



{Client Name}

{Month} {Year}

{Client Name} Red Team Assessment Report

Client Details

Company Name: {Client Name}

Contact Person: {Person Name}

Address: {Address Name}

Email: {Email Address}

Telephone: {Telephone Number}

Document History

Version	Date	Author	Remark
1.0	{Date}	{Author}	Document Creation

SAMPLE

Table of Contents

1. About Payatu.....	5
2. Document Map.....	6
3. Introduction.....	7
3.1 Scope and Objective	7
3.1.1 Objective.....	7
3.1.2 Scope	7
3.2 Exclusions	8
3.3 Goals.....	9
3.4 Deliverables.....	9
3.5 Findings.....	9
4. Executive Summary	10
4.1 Summary	10
4.1.1 Summary of Strength	10
4.1.2 Summary of Weaknesses	11
4.1.3 Conclusions.....	12
4.2 Vulnerability Chart.....	13
4.3 Findings by Category	14
4.4 Table of Findings	15
4.5 Table of Strengths/Failed test cases.....	16
5. Detailed Findings & Proof of Findings	17
5.1 Recon - Enumerated subdomains of <client-name> and performed stealthy scan.....	17
5.2 Recon - Creating Fake employee ID card is possible	28
5.3 Physical Security - Got Access to the building and ODC via tailgating/Sneaking multiple USB devices and a mini Laptop inside.....	30
5.4 Social Engineering - Gathered Employee credentials via email phishing activity	33
5.5 Social Engineering - Gained physical access to machines via social engineering with Employees who were on the premises.....	35
5.6 Social Engineering - Gathered Employee credentials via Fake Wi-Fi captive portal.....	38
5.7 Exploitation - Malware implant execution and obtaining a reverse beacon (privileged access)	41
5.8 Exploitation - NBT-NS and LLMNR poisoning to gather NTLMv2 hashes of domain users	44
5.9 Exploitation - Kerberoasting attack.....	46
5.10 Exploitation - Ineffective Access Control to the network.....	49
5.11 Web Application Security - Default credentials login (Internally hosted web apps)	52
5.12 Network Segregation - No network segregation for different internal sites/subnets	56
5.13 External Attacks - No 2FA for VPN access and no external device authentication (Gained VPN Access to internal network using credentials gathered during phishing activity)	59
6. Detailed Failed Test Cases / Strengths	61
6.1 Physical Security - Physical security at the perimeter was very strong at these locations (Location-2, Location-3).....	62
6.2 Exploitation: Failed to move laterally across different systems (LAPS Implementation)	63
6.3 Observation: System patch level is good against critical remote vulnerabilities	65
7. We Prescribe	66
Annexure A (Red Team Assessment storyline)	69
1.1 Location-1	69
1.2 Location-2	73

1.3	Location-3	74
1.4	Location-4	75
1.5	Email Phishing Activity	78
1.6	Password Spray.....	79
1.7	VPN Login	79
	Annexure B (Tools Used).....	80

SAMPLE

1. About Payatu

Payatu is a research-focused security testing service organization specialized in IoT and embedded products, web, mobile, cloud & infrastructure security assessments, and in-depth technical security training. Our state-of-the-art research, methodologies, and tools ensure the safety of our client's assets.

At Payatu, we believe in following one's passion, and with that thought, we have created a world-class team of researchers and executors who are bending the rules to provide the best security services. We are a passionate bunch of folks working on the latest and leading-edge security technology.

We are proud to be part of a vibrant security community and don't miss any opportunity to give back. Some of the contributions in the following fields reflect our dedication and passion.

- nullcon - nullcon security conference is an annual security event held in Goa, India. After years of effort put in the event, it has become a world-renowned platform to showcase the latest and undisclosed research.
- hardware.io - Hardware security conference is an annual hardware security event held in The Hague, Netherlands. It is being organized to answer emerging threats and attacks on hardware. We aim to make it the largest platform, where hardware security innovation happens.
- Dedicated fuzzing infrastructure - We are proud to be one of the few security research-based companies to own an in-house infrastructure and hardware for distributed fuzzing of software such as browsers, client, and server applications.
- null - It all started with null - The open security community. It's a registered non-profit society and one of the most active security community. null is driven totally by passionate volunteers.
- Open source - Our team regularly authors open source tools to aid in security learning and research.
- Talks and Training: Our team delivers talks/highly technical training in various international security and hacking conference, i.e., DEFCON Las Vegas, BlackHat Las Vegas, HITB Amsterdam, Consecwest Vancouver, nullcon Goa, HackinParis Paris, Brucon Belgium, zer0con Seoul, PoC Seoul to name few.

We are catering to a diverse portfolio of clients across the world, who are leaders in banking, finance, technology, healthcare, manufacturing, media houses, information security, and education, including government agencies. Having various empanelment and accreditations, along with a strong word of mouth, has helped us win new customers. Our thorough professionalism and quality of work have brought repeat business from our existing clients. We thank you for considering our security services and requesting a proposal. We look forward to extending the expertise of our passionate, world-class professionals to achieve your security objectives.

2. Document Map

This report consists of the following sections:

- **Introduction and Objectives**

In this section, we will discuss about the general information regarding the Red Team Assessment, including the scope duration, goals, deliverables and findings.

- **Executive Summary**

High level view of the information gathered during the assessment, usually using graphs or comparative numbers. This section is meant to provide a general understanding of the security status of the organization.

- **Vulnerability Chart**

This section is meant to provide an overview of the vulnerabilities found during Red Team assessment of <client-name>. The same are sorted by their severity.

- **Detailed Vulnerability Information**

For each issue, this section includes all relevant details, including a detailed security advisory, and all variants, affected URLs, examinations, screenshots, and fix recommendations. This section is used both to educate on the nature and impact of the different issues, and to provide guidance for their remediation.

- **Detailed Strengths Information/Failed test cases**

For each failed test case, this section includes the description of the attack vector and why it failed.

- **Overall Recommendation**

This section contains the overall recommendations for the found security issues

- **Annexure A (Red Team Assessment storyline)**

This section describes the whole assessment in detail with a timeline. This helps the reader of this report to understand the assessment process and outcome in a better way.

- **Annexure B (Tools Used)**

This section specifies the tools used in different phases of the assessment.

3. Introduction

This document provides the Red Team assessment report performed by **Payatu** on targets as defined in the scope section.



3.1 Scope and Objective

3.1.1 Objective

The objective of the assessment is to obtain a realistic measure of risk against <client-name> infrastructure and applications. The Red Team assessment will be replicating the type of attacks that could be initiated from the Internet, intranet, wireless, and physical security breach. It will provide the client with a controlled security test against the identified organizational assets, the interfacing systems, and the environment. Based on the outcome of a qualified and quality security assessment, <client-name> can actively measure the effectiveness of security controls based on the following:

- Network infrastructure security
- Wireless infrastructure security
- Application security
- Physical security
- User awareness
- Incident response process
- Patch management process
- Access control effectiveness
- Security Operations Centre (SOC) effectiveness
- Security product deployment/configuration effectiveness

3.1.2 Scope

- Replicating the type of attacks that could be initiated from the Internet on <client-name> infrastructure and applications including but not limited to
 - Web Servers and/or Applications
 - Mobile Application
 - Network
 - Servers

to identify vulnerabilities which can be exploited to

- Interfere with the operation of the server/application to cause a denial of service
- Bypass the security controls implemented
- Enumerate accounts
- Gain privileged access to the infrastructure/network/applications in scope
- Gain access to <client-name> internal network
- Copy/Access data

- All types of social engineering attacks like phishing, onsite/physical pretexting, impersonation etc. to extract credentials or other sensitive information from employees.
- Attempt to breach physical security checks at 4 locations as mentioned below to perform subsequent assessment including but not limited to
 - Gaining access to wireless Infrastructure and/or Intranet
 - Compromising/gaining access to email server
 - Gaining privileged access of enterprise portal & associated applications
 - Gaining domain admin privilege for normal user account or compromise domain admin accounts
 - Installation of functional and yet rogue access points
 - Tailgating inside work areas
 - Acquiring access to sensitive areas like Data Center
 - Planting devices inside <client-name> premises
 - Compromising/gaining access to application/database/backup/file server or any other business-critical server hosted inside/outside datacenter
 - Data exfiltration using an encrypted channel to avoid detection

The physical security assessment would be conducted at the following <client-name> locations:

Name of Premise	Address
Location-1	
Location-2	
Location-3	
Location-4	



3.2 Exclusions

- Compliance/Regulation based vulnerability assessment/audit such as PCI-DSS, ISO, etc.
- Implementation assistance
- Supply of any tool/technologies
- Testing sensor-based physical access control system
- Jumping walls, lock-picking/breaking, damaging office property
- Coercion technique as part of social engineering attacks



3.3 Goals

- Compromise/Gain access of Email Server
- Gain superuser privilege access of enterprise portal & associated applications
- Gain access to physical access of Data Center
- Compromise/Gain access the application/database/backup/file server or any other business-critical server hosted in inside/outside data center
- Gain domain admin privilege for normal user account or compromise domain admin accounts.
- Extraction of credentials or other sensitive information from employees using social engineering or any other technique
- Data exfiltration using an encrypted channel



3.4 Deliverables

- Data gathered during OSINT (Open Source Intelligence) Phase
- Attack test case executed during the life cycle of Red Team assessment
- The result of all the test cases (pass/failed) performed
- Vulnerabilities discovered or loopholes identified
- Story/details about successful attack path along with attack timeline.
- Proof of vulnerabilities detected and successful exploitation
- Business and technical impact
- Any custom scripts or proof of concepts developed for exploitation as part of the assessment
- Recommendations to fix the issues/vulnerabilities
- Any other relevant security issues will be shared
- Assessment of the actions, policies, and procedures of the Company's detection and response team (CIRT)
- Assessment of the effectiveness of security solutions in use



3.5 Findings

The Payatu security team performed real-time Red Team assessment on <client-name> infrastructure (physical & digital). This assessment aims to uncover any security issues in the assessed physical/digital infrastructure, explain the impact and risks associated with the issues found, and provide guidance in the prioritization and remediation steps.

The security assessment revealed 3 high issues, 8 medium, and 1 informational issue. The consolidated summary of the assessment has been presented in the Executive Summary section. Additional information is contained within the Detailed Vulnerability Information section of this report.

4. Executive Summary



4.1 Summary

The Red Team assessment was performed from **<start-date>** to **<end-date>**. After analyzing all the security issues found during the Red Team assessment, we can say that the overall security posture of the organization is **<security-level>** considering the ratings of all the vulnerabilities found, the business risk they possess when combined and our experience with other such engagements.

The below summary and this entire report are based on an outsider's (attacker's) perspective and our observations/findings as we performed different attacks during the Red Team engagement. We did not have any knowledge about other preventive/detective controls that were implemented by the organization.

The detailed version of the below summary can be found below in the 'Detailed Findings and Proof of Findings' and 'Detailed Failed test cases/Strengths' section. The details of the exact actions that were performed during the Red Team assessment and the outcome for each action is described in the storyline section (Annexure A) of the report, along with a proper timeline.

We have used the Common Vulnerability Scoring System (CVSS) standard to rate vulnerabilities that we found during the assessment. A separate excel sheet has been provided along with this report that contains all the CVSS score parameters and the final rating calculated for the vulnerabilities listed in the table of findings.

4.1.1 Summary of Strength

On the positive side, we tried to clone the employee RFID card, but we failed, as the RFID card used by <client-name> cannot be cloned due to encryption. Some of the sites require RFID card validation at entry, and hence the photo ID card we had created did not work. As a result, we could not manage to get inside, even after multiple attempts. In one of the locations, which had RFID validation, we still managed to sneak in an opportune moment without a valid RFID card.

We also tried to physically get inside network/server rooms wherever possible, but all of them were secured.

From our observation, we can say that the Active directory is very securely configured as we tried a variety of attacks against the active directory with a goal to attain 'privileged' access to domain controller/exchange/database/email server, but none of them worked.

We also observed that the patch level of the systems as good as we could not find any high-value targets with a critical vulnerability which could be exploited to gain access to the remote system. LAPS (Local Administrator Password Solution) implementation prevented us from moving across different systems laterally, even after dumping local admin account hashes.

Either the monitoring function did an excellent job in blocking our phishing emails/links early in time, or the employee awareness on phishing attacks were very good, or it was a combination of both factors working together, which drastically reduced our success rate (7%) for the phishing attack.

We enumerated the external subdomains of <domain-name> to identify any critical vulnerabilities, but we didn't find any.

Though we had access to the <client-name> internal network for over two months, we failed to exploit (gain access to) high-value targets such as email servers, domain controllers, exchange servers, etc. which is a commendable fact. However, it should also be noted that we could not touch all the high-value targets as the <client-name> network was too large for us to cover, given the limited time in which the assessment was to be completed.

4.1.2 Summary of Weaknesses

We were able to get inside 2 out of 4 locations defined in the scope by tail-gating/card spoofing and perform social engineering on employees to convince them to handover their unlocked system to us. We also booted Kali Linux on a few other unattended workstations present in the ODC's, impersonating as someone from the 'Infrastructure Services (IS)' team performing a routine audit. We got access to internal networks, few critical web panels, and gained credentials of hundreds of employees for various portals via email and phishing using the Wi-Fi captive portal.

After gaining access to internal networks, we also pulled/extracted a lot of data from active directory, which gave us more ideas about the size of the network, different sites, locations and subnets, important employees responsible for administrative tasks.

We were also able to download and execute our malware bypassing the defensive software installed on the host/network and persist in the network for more than 2 months without being identified/shut down. We collected a lot of users' hashes by tapping into the network, cracked a few of them, but none of the service accounts are in the higher privilege group. We could not crack any of the hashes for the privileged service accounts because the password length made it difficult to crack the passwords within a given time.

We also exfiltrated data gathered from some <client-name> system(s) to our C2 (Command and Control) server. The same was not detected by existing controls.

We were also able to gain complete access to <client-name> internal network by logging into the VPN from outside using the obtained employee credentials as there was no two-factor authentication (2FA) implemented on the VPN. We logged into a few critical internal web panels which were using the default passwords.

None of the activities that we performed after gaining access to <client-name> internal network was flagged for review, which suggests that the team monitoring the network/systems needs to strengthen the log aggregation, log correlation, log analysis techniques and improve on response time.

4.1.3 Conclusions

During Red Team assessments, our final goal is to gain privileged access to at least one high-value target (Domain controller, Email server, Sensitive file server, etc.), which we failed to achieve in this case. However, we had 80+ working credentials of different employees, of which we successfully validated 20 credentials and could access the internal network (via VPN). We were also able to bypass physical security at 2 locations, execute our malware, and persist for more than two months without being detected. This shows a failure of security controls at different stages.

Considering that we could not exploit a high-value target during the period of the assessment but persisted in the network to have more opportunities to exploit high-value targets, we can say that <client-name> stands at a moderate security level.

We primarily suggest implementing these security controls in order to effectively remediate against the majority of the security risks that were found during the assessment.

- Tighten the physical security controls, especially at the campus entrances across all the locations. It is recommended to install metro style doors at all the entry points, which allows only one person to pass at once. In addition to this, it is advisable to have a card validation mechanism at the entry to the campus to prevent the use of dummy cards.
- Strengthen the log aggregation and anomaly detection with an objective to detect and block adversaries and malicious/unusual activities happening inside the internal network.
- Enable two-factor and device authentication on VPN access.

The above summary only holds true for the time we got access to the campus/internal network. New vulnerabilities can get introduced anytime with changes in network/Host-based software.



4.2 Vulnerability Chart

The discovered vulnerabilities table and chart illustrated below provides a snapshot view of the number and severity of issues discovered during this security assessment.

The vulnerabilities are scored/rated based on the CVSS scoring system v3.0. A separate excel sheet has been provided along with the report, which contains all the CVSS scoring parameters for each vulnerability found.

The statements defining the High, Medium, and Low risk are based on the official CVSS specification document.

HIGH

High risk vulnerabilities can have a catastrophic adverse effect on the organization or individuals associated with the organization (e.g., employees, customers).

MEDIUM

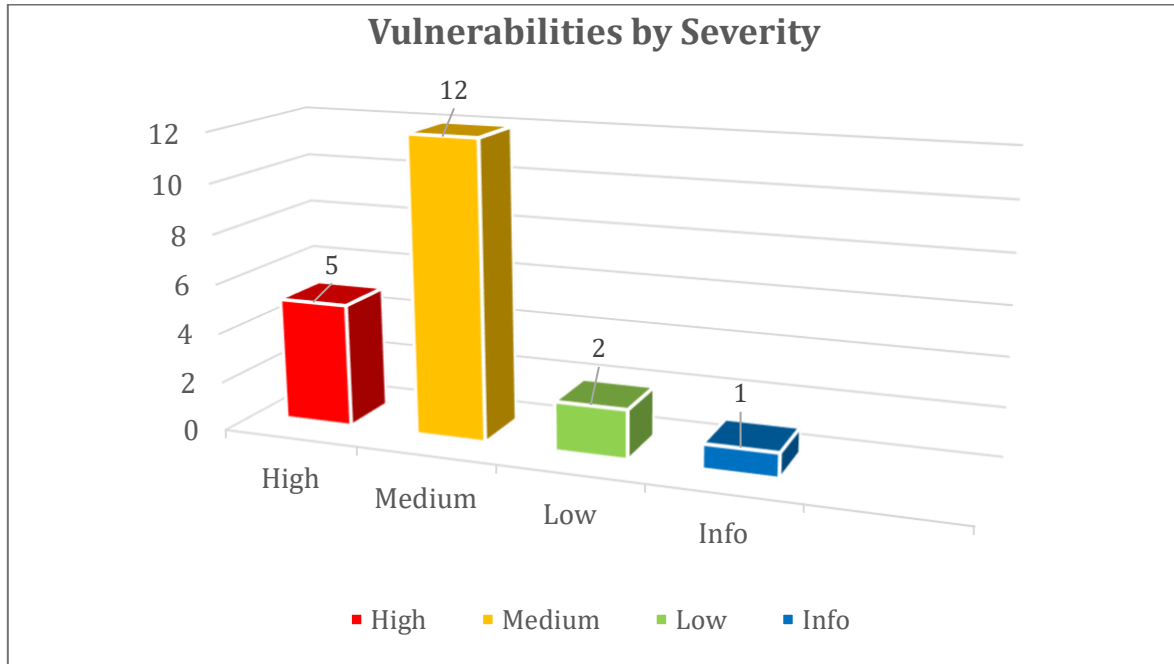
Medium risk vulnerabilities can have a serious adverse effect on the organization or individuals associated with the organization (e.g., employees, customers).

