Towards Zero Trust: The Design and Implementation of a Secure End-Point Device for Remote Working

Kemal Bicakci
Informatics Institute
Istanbul Technical University
Istanbul, Turkey
0000-0002-2378-8027

Yusuf Uzunay
Securify Information Tech. and Security
Training Consulting Ltd.
Ankara, Turkey
0000-0001-8768-6620

Mansoor Khan
Guvenpark Information Tech.
Research and Development Ltd.
Istanbul, Turkey
0000-0002-8301-6539

2021 International Conference on Information Security and Cryptology (ISCTURKEY) | 978-1-6654-0776-2/21/$31.00 ©2021 IEEE | DOI: 10.1109/ISCTURKEY53027.2021.9654298

***Abstract*—COVID-19 pandemic and lockdowns forced employees across the world to work from home. Remote working has become a necessity rather than a choice. However, in order to meet this increasing demand, the most pressing security concerns of organizations should be addressed. In this paper, we present the design and implementation of ProGun, an end-point device (a USB dongle) for remote working. We present the hardware/software co-design of ProGun, by which most security risks due to lack of physical protection could be mitigated. We also discuss choices we made among many alternatives for user authentication and their security and usability implications in a remote working environment.**

***Index Terms*—security, remote working, authentication, two-factor authentication, risk based authentication, security device, zero trust.**

I. Introduction

Both in Europe and USA, before the pandemic only less than 3% of workforce worked remotely [1]. Because of this limited adoption, prior to COVID-19, most organizations had little technical preparation for supporting this practice. However nearly half of employees worked remotely full-time during the pandemic [2]. The rapid shift to remote working is frightening especially from security point of view. Traditionally, cyber security is relied on physical security of offices, networks, and computers; not present anymore in remote workforces. According to a recent report from Interpol, cyber attacks are at an alarming pace during the ongoing COVID-19 crisis and is expected to accelerate even further [3].

To defend against the cyber attacks, we already have effective techniques such as encryption, authentication, access control and many others. However each of these technical means is not sufficient when used alone hence the challenge remains in combining these conceptual ideas and turning them into a tangible product. ProGun is a device we have developed for this purpose. In this paper, we present the design and implementation of ProGun and discuss how it could protect employees and organizations from the concerned cyber risks of remote working.

The rest of the paper is organized as follows. In Section 2, we start with the requirements and list the major in-scope security threats of remote working we uncovered using STRIDE threat modeling method. In Section 3, we present the design and implementation of secure end-point device called ProGun to satisfy the identified security requirements. Our presentation involves both hardware and software aspects. In Section 4, one of the most crucial requirements, authentication of remote users in a secure and usable fashion is discussed by first exploring the whole design space and then giving the justifications for the choices we have made. In Section 5, we overview the previous work on security of remote working. In Section 6, we finish up with our concluding remarks and future work directions.

II. Threat Modeling and Security Requirements for Remote Working

Given the wide range of constraints and choices, the design and implementation a security device for remote working is a herculean task. A proper starting point could be to begin with the security requirements. More precisely, first we should answer this question: "what security threats do we want to address with this end-point device?" STRIDE is a simple yet powerful semi-structured threat modeling method to answer this question of great importance [4]. This method enables us to enumerate and prioritize security threats in just four steps:

1) Model your system (use data flow diagrams (DFDs) with some required extensions like trust boundaries).
2) Identify the threats (brainstorm using STRIDE mnemonic which stands for Spoofing, Tampering, Repudiation, Information Disclosure, Denial of Service, and Elevation of Privilege).
3) Decide on each threat (options include avoiding, addressing, accepting, transferring and ignoring).
4) Check your work.

We carried out a threat modeling exercise using STRIDE. Due to space limitations, we do not present all of its details here. Instead, we first make a few key observations related to our case and then present the most relevant threats we discovered together with our decisions for treatment.

Some of our key observations after we apply STRIDE method for a typical use case of ProGun are as follows:

1) It is straightforward to draw a DFD of the system (see Figure 2 for the conceptual model which could be the starting point for drawing it).
2) A trust boundary is defined as a location in a DFD where the level of trust changes. There is a natural trust boundary in our DFD between internal network and external world. There is usually a VPN server or a firewall located on this border.

