VANTAGE POINT

# Web-Hacking Launchpad

## Hacking Lab Guide & Syllabus

Candidate Handbook

# 1. Introduction

**Web Hacking Launchpad (WHL)** is a hands-on web security event designed to strengthen your practical skills in identifying and exploiting vulnerabilities in modern web applications. The target, ThreadsApp, is a social media-style application with features such as account management, post creation, image uploads, and admin functionality. WHL is built to sharpen your testing mindset, improve your exploitation techniques, and help you understand how real-world web vulnerabilities appear in application workflows.

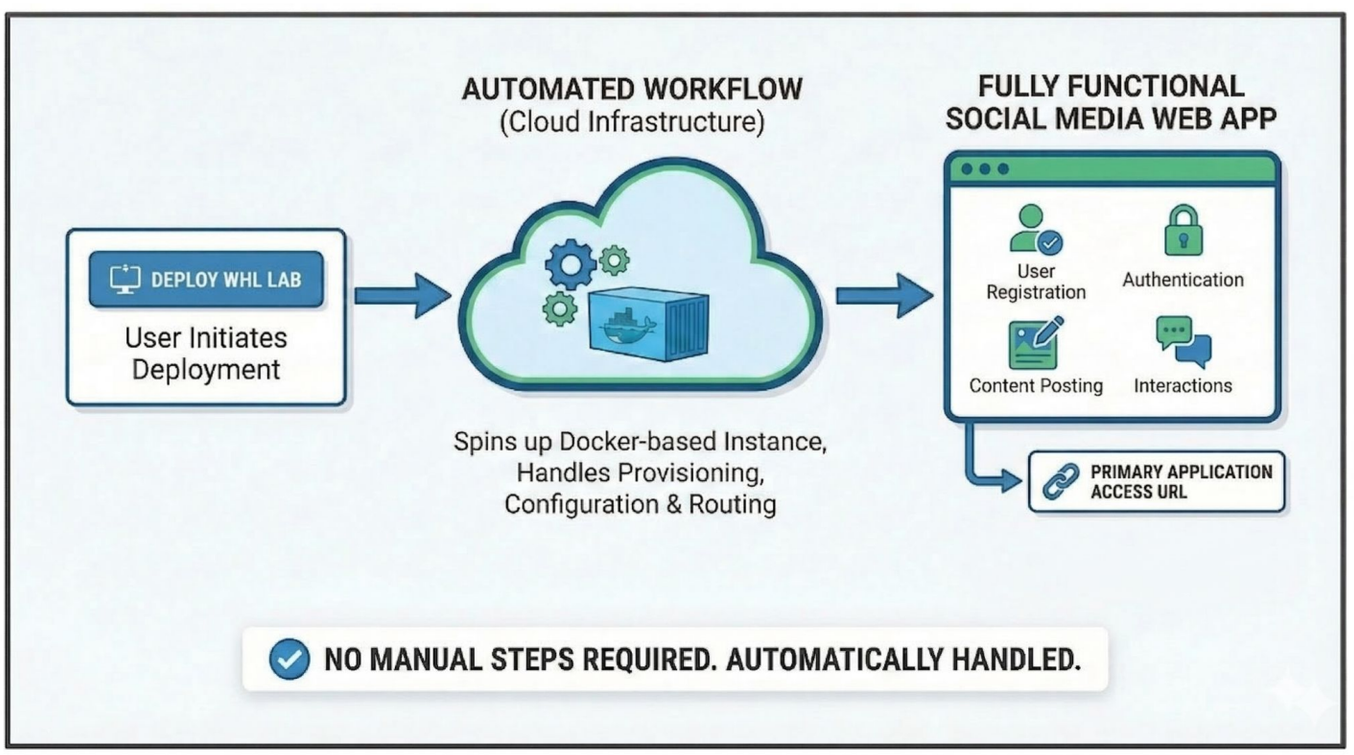
### IMPORTANT: Before You Begin

Before solving challenges, ensure that you can see the Target Lab Details in the "Ready To Tackle Challenges?" section on the dashboard.
If not, just click the button that says 'Deploy Lab' and the target application(s) will be deployed for you to work on.

# 2. About The Lab Infrastructure

When you deploy the WHL lab environment, an automated workflow provisions an instance of the vulnerable ThreadsApp application in the cloud. The lab is designed to behave like a realistic web application and includes multiple features commonly seen in modern platforms, such as user registration, authentication, posting functionality, media handling, account management, and administrative actions.