LOW

Low risk vulnerabilities can only have a limited adverse effect on the organization or individuals associated with the organization (e.g., employees, customers).

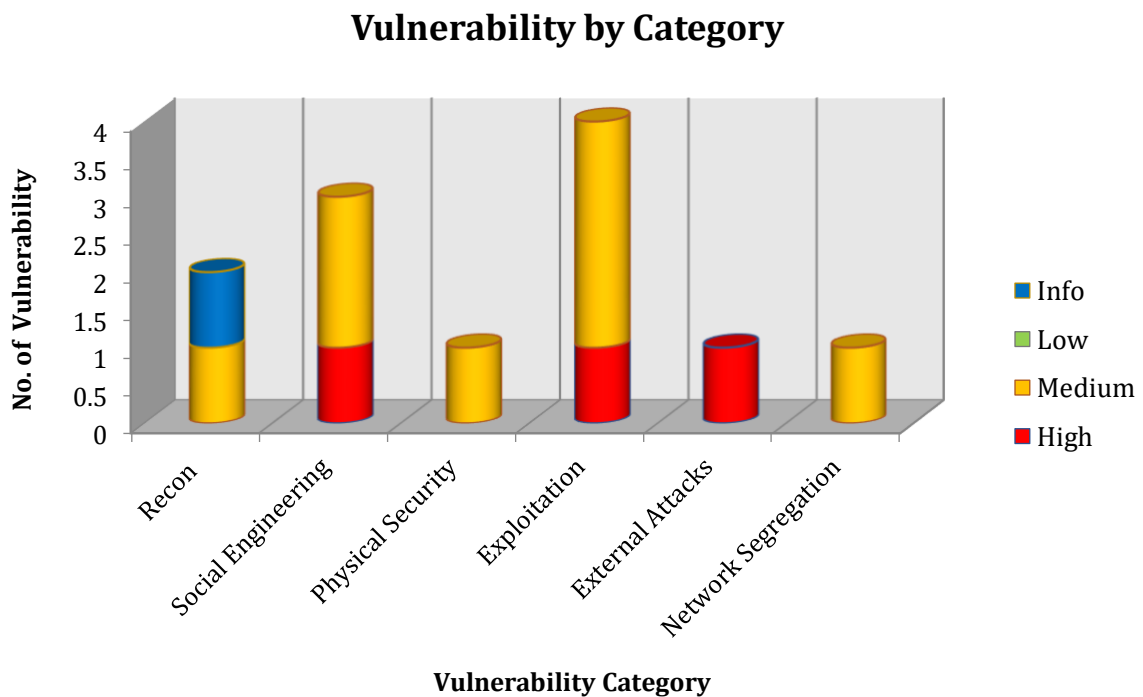
INFO

This is not a security problem but is included as a commentary on security controls examined.



4.3 Findings by Category

Below given chart shows the vulnerability matrix based on the category of vulnerabilities.





4.4 Table of Findings

Ref	Finding	Severity	Status
1	Recon - Enumerated subdomains of <client-name> and performed vulnerability assessment on them	INFO	Not Fixed
2	Recon - Creating a Fake employee ID card is possible	MEDIUM	Not Fixed
3	Physical Security - Got access to the building/ODC via tailgating/ Sneaking in multiple USB devices and a mini laptop inside	MEDIUM	Not Fixed
4	Social Engineering - Gathered Employee credentials via email phishing activity	HIGH	Not Fixed
5	Social Engineering - Gained physical access to machines via social engineering with Employees who were on the premises	MEDIUM	Not Fixed
6	Social Engineering - Gathered Employee credentials via Fake Wi-Fi captive portal	MEDIUM	Not Fixed
7	Exploitation - Malware implant execution and obtaining a reverse beacon (with SYSTEM privilege)	MEDIUM	Not Fixed
8	Exploitation - NBT-NS and LLMNR poisoning to gather NTLMv2 hashes of domain users	MEDIUM	Not Fixed
9	Exploitation -Kerberoasting attack	MEDIUM	Not Fixed
10	Exploitation - Ineffective Access Control to the network	MEDIUM	Not Fixed
11	Web Application Security - Default credentials login (Internally hosted web apps) [we could login to 30% of the found web panels]	HIGH	Not Fixed
12	Network Segregation - No network segregation / Access to critical internal web panels.	MEDIUM	Not Fixed
13	External Attacks - No 2FA for VPN access and no external device authentication (Gained VPN Access to internal network using credentials gathered during phishing activity)	HIGH	Not Fixed



4.5 Table of Strengths/Failed test cases

Ref	Finding	Severity
1	Physical Security – Physical security at the perimeter was very strong at these locations (Location-2, Location-3)	N/A
2	Exploitation: Failed to move laterally across different systems (LAPS Implementation)	N/A
3	Observation: System patch level is good against critical remote vulnerabilities	N/A

SAMPLE

5. Detailed Findings & Proof of Findings

Note: It is advised to read the storyline section first before reading the detailed findings and proof section to get a clear understanding of the whole assessment and the outcome with the time.

5.1 Recon – Enumerated subdomains of <client-name> and performed stealthy scan.

Severity: INFO

Description:

A subdomain is a domain that is part of a larger domain. Example sub1.domain.com has a subdomain sub1 for the domain domain.com. Subdomain enumeration is the process of finding valid sub-domains for one or more domain(s). Sub-domain enumeration can reveal many domains/subdomains in the scope of a security assessment, which in turn increases the chances of finding vulnerabilities. Finding applications running on hidden, forgotten sub-domains may lead to uncovering critical vulnerability. Often, the same vulnerabilities tend to be present across different domains/applications of the same organization. Enumerating subdomains increases the attack surface for an attacker.

Location	Attack Vector Worked
N/A	✓

Possible Impact / Consequence: N/A

Recommendation:

- There is no security issue in the enumerated external domains so far.

Tools Used:

- DNSRecon

References:

- <https://blog.sweepatic.com/art-of-subdomain-enumeration/>

Proof of Vulnerability:

We discovered and enumerated all the subdomains of <client-name> using different techniques. Once, we had the final list, we started looking for vulnerabilities in each of the subdomain using automated and manual approach. We also used Google, Shodan, etc. to find out any unprotected exposed critical web panels, database ports, etc.

We performed minimal vulnerability scanning on the external subdomains with an objective to uncover any high-risk vulnerability at different intervals which could give us privileged access to any web application/host or reveal any sensitive data.

Below are the mentioned vulnerability assessment activities we performed on the Internet facing subdomains of <client-name>

- Tried logging in with obtained employee's credentials (gathered during the assessment) on login pages
- Checked for any outdated version of software components which could be exploited
- Tried Logging in with default credentials on login pages
- Used tools like Wfuzz/Gobuster to brute-force sensitive files and directories
- Used tools like Nikto and Shodan to find vulnerabilities
- Checked for SQL Injection and other various security misconfiguration vulnerabilities

Observation:

It was observed that, the domains on which we found open ports were mostly running web applications. We did try to bypass login panels using SQL Injection, default passwords, checked for various misconfiguration vulnerabilities and forced browsing, but none of the attack vectors worked.

We were also able to login to web-based VPN panels using the obtained employee credentials gathered during the Red Team assessment.

We also looked for any outdated services running on the subdomains which could be exploited. We observed that most of the servers are running latest version of applications.

We were able to login into some web panels using the employee credentials gathered during the Red Team assessment, but we didn't find any sensitive data after logging inside the application.

All the subdomains that we enumerated and performed vulnerability assessment on them with an objective to uncover any high-risk vulnerability is mentioned in the subdomain enumeration excel sheet.

<client-name> subdomain discovery using different tools

We discovered and enumerated all the subdomains of <client-name> using different techniques. Once, we had the final list, we started looking for vulnerabilities in each of the subdomain using automated and manual approach. We also used Google, Shodan, etc. to find out any unprotected exposed critical web panels, database ports, etc.

We performed minimal vulnerability scanning on the external subdomains with an objective to uncover any high-risk vulnerability at different intervals which could give us privileged access to any web application/host or reveal any sensitive data.

Below are the mentioned vulnerability assessment activities we performed on the Internet facing subdomains of <client-name>

- Tried logging in with obtained employee's credentials (gathered during the assessment) on login pages
- Checked for any outdated version of software components which could be exploited
- Tried Logging in with default credentials on login pages
- Used tools like Wfuzz/Gobuster to brute-force sensitive files and directories
- Used tools like Nikto and Shodan to find vulnerabilities
- Checked for SQL Injection and other various security misconfiguration vulnerabilities

Observation

It was observed that, the domains on which we found open ports were mostly running web applications. We did try to bypass login panels using SQL Injection, default passwords, checked for various misconfiguration vulnerabilities and forced browsing, but none of the attack vectors worked.

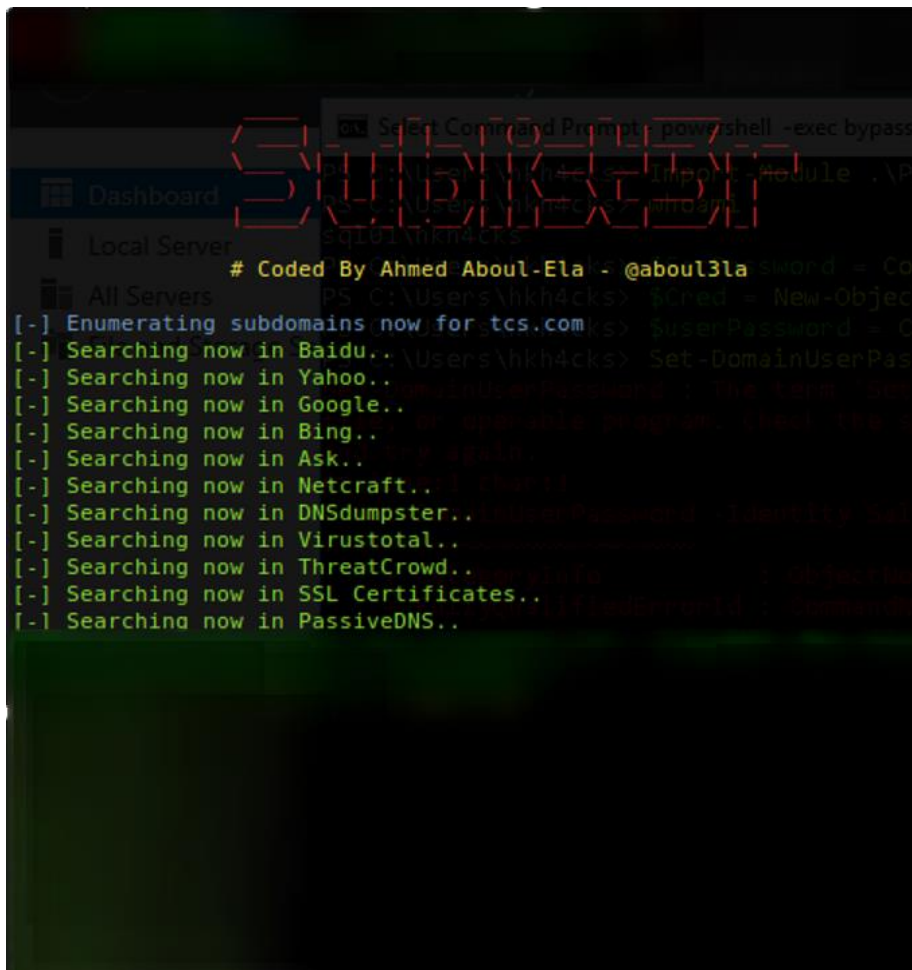
We were also able to login to web-based VPN panels using the obtained employee credentials gathered during the Red Team assessment.

We also looked for any outdated services running on the subdomains which could be exploited. We observed that most of the servers are running latest version of applications.

We were able to login into some web panels using the employee credentials gathered during the Red Team assessment, but we didn't find any sensitive data after logging inside the application.

All the subdomains that we enumerated and performed vulnerability assessment on them with an objective to uncover any high-risk vulnerability is mentioned in the subdomain enumeration excel sheet.

<client-name> subdomain discovery using different tools



```

Select Command Prompt: powershell -exec bypass
PS C:\Users\hkh4cks> Import-Module .\Pc
PS C:\Users\hkh4cks> .\subdomain
SQL01\hkh4cks>

# Coded By Ahmed Aboul-Elakr @aboul3la
PS C:\Users\hkh4cks> $Cred = New-Object
PS C:\Users\hkh4cks> $UserPassword = Co
PS C:\Users\hkh4cks> Set-DomainUserPass

[.] Enumerating subdomains now for tcs.com
[.] Searching now in Baidu.. \Users\hkh4cks>
[.] Searching now in Yahoo.. \Users\hkh4cks>
[.] Searching now in Google.. \Users\hkh4cks>
[.] Searching now in Bing.. \Users\hkh4cks>
[.] Searching now in Ask.. \Users\hkh4cks>
[.] Searching now in Netcraft.. \Users\hkh4cks>
[.] Searching now in DNSdumpster.. \Users\hkh4cks>
[.] Searching now in Virustotal.. \Users\hkh4cks>
[.] Searching now in ThreatCrowd.. \Users\hkh4cks>
[.] Searching now in SSL Certificates.. \Users\hkh4cks>
[.] Searching now in PassiveDNS.. \Users\hkh4cks>
  
```

Port scanning each subdomain at various time intervals and listing out results for manual enumeration

```
Nmap scan report for [REDACTED]
Host is up [REDACTED]
Not shown: [REDACTED] filtered ports
PORT      STATE SERVICE
```

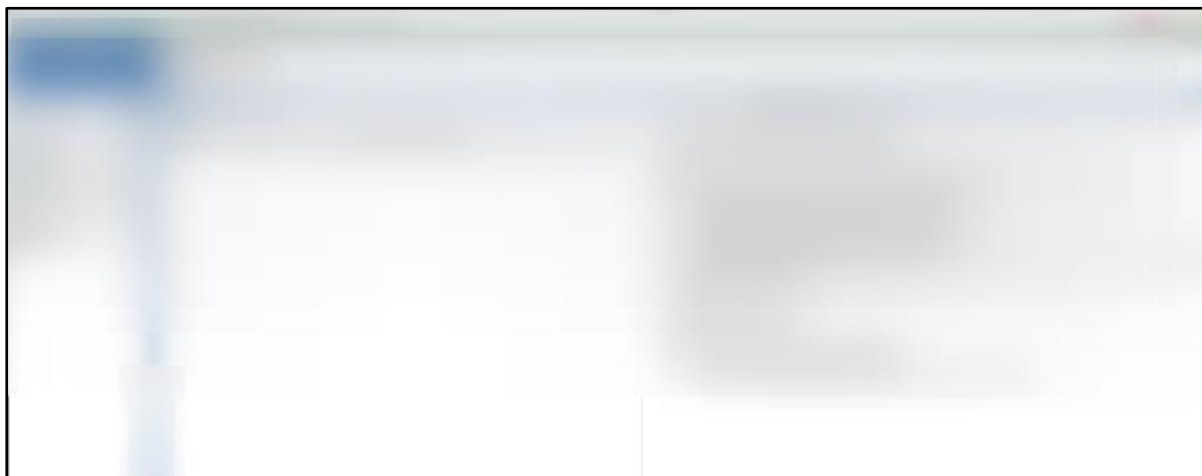
```
Nmap scan report for [REDACTED]
Host is up [REDACTED]
Not shown: [REDACTED] filtered ports
PORT      STATE SERVICE
```

Screenshot of <url> subdomains sheet

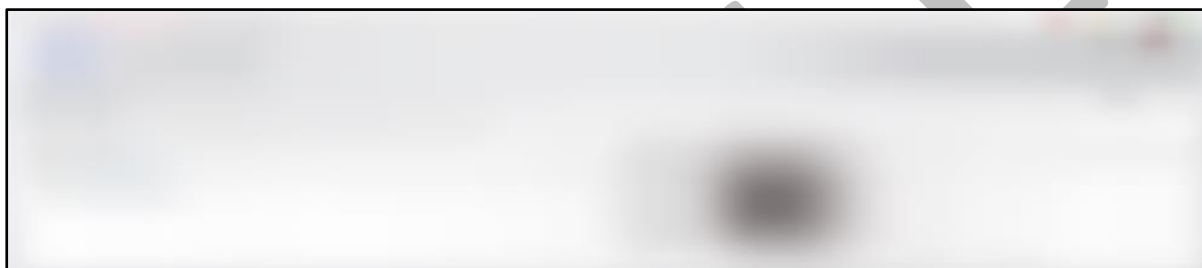
1	Subdomain	IP	Ports Open	Remarks	Vulnerability assessment remarks (Only Found vulnerabilites are listed here)

We were able to login into these web applications using the obtained employee credentials.

