Project Final Report

Penetration Testing Scenario

Group 53

PM Marcin Masternak

PM Mateusz Chyla

Mohammed Almouhanna

David Innes

Tomasz Fikier

Jakub Hurkala

Project Sponsor

Dr Jawad Ahmad

Project Client

Mr. Robert Ludwiniak

Edinburgh Napier University

Table of content

Executive summary **3**
Delivered product **4**
Client approval **7**
Closing audit **8**
Lessons learned **12**
Team contribution **15**
Appendices **16**
Appendix 1 – Key deliverables overview **16**
Appendix 2 – Client approval evidence **20**
Appendix 3 – Closing Audit evidence **21**
I. Project Health Register screenshots 21
II. Evidence of Customer editing the satisfaction survey: 23
Appendix 4 – Lessons learned evidence **24**
I. Communication and data exchange evidence. 24
II. Issue mitigation and contingency plan evidence. 26
III. Cooperation with supplier - evidence. 31
Appendix 5 – Example of detailed research document **32**
References **38**

Note!

This document contains appendices evidencing statements made throughout the report. For reader convenience hyperlinks have been placed in key sections. Links can be used to jump to relevant evidence. Return links are located at the end of appendices.

Executive summary

There is no doubt that security is one most critical aspect that needs to be considered when implementing any computer/information system. Negligence in this area can lead to financial loses, legal challenges and loss of reputation. Companies, institutions and individuals have been tackling this issue by various means. One of them is concept of penetration testing or “ethical hacking”. It is non-malicious attempt, to find and explore vulnerabilities present in the system. Once security holes are exposed, preventative measures can be introduced.

Client of this project is responsible for running cyber security and penetration testing class at Napier University. Running such class requires appropriate test environment. The one currently used, is based on Windows Server 2003 and customer’s wish is for it to be updated to more recent operating system. This system should be host to number of services:

- e-commerce web site (dynamic, with multiple access levels and utilizing database)
- server roles (email, shared storage, DHCP*, DNS*, ...)

Other key requirement is to provide number of vulnerabilities that can be introduced into the system.

Shortlisted areas of interest are authentication, encryption, password attacks, reverse-shell attacks, privilege escalation, cross-site scripting, SQL* injections, DHCP, ARP* and DNS attacks.

Implemented vulnerabilities are going to be explored by students in the laboratories. They are supposed to be appropriately challenging and provide educational value.

It should be mentioned that even though the customer provided the list of possible exploits, they are open to other options, if they meet the above condition.

Project was delivered in cooperation by two teams of six students. Team 53, which is subject to this report, is made up of students attending Networking, Cybersecurity and Web Development courses.

Project went through planning and research stage. Team interpreted customer’s needs and presented the proposal. Series of meetings with customer took place and resulting specification was recorded in form of MoSCoW priority sheet. At the end of the process Customer was presented with the outcome. They were satisfied with the scope and quality of delivered product and expressed their approval.

Project was developed in line with P3.express framework. Vital metrics were monitored on weekly basis and summarized in final audit. It showed overall high quality of Project Management and Delivery process.

[1] DHCP - Dynamic Host Configuration Protocol
[2] DNS - Domain Name System
[3] SQL - Sequence Query Language
[4] ARP - Address Resolution Protocol

Delivered product

Entry project deliverables were specified in the project brief. Later, during introductory and planning stage of project, they were reviewed by the team and the customer. That resulted in requirements specification and prioritization sheet based on MoSCoW model.

Functional breakdown of delivered project was prepared in form of a mind map.

In this chapter both of those methods will be used to compare planned deliverables with the ones achieved. Brief description will be provided for accomplished tasks.

Color coding will be used to specify completion stage of each item:

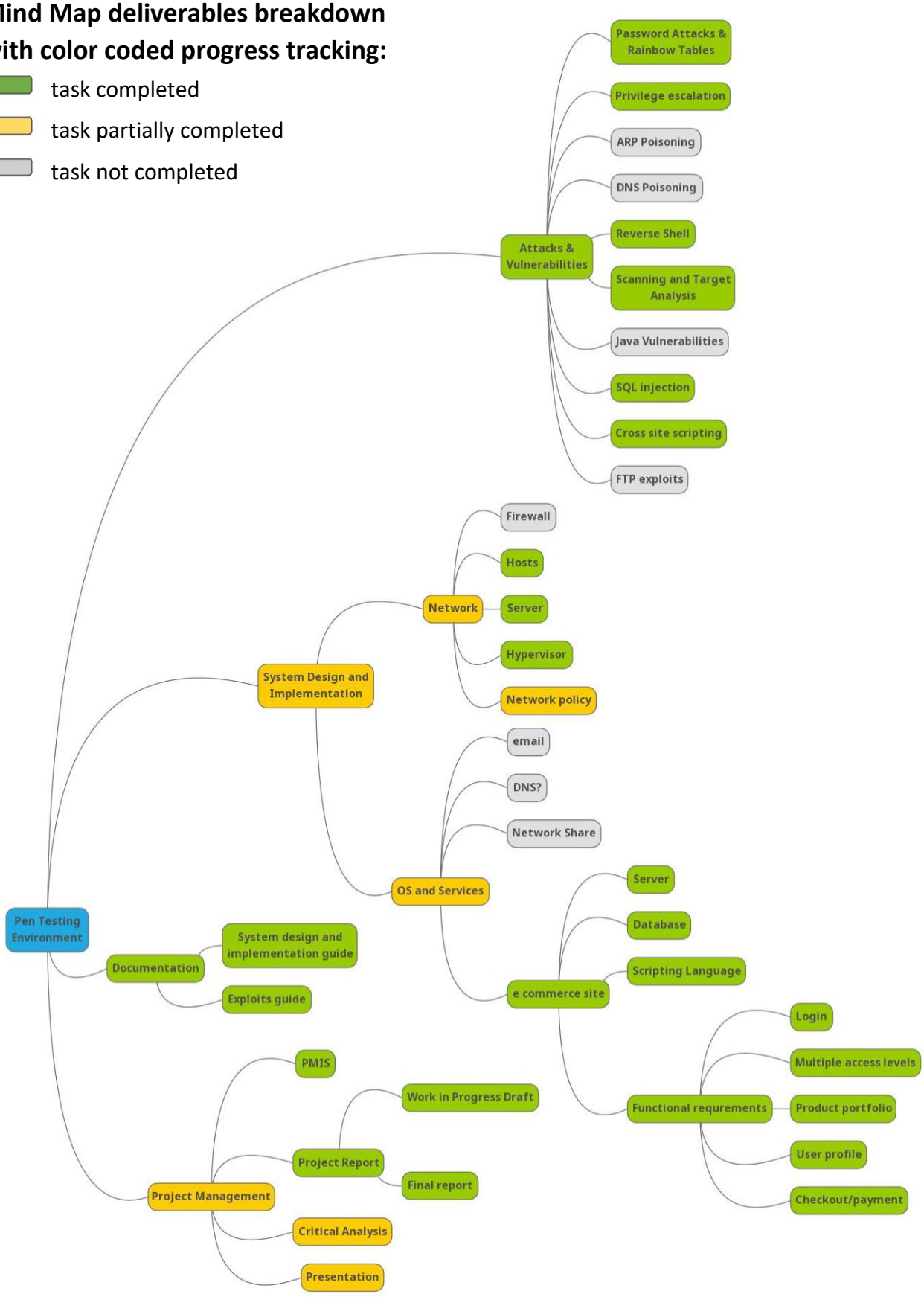
task completed	
task partially completed	task not completed

MoSCoW priority table:

Must have	Should have	Could have	Won't have
Configured Server VM	Password Attacks Privilege escalation ARP Poisoning DNS Poisoning Reverse Shell Scanning and Target Analysis Java Vulnerabilities SQL injection Cross site scripting FTP exploits	Backup Server Firewall configuration Hosts configuration Network policy definition Email server DNS server	Encryption
Appropriate number of specified Attacks & Vulnerabilities implemented.			
E-commerce web site	Login Multiple access levels Product portfolio User profile Payment/checkout		
	Network Share Documentation		

Mind Map deliverables breakdown with color coded progress tracking:

- task completed
- task partially completed
- task not completed

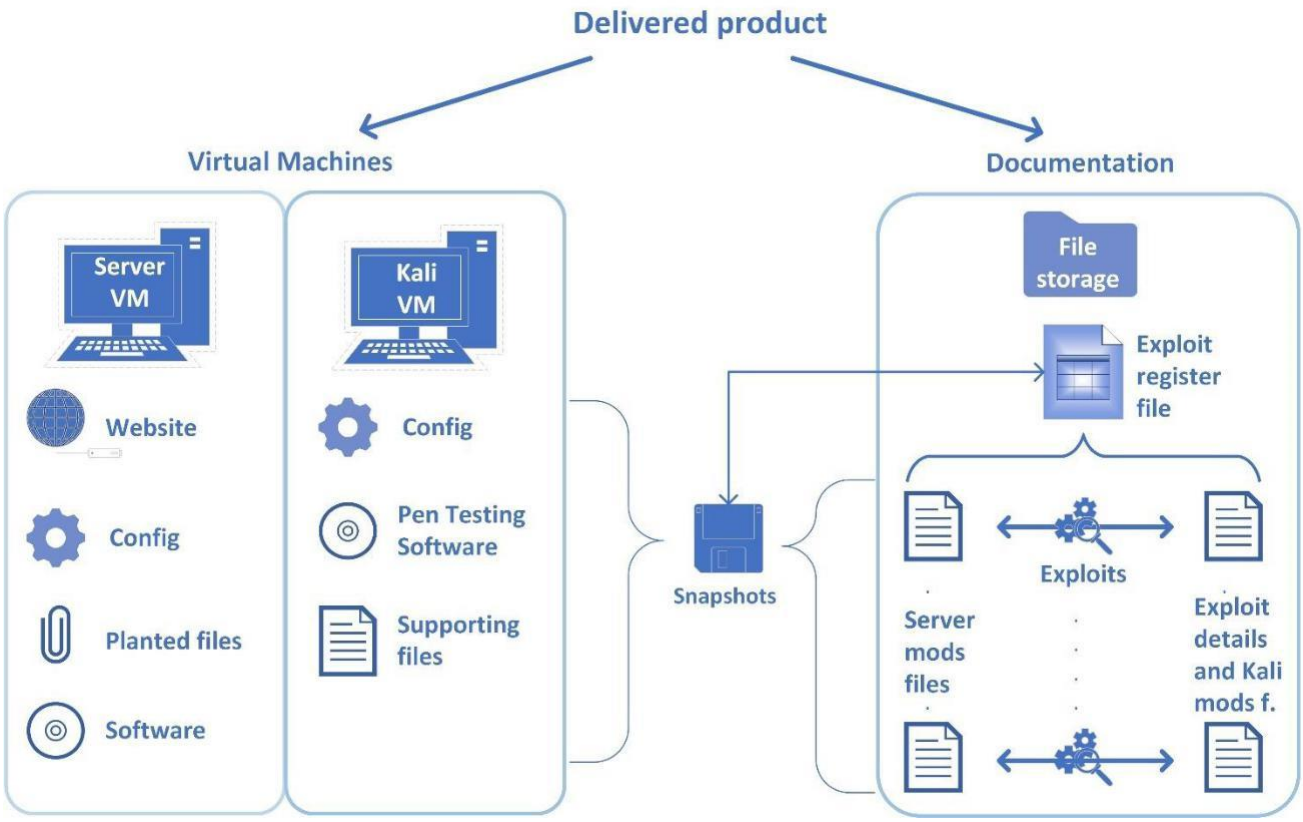


Final product subsystems and delivery method:

Product will be handed in as two parts. Configured Virtual Machines and relating documentation. Server machine will be configured to provide required services. It may also contain additional files and modifications that are necessary for some exploits to work.

Kali Machine will contain all necessary tools to perform the attacks. It will also contain supporting files like rainbow tables.

Each exploit will be listed in the Register and explained in detail in separate document. Register file will link information about the exploit, method of execution, necessary modifications to the machines and relevant snapshot of the server VM.



Brief description of key deliverables can be found in: Appendix 1 – Key deliverables overview

Client approval

Customer was presented with the deliverables using following methods:

- Documentation files describing researched exploits (including step by step walk through).
- Documentation files regarding the website.
- Video presentations for website and researched exploits.

Customer was also presented with Final Report, containing deliverables listing, and prioritization in form of MoSCoW table and Mind Map. Those details are provided in line with the content of PMIS* and can be found on pages 4 and 5 of this document.

Customer's Statement:

I confirm that the delivered product meet all primary requirements described in project specification. I further confirm that key deliverables are included in final product.

I sign this as confirmation and proof of approval of the product and I agree that the deliverables have been handed over.

Signed: Robert Ludwiniak
Date: 09/04/2020

Evidence can be found in: Appendix 2 – Client approval evidence

[1] PMIS – Project Management Information System

Closing audit

Developing a project can be a complex task. It involves multiple elements that need to pe properly executed and coordinated. Technical details, documentation, time management, budgeting, risk analysis and customer communications are example aspects of project delivery process.

Good product is not the only measure of successful project. The process itself must meet certain standards as well. Structured approach is essential to enable team members and stakeholders to participate and contribute to their best abilities.

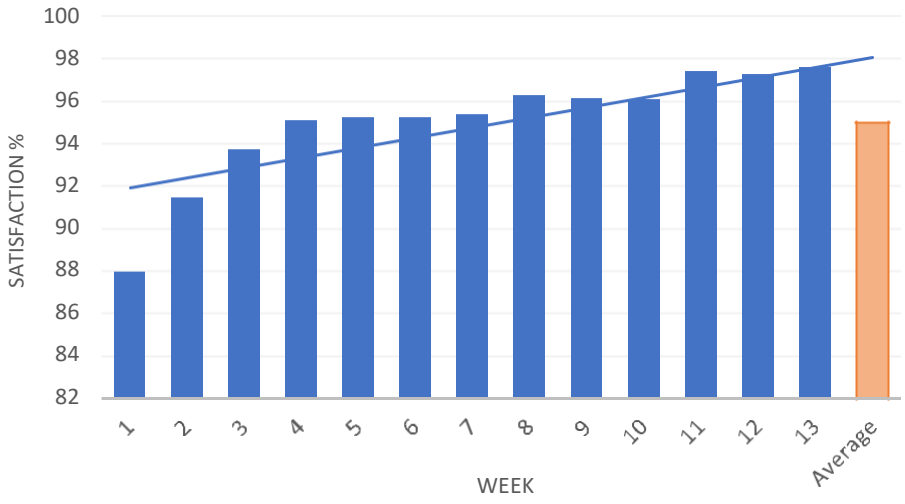
In delivering this project we used P3.express project management framework. It provides a number of tools and metrics that are useful in monitoring and evaluating quality of delivery process. Project metrics were monitored on weekly basis using spreadsheets included in PMIS*. Results were summarized at the end of the project and can be found below:

[[1] PMIS – Project Management Information System]

I. To measure team satisfaction with the Project Management methods, we used Project Health Register.

Team was questioned for their opinion on the Project Management process on the weekly basis. Health Register file was shared with all team members using Microsoft Cloud. Each team member was asked to fill team satisfaction survey before the end of the weekly cycle so that any issues could be discussed on the Monday meetings.