The application is intentionally designed with security flaws spread across different features, endpoints, and business logic. You begin only with the provided target details and must independently explore the application to understand how it works, what technologies or components may be involved, and where weaknesses exist. In simple words, **A Black Box Pentest Scenario**




## As part of the deployment process, the lab environment includes the following core components:

### Web Application Interface:
A functional social media application that supports user interaction, authentication, posting & media uploads, and account-related actions.

### Backend APIs & Data Layer:
Server-side endpoints and database systems are responsible for processing requests, managing data, and enforcing business logic.

### Administrative & Supporting Features:
Additional privileged functionality that may expand the application's attack surface for deeper testing.

Once the environment is successfully provisioned, the system will provide:

- Primary application access URL

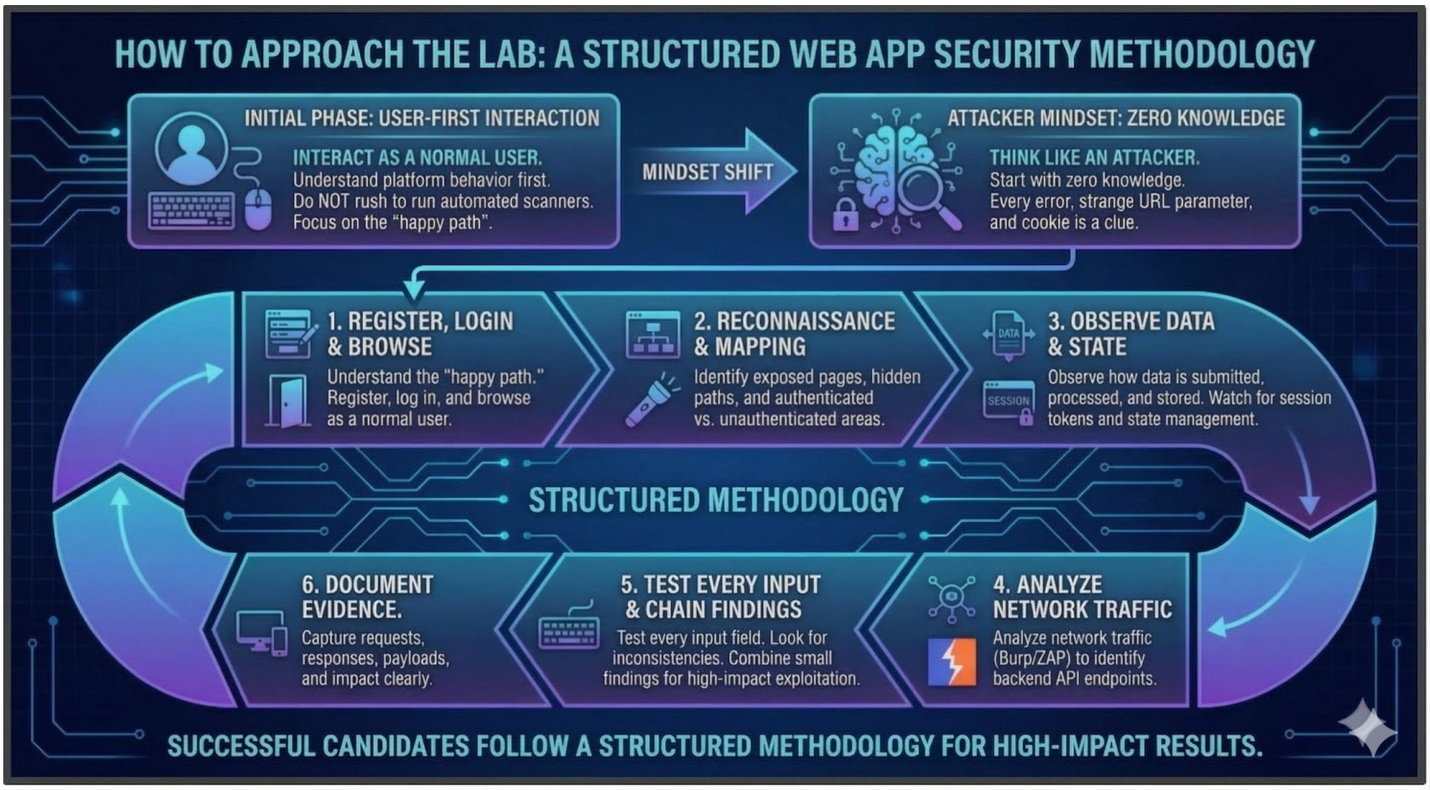
# 3. How To Approach The Lab

Once your lab URL is visible, begin by interacting with the application as a normal user. Your initial focus should be on understanding how the platform behaves before shifting into an offensive testing approach. Do not rush to run automated scanners immediately. Successful candidates follow a structured methodology.

