Vulnerability Report for Hardcoding Issues - Part 2 in DIVA Application

- **Title:** Vulnerability Report for Hardcoding Issues - Part 2 in DIVA Application

- **Severity:** Critical

- **Description:**

  The Diva application stores a vendor key in a text file within the application directory, exposing a hardcoded sensitive credential. This issue could lead to unauthorised access to third-party services, potentially compromising the integrity and confidentiality of the vendor's data.

- **Impact:**

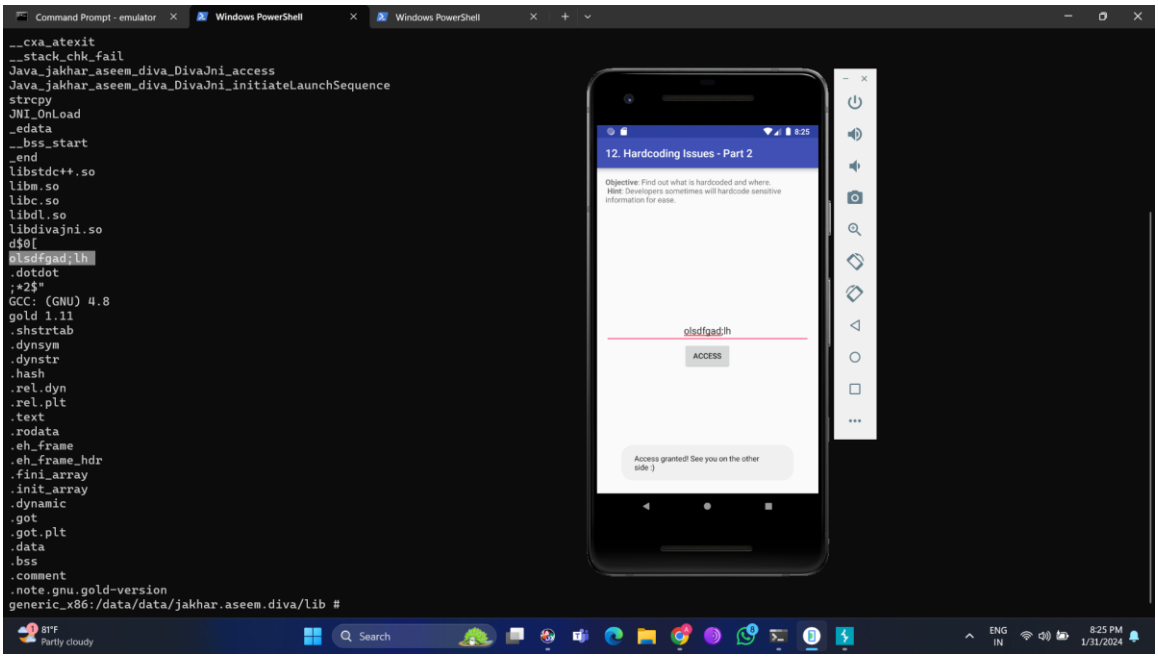
  1. Unauthorised Access: Attackers can easily obtain the hardcoded vendor key, leading to unauthorised access to third-party services.
  2. Data Exposure: The compromised key may result in exposure of sensitive information stored or processed by the third-party service.

- **Steps to Reproduce:**

  1. Login to the application.
  2. Click on the “Hardcoding Issues - Part 2” option.
  3. Open the “diva-beta.apk” file in the jadx application.
  4. In jadx open the ‘jakhar.assem.diva’ folder, present in the ‘Source code’ folder.
  5. Search for ‘Hardcode2Activity’ file and open it.
  6. The file contains the keyword “DivaJni” which indicates 3rd party interaction.
  7. Open the terminal and connect your device/emulator using the command “adb shell”.
  8. Use the command “cd data/data/jakhar.aseem.diva” to enter the directory.
  9. Get the root access by using the “su” command and list the contents of the directory using “ls”.
  10. Use the command “cd lib” to change the directory and list its contents.
  11. The directory contains a file named “libdivajni.so”.
  12. Read its contents by using the command “strings libdivajni.so”.

13. The file contains several vendor key options, out of which only one is correct.
14. Manually enter each and every key until you get “Access granted! See you on the other side :)”

- **PoC (Proof of Concept):**



- **Remediation:**

1. Use Secure Credential Storage: Avoid storing sensitive information like vendor keys in plaintext files. Utilise secure credential storage mechanisms provided by the operating system or secure configuration files.
2. Implement Environment Variables: Store sensitive information like vendor keys as environment variables rather than hardcoding them within the application.
3. Access Controls: Restrict access to the text file containing sensitive information to authorised personnel only..

- **CWE (Common Weakness Enumeration):**

1. CWE-798: Use of Hard-coded Credentials
2. CWE-256: Unprotected Storage of Credentials