Digisuraksha CyberSecurity Internship

Name: Prashik Intern ID : - 193

Task 1

What is GitHub?

GitHub is a cloud-based platform that helps developers manage and collaborate on code using **Git**, a distributed version control system. It allows multiple developers to work on projects simultaneously without interfering with each other's changes.

🔑 **Key GitHub Concepts**

- **Repository (Repo):** A storage space for your project, including code, files, and history.
- **Commit:** A snapshot of changes made to files.
- **Branch:** A separate version of the code for developing features independently.
- **Merge:** Combines code from different branches.
- **Pull Request (PR):** A request to merge code changes, typically reviewed by others.
- **Fork:** A personal copy of someone else's repository to experiment with.
- **Clone:** Copies a repository from GitHub to your local computer.
- **Push:** Sends your local changes to the GitHub server.
- **Pull:** Downloads the latest changes from GitHub to your local machine.

🔧 **Basic Git Commands**

```
git init              # Start a new Git repository
git clone <url>       # Download a repository from GitHub
```

```
git status                # Check the current state of your repository
git add <file>        # Stage a file for commit
git commit -m "msg"   # Save your staged changes with a message
git push              # Send committed changes to GitHub
git pull              # Get the latest changes from GitHub
```

🖥️ Part 2: Basic Linux Commands

Linux is an open-source operating system commonly used in software development. Knowing basic Linux commands is essential for navigating and managing files through the terminal.

📂 File and Directory Commands

```
pwd             # Show current directory
ls              # List files and folders
cd <dir>        # Change to another directory
mkdir <name>    # Create a new directory
touch <file>    # Create a new empty file
cp <src> <dst>  # Copy a file or folder
mv <src> <dst>  # Move or rename files
rm <file>       # Delete a file
rmdir <dir>     # Delete an empty directory
```

📄 Viewing and Editing Files

```
cat <file>      # View file contents
less <file>     # Scroll through file contents
nano <file>     # Open a file for editing using Nano editor
```

⚙️ System and Process Management

```
top             # Display real-time system processes
ps              # Show running processes
```

```
kill <pid>     # Terminate a process using its PID
```

🔎 **Helpful Tools**

```
man <command>  # Display manual/help for a command
history        # Show previously used commands
clear          # Clear the terminal screen
```