Graph presents average team satisfaction over the period of 13 weeks.



It can be observed that team mostly approved of the PM approach. There was some uncertainty to the beginning, with team members gradually showing more confidence over the course of the project.

II. Customer was pooled for their opinion on the PM process with questionnaire form sent via email.

Customer satisfaction survey.				
Please insert values in range 0-100				
		Jan	Feb	Mar
Is it easy to communicate with us and let yourself understood?		90%	90%	90%
Are we responsive enough in our communications?		90%	90%	90%
Is our approach to the project clear and transparent enough for you?		95%	95%	95%
Have we been providing you with sufficient information about our current progress?		100%	100%	100%
Have we been providing you with sufficient information about our planned progress for the future?		80%	80%	80%
Overall how would you rate your experience as a Customer?		90%	90%	90%
Is there someting that we could change to make your experience as a Customer of this project better.		all good		

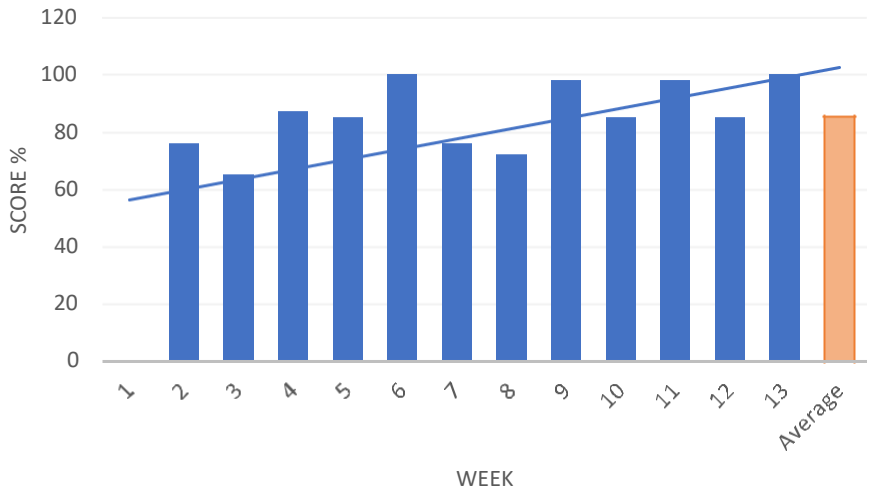
It can be observed that our approach to Project Management and Delivery was highly rated by the customer.

III. Project health was monitored on weekly basis using "Weekly Audit" sheet.

This was done to verify that weekly Project Management routines were accomplished:

- check if progress data is reliable
- check if Sponsor, Customer and Team members have checked the progress board (Trello)
- check if action plan was defined for any deviations from project plan
- was there an effective weekly meeting with stakeholders?

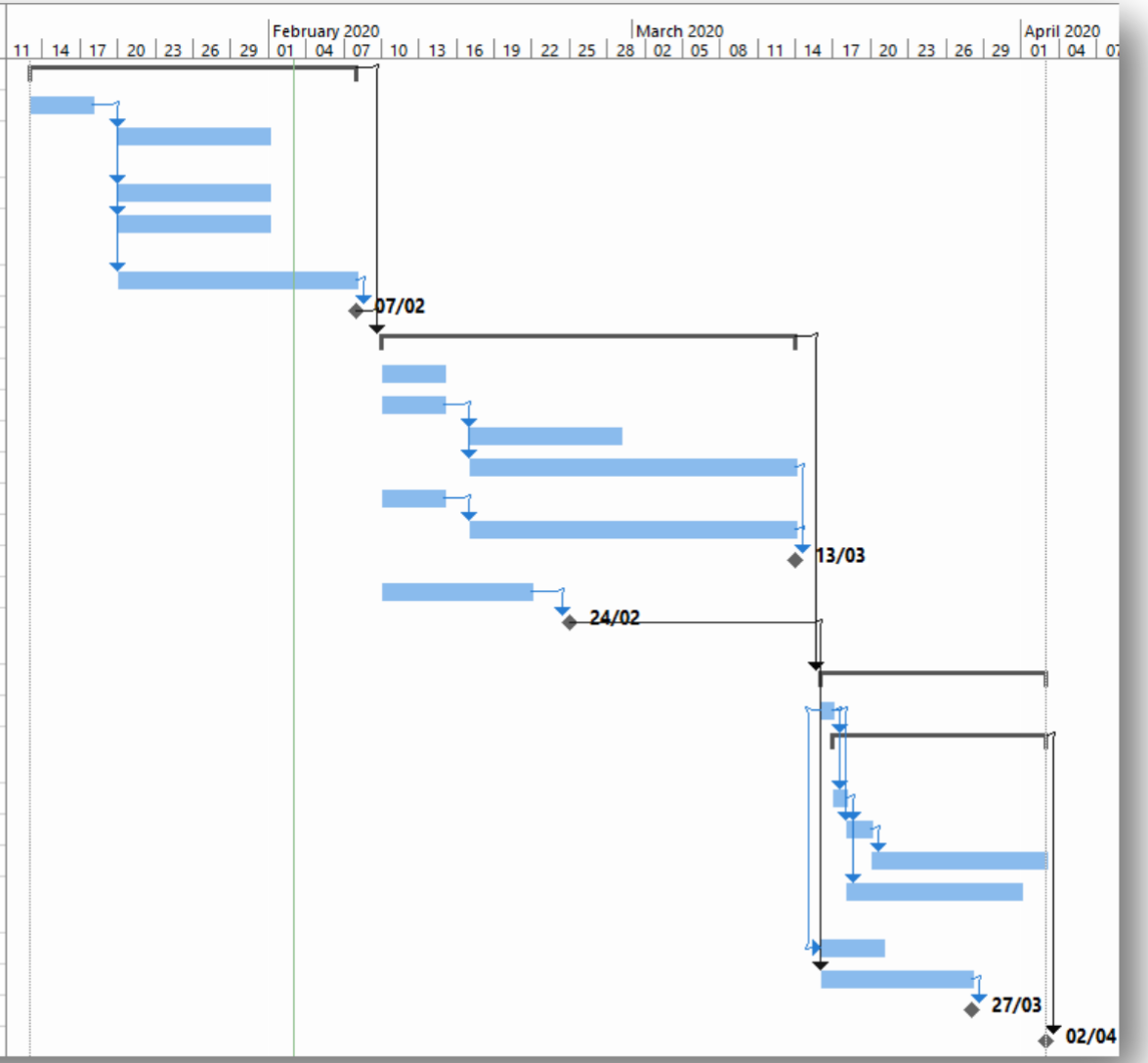
Graph below visualizes average level of completion of PM weekly routines:



IV. Tasks delivery rate was monitored using Progress Register Sheet.

Completion level of planned weekly tasks was checked every Monday. After tasks were checked against the Project Plan - Progress Register was updated.

ID	Task Name	Duration	Start	Finish
1	**Planning**	**20 days**	**13 Jan**	**07 Feb**
2	Team Creation	5 days	13 Jan	17 Jan
3	Client and Sponsor communications	10 days	20 Jan	31 Jan
4	Role establishment	10 days	20 Jan	31 Jan
5	Initiating cooperation with supplier	10 days	20 Jan	31 Jan
6	Creating PMIS	15 days	20 Jan	07 Feb
7	PMIS Submitted	0 days	07 Feb	07 Feb
8	**Research & Prototyping**	**25 days**	**10 Feb**	**13 Mar**
9	Network design	5 days	10 Feb	14 Feb
10	Operating System	5 days	10 Feb	14 Feb
11	Services hosted	10 days	17 Feb	28 Feb
12	Attacks and Vulnurabilities	20 days	17 Feb	13 Mar
13	Web Stack choice	5 days	10 Feb	14 Feb
14	Web site design	20 days	17 Feb	13 Mar
15	Research completed	0 days	13 Mar	13 Mar
16	Progress report preparation	10 days	10 Feb	21 Feb
17	Midpoint progress report submitted	0 days	24 Feb	24 Feb
18	**Implementation**	**14 days?**	**16 Mar**	**02 Apr**
19	Functional network	1 day	16 Mar	16 Mar
20	**Merging components in production environment**	**13 days?**	**17 Mar**	**02 Apr**
21	OS Services implementation	1 day?	17 Mar	17 Mar
22	Web stack installation	2 days	18 Mar	19 Mar
23	Web site implementation	10 days	20 Mar	02 Apr
24	Exploits implementation and testing	10 days	18 Mar	31 Mar
25	Finalizing documentation	5 days	16 Mar	20 Mar
26	Final report preparation	10 days	16 Mar	27 Mar
27	**Final report submitted**	0 days	27 Mar	27 Mar
28	**Project delivery**	0 days	02 Apr	02 Apr



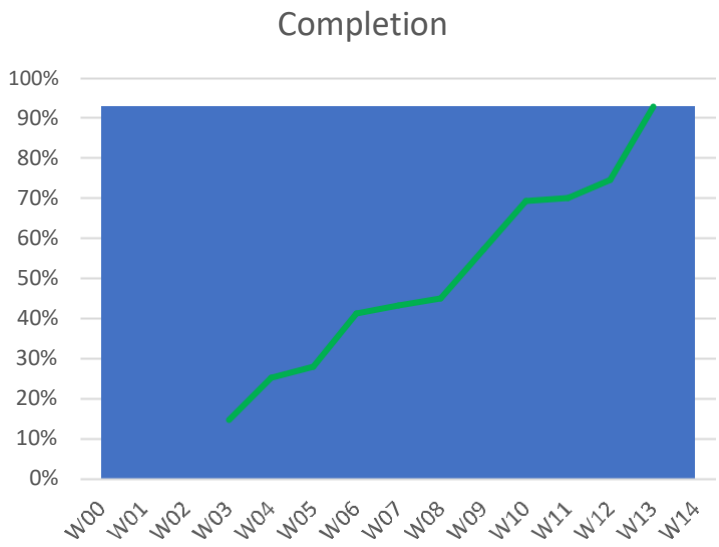
Modified version of P3.express Progress Register template was used. Changes to the sheet were necessary as this project did not have a budget and the main resource was time. Meaning of following values have been adjusted:

- **Planned cost** – amount of time assigned to the project to date

  * in scale 1 to 10 per week, where 10 represents 48hrs (8hrs x 6 members)

- **Actual cost** – amount of time spent weekly on the project to date

  * in scale 1 to 10per week, where 10 represents 48hrs (8hrs x 6 members)



- **Earned value** – proportion of tasks planned for the week being completed to date [in scale 1-10 per week]

Data	W00	W01	W02	W03	W04	W05	W06	W07	W08	W09	W10	W11	W12	W13
Actual Duration	0	1	2	3	4	5	6	7	8	9	10	11	12	13
Planned Total Duration	14	14	14	14	14	14	14	14	14	14	14	14	14	14
Actual Cost	0	10	20	30	40	50	60	70	80	90	100	110	120	130
Earned Value	0	8	13	25	38	45	58	65	72	86	97	105	112	130
Information	**W00**	**W01**	**W02**	**W03**	**W04**	**W05**	**W06**	**W07**	**W08**	**W09**	**W10**	**W11**	**W12**	**W13**
Earned Schedule	0	0.8	1.3	2.5	3.8	4.5	5.8	6.5	7.2	8.6	9.7	10.5	11.2	13.0
Completion	#N/A	#N/A	#N/A	15%	25%	28%	41%	43%	45%	57%	69%	70%	75%	93%
Forecast At-Completion Cost	#N/A	#N/A	#N/A	140	140	140	140	140	140	140	140	140	140	140
Forecast Total Duration	#N/A	#N/A	#N/A	17	15	16	14	15	16	15	14	15	15	14

Completion chart shows steady progress rate with three periods of slower development. Those can be linked to:

- period of uncertainty after initial organizational and research stage
- issues with selecting appropriate website technology
- slow down due to the changed work model forced by the COVID-19 lockdown situation

Evidence for this section can be found in: Appendix 3 – Closing Audit evidence

Lessons learned

I. Introduction to “Lessons Learned”

During the lifetime of the project number of significant events and processes will occur. Each of them will have some impact on one or more of the following:

- deliveries
- timing/scheduling
- quality of the delivery process

Lessons learned section is concerned with monitoring those events and evaluating the way they affect the project. This approach mimics natural learning process that can be observed in all sentient creatures. In nature, knowledge is best retained when it is backed by experience. With lessons learned, teams or organizations can take a note of situations that are desirable and should be repeated and those that are harmful and should be avoided. This knowledge can be then applied to the next venture or in later stages of ongoing project.

Monitoring is best to be done in regular intervals through the project with summary created on competition. If using agile approach following steps can be taken at the end of each sprint.

- identify the event
- take a note of its impact on the project
- identify the event and its impact as positive, neutral or negative
- when it comes to negative and neutral events describe improvements and preventative measures (that were taken or could be taken)

In this project key events were noted in Risk and Issue Register.

During the lifetime of the project team had ongoing discussion about deliverables, newly arising circumstances and delivery process. This was documented with meeting minutes, emails, team satisfaction survey and team chat history.

II. Following factors were identified as having significant impact on the project. As such they should be noted for future reference:

1. Importance of well-designed communication, data exchange and storage system.

   From the very beginning of the project, our Team established and maintained effective communication channels. We took advantage of Office365 subscription available to the students of Napier University. Over the course of the project we actively used following applications:

   - Microsoft Teams –group chat and ad hoc file sharing
   - Microsoft Outlook –email.
   - Trello board integrated with Microsoft Teams –live progress-management.
   - Microsoft One Drive –PMIS and project files sharing.

