EXCELLIUM

Cyber Solutions by Thales

Intrusion & Application Security

Internships 2023-2024

Latest update: 2023-09-12

Discover all our internships on our website.

https://excellium-services.com/internships/

SCAN ME

Contents

1 Introduction

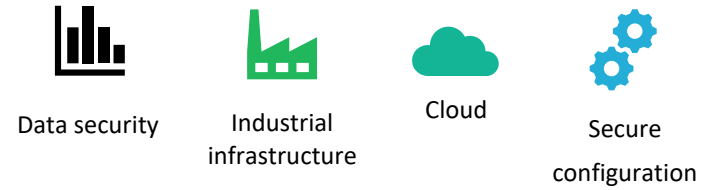
1.1 Excellium Cyber Solutions by Thales

Excellium Cyber Solutions by Thales is a cyber-security consulting and technology Integration Company established since 2012 in Luxemburg and Belgium, with activities worldwide. The group employs over 200 people acting exclusively in the information security domain.

1.2 What can we do?

1.2.1 Advisory Services

1.2.2 Cybersecurity Infrastructure



1.2.3 Managed Detection And Response

Cyber Security Operation Center
CSOC

- SOC as a Service
- SOC customer
- 24/7 monitoring
- Vulnerability Scans
- DDoS testing

Computer Security Incident Response
CERT-XLM

- EDR as a Service
- Threat Intelligence detection and response
- DFIR
- Vulnerability Management

1.2.4 Test, Adapt And Prevent

- Red teaming
- Penetration Testing
- Application testing/code review
- Threat Intelligence Reports

2 Intrusion & Application Security Team

The team has two different practices with people collaborating together.

Intrusion

Penetration testing & red teaming

Application Security

Vulnerability assessments and integration of the security in software development

Structured catalogue of services

- Over 20 types of security testing packages
- Penetration testing of simple applications to large cloud environments
- Large skill set that can be deployed during Red Team engagements
- Attack surface exposure though actual testing

Security experts combining manual and automated testing

- Trusted and certified consultants with proven experience
- Collaborative and fully ethical approach
- From black-box testing to white-box testing with secure code reviews

Isolated tests or global assessment programme

- Individual testing to close security gaps for specific perimeters
- As a managed service, from identification of assets to remediation follow-up

Continuous testing capabilities

- Recurrent penetration tests
- Automated scanning and assisted penetration testing

Certified and screened professionals

- Certifications from OffSec and GIAC (SANS)

Professional tooling

- Burp Professional, Cobalt Strike…
- Collaboration, methodology and reporting: Dradis

Field experience

- Hundreds of assessments per year
- Accredited Red Team (TIBER) provider
- Clients in all industries
  (banking and insurance, energy,
  transportation, healthcare, education…)

In numbers for 2022:

- 10 000+ hours of penetration tests, AppSec and client support
- 200 assessments with 65 clients in various activity sectors
- 10 000+ people tricked by our phishing and vishing campaigns
- 8 technical blog posts
  - https://excellium-services.com/blog/
- 23 CVE published (more than 100 since 2015):
  - https://excellium-services.com/cert-xlm-advisory/

3 Internships

3.1 Introduction

Are you passionate about cybersecurity and eager to make a significant impact in the world of penetration testing and/or application security?

We offer exciting internship opportunities where you'll immerse yourself in team of experienced consultants.

Have a look to our list internship topics you can choose from. 😉

Position: IT Security Intern 🦁🦁

Duration: 4 to 6 months

Locations

Excellium Services S.A.

5 rue Goell

L-5326 Contern, **Luxembourg**

Excellium Services Belgium N.V.

