TALLINN UNIVERSITY OF TECHNOLOGY
School of Information Technologies

Kirill Kurkin 201740IVSB

PCI DSS Compliant BYOD Implementation Framework

Bachelor's thesis

Supervisor: Mohammad Tariq Meeran PhD

Tallinn 2023

TALLINNA TEHNIKAÜLIKOOL
Infotehnoloogia teaduskond

Kirill Kurkin 201740IVSB

PCI DSS nõuetele vastav BYOD rakendamise raamistik

Bakalaureusetöö

Juhendaja: Mohammad Tariq Meeran PhD

Tallinn 2023

Author’s declaration of originality

I hereby certify that I am the sole author of this thesis. All the used materials, references to the literature and the work of others have been referred to. This thesis has not been presented for examination anywhere else.

Author: Kirill Kurkin

14.05.2023

Abstract

The aim of this bachelor thesis is to develop a framework for implementing BYOD (Bring Your Own Device) policy that is compliant with the PCI DSS (Payment Card Industry Data Security Standard).

The literature review covers the requirements of the PCI DSS standard and existing literature on BYOD policies. Based on the analysis, an implementation framework for securing mobile devices under a BYOD policy was created.

The resulting framework was designed to be compatible with the PCI DSS standard, so private companies that handle payment card information could use this policy to secure their mobile devices under a BYOD policy implementation. Companies can also customize the framework to fit their unique needs and resources. The proposed framework was tested on three different OSs (operating systems): Windows, Linux and macOS. As a result of test conducted, the framework covers all applicable PCI DSS requirements for mobile devices owned by employees. However, the implementation framework may not be completely suitable for companies that follow other cybersecurity standards due to possible conflicts in requirements or due to the active usage of solutions different from that described in the work.

In summary, this thesis provides a valuable contribution to the development of a PCI DSS compliant BYOD implementation framework. It can help organizations effectively manage the risks associated with the use of personal devices in the workplace while maintaining compliance with industry standards. The framework can be customized to fit the unique needs of individual organizations, and can serve as a basis for creating additional sample security policies for other cybersecurity standards.

This thesis is written in English and is 60 pages long, including 6 chapters, 2 figures.

Annotatsioon

PCI DSS nõuetele vastav BYOD rakendamise raamistik

Käesoleva bakalaureusetöö eesmärk on välja töötada raamistik BYOD (Bring Your Own Device) poliitika rakendamiseks, mis on kooskõlas PCI DSS (Payment Card Industry Data Security Standard) standardiga.

Kirjanduse ülevaade hõlmab PCI DSS standardi nõudeid ja olemasolevat kirjandust BYOD poliitikate kohta. Analüüsi põhjal loodi rakendusraamistik mobiilseadmete turvamiseks BYOD poliitika alusel.

Saadud raamistik oli loodud nii, et see ühilduks PCI DSS-standardiga, nii et maksekaarditeavet töötlevad eraettevõtted saaksid seda poliitikat kasutada oma mobiilseadmete kaitsmiseks BYOD-i poliitika rakendamise alusel. Ettevõtted saavad ka kohandada raamistikku vastavalt oma ainulaadsetele vajadustele ja ressurssidele. Kavandatud raamistikku testiti kolmes erinevas operatsioonisüsteemis: Windows, Linux ja macOS. Läbiviidud testi tulemusel hõlmab raamistik kõiki kohaldatavaid PCI DSS-i nõudeid töötajatele kuuluvatele mobiilseadmetele. Teisi küberturvalisuse standardeid järgivatele ettevõtetele ei pruugi aga rakendusraamistik võimalike nõuete vastuolude või töös kirjeldatust erinevate lahenduste aktiivse kasutamise tõttu täielikult sobida.

Kokkuvõttes annab see lõputöö väärtusliku panuse PCI DSS-iga ühilduva BYOD juurutusraamistiku väljatöötamisse. See võib aidata organisatsioonidel tõhusalt juhtida isiklike seadmete töökohal kasutamisega seotud riske, säilitades samal ajal vastavuse tööstusstandarditele. Raamistiku saab kohandada nii, et see sobiks üksikute organisatsioonide ainulaadsete vajadustega ja see võib olla aluseks muude küberturvalisuse standardite jaoks täiendavate näidisturbepoliitikate loomisel.

Lõputöö on kirjutatud inglise keeles ning sisaldab teksti 60 leheküljel, 6 peatükki, 2 Figuret.

List of abbreviations and terms

ACL	Access Control List
AD	Active Directory
ASV	Approved Scanning Vendor
BIN	Bank Identification Number
BYOD	Bring Your Own Device
CDE	Cardholder Data Environment
DNS	Domain Name System
GPO	Group Policy Object
GUI	Graphical User Interface
HIPAA	Health Insurance Portability and Accountability Act
HTTP	Hypertext Transfer Protocol
HTTPS	Hypertext Transfer Protocol Secure
ICMP	Internet Control Message Protocol
IP	Internet Protocol
IT	Information Technology
Malware	A piece of software designed to infect or cause disruption to a computer or a computer network
MAM	Mobile Application Management
MDM	Mobile Device Management
MFA	Multi-Factor Authentication
NAT	Network Address Translation
NSC	Network Security Control
NTP	Network Time Protocol
OS	Operating System
PAM	Pluggable Authentication Modules
PAN	Primary Account Number
PCI DSS	Payment Card Industry Data Security Standard
PCI SSC	Payment Card Industry Security Standards Council
RDP	Remote Desktop Protocol

VLAN	Virtual Local Area Network
VM	Virtual Machine
VPN	Virtual Private Network
Vulnerability	Weakness in a computer system that can be exploited
XML	Extensible Markup Language

Table of contents

List of figures

1 Introduction

It cannot be denied that IT (information technology) has become an integral part of almost every organization of any size. Usage of information technology by companies implies, but is not limited to: holding their employees' and customers' data, communicating with other companies, communicating with customers and automating daily jobs connected to companies' business process. Information technology drastically increases productivity of employees and whole company performance as a result. Usually, information technology devices are provided by the company itself, though, Bring Your Own Device phenomenon allows employees to bring their own mobile devices to their workplace and use them to achieve their tasks, rather than using officially provided device [1]. "Mobile devices" term in this work refers to mobile phones, tablets and laptops.

Bringing your own device to work means that the company management does not have to spend money on information technology devices for every employee, so they may redirect their cash resources to other aspects of the organization [2]. Though, employees' own devices should still be hardened and all relevant security policies created to prevent cyberattacks. Additionally, companies should not forget that running BYOD network introduces new threats to the cybersecurity, which should definitely be addressed in their security policy.

According to statistics provided by Microsoft in 2013, 67% of people use their own devices at work, even if it is forbidden [3]. It quite clearly shows that using your own mobile device on work has become habitual for everyone. Because of this, implementing BYOD policy is more effective rather than completely banning personal devices at work. Though, it may be challenging for companies to create their own BYOD security policy. Two main reasons may be the lack of knowledge related to cybersecurity by management of a company and the lack of necessary guidance by international cybersecurity standards.

PCI DSS is an information security standard designed to ensure that companies that process, store or transmit credit card information maintain a secure environment with cardholder data [4].

When it comes to PCI DSS, the standard does not separate mobile devices from other computing devices, though, still provides hints for securing them in the "Good Practices" section [5]. However, employees owned devices have completely different risks associated with them and hence should be separated from mobile devices in the context of PCI DSS compliance.

1.1 Problem statement

As stated earlier, usage of BYOD policy in a company not only implies exposure to already existing threats connected to using information technology in general, but also introduces absolutely new threats that are exclusive to the BYOD policy. An example is a probable unpreventable data theft in case of a stolen or lost device. If employee personal mobile device has no proper security controls to wipe work-related data or block access to it when the device is lost, a data breach may occur as anyone who finds a device may access the data [6].

For companies compliant with PCI DSS that wish to implement BYOD policy, there is also a need to implement a comprehensive security framework that is compliant with PCI DSS to ensure the secure use of personal mobile devices for payment cards processing. However, organizations often lack the necessary knowledge and guidance to create a PCI DSS compliant BYOD implementation framework, which can result in increased cybersecurity risks and potential data breaches. As a result, organizations may struggle to properly secure and manage the use of personal mobile devices in the workplace, leaving them vulnerable to various cyber threats.

Therefore, the problem that this thesis aims to solve is the lack of a standardized and practical framework for organizations to implement a PCI DSS compliant BYOD policy. This policy should adequately address the security risks associated with employee-owned mobile devices used for payment cards processing.

1.2 Research questions

- What PCI DSS requirements are applicable for BYOD devices?
- Which of the requirements should be implemented technically, administratively and physically?
- What are the existing PCI DSS implementation frameworks for BYOD?

1.3 Research goal

The research has the following goals:

- Analyse the technical and organizational challenges of implementing a BYOD policy in an organization in compliance with PCI DSS.
- Design a framework for implementing a PCI DSS compliant BYOD policy that addresses the identified challenges.
- Evaluate the effectiveness of the framework through testing of different operating systems.
- Identify areas for future research on PCI DSS compliant BYOD policies.

2 Literature review

To develop a robust and effective BYOD program it is necessary to have a strong theoretical foundation that may be provided by existing research and literature. Reviewing the PCI DSS requirements before reviewing existing works ensures that the proposed BYOD implementation framework is aligned with the standard and that any existing frameworks are evaluated in the context of the standard's requirements with their potential gaps or weaknesses identified.

2.1 Review of PCI DSS

The newest version of the PCI DSS standard – version 4.0 will be used in the thesis. PCI DSS 4.0 includes two approaches: Defined Approach and Customized Approach. Defined Approach allows an organization to implement security controls in order meet the standard requirements by providing an explicit guidance on how to do so [5]. Defined Approach also includes the set of described testing procedures for every requirement that an assessor can use to verify the compliance. The Customized Approach, on the other hand, gives organizations a chance to focus on the objective of a requirement and implement appropriate safeguards without strictly following the requirement of the standard [5]. This gives a company more freedom and space for creativity for implementing safeguards. However, the Customized Approach does not include any steps for testing the implementation because of the uniqueness of each customized implementation [5].

Given this information, the Defined Approach was chosen along with some of its testing procedures to implement and test the implementation of each applicable requirement.

The categorization of controls for applicable requirements of BYOD policy devices has been derived from the HIPAA (Health Insurance Portability and Accountability Act) Security Rule, which outlines the security standards for protecting electronic protected health information [7]. The controls in the Security Rule are clearly classified into three categories, namely: administrative, physical and technical safeguards [8]. The same approach was used to categorize controls for satisfying PCI DSS requirements in this thesis.

2.1.1 Administrative controls

Administrative controls include necessary policies and procedures used to manage information security [9]. Such controls may be implemented by updating the general procedures and documentation of the organization to include BYOD in it. All the principal PCI DSS requirements in [5] were outlined as those that should be adressed with administrative controls. First 11 requirements with their first two sub-requirements and the whole last requirement focus on documenting operational procedures and security policies, as well as documenting roles and responsibilities [5]. This approach helps ensure that an organization has a clear and formal process for managing its cardholder data security program, which is critical to maintaining compliance with PCI DSS.

2.1.2 Physical controls

In context of PCI DSS, physical controls could be viewed as physical security measures implemented to protect buildings and equipment that contain cardholder data [9]. PCI DSS has a specific requirement that adresses physical access, which is "Requirement 9: Restrict Physical Access to Cardholder Data" [5]. The requirement focuses on securing facility security controls and physically securing media with cardholder data [5].