<url>

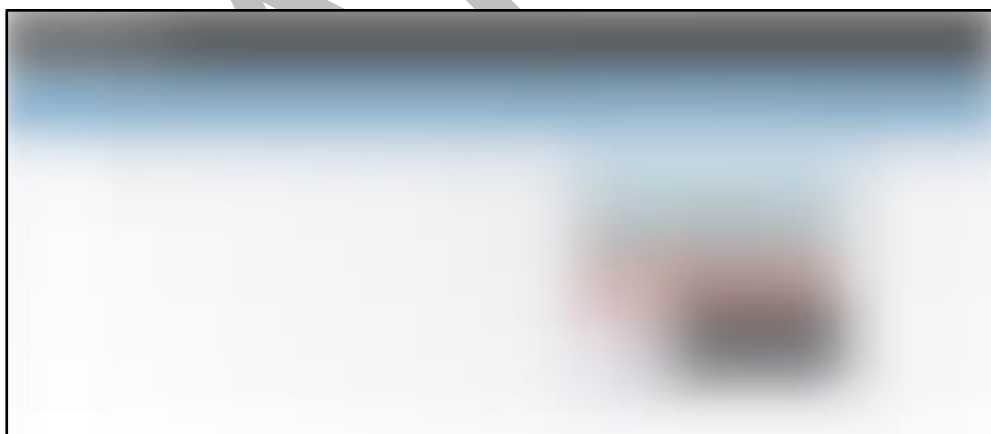


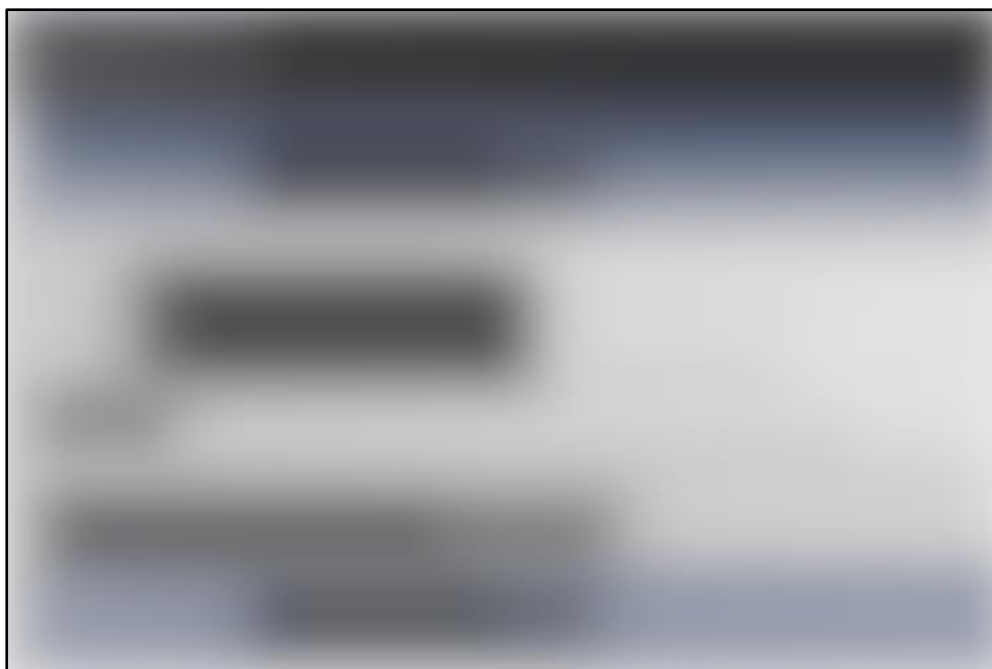
<url>



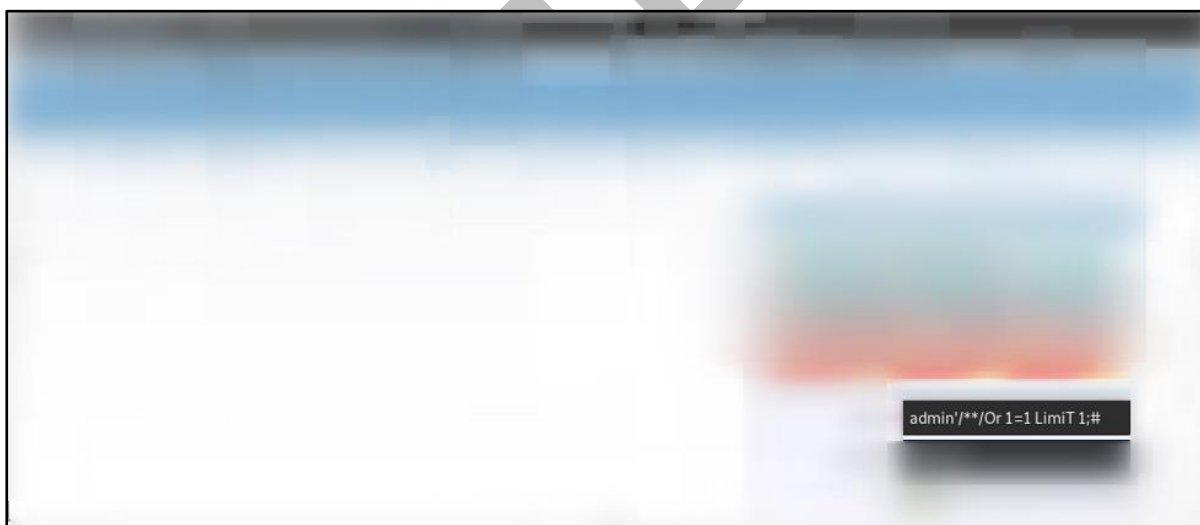
We didn't find any sensitive data or vulnerabilities after logging into these web panels. VPN login has been covered in detail in a separate section (4.13).

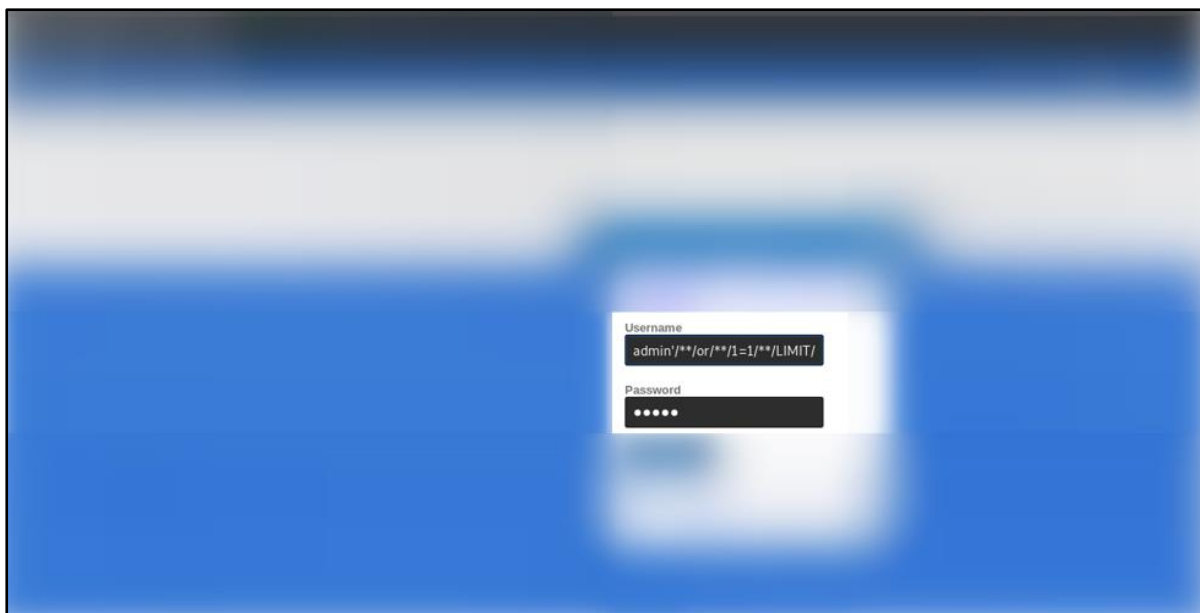
We tried logging into all the login pages with common default passwords but we failed. Below are the few screenshots of failed attempts to login with default credentials on few subdomains.





We also tried to perform SQL Injection on the login pages in order to bypass the login control but we failed. Below are few screenshots of failed attempts to bypass login page using SQL injection attacks.





We found out that <URL> was running an Typo3 CMS. Since it was protected by Web Application Firewall (WAF), we didn't enumerate further, as it may alert the Security Operations Center (SOC) analyst.

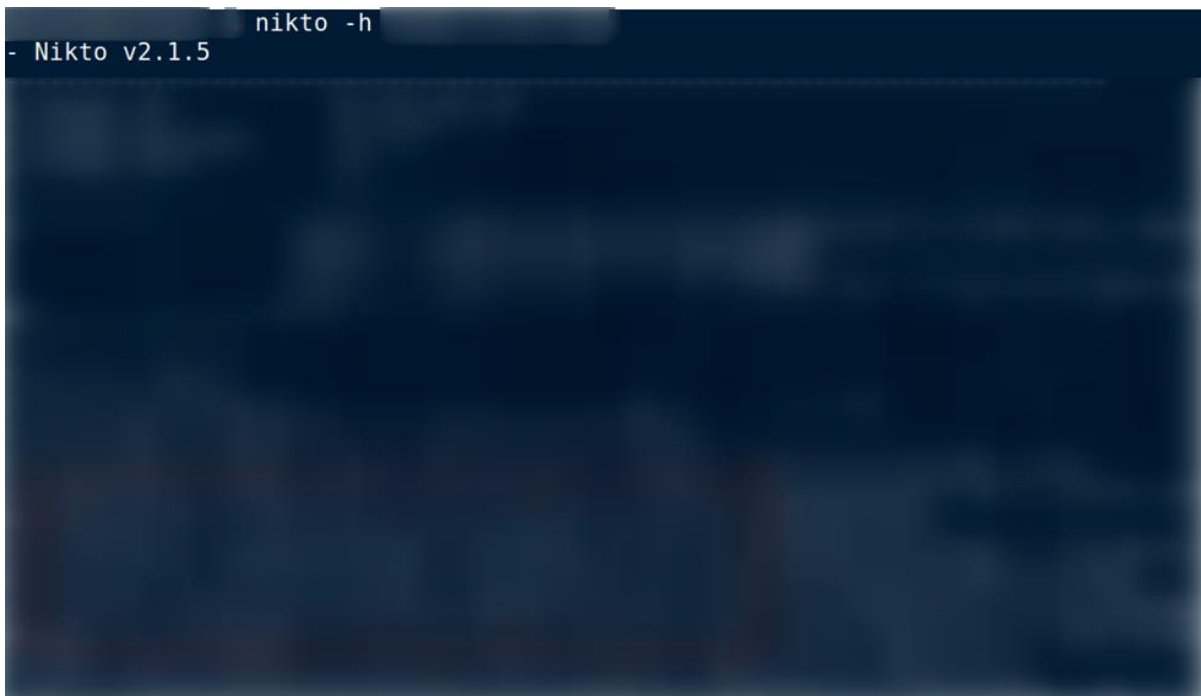


We also ran directory bruteforcing tools on few web panels to uncover any hidden/unprotected web panel or file at random intervals but we failed to discover any. Below is a screenshot of a failed attempt to discover any unprotected file/directory.

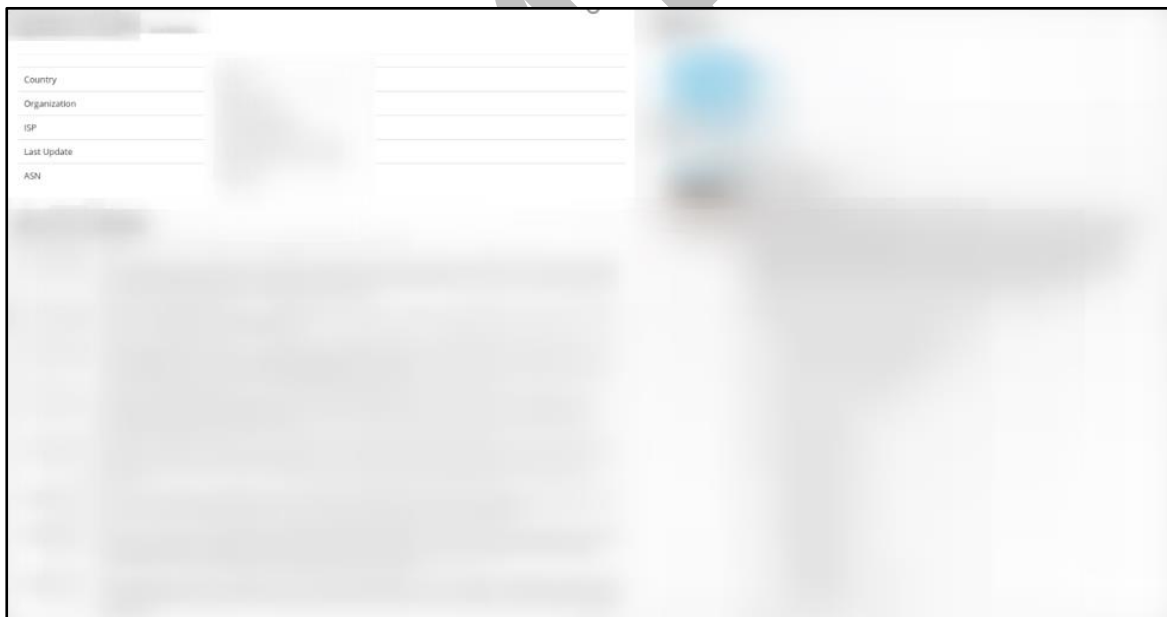

```
-----  
Directories found during testing:  
Dirs found with a 200 response:  
  
Dirs found with a 403 response:
```

We also ran Nikto on most of the web applications in order to discover any high risk vulnerability. We didn't find any such high-risk vulnerability on any of the domains. Below are few screenshots of the output produced by Nikto.

```
nikto -h  
- Nikto v2.1.5
```



Used services like Shodan to find out open web panels, databases, Jenkins servers, etc. exposed to the Internet. We tried to uncover any high-risk vulnerability on the exposed IP's but we didn't find any.



We also manually enumerated each open port to find any critical vulnerability. But we didn't find any vulnerabilities.

```
telnet
Trying 10.10.10.10:
Connected to 10.10.10.10.
Escape character is '^]'.
C

User Access Verification
Username:
```

```
$ssh root@
```

Summary

At the time of writing, we didn't find any high-risk security issues in any of the external subdomain. The Internet facing subdomains of <client-name> are securely configured.

****The spreadsheet which contains data of all the external subdomains that we enumerated would be shared along with the final report.***

5.2 Recon - Creating Fake employee ID card is possible

Potential Impact: MEDIUM

Description:

An identity card (also called a piece of identification or ID) is a document which may be used to prove a person's identity. It is generally issued in a small, standard credit card size form. It is issued to all the staff, vendors and contractors, usually with RFID capabilities. It is the most basic form of visual authentication and can get past any security barricade at perimeter which lacks RFID based authentication.

The layout of the <client-name> ID card can be obtained via simple Open-source intelligence (OSINT) Techniques. OSINT is the process of collecting data from publicly available sources to be used in an intelligence context. Most of the times, a simple Google search is enough to obtain sensitive information like the layout and design of the ID Card.

Location	Attack Vector Worked
N/A	✓

Possible Impact / Consequence:

- An attacker can impersonate as a legitimate <client-name> employee by creating a fake ID-card and use that to enter the campus premises via tail-gating.

Recommendation:

- Request the respective hosting sites to remove all the scanned copies of the <client-name> ID-cards from the Internet
- Spread awareness to the employees not to upload a picture of their ID-card on the Internet/Social media
- Install readers to check the validity of the card issued to an associate at the perimeter level itself

Tools Used:

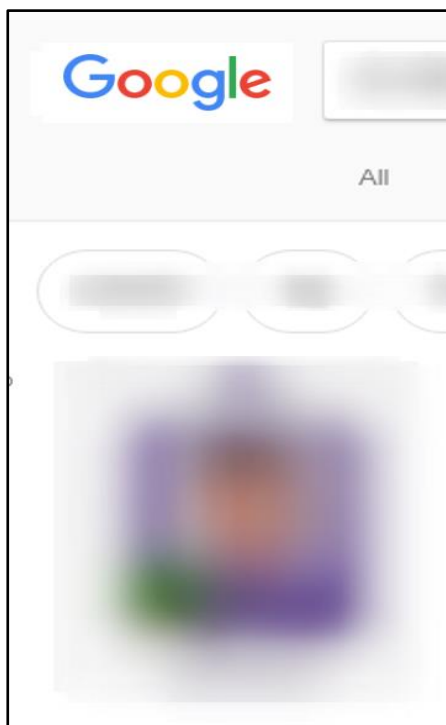
- Manual

References:

- https://en.wikipedia.org/wiki/Open-source_intelligence

Proof of Vulnerability:

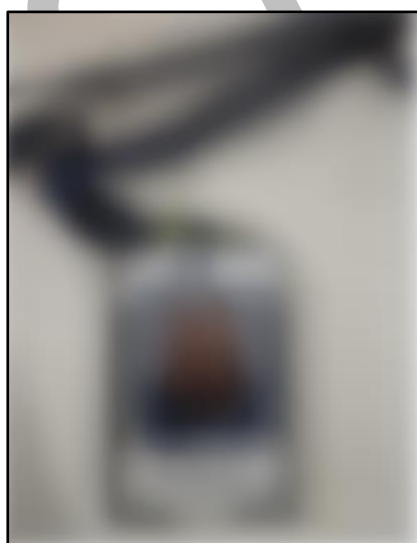
A simple search on the Internet with this keyword "<client-name> ID card" would show up clear images of scanned <client-name> ID card.



URLs where these images are hosted:

<URL's>

Using the images found in the above mentioned. Links, we created a fake <client-name> ID-card and lanyard and were successful in gaining access to 2 out of 4 premises (Location-2, Location-3) using that fake ID card.