   Early establishment of robust communication and data exchange channels was a key to effective cooperation. There were two teams working on the project and we successfully convinced our colleagues to accept our communication system. That proved particularly useful when social distancing regulations were introduced due to COVID19 pandemic. Having that system in place allowed nearly seamless transition to remote model of operation.

2. Importance of issue mitigation strategy and contingency plan.

   During the lifetime of the project we encountered number of issues:

   - Delayed kick off – due to differences between initial brief listed on ”Projex” site and actual customer’s expectations.
   - Technical issues – despite initial agreement, customer was not able to provide hosting for virtual machines.
   - Development issues – it took three attempts at various technologies before teams came up with the best way to develop the website. That caused 3 weeks delay to this part of the project.

   Team managed to avoid major delay through the combination of following methods:

   - issue logging using RIC register
   - regular meetings where any arising problems were discussed and decisions on how to mitigate them were made
   - contingency period of two weeks, towards the end of the timescale was included in the project plan
   - MoSCoW prioritization system allowing the team to focus on essential tasks and put those less crucial on hold

   The conclusion is that it is highly important for the team to be ready for unexpected situations. Well defined priorities, risk monitoring and regular team meetings are essential parts of issue management. Sometimes, despite the best preparation, delay is unavoidable. Therefore, appropriate contingency period must be included in the project plan.

3. Importance of close cooperation with suppliers and other stakeholders

Due to the nature and complexity of the project two teams were allowed to work on it. We were supposed to deliver two similar yet slightly different versions of the product. One was designed as laboratory practice while the other is meant to be the coursework.

During the course of the project we maintained very close working relationship with our partner team (group 64). We set up communication channels (Microsoft Teams chat) dedicated to inter-team communication. Common data sharing area was created as well (using Microsoft cloud). Both teams actively contributed to the online conversation, added and edited shared resources. All client meetings were attended by representants of both groups. We also held weekly Monday briefings in Jack Kilby Centre.

Working closely together provided number of benefits:

- Team 53 could benefit from advice and expertise of some more experienced colleagues from team 64. One of them was member of ENUSEC (Edinburgh Napier University Security Association) and his input was highly valued. In return, Team 53 contributed in line with our strength, by assisting with creating detailed documentation and video presentations.
- Some parts of delivered product were supposed to be very similar if not the same for both teams, example of that is the website. We were able to create inter-team, task driven group, that was working on web development. That saved a significant amount of time.
- We could brainstorm and double check our ideas on wider forum
- By working together members of both teams created new professional connections that may prove beneficial in future projects.

Overall our good working relationship with the other team contributed greatly towards quality of research and speed of delivery.

4. Lack of result driven team members evaluation can affect the progress and scope.

All team members showed enthusiasm and engagement with the project. Customer, client and intra team meetings were regularly attended by everyone. There was active discussion held on communication channels, with all members expressing their opinions about the current issues and contributing ideas. We evaluated member contribution based on this activity and as a result, scores were roughly equal. However, we did not precisely measure the actual output and its quality for each participant. While output from all team members was satisfactory, towards the end of the project it became apparent that some individuals contributed more. Stricter, individual progress monitoring could induce more consistent results. That could allow for earlier delivery with possibly more of optional subsystems included in final product.

Evidence for this section can be found in: Appendix 4 – Lessons learned evidence

Team contribution

Performance of team members was monitored on weekly basis using Team Contribution sheet. Each week, individuals were awarded contribution value based on their share of Team's total output for the week. Other factors influencing the score were attendance to the meetings and level of activity in communication channels.

Project Sponsor (Dr Jawad Ahmad) was granted access to the PMIS*, including the Team Contribution sheet.

Due to the nature of the team, objective evaluation was difficult to introduce. With each team member having the same level of seniority, decision on marking had to be taken as consensus. Values for each week were decided on weekly meetings.

The content of the Contribution Sheet was shared with the Team and Project Sponsor via the Microsoft Cloud.

Table below shows consistent levels of contribution from all team members over the course of the project.

Week	Marcin Masternak	Mohammed Almouhanna	Mateusz Chyla	Tomasz Fikier	Jakub Hurkala	David Innes	Total
2	16.0%	16.0%	17.0%	17.0%	17.0%	17.0%	100.0%
3	17.0%	16.0%	16.0%	17.0%	17.0%	17.0%	100.0%
4	17.0%	17.0%	16.0%	16.0%	17.0%	17.0%	100.0%
5	17.0%	17.0%	17.0%	16.0%	16.0%	17.0%	100.0%
6	17.0%	17.0%	17.0%	17.0%	16.0%	16.0%	100.0%
7	16.0%	17.0%	17.0%	17.0%	17.0%	16.0%	100.0%
8	16.0%	16.0%	17.0%	17.0%	17.0%	17.0%	100.0%
9	17.0%	16.0%	16.0%	17.0%	17.0%	17.0%	100.0%
10	17.0%	17.0%	16.0%	16.0%	17.0%	17.0%	100.0%
11	17.0%	17.0%	17.0%	16.0%	16.0%	17.0%	100.0%
12	17.0%	17.0%	17.0%	17.0%	16.0%	16.0%	100.0%
13	16.0%	17.0%	17.0%	17.0%	17.0%	16.0%	100.0%
Total Contribution:	**16.7%**	**16.7%**	**16.7%**	**16.7%**	**16.7%**	**16.7%**	**100.0%**

[2] PMIS – Project Management Information System

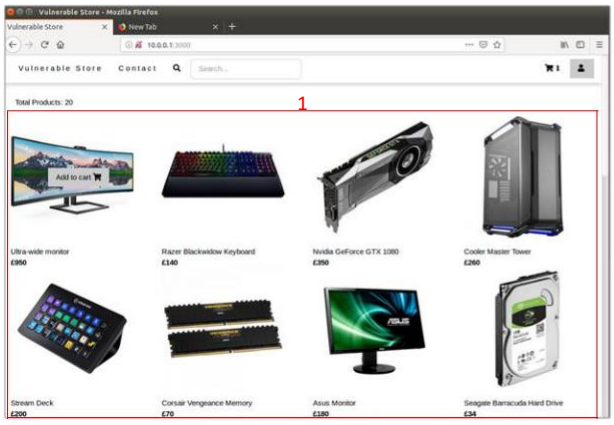
Appendices

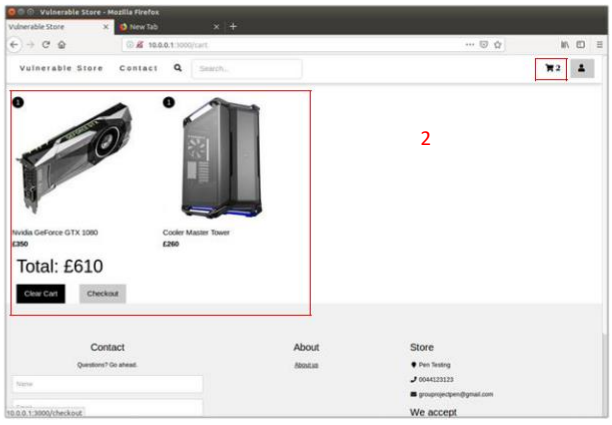
Appendix 1 – Key deliverables overview

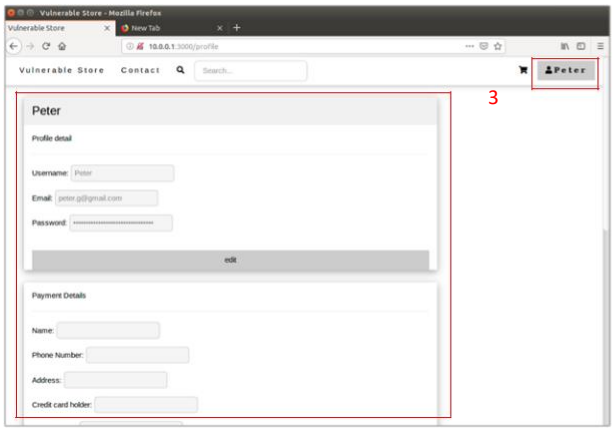
I. Website

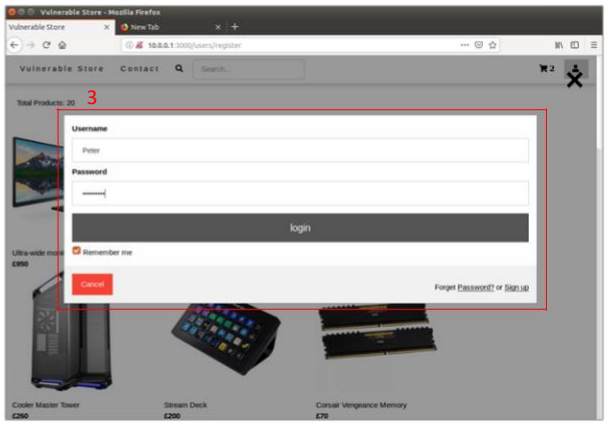
Website was created using NodeJS. Site is emulating online store selling computer accessories. Implemented functionality includes:

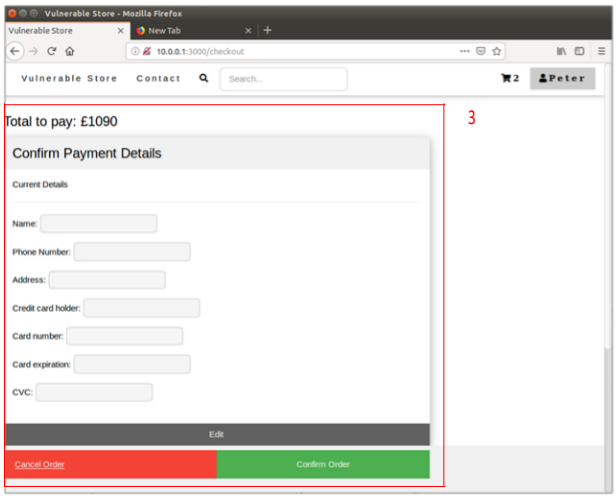
1) product portfolio
2) checkout and payment options
3) user profile registration and log-in
4) elevated access for administrator

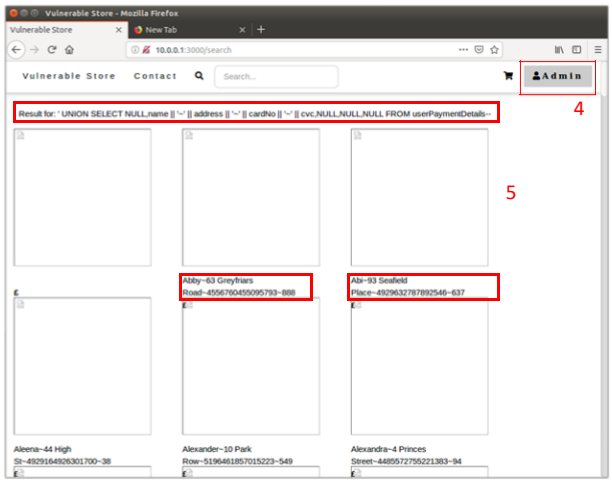












5) Website was tested for vulnerabilities to attacks like SQL injection and Cross Site Scripting. Such vulnerabilities were detected and will be described in separate point.

II. Configured victim machine

Windows server 2012 was chosen to host the website. It provides good balance between the age of the system and its utilization in the industry vs number of known vulnerabilities affecting it. There are more exploits available for older systems, but those have smaller presence therefore educational value is smaller as well.

System was configured and some additional software was installed. This was done to emulate the commercial server and to prepare grounds for various exploits. Some of the configurations/modifications are listed below:

- DHCP service added and configured
- OpenSSH server installed
- Firewall ports were open to allow access to above services (TCP 22, UDP 67)
- NodeJS network runtime environment was installed and e-commerce website source files were downloaded from the project's GitHub repository.
- IE Enhanced Security was disabled
- Microsoft PowerPoint 2013 was installed (necessary for exploit creating reverse shell by opening modified ppsx file)

III. Configured Attacker Machine

Kali Linux was chosen as an OS for this machine. It comes with range of penetration testing tools pre-installed (i.e. Metasploit framework or Hydra). On top of that some additional software was installed:

- Raw-Packet (set of Python scripts developed for penetration testing)

IV. Attacks researched and tested

This section describes principles behind a range of researched attacks. Detailed description on how to perform them are listed in separate files as part of Product Documentation.

I. Password attacks:

- **SSH (Secure Shell) brute force attack.** Here we enter the system using OpenSSH terminal server installed on the machine. We take advantage of its default settings where access is granted based on Active Directory user credentials and no time-out mechanism is implemented. That allows us to try multiple combinations of usernames and passwords until match is found. Process is automated using Metasploit framework.

- **Website login brute force attack.** Here we are using the Hydra password cracking tool to make multiple login attempts to the website until there is a match. Since no timeout feature is implemented in the service that can be done with high speed. To do that, details of request that is sent to the server during the login process must be established first. That can be done by inspecting the code using developer tools included in the browser. Once established, request can be passed to Hydra as a parameter. We can also specify details of attack i.e. - whether we want to use list of usernames and passwords or purely brute force by generating those details.

II. DHCP (Dynamic Host Configuration Protocol) starvation and spoofing attack.

This attack was not specifically listed in MoSCoW table, however in customer meetings it was mentioned, that they are interested in wide range of vulnerabilities and the team should deliver as many interesting exploits as they can.

This attack takes advantage of the underlaying mechanism behind DHCP protocol. Attacker fills up lease pool of the victim DHCP server with bogus requests. Once server has no more addresses to assign and stops responding to new requests, attacker deploys their own server. Unaware machines will receive malicious addressing, gateway and DNS information specified by attacker.

III. Eternal Blue MS17_010

This exploit takes advantage of vulnerability in Microsoft's implementation of Server Message

Block protocol (SMBv1). This protocol allows for sharing access to the files, printers and serial ports. It contains three significant bugs that can be exploited by sending specially crafted packets. Simplifying, attacker can manipulate kernel's non-paged memory leading to buffer overflow and execution of malicious code. We are using Metasploit framework as it contains module that automates this task. It uses the exploit to create a new session and open reverse shell from the victim machine to the attacker.

IV. MS14-060 Sandworm

Uses vulnerability in OLE - Windows Object Linking and Embedding standard (responsible for exporting digital content between different applications). Because of the OLE bug, opening specially crafted Power Point file will cause execution of arbitrary code. Attack involves

element of phishing as the file must be placed in shared folder on the network and then be open by unaware person. That triggers malicious code which downloads exploit controller and executes it. That in turn opens the reverse shell to the attacker's machine. This attack can be performed using Metasploit framework.

