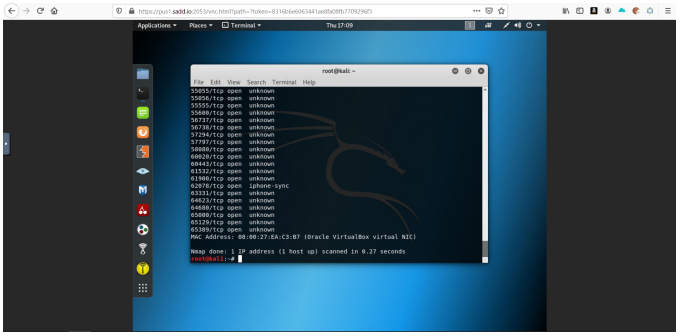

Fig. 29 - Running an NMAP scan on 10.152.152.10.

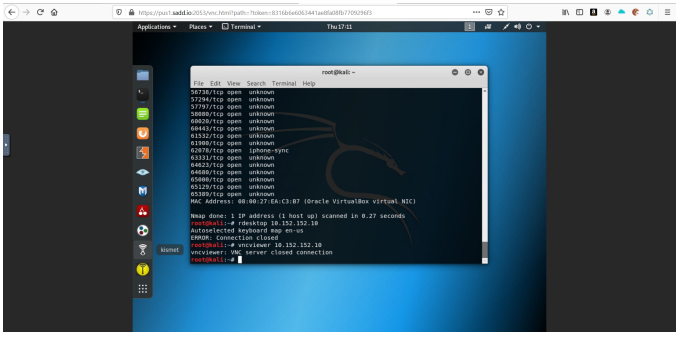

Fig. 30 - Attempting to connect to 10.152.152.10 via the rdesktop and vncviewer command.

V. Discussions

When using a service such as SADD is very important for the user to identify their goals. Does the user want to maintain Anonymity? Does the user want Security? Or does the user only care about their Privacy? Security and Privacy are automatically provided by SADD, but Anonymity can be easily ruined by the user through very simple mistakes.

Users could easily ruin their Anonymity on or off of SADD by doing some of the following:

1) Logging into personally identifiable websites.
2) Conducting activities that the user does on a normal basis, such as visiting a particular page on a blog everyday at a specific time.
3) Using the same keywords that the user normally uses when searching the internet.
4) Performing some reaction as a result of a personal life event.

Many internet users who want to remain anonymous on the internet constantly compare SADD to VPNs. When in fact SADD is much different than a VPN because it not only anonymizes the user, but also protects their system via isolation, and maintains privacy by shredding the Virtual Machines altogether. [11] VPNs on the other hand only provide anonymity at the network level. That anonymity is only ever maintained if the VPN provider does not keep any logs or any PII that is correlatable to its users.

As with all online services, there has to be a level of trust. If

the user does not have complete and full access or control over what is being used, there needs to be trust. Many VPNs state that the VPN does not keep logs but when the VPN is subpoenaed tons of user information as well as logs are surrendered. It is always a good idea for users to look at the Privacy Policy of every service of choice.

In terms of safety, SADD has a bigger impact than VPNs. If every business in the world had an inhouse SADD backend server to generate desktops, it would be extremely difficult to identify users for a couple of reasons:

1) All generated desktops are routed through Tor, therefore, the internet would view the connection as a Tor user.
2) Now that the SADD server is decentralized, it would be difficult to attack the server.

VPNs are definitely more identifiable at the network level. This is because when a user connects to a VPN the user is simply using the ISP and sometimes DNS settings of that VPN (if configured correctly) to connect to the internet. Any misconfigurations of the VPN could actually hurt the user's anonymity.

However, VPNs do have the upper hand in terms of speed and latency. SADD already has a lot of latency because of the Tor network and the user's distance from the backend server. Not to mention that SADD also used Cloudflare CDN to deliver content to its users, adding more latency.

[12] Over the previous two years, some users have shown concern on multiple different blogs and forums about the nefarious activities that SADD could be used for. [13] One blog in particular, DarkWebList, states that "Common Law-Abiding people" are not so concerned about destroying their browser history. [14] This is not an entirely true statement as a survey conducted by the Pew Research Center, ultimately determined that Americans alone think that their everyday privacy is very important to them.

[2] Though the Hackers Online Club article states that SADD could be used to conduct nefarious actions, it is not condoned. If an access token could be identified with being associated with nefarious use, it would be immediately disabled. It would be tough to regulate nefarious actions because:

1) What would alert the SADD server maintainers that nefarious activities are taking place, if all network traffic is routed through Tor?
2) How could the SADD server maintainers observe that malicious activities are occurring without violating the Privacy of the users?