Orion Bldg, Belgicastraat 13

B-1930 Zaventem, **Belgium**

Send your application to a unique address for Belgium and Luxembourg:

recruitment@excellium-services.lu

3.2 Recruitment process

1) **Pick your 3 favourite internships subjects** that you find the most relevant with your experience and your motivation, you can also BYOS, or "Bring Your Own Subject". 😉

2) **Contact our Human Resources department** (either in Belgium or Luxembourg). Send them your CV, tell them about the subjects you like and why you think you would be a good candidate for them. Give us also details about the expected duration of your internship and the period. The interview could have parts in French, English and Dutch, depending on the languages indicated in your CV.

   Our priority is to onboard our trainees as full time employees in the team after their internship. However, we are open to propose our trainings for students that are in the middle of their scholarship (if it's not their final internship).

3) **Solve our challenge.** We will give you a small challenge to solve. We estimate that the challenge requires a few hours to solve it and automate the exploitation steps. We are open to give you hints if you are stuck with the challenge. Let us know what you tried and we will be happy to help and check that everything is working properly.

4) **Send us a small report (or "write-up"), redacted in English**, that describes your approach to solve the challenge. Depending on the challenge, we may also request recommendations or remediation steps. This will be clearly stated if this is the case.

5) **Do a technical interview with the Team Leader or the Team Leader's deputy.** This is last step of our recruitment process. We will discuss with you during about 1h00 and 1h30 about your school, personal and professional experiences. We are interested by what makes you motivated and passionate about IT and IT security. Tell us about any personal project (including personal sites, scripting, tools, "IT @ Home", home automation…), contribution to open source projects, participation in CTF (either as a player or challenge creator…), participation in IT Security groups, bug bounties, CVE…

6) **We will review the different applications and propose an internship to the best candidates.** First applications (and first resolutions of the challenge) will be considered by priority.
   We may also propose internships for our other departments as alternative if we cannot accommodate all candidates. All our internships at Excellium are exclusively related to IT Security.

4 Subjects

4.1 Advanced phishing techniques

📖 Description

This internship will focus on perfectioning phishing scenarios and tools used during phishing exercises. In a second phase, the focus will be to develop less known and newer phishing techniques.

Phishing exercises are used to raise awareness about cybersecurity threats within an organization. Phishing attacks remain one of the most common and successful methods employed by cybercriminals to infiltrate systems and compromise data.

Penetration testers also rely on phishing attack for initial foothold during red team engagement, for example.

However, the old-fashioned malware.exe and fake login page aren't that reliable as they used to be, organization are catching up quickly. Attackers are evolving too. Plenty of new techniques are being discovered to trick both email gateway protections and employees.

🎯 Objectives

- Phishing Simulation Development: Create realistic and evolving phishing simulations that mimic current cyber threats and tactics.
- Study newer phishing tactics some examples:
    - Browser In The Browser (BITB) Attacks
        - Chromium App mode & fake address bar
        - noVNC 2FA stealer
    - ClickOnce Phishing
    - Double-bounced attacks with email spoofing
    - Advanced fingerprinting
        - How much information can you get on a single click?
    - Dynamic Device Code Phishing
- Create advanced phishing package(s) ready to be used based on your studies.
- Data Analysis: Analyze the results of phishing exercises to identify trends, vulnerabilities, and areas for improvement.

💻 Skills Involved

- **Web Development**: Building realistic phishing websites and email templates.
- **Scripting**: Being able to write script to automate deployment/launch phishing campaign
- **Cybersecurity Awareness**: Understanding the latest phishing techniques and tactics.

References

- Phishing With Chromium's Application Mode
- Browser In The Browser (BITB) Attack
- Double-bounced attacks with email spoofing – 2022 trends
- Black hat - CLICKONCE AND YOU'RE IN
- File Archiver In The Browser
- Dynamic Device Code Phishing

4.2 BadUSB payload creation and rogue devices for Physical Intrusion & Red Teaming

📖 Description

A "BadUSB" is a malicious device used to perform harmful actions without user consent, often involving keystroke injection or compromising system security. It poses a significant security threat as it can execute attacks without user interaction or detection.