V. SQL (Structured Query Language) injection and database information extraction.

Data driven applications like websites tend to use database and scripting language combination to display dynamic content. If user input is not appropriately filtered it is possible to take advantage of forms and dialog boxes to pass SQL code back to the server. Certain usage of escape characters and SQL commands can disguise user-entered text as part of the code. That will result in additional output being returned. With trial and error investigation, structure and content of the database can be extracted. In this case, user and payment details were obtained. Then "Hashcat" tool was used to crack the passwords stored as md5 (Message Digest 5) hash strings.

VI. Meterpreter remote shell.

Meterpreter is a dynamic payload (code) that is injected into the victim machine memory. It attaches itself to one of the hosts processes and can dynamically extend its functionality by downloading additional modules from remote host. Meterpreter functionality is included in the Metasploit suite. Setting it up involves following steps:

- generate executable using Metasploit's sub-module called "Venom"
- deliver the file to the victim machine and execute it

As mentioned above, client-side payload must be delivered as an executable. That can involve some kind of social engineering or can be done by attacker if the system has already been compromised. In second case Meterpreter would act as an ongoing method of access after initial intrusion.

After full connection is established, Meterpreter provides multiple functionalities like:

- File management
- User management
- Privilege escalation
- Other shell functionalities

Above paragraphs provide brief description of some of the key developed attacks. Example of detailed walk-through document can be found in:

Appendix 5 – Example of detailed research document

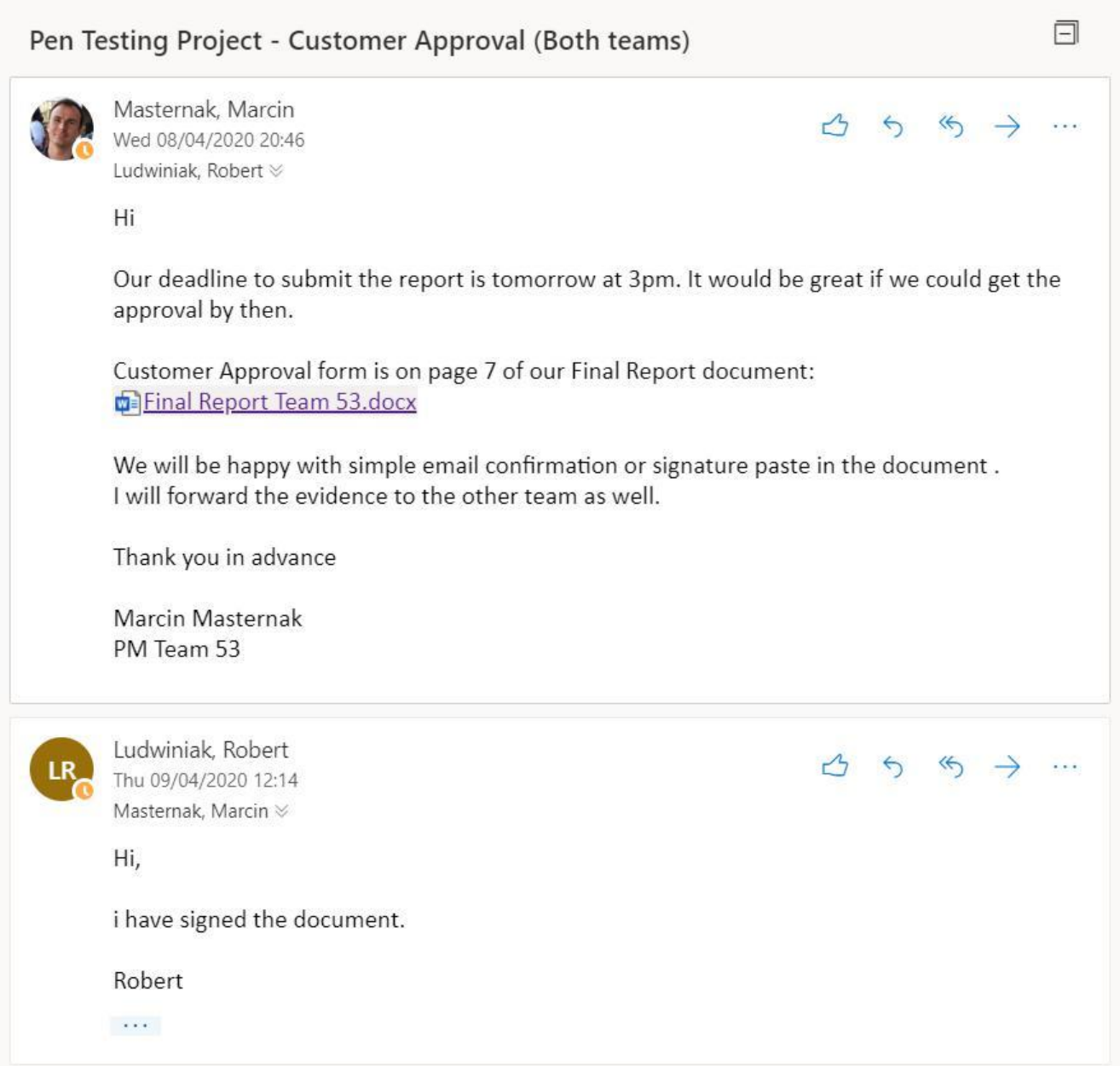
This is the end of this appendix.

If you used a link to get here and wish to pick up where you left off, click:

Return to Delivered Product

Appendix 2 – Client approval evidence

Email confirming that Customer signed the Approval section of this Report:



Pen Testing Project - Customer Approval (Both teams)

Masternak, Marcin
Wed 08/04/2020 20:46
Ludwiniak, Robert

Hi

Our deadline to submit the report is tomorrow at 3pm. It would be great if we could get the approval by then.

Customer Approval form is on page 7 of our Final Report document:
Final Report Team 53.docx

We will be happy with simple email confirmation or signature paste in the document .
I will forward the evidence to the other team as well.

Thank you in advance

Marcin Masternak
PM Team 53

LR
Ludwiniak, Robert
Thu 09/04/2020 12:14
Masternak, Marcin

Hi,

i have signed the document.

Robert

This is the end of this appendix.

If you used a link to get here and wish to pick up where you left off, click:

Return to client approval

Appendix 3 – Closing Audit evidence

I. Project Health Register screenshots

Preparation Audit

Weight	Item	Response
5	Is a Sponsor appointed to the project from the beginning?	100
4	Is the Sponsor involved enough in the high-level aspects of the project?	80
3	Does the Sponsor avoid getting themselves involved in the details?	100
2	Is the Sponsor familiar enough with the P2.express method?	90
4	Is there a clear Charter available from the beginning?	
5	Is the Project Manager appointed to the project from the beginning?	100
5	Is the Project Manager familiar enough with the P2.express method?	80
3	Are the Consultants clearly appointed to the project?	100
3	In case you're going to have a PM Support, are they clearly assigned to the project?	
4	In case you're going to have a PM Support, are they familiar enough with the P2.express method?	
5	In case you're going to have parts of the project done with internal teams, are all Team Leaders clearly appointed to the project?	100
3	In case you're going to have parts of the project done with internal teams, are all Team Leaders familiar enough with the P2.express method?	80
5	Did you set up the whole PMIS?	100
5	Is it clear and easy for everyone how to access any element in the PMIS?	100
3	Did you create the Configuration in a workshop?	80
3	Are there more than 30 elements in the Configuration mind-map?	100
5	Did you identify at least 10 risks in the RICs element?	80
4	Did you use a workshop for identifying and planning risks?	60
5	Do you have effective, actionable plans for risks?	80
	Score:	90

Help | Planned Improvements | **Preparation Audit** | Sheet1 | Weekly Audit | Customer Satisfactior ...

Weekly Audit

Item	W01	W02	W03	W04	W05	W06	W07	W08	W09	W10	W11	W12	W13	W14
Are the progress data reliable?		100	100	100	100	100	100	100	100	100	100	100	100	
Did the Project Manager add a short note to the weekly measures?		50	0	50	100	100	100	50	50	100	100	50	100	
Did the Sponsor check the Dashboard?		100	100	100	100	100	100	100	100	100	100	100	100	
Did all Team Leaders check the Dashboard?		100	100	100	100	100	100	100	100	100	100	100	100	
Did the Customer PM check the Dashboard?		100	0	100	0	100	0	0	100	0	100	0	100	
Did you create an actionable plan for recovering the deviations?		50	50	50	100	100	50	50	100	100	100	100	100	
Did you have en effective weekly kick-off with relevant stakeholders?		50	70	100	90	100	100	90	100	90	90	100	100	
Score %:		76	65	87	85	100	76	72	98	85	98	85	100	85.5833
Week	1	2	3	4	5	6	7	8	9	10	11	12	13	Average

Help | Planned Improvements | Preparation Audit | Sheet1 | **Weekly Audit** | Customer Satisfactior ...

Team Satisfaction Survey

Team satisfaction survey														
Please fill in with values 0 - 100 (apart last row)														
	Week	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**13**
Do you know what is expected from you, and what you can expect from others?		50	80	90	100	90	100	100	100	100	100	100	100	100
How effective is the communication between you and the person you directly report to?		50	80	100	100	100	100	100	100	100	100	100	100	100
How effective is the communication between you and your teammates?		90	100	100	100	100	100	100	100	100	100	100	100	100
How effective is the communication between you and your peers in the customer side?		70	90	90	80	80	80	90	90	70	70	90	80	90
How effective is the communication between you and your peers in the supplier side?		60	60	70	80	80	80	60	60	90	90	100	100	100
Are the project targets set realistically?		70	70	80	80	80	80	80	90	90	90	90	90	90
Does the project management system protect you, and help you work on the project comfortably?		70	70	70	80	80	80	80	80	80	80	80	80	80
Is the Project Manager supportive?		90	90	90	90	90	80	80	90	90	90	90	90	90
Do you have a clear image of the project as a whole? Do you know your role in this mission?		90	90	90	90	100	100	100	100	100	100	100	100	100
Overall, are you happy working in this project?		90	90	90	90	90	90	90	90	90	90	90	90	90
Overall, how do you rate the project management system?		80	80	80	70	70	70	80	90	80	80	80	80	80
Do you have any suggestions for improving the project management system?														
		74	**82**	**86**	**87**	**87**	**87**	**87**	**90**	**90**	**90**	**93**	**92**	**93**

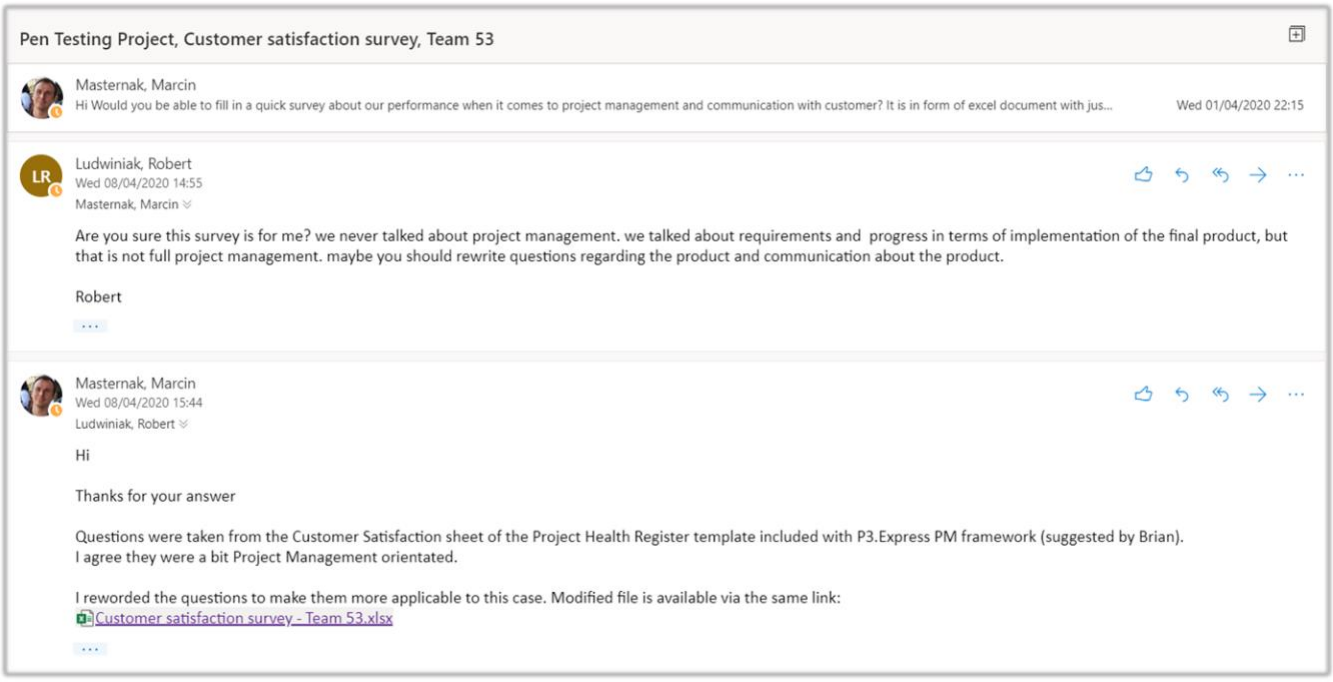
Team Satisfaction Summary

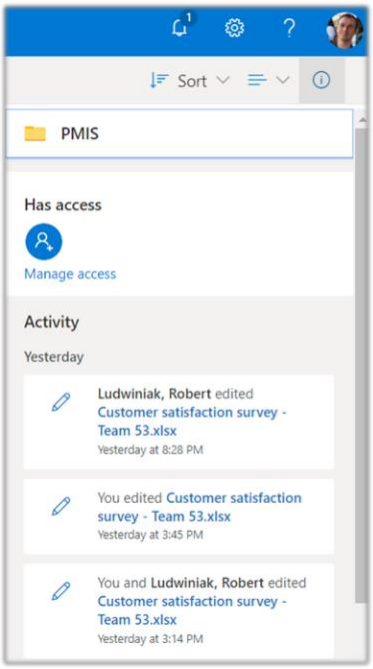
Overall team satisfaction percentage

Week	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**13**	Average
Satisfaction %	88	91	94	95	95	95	95	96	96	96	97	97	98	95

David | Jacob | Marcin | Matt | Tomasz | Mohammed | **Summary**

II. Evidence of Customer editing the satisfaction survey:





This is the end of this appendix.