2.1.3 Technical controls

Technical controls are the technologies, such as specialized software and network equipment designed to control access to information and ensure its integrity [8], [9]. All the PCI DSS requirements in [5], except requirement 6 can be covered by using technical controls. Requirement 6 controls the secure development of software, and only administrative controls should be applied to BYOD devices to meet this requirement [5].

The implementation framework in the thesis will focus on the general requirements listed above and their first-level sub-requirements with different types of controls for compliance rather than on every detailed sub-requirement. Nevertheless, the testing steps from sub-requirements will be chosen from every general requirement to ensure the efectiveness of the resulting framework. This will keep the resulting implementation framework as concise as possible.

2.2 Review of similar works

With specific PCI DSS requirements identified to be addressed, it is necessary to gain an insight about the potential challenges associated with creating the PCI DSS compliant implementation framework for BYOD. One effective way that is used in the thesis is to conduct a review of the existing literature on the implementation of PCI DSS compliant BYOD frameworks. The review highlights the approaches and strategies adopted by researchers and practitioners in implementing BYOD policies that comply with the PCI DSS standard.

Thorough review of the literature showed that there is no much work done in the domain of personal devices security for processing payment card data. Keywords used by the author for conducting searches using academical search engines were the following: “PCI DSS”, “BYOD”, “mobile”, “implementation”. However, no results containing specific guidelines for securing BYOD devices according to PCI DSS requirements came up, except one source from the PCI SSC (Payment Card Industry Security Standards Council).

As mobile payments become more popular, merchants are increasingly accepting payments via mobile devices to meet the demands of their customers. However, with the convenience of mobile payments comes the potential security risk, which can lead to data breaches and financial losses. [10]

The PCI SSC has developed specific guidelines to help merchants secure their mobile payment acceptance systems. The guidelines are contained in the “PCI Mobile Payment Acceptance Security Guidelines for Merchants” document. These guidelines are designed to help merchants protect their customers' payment card data while accepting payments via mobile devices [11]. The document includes guidance for securing both the payment-acceptance solution and the mobile device. Only the latter guidance will be reviewed for the reason of being directly connected to the thesis scope.

The very first guideline aims to prevent unauthorized physical access to the device with several suggestion, such as locking the device in a cabinet, tethering it to a counter, or putting it under 24-hour surveillance [11].

To prevent unauthorized logical device access, the document suggests that the merchant restricts logical access to the mobile device to only authorized personnel by using operating system multi-user support that separates user accounts and application data. The merchant should also use logical device access protection methods such as biometrics, complex passwords, or multi-factor authentication. Full device encryption is a measure to potentially prevent users from disabling device-level authentication an provide additional protection in the event of device loss or theft. [11]

Protection of mobile device from malware can be achieved by installing and regularly updating the latest anti-malware software or by employing MAM (Mobile Application Management) or MDM (Mobile Device Management) solutions that can evaluate and remove malicious software. Merchants should not circumvent any security measures on the mobile device and install only the trusted software. [11]

The guidance also recommends that merchants ensure the mobile device is in a secure state by scanning it with security software that can detect excessive app privileges and cleartext password within apps. Disabling USB debugging, enforcing trusted sources and utilize logging and monitoring solutions also contributes to the secure state of the device. Unnecessary device functions should also be disabled to minimize the attack vector. [11]

As the mobile devices can also be lost, there is a need to take actions that would prevent the malicious access in such cases. To achieve this, the document suggests marking the mobile device with unique identifier and recording identifying attributes [11]. Additionally, there should be a process for detecting and reporting the loss or theft of the mobile device [11].

Overall, the “PCI Mobile Payment Acceptance Security Guidelines for Merchants as End-Users” provides a valuable resource for merchants to secure their mobile payment acceptance systems and protect their customers' payment card data. Nevertheless, the document does not specifically cover the BYOD policy devices. Additionally, it does not give the specific solutions that could be implemented, but rather some general hints for securing mobile devices. Though, the hints could serve as a basis for further research on probable solutions for PCI DSS requirements.

3 Methodology

The purpose of this work is to create an implementation framework for BYOD policy that would be compliant with PCI DSS standard.

First of all, a thorough literature review of relevant publications and resources was done to gain an understanding of the PCI DSS and BYOD best practices. As a starting point, it included a review of the PCI DSS with a focus on the security controls it mandates and determining how the requirements of the standard apply to BYOD implementation. The next step done was a review of existing solutions proposed by other researchers, including sources from conference papers, journal papers and credible websites. The collected theoretical background provided additional insights and perspectives on the issue.

Using the information gathered from the literature review, a framework for implementing BYOD policies that comply with PCI DSS was designed. This included guidelines to cover PCI DSS requirements applicable to BYOD devices. After the framework was developed, it was tested by applying it to a test environment to identify any gaps or potential issues, as well as to estimate its effectiveness as a whole. The tests were done according to the test steps that are described in the sub-requirements of PCI DSS on a range of devices including Windows PCs, Macs, and Linux machines to ensure the framework compatibility with different operating systems.

4 Implementation framework

When choosing an appropriate method for securing BYOD devices, organization should also bear in mind that it is necessary to ensure that the personal data of employees cannot be accessed by the organization’s computing environment and leaked in the working process. For this, organizations can use VMs (virtual machines) to separate business and personal data on employee-owned devices, providing a more secure environment for sensitive business information [12]. By creating a separate VM for business data, organizations can ensure that confidential data is not exposed to personal applications or malware on the employee's device. The business VM can be configured with the necessary security controls and monitoring tools, such as firewalls and intrusion detection systems, to help protect against unauthorized access and data breaches. Employees can access the business VM securely from their personal device, while the rest of the device remains separate and unconnected to the business VM. This approach enables organizations to implement a BYOD policy, while minimizing the risk of data breaches and maintaining compliance with data protection regulations.

VMs in an organizational environment could be provisioned in two ways. The first possibility would be that employees get the preconfigured VM image and install it locally on their machine. Another possible way for the company to use VMs is to host them on the company’s server and remotely deploy virtual machines or virtual desktops to the users [13]. Allowing users to host virtual machines themselves may appeal to some employees, as they can personalize their machines. Additionally, employers do not have to maintain any virtual machines in this case and pay any costs connected to the VM hosting and maintenance. Alternatively, if virtual machines are deployed by the company on its server, the organization can ensure that all the necessary configurations are done in the right way and were not altered, making virtual machines less prone to data breaches.

While both methods may have their own advantages and disadvantages, in the context of PCI DSS it is necessary to compare how well each type of deployment may cover PCI DSS requirements. VM deployed on an employee’s personal devices will not be able to cover all the outlined requirements. Namely, the implementation of requirement 3 is not possible to control as the employee may be able to copy PAN (Primary Account Number) and use it on their device. Physical controls covering requirement 9 cannot be implemented at all due to the fact that organizations cannot enforce and control physical security of employee’s own device due to the privacy reasons. Finally, security actions that require the availability of the machine at specific time, such as scheduled vulnerability and malware scans may be skipped if the employee’s personal computer was turned off. For a VM hosted on an organizational server, these limitations are not present, as the organisation has the full control of their server.

Having considered the information above, it can be concluded that allowing users to host VMs themselves introduces several limitations, which may imply a huge risk to the overall security of the company. Some of them may be avoided if the hypervisor is configured to prevent any changes to the virtual machine. Though, it may be challenging to implement, as users with administrative access should still be able to modify everything on their machine. Additionally, some requirements, such as physical security requirements are impossible to implement if VM is hosted on the employee’s personal device. Even though the cardholder data may be on a virtual drive, the virtual drive is still present on the employee’s machine. This may raise the issue of the employee’s personal device being in scope of the PCI DSS. And indeed, PCI DSS Virtualization Guidelines document clearly defines that virtual machine in scope of PCI DSS automatically means that its hypervisor will also be in scope of PCI DSS and employee’s personal device as a result [14]. For this reason, the scenario with VM deployed on the server of an organization will be used in this work. Especially the technical controls will be paid attention to.

4.1 Administrative controls

Administrative controls are mostly written documents that contain the best security practices, as well as various company policies, including the security policy. Companies may keep these documents either on the shared drive with permissions configured or store them in a cloud service, such as Microsoft Sharepoint that allows sharing files and setting permissions on who can access them [15]. This may also be useful if a company already uses other Microsoft 365 solutions.

Another possibility is to use Atlassian Confluence software as a corporate wiki to share knowledge among employees [16]. It would also allow employees to use the software in their work and create documentation that could be shared with others.

The choice of the solution to store company policies is important, but attention should be paid that these policies also include BYOD usage policies if BYOD devices are allowed in the company. This policy for BYOD may be created as a standalone policy or incorporated into the existing security policy of a company. It may also be a good idea to create separate rules for employees that would like to work using their own devices and give them access to the virtual machine only after they have acknowledged the rules by signing them, either digitally or on paper.

4.2 Physical controls

The physical controls in case of virtual machines apply to the server that is hosting them. This is due to the huge risk of harm that can be done if an attacker gains access to all VMs and network via only one physical host [14]. To prevent this, appropriate physical controls should be in place, which are:

- Place CCTV cameras in front of the room with the server that hosts VMs.
- Use locks or cards to give physical access to the server only to server administrators.
- Disable all the unused ports of the server.

As the security of servers is not the main focus of the thesis, only general guidelines are provided that could serve as a basis for further server physical security enforcement for organizations.

However, there is present one another component of physical security besides the server, which is the personal computer of an employee that connects to the virtual machine. As companies cannot fully control how and where their employees use personal devices, only recommendations for the secure use of devices could be provided. This may include:

- Do not leave your device unattended.
- Use carrying cases when transporting devices to prevent physical damage.
- Use screen privacy filters when in public spaces.

4.3 Framework for Windows

The following frameworks for Windows, Linux and macOS machines will include the PCI DSS requirement and guidelines on how to comply with the requirement. Default built-in and open-source solutions will be given a preference. If a requirement cannot be satisfied using a default or open-source solution, multiple other possibilities will be given with some examples on how to comply with the requirement using one of them.

For framework testing purposes, the following requirements were chosen for all of the OSs:

- Requirement 1.3
- Requirement 1.5
- Requirement 3.4
- Requirement 5.3
- Requirement 7.3
- Requirement 8.3
- Requirement 10.2
- Requirement 10.3
- Requirement 10.6

In this framework, solutions for Windows machines are compatible with Windows Server 2019 for setting policies and Windows 10 machine as client operating systems. This means that the solution may not correctly work on older versions and have configurations names changed for newer or older versions.

Requirement "1.3 Network access to and from the cardholder data environment is restricted" [5].

This requirement dictates that inbound and outbound connections should be restricted to the necessary only [5]. This can be achieved with the help of Group Policies set up by domain administrator on a domain controller. More details are presented in Appendix 2.

When the virtual machine joins the domain, the requirement will be satisfied, as group policy will be automatically applied to it.

Requirement “1.4 Risks to the CDE from computing devices that are able to connect to both untrusted networks and the CDE are mitigated” [5].

This requirement dictates that system components with cardholder data must not be directly accessible from untrusted networks with NSCs (Network Security Controls) implemented between trusted and untrusted networks [5]. CDE (cardholder data environment) on the network can be secured with the following steps:

1. Implement a physical or virtual firewall between the trusted virtual machines servers segment and untrusted networks.
2. Configure the firewall to restrict any inbound traffic from untrusted networks to servers with VMs, except stateful responses originating from the trusted network using ACLs (access control lists).
3. Additionally implement VLAN (virtual local area network) to isolate servers with VMs from other segments.
4. Implement a VPN (virtual private network) solution to control access to the trusted network from untrusted networks, so that users working with their own device from home should use VPN to connect to the internal network and be able to further connect to their VM.