This internship opportunity focus on gaining hands-on experience in developing various BadUSB payloads and prepare rogue devices for red team engagement and physical intrusion.

🎯 Objectives

During this internship, you will:

- Creation of DIY BadUSB
- Remote controlled BadUSB
- Learn to create custom BadUSB payloads capable of compromising target systems (with and without user interaction)
- Find innovative ways to masquerade BadUSB devices.
- Preparing Raspberry/Network implant controlled remotely.
- Collaborate with our red teaming experts to simulate real-world physical intrusion scenarios.
- Assess and document the effectiveness of physical security controls.
- Contribute to the development of innovative attack techniques and tools.

💻 Skills Involved

- **Programming**: Proficiency in scripting languages like Python, Bash and C++ (Arduino)
- **Hardware Hacking**: Familiarity with hardware components and soldering skills.
- **Cybersecurity Knowledge**: Basic understanding of cybersecurity principles and attack vectors.
- **Operating Systems**: Experience with Linux and Windows environments.

References

- Vincentyiu – Red team hardware & gadgets- USBNinja
- Turning a Regular USB into USB Rubber Ducky
- Owning the Network with BadUSB
- ZDnet - Rare BadUSB attack detected in the wild against US hospitality provider

4.3 Creation of a tool to analyze the behavior of JavaScript libraries obtained via NPM

📖 Description

Recent applications based on JavaScript technology use the NPM "Node Package Manager" system to manage project dependencies. The modules used regularly come from the Internet and can therefore potentially contain malicious code.

🎯 Objectives

The aim of this research is to design and implement a tool to analyze a package from NPM to detect strange behavior (e.g.: data exfiltration to a third-party site, addition of unsolicited listeners, etc.).

The tool will be based on existing Open-Source technology bricks.

💻 Skills Involved

- Web technologies.
- HTTP protocol 1.1 and 2.0
- NPM and containerization techniques (e.g. Docker).

References

- nmpjs
- SauceLabs – Headless Browser Testing 101

4.4 Using package manager to identify applications affected by a vulnerable component

📖 Description

Modern applications are almost all based on external components (also known as dependencies), most of which are open source. The Log4Shell vulnerability showed just how critical a vulnerability in a component used by multiple projects can be, and how it can jeopardize a company for a long time to come.

🎯 Objectives

The aim of this study is to analyze how a package manager, such as "Nexus" or "Artifactory", can be used to quickly identify which applications are impacted when a critical vulnerability is discovered on an external component.

💻 Skills Involved

- Package management.
- Web APIs.
- Software Development.

References

- Sonartype
- JFrog
- Log4Shell – Wikipedia
- Lunasec blog – Log4j 0-day
- righettod's GitHub – log4shell analysis

4.5 The latest cryptographic standards for application security

📖 Description

Cryptography is omnipresent in modern applications. The correct use of cryptographic standards (signature, encryption, fingerprinting, key derivation...) remains a complex subject, and it is very easy to introduce vulnerabilities during implementation.

🎯 Objectives

The aim of this research is to identify and study new cryptographic algorithms in order to define a list of algorithms whose use is to be preferred and the appropriate mode of use.

Depending on the context, the student will be able to demonstrate, via Proof Of Concepts, examples of attacks justifying the use of the algorithms recommended in the results of the study.

💻 Skills Involved

- Cryptography

References

- Google Blog - Introducing Tink Cryptography
- Google Tink
- https://www.keylength.com/
- https://www.nist.gov/topics/cryptography

4.6 Mobile Penetration Testing and Security Verification Standard

📖 Description

Mobile security mainly revolves around protecting data. Apps on your smartphone store a lot of personal and sensitive information, so if someone compromises your phone, they get access to your whole life. This becomes even more important as mobile malware becomes more common.

To ensure mobile app security, it's crucial that apps handle, store, and protect sensitive information correctly. Modern operating systems have tools for this, but they must be used properly.