If you used a link to get here and wish to pick up where you left off, click: Return to Closing Audit

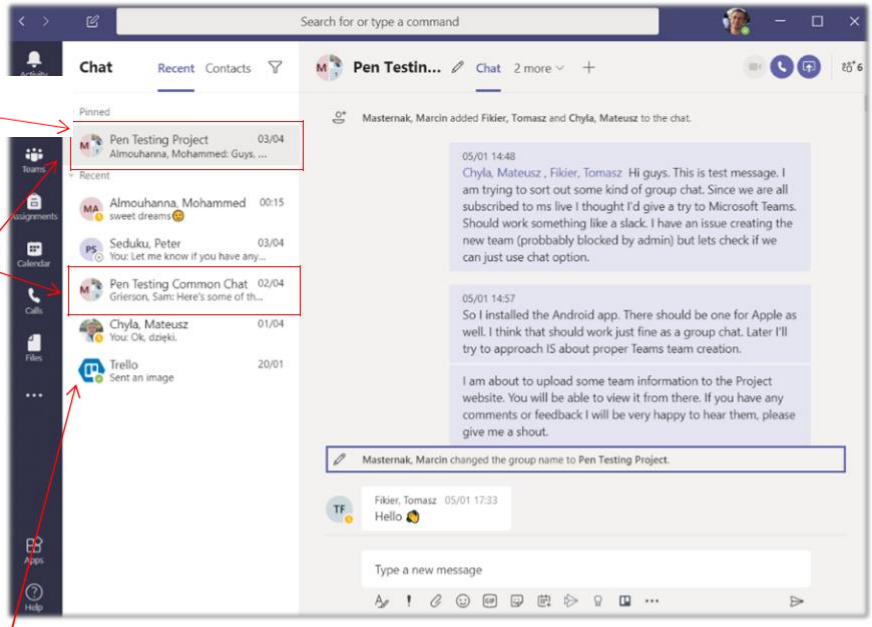
Appendix 4 – Lessons learned evidence

I. Communication and data exchange evidence.

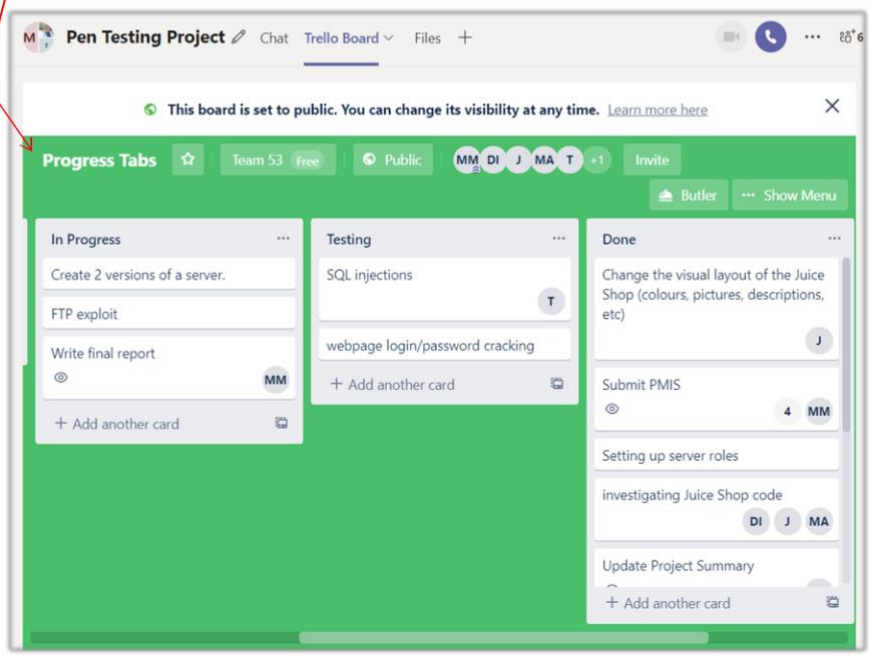
Screenshots below show our extensive usage of various communication channels, data exchange and online management techniques. Those were crucial to the success of the project and emergency operation under lockdown conditions.

Microsoft Teams was used as our main communication channel.

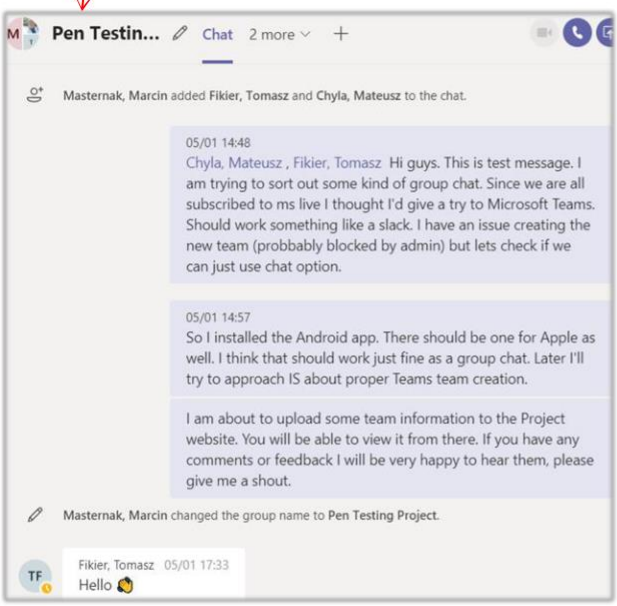
We introduced two chats. One for internal communication and one for combined discussion with other group.

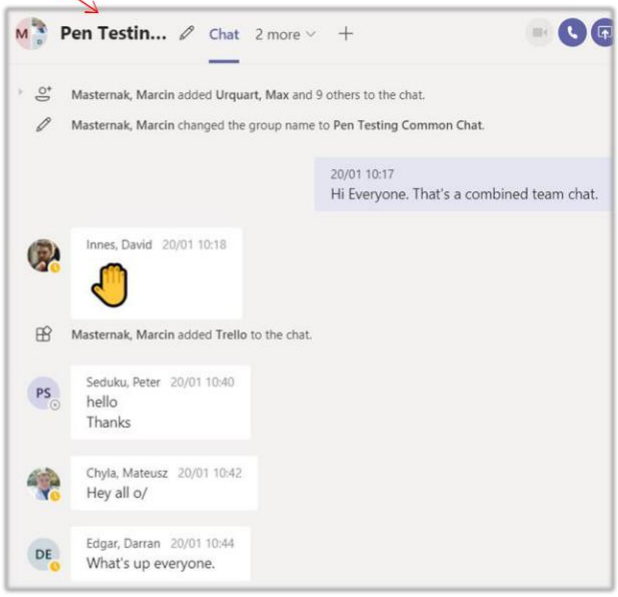


Trello board was integrated into our chats. That allowed for dynamic tasks assignment and progress monitoring.

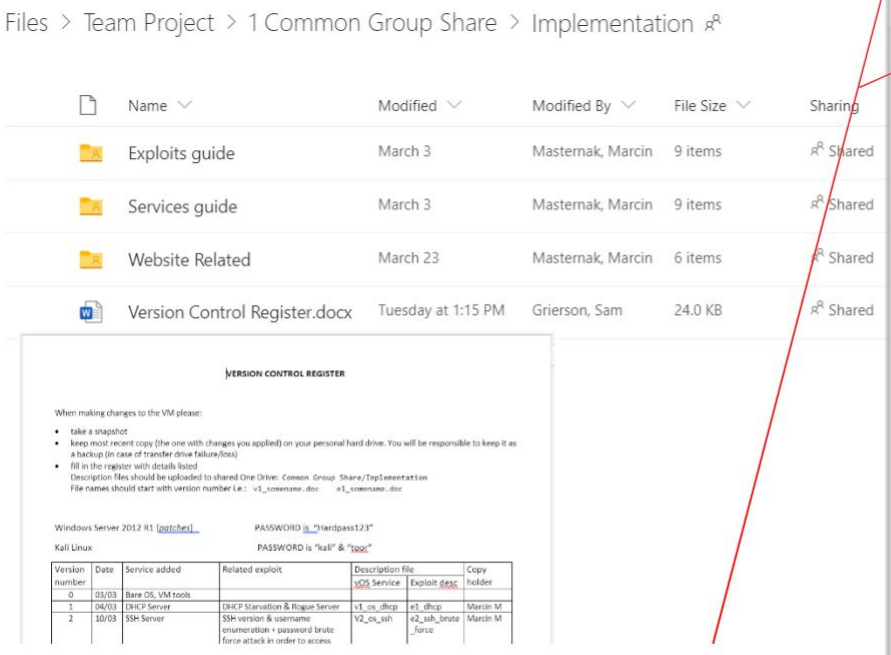


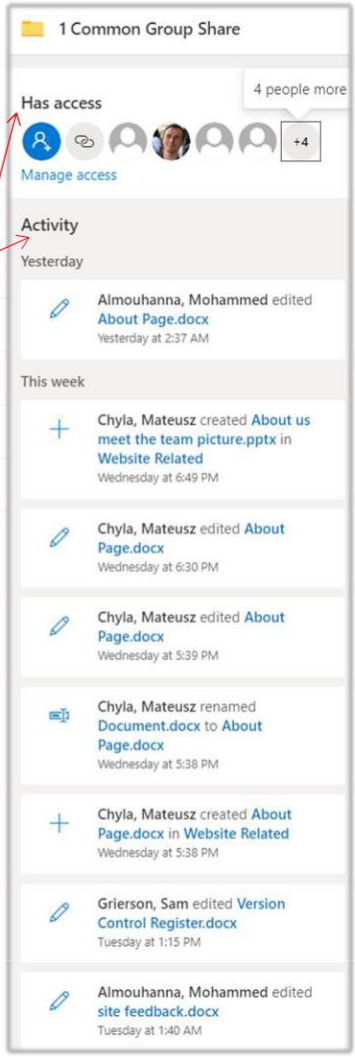
First messages exchanged with our communication channels.
Internal Team Chat and Common Team Chat.





Data storage and exchange structure was created on PMs One Drive:





Members of both teams were granted appropriate access levels and they actively contributed and edited resources.

II. Issue mitigation and contingency plan evidence.

All arising issues were recorded using RIC register.

ID	Description	Type	Date Identifie	Impact	Probability	Importance	Response	Owner
1	Scheduling Issues - due to the nature of the project precise scheduling can be difficult, that can lead to underestimating time needed for particular tasks resulting in delays.	Risk	1/13/2020	80	Medium	40	MITIGATED - Workshop with team members, suppliers and customer. Discussion about realistic timing particular tasks. Agreeing to work in short sprints delivering in small chunks.	Marcin M
2	**Specification Breakdown Issues**	Risk	1/13/2020	80	Medium	40	Workshop with team members, suppliers and customer. Customer is asked to provide more detailed specifications/expectations at next meeting.	Tomasz F
3	**Poor Productivity** - Team not producing results despite contributing appropriate amount of time. May be caused by multiple reasons ranging from lack of skill to poor organization.	Risk	1/13/2020	150	Medium	75	MITIGATED - Introducing short work cycles of 2-3 weeks where results need to be presented at the end of each cycle. Introducing accountability by assigning tasks on Trello board.	Marcin M
4	**Scope creep** - New, non-essential functionalities being added to the system during the course of the project. Leads to delay and can affect other subsystems.	Risk	1/13/2020	100	Medium	50	MITIGATED - Establish priorities with customer. Specify and visualize deliveries using configuration map. Use weekly meetings to identify unnecessary development efforts.	Marcin M
5	**Kick off delay -** There are number of processes taking place at the initial stage of the project. Team creation and assigning member roles, establishing communications, Client and Sponsor meetings/negotiations, defining project's requirements and other. All of them are a possible sources of delay, meaning that significant amount of time can be lost in the organizational stage. That leaves less time for actual implementation.	Risk	1/13/2020	100	Medium	50	MITIGATE - Follow P3 express guide. Create team and establish internal communications early. Take notes and minutes from the meetings and plan the agenda for further ones. Spot sources of delay and agree on tasks delegation among the team members. (I.e. if technical team member does not have anything to do in their field yet, they may aid in organizational tasks).	Marcin M
6	**Kick off delayed** -Client's expectations were slightly different and less specific then indicated by the project brief. That resulted in increased amount of initial research required, leading to delay of about week.	Issue	2/3/2020	100		100	Progress will be carefully tracked over next few weeks. Technical decisions have been made now and tasks were assigned. Increased input from team members over next couple weeks should be sufficient to make up for the delay. There are 2 weeks of contingency over the Easter break as a last resort.	Marcin M
7	**Communication and cooperation issues. Supplier risk -** There are multiple stakeholders in the project. Client, Sponsor and two teams of six. It is possible for misunderstandings and differences of opinions to form. Workload division between two teams, and members relative lack of team-work experience is a	Risk	1/13/2020	200	Low	60	Weekly team, client and sponsor meetings. Initial agreement on rules of cooperation. Establishing communication channel and data share with external team using MS Teams and One Drive. Creating and making sure everyone is familiar with PMIS documents (i.e. RIC register - can help solve problems)	Marcin M

8	**Technology risk. Infrastructure problems and data loss** - for successful delivery teams require platforms for data storage/exchange, communications, OS and network virtualization. Failure of any of those services can have a varying impact. From minor communication issues to loss of the entire progress to date.	Risk	1/13/2020	100	Medium	50	MITIGATED - Research and PM data is stored in Microsoft cloud providing very high level of reliability. Additional copy is located on PM's hard drive. Data that can not be backed up in cloud like live systems and test configurations are to be implemented as VM's and backed up individually by each team member.	Marcin M
9	Customer will not provide the hosting for development server. Servers provided by the University are not suitable due to necessity of implementing multiple Windows hosts, network and firewall for testing.	Issue	2/3/2020	50		50	MITIGATED - Each team member will create virtualized environment on their own machine. Services and exploits will be researched in chunks. Partial implementations will be merged on master VM copy later in the project.	Tomasz F
10	**Team member unavailability and underperformance.** - That could be due to illness, family commitments, other unexpected situations or lack of engagement. Main resource for this project is the time that team members can put in. Losing one or more team members for significant period of time will have negative impact on delivery time.	Risk	1/13/2020	100	Medium	50	MITIGATED/PLANNED FOR - Online communication channels, cloud storage and distributed workload allows team members to contribute remotely with flexible hours. Members contribution is recorded in PMIS. Regular team meetings are held and underperforming will be discussed on forum. In case of more serious incapacity/unavailability, extra contribution can be made over the Easter holidays.[CONTINGENCY PERIOD]	All team
11	Website Development issue - current approach turned out to be too complicated and had to be scrapped.	Issue	2/24/2020	200		200	MITIGATED - new website will be designed using team members templates from previous project, that will allow for quick deployment and should make up for some of the lost time	Jakub Hurkala
12	Difficulty introducing prototype. Lack of common deployment environment causes the problem with integrating sub tasks.	Issue	3/3/2020	100		100	Development VM was created on dedicated portable hard drive. VM will be passed around team members. Version control will be implemented using online register and VM smapshots. Individual team members will also perform hard backups on their own machines.	All Team
13	Communication, data exchange and delivery risk due to COVID outbreak. Due to the outbreak all meetings and live teaching are cancelled. That makes cooperation between team members more difficult. Also communication with Sponsor and Client will be affected.	Issue	3/17/2020	200		200	MITIGATED - Utilizing existing online communication channels. Expanding utilization of online cooperation tools. Customer and Client meetings will be held using Skype and WebEx. Progress demostration will be done using desktop video capture uploaded to MS Stream corporeate video service. There is ongoing discussion within team on how to minimize impact of the situation.	All team
14	Difficulty with communication due to overloaded and poor quality communication channels. i.e. Skype conference call with client was suffering from poor quality.	Issue	3/20/2020	100		100	MITIGATED - changed comms channels. Team starded using Discord for live streams as it provides superior call quality.	

