## **Phase 1: Foundations of Offensive Security (Weeks 1-4)**

### **Topics:**

* Cybersecurity fundamentals
* Common attack vectors: SQLi, XSS, CSRF, LFI/RFI
* Basic cryptography concepts
* Introduction to Linux & Bash scripting

### **Resources:**

* [Hacker101](https://www.hacker101.com/)
* [The Web Application Hacker’s Handbook](https://www.amazon.com/Web-Application-Hackers-Handbook-Exploiting/dp/1118026470)
* TryHackMe: [Pre Security Path](https://tryhackme.com/)

### **Project:**

* Create a **basic penetration testing lab** using **Docker** (e.g., DVWA, WebGoat).
* Practice **SQL injection** on your lab.
* Write a **simple Bash script** to automate XSS payload testing.

## **Phase 2: Web Application Security & Exploitation (Weeks 5-8)**

### **Topics:**

* Advanced SQL injection (blind, time-based, WAF bypass)
* XSS (DOM-based, stored, reflected)
* CSRF attacks & mitigation
* Introduction to Burp Suite

### **Resources:**

* PortSwigger Web Security Academy
* OWASP Top 10: https://owasp.org/www-project-top-ten/

### **Project:**

* **Develop a simple web application** (JavaScript, SQL) and try to hack it.
* Automate **SQL injection scanning** in Java.
* Perform **Pentesting on a real-world CTF challenge** (Hack The Box, TryHackMe).

## **Phase 3: System & Network Exploitation (Weeks 9-12)**

### **Topics:**

* Network scanning & enumeration (Nmap, Netcat, Wireshark)
* Privilege escalation techniques
* Reverse shells & payload delivery
* Intro to Metasploit Framework

### **Resources:**

* Nmap Network Scanning
* TryHackMe: Linux Privilege Escalation
* [PentesterLab](https://pentesterlab.com/)

### **Project:**

* Write a **custom port scanner** in Java.
* Exploit **a vulnerable Linux machine** on TryHackMe.
* Create a **reverse shell using Bash & Java**.

## **Phase 4: Exploit Development & Malware Analysis (Weeks 13-16)**

### **Topics:**

* Buffer overflows & exploit writing
* Shellcode development
* Malware analysis basics
* Intro to Assembly & debugging

### **Resources:**

* Hack The Box: Buffer Overflow Guide
* [LiveOverflow YouTube](https://www.youtube.com/c/LiveOverflow)
* [Malware Unicorn](https://malwareunicorn.org/)

### **Project:**

* Develop a **basic keylogger in Java**.
* Write a **simple exploit for a buffer overflow vulnerability**.

## **Phase 5: Advanced Attacks & Red Teaming (Weeks 17-20)**

### **Topics:**

* Active Directory exploitation
* Social engineering techniques
* Advanced persistence techniques
* Bypassing security controls

### **Resources:**

* TryHackMe: Red Teaming
* The Art of Exploitation

### **Project:**

* Perform **a phishing attack simulation**.
* Simulate **Active Directory attacks** using TryHackMe labs.

## **Phase 6: Real-World Pentesting & Bug Bounty (Weeks 21-24)**

### **Topics:**

* Web & API pentesting methodology
* Writing bug bounty reports
* Evasion techniques

### **Resources:**

* Bug Bounty Hunter Methodology
* [CTF Platforms: Hack The Box, Root Me, CTFtime](https://ctftime.org/)

### **Project:**

* Participate in a **real bug bounty program**.
* Complete **at least 3 Hack The Box challenges**.

## **Next Steps**

* Build a **portfolio** with write-ups of your findings.
* Get a **certification** (CEH, OSCP).
* Contribute to **open-source security tools**.

This plan ensures you develop both **theoretical** and **practical** expertise. Let me know if you'd like adjustments! 🚀

## **SECOND PROMPT(coding projects)**

**6-Month Cybersecurity & Offensive Security Learning Plan**

### **Phase 1: Foundations of Offensive Security (Weeks 1-4)**

#### **Topics:**

* Cybersecurity fundamentals
* Common attack vectors: SQLi, XSS, CSRF, LFI/RFI
* Basic cryptography concepts
* Introduction to Linux & Bash scripting

#### **Projects:**

1. **Automated XSS Scanner** (Java/JavaScript)
   * Write a script that scans a given web page for potential XSS vulnerabilities.
2. **SQL Injection Testing Tool** (Java)
   * Create a tool that tests for SQL injection vulnerabilities in web applications.

### **Phase 2: Web Application Security & Exploitation (Weeks 5-8)**

#### **Topics:**

* Advanced SQL injection (blind, time-based, WAF bypass)
* XSS (DOM-based, stored, reflected)
* CSRF attacks & mitigation
* Introduction to Burp Suite

#### **Projects:**

1. **Custom Web Vulnerability Scanner** (JavaScript, SQL)
   * Develop a scanner that tests web applications for common vulnerabilities.
2. **Automated SQL Injection Exploiter** (Java)
   * Extend the SQL injection tool to automate data extraction and bypass security measures.

### **Phase 3: System & Network Exploitation (Weeks 9-12)**

#### **Topics:**

* Network scanning & enumeration (Nmap, Netcat, Wireshark)
* Privilege escalation techniques
* Reverse shells & payload delivery
* Intro to Metasploit Framework

#### **Projects:**

1. **Custom Port Scanner** (Java)
   * Develop a port scanner to identify open ports and running services on a network.
2. **Reverse Shell Implementation** (Java & Bash)
   * Create a simple reverse shell for remote access and control.

### **Phase 4: Exploit Development & Malware Analysis (Weeks 13-16)**

#### **Topics:**

* Buffer overflows & exploit writing
* Shellcode development
* Malware analysis basics
* Intro to Assembly & debugging

#### **Projects:**

1. **Keylogger Development** (Java)
   * Implement a simple keylogger that logs user input.
2. **Basic Buffer Overflow Exploit** (Java/C)
   * Develop a program that demonstrates a buffer overflow vulnerability.

### **Phase 5: Advanced Attacks & Red Teaming (Weeks 17-20)**

#### **Topics:**

* Active Directory exploitation
* Social engineering techniques
* Advanced persistence techniques
* Bypassing security controls

#### **Projects:**

1. **Custom Phishing Simulation Tool** (JavaScript, HTML)
   * Build a phishing website to test social engineering techniques.
2. **Credential Harvesting Tool** (Java)
   * Develop a tool to capture and analyze user credentials securely for educational purposes.

### **Phase 6: Real-World Pentesting & Bug Bounty (Weeks 21-24)**

#### **Topics:**

* Web & API pentesting methodology
* Writing bug bounty reports
* Evasion techniques

#### **Projects:**

1. **Automated API Fuzzer** (JavaScript, Java)
   * Develop a tool that fuzzes API endpoints for vulnerabilities.
2. **CTF Challenge Bot** (Java)
   * Build a bot that automates solving common Capture The Flag (CTF) challenges.

### **Next Steps:**

* Document and publish project findings in a portfolio.
* Contribute to open-source security tools.
* Prepare for industry certifications (CEH, OSCP).