This research is funded by TUBITAK (The Scientific and Technological Research Council of Turkey) under the grant No: 3200964.

TABLE I
MAIN SECURITY THREATS AND CHOSEN TREATMENTS FOR PROGUN.

#	Threat Type	Threat Name	Chosen Treatment	Residual Risk
1	Information Disclosure	Information leakage to remote unauthorized users	Strong authentication & secure booting with hardened OS	leaks to authenticated users if least-privilege principle is not correctly enforced
2	Tampering	Unauthorized corruption of sensitive files and documents	Strong authentication & secure booting with hardened OS	accidental corruption
3	Information Disclosure	Eavesdropping over the network	Use of VPN and encryption	exploiting a vulnerability in VPN or ProGun software
4	Spoofing	Impersonate authorized users	Strong authentication	software vulnerabilities
5	Repudiation	Denial of activities	Log audits	Log tampering

3) We could identify another trust boundary in Figure 2 since the USB device is trusted (as we will discuss later) whereas a local computer (local PC) is not.

The most prominent threats identified in our analysis and aimed to be mitigated in our work with the design of ProGun are provided in Table 1. Risks remaining after the treatment efforts are also provided in this table.

III. THE DESIGN AND IMPLEMENTATION OF A SECURE END-POINT DEVICE

We now present the design and implementation of ProGun to defend against the main security threats listed in Table 1.

As a starting point we assume there is a workstation located inside the intranet of an organization which should be accessed from a remote location securely. ProGun is a hardware with embedded software designed with multiple security layers to make it hard for attackers to bypass the applied security countermeasures. More specifically, ProGun is a USB dongle containing a hardened and restricted operating system which can be run on any COTS computer to make it secure for use in a remote location. As we will explain, the dongle also contains logical interfaces and features for authentication and additional security features.

The main use case for ProGun is to use the USB Dongle on a local machine from a remote location and access securely the workstation which is situated in the office. The ultimate goal is to provide a user experience that feels like user is working from the office.

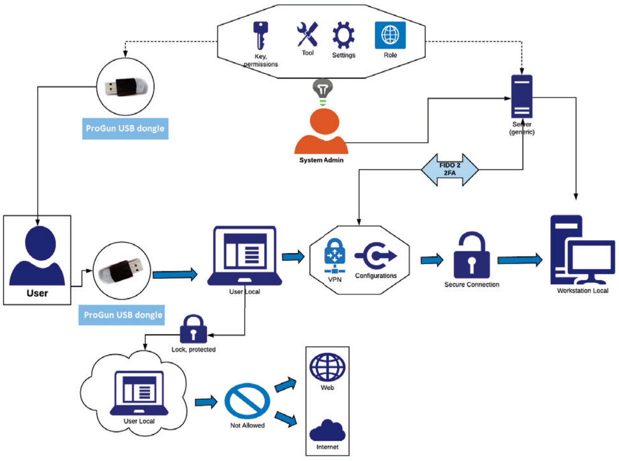

Fig. 1. A typical use case for ProGun.

As shown in Figure 1, with ProGun solution, any remote computer, secure or not, can be employed. In order to achieve this, ProGun needs to interact with the internal components of the computer like hard disk and manage external devices connected to it. By default, the users are not allowed to use hard disk and other devices once ProGun is in use. Only allowed devices can be used after configured by the system administrator.

One of the main problems with the remote computers is that their operating systems are usually not managed by the IT department of the employee's company, which makes it hard to ensure the security of the system for instance the operating system could be infected by a virus or any other harmful software.

ProGun's solution is to provide a secure operating system booted from a portable USB dongle. ProGun USB has enough non-volatile memory to store a hardened operating system which is also encrypted with a key derived from a passphrase only known by a specific user (owner of the ProGun).

The users plug-in the dongle to the computer and boot the system with the secure operating system. This operating system does not allow any access to internal storage units which means that no data could be transferred to the user's hard disk. It is also not possible to upload any file from the local computer to the workstations and servers inside the intranet.

The booted operating system provides a user interface to connect to the local wireless or wired network. Users could make a selection between options. On the other hand, users do not have access to the Internet directly. The VPN configuration is ready inside the dongle personalized for the user and when the local connection is established, this VPN connection is fixed automatically. All the connections are forwarded to the VPN tunnel.