Issues were discussed in the weekly meetings. Below are example minutes, taken at one of the meetings, where two key issues affecting the project at the time were discussed:

- Hosting the Server and version monitoring
- New approach to the website development

DATE OF MEETING	03.03.2020	**TIME**	14:30
LOCATION	Room MER_C28	**MINUTE TAKER**	Marcin Masternak
PRESENT	David Innes, Mohammed Almouhanna, Marcin Masternak, Jakub Hurkala, Team 64		
APOLOGIES	Mateusz Chyla, Tomasz Fikier		

ITEM	TOPIC	DISCUSSION/ACTION	TEAM MEMBER	TIMESCALE
1	Meeting with the client.	We discussed • how to deploy test VM and introduce version control • research progress • functioning website and demo VM deadline • customer satisfaction sheet	All	15 min
2	Team meeting	Team members discussed outcomes of customer meeting.	All	15 min

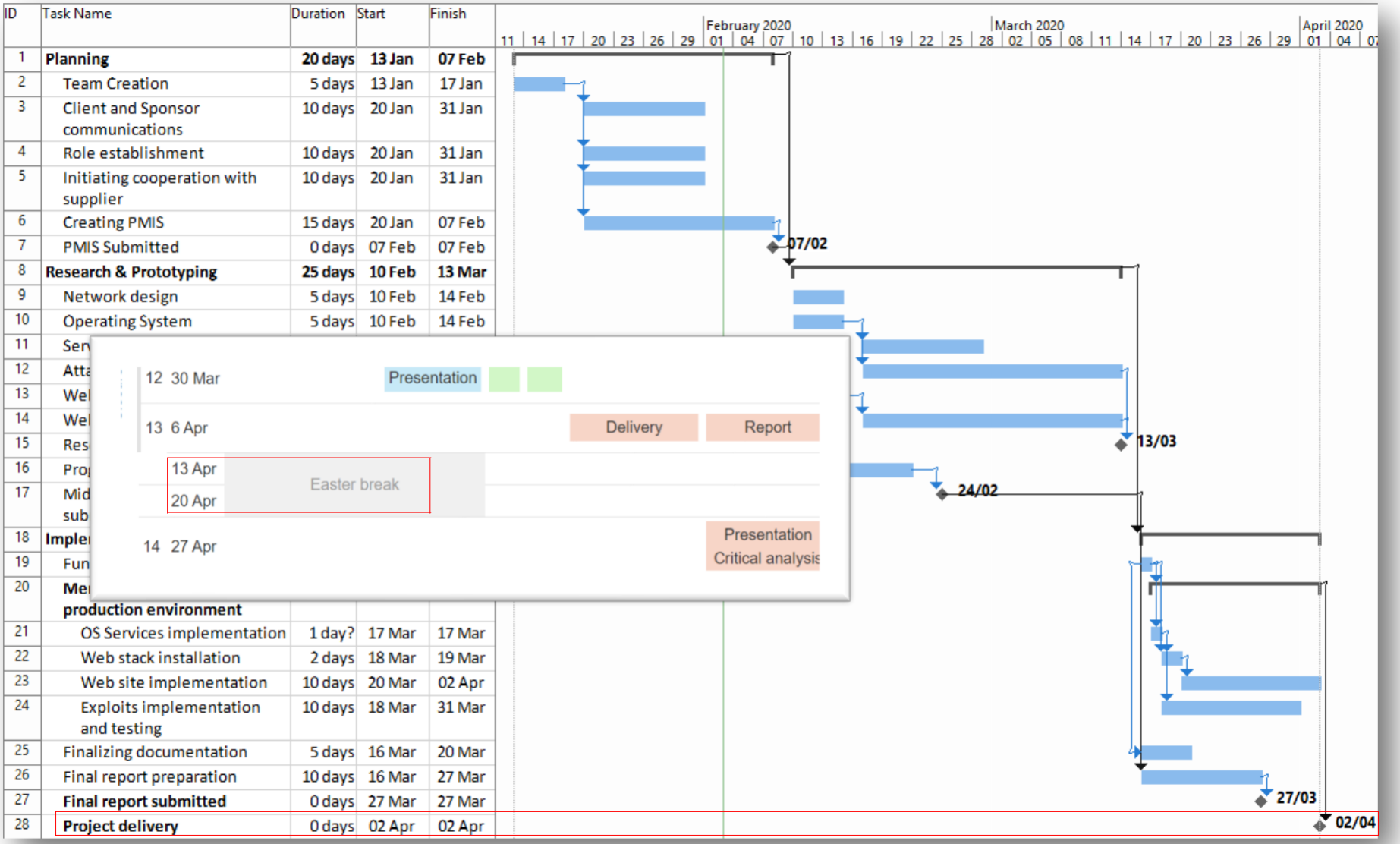
NOTES:

- Decision was taken to put a potable hard drive in circulation. Person holding the drive will implement their research, create a backup and pass the drive over to the next team member. Version control sheet will be created and maintained.
- Windows Specific Vulnerabilities Survey and Password Cracking were discussed. Multiple exploits mentioned in this report were researched and tested and are ready for implementation.

  Password cracking is in development and on track.

  There is new approach to the website. Site will be shared on Git. It was agreed that cooperation is needed to finish off functionality and design.

Contingency period was included in the Project Plan. As shown in the Module Timeline two weeks over Easter break were reserved for applying finishing touches to the product.



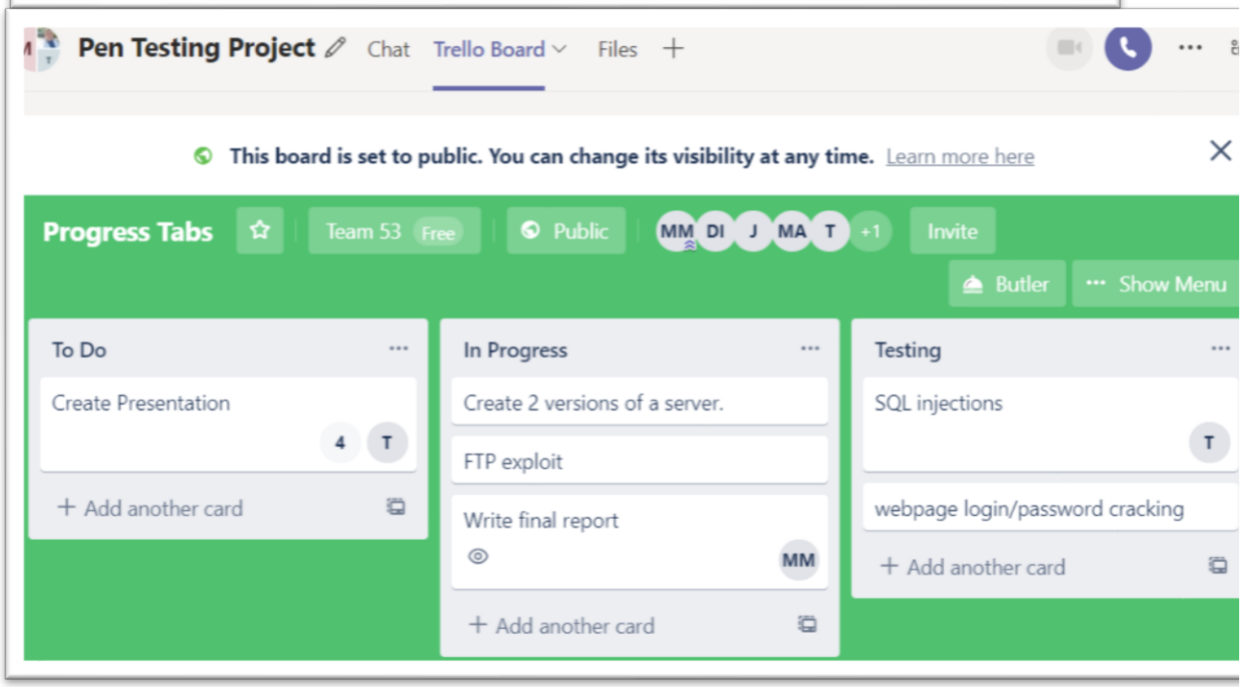
ID	Task Name	Duration	Start	Finish
1	**Planning**	**20 days**	**13 Jan**	**07 Feb**
2	Team Creation	5 days	13 Jan	17 Jan
3	Client and Sponsor communications	10 days	20 Jan	31 Jan
4	Role establishment	10 days	20 Jan	31 Jan
5	Initiating cooperation with supplier	10 days	20 Jan	31 Jan
6	Creating PMIS	15 days	20 Jan	07 Feb
7	PMIS Submitted	0 days	07 Feb	07 Feb
8	**Research & Prototyping**	**25 days**	**10 Feb**	**13 Mar**
9	Network design	5 days	10 Feb	14 Feb
10	Operating System	5 days	10 Feb	14 Feb
11	Ser			
12	Atta			
13	Wel			
14	Wel			
15	Res			
16	Pro			
17	Mid sub			
18	**Imple**			
19	Fun			
20	**Me production environment**			
21	OS Services implementation	1 day?	17 Mar	17 Mar
22	Web stack installation	2 days	18 Mar	19 Mar
23	Web site implementation	10 days	20 Mar	02 Apr
24	Exploits implementation and testing	10 days	18 Mar	31 Mar
25	Finalizing documentation	5 days	16 Mar	20 Mar
26	Final report preparation	10 days	16 Mar	27 Mar
27	**Final report submitted**	0 days	27 Mar	27 Mar
28	**Project delivery**	0 days	02 Apr	02 Apr

In order to avoid delay, independent tasks were planned for simultaneous development. MoSCoW method and Trello board were used to prioritize task assignments.

MoSCoW Priority Table

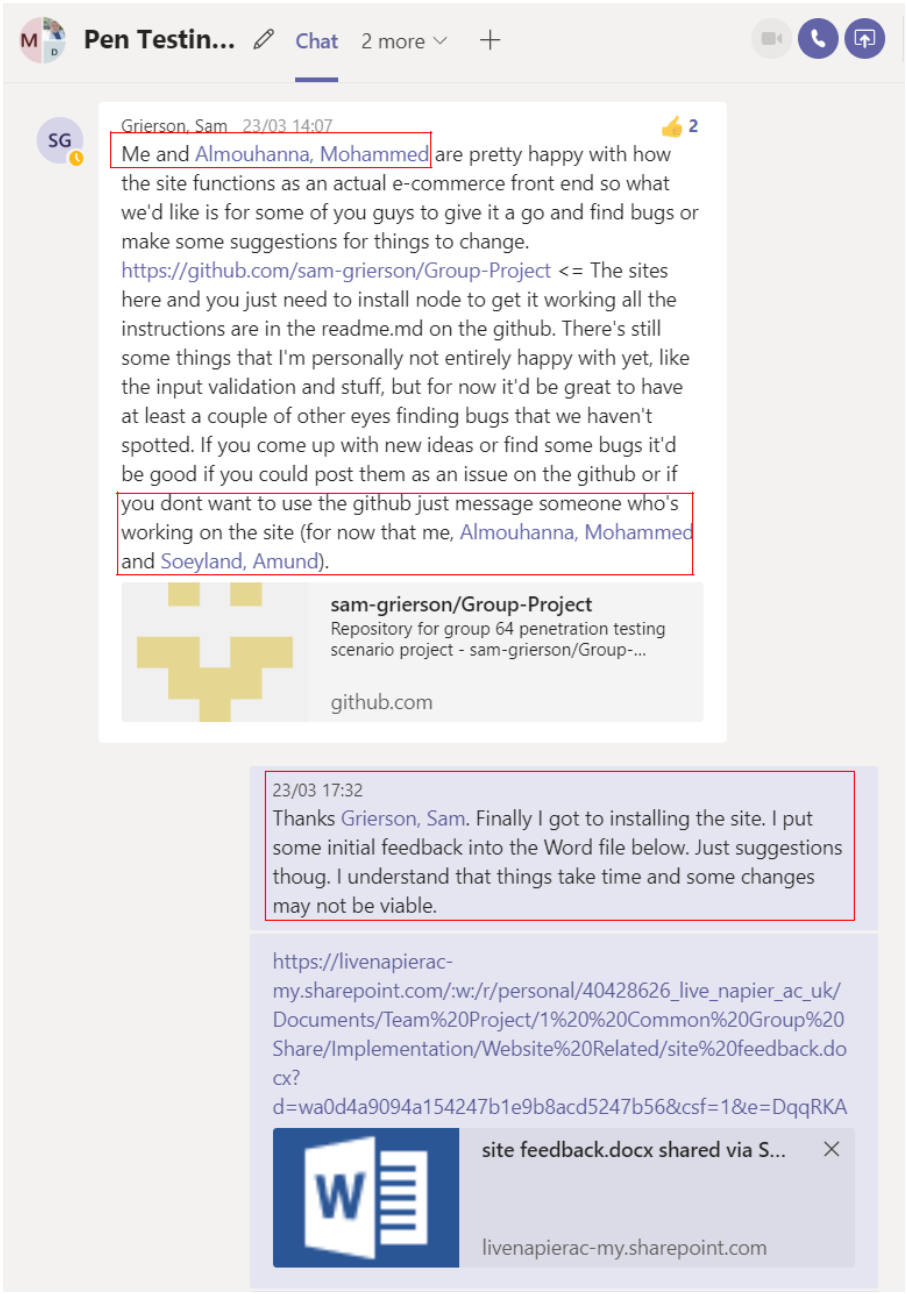
Penetration Testing Scenario

Must have	Should have	Could have	Won't have
Configured Server Machine	Password Attacks Privilege escalation ARP Poisoning DNS Poisoning Reverse Shell Scanning and Target Analysis Java Vulnerabilities SQL injection Cross site scripting FTP exploits	Backup Server Firewall configuration Hosts configuration Network policy definition Email server DNS server	Encryption
Appropriate number of specified Attacks & Vulnerabilities implemented.			
E-commerce web site	Login Multiple access levels Product portfolio User profile Payment/checkout		
	Network Share Documentation		



III. Cooperation with supplier - evidence.

This is example of inter-team cooperation. Here Sam Grierson, Amumd Soyeyland (Team 63) and Mohammed Almouhanna (Team 53) are mentioned as working together on the website. Marcin Masternak (Team 53) contributes by providing the feedback. Screenshot was taken from "Common Group Chat" channel in Microsoft Teams.



