* The final project deadline is **Aug15, 2025** and in case you miss the deadline, you will have time until **Aug17 midnigh**t to submit the report
* ALL the group members need to submit **(INDIVIDUAL SUBMISSIONS)**
* The presentations will take place on 2 days in the final week
* ***Please ping in the malware group to confirm which groups would want to present on which day***
* For the final report, please note that you are supposed to present the same like a Malware Analyst.
* The report should include the findings from an aftermath of a breach. (Title page, Table of Contents, Executive Summary, Static/Dynamic analysis findings with evidence, your hypothesis on how the attack would have happened; what measures the company needs to take to omit them, security rating before and after your recommendations, and recommendations need to be case specific and not generic and your final draw/conclusion
* For the presentation, it can be from 10 mins to 20 mins (max) and be presentable on that day as you will be presenting as a security professional.
* There will be QnA after the presentation.
* Demo: You are to present a live demo of detonating and analyzing your sample live. Be prepared with the recording of the same during your practice run to avoid any hiccups during the presentation in case, the live attack does not work.

Feel free to ping me; should you have any more queries.

The final report should include:

## **1. Executive Summary**

* High-level overview of the malware.

**Content**:

* + Brief description of the malware (e.g., type, name, or identifier).
  + Key findings (e.g., primary purpose, impact, and severity).
  + Summary of infection vector (e.g., phishing, exploit kit, etc.).
  + High-level recommendations (e.g., mitigation strategies, containment).

*Audience*: Managers, decision-makers.

## **2. Introduction**

* **Purpose**: Set the context for the analysis.
* **Content**:
  + Background of the incident (e.g., how the malware was discovered).
  + Scope of the analysis (e.g., static, dynamic, or reverse engineering).
  + Objectives of the report (e.g., understanding behavior, aiding remediation).
  + Date and time of analysis.
  + Tools and methodologies used (e.g., IDA Pro, Wireshark, sandbox environments).

## **3. Malware Overview**

* **Purpose**: Provide basic details about the malware sample.
* **Content**:
  + **Sample Information**:
    - File name, size, and type (e.g., executable, DLL, script).
    - Hash values (MD5, SHA-1, SHA-256).
    - Date of first submission to public repositories (e.g., VirusTotal).
    - Source of the sample (e.g., email attachment, URL, sandbox dump).
    - Compile time (if available).
  + **Malware Classification**:
    - Type (e.g., trojan, ransomware, worm, spyware).
    - Family or variant (if known, e.g., Emotet, WannaCry).
    - Any known aliases or detection names (e.g., from AV vendors).
  + **Initial Indicators**:
    - File metadata (e.g., compilation timestamp, digital signatures).
    - Packing or obfuscation techniques (e.g., UPX, custom encryption).

Details of the sample(s) you analyzed:

## **4. Analysis Environment**

* + Document the controlled environment where analysis was performed: **Environment Setup**:
    - Sandbox or virtual machine configuration (e.g., Windows 10, 64-bit).
    - Monitoring tools (e.g., Process Monitor, ProcDot).

## **5. Infection Vector and Delivery Mechanism**

* **Purpose**: Describe how the malware enters and spreads within a system or network.
* **Content**:
  + Initial infection method (e.g., malicious email attachment, drive-by download).
  + Exploitation techniques (e.g., CVE-2021-1234, social engineering).
  + Propagation mechanisms (e.g., network shares, removable media).
  + Associated campaigns or threat actors (if known).

## **6. Static Analysis**

* **Purpose**: Detail findings from examining the malware without execution.
* **Content**:
  + **File Structure**:
    - Headers and sections (e.g., PE headers for Windows executables).
    - Anomalies (e.g., corrupted headers, unusual section sizes).
  + **Strings Analysis**:
    - Extracted strings (e.g., URLs, file paths, registry keys).
    - Encoded or obfuscated strings and decoding methods.
  + **Code Analysis**:
    - Key functions or APIs (e.g., CreateProcess, WriteFile).
    - Embedded resources (e.g., additional payloads, configuration files).
  + **Obfuscation Techniques**:
    - Packers, encryptors, or anti-analysis methods (e.g., anti-debugging).
    - Tools used for unpacking or deobfuscation.

## **7. Dynamic/Behavioral Analysis**

* **Purpose**: Describe the malware’s behavior when executed in a controlled environment.
* **Content**:
  + **Behavioral Observations**:
    - File system changes (e.g., created, modified, or deleted files).
    - Registry modifications (e.g., persistence mechanisms).
    - Network activity (e.g., C2 communication, data exfiltration).
    - Process and memory activities (e.g., injection, privilege escalation).
  + **Payload Delivery**:
    - Dropped files or secondary payloads.
    - Execution chain (e.g., parent-child processes).
  + **Anti-Analysis Techniques**:
    - Detection of virtual environments or debugging tools.
    - Time-based delays or environmental checks.