When this requirement is satisfied, the segment of the network that contains servers with virtual machines will be isolated from other segments with the help of network devices. As the further management and maintenance of these network devices within PCI DSS compliance will be then done in a similar way as network devices within other segments, they will not be considered in further requirements.

Requirement “1.5 Risks to the CDE from computing devices that are able to connect to both untrusted networks and the CDE are mitigated” [5].

This requirement dictates that security controls must always be running and must not be alterable by users [5]. This can be achieved with the help of Group Policies set up by domain administrator on a domain controller by following these steps:

1. In the Group Policy Management tool, create or edit a GPO.
2. Navigate to “Windows Defender Firewall” administrative template.

3. Enable “Windows Firewall: Protect all network connections” policy for both domain and standard profiles.
4. Apply the GPO to the Organizational Unit containing VMs.

When the virtual machine joins the domain, the requirement will be satisfied, as group policy will be automatically applied to it.

Requirement “2.2 System components are configured and managed securely” [5].

This requirement dictates that only necessary services and functions are enabled, non-console administrative access is secured with encryption and vendor default accounts are removed or their passwords changed if in use [5]. The part with services management of the requirement can be satisfied with the help of Group Policies set up by domain administrator on a domain controller by following these steps:

1. In the Group Policy Management tool, create or edit a Group Policy Object.
2. Navigate to “System Services” setting.
3. Change the start-up mode to disabled for the services that will not be used on virtual machines.

When the virtual machine joins the domain, the described part of the requirement will be satisfied, as group policy will be automatically applied to it. To comply with other parts of the requirement, follow these guidelines:

1. Use built-in Windows RDP (Remote Desktop Protocol) to connect users to the virtual machines, as Remote Desktop sessions are encrypted and prevent malicious actors from viewing remote sessions by using eavesdropping attack [17].
2. Delete all the unused vendor default accounts before providing employees access to the VMs.
3. Change passwords for all the vendor default accounts that will be used to the secure ones with a minimum length of 12 alphanumeric characters before providing employees access to the VMs.
4. Do not install any services on the VMs that will change their primary function, which should be a simple work with documents, including on the Internet, and development.

Requirement “3.4 Access to displays of full PAN and ability to copy PAN is restricted” [5].

This requirement dictates that only four last digits and bank identification number BIN (bank identification number) of PAN may be displayed, except for personnel with business need, and that PAN cannot be copied or moved when using remote-access technologies [5]. This can be achieved with the following steps:

1. Create a shared network drive for employees or mount an existing one to the VMs.
2. Create a folder or multiple ones that will contain files with PAN and provide necessary access to the authorized personnel in the folder properties tab based on the roles of employees. Users can be divided into groups in AD (Active Directory) based on their roles and groups can be given access to files then.

Additionally, the following steps should be taken to prevent users from copying PAN in the remote session:

1. In the Group Policy Management tool, create or edit a Group Policy Object.
2. Navigate to "Device and Resource Redirection" administrative template.
3. Enable "Do not allow Clipboard redirection" policy.
4. Apply the GPO to the Organizational Unit containing VMs.

When the virtual machine joins the domain, the requirement will be satisfied, as group policy will be automatically applied to it.

Requirement "4.2 PAN is protected with strong cryptography during transmission" [5].

This requirement dictates that PAN is encrypted when sent via end-user messaging solutions [5]. These guidelines will help in complying with the requirement:

1. Enforce users administratively to use emails for sending sensitive data, including PAN. This can be defined in the acceptable use policy and security policy.
2. Enforce users administratively to encrypt emails with PAN. The most popular email service providers, Gmail and Outlook, have options to encrypt emails.

Requirement "5.2 Malicious software (malware) is prevented, or detected and addressed" [5].

This requirement dictates that the anti-malware solution is deployed and is able to detect and remove known types of malware [5]. The requirement can be satisfied by installing any popular antivirus of your choice on virtual machines or leaving the default Microsoft Defender antivirus, as it performs necessary functionality and easily modifiable with group policies.

Requirement “5.3 Anti-malware mechanisms and processes are active, maintained, and monitored” [5].

This requirement dictates that anti-malware solution is automatically updated, performs periodic scans or behavioural analysis, performs automatic scans or behavioural analysis when removable media is inserted and cannot be altered [5]. All these sub-requirements can be satisfied with the help of Group Policies set up by domain administrator on a domain controller. More details are presented in Appendix 3.

When the virtual machine joins the domain, the requirement will be satisfied, as group policy will be automatically applied to it. Please note, that in newest versions of Windows the option named “Tamper Protection” should be manually turned off for group policies to work as intended. This option prevents changes to several system security settings when turned on, which results in inability for group policies to control some aspects of the Windows Defender Antivirus [18].

Requirement “7.3 Access to system components and data is managed via an access control system(s)” [5].

This requirement dictates that access control system is in place and grants access based on a user’s need to know and role with default deny permission [5]. This can be achieved with the following steps:

1. In the Shares section in Server Manager, turn “Enable access-based enumeration” setting on for all your shared network drives.
2. Remove default allow permissions for everyone and grant access to folders for groups based on their need for access.

Requirement “8.2 User identification and related accounts for users and administrators are strictly managed throughout an account’s lifecycle” [5].

This requirement dictates that user session cannot be idle for more than 15 minutes [5]. This can be achieved with the help of Group Policies set up by domain administrator on a domain controller by following these steps:

1. In the Group Policy Management tool, create or edit a Group Policy Object.
2. Navigate to “Security Options” setting.
3. Change “Interactive logon: Machine inactivity limit” policy to 900 seconds (15 minutes) or less.
4. Apply the GPO to the Organizational Unit containing VMs.

When the virtual machine joins the domain, the requirement will be satisfied, as group policy will be automatically applied to it.

Also note, that access to VMs and all other permissions to company resources has to be granted to domain users with necessary policies set up by domain administrator, but not local users in order to cover other requirements as well.

Requirement “8.3 Strong authentication for users and administrators is established and managed” [5].

This requirement dictates that users are locked out to a minimum of 30 minutes after not more than 10 logon attempts, are forced to change uniquely assigned password at first logon, have passwords with minimum length of 12 alphanumeric characters if they are used, cannot use the same password as any of their 4 last passwords if passwords are used and have their passwords changed at least every 90 days if passwords is the only authentication method [5]. All these sub-requirements can be satisfied with the help of Group Policies set up by domain administrator on a domain controller by following these steps:

1. In the Group Policy Management tool, create or edit a Group Policy Object.
2. Navigate to “Account Lockout Policy” setting.
3. Set “Account lockout threshold” policy setting to 10 attempts, “Account lockout duration” to 30 minutes and “Reset account lockout counter after” to 30 minutes.
4. Navigate to “Password Policy” setting.
5. Set “Enforce password history” to 4 passwords, “Maximum password age” to 90 days, “Minimum password age” to any based on your company security policy, “Minimum

password length” to 12 characters and enable “Password must meet complexity requirements” policy.

When the virtual machine joins the domain, the requirement will be satisfied, as group policy will be automatically applied to it. In addition, to enforce users to change their passwords at first logon, assign a secure random password for a new user based on this guideline, and enable “User must change password at next logon” account option when creating users.

Requirement “8.4 Multi-factor authentication (MFA) is implemented to secure access into the CDE” [5].

This requirement dictates that all access into the CDE should use MFA (multi-factor authentication) [5]. The most convenient way to implement MFA is to use smartcards, since smartcards together with their PIN provide a two-factor authentication [19]. To comply with the requirement in case server hosting VMs acts as a CDE, follow these steps:

1. Setup smartcards infrastructure and download the root certificate.
2. In the Group Policy Management tool, create or edit a Group Policy Object.
3. Navigate to “Trusted Root Certification Authorities” setting and import the certificate.
4. Navigate to “Security Options” setting and enable “Interactive logon: Require Windows Hello for Business or smart card” policy.
5. Apply the GPO to the Organizational Unit containing VMs.

When the virtual machine joins the domain, the requirement will be satisfied, as group policy will be automatically applied to it.

Requirement “10.2 Audit logs are implemented to support the detection of anomalies and suspicious activity, and the forensic analysis of events” [5].

This requirement dictates that audit logs are enabled for all system components and captures invalid attempts for logical access, management of accounts, actions with elevated access, actions with system objects, reading and changing the flow of audit logs [5]. This requirement can be satisfied by using default Windows Event Viewer. It acts as a log collector for all the necessary event logs. To audit all necessary actions, follow these steps:

1. In the Group Policy Management tool, create or edit a Group Policy Object.
2. Navigate to “Audit Policy” setting and enable successful and failed attempts for all the events as necessary according to your security policy.
3. Navigate to “Audit Policies” setting and enable the following policy settings: “Audit File System”, “Audit Kernel Object”, “Audit Sensitive Privilege Use”, “Audit Credential Validation”, “Audit Other Account Logon Events”, “Audit User Account Management”, “Audit Audit Policy Change”.
4. Apply the GPO to the Organizational Unit containing VMs.
5. Enable the “Audit File System” policy setting for the server that shares the folder either by creating a new policy or modifying the existing one.

When the virtual machine joins the domain, the requirement will be satisfied, as group policy will be automatically applied to it. You may now forward events to the event logs collector. Please note that for steps 2 and 3 it is recommended to enable both successful and failed attempts for all settings, but these events can be customized based on your business need and standard requirements, as allowing all events may quickly fill the event log with regular events [20]. For example, it is only required to capture invalid logical access by PCI DSS, so you must enable “Audit Credential Validation” setting with at least failed events [5].

In addition, to make sure that audit logs are generated for sensitive files, it is also necessary to add users and actions that will be audited for a specific folder or file. To accomplish this, go to the properties of a folder or a file that will be audited, open the “Advanced Security Settings” setting and specify users and actions in the “Auditing” tab that will be audited. Specify type as “All” to make sure that both failed and successful attempts to access the data are audited.

Requirement “10.3 Audit logs are protected from destruction and unauthorized modifications” [5].

This requirement dictates that audit logs can be read only by employees with business need, cannot be altered and are backed up [5]. As described earlier, event logs can be forwarded to the central event logs collector that acts as a backup for logs. To comply with other part of the requirement, follow these steps:

1. Deny all the permissions to all regular users for the location of event logs.
2. In the Group Policy Management tool, create or edit a Group Policy Object.
3. Navigate to “Restricted/Permitted snap-ins“ administrative template and disable the use of “Event Viewer“ and “Event Viewer (Windows Vista)” snap-ins.
4. Apply the GPO to the Organizational Unit containing VMs.

When the virtual machine joins the domain, the requirement will be satisfied, as group policy will be automatically applied to it.

Requirement “10.6 Time-synchronization mechanisms support consistent time settings across all systems” [5].

This requirement dictates that system time is synchronized with the use of time-synchronization technology [5]. For these purposes, the NTP (Network Time Protocol) will be used. This can be done with Group Policies set up by domain administrator on a domain controller by following these steps:

1. In the Group Policy Management tool, create or edit a Group Policy Object.
2. Navigate to “Time Providers“ administrative template.
3. Turn “Enable Windows NTP Client“ setting on and provide details of NTP server for the “Configure Windows NTP Client“ setting.
4. Apply the GPO to the Organizational Unit containing VMs.

