The Ultimate Guide to 360 TryHackMe Free Rooms: Master Cybersecurity for Free

Author: RxPl0it **Site:** denizhalil.com **Date:** August 2026

INTRODUCTION

Starting a career in cybersecurity can feel like standing at the foot of an impossibly tall mountain. With endless sub-fields—ranging from web application security and digital forensics to reverse engineering, malware analysis, and cloud security—it is remarkably easy for beginners to feel overwhelmed about where to start, which path to follow, and what skills to prioritize. Fortunately, hands-on platforms like **TryHackMe (THM)** have completely revolutionized modern cybersecurity education by making complex technical topics accessible through browser-based labs. While THM offers a premium subscription, its extensive library of high-quality **free rooms** contains a wealth of practical knowledge that can take you from a complete novice to a competent security enthusiast. In this guide, we break down these free resources into structured, domain-specific learning paths, empowering you to build genuine, real-world skills and gain practical confidence without spending a single penny.

LEARNING OBJECTIVES

By working your way through this comprehensive collection of free TryHackMe rooms, you will be able to:

- **Understand Core Security Concepts:** Build a solid foundation in operating systems (Linux & Windows), networking protocols, and basic security models like the CIA Triad.
- **Master Essential Security Tools:** Gain hands-on experience with industry-standard tools including Nmap, Burp Suite, Metasploit, Wireshark, CyberChef, and Hydra.
- **Develop an Offensive Mindset (Red Teaming):** Learn how to perform active/passive reconnaissance, exploit web vulnerabilities (OWASP Top 10), execute privilege escalation, and breach target networks.
- **Build Defensive Capabilities (Blue Team & SOC):** Understand log analysis, incident response workflows, digital forensics (DFIR), memory analysis, and threat hunting techniques.
- **Apply Knowledge in Practical Environments:** Solve realistic Capture The Flag (CTF) challenges across various difficulty levels (Easy, Medium, Hard) to test your technical skills.

WHY DID I CURATE THIS TRYHACKME FREE ROOMS RESOURCE?

When I first stepped into the vast world of cybersecurity, I quickly realized that breaking into the industry comes with a steep learning curve. Beginners are instantly bombarded with thousands of tutorials, tool recommendations, and conflicting advice on where to begin. This constant state of information overload, combined with expensive subscription paywalls on many popular training platforms, creates an intimidating barrier for enthusiastic learners who simply want to get their hands dirty. While platforms like TryHackMe offer exceptional value, their free content is often scattered across dozens of different categories and skill levels. Without a clear path, self-taught students frequently waste precious time bouncing between rooms that are either too basic to be useful or too complex for their current technical level. This fragmented experience often leads to frustration, burnout, and the false belief that practical cybersecurity training is inaccessible without financial investment.

I created and categorized this master collection of TryHackMe free rooms to solve this exact problem and eliminate the guesswork for you. By curating every accessible free room and organizing them into a logical, domain-specific roadmap—spanning everything from basic OS fundamentals to advanced CTFs—I wanted to build a structured, cost-free curriculum that empowers anyone in the world to master practical cybersecurity at their own pace.

Key Reasons Behind This Project:

- **Eliminating Financial Barriers:** Providing a 100% free, high-quality learning roadmap so that students, career-changers, and independent researchers can build real-world skills regardless of their budget.
- **Overcoming Information Overload:** Filtering out the noise by organizing scattered platform rooms into clear, logical categories that mirror real-world security domains.
- **Saving Valuable Time:** Giving you a direct, plug-and-play resource so you can spend less time searching for what to study next and more time actually hacking.
- **Fostering Structured Progression:** Ensuring a smooth learning curve that naturally guides you from fundamental computing concepts all the way to complex, multi-stage CTF challenges.

CAN YOU BECOME A CYBERSECURITY EXPERT USING ONLY THIS RESOURCE?

