THOMAS MORE

Netcure SOC Project

Internship Journal

Baisangur Dudayev
Bachelor Electronics-ICT - Cloud & Cybersecurity

2024 - 2025

Table of Contents

1. Introduction

This internship logbook provides a chronological overview of the activities, tasks, and responsibilities I carried out during my internship at Netcure. The document describes my daily progress, the technical challenges I encountered, and the solutions I implemented.

The main focus of my internship was on security automation and reporting within a SOC (Security Operations Center) environment. This included developing and improving reporting scripts (e.g., Guardian360, CyberAlarm, and WatchGuard), building automated workflows using n8n, integrating external services such as VocalNotify, and contributing to the implementation of TheHive for incident management.

Importantly, this logbook was written day by day during the internship itself, ensuring that the content reflects the real-time tasks, challenges, and knowledge gained. The document is therefore not only a technical record but also a reflection of my development in teamwork, communication, and professional practices in a cybersecurity setting.

2. Weekly Log

Week 1 (24/02 – 28/02)

Monday
Upon arrival, I received an introduction to the company from Erik and Tosin, followed by an explanation of the SOC project that Bryan and I would be working on. I conducted research on Grafana and the ELK stack — the main technologies for the project — and installed Vagrant and updated VirtualBox to prepare my development environment.

Later, Marieke provided an overview of the NIS 2 directive and its relevance to Netcure's services. We discussed how the SOC project supports compliance by enabling centralized monitoring, threat detection, and reporting.

Additionally, I met other interns from Thomas More Mechelen and reviewed their progress. I began troubleshooting issues with Vagrant on my laptop, which required further investigation. Administrative onboarding was completed, including signing documents, receiving badges, and a tour of the company's infrastructure.

Tuesday
I met the other intern who had been absent on Monday and reviewed the work completed by the Thomas More Mechelen interns, who had started a week earlier. Together with Tosin, I attempted to resolve a persistent technical issue, but without success.

I encountered recurring startup problems with my laptop, which also affected the Vagrant installation. To ensure a stable development environment, I performed a clean reinstall of Windows (using a USB installation drive provided by another intern). After installation, the Wi-Fi did not function because the drivers had to be installed manually. I resolved this at home by connecting an Ethernet cable to download and install the correct Wi-Fi drivers.

Once connectivity was restored, I upgraded to Windows 11 and reinstalled all required tools, including Vagrant and VirtualBox. During the afternoon, I analyzed code from other interns and assisted one of them in troubleshooting an Ansible error.

Wednesday
I installed several required programs and restored backups of important files to my laptop. I continued analyzing code developed by other interns.

Tosin assigned me the task of writing code to generate and use new SSL/TLS certificates in Ansible for the ELK stack. I completed this task and verified its functionality. Additionally, I provided guidance to another intern on specific Linux commands and utilities, explaining their usage in a development environment.

Since the test environment for the virtual machine was not yet ready, Tosin instructed us to focus on documenting the existing code. I spent the remainder of the day preparing and improving technical documentation.

Thursday
Our SOC team — consisting of myself, Bryan (absent on most Thursdays), two interns from Thomas More Mechelen, and our coach Tosin — discussed the next steps for the project. We also held a brainstorming session on how to securely store and manage the project's keys and secrets. This provided me with a clearer understanding of the overall system architecture.

I continued working on an existing issue with SSL/TLS certificates. Additionally, one of the interns needed to modify application port settings and requested my assistance. I demonstrated the process, explaining relevant troubleshooting commands. We first tested the changes manually in the configuration files of the live application on his laptop to ensure proper functionality. He then automated the process in the application code, while I resumed work on the certificate task.

Upon further investigation, I discovered that the application already generates unique certificates for each installation, making the creation of custom certificates unnecessary. Based on this finding, we decided not to proceed with the new certificate generation process.

Friday
We continued discussions on how to securely store and implement secrets and passwords within the SOC environment. One proposed solution required the --ask-vault-pass command to provide a password during code execution. Since this is an Ansible command, it cannot be executed natively in Windows — only in Linux. While this would not be an issue in the Linux-based test environment, we needed a way to make it work locally on Windows systems.