Users can access to the remote workstations, servers and applications only through the VPN tunnel. Remote Desktop is the main application for our use case. With it, users can have an access to their workstations in their office with their local laptops remotely e.g., from home. The dongle also has other hardware components such as GNSS sensor. Using this sensor which connects to GPS, the system could log the location and time information. With this logging, the system admin could audit location of users while they are connected. By this logging, users who cheats or violates the security policy in some other way could be held accountable for their actions. The location information is also used to restrict the dongle's use only to a specific country or a city. The access policy is managed by the administrator.

A. Conceptual Model

The conceptual model is given in Figure 2 showing the secure tunneled connection between local machine and remote workstation over the VPN server. Secure USB (ProGun device) which contains a customized operating system, eMMC, and a microcontroller for logical interfaces is also seen in this figure.

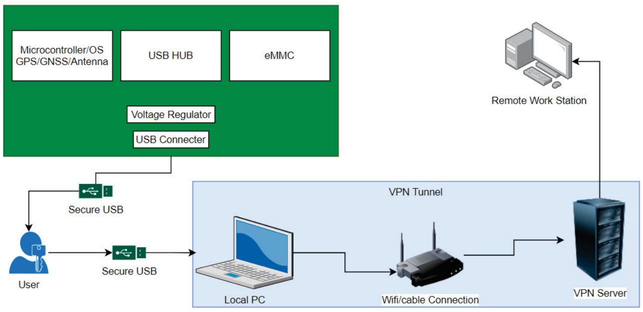

Fig. 2. Conceptual model for Progun.

B. Hardware Design

ProGun hardware consists of a USB connector 3.0, eMMC, Voltage Regulator, USB HUB, Antenna, GNSS/GPS and NAND Flash Controller. A microcontroller is connected to the USB 3.0 port via USB HUB as shown in Figure 3. It has the following properties: USB 2.0 device support, GNSS/GPS support, FIDO2 support and Real Time Operation System (RTOS) support.

USB Connector provides the necessary power to the system and connected to the microcontroller and memory via USB HUB. eMMC is a memory area where the operating system (ProGun OS) is stored. USB HUB orients the signals from USB 3.0 to memory and microcontroller. While one of the outputs of the USB HUB boots the system over the memory, the other output port communicates with the microcontroller and GNSS/GPS module. This module collects the user's coordinates instantly to determine whether the user is in the allowed working zone.

ProGun dongle contains both physical and logical interfaces, Physical interface contains the eMMC and the logical Interface provides the GNNS / Antenna interfaces.

C. Software Design

ProGun dongle is customized for its users. The system administration generates an image for each user so that a specific VPN configuration is automatically available once the dongle is in use. VPN configuration information is hidden from the user and cannot be tampered with for security reasons. The disk image contains the following components: GRUB2 software, Encrypted OS Image, Encrypted ProGun Service Software and Encrypted Configuration data (VPN Credential).

For the research and development of restricted operating system image that can be booted from a USB storage, we get help from Mkosi project [5] and use Debian 11 (Bullseye) distribution. For disk encryption, we use Linux Unified Setup (LUKS) [6], a disk encryption specification with AES-XTS-256 algorithm.

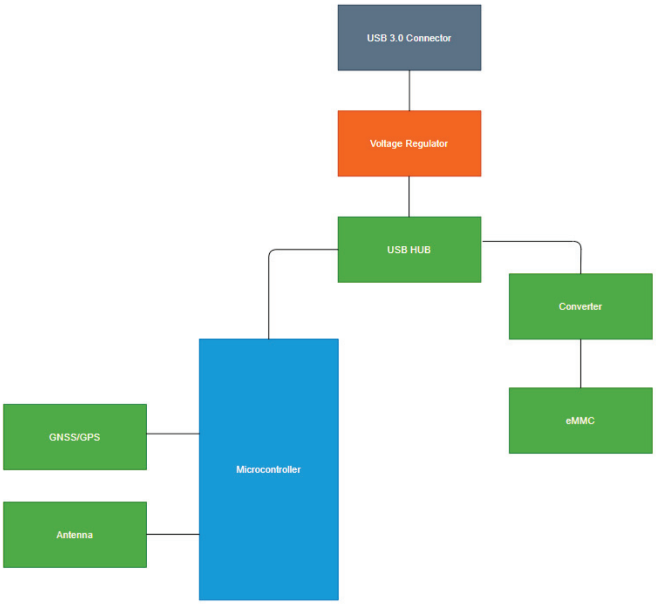

Fig. 3. Hardware Design of ProGun.