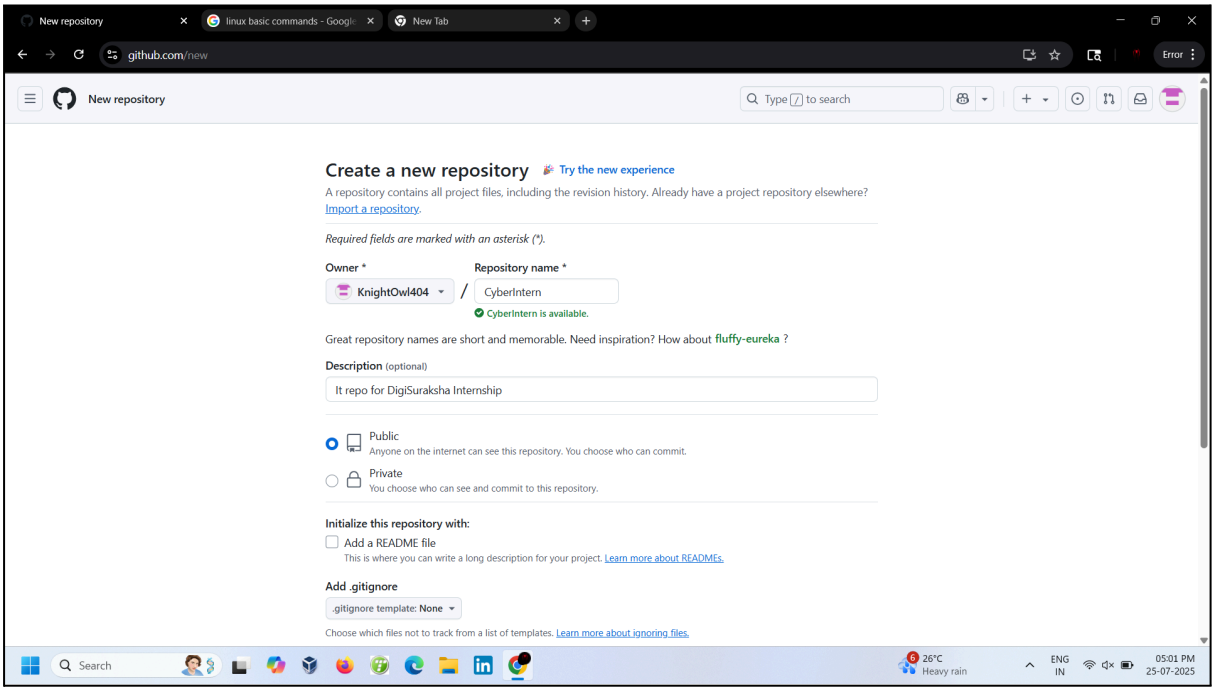
Summary

- GitHub is a collaborative platform that uses Git to manage version control.
- Linux commands allow efficient file and system management via the terminal.
- Understanding both tools is essential for developers and system administrators.

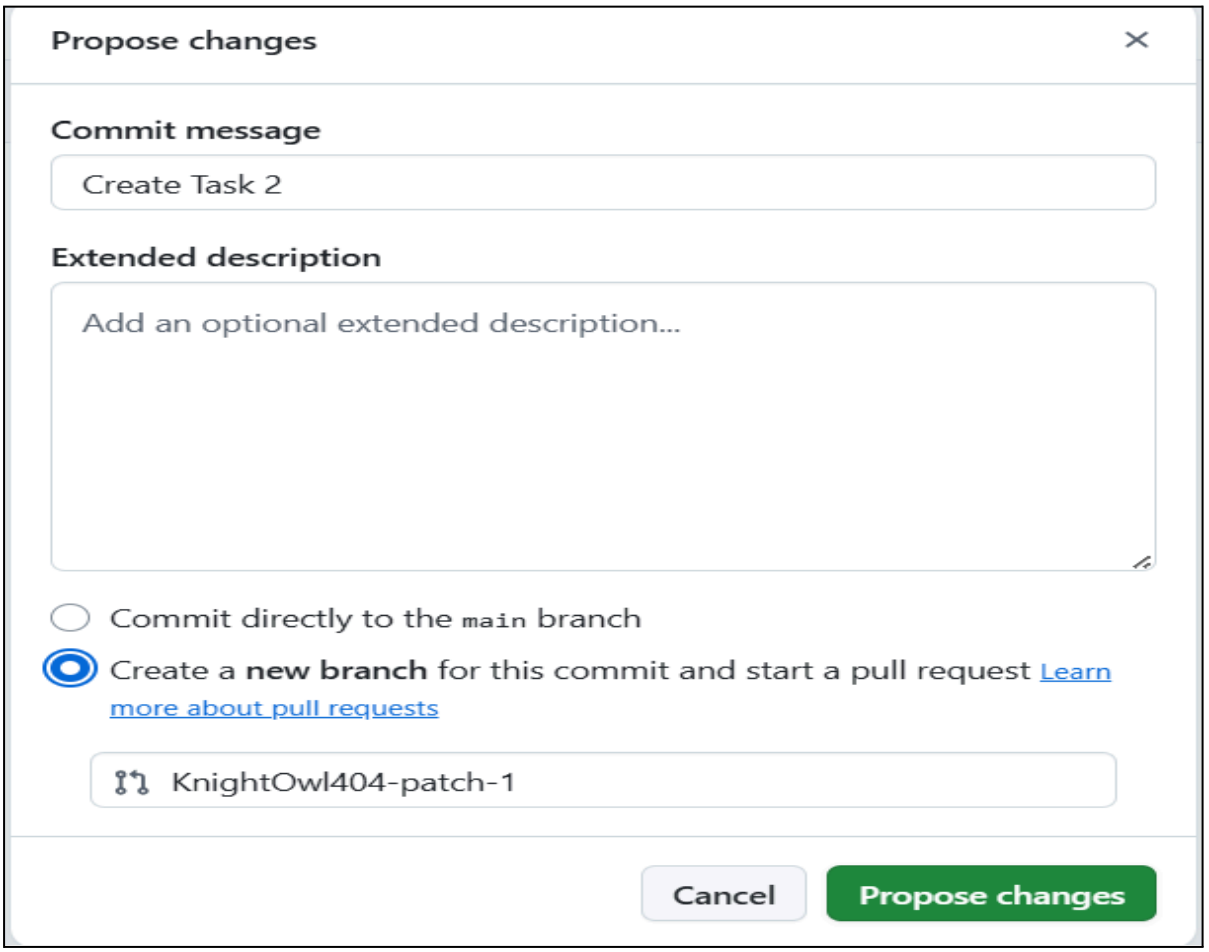
Task 2 :