5.3 Physical Security - Got Access to the building and ODC via tailgating/Sneaking multiple USB devices and a mini Laptop inside

Note: This attack vector worked because we were able to create a fake employee ID card. Tailgating is dependent on 4.2 (Creating Fake employee ID card is possible) attack vector.

Severity: MEDIUM

Description:

Tailgating, sometimes referred to as piggybacking, is a physical security breach in which an unauthorized person follows an authorized individual to enter a secured premise. Tailgating provides a simple social engineering-based way around many security mechanisms one would think of as secure.

Location	Able to get entry into the campus	Access to the number of ODC(s) / Work areas	USB devices (5 in total) carried inside ODC's
Location-1	✓	✓ (3)	✓
Location-2	✗	NA	NA
Location-3	✗	NA	NA
Location-4	✓	✓ (3)	✓

Possible Impact / Consequence:

- A successful tailgate attempt can help an attacker to get inside sensitive areas which could lead to loss of sensitive data/sensitive documents.
- An attacker can tailgate inside the campus/ODC and can perform multiple attacks on the systems, networks, etc
- An attacker can steal/take pictures of sensitive documents or place a backdoor in the network.

Recommendation:

- Tighten the physical security especially at the entrance across all locations.
- Since it was very difficult for us (Red Team) to tailgate through metro style gate and very easy to tailgate through single glass doors, we would recommend installing metro-style gate everywhere at the perimeter which allows only one person to go through at one time.
- It was impossible for us to get inside campuses where there was a RFID enabled metro-style entry gate (Location-3).

Tools Used:

- Manually wearing a fake <client-name> ID-card

References:

- https://en.wikipedia.org/wiki/Physical_security

Proof of Vulnerability:

We were successful in getting inside 2 locations (buildings as well as ODC) out of 4 from the list of the locations mentioned in the table above via tail-gating and wearing a fake <client-name> ID-card printed on a non-RFID piece of plastic.

This is our overall observation about the Physical security on all the locations:

- Location-1 – **Moderate**
- Location-2 – **Strong**
- Location-3 – **Very Strong**
- Location-4 – **Weak**

The exact details as how we got inside each location and what activities we performed is described in detail in the storyline section of this report. In the above section, we have mentioned 4 different levels of physical security, each of which is explained here:

Weak: No RFID check at the campus/building entry point (Very easy to get inside with a fake ID-card)

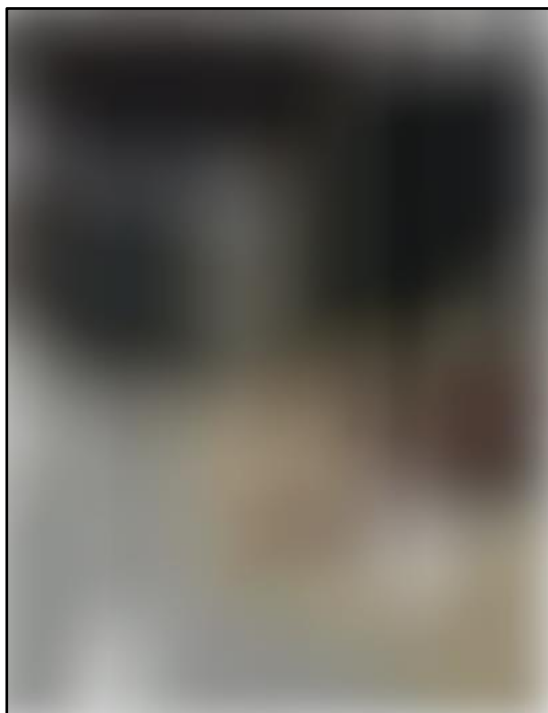
Moderate: Glass door at the campus/building entry point (moderately easy to get inside via tail-gating)

Strong: Metro style RFID enabled doors at the campus/building entry point (Tough to get inside via tail-gating. Guards were vigilant against any tail-gating attempt. Bypassed via alternate doors where metro style RFID doors were not in place/situational circumstances)

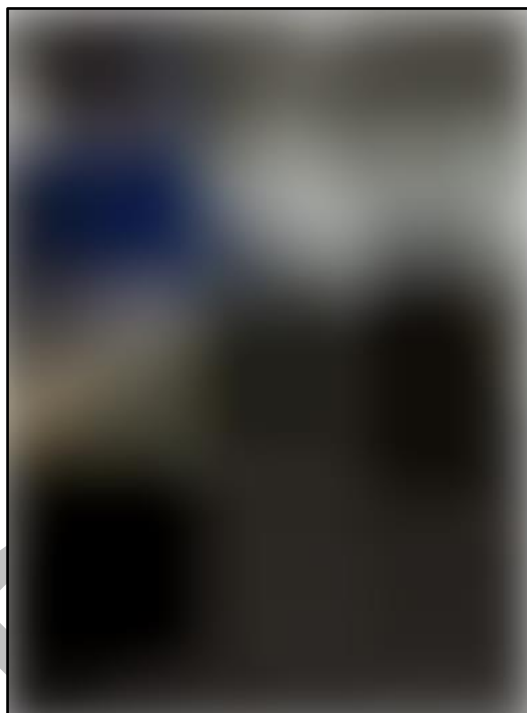
Very Strong: Metro style RFID enabled doors at the campus/building entry point and no other entry point. Guards were very vigilant against any tail-gating attempt. (we could not get inside even after multiple attempts)

Below are the few pictures we took as proof after getting inside:

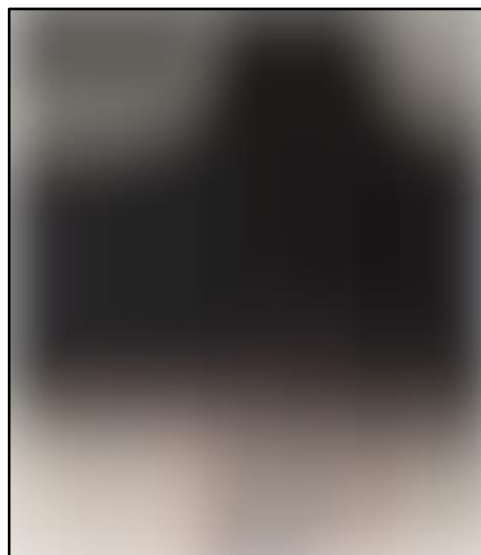
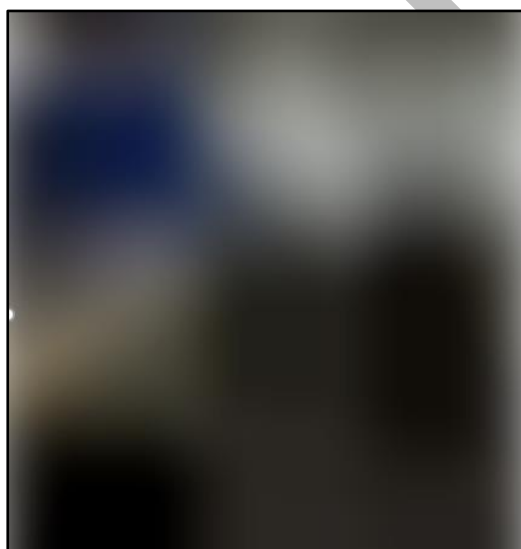
Location-1



Location-2



We also sneaked in multiple USB drives loaded with different tools inside the ODC's which were used to launch different attacks using an empty workstation.



5.4 Social Engineering - Gathered Employee credentials via email phishing activity

Potential Impact: HIGH

Description:

Social engineering, in the context of information security, refers to psychological manipulation of people into performing actions or divulging confidential information. This differs from social engineering within the social sciences, which does not contain the divulging of confidential information.

Phishing is a type of social engineering attack often used to steal user data, including login credentials and credit card numbers. It occurs when an attacker, masquerading as a trusted entity, dupes a victim into opening an email, instant message, or text message. The recipient is then tricked into clicking a malicious link, which can lead to the installation of malware, the freezing of the system as part of a ransomware attack or the revealing of sensitive information.

Location	Attack Vector Worked
N/A	✓

Possible Impact / Consequence:

- An attacker can perform phishing remotely on employees using a fake website disguising as a legitimate <Client-name> login page which could lead to email/VPN/domain/internal web panels credentials leakage and sensitive data loss.

Recommendation:

- Spread awareness among employees to not enter their credentials on any suspicious phishing emails.
- Monitoring team should be more alert when any such mass phishing emails are received and should immediately block the phishing domain and inform the employees.

Tools Used:

- Email

References:

- <https://www.owasp.org/index.php/Phishing>
-

Proof of Vulnerability:

We obtained a few valid Outlook Web Access (OWA) credentials using Wi-Fi captive portal phishing at <Location-4>. Using those credentials, we analyzed the Outlook Web Access API endpoints. Using the Mailsniper script, we collected data of 15,000 employee email address Outlook web application.

Email Phishing statistics (This is an approximate value)

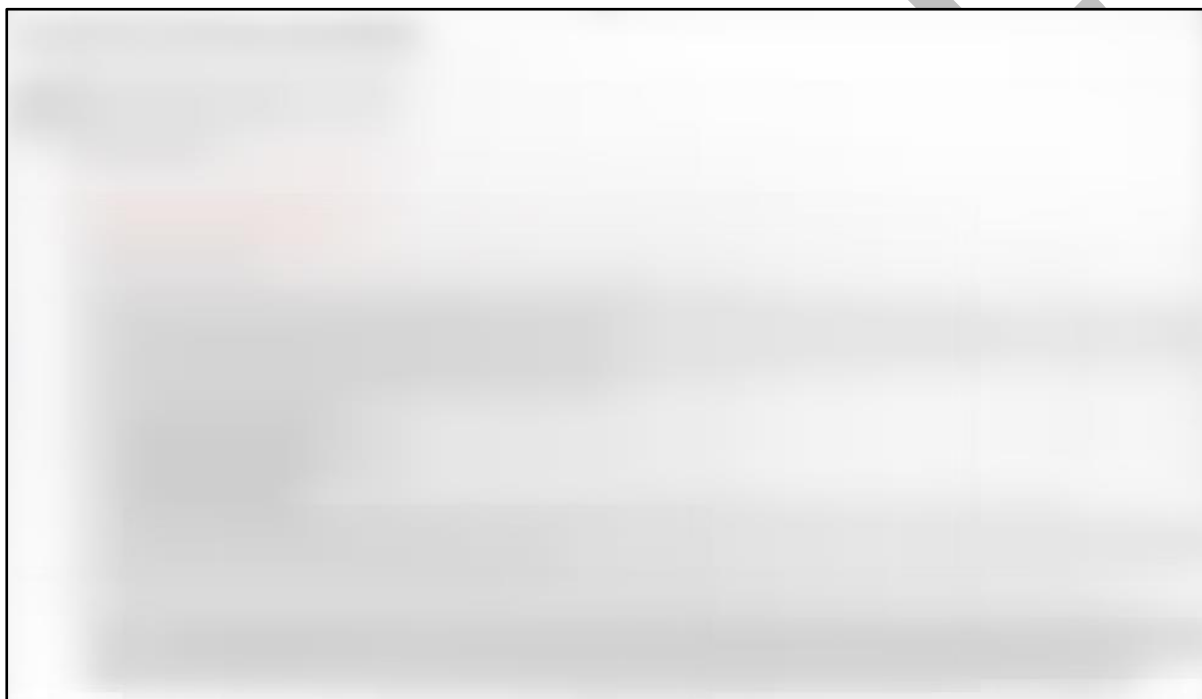
Phishing Email sent: 5,400

Link clicked: 700+

Data submitted: 450+

Credentials verified to be working: 35+

Screenshot of a Phishing email sent to an employee



Successfully logged in to <url> using one of the obtained credentials



5.5 Social Engineering – Gained physical access to machines via social engineering with Employees who were on the premises

Note: We could only attempt this attack vector in those locations where we were able to get inside the campus and then inside the ODC(s) via tail-gating. It did not work for those locations where we could not even enter the campus.

Severity: MEDIUM

Description:

Social engineering, in the context of information security, refers to psychological manipulation of people into performing actions or divulging confidential information. This differs from social engineering within the social science, which does not contain the divulging of confidential information.

On Site Phishing refers to the act of social engineering the company staff at the physical office location. This usually includes impersonation that is pretending or pretexting to be another person with the goal of gaining access physically to a system or building.

Location	Able to access ODC(s)	Attack vector attempted	Attack vector worked
Location-1	✓	✓	✓
Location-2	✗	NA	NA
Location-3	✗	NA	NA
Location-4	✓	✓	✓

Possible Impact / Consequence:

- An attacker who is already inside the premises can perform social engineering attack on employees impersonating someone from 'IS/IT' team, which could lead to credentials leak, malware implant, sensitive file transfer, etc.

Recommendation:

- Spread awareness among employees to not handover 'unlocked' systems to any unknown person impersonating as someone from IT/IS team.
- Spread awareness among employees to not enter their credentials on any popup/webpage when told to do so by an unknown person.

Tools Used:

- N/A

Proof of Vulnerability:

It was very easy for us (Red Team) to engage employees into conversation impersonating as an employee from 'IS/IT' team and convince them to handover their systems to us or let us download and execute our custom malware or scripts which could gather sensitive data or ask for their domain credentials.

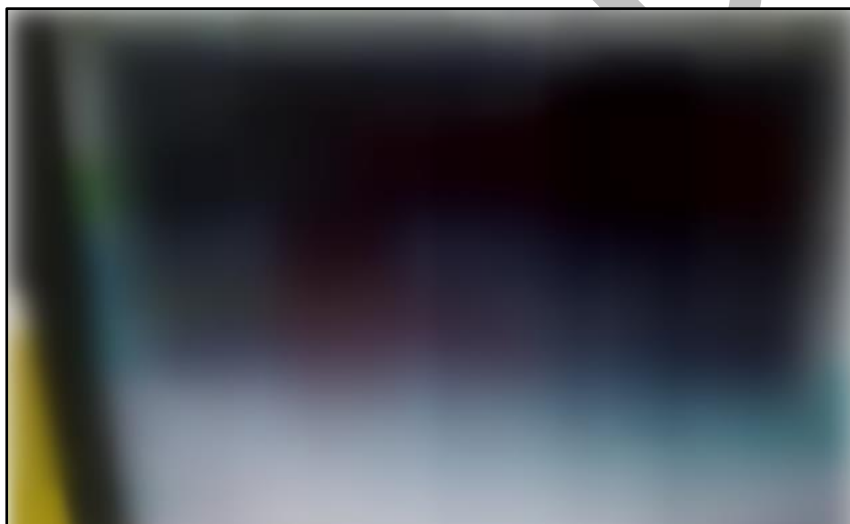
The below table covers the number of people we interacted with and what actions we performed after getting inside the ODC(s) at different locations:

Location	No. of people interacted with	Successful attempts (handed over their systems/helped to sit on an empty workstation)	Actions performed
Location-1	5	3	System/Network enumeration using PowerShell scripts, credentials gathering, Empty workstation handover, Data exfiltration (Enumeration output), Malware Execution, Data exfiltration (Enumeration output).
Location-2	0	0	N/A
Location-3	0	0	N/A
Location-4	12	9	System/Network enumeration using PowerShell scripts, credentials gathering, Empty workstation handover, Data exfiltration (Enumeration output), Malware Execution, Data exfiltration (Enumeration output).

For more details, as what actions we performed in each location and why certain actions failed, please refer to the storyline section of this report.



We could easily convince employees to download and execute multiple PowerShell scripts in order to perform enumeration, gather domain/email credentials and then successfully exfiltrate the data to our remote C2 (command and control) server.



The data we exfiltrated contained enumeration output (network details, system details, updates installed, Active directory enumeration for a domain: Users, Computers, Subnets, Sites, Domain controllers, Groups, emails, last password change, etc.)

We gathered and exfiltrated data from 2 <domain-name> domain in Location-1 and <Location-4> during the whole assessment. We avoided touching any non-<client-name> domain/Active directory during the whole assessment as that does not directly come under security policy as per our knowledge.

5.6 Social Engineering – Gathered Employee credentials via Fake Wi-Fi captive portal

Note: We could only attempt this attack vector in those locations where we were able to get inside the campus and the laptop bags were not scanned using X-ray machine at the perimeter (We carried our portable Wi-Fi captive portal phishing setup along with the battery pack inside the laptop bag). It did not work for those locations where we could not even enter the campus.

Severity: MEDIUM

Description:

Social engineering, in the context of information security, refers to psychological manipulation of people into performing actions or divulging confidential information. This differs from social engineering within the social science, which does not contain the divulging of confidential information.

A Fake Wi-Fi Captive portal spawns a new SSID with a name similar or same as the actual company SSID with the intent of making users connect to it and enter their credentials on the captive page.

Location	Able to enter campus	No X-Ray scanner at the entry-gate(s) of the campus	Attack worked
Location-1	✓	✓	✓
Location-2	✓	NA	NA
Location-3	✓	NA	NA
Location-4	✓	✓	✓

Possible Impact / Consequence:

- An attacker who is already inside the premises can perform social engineering attack on employees using a fake captive Wi-Fi portal which could lead to credentials leakage and sensitive data loss.

Recommendation:

- Install X-Ray scanners at the campus entry points where it is not implemented.
- Increase the physical security on the premises so that an attacker cannot get inside the premises without having the valid RFID enabled ID-card.
- Spread awareness amongst employees to not enter their credentials on any suspicious Wi-Fi access point.