The MCU firmware runs on the microcontroller of the ProGun USB Dongle. It is developed in C language and use Zephyr operating system. This firmware is in communication with the ProGUN OS booted on the local PC. This communication basically has two purposes:

First, it provides FIDO2 support [7]. With the firmware, user authentication could be carried out using public-key cryptography with the FIDO2 standard.

Second, it provides GPS location information to OS services at certain intervals. The firmware periodically communicates with the GNSS/GPS module and passes the location information when requested.

GRUB2 software is unencrypted on the eMMC Memory. It is the first code to be executed by the bootloader software on the client PC. The integrity of this software on the ProGun USB Dongle is preserved.

GRUB2 presents an interface to the user so that s/he can enter a password and generates the cryptographic key. Then, with this key, ProGun decrypts the operating system image. As mentioned, hard disk is encrypted using LUKS. ProGUN OS (Linux Operating System and Services) is stored encrypted on the ProGUN USB Dongle. Necessary drivers for common hardware are also provided within the image.

Basic tasks of ProGun OS are as follows:

1) Connecting to the target VPN server using DHCP or static network settings at startup. VPN credentials available on the image are used for the connection. We aimed at providing the same user experience and restrictions for different VPN technologies.
2) User's RDP connection is also established automatically to make target PC desktop available. The desktop environment enables the user to make video and audio calls by enabling the local microphone, speaker and camera devices for the target system with the remote desktop connection.

3) MCU is in communication with firmware to provide FIDO2 support and receive GPS data.
4) Allowing or blocking access to different peripherals such as storage device, Bluetooth/WiFi adapter, card reader, serial port converter, camera, keyboard, mouse within the scope of defined rules. The user is prevented from accessing the Internet without using VPN connection. The user is also prevented to access USB devices (except ProGun USB Dongle).

D. *Other Use Cases*

As already described, the primary use case for ProGun is to provide each use a private and personalized USB dongle. In this scenario, system admin generates the customized ProGun OS image of the dongle using a system management software. Permissions, VPN configuration and other relevant information is also embedded into the image.

In a second use case where a ProGun dongle is not provided to the user (maybe due to cost or logistic reasons), he/she can use a standard USB device having the customized image of the ProGun OS containing the required configuration downloaded from a system server. However, this choice is not as secure as the original use case since many features of the ProGun hardware is lacking.

Another use case might involve the installation of customized OS image directly to the local PC. However this might be considered too restricted since now local PC could only be used for remote working and not for personal purposes.

IV. User Authentication for Remote Working

One of the crucial security requirements in a remote working environment is user authentication; how to reliably identify the remote user attempting to gain access to the company's Intranet and resources. If this basic requirement is not satisfied, most (if not all) items given in Table 1 becomes in jeopardy.

First and foremost, we make again a couple of key observations as follows:

1) The danger of depending only on passwords for user authentication is well documented. In fact, in 2017 it was reported that among all security breaches, 81% leveraged stolen or weak passwords and 43% were due to social engineering attacks [8].
2) The availability of a hardware token (ProGun USB dongle) provides an opportunity to achieve a usable multi-factor authentication (MFA) scheme. In fact, our choice to implement FIDO2 standard is in line with this observation.
3) Even with the available FIDO2 methods as mentioned above, especially if biometrics is not employed, user authentication related risks are not totally prevented. For instance suppose the loss or theft of the USB dongle which is not uncommon, then, the attacker only needs to obtain the password, which is not difficult with simple social engineering tricks [9]. The offer of single factor (passwordless) authentication option by FIDO2 only magnifies the risk.