As a Mobile Application Security Intern, you will have the opportunity to choose your focus between Android or iOS applications. You will gain in-depth knowledge in OWASP Mobile Application Security Verification Standard (MASVS).

You will learn how to assess these points using OWASP testing guide (MASTG).

🎯 Objectives

- Get familiar with Mobile penetration testing methodology and the Security Verification Standard and understand which points are essential, advanced security, hardening measures users or user privacy related (L1, L2, R, P)
- Experiment with existing mobile security tooling (Frida, Objection, Drozer, GrapeFruit, MobSF, …)
- Create or contribute to one open-source tool(s) focusing on mobile security and mobile penetration testing.
- Create a Mobile Application Vulnerability Assessment report template based on MASTG v2 Atomic test
- **Optional**: Develop a vulnerable mobile application as a Capture The Flag (CTF)-like challenge with original vulnerabilities based on your studies.

💻 Skills Involved

- Mobile Penetration Testing: Understand how mobile application security can be assessed.
- Scripting: Python/Bash or any scripting language for security testing tool development
- Basic understanding of mobile application development (Android and/or iOS).

References

- Mobile Application Security with OWASP
- Mobile Application Security Verification Standard
- Vulnerable Mobile applications
- HackTricks - Android Application Pentesting
- HackTricks- iOS Application Pentesting
- Frida iOS playground

4.7 Cloud deployment automation for Red Team engagement

📖 Description

Red team infrastructure is composed multiple cloud components. For example, a typical Red Team cloud infrastructure can include:

- **Phishing Server**: This component is utilized to craft and execute phishing campaigns.
- **Payload Server**: Used to store payload code for use in various attacks.
- **Command and Control (C2) Team Server:** The central hub for communication and orchestration among Red Team operators, allows coordination and control during operations.
- **Redirectors**: These serve as intermediaries for mail, HTTPS, and DNS traffic, functioning as proxies that shield critical assets while facilitating secure redirection of incoming data.
- **Vault warden**: An optional element offering a secure data storage solution for the engagement's duration.
- **Red ELK:** This component essentially acts as a Security Operations Center (SOC) tailored for Red Teams. It aids in monitoring the overall infrastructure, enhancing situational awareness, and detecting any intrusion attempts by the Blue Team into your assets.

Setting up all the components manually for every new engagement can be very time consuming.

That's where infrastructure as code (IAC) and tools like Ansible and Terraform come into play. This internship offers you the chance to dive into the world of cybersecurity and automation, combining your scripting skills with cloud infrastructure management and IAC techniques.

🎯 Objectives

- Gain hands-on experience in using Ansible and Terraform to automate the deployment of a Red Team infrastructure. Explore different cloud providers and understand their offerings to ensure efficient and secure infrastructure provisioning.
- Document your work thoroughly, creating comprehensive guides that detail the steps taken, configurations used, and challenges overcome during the automation process.

💻 Skills Involved

- **Scripting**: Develop scripts and automation routines using Python, Bash, or other scripting languages to streamline infrastructure deployment.
- **Cloud Infrastructure**: Learn how to navigate cloud platforms (e.g., AWS, Azure, GCP) and understand their services for optimal resource allocation.
- **Infrastructure as Code (IAC):** Explore the principles of IAC and how it enables the efficient management of infrastructure through code.
- **Ansible/Terraform**: Become proficient in Ansible and Terraform, two powerful IAC tools used for configuration management and infrastructure provisioning.

References

- SSE blog - Building a Red Team infrastructure in 2023
- Red Teamer tips - Automating Red Team Infrastructure with ansible
- Ansible – Ansible and AWS documentation
- Rastamouse – Infrastructure As Code with Terraform and Ansible
- Praetorian's approach to Red Team Infrastructure

4.8 Security Issues in Large Language Model (LLM)

📖 Description

Large Language models (LLMs) have revolutionized various aspects of enterprise environments, from generating text and translating languages to crafting creative content. The integration of LLMs into applications offers incredible capabilities, but it also introduces critical security challenges. Even without direct compromise of connected applications, LLMs can become attractive targets for attackers.