The short answer is that while this curated collection of free TryHackMe rooms provides an exceptionally strong foundation, becoming a true cybersecurity expert is an ongoing journey that requires real-world experience, continuous practice, and deep specialization over time. Hands-on labs are brilliant for cultivating technical muscle memory and understanding core mechanics, but mastering the field demands moving beyond pre-configured virtual environments into complex, unpredictable enterprise infrastructures. Think of this comprehensive resource as your launchpad—it bridges the gap between zero knowledge and

practical operational competence, giving you the exact momentum needed to confidently break into the industry. By systematically working through these categorized rooms, you will build a solid practical baseline that far exceeds what static textbooks or purely theoretical lectures can offer. You will gain direct experience with essential security tools, master offensive and defensive methodologies, and develop a well-rounded understanding of diverse domains ranging from web application security and privilege escalation to digital forensics and cloud infrastructure. Furthermore, this broad exposure allows you to experiment across different fields, helping you identify whether your true passion lies in Red Teaming, Blue Teaming, Threat Hunting, or Security Engineering, while simultaneously preparing you for entry-level security certifications like CompTIA Security+, eJPT, or PJPT.

However, bridging the gap from a capable junior practitioner to a seasoned industry expert requires expanding your horizon beyond guided platform labs. True expertise involves navigating unguided, real-world networks, mastering complex advanced attack vectors like Active Directory forest takeovers or custom binary exploit development, and developing crucial soft skills such as risk management and executive report writing. Because the threat landscape evolves daily with new vulnerabilities and defensive controls, maintaining expertise is a lifelong commitment to learning, building custom home labs, analyzing novel CVEs, and constantly adapting to new technologies.

Key Takeaways for Your Learning Journey:

- **Building Practical Competence:** Completing this roadmap equips you with genuine, hands-on skills and operational familiarity with industry-standard tools, elevating your practical ability far beyond theoretical knowledge.
- **Preparing for Industry Gateways:** The technical baseline established here directly aligns with the foundational knowledge required for entry-level roles (such as Junior SOC Analyst or Junior Pentester) and hands-on beginner certifications.
- **Discovering Your Specialized Path:** Working through varied domains—including Web, Forensics, Malware Analysis, Cloud, and PrivEsc—enables you to discover your specific strengths and choose an expert specialization.
- **Scaling Beyond Guided Environments:** Transforming foundational competence into senior-level expertise requires continuous real-world practice, building home labs, analyzing new threats, and mastering soft skills like risk assessment and reporting.

360 TOP TRYHACKME FREE ROOMS

Intro Rooms

Starting your cybersecurity journey can feel overwhelming given the vast array of topics to cover. The **Intro Rooms** collection on TryHackMe serves as the ideal launchpad, designed to introduce core concepts, essential tools, platform mechanics, and fundamental security disciplines. Whether you are setting up your VPN connection for the first time, learning the

basics of web technology and cryptography, or taking your initial steps into threat hunting and SOC operations, these free rooms provide the foundational knowledge necessary to build a solid cybersecurity skillset.

- TryHackMe | Microservices Architectures – Learn the fundamentals of microservices design, containerized deployment, and the unique security considerations surrounding modern application architectures.
- TryHackMe | Jupyter 101 – Discover how to use Jupyter Notebooks for data analysis, scripting, and documenting cybersecurity research effectively.
- TryHackMe | Intro To Pwntools – Get started with Pwntools, a Python library built for rapid prototyping and developing binary exploitation scripts.
- TryHackMe | Introduction to Flask – Understand how lightweight web applications are built using the Python Flask framework and how web routing functions.
- TryHackMe | Intro PoC Scripting – Learn the basics of writing simple Proof-of-Concept (PoC) scripts to automate vulnerability demonstrations.
- TryHackMe | Bypass Really Simple Security – Explore common logical flaws in basic authentication mechanisms and learn how to bypass weak security controls.
- TryHackMe | Web Application Basics – Gain a introductory understanding of how web applications operate, including client-server architecture and HTTP fundamentals.
- TryHackMe | Cryptography Basics – Master fundamental cryptographic concepts, including symmetric/asymmetric encryption, hashing, and ciphers.
- TryHackMe | CyberChef: The Basics – Learn how to manipulate, encode, decode, and analyze data using CyberChef, the "Cyber Swiss Army Knife."
- TryHackMe | SOC Fundamentals – Explore the core roles, workflows, and tools utilized within a Security Operations Center (SOC).
- TryHackMe | Networking Concepts – Build a baseline understanding of network models, IP addresses, subnets, and core protocols.
- TryHackMe | Search Skills – Enhance your Open-Source Intelligence (OSINT) and research capabilities using advanced web search techniques.
- TryHackMe | Windows Command Line – Get hands-on experience navigating the Windows OS using the Command Prompt (cmd.exe) and built-in utilities.
- TryHackMe | Hosted Hypervisors – Understand virtualization technology, Type-2 hypervisors, and how to configure virtual environments.
- TryHackMe | Enumeration & Brute Force – Learn basic target enumeration strategies and password brute-forcing techniques.
- TryHackMe | Introduction to CryptOps – Explore how cryptographic operations are integrated into real-world operational security workflows.
- TryHackMe | Linux File System Analysis – Understand the structure of Linux directory trees and how to inspect file system metadata for security analysis.

