. Exploit Development Basics

Activities:

- **Tools:** GDB, radare2.
- **Tasks:** Analyze and exploit a binary vulnerability.
- **Brief:**
- Binary Analysis: Use strings and GDB on a vulnerable C program. Summarize 3 findings in 50 words.

C Prrogram

```
#include <stdio.h>
#include <string.h>
#include <stdlib.h>

void hacked( ) {
    printf("You hacked me!\n");
    system("/bin/sh");
}

void vuln() {
    char buffer[64];
    fgets(buffer, sizeof(buffer), stdin);
    printf("You said: %s\n", buffer);
    hacked();
}

int main( ) {
    vuln( );
    return 0;
```

}

Sensitive strings exposed: Hardcoded credentials and dubious function names (such as gets a nd system) were exposed through the use of strings, suggesting unsafe operations; a stack-based buffer overflow vulnerability was confirmed by GDB, which showed input overwriting the return address; shell access was possible because the exploit causes arbitrary code execut ion, allowing shell access.

- Exploit PoC: Craft a buffer overflow payload; test in a VM.