**Mindset**: Think like an attacker with external access. Start with zero knowledge. Every error message, every strange URL parameter, and every cookie is a clue.

## General Methodology

- Register an account, log in, and browse as a normal user. Understand the "intended workflow."
- Identify exposed pages, hidden paths, and differences between authenticated vs. unauthenticated areas.
- Observe how data is submitted, processed, and stored. Watch for session tokens and state management
- Analyze network traffic (Burp/ZAP) to identify backend API endpoints.
- Test every input field. Look for inconsistencies. Combine small findings to achieve high-impact exploitation.
- Pay attention to file upload functionality, account features, admin-only areas for critical vulnerabilities.




# 4. Domain-Specific Objectives

> To perform well in WHL, participants should demonstrate practical skills in identifying, analyzing, and exploiting vulnerabilities in a modern web application from a black-box perspective.

- Demonstrate deep knowledge of HTTP(S), authentication flow, and common web application behavior.
- identifying standard and complex vulnerabilities (RCE, NoSQLi, XSS, SSRF, IDOR, etc.) in a black-box environment.
- Understand how application functionality, business logic flaws, and insecure workflows can be abused from an attacker's perspective.
- The ability to chain multiple minor vulnerabilities to achieve a high-impact goal
- Use manual testing along with standard tools effectively when default approaches do not immediately reveal the issue.

# 5. Scope of Vulnerabilities

Your mission is to identify and exploit a wide range of real-world vulnerabilities that impact modern web applications and APIs. These include, but are not limited to:

- NoSQL Injection (NoSQLi)
- Cross-Site Scripting (XSS)
- Server-Side Request Forgery (SSRF)
- Remote Code Execution
- JWT misconfiguration
- Chaining issues for higher impact vulnerability.
- More vulnerabilities across access controls.

# 6. Skills & Tools Required

## Core Competencies

### Web Application Fundamentals

Strong understanding of HTTP(S), client-server architecture, cookies, sessions, authentication flows, and common API communication patterns.

### Vulnerability Discovery & Exploitation

Ability to detect and exploit weaknesses across input handling, access control, data processing, and business logic to achieve impactful results.

### Tools & Automation:

Proficiency with popular web security testing tools and the ability to script automation workflows.

## Recommended Toolset

You should be proficient with the following tools:

- Burp Suite / OWASP ZAP
- Browser Developer Tools
- Fuzzing tools like FFUF, Dirb, Gobuster
- Nmap
- Sqlmap
- Metasploit

# 7. Exam Format & Rules

1

**Lab Access:**

You have access to the labs for as long as your subscription time. As this is a free event, you can renew the subscription anytime.

2

**Challenges:**

10 hands-on challenges testing web app vulnerabilities.

3

**No Flags (Not a CTF):**

This is not a traditional CTF. You can still receive credit for partial progress, so submit whatever you complete.

4

**Time Limits & Resets:**

If you miss the time limit, you have a maximum of **3 reset requests**.

For resets, Use the live chat or email us at training@vantagepoint-security.com

# 8. Evaluation Criteria

> Once you submit your proof of concept, the team manually evaluates your submission.

80%

### Technical Execution:

Successfully and fully solving the challenge, demonstrating a thorough understanding and execution of the core objectives.

20%

### Reporting & Creativity:

Submitting detailed reports with screenshots to showcase the approach taken, as well as demonstrating creativity and out-of-the-box thinking.