- TryHackMe | Threat Hunting: Foothold – Learn how to identify initial access tactics and detect early adversary footholds in a network.
- TryHackMe | Threat Hunting: Introduction – Discover the proactive methodology of threat hunting to search for undetected malicious activity.
- TryHackMe | Preparation – Learn essential incident response preparation steps to build effective defense strategies before an attack occurs.
- TryHackMe | Intro to Logs – Understand what system and application logs are, why they matter, and how to analyze them for security events.
- TryHackMe | Intro to Threat Emulation – Explore how security teams mimic adversary behavior to evaluate and improve defense mechanisms.
- TryHackMe | Security Engineer Intro – Get an overview of the Security Engineering discipline, focusing on designing, building, and maintaining secure systems.
- TryHackMe | Intro to Docker – Learn the fundamentals of containerization, managing Docker images, and running secure containers.
- TryHackMe | SDLC – Explore the Software Development Life Cycle and how security is integrated throughout development phases (DevSecOps).
- TryHackMe | Welcome – An introduction to the TryHackMe community, platform structure, and learning methodologies.
- TryHackMe | How to use TryHackMe – A quick walkthrough on how to navigate THM rooms, complete tasks, and track your learning progress.
- TryHackMe | Tutorial – Learn how to deploy vulnerable machines on the platform and submit your first flags.
- TryHackMe | OpenVPN – A practical guide to setting up OpenVPN to connect your personal machine securely to TryHackMe's internal network.
- TryHackMe | Learning Cyber Security – An overview of offensive and defensive security career paths to help guide your learning journey.
- TryHackMe | Starting Out In Cyber Sec – Learn about different industry roles, certifications, and essential skills needed to start a career in security.
- TryHackMe | Introductory Researching – Master the art of information gathering and leveraging documentation to solve complex technical problems.
- TryHackMe | Regular expressions – Learn how to construct RegEx patterns to search, parse, and extract specific log data efficiently.

Linux Fundamentals

A strong grasp of the Linux operating system is an absolute requirement for any aspiring cybersecurity professional. Linux powers the vast majority of web servers, cloud infrastructure, and security-focused distributions like Kali Linux and Parrot OS. This section covers key concepts including Kernel modules, core terminal commands, and system navigation.

Additionally, supplementary external resources are included to help you build a solid Linux foundation even if platform rooms shift over time.

- TryHackMe | Modules – Learn about Linux Kernel modules, how to list, load, and unload drivers, and understand how hardware interfaces with the OS.
- TryHackMe | Linux Fundamentals Part 1 – Get introduced to basic terminal commands, directory navigation, and essential file manipulation operations.
- Basics of Linux – A comprehensive interactive guide covering fundamental to advanced Linux concepts, serving as a great free alternative for deep studying.
- A very deep dive book – A thorough, 900+ page reference book ideal for mastering Linux administration, system architecture, and command-line mastery.
- A reddit community where Linux challenges are posted daily – A practical community offering daily hands-on lessons and challenges to hone your sysadmin skills over 20 days.

Windows Fundamentals

Understanding the Windows architecture is critical for both offensive and defensive security professionals, as Windows environments dominate corporate networks worldwide. This multi-part series introduces you to the core structural components of the Windows operating system, covering everything from basic GUI interaction to security settings, system monitoring tools, and administrative utilities like User Account Control (UAC) and PowerShell.