In another conversation, SADD could be very beneficial to not only individuals and businesses, but governmental entities as a whole. SADD's modular backend design allows entities to enhance anonymity even more than what SADD already provides. Entities could do this by purchasing a SADD backend server through the TBD plan on the website. The TBD plan gives the purchasing entity physical access to the server. With physical access, the proper physical protocols, and network configurations, entities could easily make the purchased server only accessible through the LAN connections. If the entity does not disclose to some outside source that a SADD server has been purchased or leaks the LAN IP address of the server, it is not likely that the server would be in mind for a malicious actor to attack.

VI. Conclusion

In conclusion, SADD does an excellent job of maintaining the Anonymity, Privacy, and Security of all of its users. As with any online service, there is always room for improvement. SADD is a very generalized service that has many possible applications including but not being limited to:

1) A platform for anonymous cryptocurrency transactions.
2) A platform for testing malicious software.
3) A platform to provide a learning environment for students who have an interest in both software development and cyber security.
4) A platform for cyber threat mitigation through cloud-based isolation.
5) A platform to provide simplified regulatory compliance service for businesses.
6) A platform to aid in regulatory and litigation risk reduction for eDiscovery, privacy non-compliance, and liability.

References

[1] "Scalable Anonymous Disposable Desktops." *SADD*, Jackson CS Consulting, LLC, 25 Mar. 2018, https://sadd.io/.

[2] Patil, Chandrakant. "All-In-One Anonymity, Privacy and Security Platform Sadd.io." *All-In-One Anonymity, Privacy, and Security Platform SADD.IO*, HackersOnlineClub, 3 Oct. 2020, https://hackersonlineclub.com/all-in-one-anonymity-privacy-and-security-platform-sadd-io/

[3] Jackson, Desmond Armani. *SCALABLE ANONYMOUS DISPOSABLE DESKTOPS (SADD)* . 22 Jun. 2021, https://sadd.io/patent.pdf

[4] Jackson, Desmond Armani. *SCALABLE ANONYMOUS DISPOSABLE DESKTOPS (SADD)* . 11 Apr. 2023, https://sadd.io/patent2.pdf

[5] Johnson, Aaron, et al. "Users Get Routed:Traffic Correlation on Tor by Realistic Adversaries." *Users Get Routed*, ACM, 8 Nov. 2013, www.ohmygodel.com/publications/usersrouted-ccs13.pdf.

[6] "What You Must Know About OS Fingerprinting." *Infosec Resources*, InfoSec, 11 Mar. 2015, https://resources.infosecinstitute.com/must-know-os-fingerprinting/.

[7] Kaufman, Lori. *Learn How to Securely Delete Files in Windows*. How-To Geek, 12 July 2017, www.howtogeek.com/72130/learn-how-to-securely-delete-files-in-windows/.

[8] Mardisalu, Rob. "100+ VPN Logging Policies Debunked (2019): TheBestVPN.com." *TheBestVPN.com*, TheBestVPN, 15 May 2019, https://thebestvpn.com/118-vpns-logging-policy/.

[9] Prince, Matthew. "Announcing 1.1.1.1: the Fastest, Privacy-First Consumer DNS Service." *The Cloudflare Blog*, Cloudflare, 1 Apr. 2018, https://blog.cloudflare.com/announcing-1111/.

[10] Kumar, Mohit. *Warning: Critical Tor Browser Vulnerability Leaks Users' Real IP Address-Update Now*. The Hacker News, 4 Nov. 2017, https://thehackernews.com/2017/11/tor-browser-real-ip.html.

[11] Patrick, and 0brand. "Security in Real World." *Whonix*, Whonix, 27 Feb. 2018, https://www.whonix.org/wiki/Security_in_Real_World.

[12] "Does a VPN Protect against Computer Viruses?" *Buffered.com*, Buffered Ltd, 2018, https://buffered.com/faq/vpn-protect-computer-viruses/.

[13] Brinkmann, Martin. "Sadd: Anonymous Virtual Desktops with Tor Built-in - GHacks Tech News." *GHacks Technology News*, GHacks, 23 Aug. 2018, https://www.ghacks.net/2018/08/23/sadd-anonymous-virtual-desktops-with-tor-built-in/.

[14] "What Is SADD.IO and How Does It Work? Latest Guide." *Dark Web List - Dark Web Search - Dark Web News*, Dark Web List, 14 June 2019, https://www.darkweblist.com/2019/hacking/what-is-sadd-io-and-how-does-it-work/.

[15] Madden, Mary, and Lee Rainie. "Americans' Attitudes About Privacy, Security and Surveillance." *Pew Research Center: Internet, Science & Tech*, Pew Research Center, 24 Mar. 2016, https://www.pewresearch.org/internet/2015/05/20/americans-attitudes-about-privacy-security-and-surveillance/.