Include screenshots or logs of significant behaviors if helpful.

## **8. Code/Reverse Engineering Analysis (if applicable)**

* Reverse Engineering (Optional)
* **Purpose**: Provide deeper insights into the malware’s code and functionality.
* **Content**:
  + Disassembled or decompiled code snippets (if relevant).
  + Key algorithms (e.g., encryption, data encoding).
  + Command-and-control (C2) protocol details.
  + Custom protocols or data structures.
  + Limitations of the analysis (e.g., heavily obfuscated code).

## **9. Indicators of Compromise (IOCs)**

* **Purpose**: Provide actionable data for detection and response.
* **Content**:
  + File-based IOCs (e.g., hashes, file names, paths).
  + Network-based IOCs (e.g., IPs, domains, URLs, user agents).
  + Registry-based IOCs (e.g., keys, values).
  + Behavioral IOCs (e.g., specific process patterns).
  + YARA rules or other detection signatures (if applicable).
  + Format IOCs in a structured way (e.g., JSON, CSV) for integration with SIEM tools.

Preferably present these in a table or in STIX/JSON format if integrating with tools.

## 10. Impact Assessment

* **Purpose**: Evaluate the potential or observed damage caused by the malware.
* **Content**:
  + Affected systems or data (e.g., encrypted files, stolen credentials).
  + Business impact (e.g., downtime, financial loss).
  + Scope of compromise (e.g., single host, network-wide).
  + Potential for lateral movement or persistence.

## **11. Mitigation & Detection Recommendations**

* Mitigation and Remediation
* **Purpose**: Provide actionable steps to contain and eradicate the malware.
* **Content**:
  + **Immediate Actions**:
    - Isolation of affected systems.
    - Blocking of malicious IPs or domains.
  + **Remediation Steps**:
    - Removal of malicious files and registry entries.
    - Patching vulnerabilities (e.g., specific CVEs).
    - Resetting compromised credentials.
  + **Prevention Strategies**:
    - Endpoint protection updates.
    - Employee training on phishing awareness.
    - Network segmentation and monitoring.

## **12. Timeline of Events**

Include a timeline of observed or inferred events based on logs and malware behavior.

## **13. Attribution (Optional)**

* **Purpose**: Identify potential threat actors or campaigns (if feasible).
* **Content**:
  + Links to known threat actors or groups (e.g., APT28, Lazarus Group).
  + Evidence supporting attribution (e.g., code similarities, infrastructure).
  + Confidence level in attribution (e.g., low, medium, high).
  + Limitations of attribution analysis.

## **14. Conclusion**

* **Purpose**: Summarize findings and reinforce key points.
* **Content**:
  + Recap of malware’s capabilities and impact.
  + Effectiveness of current defenses.
  + Future monitoring or research recommendations.

## **15. Appendices**

* **Purpose**: Include supplementary information for technical readers.
* **Content**:
  + **Appendix A: Tools Used**:
    - List of tools with versions (e.g., Ghidra 10.1, Cuckoo Sandbox).
  + **Appendix B: Raw Data**:
    - Logs, packet captures, or memory dumps (if applicable).
  + **Appendix C: Code Snippets**:
    - Relevant disassembled or decompiled code.
  + **Appendix D: References**:
    - External resources (e.g., CVE details, threat intelligence reports).

## **16. Formatting and Presentation**

* **Purpose**: Ensure clarity and professionalism.
* **Guidelines**:
  + Use clear headings and subheadings.
  + Include tables, diagrams, or screenshots for complex data (e.g., network traffic, process trees).
  + Maintain a consistent tone (technical but accessible).
  + Proofread for accuracy and clarity.
  + Use a standardized template for consistency across reports.

## **17. Additional Notes**

* **Audience Consideration**: Tailor technical depth based on the audience (e.g., more diagrams for executives, more code for analysts).
* **Timeliness**: Ensure the report is delivered promptly to support incident response.
* **Confidentiality**: Mark the report with appropriate sensitivity levels (e.g., “Internal Use Only”).
* **Version Control**: Track revisions if the report is updated.

## **18. References & Supporting Material**

* Links to related malware family analyses or CVEs.
* Links to threat intelligence reports.
* Appendix: full logs, screenshots, disassembly snippets, etc.