- TryHackMe | Windows Fundamentals 1 – Explore the basic Windows user interface, system settings, file management, and core administrative tools.
- TryHackMe | Windows Fundamentals 2 – Dive into system administration utilities, such as Task Manager, System Configuration, Disk Management, and Registry Editor.
- TryHackMe | Windows Fundamentals 3 – Learn about key Windows security features, including Windows Defender, Firewalls, User Account Control (UAC), and Event Viewer.

Basics Rooms

Before jumping into advanced exploits or defensive tools, you must understand the core concepts that underpin computing infrastructure and security frameworks. The **Basics Rooms** section provides a wide-ranging overview of computer architecture, virtualization, data encoding, security principles, compliance standards, and fundamental pentesting methodologies. It bridges the gap between general IT knowledge and practical cybersecurity execution.

- TryHackMe | Operating Systems: Introduction – Learn how operating systems manage system hardware, processes, memory allocation, and file storage.
- TryHackMe | Windows Basics – Understand basic Windows terminology, built-in features, and standard system operations.

- TryHackMe | Data Representation – Learn how computers store and interpret data using binary, hexadecimal, ASCII, and Unicode.
- TryHackMe | Data Encoding – Explore common data formats such as Base64, URL encoding, and Hex encoding used to transport data safely.
- TryHackMe | The CIA Triad – Master the core security model based on Confidentiality, Integrity, and Availability.
- TryHackMe | Inside a Computer System – Take a tour of computer hardware, including the CPU, RAM, storage drives, and motherboard functions.
- TryHackMe | Computer Types – Learn about different hardware environments, ranging from personal computers and servers to embedded devices.
- TryHackMe | Virtualisation Basics – Explore how virtual machines operate, hypervisors work, and modern labs are constructed.
- TryHackMe | Writing Pentest Reports – Master the essential skill of documenting security findings and creating professional penetration testing reports.
- TryHackMe | Bypass Really Simple Security – Identify logical flaws in application security controls and learn basic bypass strategies.
- TryHackMe | Insecure Randomness – Understand how weak pseudo-random number generators (PRNGs) lead to cryptographic vulnerabilities.
- TryHackMe | Hypervisor Internals – Gain deeper knowledge into how Type-1 and Type-2 hypervisors manage virtualized CPU, memory, and device drivers.
- TryHackMe | Splunk: Exploring SPL – Learn how to search, filter, and correlate log data using Splunk Search Processing Language (SPL).
- TryHackMe | ParrotPost: Phishing Analysis – Analyze email headers, attachments, and embedded URLs to identify malicious phishing attempts.
- TryHackMe | x86 Architecture Overview – Understand CPU registers, memory stacks, and instruction sets crucial for binary exploitation and reverse engineering.
- TryHackMe | Threat Intelligence for SOC – Discover how threat intelligence feeds and Indicators of Compromise (IoCs) assist SOC analysts in incident response.
- TryHackMe | Basic Pentesting – Apply foundational penetration testing techniques to compromise a vulnerable target machine.
- TryHackMe | Pentesting Fundamentals – Learn about rules of engagement, pentesting scopes, methodologies, and legal considerations.
- TryHackMe | Principles of Security – Study foundational defense models, access control concepts, and security design principles.
- TryHackMe | The Hacker Methodology – Understand the step-by-step process attackers use, from initial reconnaissance to maintaining persistence.
- TryHackMe | Physical Security Intro – Explore physical access controls, social engineering techniques, and security measures protecting physical hardware.

- TryHackMe | Linux Strength Training – Sharpen your Linux command-line skills through a series of practical system administration challenges.
- TryHackMe | OpenVAS – Learn how to configure, run, and analyze automated network vulnerability scans using OpenVAS.
- TryHackMe | ISO27001 – Understand the global ISO/IEC 27001 standard for managing Information Security Management Systems (ISMS).
- TryHackMe | UltraTech – Practice basic web application reconnaissance, API testing, and hash cracking on a beginner CTF machine.