This internship explores the potential vulnerabilities and attack vectors associated with LLMs and aims to build a strong understanding of their security implications.

🎯 Objectives

- **Explore LLM Vulnerabilities**:
  - Remote control of LLMs
  - Leaking/exfiltrating user data
  - Persistent compromise across sessions
  - Spread injections to other LLMs
  - Compromising LLMs with tiny multi-stage payloads
  - Automated Social Engineering
  - Targeting code completion engines
- Define an approach/testing methodology
- **Build a CTF-Like Challenge**: Create a Capture The Flag (CTF)-style challenge that simulates real-world scenarios involving vulnerable LLM integrations. This challenge will provide an opportunity for hands-on learning and practical experience in securing LLMs.

💻 Skills Involved

- Web application penetration testing
- LLM Integration
- Web development

References

- OWASP Top 10 for Large Language Model
- greshake's github – LLM security

4.9 Identifying and preventing vulnerabilities in Electron apps

📖 Description

ElectronsJS is a framework that allows to create cross-platform desktop application. One of the particularities of Electron is that it embeds Chromium browser and NodeJS in desktop applications.

This ability to create desktop applications using web technologies like HTML, CSS, and JavaScript makes it a popular option among developers and companies nowadays.

However, using web-like technologies also implies that some known web vulnerabilities can affect Electrons apps on top of other weaknesses.

Your mission will be to identify, analyze, and prevent vulnerabilities in Electron applications. By the end of this internship, you will have developed a robust methodology for penetration testing and be equipped to provide valuable recommendations for patching and securing Electron applications during security assessments.

🎯 Objectives

- Identify and study some of the known vulnerabilities and attack vectors on Electron Apps.
- Create a vulnerable Electron Sandbox application as a practical demonstration of these vulnerabilities. This hands-on experience will help you gain a deeper understanding of potential threats.
- Perform comprehensive security assessments on Electron applications, using the knowledge and insights gained during the internship to identify vulnerabilities and weaknesses.
- Identify actionable recommendations for preventing and mitigating vulnerabilities in Electron applications. Establish best practices for secure Electron app development.

💻 Skills Involved

- **Web Development**: Electron applications are built using web technologies. This includes proficiency in HTML, CSS, and JavaScript.
- **JavaScript/Node.js**: Given that Electron embeds Node.js, understanding of JavaScript and Node.js is crucial. Y
- **Code Review:** Conduct code reviews to identifying vulnerabilities. You'll need to scrutinize an application's source code, looking for common security issues such as injection attacks, access control problems, and data validation errors.
- **Penetration Testing**: Key skill for evaluating the security of Electron applications. This involves actively attempting to exploit vulnerabilities and weaknesses, simulating real-world attacks to uncover potential risks.

References

- ElectronJS web's site
- DoyenSec – Democratizing Electron Security
- Discord Desktop Remote Code Execution
- Facebook Messenger Desktop App Arbitrary File Read
- Electronegativity – Electron security tool

4.10 Canary platform and token deployment automation

📖 Description

A Canary Token is a tool that acts as a decoy to detect and alert when someone interacts with it. It's like a digital tripwire that signals potential unauthorized access or suspicious activity in a network or system. The alerts can be sent to a variety of sources, including, emails webhook, system logs and so on.

They can be strategically placed to mimic sensitive data, applications, or systems, effectively luring potential threats into revealing themselves.

For instance, instead of employing actual malicious files, penetration testers can use legitimate files embedded with a Canary Token. This tactic enables them to determine precisely when a file is accessed or modified, providing concrete evidence of user interaction with potentially harmful elements.