When the virtual machine joins the domain, the requirement will be satisfied, as group policy will be automatically applied to it.

Requirement “10.7 Failures of critical security control systems are detected, reported, and responded to promptly” [5].

This requirement dictates that failures of security controls are detected and addressed promptly [5]. If all the components described above in this framework are correctly configured, they will generate event logs that can be seen in event viewer and that are send to central logs collector in case the failure of security control system happens. These events should generate alerts, for example by sending emails to system administrators with the help of Powershell script.

Requirement “11.3 External and internal vulnerabilities are regularly identified, prioritized, and addressed” [5].

This requirement dictates that internal vulnerability scans are performed every three months, including authenticated scanning [5]. To comply with the requirement, you should setup the vulnerability scanning process in your organization. There are a lot of vulnerability scanners available, so the choice depends on the need of organization. When the vulnerability scanning process is set up, you can easily add IP (Internet Protocol) addresses of VMs for the internal scanning. For authenticated scans, administrators should create the system account that can login into VMs. It is recommended to use the solution from PCI SSC ASV (Approved Scanning Vendor), as it will be also used for external scans as a part of a requirement. Having only one solution for vulnerability scanning makes it easier to manage vulnerability scanning process.

Testing the Windows framework: To test effectiveness of the resulting implementation framework, a Windows server with “dc1” hostname was created first. It acts as a domain controller and an NTP server. This is needed due to the fact that the implementation framework relies on group policies. The server was given a static IP address to ensure that no functionality breaks that relies on the hardcoded IPs in configuration.

The initial setup of the domain environment was done in the very beginning. It involved promoting the server to domain controller and creating the “framework.test” domain, creating two users: “employee1” and “employee2”. An Organizational Unit “VMs” was created that contains both users and the Windows client computer used for testing. Additionally, a “shared_folder” share was created that contains two folders: “CDE” with sensitive credit cards data and “Work files” that contains files that employees store and share with each other. Inside the “CDE” folder a file “card_data” was created. It is expected that both of created users can access “Work files” folder and create and delete files in there, but only “employee1” has the read-only access to the “CDE” folder and the file inside it. For this, permissions were configured in the properties of folders. To comply with the auditing requirement, the “CDE” folder was included in the auditing with all possible successful and failed actions by all users.

After the initial setup, all the configurations of group policies for chosen requirements were done according to the implementation framework. For firewall testing purposes, all ports, protocols and apps were selected as allowed for both incoming and outgoing connections with the ICMP (Internet Control Message Protocol) protocol explicitly blocked for outgoing connection. This means that the client will not be able to ping other hosts. Group Policy Object was linked to the “VMs” Organizational Unit. Next, the Windows 10 client VM was run. The local user “local_user” was created for the initial setup. The next step done was change the Ethernet adapter properties and use our domain controller as a DNS (Domain Name System) server. Then, the machine was joined to the domain using domain administrator credentials.

After signing in to the “employee1” account with created password, the system immediately required the password to be changed. No passwords were accepted if they did not correspond to the requirement 8.3. Therefore, requirement 8.3 was successfully satisfied.

After the successful sign in with the new secure password, the firewall was opened and tried to be turned off. The option to turn firewall on and off is greyed out, so firewall is on and users cannot turn it off. Therefore, requirement 1.3 was successfully satisfied.

In addition, the created outbound rule to block ICMPv4 traffic was tested by trying to ping the IP address of “dc1” from the command prompt. The echo request failed as suggested by the output that displayed general failure text. Therefore, requirement 1.5 was successfully satisfied.

The next system component to be tested was the Microsoft Defender Antivirus. By default, it was allowed to turn off the firewall by elevating privileges. As mentioned in the framework, this is due to the tamper protection option. After the tamper protection was turned off, the real-time protection option became greyed out. Therefore, requirement 5.3 was successfully satisfied.

Next, the access to the shared folder was tested. The CDE folder was accessible to “employee1” and the file contents inside could be read. The file could not be deleted or changed and no new files or folders could be created. Inside the “Work files” folder, the folder for “employee1” files with a sample file was created. According to the testing scenario, access works as expected for the “employee1” user. The following step was to test the access for the “employee2” user. The current user was signed out and “employee2” account was used to sign in. The password for “employee2” was also changed and the login was successful. The user could successfully read the file of "employee1" and create their own folder for files. However, the "CDE" folder was not even visible to the user. As the access controls were working properly, requirement 7.3 was successfully satisfied.

In the previous testing step, the "employee1" user accessed the file with sensitive data. This event should have been audited according to the group policy set up. To see the corresponding audit event, the Event Viewer was opened on the domain controller server. The audit event was found that correctly captured the successful attempt to access the sensitive file by "employee1". When the Event Viewer was tried to be opened on the client VM, the error message was displayed, informing the user that the action is restricted by the group policy. Consequently, requirements 10.2 and 10.3 were successfully satisfied.

To make sure that the client correctly synchronizes time with the NTP server, the following command was run in the command prompt:

```
w32tm /query /peers
```

As an output of the command, the IP of the domain controller was shown as an NTP peer with the active state. Therefore, requirement 10.6 was successfully satisfied.

The last requirement left that was not yet tested was requirement 3.4. To correctly test the effectiveness of the implementation framework for this requirement, the remote connection was needed to be set up. This is also the initial solution proposed in this work and therefore should definitely be tested. First of all, the VPN was set up on the Windows server using Remote Access feature [21]. NAT (Network Address Translation) was also set up, as the connection would happen from the external network and public IP address will be used. The Point-to-Point Tunneling Protocol (PPTP) was used as a VPN protocol for demonstration purposes, even though it has weak security and should be replaced with more secure protocol in the production environment [22]. Secondly, the remote connection was allowed on the Windows client. This was done by editing a GPO on the Windows server and enabling the "Allow users to connect remotely by using Remote Desktop Services" policy in the "Connections" setting. Then both users were added to the local "Remote Desktop Users" security group. Next, the laptop with Windows 10 that acts as an employee's remote device connecting to the company local network through

VPN was connected to the mobile data to ensure it is not on the same network as the Windows 10 VM. Then the laptop was connected to the corporate network through VPN using “employee1” credentials. Finally, the Remote Desktop session was successfully launched using “employee1” credentials. Now, the requirement 3.4 could be tested. For this, a sample text file with some text was created on the remotely connected VM. The second text file was created on the laptop with no contents. The contents of the file from the remote connection were copied, but could not be pasted to the text file on the laptop. Therefore, requirement 3.4 was successfully satisfied.

4.4 Framework for Linux

This framework includes solutions for Ubuntu 22.04 machine. This means that commands and configuration files locations may differ from those that are present on older or newer versions of Ubuntu. Additionally, these solutions may not be completely implementable on other Linux distributions.

Requirement “1.3 Network access to and from the cardholder data environment is restricted” [5].

This requirement dictates that inbound and outbound connections should be restricted to the necessary only [5]. This can be achieved by following these steps:

1. Using a firewall tool of your choice, such as “iptables” or “ufw”, create rules to allow necessary inbound and outbound traffic according to your needs.
2. Configure default firewall behaviour to deny all inbound and outbound traffic or create rules to deny all inbound and outbound traffic and place them as the last rules.

Requirement “1.4 Risks to the CDE from computing devices that are able to connect to both untrusted networks and the CDE are mitigated” [5].

This requirement dictates that system components with cardholder data must not be directly accessible from untrusted networks with NSCs implemented between trusted and untrusted networks [5]. As already described in the framework for Windows, this can be done by network segmentation using network devices. Once it is done and configured properly, all the systems in the segment will be secured.

Requirement “1.5 Risks to the CDE from computing devices that are able to connect to both untrusted networks and the CDE are mitigated” [5].

This requirement dictates that security controls must always be running and must not be alterable by users [5]. By default, in Linux environment, non-root users will not be able to change firewall settings and see rules. Though, permissions on firewall configuration files should still be removed from users. To further secure firewall, you may use file permissions to restrict access to the command executable file. This can be done with the following command:

```
sudo chmod o-x <path to command>
```

This command removes the execute permissions for all other than the owner and group users for the command executable file at the given location. As a result, users without execute permissions will not be able to execute the command.

Requirement “2.2 System components are configured and managed securely” [5].

This requirement dictates that only necessary services and functions are enabled, non-console administrative access is secured with encryption and vendor default accounts are removed or their passwords changed if in use [5]. To disable unnecessary services in Linux, run the following command:

```
sudo systemctl disable <service>
```

To comply with other parts of the requirement, follow these guidelines:

1. Delete all the unused vendor default accounts before providing employees access to the VMs.
2. Change passwords for all the vendor default accounts that will be used to the secure ones with a minimum length of 12 alphanumeric characters before providing employees access to the VMs.
3. Do not install any services on the VMs that will change their primary function, which should be a simple work with documents, including on the Internet, and development.
4. Install xrdp package to the machine. It will enable both Windows and Linux users to connect to the virtual machine running Ubuntu Linux over an encrypted channel.

Requirement “3.4 Access to displays of full PAN and ability to copy PAN is restricted” [5].

This requirement dictates that only four last digits and BIN of PAN may be displayed, except for personnel with business need, and that PAN cannot be copied or moved when using remote-access technologies. This can be achieved with the following steps:

1. Disable clipboard redirection for “xrdp” server by setting “cliprdr” setting to “false” in configuration file at the location “/etc/xrdp/xrdp.ini” [23].
2. Restart the “xrdp” service.

Requirement “4.2 PAN is protected with strong cryptography during transmission” [5].

This requirement dictates that PAN is encrypted when sent via end-user messaging solutions [5]. For guidelines on how to comply with the requirement, refer to the framework for Windows requirement 4.2 implementation. For Linux, you can use the same email service providers as for Windows. The same type of administrative control can also be used for both Linux and Windows system, which can be either acceptable use policy or security policy or a combination of both.

Requirement “5.2 Malicious software (malware) is prevented, or detected and addressed” [5].

This requirement dictates that the anti-malware solution is deployed and is able to detect and remove known types of malware [5]. The requirement can be satisfied by installing any popular antivirus, such as Kaspersky Endpoint Security for Linux or GravityZone Business Security [24], [25]. In contrast to these two antivirus solutions, there is also a free open-source antivirus engine ClamAV available for Windows, Linux and Mac [26]. The choice depends on your business needs and finances.

Requirement “5.3 Anti-malware mechanisms and processes are active, maintained, and monitored” [5].

This requirement dictates that anti-malware solution is automatically updated, performs periodic scans or behavioural analysis, performs automatic scans or behavioural analysis when removable media is inserted and cannot be altered [5]. Most anti-malware solutions will update automatically if such option is not turned off. Anti-malware behaviour for automatic scans and external media scans varies from vendor to vendor. If the solution does not provide a possibility for automatic scans, including those for external media when it is plugged in, you must compensate this with bash scripts, which are able to perform periodic scans with the help of “cron” jobs and scan external media when it is plugged in and a “udev” rule is activated. To prevent users from altering or disabling anti-malware solution, you may simply remove execute permission for the command executable file as already described for requirement 1.5. Additionally, you may implement Ansible or similar configuration management and application deployment software on one of your Linux servers and centrally manage VMs with it [27]. It will allow to push latest anti-malware solution updates to VMs and make sure that they have all the necessary scripts and “cron” jobs.

Requirement “7.3 Access to system components and data is managed via an access control system(s)” [5].

This requirement dictates that access control system is in place and grants access based on a user’s need to know and role with default deny permission [5]. This can be achieved by following these steps:

1. Create groups and add users there based on their job roles.
2. Granting permissions for files only for those group that need access to it. Remove all permissions for other users.