To address this, I installed Windows Subsystem for Linux (WSL) to create a local Linux environment. I tested the setup by writing a simple Ansible configuration and executing it within WSL, successfully using the `--ask-vault-pass` command with Ansible Vault to securely handle secrets during code execution.

Later in the day, Bryan and I received additional information about the NIS 2 directive and how the company assists clients in implementing it. Finally, I worked on enabling the Ansible code to run via WSL on my virtual machine, allowing us to simulate a realistic test environment.

Week 2 (3/03 – 07/03)

Monday
I continued working on enabling Windows Subsystem for Linux (WSL) to integrate properly with the project's virtual machine environment, ensuring that Linux-based tools such as Ansible could be executed and tested in a realistic setup. The configuration was completed successfully, and I documented the setup and usage instructions.

The interns in Mechelen frequently required assistance with implementation tasks, so I regularly provided support, troubleshooting various errors and guiding them through solutions. Collaboration with the other interns is going well.

Tuesday
Together with the interns in Mechelen, I worked on configuring HTTPS for Kibana, which required generating SSL/TLS certificates. I focused on the certificate creation process.

In parallel, I assisted the Mechelen interns multiple times with errors, primarily related to WSL usage and configuration. This support took a significant amount of time but ensured their progress. To improve my own understanding, I reviewed a training video from Tosin on Kibana certificates and conducted further research on the topic.

Wednesday
I resolved a WSL synchronization issue, improving its performance and enabling more efficient workflows. I documented the fix and assisted another intern with its implementation.

As a team, we discussed which logs to retain from the security platforms used by clients. I expanded my notes and researched the optimal approach for creating and managing Logstash configuration files to handle these logs.

Additionally, I resolved several VirtualBox errors and reviewed an issue in Tosin's code, which he ultimately identified. Tosin also asked me and one of the Mechelen interns to review Elastic documentation on key system configurations for the production environment and to automate selected configurations using Ansible.

Thursday
I started implementing important system configurations together with one of the Mechelen interns. We split up the work to complete it more efficiently. Another intern from Mechelen ran into several errors and asked for my assistance, so we worked together to resolve them.

I also solved a Grafana issue and an SSH problem for him before continuing with my own tasks. The configurations I implemented — using an Ansible playbook — included:

Configuring system resource limits

Setting the TCP retransmission timeout

Defining the JNA temporary directory

Additionally, the interns were provided with an extra desktop computer equipped with a VPN connection, enabling us to configure the virtual machines in the production environment.

Friday
I began the day by testing the implemented system configurations, all of which worked correctly. I then wrote documentation for the API and assisted three colleagues with technical questions. Later, we held a brainstorming session on log filtering, after which I continued configuring the log input and filtering process.

During the day, a security-related incident occurred: individuals without proper badges were allowed access to our floor by another company's employee. As this raised concerns about building access control, Tosin and I verified the situation with the reception. The incident was resolved without further issues, but it highlighted the importance of physical security measures alongside cybersecurity.

Week 3 (10/3 – 14/3)

Monday
I assisted an intern with errors in his VM and Linux terminal. In addition, I wrote and tested code to filter logs for CyberAlarm and WatchGuard.

At the end of the day, another intern asked for help with an issue he had been struggling with the entire day. I analyzed the Elastic service logs, identified the root cause, and resolved the problem.

Tuesday
Together with Bryan, I resolved several errors in his code. Afterwards, I continued working on log filtering using Logstash and Ruby.

Wednesday
Together with Bryan, I again worked on resolving errors in his code and on improving the log filtering process. In addition, I found a partial solution in Logstash for dynamically extracting data from log files.

Thursday
I optimized the filter code, allowing the logstash.conf file to handle multiple log sources without requiring separate code blocks for each one. I also made further progress in dynamically parsing data from log files.

Friday
The day started with preparations for the project plan presentation, which I delivered later in the day. I received positive feedback on my work over the past weeks, with a minor remark regarding time management after breaks.