This is the end of this appendix.

If you used a link to get here and wish to pick up where you left off, click: Return to Lessons Learned

Appendix 5 – Example of detailed research document

SQL INJECTIONS WALKTHROUGH

Quick Nmap scan shows that the server has port 3000 open for http services.

```
root@root:~# nmap -sV 10.0.0.0/24
Starting Nmap 7.70 ( https://nmap.org ) at 2020-03-26 16:14 GMT
Nmap scan report for 10.0.0.2
Host is up (0.00098s latency).
Not shown: 987 closed ports
PORT      STATE SERVICE            VERSION
21/tcp    open  tcpwrapped
80/tcp    open  http               Microsoft IIS httpd 8.0
135/tcp   open  msrpc              Microsoft Windows RPC
139/tcp   open  netbios-ssn        Microsoft Windows netbios-ssn
445/tcp   open  microsoft-ds       Microsoft Windows Server 2008 R2 - 2012 microsoft-ds
3000/tcp  open  http               Node.js (Express middleware)
3389/tcp  open  ssl/ms-wbt-server?
49152/tcp open  msrpc              Microsoft Windows RPC
49153/tcp open  msrpc              Microsoft Windows RPC
49154/tcp open  msrpc              Microsoft Windows RPC
49155/tcp open  msrpc              Microsoft Windows RPC
49156/tcp open  msrpc              Microsoft Windows RPC
49157/tcp open  msrpc              Microsoft Windows RPC
MAC Address: 00:0C:29:75:10:F7 (VMware)
Service Info: OSs: Windows, Windows Server 2008 R2 - 2012; CPE: cpe:/o:microsoft:windows
```

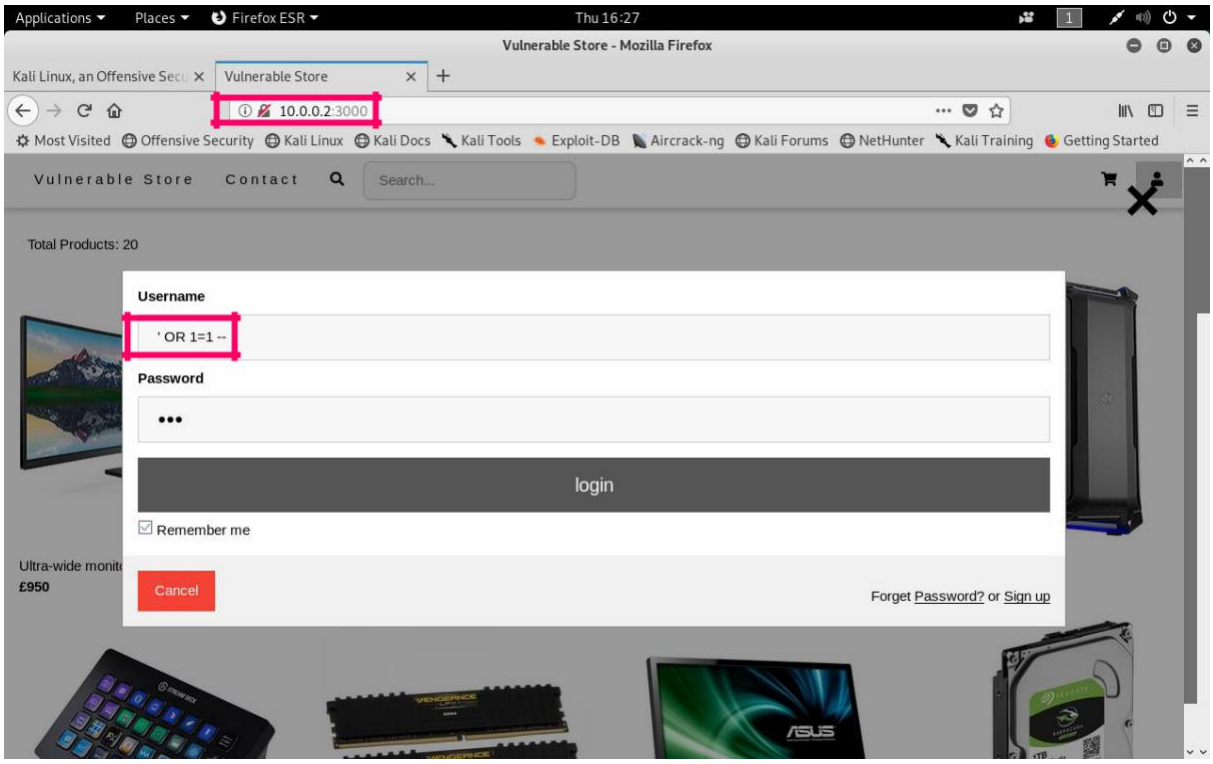
From Kali machine browse server's address with the port number. That will bring up the home page; login using ' OR 1=1-- with any password. It bypasses login verification and logs in as Admin. Consider this:

```
name = 'admin' OR 1 = 1
```

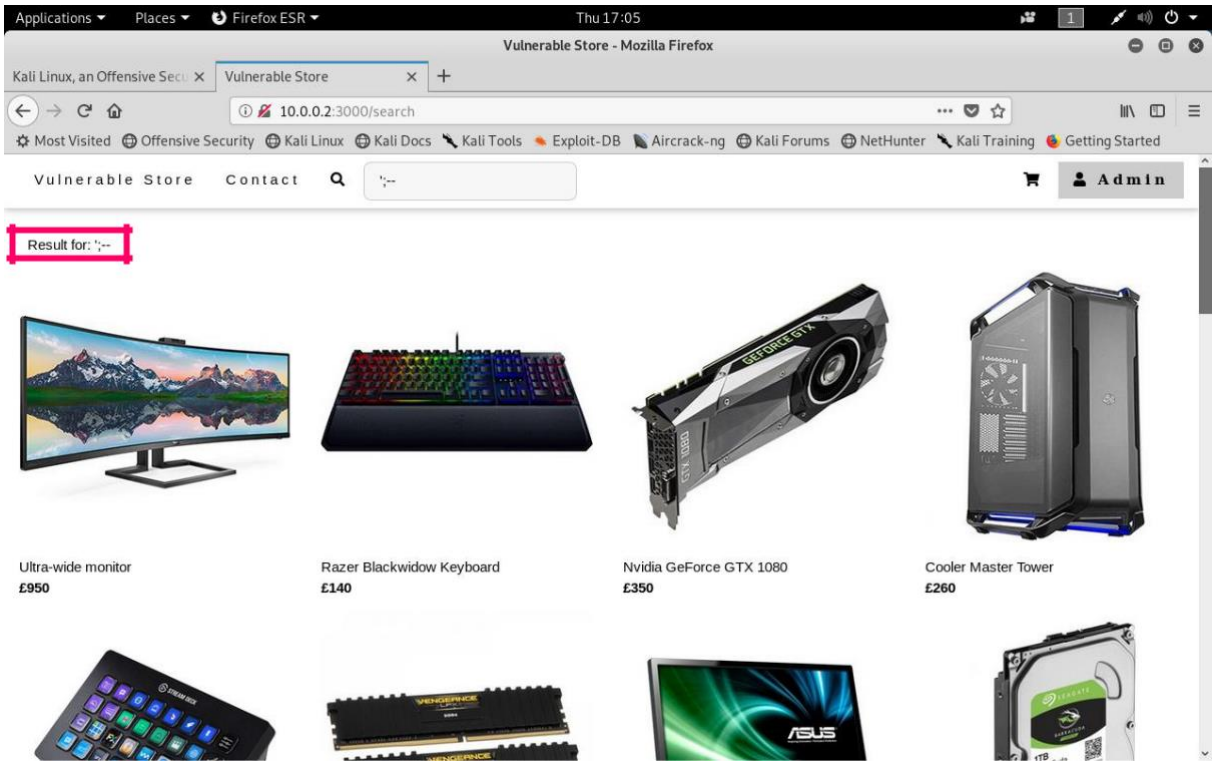
,

```
--
```

everything after double dash is seen as a comment



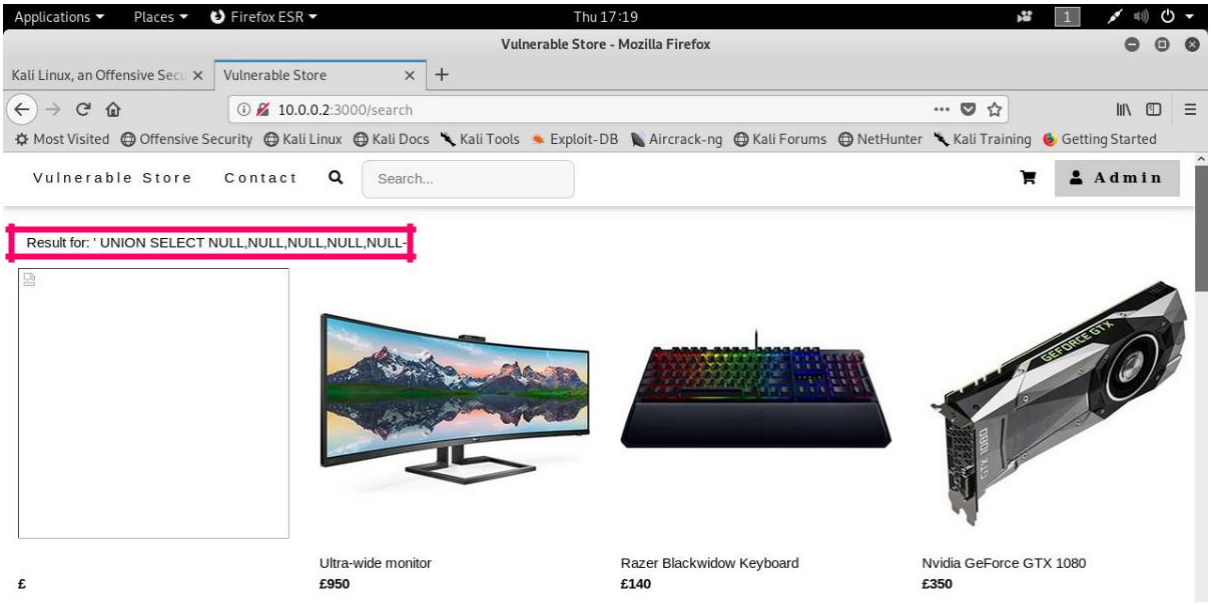
To see if application is vulnerable to SQL injections enter `';--` in the search box, if it comes back with an error that means it's been sanitised, if not then happy days.



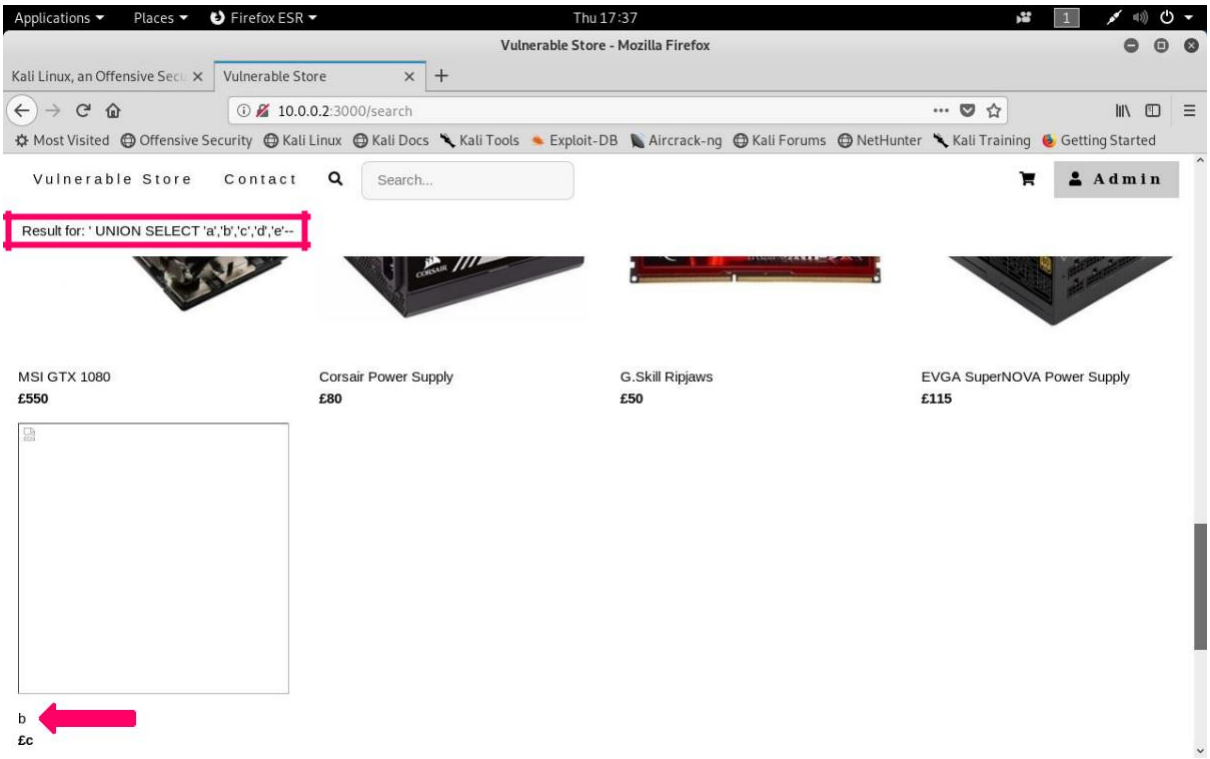
When an application is vulnerable to SQL injection and the results of the query are returned within the application's responses, the UNION keyword can be used to retrieve data from other tables within the database. The UNION keyword lets you execute one or more additional SELECT queries and append the results to the original query.
To carry out an SQL injection UNION attack, you need to ensure that your attack meets these two requirements. This generally involves figuring out:

- How many columns are being returned from the original query?
- Which columns returned from the original query are of a suitable data type to hold the results from the injected query?