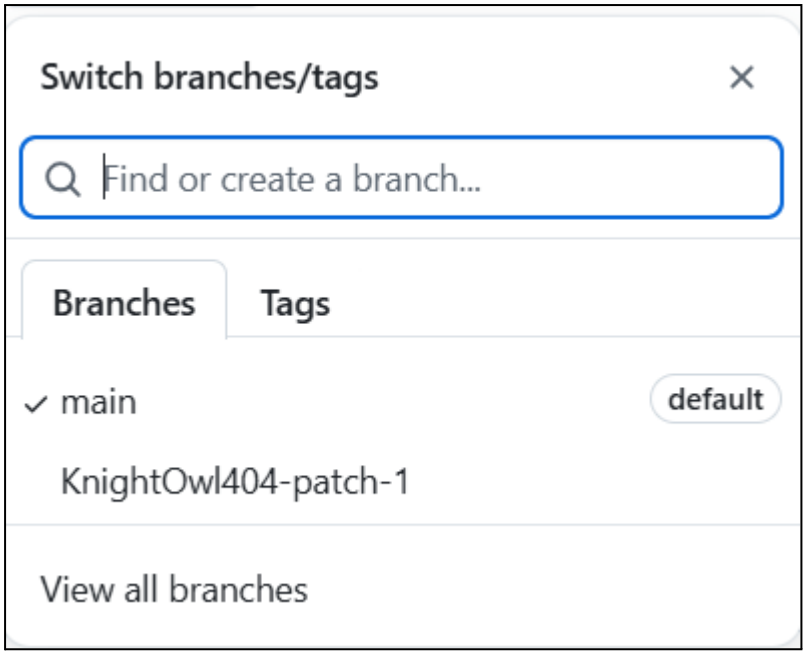
1. Create the GitHub Repository

1. Go to https://github.com.
2. Click the **"+" icon** (top right) → **New repository**.
3. Fill in:
    - Repository name (e.g., CyberIntern)
    - Description (optional)
    - Choose **Public** or **Private**
4. Click **Create repository**.



To create tree structure, create new branch when committing and then click on save changes





Branches look like this.

Task 3

Choosing Team

Which is red team.

Task 4

🧪 Digital Forensics

- The science of collecting, analyzing, and preserving digital evidence.

- Typically used in cybercrime investigations, data breaches, legal cases.

Key Areas:

- Disk and memory analysis
- Email analysis
- Malware forensics
- Network forensics
- Mobile forensics

🌐 OSINT (Open-Source Intelligence)

- Gathering intelligence from publicly available sources.
- Used in ethical hacking, investigations, journalism, threat hunting.

Key Focus:

- Finding hidden or non-indexed information from the web.
- Correlating metadata, social profiles, breached data, infrastructure.

🧰 2. Tools to Learn (and Categorized)

🧪 Digital Forensics Tools

Category	Tools
Disk Imaging	FTK Imager, dd, Guymager
Analysis	Autopsy, Sleuth Kit, X-Ways Forensics

Memory Forensics	Volatility, Rekall
Network Forensics	Wireshark, NetworkMiner
Mobile Forensics	MOBILedit, Cellebrite (commercial), Andriller

🌐 OSINT Tools

Category	Tools
People Search	Maltego, Spiderfoot, Sherlock, Skopenow
Social Media	OSINTgram, Twint, SocNetV
Domain & Infra	theHarvester, Shodan, Amass, Sublist3r
Breach Data	Dehashed, HaveIBeenPwned, IntelligenceX
Metadata	ExifTool, FOCA