In addition, I signed an NDA. Afterwards, Bryan and I continued working on the log filtering process, which remains time-consuming but is still progressing according to schedule.

Week 4 (17/03 – 21/03)

Monday
The production VM experienced issues with outbound internet connectivity. After investigation, I identified the cause as a DNS problem and resolved it by changing the DNS in `/etc/resolv.conf` from the localhost IP to Google's DNS server.

I also updated the Python script for CyberAlarm, which is responsible for API calls to retrieve logs. The modifications ensured that only level 1–3 logs from subscribed clients are collected, and that the correct permissions are automatically assigned to the newly created `log.json` file. This task gave me valuable hands-on experience in scripting and working with API calls.

Tuesday
I spent a significant amount of time debugging and testing the CyberAlarm script. The main challenge was implementing the *`search_after`* functionality, which required storing the value returned by Elasticsearch separately and retrieving it in the script. This ensures that log collection can resume from the correct point where it last stopped.

Wednesday
I continued debugging and testing the CyberAlarm script, further expanding and improving its functionality. I successfully solved the `search_after` challenge, enabling the script to dynamically retrieve logs based on the last collected entry.

In addition, we started weekly project presentations, with a different team member presenting each week to keep everyone aligned on progress.

Thursday
I fixed additional issues in the CyberAlarm script and improved its error handling, ensuring that problems are now clearly reported and written to a dedicated error log file.

We also attended a guest lecture on AI/automation and how it could be integrated into our project.

Friday
I prepared a new project plan presentation and updated the script by adjusting file paths and permissions. A fellow intern requested an additional log field that combines values from existing fields, which I implemented in the code.

However, testing revealed issues in the environment: the filtering work that Bryan and I performed was hindered, likely due to the test VM having insufficient CPU cores to run multiple Logstash filter pipelines simultaneously.

In the afternoon, I left the internship site earlier to deliver the project plan presentation on campus. I received detailed feedback on the content, which I will incorporate into both my next presentation and the project plan document for submission.

Week 5 (24/03 – 28/03)

Monday
Before the official start of the day, I spent an hour following the Hugging Face AI course on SmolAgents, knowledge that will be relevant in the coming weeks.

During the internship day, I worked on a new script to retrieve probes from client devices. The script is designed to remain active until all probes are collected for subscribed customers.

Additionally, I assisted fellow interns with a visualization issue in Grafana. Together with Tosin, I reviewed his Logstash code and attempted to resolve a related problem.

Tuesday
I continued working on the probe collection script, debugging, adding filtering, and finalizing its functionality. I then documented the code to ensure clarity and maintainability.

Additionally, I reviewed the WatchGuard API documentation to gain a deeper understanding of its features and integration possibilities.

Wednesday
As a team, we discussed which additional data still needed to be retrieved through new API calls. Bryan and I wrote the necessary scripts to handle these requests.

I also gave our weekly progress presentation to Erik and several other Netcure employees. The presentation went well and highlighted the significant progress made since last week, showing that we remain on schedule.

Later in the day, I assisted another intern with a Grafana API key issue where the key only had read permissions and could not perform write operations. Finally, I added documentation on creating an Elasticsearch API key that works with Grafana.

Thursday
Lasse, a Netcure employee, gave a presentation to our team on how reports from different solutions are currently generated. This served as an introduction for next week, when we will begin working on report automation. I took extensive notes and asked questions, as the current process is still largely manual and time-consuming.

In addition, I completed the final set of new log retrieval scripts. All scripts were tested, renamed with logical identifiers, and scheduled in a cron job for automatic execution — some every 5 minutes, others every 24 hours, depending on their function.

I also investigated an issue with two existing scripts that process encrypted files. These files could not be automatically decrypted in Logstash. After extensive troubleshooting, it appeared that decryption only worked when the encrypted files were deleted and re-encrypted, possibly due to an outdated key being used.

Finally, I tested the Grafana Ansible playbook created by another intern to verify its functionality on my own laptop. Although the creator had already tested it, Tosin requested additional validation before deploying it to the test environment.