🎯 Objectives

During this internship, you will:

- Experiment with the use of Canary Tokens**:** Gain hands-on experience in deploying and managing Canary Tokens in different network and system environments. This includes understanding their placement, configuration, and potential use cases.
- Develop tool or interface that facilitate the deployment, monitoring, and management of Canary Tokens.
- Automate deployment of multiple tokens
- Create DNS token with custom domain names.
- Develop the capability to trigger different types of events based on Canary Token interactions. This might include alerts, notifications, or automated responses tailored to specific scenarios.
- Monitoring of active tokens

💻 Skills Involved

- **Web Development:** Proficiency in front-end and back-end web development is crucial for creating user-friendly interfaces and the backend logic for managing Canary Tokens.
- **Scripting and Automation**: automating the deployment and monitoring of Canary Tokens, improving efficiency and scalability.
- **Webhook integration:** For alerting and reporting functionality.

References

- GitHub OpenCanary
- How do I create a custom Canarytokens domain?
- Generate CanaryToken

4.11 Exploring the attack surface of Azure AD

📖 Description

Azure AD serves as an identity management platform for Microsoft Applications. It plays a crucial role in authenticating and authorizing users, enabling secure access to various Microsoft services, applications, and Azure resources.

In this task, you'll delve into understanding the potential vulnerabilities and attack vectors associated with Azure AD from

🎯 Objectives

- **Experiment with Azure Active Directory:**
  - Gain hands-on experience with Azure AD, exploring its configuration and settings.
  - Understand how Azure AD integrates with Microsoft applications and Azure Resource Manager.
  - Explore user management, authentication, and authorization within Azure AD.

- Explore Weaknesses in Azure AD. For example:
  - Exploiting Password Hash Synchronization:
    - Password hash synchronization (PHS) is a feature that synchronizes password hashes from an on-premises Active Directory to Azure AD. Understand how this process works.
    - Experiment with scenarios where PHS may be exploited, such as brute-force attacks, password spraying, or other password-related vulnerabilities.
  - AD Connect Vulnerabilities:
  - AD Connect is a tool used to synchronize on-premises Active Directory with Azure AD. Identify potential vulnerabilities in the AD Connect configuration.
  - Experiment with common misconfigurations or insecure settings in AD Connect that could lead to security risks.
  - Investigate how attackers could abuse AD Connect vulnerabilities for privilege escalation or unauthorized access.

💻 Skills Involved

Azure: Proficiency in Azure services and how they interact with Azure AD and understanding security features and how to configure them.

Penetration Testing: Knowledge of penetration testing methodologies, tools, and techniques. Ability to simulate real-world attacks on Azure AD to identify vulnerabilities.

Active Directory (AD): Understanding of on-premises Active Directory and its integration with Azure AD. Knowledge of AD security best practices.

References

- https://github.com/swisskyrepo/PayloadsAllTheThings/blob/master/Methodology%20and%20Resources/Cloud%20-%20Azure%20Pentest.md
- https://www.synacktiv.com/en/publications/azure-ad-introduction-for-red-teamers.html
- https://blog.sygnia.co/guarding-the-bridge-new-attack-vectors-in-azure-ad-connect

4.12 Attack surface of AWS

📖 Description

In this internship, you will be tasked with conducting a comprehensive analysis of the attack surface within Amazon Web Services (AWS).

🎯 Objectives

You will explore attack's scenario on AWS involving leak of developer credentials for an AWS environment. Your responsibility will be to identify potential weaknesses in the AWS infrastructure.

- Identify security vulnerabilities, misconfigurations, and weaknesses within AWS resources and services.
- Evaluate the potential impact of the identified vulnerabilities on the overall security posture of the AWS infrastructure.
- Document your findings, assessments, and recommendations comprehensively.
- Outlines the common vulnerabilities in AWS, their potential impact, and steps to remediate them.