Even though default Linux permissions are quite simple, there is a way to get a more finer-grained control of permissions with the help of Linux ACLs [28].

It is more convenient to have the same user accounts as in Windows AD and be able to use the shared folder that is hosted on a Windows server we created before. To access the shared folder with the same credentials you use to login into Linux, follow these steps:

1. Use “realmd” package and join Linux machine to the domain, so it can use user accounts from the Active Directory. Passwords for user accounts will be bidirectionally synchronized.
2. Mount a shared network drive that was created for the Windows framework.

You will now be able to work with files in the shared network drive and have the same permissions as configured by the system administrator on the Windows folder properties tab. Alternatively, you may create a network share on one of your Linux servers using Samba and mounting it on Linux VMs.

Requirement “8.2 User identification and related accounts for users and administrators are strictly managed throughout an account’s lifecycle” [5].

This requirement dictates that user session cannot be idle for more than 15 minutes [5]. This can be achieved with the help of “dconf” locks in Linux. For this, you will need to set the “idle-delay” key on the “org/gnome/desktop/session” path to 900 seconds by creating and editing a file in the “/etc/dconf/db/local.d” directory, then lock “/org/gnome/desktop/session/idle-delay” key, as well as the “/org/gnome/desktop/screensaver/lock-delay” key in the “/etc/dconf/db/local.d/lockdown” configuration file [29]. This will set the default idle period to 15 minutes and prohibit users from changing this setting.

Requirement “8.3 Strong authentication for users and administrators is established and managed” [5].

This requirement dictates that users are locked out to a minimum of 30 minutes after not more than 10 logon attempts, are forced to change uniquely assigned password at first logon, have passwords with minimum length of 12 alphanumeric characters if they are used, cannot use the same password as any of their 4 last passwords if passwords are used and have their passwords changed at least every 90 days if passwords is the only authentication method [5].

When new users are created, assign a random secure password to them based on the secure password features provided in this requirement. Then, issue the following command:

```
passwd --expire <username>
```

This command expires the password that is assigned to user and will force them to change the password on their next login [30].

In order to enforce secure passwords, you should edit the “/etc/pam.d/common-password” file and configure passwords to include minimum 12 characters, include at least 1 lowercase letter and include at least 1 digit with the history of 4 passwords [31].

To set the password maximum age to 90 days, edit the "/etc/login.defs" file and change the maximum age in days value to 90 [32]. Please note, that this policy applies only to newly created account, so the file must be modified before creating users [32]. Otherwise, it is needed to manually issue a command "chage" and set password maximum age for every created user [32].

Finally, to comply with the lockout requirement, you should edit the "/etc/pam.d/common-auth" file to deny login attempts after 10 failed ones and set the unlock time to 1800 seconds (30 minutes) by using "pam_faillock" module that specifically keeps a record of failed login attempts. [33], [34].

Requirement "8.4 Multi-factor authentication (MFA) is implemented to secure access into the CDE" [5].

This requirement dictates that all access into the CDE should use MFA [5]. As mentioned earlier in the framework for Windows, smartcards can be used to comply with the requirement. To be able to use smartcards for login on Linux, you would need to follow these steps after you have set up the smart cards infrastructure:

1. Install drivers for smartcards and smartcard readers.
2. Install the PAM (Pluggable Authentication Modules) module to allow X.509 certificate logins [35].
3. Configure the PAM module and add it to PAM stack [35].
4. Set the certificate authority on the machine [35].
5. Map certificates to usernames [35].

This is a general overview of the process that does not include many factors that are unique for each organization.

Requirement "10.2 Audit logs are implemented to support the detection of anomalies and suspicious activity, and the forensic analysis of events" [5].

This requirement dictates that audit logs are enabled for all system components and captures invalid attempts for logical access, management of accounts, actions with elevated access, actions with system objects, reading and changing the flow of audit logs [5]. To comply with the requirement, you may use the "auditd" package for Linux, which is a tool responsible for writing audit log files to the disk [36]. With "auditd" package you can specify rules for files and actions you would like to capture access attempts to [36]. Additionally, you can send audit logs generated with this package to the remote “syslog” server for central collection and analysis of logs [37]. In addition to “auditd” logs, “var/log/” collects many other useful logs that you may need to collect based on your business need.

Requirement “10.3 Audit logs are protected from destruction and unauthorized modifications” [5].

This requirement dictates that audit logs can be read only by employees with business need, cannot be altered and are backed up [5].

As described in the implementation guide for requirement 10.2, you can forward logs to the central logs collector server. This can be done with “rsyslog” system as an example. This way, logs can be backed up from the central logs collector.

To make sure that logs cannot be altered and accessed by unauthorized users, you must set the read-only permissions for the groups of users that need access to logs and remove all permissions for all other users.

Requirement “10.6 Time-synchronization mechanisms support consistent time settings across all systems” [5].

This requirement dictates that system time is synchronized with the use of time-synchronization technology [5]. New Ubuntu versions have the “timesyncd” package installed by default, which is used to synchronize time over a network [38]. To enforce time-synchronization, you should simply change the package configuration by adding your NTP server IP address to it and restarting the service.

Requirement “10.7 Failures of critical security control systems are detected, reported, and responded to promptly” [5].

This requirement dictates that failures of security controls are detected and addressed promptly [5]. With all the components set up, logs will be generated that will be sent to the central logs collector server. This server must analyse all the incoming logs and generate alerts if critical components fail on some server. This can be done either by manually writing predefined rules or taking advantage of machine learning.

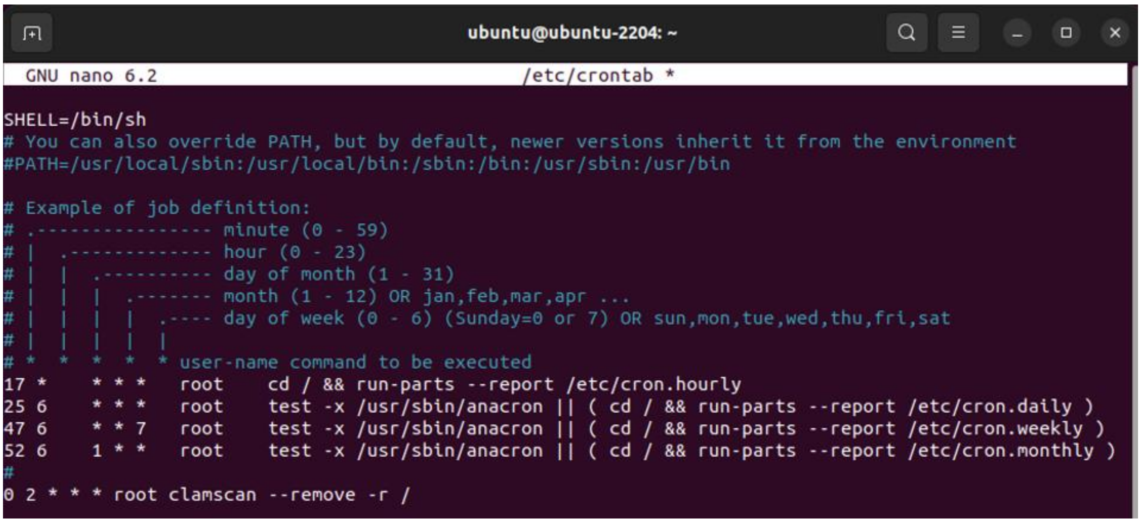
Requirement “11.3 External and internal vulnerabilities are regularly identified, prioritized, and addressed” [5].

This requirement dictates that internal vulnerability scans are performed every three months, including authenticated scanning [5]. As with the Windows platform, there are many vulnerability scanners capable of scanning Linux platforms. You may need to install agents, which may be deployed and managed with the help of Ansible.

Testing the Linux framework: To test the implementation framework, the Ubuntu virtual machine was created.

For firewall testing purposes the rule was created to allow only HTTPS (Hypertext Transfer Protocol Secure) and DNS traffic using “ufw” commands. To avoid denying traffic that may be used to comply with other requirements in the future, the firewall rules were tested with the superuser account and later reset. A website that operates on both HTTP (Hypertext Transfer Protocol) and HTTPS protocols was opened in the browser. The page could be loaded with HTTPS protocol, but when switched to HTTP protocol, the connection timed out. As the configuration is stored on the machine, any new user accounts will receive the same firewall rules. Therefore, requirement 1.3 was successfully satisfied.

Next, ClamAV was installed as an anti-malware solution and cron job was set up to execute scanning at 02:00 every day. An example of such cron job can be seen on from the last line on Figure 1. For the user management to become easier, the machine was joined to the Active Directory that was already created in the Windows framework testing part. An account “employee3” was created on the domain controller to be tested on the Ubuntu machine. A secure password was assigned to the user and had to be changed on the next sign on. The domain controller server was also set as an NTP server on the Ubuntu machine in order to successfully join to the domain. The “auditd” package was installed and a folder “Documents” in the home folder of “employee3”, which would be created once domain account is used for login, was configured to be audited with an identifying string.



```
  GNU nano 6.2                              /etc/crontab *

SHELL=/bin/sh
# You can also override PATH, but by default, newer versions inherit it from the environment
#PATH=/usr/local/sbin:/usr/local/bin:/sbin:/bin:/usr/sbin:/usr/bin

# Example of job definition:
# .---------------- minute (0 - 59)
# |  .------------- hour (0 - 23)
# |  |  .---------- day of month (1 - 31)
# |  |  |  .------- month (1 - 12) OR jan,feb,mar,apr ...
# |  |  |  |  .---- day of week (0 - 6) (Sunday=0 or 7) OR sun,mon,tue,wed,thu,fri,sat
# |  |  |  |  |
# *  *  *  *  * user-name command to be executed
17 *    * * *   root    cd / && run-parts --report /etc/cron.hourly
25 6    * * *   root    test -x /usr/sbin/anacron || ( cd / && run-parts --report /etc/cron.daily )
47 6    * * 7   root    test -x /usr/sbin/anacron || ( cd / && run-parts --report /etc/cron.weekly )
52 6    1 * *   root    test -x /usr/sbin/anacron || ( cd / && run-parts --report /etc/cron.monthly )
#
0 2 * * * root clamscan --remove -r /
```

Figure 1. Example of cron job for scheduling an anti-malware scan on Ubuntu Linux.

All the configuration to comply with the remaining chosen requirements was done on the superuser account according to the implementation framework. Then, the superuser account was changed to the “employee3” account. Once the password created on the domain controller was entered, the password has to be changed according to the option selected when creating the user on the Windows server. However, the policy for the password was not enforced from the group policy, but from the requirements set up in the Linux implementation framework. This was noticed due to the fact that the password was successfully validated without any special characters actually required by the group policy. Nevertheless, the password has to be changed with the secure one, so the requirement 8.3 was successfully satisfied.

When the “employee3” account logged in, the terminal was launched. The “ufw” command was entered in order to attempt to modify firewall rules. As expected, the permission was denied accordingly to the solution provided in the implementation framework that blocks the execution permissions on the executable file. Therefore, requirement 1.5 was successfully satisfied.

For the next test, the commands for antivirus modification and scanning were tried to be run. Similarly to the firewall modification attempts, these commands could not be run due to the denied permissions. Therefore, requirement 5.3 was successfully satisfied.