Task 5

1. `youtube-research.md`

- **Link**: YouTube Malware Talk
- **Focus**:

    - Static vs Dynamic analysis
    - Key tools mentioned (IDA Pro, Ghidra, Wireshark, etc.)
    - Behavioral analysis using sandboxes
    - Registry persistence and process hollowing

✅ 2. `full-chain-exploit.md`

- **Link**: taszk.io full chain
- **Focus**:
    - Android browser-based RCE exploit chain
    - Vulnerabilities exploited (e.g., WebView bugs, JavaScript engine)
    - Memory corruption (heap spraying, ROP chains)
    - Real-world APT TTPs (Tactics, Techniques, and Procedures)

✅ 3. `macos-pkg-analysis.md`

- **Link**: macOS PKG Malware
- **Focus**:
    - Structure of `.pkg` files
    - How attackers embed malicious scripts in preinstall/postinstall
    - Manual unpacking and analysis
    - `pkgutil`, `Suspicious Package`, and `macOS Console`

✅ 4. `terabox-lab-notes.md`

- **Link**: TeraBox Sample

- **Focus** (based on contents):
    - Reverse engineer shared samples (if executable/memory dumps/logs are present)
    - Use **CAPE sandbox**, **PEStudio**, or **x64dbg**
    - Try behavioral detection: processes spawned, registry keys, file drops

✅ 5. `sysinternals-tools.md`

- **Link**: Microsoft Sysinternals
- **Focus**:
    - Key tools:
        - `Process Explorer` → Visualize parent-child process tree
        - `Autoruns` → Persistence detection
        - `Procmon` → File/Registry/Process events in real-time
        - `TCPView`, `Strings`, `Sigcheck`
    - Real use cases in malware detection

Task 6

Proof of Concept (POC)

This repository contains my research, experimentation, and POC developed using the allotted security tools.

🛠️ Tools Used

- Tool 1: Hex2dec - A command-line tool for converting hexadecimal to decimal (and vice versa), often used in memory and binary analysis.

- Tool 2: NotMyFault - A Sysinternals tool used to deliberately crash systems or trigger system faults, helpful for testing crash dump analysis and Blue Screen debugging procedures.

📌 Objective

To study and demonstrate real-world malware analysis and memory investigation concepts such as:

- Converting hex values from memory dumps

- Generating and analyzing system crash dumps using deliberate faults

Tool : Hex2Dec

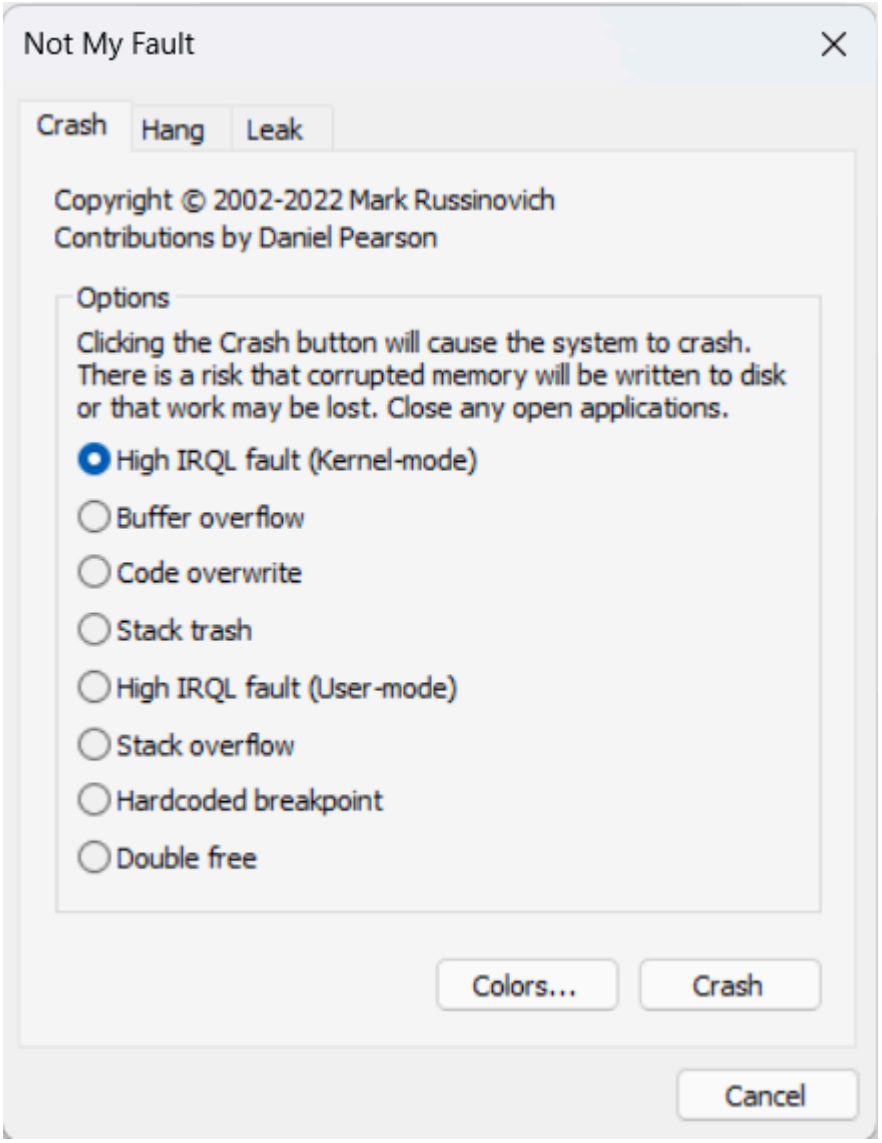
```
Windows PowerShell
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Install the latest PowerShell for new features and improvements! https://aka.ms/PSWindows

PS C:\Users\akash\Downloads\Hex2Dec> .\hex2dec.exe

Hex2dec v1.1 - converts hex to decimal and vice versa
Copyright (C) 2001-2016 Mark Russinovich
Sysinternals - www.sysinternals.com

Usage: hex2dec [decimal|hex]
Include x or 0x as the prefix of the number to specify a hexadecimal value.
e.g. To translate 1233 deciminal to hexadecimal: hex2dec 1233
e.g. To translate 0x1233 hexadecimal to decimal: hex2dec 0x1233
-nobanner    Do not display the startup banner and copyright message.

PS C:\Users\akash\Downloads\Hex2Dec> .\hex2dec.exe 150

Hex2dec v1.1 - converts hex to decimal and vice versa
Copyright (C) 2001-2016 Mark Russinovich
Sysinternals - www.sysinternals.com

150 = 0x96

PS C:\Users\akash\Downloads\Hex2Dec>
```

Converted dec to hex

```
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Install the latest PowerShell for new features and improvements! https://aka.ms/PSWindows

PS C:\Users\akash\Downloads\Hex2Dec> .\hex2dec.exe

Hex2dec v1.1 - converts hex to decimal and vice versa
Copyright (C) 2001-2016 Mark Russinovich
Sysinternals - www.sysinternals.com

Usage: hex2dec [decimal|hex]
Include x or 0x as the prefix of the number to specify a hexadecimal value.
e.g. To translate 1233 deciminal to hexadecimal: hex2dec 1233
e.g. To translate 0x1233 hexadecimal to decimal: hex2dec 0x1233
-nobanner    Do not display the startup banner and copyright message.

PS C:\Users\akash\Downloads\Hex2Dec> .\hex2dec.exe   0x7fff5fbff8a0

Hex2dec v1.1 - converts hex to decimal and vice versa
Copyright (C) 2001-2016 Mark Russinovich
Sysinternals - www.sysinternals.com

0x7FFF5FBFF8A0 = 140734799804576

PS C:\Users\akash\Downloads\Hex2Dec>
```

Hex to Dec

Tool NotMyFault



Triggering Hang with IRP

1. Ran `notmyfault64.exe`

2. Selected: `Hang with IRP`

3. System became unresponsive (no crash)

4. Performed Ctrl + ScrollLock x2 to force dump

5. Dump file generated at: `C:\Windows\Minidump`

Tools Used for Analysis

- BlueScreenView

- WinDbg: `!analyze -v` command used

Result:

Simulated I/O blocking bug caused by faulty driver behavior. Verified stack traces pointing to hung IRP queues.