💻 Skills Involved

Penetration Testing: Knowledge of penetration testing methodologies, tools, and techniques. Ability to simulate real-world attacks on AWS to identify vulnerabilities**.**

AWS Expertise: Knowledge of Amazon Web Services, including various AWS services, IAM (Identity and Access Management), EC2 (Elastic Compute Cloud), S3 (Simple Storage Service), RDS (Relational Database Service), etc.

AWS Security: Understanding of AWS security best practices and configurations.

References

- HackerOne - Penetration Testing on AWS: A Practical Guide
- HackTricks Cloud – AWS Security

4.13 Explore using LLMs for secure code review and assessment

📖 Description

With the ever-growing complexity of software systems and the continuous emergence of security threats, AI has emerged as a potentially valuable tool for enhancing the efficiency and effectiveness of security testing, including code reviews.

Manual code analysis, performed by security consultant is a long and tedious task. There is only a limited amount of line a human can read and analyze per day.

In this context, LLMs have shown promise in automating and augmenting code review processes:

The potential benefits of using LLMs for secure code review include the ability to analyze large volumes of code at a speed that surpasses human capabilities, detect known vulnerabilities, identify code patterns indicative of security issues, and offer recommendations for remediation.

On the other side, relying solely on LLMs for secure code review raise other limitation such as the lack of Contextual Understanding, false positive, limited to well-known vulnerabilities, …

During this internship, the student will experiment the potential and limitation of using LLM's in a security context and identify what are some of the best use cases.

🎯 Objectives

- Study possibilities of using Artificial Intelligence in both static code review and dynamic security testing.
- Search LLM model candidates capable of processing codes and/or tool output. Analyze and rate the result.
- Experiment and compare existing security tools that are using LLM.
- Fine-tune existing model and/or train your own model to identify vulnerabilities in code.
- Experiment Integrating LLM in a security assessment workflow by processing tool outputs and drive security scans.
- List use-cases where LLM's performing best

💻 Skills Involved

- Penetration testing
- LLM / Machine learning
- Scripting

References

- https://research.nccgroup.com/2023/02/09/security-code-review-with-chatgpt/
- https://semgrep.dev/blog/2023/using-ai-to-write-secure-code-with-semgrep/
- https://betterprogramming.pub/i-used-gpt-3-to-find-213-security-vulnerabilities-in-a-single-codebase-cc3870ba9411
- https://github.blog/2022-02-17-code-scanning-finds-vulnerabilities-using-machine-learning/
- https://github.com/GreyDGL/PentestGPT

4.14 Your choice

💭 You have another topic in mind related to penetration testing, red teaming or application security and would like to explore it here at Excellium?

Feel free to tell us about and we'll see what we can do!

5 Career Opportunities

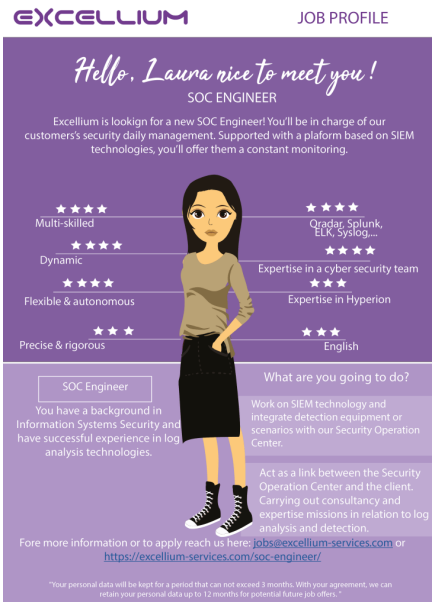
We offer opportunities for growth and development.

Your internship at Excellium can lead you to a job offer.

This is a chance to join an established consulting firm and work with leaders in your field to develop insight, experience and truly add value.

If you are striving to be the best, we want you!

Have a look at some of our job opportunities at Excellium?



More career opportunities at https://excellium-services.com/career-opportunities/