Tools Used:

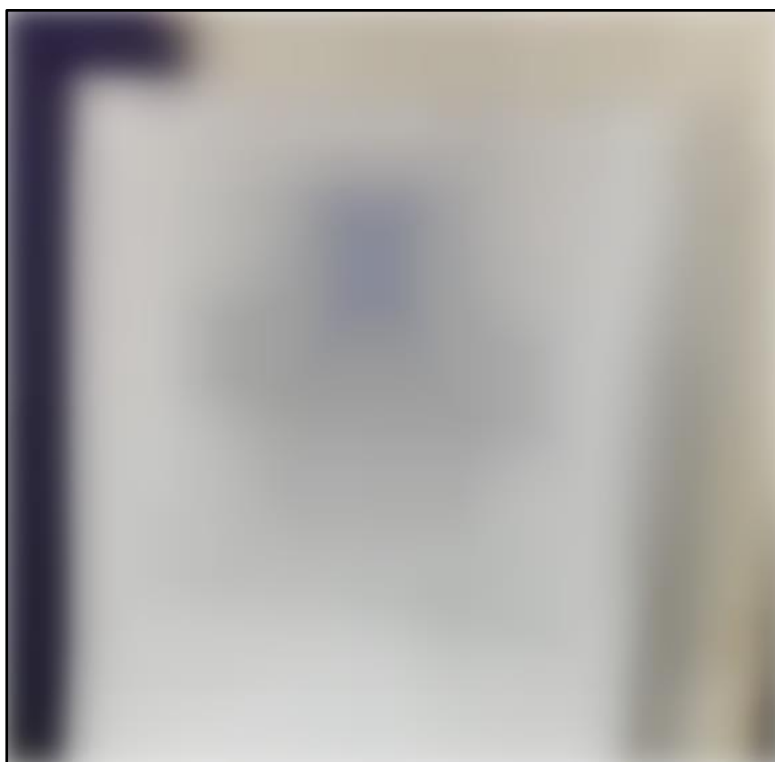
- Raspberry pi
 - Alfa wireless adapter
-

References:

- <https://www.contextis.com/en/blog/wireless-phishing-with-captive-portals>
-

Proof of Vulnerability:

After successfully getting inside the premises with our Wi-Fi captive portal hardware setup hidden in our laptop bag, we placed few fake printed notices in the canteen which said 'Free Wi-Fi for <client-name> employees.'



Upon connecting to this Wi-Fi access point, the employees were presented with a fake cloned webpage of <client-name>, which logged all the entries made to it.

Picture of fake Wi-Fi captive portal after connecting to it.



We captured 8 credentials in <Location-4> out of which 3 worked. We got access to VPN portal and email portal as they were using the same password at multiple places.

5.7 Exploitation – Malware implant execution and obtaining a reverse beacon (privileged access)

Note: We could only attempt executing malware where we were able to get access to a system after getting inside the campus and the ODC(s) via tail-gating. It did not work for those locations where we could not even enter the campus.

Severity: **HIGH**

Description:

A stealth malware is a hidden computer malware that attacks operating system processes and averts typical anti-virus or anti-malware scans. Stealth malware hide in files, partitions, and boot sectors and are adept at deliberately avoiding detection.

Location	Able to access campus & ODC(s)	Access to one / multiple systems	Malware execution	Got Reverse beacon	Maintained persistence
Location-1	✓	✓	✓	✓	✓ (3 Months)
Location-2	✗	NA	NA	NA	NA
Location-3	✗	NA	NA	NA	NA
Location-4	✓	✓	✓	✓	✓ (2 Months)

Possible Impact / Consequence:

- Malware execution could lead to sensitive data loss and credentials compromise.
- A ransomware attack can lead to a huge business impact and loss of important data.

Recommendation:

- Strengthen the central log aggregation and anomaly detection team (SOC) in order to detect such adversaries which normally bypasses Endpoint protection software.

Tools Used:

- Konboot
- Custom powershell malware

References:

- <https://en.wikipedia.org/wiki/Malware>

Proof of Vulnerability:

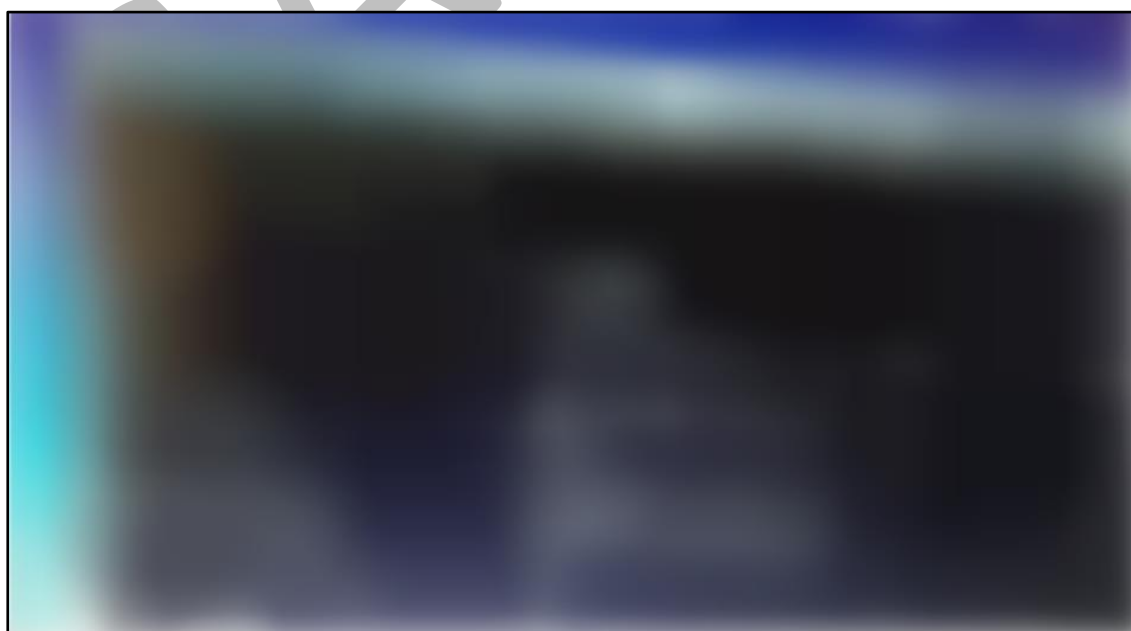
We escalated our privileges using konboot USB utility and then executed our custom non-destructive and non-propagative malware on few systems during the engagement and we successfully bypassed all relevant controls to get multiple reverse beacons every day over a period of 3 months.

We were able to remotely execute commands on multiple systems and enumerate the network further using the reverse beacon.

We were able to persist in the network with our stealth malware for more than 2 months at Location-1 and Location-4.

Privilege escalation using Konboot utility loaded in USB drive

Administrator prompt



Malware Execution and obtaining reverse beacon (with admin privilege)

Multiple active reverse beacons (with SYSTEM privilege)



Setting persistence remotely



Executing commands/attacks remotely on the infected host



5.8 Exploitation - NBT-NS and LLMNR poisoning to gather NTLMv2 hashes of domain users

Note: We could attempt to gather NTLMv2 hashes only where we were able to get access to a system after getting inside the campus and the ODC(s) via tail-gating. It did not work for those locations where we could not even enter the campus.

Severity: MEDIUM

Description:

LLMNR and NetBIOS are two name resolution services built in to Windows to help systems find address names from other devices on the network. However, addresses and address providers on the network are not verified, since Windows assumes that anyone on the network is automatically trusted. When a DNS request fails, Windows will attempt to ask other devices on the network to resolve that address over LLMNR or NBT-NS. For a service like SMB, if a host is configured to automatically authenticate over SMB then by spoofing addresses over LLMNR/NBT-NS, an attacker can easily grab hashes (Net-NTLMv2) by simply passively replying to every single LLMNR/NBT-NS request.

Location	Able to access campus & ODC(s) along with USB devices	Attack vector executed	Collected hashes
Location-1	✓	✓	✓
Location-2	✗	NA	NA
Location-3	✗	NA	NA
Location-4	✓	✓	✓

Possible Impact / Consequence:

- An attacker can perform LLMNR/NBT-NS and gather hundreds of hashes that could be cracked later. It would lead to domain credentials compromise.

Recommendation:

- Disable LLMNR and NetBIOS in local computer security settings or by group policy if they are not needed within an environment.
- Use host-based security software to block LLMNR/NetBIOS traffic.

Tools Used:

- Responder
- Nmap
- Ntlmrelayx

References:

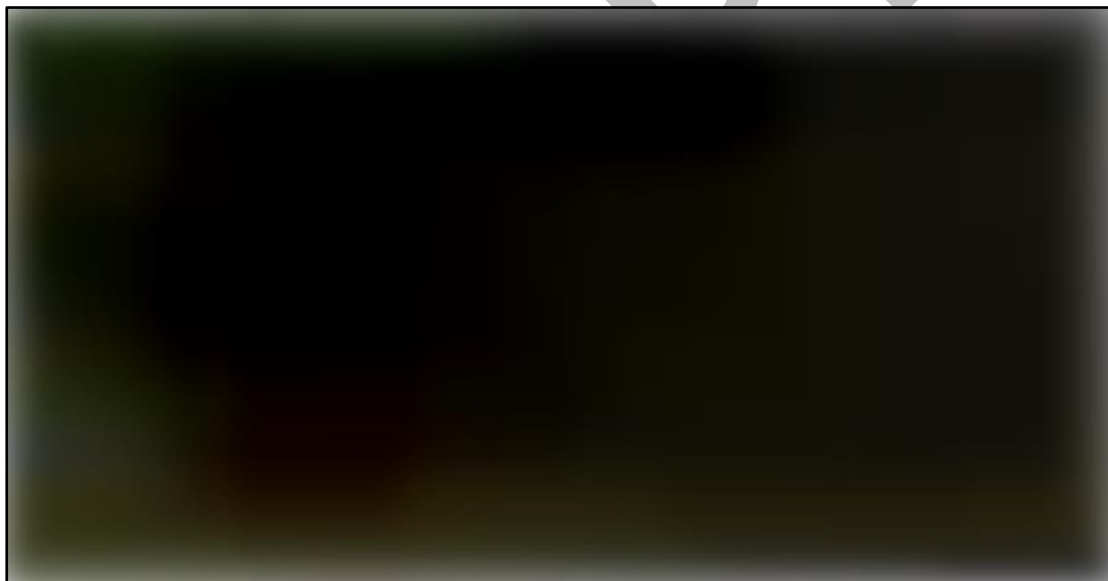
- <https://attack.mitre.org/techniques/T1171/>
-

Proof of Vulnerability:

We booted kali Linux using the USB drive on one of the empty hosts, ran responder and collected multiple Net-NTLMv2 hashes (<client-name> as well as Non-<client-name> domain) from multiple locations.



We gathered around 100+ Non-<client-name> domain hashes and 20+ <client-name> domain hashes out of which we were able to crack around 10 hashes.



5.9 Exploitation – Kerberoasting attack

Note: This attack was performed remotely through the malware beacon, and we could only attempt to perform this attack where we were able to execute the malware on the system after getting inside the campus and the ODC(s) via tail-gating. Since it's an attack against the domain controller, we did not attempt it from multiple locations, as the results would be the same.

Severity: MEDIUM

Description:

The process of cracking Kerberos service tickets and rewriting them in order to gain access to the targeted service is called Kerberoast. This is a very common attack in Red Team engagements since it doesn't require any interaction with the service as legitimate active directory access can be used to request and export the service ticket, which can be cracked offline in order to retrieve the plain-text password of the service. This is because service tickets are encrypted with the hash (NTLM) of the service account, so any domain user can dump hashes from services without the need to get a shell into the system that is running the service.

We performed a Kerberoasting attack after logging in with the <domain-name> domain credentials on the system.

Location	Able to access campus & ODC(s)	Malware executed	Attack vector performed
Location-1	✓	✓	✓
Location-2	✗	NA	NA
Location-3	✗	NA	NA
Location-4	✓	✓	✓

Possible Impact / Consequence:

- An attacker could gain access to user accounts with higher privilege (Domain Admin, Enterprise Admin, etc)
- An attacker could get privileged access to 'Domain Controller'.

Recommendation:

- The best mitigation for this attack is to ensure your service accounts that use Kerberos with SPN values leverage long and complex passwords.
- To detect the attack in progress, monitor for abnormal account usage. Service accounts traditionally should be used from the same systems in the same ways, so it is possible to detect authentication anomalies. Also, you can monitor for service ticket requests in Active Directory to look for spikes in those requests.

Tools Used:

- Powershell Empire kerberoasting module
-

References:

- <https://blog.stealthbits.com/extracting-service-account-passwords-with-kerberoasting/>
-

Proof of Vulnerability:

We performed kerberoasting on <domain-name> domain remotely through our malware beacon. The malware was running under the context (privilege) of a <domain-name> domain user which allowed us to interact with the domain controller. We got service tickets for multiple service accounts and we were able to crack few service accounts and got plaintext passwords.

We could not crack the service ticket for the service users who were in the 'Enterprise Admin' group because the password was too complicated to be cracked.

None of the service users we got the plaintext password for, were in the 'enterprise admin' or any other higher privileged group.

We cracked the obtained hashes and got plaintext password for 11 service accounts but none of them had any "privileged access", therefore those service accounts were not of much use to us.

We had a separate hash cracking machine on the cloud which we used to crack the obtained hashes. Our hash cracking machine had very powerful specifications (64 GB GPU, 128 CPU cores, 8 different powerful GPU's and 488 GB RAM) which allowed us to use different hash cracking techniques with a good hash cracking rate.

We got a total of 76 service account hashes, out of which we could crack 11. The ones which were cracked were easy to crack because of low password complexity. We used a mixture of Dictionary and Rule-based attacks to crack the hashes. Our hash cracking machine ran for 14 hours before we stopped the hash cracking process. During those 14 hours, we cracked 11 of the service account hashes.

Other service accounts hashes



SAM

5.10 Exploitation - Ineffective Access Control to the network

Note: This attack was performed remotely through the malware beacon, and we could only attempt to perform this attack where we were able to execute the malware on the system after getting inside the campus and the ODC(s) via tail-gating. Since it's an attack against the domain controller, we did not attempt it from multiple locations, as the results would be the same.

Severity: MEDIUM

Description:

The process of cracking Kerberos service tickets and rewriting them in order to gain access to the targeted service is called Kerberoast. This is a very common attack in Red Team engagements since it doesn't require any interaction with the service as legitimate active directory access can be used to request and export the service ticket, which can be cracked offline in order to retrieve the plain-text password of the service. This is because service tickets are encrypted with the hash (NTLM) of the service account, so any domain user can dump hashes from services without the need to get a shell into the system that is running the service.

We performed a Kerberoasting attack after logging in with the <domain-name> domain credentials on the system.

Location	Able to access campus & ODC(s)	Malware executed	Attack vector performed
Location-1	✓	✓	✓
Location-2	✗	NA	NA
Location-3	✗	NA	NA
Location-4	✓	✓	✓

Possible Impact / Consequence:

- An attacker could gain access to user accounts with higher privilege (Domain Admin, Enterprise Admin, etc)
- An attacker could get privileged access to 'Domain Controller'.

Recommendation:

- The best mitigation for this attack is to ensure your service accounts that use Kerberos with SPN values leverage long and complex passwords.
- To detect the attack in progress, monitor for abnormal account usage. Service accounts traditionally should be used from the same systems in the same ways, so it is possible to detect authentication anomalies. Also, you can monitor for service ticket requests in Active Directory to look for spikes in those requests.

Tools Used:

- Manual
-

References:

- https://en.wikipedia.org/wiki/Network_Access_Control
-

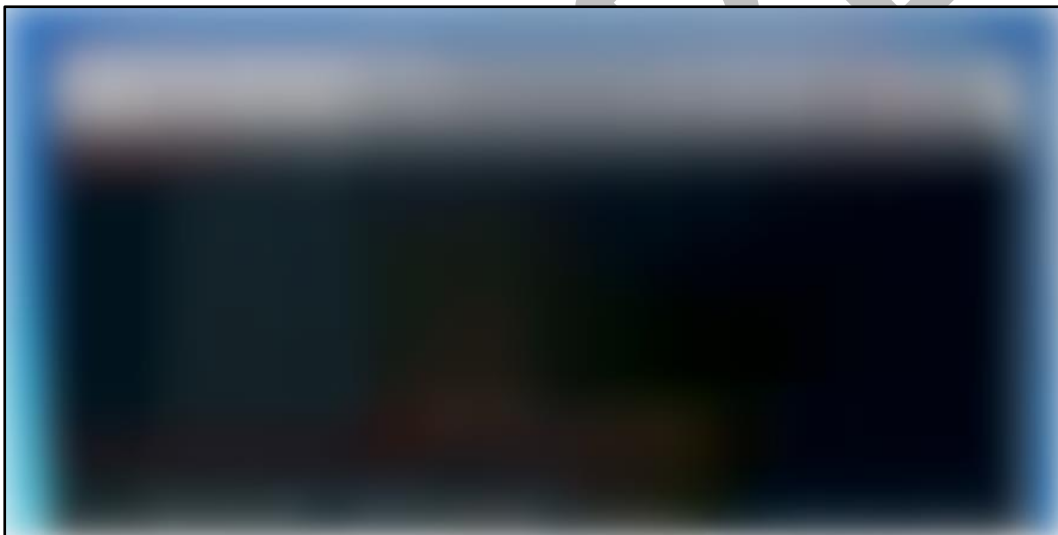
Proof of Vulnerability:

We found out that even when Cisco NAC agent is installed, we got the IP address from DHCP.