Next, the shared network folder was accessed using the graphical file manager. After providing the same credentials as those that were used to login into “employee3” account, the network share was opened. It contained the “shared_folder” folder that contains regular work files and sensitive data. Though, only the “Work files” folder was visible as configured with permissions on the Windows server. Folders for both users created in the Windows framework testing part were visible and a folder for “employee3” user files was successfully created. Additionally, basic permissions effectiveness was tested by trying to access the superuser account directory. As a result, permission was denied. Final step to test no access flaws were present was trying to execute a command with “sudo” elevation. As the user was not in the “sudoers” file, the command was denied. Consequently, the access control works as expected and requirement 7.3 was successfully satisfied.

The next step to be tested was the audit logs collection requirement. After trying to run the “auditd” command, the output informed that this can only be done by the root user. Then, an empty text file was created in the “Documents” folder in the “employee3” user home directory. Based on the configuration done, this action should have generated an audit log. After switching user to the superuser and reading logs, the entry with the event identifying string created during initial setup was found. Additionally, the “employee3” user was not able to read logs because of denied permissions. Therefore, requirements 10.2 and 10.3 were successfully satisfied.

Even though the Windows server was already set up as an NTP server on the Ubuntu machine, it is necessary to confirm that settings persist on the “employee3” user account as well. The following command was used to see the information about the synchronization of time:

```
timedatectl timesync-status
```

From the output it was seen that the NTP communication works as expected as the Ubuntu machines synchronizes the time with the Windows server. Therefore, requirements 10.6 was successfully implemented.

Finally, to test the effectiveness of the whole solution and requirement 3.4, the same laptop with Windows 10 used for testing the Windows framework was connected from the external network to the network with Ubuntu VM. Then, using the Remote Desktop Connection, the connection was made. The initial connection failed due to the fact that Ubuntu used “sssd” to evaluate GPOs and provide access based on their policies [39]. As

Windows users have to be added to the corresponding group to be able to remotely connect to the machine, the login failed. As the solution, the “xrdp-sesman” user was granted the remote desktop rights in the “sssd” configuration file, and the “sssd” service was restarted [39]. After this, the “employee3” user could be successfully logged in on the remote machine. Finally, the terminal was run and sample text was written in it and copied. Then the copied text was attempted to be pasted to the empty text file on the laptop host. As expected, the text could not be pasted, therefore requirement 3.4 was successfully satisfied.

4.5 Framework for macOS

In framework for macOS devices, macOS 12 Monterey was taken as a basis. It implies that guidelines provided may not be achievable on newer or older versions of macOS or settings and configuration names are different.

Requirement “1.3 Network access to and from the cardholder data environment is restricted” [5].

This requirement dictates that inbound and outbound connections should be restricted to the necessary only [5]. macOS has a default application firewall with GUI (graphical user interface), but it has limitations of managing only incoming connections [40]. To satisfy the requirement, you must use a more powerful firewall, such as a built-in “pf” tool that allows to block or allow packets based on their source and destination addresses, as well as their source and destination ports [41]. In the tool configuration, create rules to allow only necessary inbound and outbound connections and deny all other incoming and outgoing connections. You must pay attention that rules in “pf” are always fully evaluated from top to bottom with the last matching rule dictating the action to be taken [42].

Requirement “1.4 Risks to the CDE from computing devices that are able to connect to both untrusted networks and the CDE are mitigated” [5].

This requirement dictates that system components with cardholder data must not be directly accessible from untrusted networks with NSCs implemented between trusted and untrusted networks [5]. As with both Windows and Linux frameworks, this can be done by network devices, such as firewalls, routers and switches to segment the network with the server provisioning VMs.

Requirement “1.5 Risks to the CDE from computing devices that are able to connect to both untrusted networks and the CDE are mitigated” [5].

This requirement dictates that security controls must always be running and must not be alterable by users [5]. To prevent users from changing firewall rules, you must remove all the permissions from the “pf” tool configuration file. You can do it either with “chmod” command or using “Get Info” tab in GUI. By default, regular users will not be able to disable and enable “pf” tool, so the requirement is satisfied. Also, to make sure that Application Firewall rules will not override “pf” rules, place all “pf” rules below the anchor that integrates application firewall into “pf” in the configuration files [42].

Requirement “2.2 System components are configured and managed securely” [5].

This requirement dictates that only necessary services and functions are enabled, non-console administrative access is secured with encryption and vendor default accounts are removed or their passwords changed if in use [5]. To disable unnecessary services in macOS, run the following command:

```
sudo launchctl disable <service>
```

To comply with other parts of the requirement, follow these guidelines:

1. Delete all the unused accounts before providing employees access to the VMs.
2. Change passwords for all the vendor default accounts that will be used to the secure ones with a minimum length of 12 alphanumeric characters before providing employees access to the VMs.
3. Do not install and run any services on the VMs that will change their primary function, which should be a simple work with documents, including on the Internet, and development.

Requirement “3.4 Access to displays of full PAN and ability to copy PAN is restricted” [5].

This requirement dictates that only four last digits and BIN of PAN may be displayed, except for personnel with business need, and that PAN cannot be copied or moved when using remote-access technologies. macOS computers have the default solution named “Remote Management” built-in that allows other Mac computers to connect remotely to the machine. However, it does not include an option to disable clipboard redirection, so any user that remotely connects to the Mac computer may copy confidential information and paste it on their local machine. So, third-party solution must be used. As an example, TeamViewer could be used as a simple-to-use solution with an ability to disable clipboard synchronization.

Requirement “4.2 PAN is protected with strong cryptography during transmission” [5].

This requirement dictates that PAN is encrypted when sent via end-user messaging solutions [5]. For guidelines on how to comply with the requirement, refer to the framework for Windows requirement 4.2 implementation. macOS allows to use the same email service providers as for Windows. The same type of administrative control can also be used for all systems, which can be either acceptable use policy or security policy or a combination of both.

Requirement “5.2 Malicious software (malware) is prevented, or detected and addressed” [5].

This requirement dictates that the anti-malware solution is deployed and is able to detect and remove known types of malware [5]. macOS comes with the default antivirus xProtect built-in with its own database maintained by Apple [43]. It is capable of providing real-time protection. For scheduled scans, a free solution ClamAV used for Linux framework may be used [26]. Based on the overall security policy and business needs other antivirus solutions may be chosen.

Requirement “5.3 Anti-malware mechanisms and processes are active, maintained, and monitored” [5].

This requirement dictates that anti-malware solution is automatically updated, performs periodic scans or behavioural analysis, performs automatic scans or behavioural analysis when removable media is inserted and cannot be altered [5]. As already mentioned, macOS has a security feature built-in for real-time scanning. As an example for a scheduled scanning solution, ClamAV will be considered. To schedule scans with ClamAV, cron jobs must be used in macOS. Additionally, Ansible or similar application deployment software can be used to manage macOS devices [27]. Ansible can be combined with Homebrew that allows the central application management in addition to files uploading and commands execution [44]. To prevent users from executing antivirus-related command, permissions for execution of the executable commands should be removed, following the same steps described in the Linux implementation framework for the requirement 1.5.

Requirement “7.3 Access to system components and data is managed via an access control system(s)” [5].

This requirement dictates that access control system is in place and grants access based on a user’s need to know and role with default deny permission [5]. This can be achieved by following these steps:

1. Create groups and add users to them based on their job roles.
2. Grant permissions for files only for those group that need access to it. Remove all permissions for other users.

Permission can be granted either in the command line or using the graphical interface and navigating to the folder or file information in Finder.

To access files stored in the share, Finder can be connected to the server with the AD user credentials and access files they are permitted to access. To synchronize passwords and be able to log on to AD user accounts, macOS has a built-in capability to join domain and use its accounts [45]. This is done in the “Users & Groups” system settings. After the domain is joined, domain accounts can be used to log into the machine.

Requirement “8.2 User identification and related accounts for users and administrators are strictly managed throughout an account’s lifecycle” [5].

This requirement dictates that user session cannot be idle for more than 15 minutes [5]. In macOS, this can be set in the “Security & Privacy” advanced settings. As modifying the value of this setting requires an administrator password, regular users will not be able to change it.

Requirement “8.3 Strong authentication for users and administrators is established and managed” [5].

This requirement dictates that users are locked out to a minimum of 30 minutes after not more than 10 logon attempts, are forced to change uniquely assigned password at first logon, have passwords with minimum length of 12 alphanumeric characters if they are used, cannot use the same password as any of their 4 last passwords if passwords are used and have their passwords changed at least every 90 days if passwords is the only authentication method [5].

macOS allows to set global local password policies using the following command:

```
pwpolicy -setpolicy <policy>
```

The policy is customizable and can include the requirement for the minimum password length, minimum amount of alphabetic characters, minimum amount of numeric characters, password history and maximum password age, as well as an option to require the password to be changed on next logon [46]. There are many other options, but those that were outlined should be focused on to comply with the requirement.

After the policy has been set, it should be copied to the new XML (Extensible Markup Language) file and have options for a lockout of 30 minutes after 10 failed logon attempts inserted. Then, the same command for setting a policy may be used with file for policies provided.

If the account is synchronized with the AD, some policies apply to it, such as minimum length, character classes requirements and password history. Though, default domain policy should also include password policies in order for the synchronized account on macOS to correctly receive the policy. Additionally, account lockout policies may not work correctly, therefore, it is still needed to use “pwpolicy” tool for setting lockout rules.

Requirement “8.4 Multi-factor authentication (MFA) is implemented to secure access into the CDE” [5].

This requirement dictates that all access into the CDE should use MFA [5]. Similarly to the solution described in Windows and Linux frameworks, macOS is also capable of using smart cards for login. Smart cards are mapped to a local account when inserted or configured to work with Active Directory via attribute mapping [47]. When the process is done by the system administrator, users will be able to login into their accounts, either stored locally or in the Active Directory, using smart cards.

Requirement “10.2 Audit logs are implemented to support the detection of anomalies and suspicious activity, and the forensic analysis of events” [5].

This requirement dictates that audit logs are enabled for all system components and captures invalid attempts for logical access, management of accounts, actions with elevated access, actions with system objects, reading and changing the flow of audit logs [5]. To comply with the requirement, the built-in “audit” package may be used. By default, it audits only login events, so the configuration of the “audit_control” file should be changed to also include the logging of reading, writing and deleting files events [48]. Additional logging information can be seen in other log files for various system components in the “/var/log” folder.

Requirement “10.3 Audit logs are protected from destruction and unauthorized modifications” [5].

This requirement dictates that audit logs can be read only by employees with business need, cannot be altered and are backed up [5]. As with Linux logs, “rsyslog” can be used to forward logs to the central logs collector. This ensures that logs are stored on various machine and can be backed up and analysed on the log collector machine. To further secure logs, permissions for folders that contains logs should be set to read-only by the owner and the owning group if there is a need with no permission for other users. For example, “/var/audit” and “/var/log” folders should be secured with permissions set up.

Requirement “10.6 Time-synchronization mechanisms support consistent time settings across all systems” [5].

This requirement dictates that system time is synchronized with the use of time-synchronization technology [5]. To synchronize time with a time server, macOS provides a convenient way to do it via the GUI in the “Date & Time” tab in the system settings. There, an NTP server may be specified to synchronize time automatically. Time can only be changed via the administrator account, so users will not be able to alter this setting.

Requirement “10.7 Failures of critical security control systems are detected, reported, and responded to promptly” [5].

This requirement dictates that failures of security controls are detected and addressed promptly [5]. As mentioned in guidelines for requirement 10.2, logs will indicate the state of various system components, including failure events. When logs are stored centrally, there should be a process in place that allows for automatic detection of failures. macOs provides a built-in graphical tool “Control” to see logs with filtering capabilities and the usage of columns and attributes for more human-readable format.