To determine number of columns required in an SQL attack use a series of UNION SELECT payloads specifying a different number of NULL values. If the number of NULLs does not match the number of columns, the database does not return anything or returns an error. The reason for using NULL as the values returned from the injected SELECT query is that the data types in each column must be compatible between original and the injected queries. Since NULL is convertible to every commonly used data type, it maximizes the chance that the payload will succeed when the column count is correct.



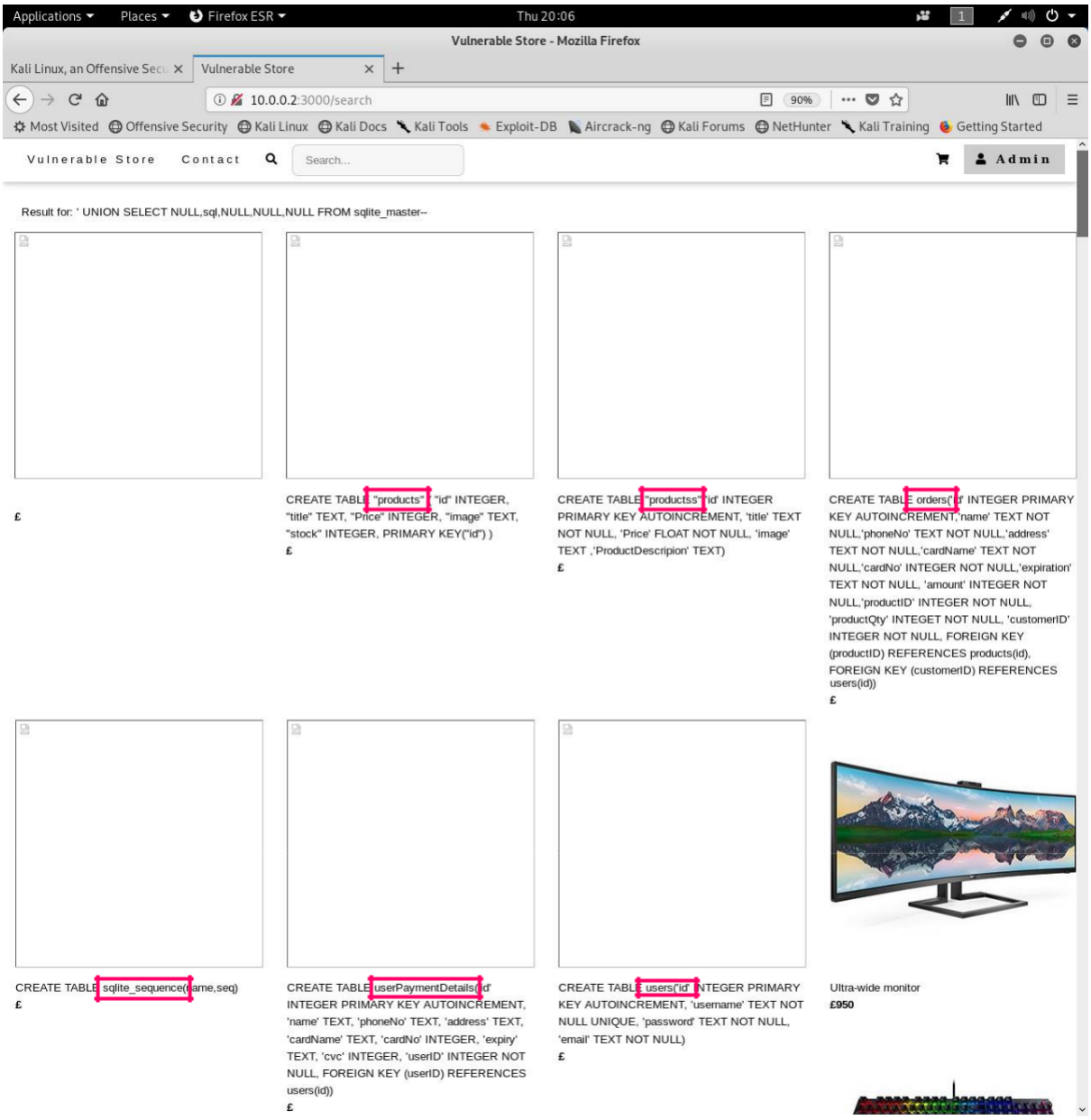
Having already determined the number of required columns, which is 5. You can probe each column to test whether it can hold string data by submitting a series of UNION SELECT payloads that place a string value into each column in turn. Use this query: **`' UNION SELECT 'a','b','c','d','e'--`** .



Query returns only the **b** value, which means that we will be able to retrieve data from the second column.

Every SQLite database has a sqlite_master table that defines the schema for the database. Query this in the search box:

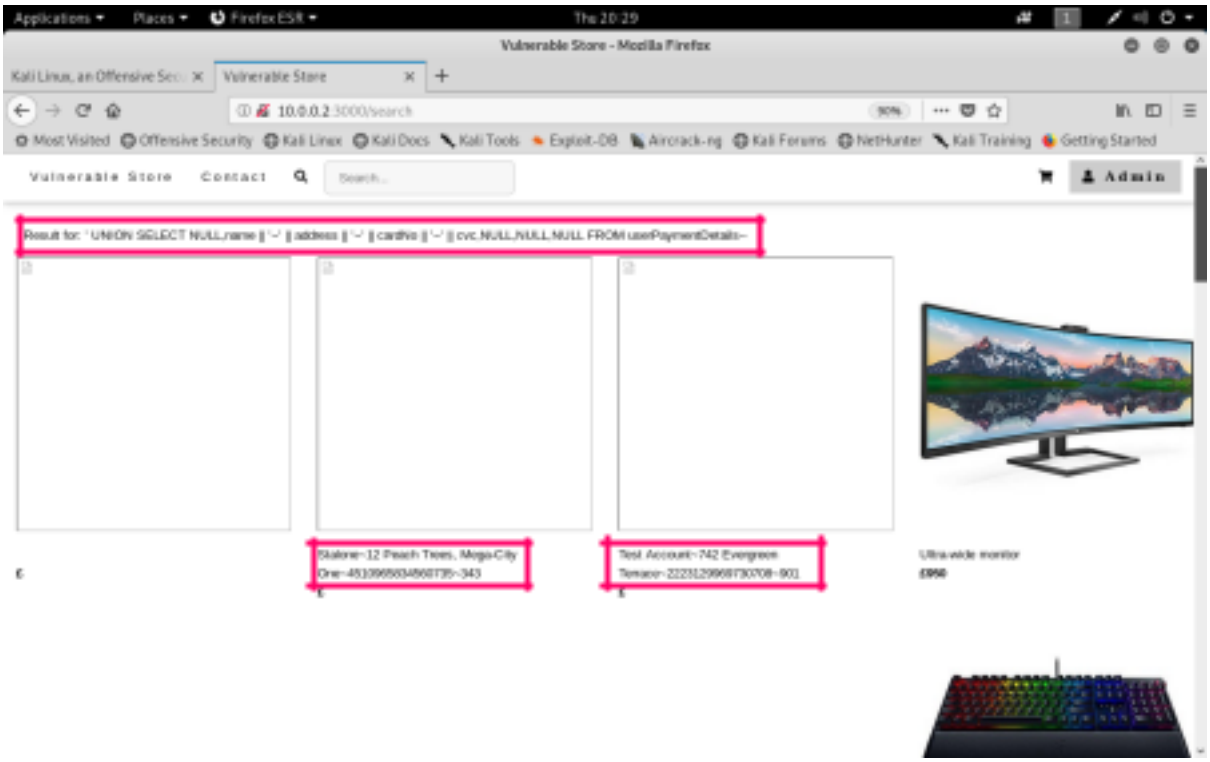
' UNION SELECT NULL,sql,NULL,NULL,NULL FROM sqlite_master-- to retrieve all table names and columns in database.



So now we can see what tables are available in the database. The one we really would like to exploit is userPaymentDetails and users.

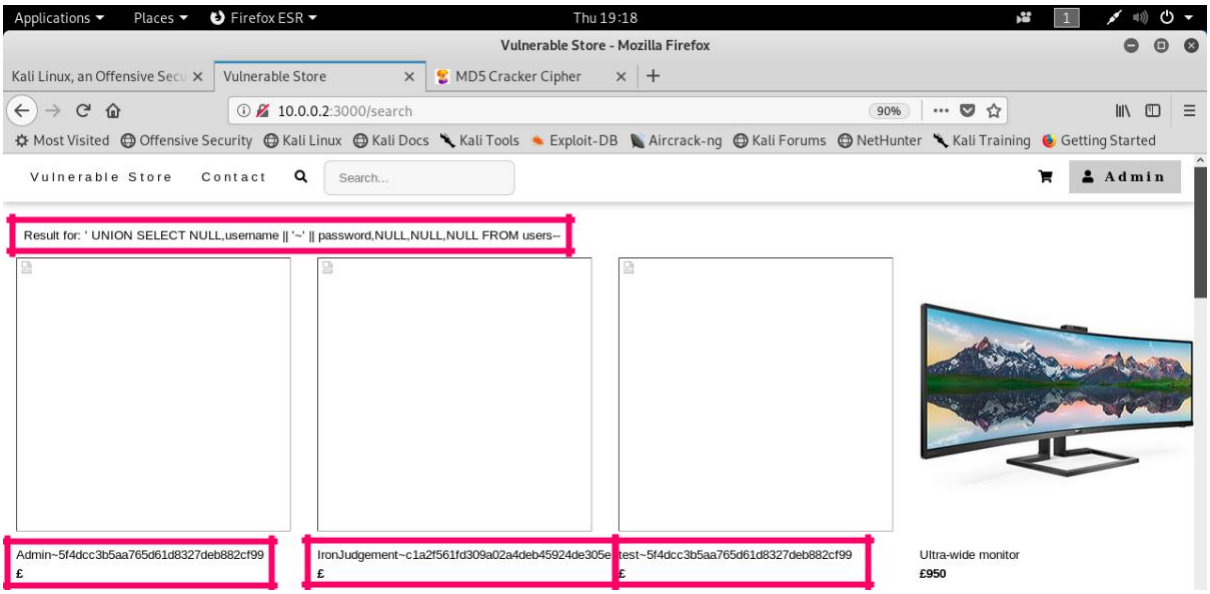
To get bank card details query in the search box:

‘ UNION SELECT NULL,name || ‘~’ || address || ‘~’ || cardNo || ‘~’ || cvc,NULL,NULL,NULL FROM userPaymentDetails--

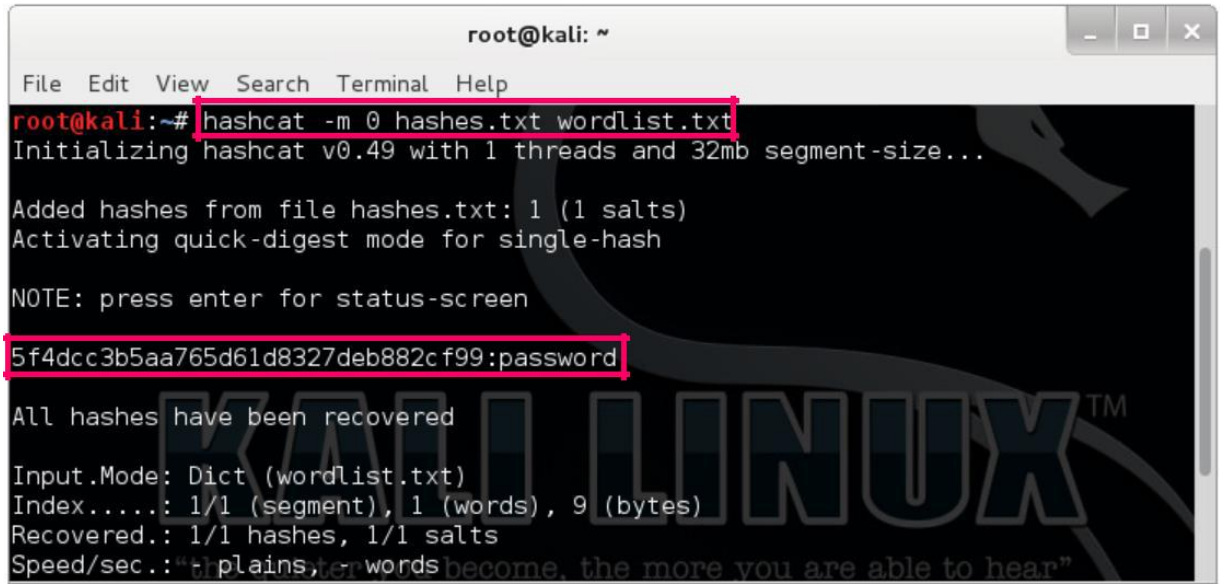


To retrieve usernames and passwords, query this in the search box:

‘ UNION SELECT NULL,username || ‘~’ || password,NULL,NULL,NULL FROM users--



Query returns usernames and passwords from the database. Passwords are hashed with MD5. We will use **hashcat** on kali to get Admin's password.



Admin's password is 'password'.

This is the end of this Appendix.

If you used a link to get here and wish to pick up where you left off, click:

Return to Deliverables Overview

References

[1] p3.express, “P3.express Online Manual” viewed 01 April 2020, <https://p3.express/online-manual/v01/overview/>