With this observations, our goal in this section is to identify and enumerate options regarding user authentication for the use case of remote working and consider pros and cons for each. Having this list of options summarized in Table 2, we discuss discuss and give justifications the choices we have made in the design of ProGun as follows:

1) Additional use of an authenticator application installed on a mobile phone for MFA is a well-justified option [10]. Most users always carry a smartphone so using it for authentication does not pose much burden with respect to usability.
2) With mobile phones, passwordless authentication could be offered with ease while still remaining in two-factor authentication (2FA) territory. The availability of biometrics (fingerprint and/or face recognition) in modern smartphones could enable mobile authenticators to implement this option. With it, OTP code or the confirmation screen could only be accessed by the user after biometric authentication is succeeded. In other words, 2FA is implemented by combining "what you have" and "who you are" factors. If password is also asked, then "what you know" factor is also present, then the solution is truly called MFA (rather than just 2FA).
3) Use of confirmations on the mobile phone instead of manual entry of OTP is more usable but requires local computer as well as the mobile phone to be online (upon user confirmation, a backend server should share this information with the computer over the network. On the other hand, the generation and use of OTP codes using TOTP standard [11] does not have this requirement (remember that before VPN connection is established, the computer is offline).
4) Use of MFA only for VPN and/or RDP access and not for OS login and & unlock opens the door for luchtime attacks, which might occur when a previously authenticated user walks away from his computer, thus allowing someone else to take over the login session and engage in egregious actions [12]. We remind that depending on re-authentication after an inactivity timeout period could not be the ultimate solution because choosing the proper timeout threshold is not any easy task.
5) Settings regarding how frequent and which kind of MFA is mandated in a remote working environment is like double-edged sword. MFA can cause usability concerns when applied harshly. On the other hand, it could not be effective if only rarely mandated. We notice that Risk Based Authentication (RBA) [13] could help us to solve this tension between usability and security. In the rest of this section, we will elaborate more on our RBA solution.

RBA could be defined as the method of applying different levels of user authentication, depending on the likelihood that access to the system is compromised. The greater the compromise risk, the more comprehensive and restrictive authentication becomes.

Risk evaluation is one of the most important issues in RBA solutions. It requires collection of contextual authentication data such as:

1) Time and location (we could complement coarse data derived from IP addresses with fine-grained geolocation

TABLE II
COMPARISON OF MULTI-FACTOR AUTHENTICATION (MFA) SOLUTIONS FOR PROGUN. RBA STANDS FOR RISK BASED AUTHENTICATION.

#	Named MFA solution	Main Advantage	Main Disadvantage	Implementation Choice
1	Password + USB Dongle	No need for additional device	Vulnerable to theft & social engineering attacks	FIDO2
2	MFA w/ authenticator for OS login & unlock	Protection against lunchtime attack	Manual entry of OTP codes	TOTP
3	MFA w/ authenticator in VPN and/or RDP access	Many online options available	No MFA for offline mode	Passwordless Authentication
4	MFA w/ authenticator & use of RBA	More usable and secure	Difficulty in risk evaluation	Custom-built solution (e.g., SecurifyID)

information collected via ProGun's GPS sensor).

2) Keystroke and mouse dynamics (useful also for continuous authentication enabling locking policies upon anomaly detection).
3) Device fingerprint (since ProGun is a customized company-controlled hardware device, there are numerous options for reliable collection of device fingerprinting data).
4) End-point health data (the basic idea here is to increase the risk level for instance if no anti-virus protection is available or the protection software is outdated. Since the end-point device in our main use case is a hardened integrity-protected device, this data is not of much use in our case).
5) Behavioral analytics (includes non-biometric behavioral data, as an example, consider the risk of an authenticated user connected to company's file server and download thousands of sensitive files).
6) Past user behavior (for instance the number of previously failed login attempts).

The overall objective is to catch any signal pointing to any anomaly and then take the required action. Possible actions are not restricted by asking the user for MFA. When the risk is above a certain point, more severe actions such as blocking the user or triggering an alarm could also be taken.

The translation of raw contextual authentication data to useful and reliable anomaly information is the real challenge here. Lack of historical data on attacks (therefore the data is usually unlabeled) as well as the difficulty in modeling legitimate user's behavior brings difficulties in modeling the problem as a standard classification setting in machine learning. However the good news is that even with simple rule based heuristics such as "ask user for MFA when a new IP is noticed" or "ask user for MFA before out-of-country access" may work out as a more usable and secure solution than static MFA. In the current version of ProGun, we have already implemented and applied this basic rule-based RBA options. We are currently working on more advanced machine learning based techniques. As part of this work, previously we implemented an RBA solution combining the keystroke data collected while user types in username and password with the other contextual data and showed that keystroke dynamics lead us to improved reliability results with respect to error rates of anomaly detection [14].