Friday
I retested the Grafana Ansible playbook created by another intern, as it had not worked on my laptop the previous day. The issue was resolved by commenting out a specific line of code, after which it functioned correctly.

I was also tasked with retrieving data and logs from another client security solution, Guardian360. For this, I wrote 15 scripts, each performing API calls to collect and store the required data. Although these scripts were somewhat simpler than the ones I had previously developed, the high volume still required significant effort. I managed to complete them all in a single day, which gave me a deeper understanding of the process.

I used AI to automate parts of the script creation process, which helped me work faster and more efficiently. Combined with my growing knowledge of API calls and scripting, this allowed me to complete the task well within the planned time. I remain largely on schedule, though some small adjustments are still needed to remove unnecessary root fields from the JSON log files.

Week 6 (31/03 – 04/04)

Monday
I continued working on the Python scripts for Guardian360 that I had started the previous week. I added filtering on the root fields so that Logstash correctly identifies which field names should be used, ensuring that only meaningful data is stored in the log files.

I also modified the scripts to allow filtering per company, rather than placing all data into a single log file. In one of the API calls, the filtering did not work as described in the documentation. Tosin indicated he would contact the Guardian360 developers for clarification. In the meantime, I continued working on the other API calls.

Tuesday
I further modified the Guardian360 scripts to enable filtering per company and by time where required. After making the necessary adjustments, I updated the scripts and added them back to the crontab for automated execution.

In addition, I developed two new scripts. Together with Bryan, I troubleshot an error in a script that generates a token for Guardian360, and we also analyzed another error message in the Logstash logs.

Finally, I placed the files required for the reporting component onto the desktop computer I use for development and testing.

Wednesday
To start working on the reporting, Excel needed to be installed in order to open specific data files. As this required administrator rights, I reported the issue to Tosin, who arranged for Lasse to perform the installation remotely across all computers.

In the meantime, I began working on and researching the automation of the Guardian360 reporting. Bryan encountered an encryption issue when one of the two crontab scripts was still active, and together we designed a solution for the problem.

I also continued the Hugging Face AI Agents course, which will be relevant in the upcoming weeks. In the afternoon, Bryan gave the weekly presentation, after which the team discussed our approach to report generation. I then carried out additional research on the technical design for the reporting solution.

Thursday
As a team, we held an in-depth discussion on how to structure the reporting most effectively. Afterwards, I began drafting the report for Guardian360.

Later in the day, Bryan gave his presentation for the second time this week, which I attended.

Friday
I continued working on the Guardian360 report. The required data can now be retrieved and displayed correctly. At this stage, I am still encountering issues with generating the graphs for visualizing the data.

Week 7 (07/04 – 11/04)

Monday
I continued working on the automation of the Guardian360 reporting script. I made significant progress and was already able to generate several graphs, which was a motivating milestone. I also filtered a large amount of data from the log files and gained further insights in the process.

For the project, I am using Python, HTML, CSS, and JavaScript, with Jinja2 to automatically populate an HTML template for the report.

I encountered a minor issue in the CSV log file where some logs appeared on the wrong line in Excel if they contained the ";" character. I tested this with my script, but fortunately, the logs were still retrieved correctly. I also validated the data included in the report, and it appears to be accurate

Tuesday
I added additional information to the Guardian360 report and performed several tests to ensure that the correct data is being retrieved and displayed properly.

Wednesday
I added another graph to the Guardian360 report and attended a presentation from one of the SOC interns on the project's progress. Following the presentation, it was requested that the reports also be generated as PDF files.

Currently, the reports are generated in HTML, which is useful because they are interactive (e.g., displaying additional data when hovering over a chart). I began working on ensuring that the dynamic information is also rendered correctly in the PDF version. At this stage, the export to PDF still needs to be performed manually.

Thursday
I successfully implemented automated PDF generation for the **Guardian360 report**, in addition to the existing HTML version. During this process, I encountered several issues, such as certain graphs not rendering correctly in the PDF output, but I was able to resolve them and complete the implementation

Friday
I integrated the DeepL API into the reporting workflow to automatically translate log data, originally in English, into Dutch and French. When executing the script, it is now possible to choose in which language the report should be generated, per company.