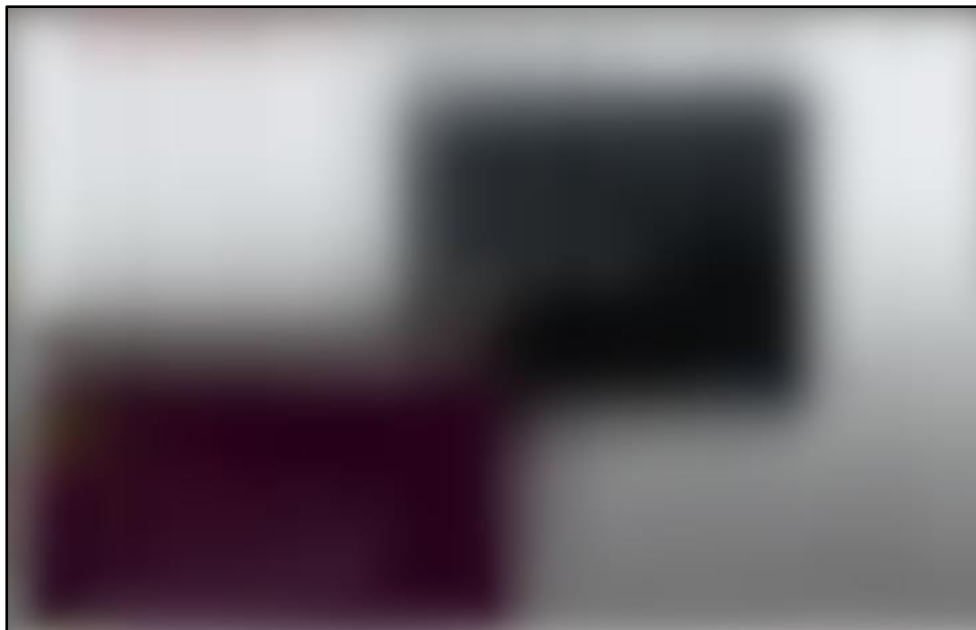
According to our observation, either the NAC server was not configured correctly, or NAC was not enabled for that specific network port.

We got connected to the network when we booted kali Linux on the same host. Ideally, we should not have got IP address from DHCP unless the required conditions are met.

We got connected to the network and got the IP address in the kali OS inside the network.



We also connected a rogue access point to the <client-name> Ethernet cable and started an access point which could give an attacker access to the <client-name> internal network if the attacker is within the range of that Wi-Fi access point. This attack vector was performed in <Location-4>. We were able to connect to that access point and access the internal network of <client-name> as shown in the screenshot below. This shows a failure of the access control mechanism.



In the above screenshot, we can see that we are connected to our rogue Wi-Fi access point '<SSID>' and we can access a web application hosted in the internal network of <client-name>.

Nmap scan on the Domain controllers using Nmap tool.



5.11 Web Application Security - Default credentials login (Internally hosted web apps)

Note: To access/login to any internally hosted web application, the attacker first needs to login to the VPN and gain access to the internal network.

Potential Impact: **HIGH**

Description:

Most web applications are protected behind authentication to delegate access. Some web applications initialize with a known username and password combination. However, it is a best practice to change the password at the first login, otherwise anyone can access the application with the default credentials.

Location	Attack Vector Worked
N/A	✓

Possible Impact / Consequence:

- Anyone in the network can access the critical web panels and make any changes.
- It could lead to compromising several internal infrastructures.

Recommendation:

- Change the default passwords to a secure password.

Tools Used:

- Manual

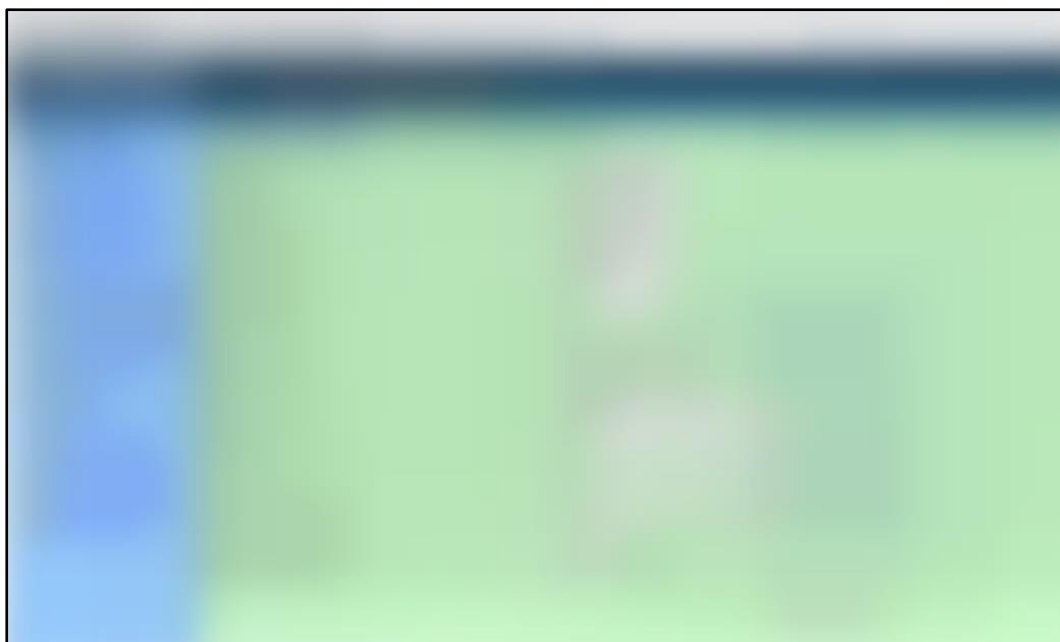
References:

- [https://www.owasp.org/index.php/Testing for default credentials \(OTG-AUTHN-002\)](https://www.owasp.org/index.php/Testing_for_default_credentials_(OTG-AUTHN-002))

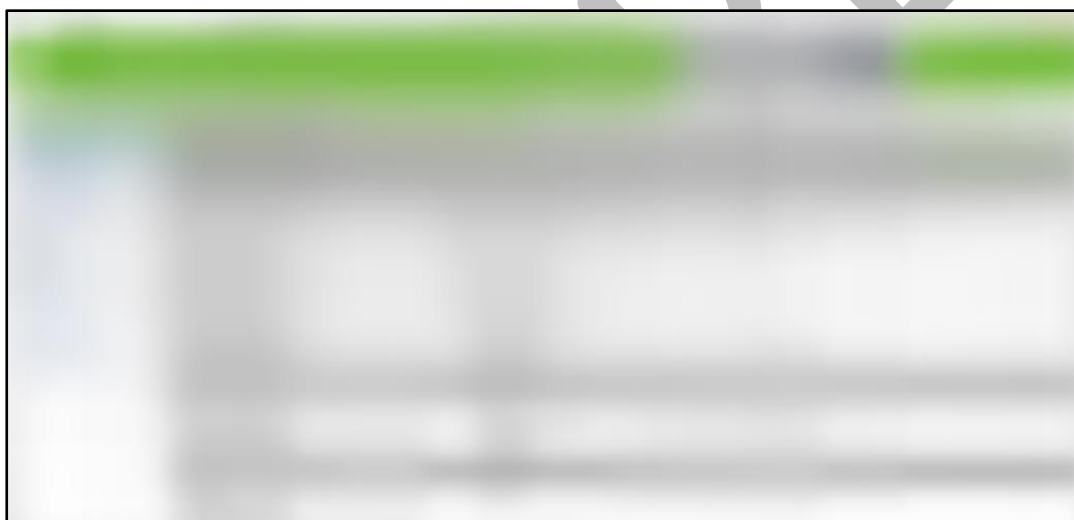
Proof of Vulnerability:

Several web panels were found and accessed with their default credentials. This can lead to sensitive data loss or an entire infrastructure compromise.

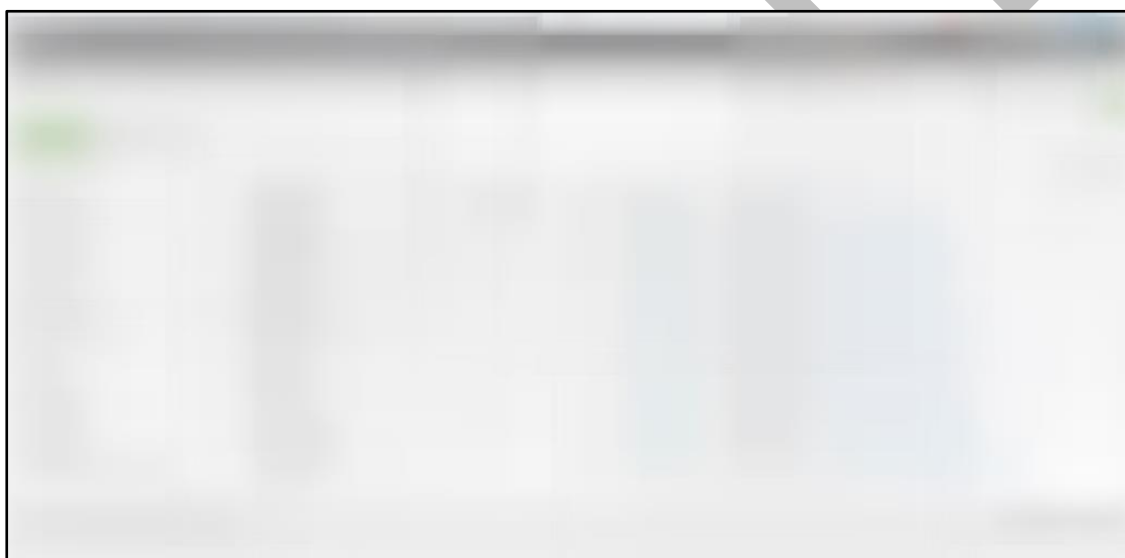
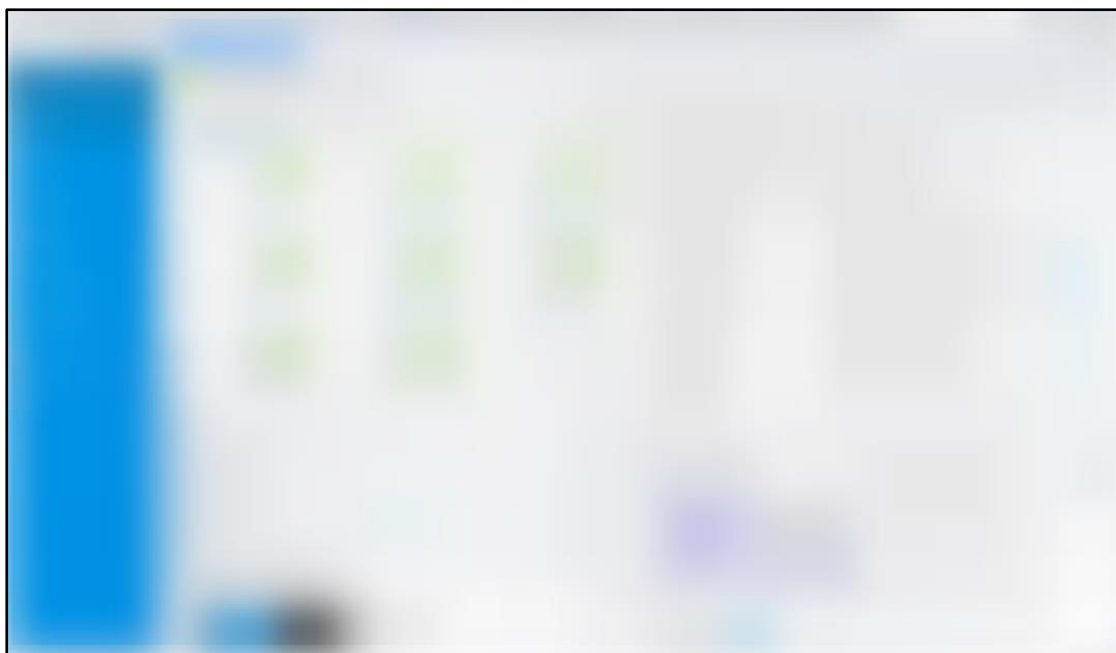
We tried logging into 100+ applications with the default credentials, and we got access to 10+ of them. A big part of those applications were printer admin panels, switch admin panels, splunk admin panels, or other such web panels that allowed default credentials login.



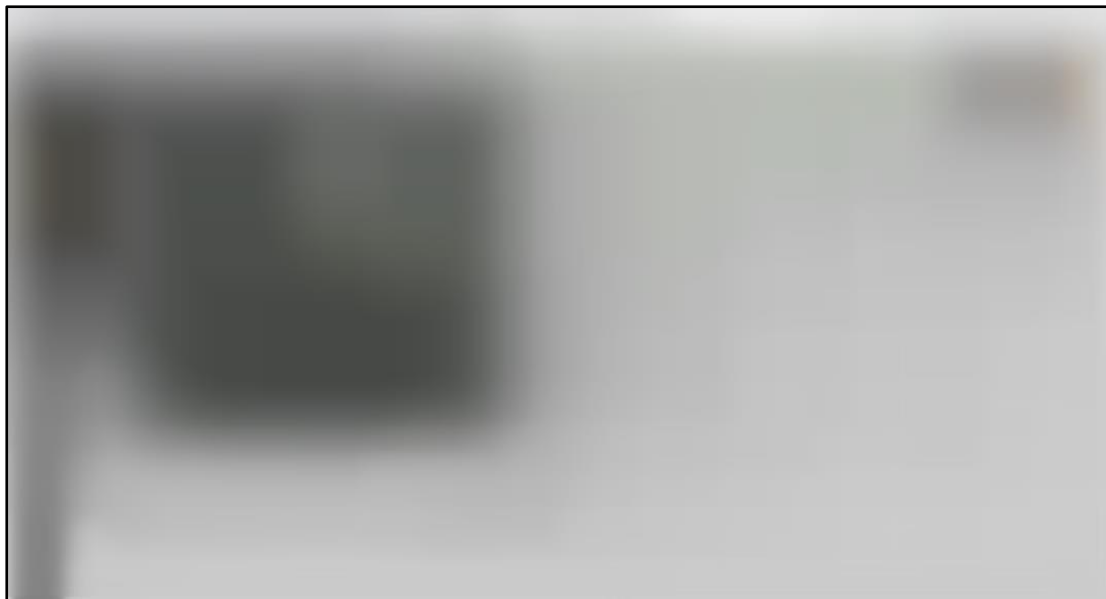
We observed that a huge number of printer admin panels were using default passwords.



However, few critical servers also had default password enabled as shown in the screenshot below.



Few camera systems allowed us access to their web panel without any authentication



SAMPLE

5.12 Network Segregation - No network segregation for different internal sites/subnets

Note: To access the different subnets, the attacker first needs to login into the VPN and gain access to the internal network.

Potential Impact: MEDIUM

Description:

Network segregation involves developing and enforcing a ruleset for controlling the communications between specific hosts and services. The aim is to restrict the level of access to sensitive information, hosts, and services while ensuring an organization can continue to operate effectively. To be effective, network segmentation and segregation measures must be carefully planned, robustly enforced, closely monitored, and be unable to be bypassed.

Location	Attack Vector Worked
N/A	✓

Possible Impact / Consequence:

- An attacker who is inside the network via VPN could access all the different subnets which drastically increases the attack surface for the attacker.
- Anyone connected to the network will have access to the whole network, including critical panels and portals..

Recommendation:

- Enable segregation on the internal network and securely configure internal VLAN's.

Tools Used:

- Manual

References:

- <https://advisera.com/27001academy/blog/2015/11/02/requirements-to-implement-network-segregation-according-to-iso-27001-control-a-13-1-3/>
-

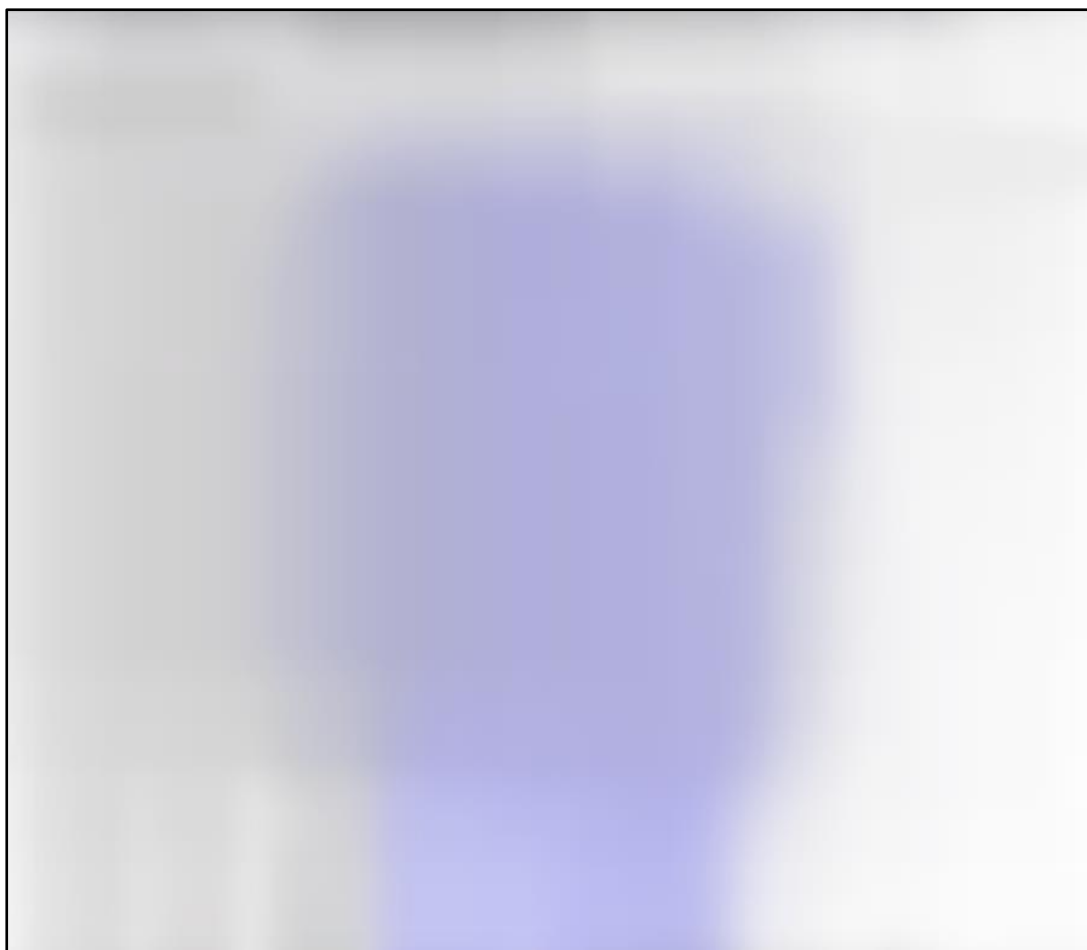
Proof of Vulnerability:

Below is the screenshot of different subnets data that we collected from active directory enumeration.