Requirement “11.3 External and internal vulnerabilities are regularly identified, prioritized, and addressed” [5].

This requirement dictates that internal vulnerability scans are performed every three months, including authenticated scanning [5]. To comply with the requirement, you should use the same vulnerability scanning chosen for Windows and Linux systems in case it is compatible with macOS. Generally, any vulnerability scanning tool may be used as long as it effectively indicates vulnerabilities. The choice depends on the needs of an organization. If the tool requires agents to be installed, it may be done via an application deployment tool, such as Ansible combined with Homebrew [44].

Testing the macOS framework: To test the implementation framework, the macOS virtual machine was created.

To test the firewall, an explicit deny rule was put in the beginning in the “pf” configuration file and only ICMPv4 traffic was allowed. Then, a “ping” command was issued to test the connection to the default gateway. The ping was successful and all packets were transmitted and received. Next, the browser was opened and Apple website was tried to be opened. As expected, the site could not be loaded. In order to avoid accidentally denying further traffic used to comply with other requirements, the firewall rules were erased. As the firewall rules and the service are stored on the machine, the same firewall rules will apply for all the users. Therefore, requirement 1.3 was successfully satisfied.

As an anti-malware solution for scheduled scanning, ClamAV was installed and cron job was set up to execute scanning at 01:00 every day. Next, to be able to centrally manage all user account for different system, macOS machine was joined to the Active Directory on the domain controller used for testing both Linux and Windows frameworks. Before actually joining the domain, the macOS time was synchronized with the domain controller to be able to join the domain. An account “employee4” was then created on the domain controller for further testing purposes. A secure password was assigned to the user and an option was chosen for the password to be changed on the next sign on.

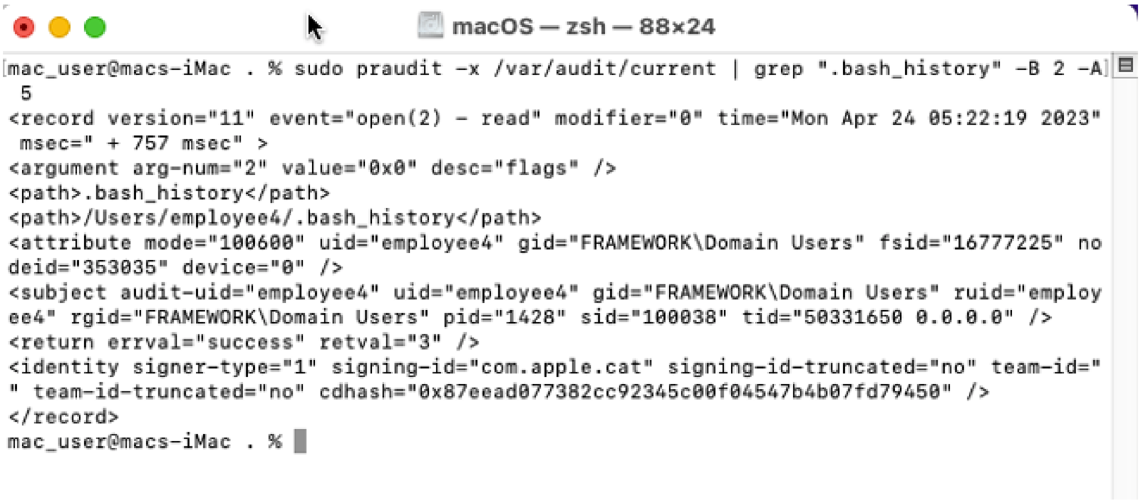
All the configuration to comply with the remaining chosen requirements was done on the local administrator account by following implementation framework. When the configuration was done, the user was changed to the “employee4”. As opposed to Linux, no domain should have been specified with the username to join the domain account. When the password for the user was specified, it was required to be changed. The secure password could only be submitted that correlates to the passwords’ requirements. Therefore, the requirement 8.3 was successfully satisfied.

When the user was logged in, the terminal was launched and the firewall configuration was tried to be changed. The permission to change the file was denied. Next, the firewall was tried to be turned off. The permission was denied for the execution of the command. Therefore, requirement 1.5 was successfully satisfied.

For the next test, the commands for antivirus modification and scanning were tried to be run. The permission was denied to execute commands. Therefore, requirement 5.3 was successfully satisfied.

To test the next requirement, the shared folder was accessed by connecting to the server using the Finder. The IP address of the domain controller was specified and no additional password was required. When the share was opened, the “Work files” folder was visible. Folders for all the users could be seen and the one was successfully created for “employee4”. The “CDE” folder was not visible, as it was configured on the Windows server to be visible only for “employee1” user. To additionally test the correct access for “employee4” account, a command was tried to be run with “sudo” elevation. The command was denied, and the requirement 7.3 was successfully satisfied.

Next, audit logs were tested for compliance. As an example, a file with terminal history was open with “cat” command by the user to generate a log entry. A new terminal window was then open with the local admin user and the “praudit” command was issued to see audit logs. The corresponding log was seen in the output of the command. The full audit log entry can be seen of Figure 2. Therefore, requirement 10.2 was successfully satisfied.



```
mac_user@macs-iMac . % sudo praudit -x /var/audit/current | grep ".bash_history" -B 2 -A
 5
<record version="11" event="open(2) - read" modifier="0" time="Mon Apr 24 05:22:19 2023"
 msec=" + 757 msec" >
<argument arg-num="2" value="0x0" desc="flags" />
<path>.bash_history</path>
<path>/Users/employee4/.bash_history</path>
<attribute mode="100600" uid="employee4" gid="FRAMEWORK\Domain Users" fsid="16777225" no
deid="353035" device="0" />
<subject audit-uid="employee4" uid="employee4" gid="FRAMEWORK\Domain Users" ruid="employ
ee4" rgid="FRAMEWORK\Domain Users" pid="1428" sid="100038" tid="50331650 0.0.0.0" />
<return errval="success" retval="3" />
<identity signer-type="1" signing-id="com.apple.cat" signing-id-truncated="no" team-id="
" team-id-truncated="no" cdhash="0x87eead077382cc92345c00f04547b4b07fd79450" />
</record>
mac_user@macs-iMac . % 
```

Figure 2. Generated audit log for accessing a file by a user.

Then, the folder that contains audit logs and the folder with other system logs were tried to be opened by the “employee4” user. The folders could not be opened as permission was denied. Logs cannot be seen in the “Console” application by user either, as it requires administrator account credentials to show log entries. Therefore, requirement 10.3 was successfully satisfied.

Before the machine joined the domain, Windows server was configured as an NTP server by the local administrator. To make sure that the configuration persists, the “Date & Time” setting is opened. The IP address of Windows server is set to be received time from and cannot be changed by the user without administrator privileges. Therefore, requirement 10.6 was successfully satisfied.

Finally, to test requirement 3.4 and ability to remotely connect to macOS, the TeamViewer software was used on both macOS machine and the laptop with Windows 10 that connects to the internal network from the external network via VPN connection. In the software on the macOS side, clipboard synchronization was disabled and any external connections were prohibited, so only users from the internal network could connect remotely. Finally, the remote connection was established and some text for demonstration was written in it and copied. This text was then tried to be pasted to the blank text file the laptop. No text was copied and nothing could be pasted. Therefore requirement 3.4 was successfully satisfied.

5 Results and analysis

As a result of this work, an implementation framework compatible with PCI DSS requirements was created for Windows, Linux and macOS machines. As a reference for creating the framework, guidelines provided by the PCI SSC were used. The effectiveness of the resulting framework was tested on all the applicable OSs using a defined set of requirements used for testing procedures. The tests involved the installation of security controls and evaluation of these controls in regards to the PCI DSS requirements. The results of all the tests show that the general proposed solution and guidelines for every requirement are successful in meeting PCI DSS requirements compliance.

The framework developed in this thesis clearly demonstrates that PCI DSS compliance is achievable on personal devices with different operating systems, which is an important aspect for both employees and their employer. Employees that are already familiar with the OS provided to them will be more effective and the company will also benefit from the productivity of employees.

The questions that were raised in chapter 1.2 were answered in the work. To answer the question of which PCI DSS requirements are applicable for BYOD devices, several requirements were outlined in chapter 2.1, including its sub-chapters.

Additionally, all the applicable requirements were divided into three categories by the type of control they could be satisfied with, namely: technical, administrative and physical controls. The detailed division can be found in chapters 2.1.1 – 2.1.3.

Lastly, research on the existing solutions was done with no specific works containing detailed guidelines found as a result, except one work written by PCI SSC. More details regarding this specific work and the search criteria can be found in chapter 2.2.

6 Summary

This work could be helpful for private companies that process cardholder data and would like to allow their employees to use their own devices for work purposes. General ideas for implementing administrative and physical controls were given with the main focus put on the technical controls. Detailed guidelines were provided where applicable that would minimize the need for research for system administrators applying security controls. Relevant commentaries were also given on what could also be done to improve the overall security of the company.

As the created implementation framework is based on the PCI DSS standard, it may not suit for companies that follow other cybersecurity standards or multiple of them. Consequently, future research could include the creation of similar BYOD implementation frameworks that are compatible with different cybersecurity standards. Moreover, possibilities for PCI DSS compliance with personal mobile phone devices, which this thesis does not offer, could also be researched.

In addition, this thesis aims to provide guidelines for medium-sized companies that are already compliant with PCI DSS and would like to allow the usage of personal mobile devices or companies that are not yet compliant with PCI DSS, but would like to pass the first audit with the permitted personal mobile devices used for work. Large enterprises usually have complex security policies, therefore, guidelines in the proposed framework may conflict with the existing security policy of a large corporation. Nevertheless, this framework may still be partly used as a foundation for the creation of the company’s own BYOD policy and implementation guide.

References

[1] "What is BYOD? Bring Your Own Device Definition," *WhatIs.com*. https://www.techtarget.com/whatis/definition/BYOD-bring-your-own-device (accessed Apr. 03, 2023).

[2] K. Kinzer, "Top 5 BYOD Business Benefits," *JumpCloud*, May 26, 2022. https://jumpcloud.com/blog/byod-business-benefits (accessed Feb. 09, 2023).

[3] "BYOD–is it Good, Bad or Ugly from the User Viewpoint?," *Microsoft Security Blog*, Jul. 26, 2012. https://www.microsoft.com/security/blog/2012/07/26/byod-is-it-good-bad-or-ugly-from-the-user-viewpoint/ (accessed Mar. 14, 2023).

[4] "Merchant Resources," *PCI Security Standards Council*. https://www.pcisecuritystandards.org/merchants/ (accessed Mar. 14, 2023).

[5] "Payment Card Industry Data Security Standard," Apr. 2022, Accessed: Mar. 14, 2023. [Online]. Available: https://docs-prv.pcisecuritystandards.org/PCI%20DSS/Standard/PCI-DSS-v4_0.pdf

[6] N-able, "Top 7 risks of bring your own device (BYOD)," *N-able*, Jun. 29, 2021. https://www.n-able.com/blog/the-top-7-risks-of-bring-your-own-device-msps-should-remember (accessed Feb. 09, 2023).

[7] "The Security Rule," *HHS.gov*. https://www.hhs.gov/hipaa/for-professionals/security/index.html (accessed Apr. 02, 2023).

[8] "Summary of the HIPAA Security Rule," *HHS.gov*, Nov. 20, 2009. https://www.hhs.gov/hipaa/for-professionals/security/laws-regulations/index.html (accessed Mar. 23, 2023).