V. RELATED WORK

Security risks of remote working in the COVID-19 era are explored by Malecki [15]. We encourage readers to consult to this paper to read and learn about best practices such as regular testing of backups, left out-of-scope in our work.

Before remote working starts to become the norm, BYOD (Bring Your Own Device) acronym mostly refers to mobile devices such as smart phones. We argue that pandemic blurs the line between BYOD and corporate device in general. A nice summary of security and privacy concerns for BYOD is given in [16].

Zero trust is a term reflecting enterprise network trends that include remote users, BYOD and cloud computing. A recent report by NIST gave a definition of zero trust architecture and provides general deployment models and use cases. According to NIST, zero trust is a new security paradigm which assumes that *"there is no implicit trust granted to assets or user accounts based solely on their physical or network location"* [27]. Recently, the zero trust concept has been applied to other domains such as supply chain management [17] and IoT networks [18], as well. We refer readers to [17] for a more elaborate definition and discussion for the zero trust concept.

Nyakomitta and Abeka noticed that VPN use does not translate directly to end-to-end security and proposed a layered solution for remote access [19]. Kraev et al. investigated the applicability of MFA in Microsoft's RDP implementation for remote users [20]. In [21], secure desktop sharing with IP tunneling was proposed as a cross-platform solution so that users are given access through a standard browser independent of the operating system in use. In [22], different solutions and protocols such as ICA [23], RFB [24] and RDP [25] for remote desktop were discussed and compared. In [26], the authors described the overall design of a security audit system which employs a proxy server to monitor the RDP, VNC, and X11 sessions.

We noticed that previous work we reviewed were either just on risks and requirements or concentrated on specifics in one particular technology useful for secure remote access. Up to our best knowledge, our work is the first presenting a specific hardware based security solution for remote working together with strong user authentication.

VI. CONCLUSION AND FUTURE WORK

The sudden transition to remote working means former cyber security models and traditional perimeter defense tools and techniques such as firewalls, IDS/IPS systems and subnetworking are no longer sufficient. Organizations seek novel solutions balancing security and user experience so that employees can still easily access what they need for work efficiently without frustrating complexities or tedious steps.

ProGun addresses the security challenges of remote working by a hardware based solution. ProGun USB dongle is cost effective to enable use of BYOD laptops for remote access securely. In this paper, we overview both hardware and software features of ProGun as well as the solution we developed for secure and usable authentication of its remote users. Our risk based authentication solution currently supports rule based policies and keystroke dynamics based user profiling. Our aim is to continue working on this part to bring continuous authentication support to the ProGun solution. Another promising future work is conducting a user study to evaluate the usability (and security) of ProGun on the field. The ultimate goal is full conformance to the zero trust architecture [27].

REFERENCES

[1] Wang, Bin, et al. “Achieving effective remote working during the COVID-19 pandemic: A work design perspective.” Applied psychology 70.1 (2021): 16-59.

[2] Remote Working Statistics, https://www.intuition.com/remote-working-statistics-you-need-to-know-in-2021/ Accessed 3 Sep 2021.

[3] Interpol. Cybercrime: Covid-19 impact. Aug 2020. https://www.interpol.int/en/News-and-Events/News/2020/INTERPOL-report-shows-alarming-rate-of-cyberattacks-during-COVID-19, Accessed 3 Sep 2021.

[4] Shostack, Adam. Threat modeling: Designing for security. John Wiley & Sons, 2014.

[5] Mkosi project, https://github.com/systemd/mkosi, Accessed 29 Oct 2021.

[6] Luks: Disc Encryption, https://guardianproject.info/tr/archive/luks/, Accessed 29 Oct 2021.

[7] Lyastani, Sanam Ghorbani, et al. “Is FIDO2 the kingslayer of user authentication? A comparative usability study of FIDO2 passwordless authentication.” 2020 IEEE Symposium on Security and Privacy (SP). IEEE, 2020.