Some parts of the report, such as the introduction, are still hardcoded. I began implementing automated translation for these static sections as well, but was unable to complete it today.

Although the team is running approximately one week behind schedule, significant progress has been made. Next week, we will begin working on the AI/automation component of the project. If time permits, I will finish implementing translation for the remaining static texts in the report.

Week 8 (14/04– 18/04)

Monday
We started the AI/automation component of the project. My task was to run Nginx and n8n in Docker containers and ensure that n8n could connect to Elasticsearch. This integration is required to enable log correlation as well as various automation and AI-driven features.

n8n is an open-source workflow tool that allows different apps, services, and APIs to be connected without traditional programming. Setting this up quickly was critical, as the rest of the team depends on it to perform their tasks.

Today, I deployed the Docker containers for Nginx and n8n and began testing the connection with Elasticsearch. Although the connection was not yet fully established, this was a solid first step in the process.

Tuesday
I continued working on the n8n setup. Together with the IT team, we also participated in a “war game” simulation, where we had to detect and defend against an attack on a client system. This exercise provided valuable practical insights into incident response.

Regarding n8n, the containers are running correctly, and n8n is able to log into the service with the proper credentials. However, it is not yet able to successfully retrieve or delete data from Elasticsearch.

Wednesday
Lasse provided feedback on the Guardian360 report, requiring a few small adjustments, which I noted for later implementation.

In addition, I started writing an Ansible playbook for n8n, allowing us to manage the entire setup as Infrastructure as Code (IaC) and deploy it directly to the production servers instead of running it only on my laptop.

At the end of the day, I also worked on improving the translations in the Guardian360 report.

Thursday
I completed the Ansible playbook for n8n, along with the Nginx configuration for the reverse proxy.

Lasse also asked me to assist with an issue in the Snipe-IT service, where the date was displayed incorrectly. I contributed to troubleshooting the problem.

n8n can now successfully query and modify data in Elasticsearch, so that part of the setup is finally functioning as intended.

In addition, I implemented further translations and made minor adjustments in the Guardian360 report based on yesterday’s feedback. I also assisted another intern with running my playbook.

Friday
I processed all feedback in the Guardian360 report and added extra elements, such as an improved cover page.

I also assisted an intern from Mechelen in setting up a connection between n8n and Elasticsearch, and advised another Mechelen intern on how to integrate AI agents into n8n.

Although the team has made substantial progress, we are somewhat behind the original planning made at the start of the internship.

Week 9 (21/04– 25/04)

Monday
Public holiday (Easter break) — no internship activities.

Tuesday
I created an AI-generated image to include in the report and worked on several improvements, such as adding a table of contents, implementing page numbering, and ensuring that the PDF version inserts page breaks at the correct points.

I also reviewed the visualizations created by other interns in Grafana. Lasse provided them with feedback, and I attended the session to better understand how to improve our own visualizations and to provide suggestions where relevant.

Finally, I attempted to remove the page number from the report's first page.

Wednesday
I again attempted to remove the page numbering from the first page of the report, but was unsuccessful, so I left it as is for now.

We also participated in a new version of the wargame simulation, where we had to detect and defend against an attack

In addition, I started developing a new automation in n8n that automatically sends emails when a certain number of security events occur within one of the Elasticsearch indices. For this, I used a test Outlook account to send the notifications.

Finally, an issue was reported in the report: the page numbers in the Vulnerabilities section were consistently off by one. I investigated and resolved the problem.

Thursday
Some employees receive phone notifications for certain events, but they also need a way to temporarily pause or reactivate these alerts. To address this, I started developing an automation workflow.

I created a Gmail account and linked it to an n8n workflow. When an email is sent to this account, n8n reads and filters the message, then executes actions based on the content. For example, it can determine whether notifications should be paused or reactivated. At this stage, I successfully configured the workflow to perform the correct actions based on email content.