[9] "Combined Regulation Text of All Rules," *HHS.gov*. https://www.hhs.gov/hipaa/for-professionals/privacy/laws-regulations/combined-regulation-text/index.html (accessed Apr. 02, 2023).

[10] "Top threats of digital payment: 5 mobile payment security threats every business should be aware of," *Build38*, Jan. 16, 2023. https://build38.com/digital-payment-security-threats/ (accessed Apr. 04, 2023).

[11] "PCI Mobile Payment Acceptance Security Guidelines for Merchants as End-Users," 2017, Accessed: Apr. 03, 2023. [Online]. Available: https://docs-prv.pcisecuritystandards.org/Guidance%20Document/Mobile/PCI_Mobile_Payment_Acceptance_Security_Guidelines_for_Merchants_v2_0.pdf

[12] "End user device security for Bring-Your-Own-Device (BYOD) deployment models - ITSM.70.003," *Canadian Centre for Cyber Security*, May 03, 2022. https://www.cyber.gc.ca/en/guidance/end-user-device-security-bring-your-own-device-byod-deployment-models-itsm70003 (accessed Apr. 10, 2023).

[13] Y. P. Evangelist Product, "Virtual Desktop Infrastructure: Delivering Employee Workstations." https://bluexp.netapp.com/blog/cvo-blg-virtual-desktop-infrastructure-vdi-delivering-employee-workstations-on-demand (accessed Apr. 10, 2023).

[14] "PCI DSS Virtualization Guidelines," 2011, Accessed: Apr. 12, 2023. [Online]. Available: https://docs-prv.pcisecuritystandards.org/Guidance%20Document/Virtualization%20and%20Cloud/Virtualization_InfoSupp_v2.pdf

[15] "Share SharePoint files or folders - Microsoft Support." https://support.microsoft.com/en-us/office/share-sharepoint-files-or-folders-1fe37332-0f9a-4719-970e-d2578da4941c (accessed Apr. 24, 2023).

[16] Atlassian, "Confluence: The Ultimate Wiki Software," *Atlassian*. https://www.atlassian.com/software/confluence/use-cases/wiki (accessed Apr. 24, 2023).

[17] "Securing Remote Desktop (RDP) for System Administrators | Information Security Office." https://security.berkeley.edu/education-awareness/securing-remote-desktop-rdp-system-administrators (accessed Apr. 15, 2023).

[18] "How to enable and disable Tamper Protection in Windows 10 | TechTarget," *Enterprise Desktop*. https://www.techtarget.com/searchenterprisedesktop/tip/What-IT-should-know-about-Tamper-Protection-in-Windows-10 (accessed Apr. 21, 2023).

[19] "Smart Card Two-Factor Authentication," 2022, Accessed: Apr. 12, 2023. [Online]. Available: https://www.emerson.com/documents/automation/white-paper-smart-card-2-factor-authentication-deltav-en-56306.pdf

[20] "How to configure Windows advanced audit policy | ADAudit Plus." https://www.manageengine.com/products/active-directory-audit/kb/configure-windows-advanced-audit-policy.html (accessed Apr. 21, 2023).

[21] "How to Set Up a VPN on Windows Server 2019 using Remote Access," *Vultr*. https://www.vultr.com/docs/how-to-set-up-a-vpn-on-windows-server-2019-using-remote-access/ (accessed Apr. 21, 2023).

[22] R. Authors, "PPTP vs. L2TP: What's the Difference? - Rublon." https://rublon.com/blog/pptp-l2tp-difference/ (accessed Apr. 21, 2023).

[23] "xRDP – How to disable Drive Redirection and Clipboard – Griffon's IT Library." https://c-nergy.be/blog/?p=17410 (accessed Apr. 16, 2023).

[24] "Kaspersky Downloads for Small to Medium Business | Kaspersky." https://usa.kaspersky.com/small-to-medium-business-security/downloads (accessed Apr. 16, 2023).

[25] "Bitdefender GravityZone Business Security," *Bitdefender*. https://www.bitdefender.com/business/smb-products/business-security.html (accessed Apr. 16, 2023).

[26] "ClamAVNet." https://www.clamav.net/ (accessed Apr. 16, 2023).

[27] "Ansible is Simple IT Automation." https://www.ansible.com (accessed Apr. 17, 2023).

[28] G. Newell, "An introduction to Linux Access Control Lists (ACLs)," *Enable Sysadmin*, Feb. 06, 2020. https://www.redhat.com/sysadmin/linux-access-control-lists (accessed Apr. 17, 2023).

[29] M. Ne, "How to Prevent Users from Changing Specific Settings in Ubuntu / Mint," *FOSTips*, Feb. 02, 2022. https://fostips.com/prevent-changing-specific-settings-ubuntu/ (accessed Apr. 18, 2023).

[30] A. Kili, "How to Force User to Change Password at Next Login in Linux," Feb. 23, 2018. https://www.tecmint.com/force-user-to-change-password-next-login-in-linux/ (accessed Apr. 22, 2023).

[31] S. Henry-Stocker, "How to enforce password complexity on Linux," *Network World*, Oct. 19, 2020. https://www.networkworld.com/article/2726217/how-to-enforce-password-complexity-on-linux.html (accessed Apr. 18, 2023).

[32] E. Amoany, "How to set user password expirations on Linux," *Enable Sysadmin*, Jul. 05, 2022. https://www.redhat.com/sysadmin/password-expiration-date-linux (accessed Apr. 18, 2023).

[33] P. Kumar, "Lock User Account After n Failed Login attempts in Linux," Dec. 18, 2019. https://www.linuxtechi.com/lock-user-account-incorrect-login-attempts-linux/ (accessed Apr. 18, 2023).

[34] "pam_faillock: lock user account after X failed login attempts in Linux | GoLinuxCloud," Apr. 13, 2021. https://www.golinuxcloud.com/pam-faillock-lock-user-account-linux/ (accessed Apr. 23, 2023).

[35] "Smart card authentication," *Ubuntu*. https://ubuntu.com/server/docs/security-smart-cards (accessed Apr. 18, 2023).

[36] "What Is Auditd in Linux: A Brief Tutorial," *Sematext*. https://sematext.com/glossary/auditd/ (accessed Apr. 18, 2023).

[37] "Audit logs not forwarded to remote syslog server on RHEL 8," *Red Hat Customer Portal*, Apr. 24, 2020. https://access.redhat.com/solutions/4986931 (accessed Apr. 18, 2023).

[38] "Time synchronisation," *Ubuntu*. https://ubuntu.com/server/docs/network-ntp (accessed Apr. 18, 2023).

[39] "xRDP – Remote Connection to Ubuntu Using Active Directory Authentication (HowTo) – Griffon's IT Library." https://c-nergy.be/blog/?p=16274 (accessed Apr. 23, 2023).

[40] "How to Block All Incoming Network Connections in Mac OS X," *OS X Daily*, Aug. 28, 2013. https://osxdaily.com/2013/08/28/block-all-incoming-network-connections-mac-os-x/ (accessed Apr. 19, 2023).

[41] "OpenBSD PF: Packet Filtering." https://www.openbsd.org/faq/pf/filter.html (accessed Apr. 19, 2023).

[42] T. S. Manjusri, "PF on Mac OS X." https://manjusri.ucsc.edu/2015/03/10/PF-on-Mac-OS-X/ (accessed Apr. 19, 2023).

[43] "Do I Need Antivirus for My Mac or Is It Built In?," *Security.org*. https://www.security.org/antivirus/mac/ (accessed Apr. 23, 2023).

[44] S. Dudhgaonkar, "Using Ansible to automate software installation on my Mac | Opensource.com." https://opensource.com/article/22/6/install-software-macos-ansible-homebrew (accessed Apr. 23, 2023).

[45] K. Kinzer, "How to Join a Mac to Active Directory via Terminal," *JumpCloud*, Mar. 04, 2022. https://jumpcloud.com/blog/how-to-join-macos-to-active-directory (accessed Apr. 23, 2023).

[46] "man page pwpolicy section 8." https://www.manpagez.com/man/8/pwpolicy/osx-10.10.php (accessed Apr. 23, 2023).

[47] "Use a smart card on Mac," *Apple Support*. https://support.apple.com/et-ee/guide/deployment/depc705651a9/web (accessed Apr. 23, 2023).

[48] “audit_control(5).” https://man.freebsd.org/cgi/man.cgi?query=audit_control&sektion=5&n=1 (accessed Apr. 24, 2023).

Appendix 1 – Non-exclusive licence for reproduction and publication of a graduation thesis[1]

I Kirill Kurkin

1. Grant Tallinn University of Technology free licence (non-exclusive licence) for my thesis “PCI DSS Compliant BYOD Implementation Framework”, supervised by Mohammad Tariq Meeran
   1.1. to be reproduced for the purposes of preservation and electronic publication of the graduation thesis, incl. to be entered in the digital collection of the library of Tallinn University of Technology until expiry of the term of copyright;
   1.2. to be published via the web of Tallinn University of Technology, incl. to be entered in the digital collection of the library of Tallinn University of Technology until expiry of the term of copyright.
2. I am aware that the author also retains the rights specified in clause 1 of the non-exclusive licence.
3. I confirm that granting the non-exclusive licence does not infringe other persons' intellectual property rights, the rights arising from the Personal Data Protection Act or rights arising from other legislation.

14.05.2023

[1] The non-exclusive licence is not valid during the validity of access restriction indicated in the student's application for restriction on access to the graduation thesis that has been signed by the school's dean, except in case of the university's right to reproduce the thesis for preservation purposes only. If a graduation thesis is based on the joint creative activity of two or more persons and the co-author(s) has/have not granted, by the set deadline, the student defending his/her graduation thesis consent to reproduce and publish the graduation thesis in compliance with clauses 1.1 and 1.2 of the non-exclusive licence, the non-exclusive license shall not be valid for the period.

Appendix 2 – Guidelines for covering requirement 1.3 on Windows

Inbound and outbound connections on Windows can be restricted with the following steps:

1. In the Group Policy Management tool, create or edit a GPO (Group Policy Object).
2. Navigate to “Windows Defender Firewall with Advanced Security” setting.
3. In the “Windows Defender Firewall Properties” setting in the “Domain Profile”, set “Firewall state” setting to “On”, set “Inbound connections” and “Outbound connections” to “Block”. This ensures that connections that have no allow rule defined will be blocked. Apply the same settings for “Private Profile” and “Public Profile” tabs and save changes.
4. Create a new outbound rule or several of them to allow necessary outbound traffic only according to your organization needs.
5. Create a new inbound rule or several of them to allow necessary inbound traffic only according to your organization needs.
6. Apply the GPO to the Organizational Unit containing VMs.

Appendix 3 – Guidelines for covering requirement 5.3 on Windows

Applicable sub-requirement in requirement 5.3 can be covered with the following steps:

1. In the Group Policy Management tool, create or edit a Group Policy Object.
2. Navigate to “Windows Defender Antivirus” administrative template.
3. Disable “Turn off Windows Defender Antivirus” policy and enable “Allow antimalware service to remain running always” policy.
4. In the “Real-time protection” section, disable “Turn off real-time protection” policy and enable “Turn on behavior monitoring” policy.
5. In the “Scan” section, enable “Scan removable drives” policy and enable “Specify the time of day to run a scheduled scan” policy with configuration that suits your needs.
6. In the “Signature Updates” section, enable “Specify the time to check for definition updates” policy with configuration that suits your needs.