Screenshots of different web applications hosted in different subnets





We could access many different sites/subnets from <client-name> domain after connecting to the VPN, which should not be the ideal/secure case.

SAM

5.13 External Attacks – No 2FA for VPN access and no external device authentication (Gained VPN Access to internal network using credentials gathered during phishing activity)

Note: To log in to the VPN and gain access to the internal network, the attacker first needs to gain access to one valid set of <url> credential.

Potential Impact: **HIGH**

Description:

Two-factor authentication (2FA), sometimes referred to as two-step verification or dual factor authentication, is a security process in which the user provides two different authentication factors to verify themselves to better protect both the user's credentials and the resources the user can access.

VPN access allows users to remotely connect to the internal network. This enables the user to access internal infrastructure and portals easily. However, this also means that anyone with the access to credentials of VPN also has access to the internal infrastructure. It is important to have 2FA to protect the service.

The <client-name> VPN service is open to everyone having a valid set of credentials, without any 2FA. This allows easy access to the internal network.

Location	Attack Vector Worked
N/A	✓

Possible Impact / Consequence:

- Anyone with a valid set of credentials can gain full access to the internal network to all the different subnets.
- An attacker staying inside the internal network for a prolonged period could lead to successful hacking attacks against internal servers, which could lead to loss of sensitive data.

Recommendation:

- Enable 2FA on VPN Service..

Tools Used:

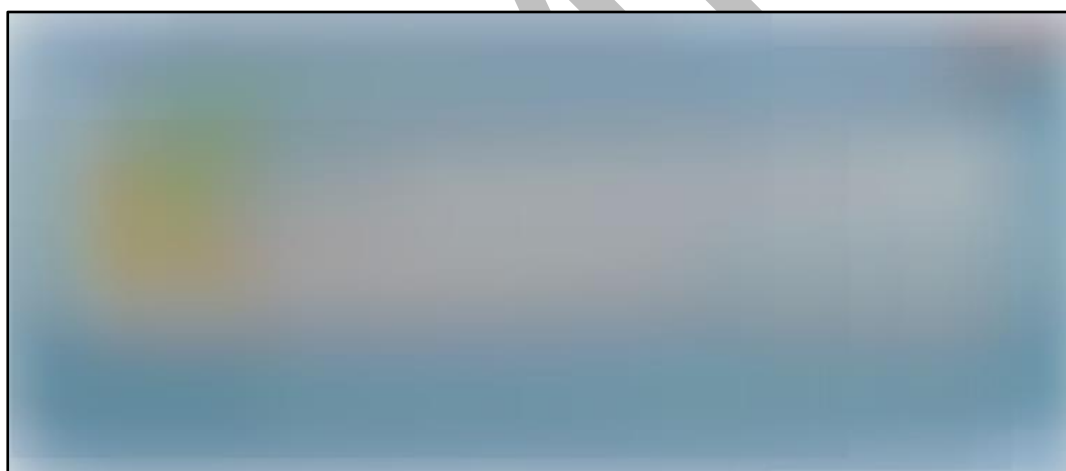
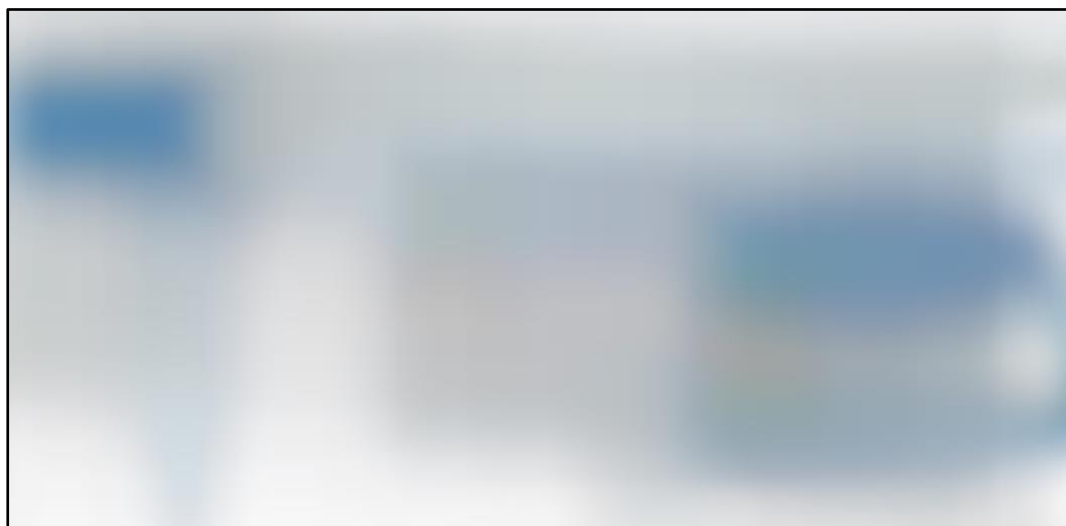
- Cisco Any Connect

References:

- <https://searchsecurity.techtarget.com/definition/two-factor-authentication>

Proof of Vulnerability:

As you can see in the screenshot, we could log in to the VPN using the obtained credentials. As per our observation, the credentials working for <url> also works for the VPN login.



As we can see in the below screenshot, we got connected to the <domain-name> VPN.

```
C:\Users\rashid>ipconfig
Windows IP Configuration

Ethernet adapter Local Area Connection 2:

Connection-specific DNS Suffix . . . : 
Link-local IPv6 Address . . . . . : 
Link-local IPv6 Address . . . . . : 
Link-local IPv6 Address . . . . . : 
IPv4 Address. . . . . : 
Subnet Mask . . . . . : 
Default Gateway . . . . . :
```

6. Detailed Failed Test Cases / Strengths

Ref	Finding	Severity
1	Physical Security: Physical security at the perimeter was very strong at these locations (Location-2, Location-3)	N/A
2	Exploitation: Failed to move laterally across different systems (LAPS Implementation)	N/A
3	Observation: System patch level is good against critical remote vulnerabilities	N/A

SAMPLE

6.1 Physical Security – Physical security at the perimeter was very strong at these locations (Location-2, Location-3)

Severity: N/A

Description:

Physical security is the protection of personnel, hardware, software, networks and data from physical actions and events that could cause serious loss or damage to an enterprise, agency, or institution. Physical security is often overlooked, and its importance is underestimated in favor of more technical threats such as hacking, malware, and cyberespionage. However, breaches of physical security can be carried out with brute force and little or no technical knowledge on the part of an attacker.

Location	Access to campus	Access to the number of ODC(s)
Location-1	✓	✓ (3)
Location-2	✗	NA
Location-3	✗	NA
Location-4	✓	✓ (3)

Tools Used:

- Manual

Detailed Description:

Since It was very difficult for us (Red Team) to tailgate through metro style gate and very easy to tailgate through single glass doors, we would recommend installing metro-style gate everywhere which allows only one person to go through at one time.

It was impossible for us to get inside campuses where there was a 2-step barricade – First through a RFID reader(bar) which checks if the card is valid or not and the second through RFID enabled entry gate.

At few locations (Location-2, Location-3) it was not possible for us to get inside via tailgating as the security was too tight and the guards were alert.

The exact details as what prevented us from getting inside the above-mentioned locations are described in detail in the storyline section of this report.

6.2 Exploitation: Failed to move laterally across different systems (LAPS Implementation)

Severity: N/A

Description:

Lateral movement usually involves activities related to reconnaissance, credentials stealing, and infiltrating other computers. When communication with the compromised systems and C&C (command and control) servers is established, threat actors need to sustain persistent access across the network. To do so, they have to move laterally within the network and gain higher privileges through the use of different tools. This, in turn, enables threat actors to have access to servers, which contain valuable information.

The "Local Administrator Password Solution" (LAPS) provides management of local account passwords of domain joined computers. Passwords are stored in Active Directory (AD) and protected by ACL, so only eligible users can read it or request its reset. For occasions when login is required without domain credentials, password management can become complex. LAPS simplify password management while helping customers implement recommended defences against cyberattacks. In particular, It mitigates the risk of lateral escalation that results when customers have the same administrative local account and password combination on many computers.

We observed that lateral movement across systems was not possible with the set of credentials we gathered during different attacks and LAPS did not allow us to move laterally across systems with local administrator account credentials.

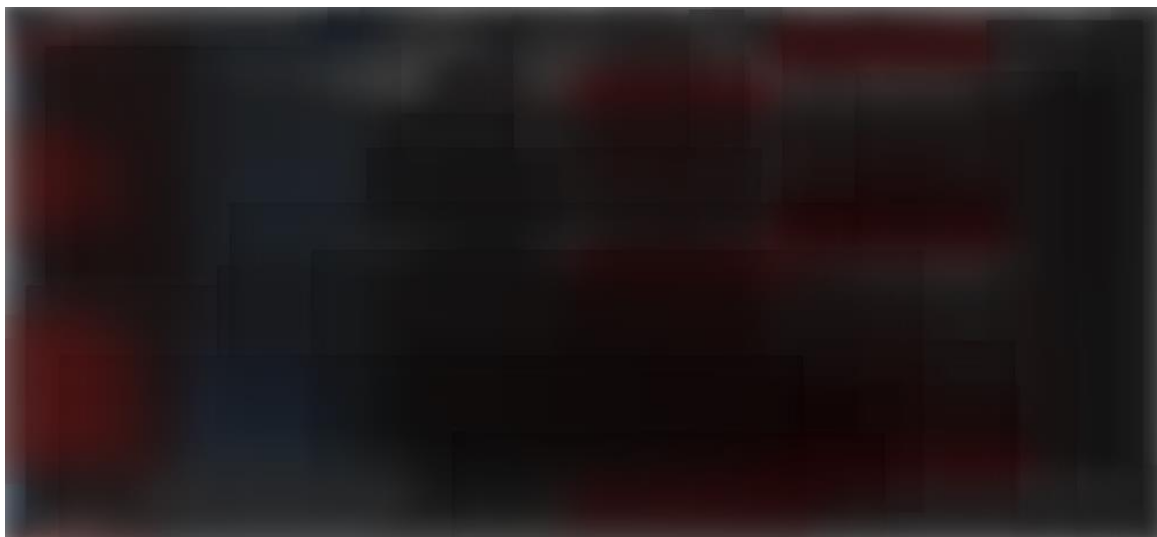
Tools Used:

- Crackmapexec
- Smbexec
- Psexec
- RDP client

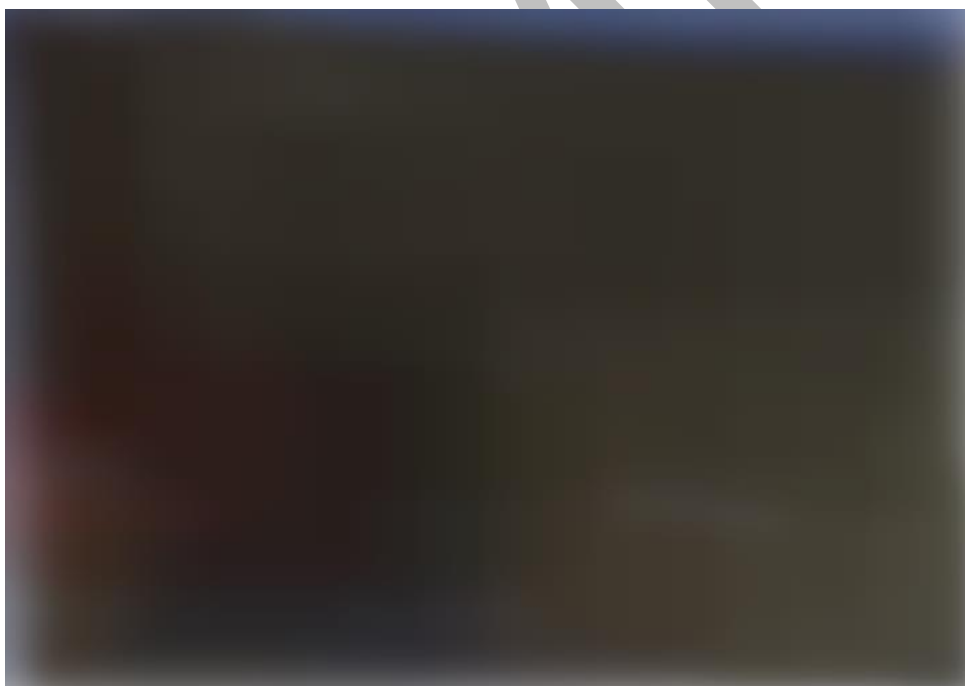
Detailed Description:

During the assessment, we got many <client-name> domain user credentials via different means but none of them had administrator access. In order to successfully move laterally, we needed to get a privileged credential which we never got during the entire assessment.

Lateral movement attempt using crackmapexec and <client-name> domain user credentials.



During the assessment, we observed that LAPS was implemented across all the machines we accessed in <client-name> domain. We were able to dump the local administrator account hashes for few machines, but we could not use the same set of hashes to login to any other system (pass the hash) as each system had a different local admin password.



The exact details as what prevented us from getting inside the above-mentioned locations are described in detail in the storyline section of this report.

6.3 **Observation:** System patch level is good against critical remote vulnerabilities

Severity: N/A

Description:

During the assessment, we found out that the system patch level is good against critical remote exploits which could lead access to the remote host.

Tools Used:

- Manual
-

Detailed Description:

During the assessment, we found out that the system patch level is good against critical remote exploits, leading to access to the remote host.

We identified many email servers during the assessment, but we could not find any critical remote vulnerabilities on them.

We classified critical hosts and tried to identify any critical remote vulnerability present on them, but we couldn't find any.

We tried looking for MS17-010 (Eternal Blue), MS14-068 (Kerberos vulnerability), CVE-2019-0708 (Bluekeep), etc. but we didn't find any.

While trying NBT-NS and LLMNR spoofing, we also observed that majority of hosts have SMB signing enabled, which prevented NTLM relaying attack against most hosts in the internal network.

We tried attacks against available Wi-Fi networks as well, and we did not find any critical vulnerabilities in the Enterprise Wi-Fi network.



7. We Prescribe

This analysis is based on the known threats as of the date of this report. We recommend that all recommendations suggested in this document be performed in order to ensure the overall security of the systems and applications. Specifically, the following action should be taken:

Ref		Severity	Recommendations
1	Recon - Enumerated subdomains of <url> and performed vulnerability assessment on them	INFO	<ul style="list-style-type: none"> There is no security issue in the enumerated external domains so far.
2	Recon - Creating Fake employee ID card is possible	MEDIUM	<ul style="list-style-type: none"> Request the respective hosting sites to remove all the scanned copies of the <client-name> ID-cards from the Internet. Spread awareness to the employees to not upload a picture of their ID-card on the Internet/social media.
3	Physical Security - Got Access to the building/ODC via tailgating/ Sneaking in multiple USB devices and a mini laptop inside	MEDIUM	<ul style="list-style-type: none"> Tighten the physical security, especially at the entrance across all locations. Since it was very difficult for us (Red Team) to tailgate through metro style gate and very easy to tailgate through single glass doors, we would recommend installing metro-style gate everywhere at the perimeter, which allows only one person to go through at one time. It was impossible for us to get inside campuses where there was a 2-step barricade – First through an RFID reader(bar) which checks if the card is valid or not, and the second through an RFID enabled entry gate.
4	Social Engineering – Gathered Employee credentials via email phishing activity	HIGH	<ul style="list-style-type: none"> Spread awareness among employees to not enter their credentials on any suspicious phishing emails. The monitoring team should be more alert when any such mass phishing emails are received and should immediately block the phishing domain and inform the employees.

5	Social Engineering – Gained physical access to machines via social engineering with Employees who were on the premises	MEDIUM	<ul style="list-style-type: none"> Spread awareness among employees to not handover 'unlocked' systems to any unknown person impersonating as someone from the IT/IS team. Spread awareness among employees to not enter their credentials on any popup/webpage when told to do so by an unknown person.
6	Social Engineering – Gathered Employee credentials via Fake Wi-Fi captive portal	MEDIUM	<ul style="list-style-type: none"> Install X-Ray scanners at the campus entry points where it is not implemented. Increase the physical security on the premises so that an attacker cannot get inside the premises without having a valid RFID-enabled ID-card. Spread awareness among employees to not enter their credentials on any suspicious Wi-Fi access point.
7	Exploitation – Malware implant execution and obtaining a reverse beacon (with SYSTEM privilege)	HIGH	<ul style="list-style-type: none"> Strengthen the central log aggregation and anomaly detection team (SOC) in order to detect such adversaries, which normally bypasses Endpoint protection software.
8	Exploitation – NBT-NS and LLMNR poisoning to gather NTLMv2 hashes of domain users	MEDIUM	<ul style="list-style-type: none"> Disable LLMNR and NetBIOS in local computer security settings or group policy if they are not needed within an environment.
9	Exploitation –Kerberoasting attack	MEDIUM	<ul style="list-style-type: none"> The best mitigation for this attack is to ensure your service accounts that use Kerberos with SPN values leverage long and complex passwords. To detect the attack in progress, monitor for abnormal account usage. Service accounts traditionally should be used from the same systems in the same ways, so it is possible to detect authentication anomalies. Also, you can monitor for service ticket requests in Active Directory to look for spikes in those requests.