I am currently awaiting management's feedback regarding the appropriate VocalNotify email address, which will define where notifications should be routed depending on the employee's request.

In addition, I created a Google Apps Script to automatically delete emails that n8n has already processed and that are older than one month.

Friday
I mainly used the day to prepare for the upcoming return session at school. I worked on the required presentation and supporting documents before leaving for campus in the afternoon.

Week 10 (28/04– 02/05)

Monday
I reviewed all the code for the reporting scripts and applied adjustments after being asked to generate the reports per company instead of per scan object.

In addition, I modified the logic so that the cover page and table of contents are only removed in the HTML version of the report, while remaining included in the PDF version.

Tuesday
I added bookmarks to the PDF report. These are clickable, correctly numbered, and lead to the corresponding sections. In the printable version of the report, I also added extra white space at the bottom of the pages.

In addition, I resolved an issue where the IP tables in the PDF report overlapped with other content. In the HTML version, the IP tables were not displayed correctly. I fixed this and added a scrollbar so the tables remain clear and readable.

Finally, I discussed with Lasse how the reports could be generated per probe of each company, instead of per company or per scan object. This is a newly requested adjustment. I already have an idea of how to implement it, but have not yet started development.

Wednesday
I gave a presentation on the progress our team has made over the past two weeks. The feedback was positive.

Together with Lasse, Bryan, and Tosin, we brainstormed on how to best implement the VocalNotify solution in n8n. Finally, I continued working on the Guardian360 report.

Thursday
Public holiday (May 1st) — no internship activities.

Friday
I corrected the page numbering in the table of contents of the Guardian360 report for sections spanning more than one page. I also adjusted the IP tables in both the HTML and PDF versions, and ensured that the page number on the cover page of the PDF version was removed.

In addition, I worked on modifying the report so that it is generated per scanner platform of a company, instead of per company.

Finally, I assisted Tosin with several issues in the CyberAlarm report, including page numbering and styling.

Week 11 (05/05– 09/05)

Monday
I completely reworked the Guardian360 script so that reports are now generated per scanner platform of each company, instead of per company or per probe. A scanner platform represents the physical location or network from which multiple probes are executed, whereas a probe refers to an individual scan object such as a host or IP. Linking reports to the correct scanner platform is therefore more accurate and meaningful.

Since this information was not included in the original log file, I used a second CSV file that maps scanner platforms to scan objects. By combining the two datasets, I was able to create the correct categorization in the reports.

Afterwards, I continued working on the VocalNotify email workflow in n8n. I had initially implemented this with Gmail, but it was later requested to migrate the workflow to Outlook. From my earlier research, I knew that this required an App ID and secret from an enterprise Azure account. I had already requested these credentials and was now able to begin implementing the workflow with Outlook.

Tuesday
I continued working on the VocalNotify workflow. I retrieved data from Elasticsearch and applied filtering based on type and criticality. This determines whether a call should be made to a Netcure employee.

In addition, a Thomas More Mechelen intern asked for help with Docker Compose, as his containers were unable to connect. I identified the error and explained the solution using a diagram, ensuring he also understood the underlying concept.

Finally, I re-ran the Guardian360 script with dummy data provided by Lasse in order to further test its functionality.

Wednesday
The VocalNotify workflow was fully developed and tested as requested. It now evaluates the number of logs, their criticality, and their type to determine whether a phone notification should be triggered.

Together with Tosin, I discussed where to retrieve the data that defines for which clients no calls should be made and to which employee the call should be routed. This information is stored in a Microsoft Lists page. I set up a Microsoft node in n8n and successfully authenticated it, but retrieving the data itself was not yet successful.

Thursday
I continued working on the Microsoft SharePoint integration. Some API calls were successful, while others failed. After further investigation, I determined that the linked Azure app required additional permissions to perform certain actions. I reported this issue, but it was ultimately decided to retrieve the data from an Excel file stored in SharePoint, rather than from Microsoft Lists.

Friday
I received a new log file with real data to test my reporting script. I further refined and improved the Guardian360 script. Since the new Guardian360 CSV log files had a slightly different format, I updated the script to be compatible with both the old and new versions.