[8] Verizon 2017 Data Breach Investigation Report, 10th edition, https://enterprise.verizon.com/content/dam/resources/reports/2017/2017_dbir.pdf, Accessed 3 Sep 2021.

[9] Farke, F. M., Lorenz, L., Schnitzler, T., Markert, P., & Dürmuth, M. (2020). “You still use the password after all”–Exploring FIDO2 Security Keys in a Small Company. In Sixteenth Symposium on Usable Privacy and Security (SOUPS 2020) (pp. 19-35).

[10] Owens, K., Ur, B., & Anise, O. (2020). A Framework for Evaluating the Usability and Security of Smartphones as FIDO2 Roaming Authenticators. Proc. WAY.

[11] M’Raihi, D., Machani, S., Pei, M., & Rydell, J. (2011). Totp: Time-based one-time password algorithm. Internet Request for Comments.

[12] Kaczmarek, T., Ozturk, E., & Tsudik, G. (2018, July). Assentication: user de-authentication and lunchtime attack mitigation with seated posture biometric. In International Conference on Applied Cryptography and Network Security (pp. 616-633). Springer, Cham.

[13] Wiefling, S., Iacono, L. L., & Dürmuth, M. (2019, June). Is this really you? An empirical study on risk-based authentication applied in the wild. In IFIP International Conference on ICT Systems Security and Privacy Protection (pp. 134-148). Springer, Cham.

[14] Bicakci, K., Salman, O., Uzunay, Y., & Tan, M. (2020, December). Analysis and evaluation of keystroke dynamics as a feature of contextual authentication. In 2020 International Conference on Information Security and Cryptology (ISCTURKEY) (pp. 11-17). IEEE.

[15] Malecki, F. (2020). Overcoming the security risks of remote working. Computer Fraud & Security, 2020(7), 10-12.

[16] Miller, K. W., Voas, J., & Hurlburt, G. F. (2012). BYOD: Security and privacy considerations. It Professional, 14(5), 53-55.

[17] Collier, Z. A., and J. Sarkis. The zero trust supply chain: Managing supply chain risk in the absence of trust. International Journal of Production Research (2021): 1-16.

[18] Dhar, S., & Bose, I. (2021). Securing IoT Devices Using Zero Trust and Blockchain. Journal of Organizational Computing and Electronic Commerce, 31(1), 18-34.

[19] Peter S. Nyakomitta & Dr. Silvance O. Abeka “Security Investigation on Remote Access Methods of Virtual Private Network” Global Journals, Volume 20 (2020).

[20] Y. Kraev, G. Firsov and D. Kandakov, “Authentication via RDP Using Electronic Identifiers,” 2021 IEEE Conference of Russian Young Researchers in Electrical and Electronic Eng. (ElConRus), 2021, pp. 2361-2365.

[21] Sridhar, S., Sanagavarapu, S., & Chitrakala, S. (2020, July). Cross-Platform Remote Desktop Sharing with IP Tunneling. In 2020 11th International Conference on Computing, Communication and Networking Technologies (ICCCNT) (pp. 1-7). IEEE.

[22] Magaña E, Sesma I, Morató D, Izal M. Remote access protocols for Desktop-as-a-Service solutions. PLoS One. 2019 Jan 4;14(1):e0207512.

[23] Gundarev D. Citrix Internals: ICA Connectivity. April 2014. BriForum, London. 2014.

[24] Richardson T, Levine J. The Remote Framebuffer Protocol. RFC 6143.

[25] Remote Desktop Protocol, architecture and features. Microsoft Developer Network, https://msdn.microsoft.com/en-us/library/aa383015(v=vs.85).aspx, Accessed 29 Oct 2021.

[26] Cui, W., Li, H., Li, W., & An, S. (2012, December). The design and implementation of remote desktop access audit system. In 2012 7th International Conference on Computing and Convergence Technology (ICCCT) (pp. 1239-1243). IEEE.

[27] S. Rose, O. Borchert, S. Mitchell and S. Connelly, Zero Trust Architecture, August. 2020, NIST-SP-800-207, https://csrc.nist.gov/publications/detail/sp/800-207/final, Accessed 30 Oct 2021.