10	Exploitation – Ineffective Access Control to the network	MEDIUM	<ul style="list-style-type: none"> Implement the NAC systems correctly so that the host machine only get connected to the network once all the conditions are met, which could include OS details, patch level, AV update details, etc
11	Web Application Security - Default credentials login (Internally hosted web apps) [we could login to 30% of the found web panels]	HIGH	<ul style="list-style-type: none"> Change the default passwords to a secure password.
12	Network Segregation - No network segregation for different internal sites/ subnets.	MEDIUM	<ul style="list-style-type: none"> Enable segregation on the internal network and securely configure internal VLAN's.
13	External Attacks – No 2FA for VPN access and no external device authentication (Gained VPN Access to internal network using credentials gathered during phishing activity)	HIGH	<ul style="list-style-type: none"> Enable 2FA (two-factor authentication) and device authentication on VPN Service

SAMPLE

Annexure A (Red Team Assessment storyline)

This section describes in detail what exact actions were performed during the Red Team assessment and what was the outcome. This will help the reader of this report to understand how the Red Team uncovered the mentioned vulnerabilities and where the organisation lacked in implementing security controls.

1.1 Location-1

<Date>

10 AM – 5 PM: Performed physical reconnaissance

- Performed physical reconnaissance on the Location-1 office from outside. This included assessment of the physical layout of the facility, possible entry points, security checks present (RFID enabled Glass door or Metro style gate) at the perimeter (campus gate/ building gate), as well as the behavioral analysis of the employees and the security staff in order to get an idea as how alert they are during different hours in the entire day.
- During the physical reconnaissance of the Location-1, we deduced that there is a possibility of entry to the campus via the west gate. The primary reason for this deduction was lack of RFID based doors at the campus entrances. We observed that the hardware (RFID scanning bar) was not present. This meant that anyone wearing a legitimate looking <client-name> Identity Card could enter the campus.

Observation:

- Possible entry from west gate. No RFID/X-Ray check at the main entrance.

<Date>

10 AM: Physical entry inside the campus

- Physical entry was gained inside the campus wearing a fake Identity card due to the lack of any RFID based checking at the campus entrance.

10:15 AM: Physical entry in a building named '<name>' via tail-gating during Morning shift time.

- After observing the entrance of a building named "<name>", we found out that, the morning time was too crowd. we inferred that the security person deployed is not very vigilant and it is possible to gain entry via tailgating with the crowd, just by wearing the valid ID card.

- I (member of the Red Team) gained access to the building and then further gained access to an <building-name> via tail-gating (**weakness**). I also carried 1 mini laptop and 5 USB devices along with me, loaded with different tools for various attacks(**weakness**).

11 AM: Administrator access on the workstation

- Occupied an empty workstation on the <building-name> (**weakness**). Since everyone was busy at their works, no one seems suspicious about me (member of Red Team).
- Booted Kali Linux on that system and got an IP address via DHCP (**weakness**). We observed here that either the NAC is not enabled on that port or the NAC server is not configured correctly as we got an IP address via DHCP, when we booted a completely different Operating system.
- Since the Windows Disk encryption is not enabled, I mounted Windows drive on Linux and browsed through user directories and dumped PII documents like <document-name> from the users home folder, Security Accounts Manager (SAM) and System hive from registry. Retrieved the local administrator and other system user hashes from the above Registry hives (**weakness**).
- Performed network enumeration and gathered important IP addresses (Gateway, DNS, etc.). Copied the collected data onto Linux filesystem and rebooted the system (**weakness**).
- Since the Full Disk Encryption was not enabled, used Konboot USB to bypass the windows authentication which gave us the local Administrator access of the machine(**weakness**).
- Dumped all the domain cached hashes in that system by using Mimikatz (**weakness**).
- Found LAPS (Local Administrator Password Solution) is implemented which generates a different local administrator password for each user (**Strength**), thus preventing us from logging into another machine using the same NTLM hash.
- Installed Obfuscated malware implant on the system by bypassing the Endpoint protection software for later access and we got reverse beacon on our C2 (COMMAND & CONTROL) server (**weakness**) .
- Once we got access to the Windows machine, we enumerated local system to find out the OS details, Endpoint protection software, DLP agent, NAC agent, System Patch level, etc.

1:30 PM: Exited the building (Just walked out wearing fake <client-name> ID-Card at lunch break time)

Observation:

- Easy to get inside campus and ODC's via tail-gating where the security personnel are not very active.
- Easy to carry Laptops, USB devices hidden in the wallet inside ODC's.
- No File system encryption.
- NAC implementation is not proper.
- Malware execution is possible bypassing the Endpoint detection.
- Data exfiltration is possible bypassing DLP monitoring.
- Identification of host based defensive software installed (Endpoint protection, DLP, Web proxy)
- Got an idea about the network size, the domain name to which the employees are connected to and gathered data about the domain which had information about approximately 10,000 domain users, domain names, their email address, privilege levels, sites, subnets, etc.

<DATE>

- Physical entry was gained inside the campus with a fake Identity card due to the lack of any RFID based checking at the campus entrance.

10 AM: Got into <building-name>

- After observing the entrance of a building named "<name>", we inferred that the security person deployed is not very vigilant and it is possible to gain entry via tailgating with the employees, just by wearing the valid ID card. (**Weakness**)
- Got access to a vacant system, booted kali Linux and ran responder tool on the system. Used responder and Social engineering to gather plain text domain credentials and hashes (Net-NTLMv2) (**weakness**).
- Enumerated systems connected to <domain-controller-name> Domain controller and <domain-name> domain after logging in with the gathered creds.
- Checked for saved passwords in Active directory group policy preferences (SYSVOL directory), but we didn't found anything. (**Strength**)

11:30 AM: Social engineering with employee to download and execute a script:

- Convinced an employee to download and execute a PowerShell script hosted on our webserver. That script enumerated the <domain-name> domain and gathered data of all the Users, Privileges, sites, subnets, email address, last password change, Active directory trusts, etc. The employee agreed to execute this script as part of the system security audit (**weakness**).

- Installed Obfuscated malware implant on the system by bypassing the Endpoint protection software with appropriate domain user privileges and we got reverse beacon on our C2 (COMMAND & CONTROL) server (**weakness**).
- Encrypted and then Exfiltrated (Uploaded) the gathered data to our USB (**weakness**).

1:45 PM: Got into another building <building-name>.

- Bypassed the windows login using Konboot. Could not get administrator rights on this system using the same technique.
- Got access to an empty windows system connected to <domain-name> domain and Booted kali Linux on it and installed the obfuscated malware implant on the windows file system. (**weakness**)
- Executed our malware which got blocked by Endpoint protection software. Failed to achieve malware persistence as we can't execute our malware on the machine. (**Strength**)
- Ran few enumeration scripts and collected data from <domain-name> Domain controller.

5:00 PM: Exited the campus (via tail-gating) .

Observation:

- Easy to get inside campus and ODC's via tail-gating where the security personnel are not available.
- Social engineering can be performed on employees.
- Malware execution is possible bypassing the Endpoint detection.
- No File system encryption.
- NAC implementation is not proper.
- Data exfiltration is possible bypassing DLP monitoring.
- Identification of host based defensive software installed (Endpoint protection, DLP, Web proxy)
- Got an idea about the network size, the domain name to which the employees are connected to and gathered data about the domain.

<DATE>

12:30 PM: Gained access to the <campus-name> campus as the System Administrator allowed entry using his access card.

- Got into the cafeteria next to the <client-name> office with our captive portal phishing setup hidden in the bag and started a fake access point with '<client-name>' name.
- Kept few paper printouts at strategic locations inside the cafeteria which said – 'Free Wi-Fi for <client-name> employees.

There were very few <client-name> employees in the cafeteria. The attack failed as the employees did not connect to the Wi-Fi.

4:30 PM: Exited the campus

1.2 Location-2

<DATE>

2 PM – Performed physical reconnaissance of <campus-name> campus.

- Performed physical reconnaissance on the <campus-name> campus from outside. This included assessment of the physical layout of the facility, possible entry points, security checks present (RFID enabled Glass door or Metro style gate) at the perimeter (campus gate/ building gate), as well as the behavioral analysis of the employees and the security staff in order to get an idea as how alert they are during different hours in the entire day.

<DATE>

9:30 AM - Attempted to get into <campus-name> campus but failed (Strength).

- The campus had tight physical security and the security personals were very alert. There were multiple entries to the campus but all of them had metro style RFID enabled doors which allowed only one person to go through at once.

<DATE>

1:30 PM - Multiple attempts to get inside <campus-name> campus but failed (Strength).

1.3 Location-3

<DATE>

1 PM - Performed physical reconnaissance of <campus-name> campus.

- Performed physical reconnaissance on the <campus-name> campus from outside. This included assessment of the physical layout of the facility, possible entry points, security checks present (RFID enabled Glass door or Metro style gate) at the perimeter (campus gate/ building gate), as well as the behavioral analysis of the employees and the security staff in order to get an idea as how alert they are during different hours in the entire day.

<DATE>

10 AM - Attempted to get into <campus-name> campus but failed (Strength).

- The campus had tight physical security and the security personals were very alert. There were multiple entries to the campus but all of them had metro style RFID enabled doors which allowed only one person to go through at once.

<DATE>

1:30 PM - Multiple attempts to get inside <campus-name> campus but failed (Strength).

<DATE>

11:30 PM - Multiple attempts to get inside <campus-name> campus but failed (Strength).

<DATE>

3 PM - Multiple attempts to get inside <campus-name> campus but failed (Strength).

1.4 Location-4

<DATE>

11 AM: Performed physical reconnaissance of <campus-name> campus.

- Performed physical reconnaissance on the <campus-name> campus from outside. This included assessment of the physical layout of the facility, possible entry points, security checks present (RFID enabled Glass door or Metro style gate) at the perimeter (campus gate/ building gate), as well as the behavioral analysis of the employees and the security staff in order to get an idea how alert they are during different hours in the entire day.
- During the physical reconnaissance of the <building-name> Office, we deduced that there is a possibility of entry to the campus via the “East Exit Gate” during the break time. The primary reason for this deduction was the observation that employees were using the “East Exit Gate” to move in and out of campus during the free hours for small breaks, This combined with the lack of RFID based checking at the “East Exit Gate” meant that anyone wearing a legitimate looking Identity Card could enter the campus along with other legitimate employees.

<DATE>

3 PM: Gathered data from 15 people via fake survey and attempted RFID cloning for multiple people.

Disguised as employees of an Eye Glass company, we interacted with 15 <client-name> employees with the pretext of a product survey to gather data as well as attempt RFID Card cloning. We hide our RFID card cloning software inside a small box for stealth and tried bringing our hardware close to the ID-Card that the employee was wearing. We failed as the card was an NFC ISO in which the data was encrypted using a custom key and thus it cannot be read/cloned (**Strength**).

4:30 PM: Exited the <building-name> building via tail-gating.

<DATE>

2:30 PM: Got into the common canteen of the building

- Went to canteen and placed fake printed documents at 5 locations to lure users into connecting to a captive portal which hosted a phishing page. Was able to carry all relevant hardware in the bag as there was no checking at the entrance (**weakness**).
- Got access to the canteen due to the lack of any RFID based checking and setup a Fake Captive Portal (Wi-Fi). All the relevant hardware could be brought in as there was no X-Ray scanner at the entrance. (**weakness**)

- Few printouts stating “Wi-Fi is available for the <client-name> employees” was placed at strategic locations to lure employees into connecting to the captive portal.

Gathered 8 valid employee credentials from the captive portal phishing attempt. (**weakness**)

4:15 PM: Exited the <building-name> building via tail-gating.

Observation:

- Wi-Fi captive portal phishing is possible
- Hosts/Network is not vulnerable to Critical vulnerabilities, NTLM Relaying attack, etc.

<DATE>

12:40 PM: Entered <campus-name> campus via exit gate.

- Tried to tail-gate through the metro style doors once inside the campus in order to get inside the main building but failed as the security guards were very alert and it was not possible to tail-gate through metro style doors (**Strength**).
- After moving around the building, I observed that the Exit door is open and the security personal is not present outside the Exit door. I entered the Exit door on the ground floor which led me to the Ground floor (**Weakness**).

1:10 PM: Entered <building-name> on 3th floor via tail-gating.

- Performed social engineering on employee(s) impersonating as someone from ‘IS’ team to download and execute a PowerShell script which asks for their domain credentials and then encrypts it and sends it to our C2 (COMMAND & CONTROL) server (**Weakness**).
- Got access to an empty workstation, Booted kali Linux and mounted windows shares and browsed through all the local user’s directories (**Weakness**). Found some sensitive client documents which had source code and few plaintext passwords. I tried logging into the system using those plain-text credentials, but it did not work (**Strength**).
- Ran responder tool on Kali Linux and performed social engineering to get credentials and Net-NTLMv2 hashes (**weakness**).
- Dumped SAM hashes of local machine and found LAPS (Local Administrator Password Solution) is implemented which generates a different local administrator password for each user (**Strength**), thus preventing us from logging into another machine using the same NTLM hash.

2 PM: Executed malware and got reverse beacon on our C2 (Command & Control) server and performed different attacks

- Tried kerberoasting attack against Domain Controller (<client-name> Domain) and gained hashes of several service accounts. Cracked a lot of hashes simultaneously but none of the service accounts were added in any higher privilege (Domain Admin, Enterprise Admin, etc.) group (**Strength**).
- Tried logging in with service accounts on different machines across the network but couldn't login due to domain policy restriction (**Strength**).
- Elevated privileges after booting the system with Konboot USB. This attack vector was possible because Full Drive Encryption was not enabled on the machine. (**Weakness**)
- Executed malware implant on the system, bypassing Endpoint detection software and used different persistence techniques to persist in the system for a long period of time. Restarted machine and logged in as domain user. Received reverse beacon with 'SYSTEM' privilege. (**Weakness**)
- Mapped the network selectively using Nmap and tried finding any critical vulnerabilities or misconfigurations which could be exploited to gain access to any high value target but couldn't find any critical vulnerabilities (**Strength**)
- Tried to laterally move across different systems using the local administrator account hash (Pass the Hash attack) but failed (**Strength**).
- Tried to check if the domain (<client-name> Domain) credential we have, has privileged access to any other host on the same subnet using crackmapexec tool but it failed (**Strength**).

5:50 PM: Exited the main building through parking lot via tail-gating (Weakness**)**

- Since the main building exit had metro style doors, I had exit via Exit-Gate, as the security Guard was not there.

Observation:

- It was very tough to get inside the main building via tail-gating as the guards were very alert.
- There was a single Exit-Gate door in the building through which tail-gating can be done.
- Full Drive encryption was not present on the systems.
- It was not very difficult to convince employees to download and execute a PowerShell script.
- Malware execution is possible bypassing the defenses.
- LAPS implementation prevented us from lateral movement.
- Active Directory is securely configured.

<DATE> - <DATE>: Maintain Access via malware beacon persistence:

- Added a local administrator user on the same machine for better persistence.
- Enumerated the internal network and hosts to find any critical vulnerability but didn't find any (**Strength**).
- We had internal access to the <client-name> network through that malware for 2 months which went undetected by the internal security team (**Weakness**).

1.5 Email Phishing Activity**<DATE>****3:15 PM: Phishing email sent to set of 3,000 employees**

- The email addresses were obtained OSINT from public available sites including social media sites and search engines, tools like Theharvester, maltego and data gathered from onsite phishing.
- Multiple attempts were made to phish the employees. This included sending phishing emails to set of 3,000 Employees with different group of interest in different time intervals.

<DATE>**10:15 AM: Phishing email sent to next set of 2,400 employees**

- Multiple attempts were made to phish the employees. This included sending phishing emails to next set of 2,400 Employees with different group of interest in different time intervals.

<DATE>**1:55 PM: Same phishing email was sent to all the employees for the second time for a greater success rate.**

- We gathered around 450 credentials from our phishing attempt (**Weakness**).
- The success ratio for our phishing attempt was medium(**Strength**).

1.6 Password Spray

<DATE>

A password spray attack was conducted on Outlook web access <url> using a Mailsniper powershell script. The attack consisted of trying a common password on approximately 5,000 accounts. The attack didn't yield any success. But we can able to verify the credentials gathered from phishing attack. (Strength)

1.7 VPN Login

01/02/2019

The VPN service was discovered during external enumeration. It was observed that the credentials for the Outlook Web Access portal were working for the VPN (Virtual Private Network) service as well. Since the VPN service did not enforce 2FA (Two Factor Authorization), it virtually provided access to the internal network.

- The VPN connection offered unrestricted access to most subnets of the network. Web Portals, critical Web panels as well as LIVE CCTV (Closed-circuit television) admin panels were discovered as a result of enumeration. It was found that few of the web panels had default passwords. (Weakness)
- Even though, we had VPN access and access to internal network, we could not exploit any high value target as the systems were patched against any remote critical vulnerabilities (Strength).

Annexure B (Tools Used)

Reconnaissance

- Sublist3r
- Amass
- Dnsdumpster
- Google dorks
- Virustotal subdomain enumeration tool
- Censys.io
- Shodan
- Aquatone
- Whois database

Enumeration

- Nmap
- Angry IP scanner
- Gobuster
- Dirb
- ADRecon
- Bloodhound
- PowerSploit

Exploitation & Post-Exploitation

- Powershell Empire
- Crackmapexec
- Responder
- NTLMRelayx
- PsExec, SMBExec, WMIExec
- Nikto
- BurpSuite
- Certutil
- Cisco AnyConnect

Hash cracking

- Hashcat
- Hatecrack
- Uniqpass

Email Phishing

- Gophish