Dear Intern

Project report is an inherent component of your internship. We are enclosing a reference table of content for the project report. Depending on the internship project (IT/Non-IT, Technical/Business Domain), you may choose to include or exclude or rename sections from the table of content mentioned below. You can also add additional sections. The key objective of this report is for you to systemically document the project work done.

|  |  |
| --- | --- |
| Internship Project Title | TCS iON RIO-125: Secured Docker Based Lab – Enforcing End-to-End Security |
| Name of the Company | Tata Tcs iON |
| Name of the Industry Mentor |  |
| Name of the Institute | Amity University Online |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Start Date | End Date | Total Effort (hrs.) | Project Environment | Tools used |
| 14-06-2025 | 27-07-2025 | 125 hrs | AWS EC2 Ubuntu (Cloud), Docker, Linux Shell | Docker, Docker Compose, Nmap, Metasploit, Hydra, Nessus, Traceroute, John the Ripper, Python, Java, MySQL, Apache2, GitHub |

**TABLE OF CONTENT**

* Acknowledgements
* Objective
* Introduction / Description of Internship
* Internship Activities
* Approach / Methodology
* Assumptions
* Exceptions / Exclusions
* Charts, Table, Diagrams
* Algorithms
* Challenges & Opportunities
* Risk Vs Reward
* Reflections on the Internship
* Recommendations
* Outcome / Conclusion
* Enhancement Scope
* Link to code and executable file
* Research questions and responses

## 1. Acknowledgements

I express my sincere gratitude to **TCS iON**, the **industry mentors**, and **Amity University Online** for providing me this valuable opportunity. I thank all those who guided and supported me throughout the project.

## 2. Objective

To design and deploy a secure Docker-based lab environment on the cloud (AWS) and perform security assessments using popular penetration testing tools, ensuring end-to-end protection of web and database services.

## 3. Introduction / Description of Internship

This internship focused on cybersecurity and DevOps integration using cloud infrastructure. The project involved creating an isolated Docker lab with static IPs, service configuration, and performing end-to-end vulnerability scanning using Metasploit, Nmap, Hydra, etc.

## Internship Activities

|  |  |
| --- | --- |
| **WEEK** | **ACTIVITY SUMMARY** |
| 1 | AWS EC2 setup, IAM user, Docker installation |
| 2 | Docker network creation, OS and services setup |
| 3 | Vulnerability scanning using traceroute, nmap, metasploit |
| 4 | Password brute-forcing (Hydra), final testing and enhancements |
| 5 | Documentation, GitHub push, video demo |

## 5. Approach / Methodology

* Provision EC2 (Ubuntu) in AWS
* Create secure Docker container with Apache, MySQL, Java, Python
* Assign private network with static IP
* Use penetration testing tools to evaluate vulnerabilities
* Document findings and implement minor fixes/enhancements

## 6. Assumptions

* All tools are compatible with Ubuntu 22.04
* Docker networking works correctly on EC2
* Security tools are pre-tested in isolated lab environment

## 7. Exceptions / Exclusions

* GUI-based tools (e.g., Wireshark) not deeply integrated due to EC2 limitations
* Nessus installation skipped due to setup complexity and time constraint
* Encrypted data-at-rest implementation not included

## 8. Charts, Tables, Diagrams

### Vulnerability Scan Results:

|  |  |  |  |
| --- | --- | --- | --- |
| **TOOL** | **TARGET IP** | **PORT/SERVICE** | **RESULT** |
| Nmap | 192.168.50.10 | 80, 3306 | Scan |
| Metasploit | 192.168.50.10 | Full TCP Scan | Success |
| Hydra | MySQL Login | root@192.168.50.10 | Failed (as expected) |
| Traceroute | Internal IP | Network path | 1 hop |

## 9. Algorithms

No custom algorithms were used. However, logic flow was followed to:

* Provision infrastructure
* Create Docker container
* Expose services with security
* Test externally using standard tools

## 10. Challenges & Opportunities

### Challenges:

* Docker networking and static IP configuration
* Installing penetration tools with dependencies
* Rockyou.txt setup and Hydra connection issues

### Opportunities:

* Hands-on with AWS & Docker security
* Deep understanding of DevSecOps pipeline
* Real-world penetration testing basics

## Risk Vs Reward

|  |  |  |
| --- | --- | --- |
| **RISK** | **MITIGATION** | **REWARD** |
| EC2 exposure to public attacks | Used private IP subnet | Hands-on AWS experience |
| Docker misconfiguration | Used official base images | Secure multi-service container |
| Tool crashes (Hydra, Nessus) | Used alternate tools | Adaptability in testing |

## 12. Reflections on the Internship

This internship was a significant learning experience in the field of cybersecurity and DevOps. It gave me real-world exposure to cloud deployment, system hardening, and security scanning using industry-standard tools.

## 13. Recommendations

1. Provide pre-built images for tools like Nessus
2. Automate Docker security patches
3. Add Jenkins/GitLab CI for security as code

## 14. Outcome / Conclusion

Successfully built and tested a secure Docker lab environment. Services were validated, scanned, and hardened. Full project delivered with documentation, GitHub repo, and a 5-minute demo video.

## 15. Enhancement Scope

1. Include container encryption
2. Add centralized logging with ELK Stack
3. Automate vulnerability scans on a schedule

## Link to Code and Executable File

GitHub: <https://github.com/Afra-Analytics-Dev/Tcs-ion-125-Project.git>

Video: <https://drive.google.com/file/d/1lfCBxIlcz-l3EOe15kAURd4y_lUGiAKU/view?usp=drive_link>

## 17. Research Questions and Responses

**1.What is the role of Docker in DevOps security?**  
Docker helps isolate services, manage dependencies, and reduce attack surface with micro-segmentation.

**2.Why are static IPs important in lab testing?**  
They ensure consistent targeting during vulnerability scanning.

**3.Can brute-force tools like Hydra be prevented?**  
Yes, by enabling fail2ban, complex passwords, and limiting login attempts.