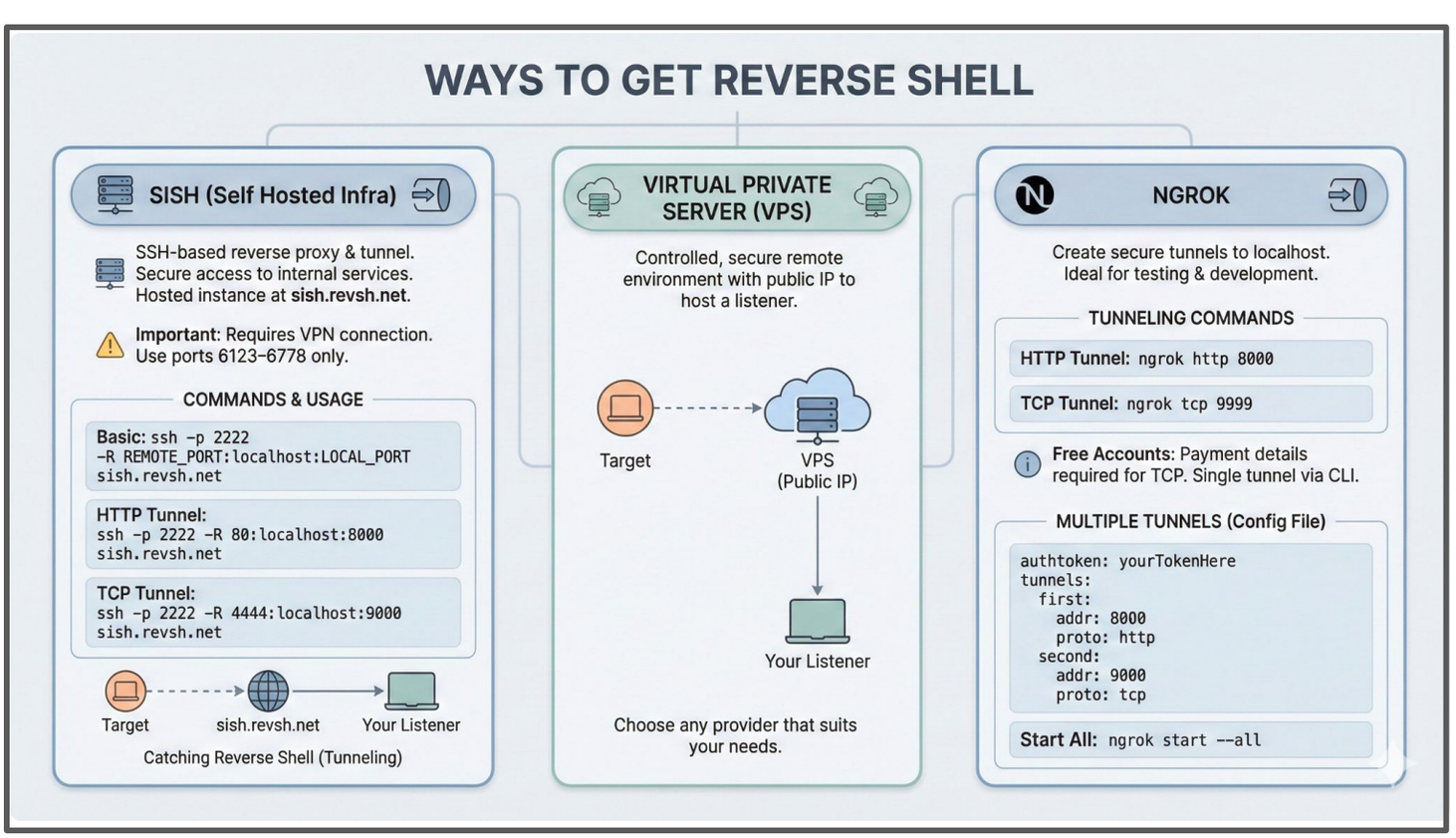
# 9. Catching Reverse Shells

- After you exploit a target, challenges may require you to obtain a reverse shell to gain interactive control. Normally, this requires a public IP address.
  However, since you are working from a local machine (likely behind NAT), we provide solutions to handle port forwarding and tunnel the connection back to you.

- **Option A:** Use our own self-hosted infra: **sish.revsh.net**

  **Option B:** Your own VPS

  **Option C:** Ngrok (Secure tunneling platform)



# 10. Do’s and Don’ts


# 11. Final Notes For Participants

- Explore carefully, test methodically, and document what you find.
- Submit whatever progress you make, even if a challenge is not fully solved.
- The goal is not just to solve challenges but to improve the way you think and test as a web hacker.
- Most importantly, use WHL as an opportunity to sharpen your real-world web hacking skills.

VANTAGE POINT

# Vantagepoint

## The Future Of Cybersecurity Start Here

Other Certifications & Labs:

- Multi-Cloud Security Expert
- Certified Mobile Appsec Expert
- Mobile Hacking Launchpad
- CVE Cipher: Decoding Threats
- Web Hacking Launchpad

training@enciphers.com

www.vantagepoint.com