I also worked on establishing the Excel integration via n8n, which was successful: I can now retrieve data from an Excel file stored in SharePoint.

Finally, I developed a script for WatchGuard360 to retrieve missing patches.

An important discussion also took place regarding the NDA. I requested clarification to ensure that I handle sensitive information in a correct and secure way, specifically regarding which information I am allowed to include in my portfolio, project plan, and realization document.

Week 12 (13/05– 16/05)

Monday
I further tested the WatchGuard script I created last week. In addition, I guided a Thomas More Mechelen intern on how to work with the server, demonstrating how to execute Linux commands via the terminal and how to get started with writing a script for WatchGuard.

Afterwards, I began working on a solution to automatically delete old documents (logs) from Elasticsearch once a certain threshold is reached. I conducted research on possible approaches, drafted a script, and performed initial tests on my laptop. I consider this script to be a crucial component of the SIEM, which is why I find it important to ensure it works correctly and is thoroughly tested.

Tuesday
I continued working on the script to automatically delete old logs. I created two versions: one implemented in n8n and another as a standalone script on the server. I then carried out several tests on both implementations.

Wednesday
I continued improving the log deletion script, focusing on adding extra functionality and running it directly on the server rather than through n8n.

In addition, I briefly worked on the Guardian360 report script to investigate why a specific element was being displayed at the bottom of each page in the PDF version.

The log deletion script was completed and tested locally. I was asked to also test it on the server before applying it across all types of logs.

Thursday
I tested the log deletion script on the server and added a few additional functionalities. I also briefly worked on the Guardian360 script to resolve a visual issue.

Friday
I implemented further improvements to the log deletion script and carried out additional tests. I then executed the script on the server, where it performed as expected and successfully removed most of the old logs.

For certain indices, however, the script needed to be executed multiple times in order to delete all outdated logs. I conducted research to investigate the cause of this issue.

Week 13 (19/05 – 23/05)

Monday
I continued investigating why the log deletion script was not functioning correctly. Identifying the cause was challenging, but I was ultimately able to resolve the issue. The script has now been finalized and works as intended.

In addition, I worked on a redacted version of the Guardian360 report so that it can be safely included in my portfolio.

Tuesday
I began researching MISP and TheHive, including completing the related TryHackMe rooms. I also attempted to set up TheHive locally on my computer in order to test it before deploying it to the production server.

In addition, I studied how TheHive works, its available features, and how to optimize its performance on the server. I also explored how TheHive can best communicate with other services running on separate servers.

Wednesday
In the morning, I continued researching TheHive and started its initial implementation.

In the afternoon, I attended CyberSec Europe together with my internship colleagues. It was a highly interesting experience and provided valuable insights into the latest developments in cybersecurity.

Thursday
I continued setting up TheHive and created an Ansible playbook to automate the installation as Infrastructure as Code.

According to TheHive's documentation, containers for Nginx and Elasticsearch are also required. Since these were already running for other services, I configured TheHive to use the existing containers instead of deploying new ones. This made the setup more complex but ensured better integration with the current environment.

Friday
Before the official internship hours, I collected the office key from Tosin at Zaventem station, as he was absent during the first part of the day.

I then continued working on the TheHive implementation. I encountered issues while running the playbook, so I decided to proceed with a manual installation via the CLI. With a few configuration changes and support from Bryan, I was able to complete the setup successfully.

I was glad to finish this implementation today, as it marked the final day of my internship.

3. Conclusion

This logbook illustrates the wide range of experiences and skills I acquired throughout my internship at Netcure. From scripting and automation to infrastructure as code and incident response, I had the opportunity to work with relevant technologies and apply them in a professional SOC environment.

In addition to technical growth, I gained valuable **professional experience** by collaborating with colleagues, assisting fellow interns, and presenting progress to supervisors. These interactions strengthened my ability to work in a team, communicate effectively, and adapt to the needs of a real-world security operations context.

Overall, this internship allowed me to combine my academic knowledge with practical application, while contributing to meaningful projects in cybersecurity and automation.