Recon

Reconnaissance (Recon) is the foundational phase of any security assessment, whether offensive penetration testing or defensive threat analysis. Before attempting to interact directly with a target, security practitioners gather intelligence to map out attack surfaces, discover exposed assets, and identify potential vulnerabilities. This section explores both passive gathering techniques—like search engine dorking, OSINT, and specialized search engines like Shodan—and active techniques that involve directly probing target networks and web content.

- TryHackMe | Cyber Kill Chain – Learn Lockheed Martin's framework describing the stages of a cyberattack, emphasizing the pivotal role of initial reconnaissance.
- TryHackMe | Passive Reconnaissance – Discover footprinting techniques to gather target information without sending direct traffic or alerting defenders.
- TryHackMe | Active Reconnaissance – Master techniques for directly interacting with target systems using tools like Nmap, Ping, and Traceroute to uncover open ports and services.
- TryHackMe | Content Discovery – Learn how to locate hidden files, unlinked directories, and secret assets on web servers using automated brute-forcing tools.
- TryHackMe | OhSINT – Put your Open-Source Intelligence (OSINT) skills to the test in a practical challenge using image metadata and digital traces.
- TryHackMe | Shodan.io – Learn how to use the search engine for Internet-connected devices to discover exposed servers, IoT devices, and infrastructure vulnerabilities.
- TryHackMe | Google Dorking – Master advanced search engine operators to expose sensitive documents, hidden web pages, and database dumps indexed publicly.
- TryHackMe | WebOSINT – Explore specialized techniques and web tools used to perform OSINT investigations on domain names, IP addresses, and websites.
- TryHackMe | Sakura Room – Solve an OSINT-focused CTF challenge requiring deep investigation across social media platforms, domain data, and image location analysis.
- TryHackMe | Searchlight - IMINT – Learn Image Intelligence (IMINT) and geolocation techniques to analyze photos and pinpoint real-world geographic locations.

Scripting

Automation is a force multiplier in cybersecurity. Writing scripts enables security professionals to build custom exploit payloads, automate repetitive network tasks, parse complex log files, and create custom tools on the fly. This section covers scripting and programming fundamentals across essential languages, including Python, JavaScript, Bash, and Rust, helping you transition from relying on existing security tools to building your own.

- TryHackMe | Custom Tooling Using Python – Learn how to write customized Python scripts to interact with network services, automate web requests, and build security utilities.
- TryHackMe | Python Basics – Master Python programming fundamentals, including variables, loops, functions, and control structures tailored for security applications.
- TryHackMe | Python Playground – Practice Python-based web reverse engineering and script execution challenges in a hands-on CTF environment.
- TryHackMe | Intro PoC Scripting – Learn how to take known vulnerability details and write functional Proof-of-Concept (PoC) exploit scripts using Python.
- TryHackMe | Peak Hill – Test your Python decoding, data manipulation, and script-building capabilities through a practical challenge.
- TryHackMe | JavaScript Basics – Understand JavaScript syntax and client-side execution, a crucial prerequisite for analyzing web vulnerabilities like XSS.
- TryHackMe | Bash Scripting – Learn shell scripting syntax to automate terminal workflows, process system outputs, and execute admin tasks on Linux.
- TryHackMe | Learn Rust – Explore the fundamentals of Rust, a memory-safe system programming language increasingly popular in modern security tools and exploit development.

Networking

Network communications form the backbone of modern IT environments and internet infrastructure. To attack or defend a network effectively, you must understand how data travels between devices, how traffic is structured, and how core protocols function under the hood. This section breaks down foundational networking architectures, essential communication protocols like HTTP and DNS, traffic inspection, and network discovery detection.

- TryHackme | Network Discovery - Scan-ta Clause – Learn fundamental network scanning and host discovery techniques through a Christmas-themed hands-on challenge.
- TryHackMe | Network Traffic Basics – Understand packet encapsulation, frame structures, and how network packets travel across local and remote networks.
- TryHackMe | Network Security Essentials – Explore defense tools and architectural patterns used to secure network perimeters, including firewalls, IDS/IPS, and VPNs.
- TryHackMe | Network Discovery Detection – Discover how defenders observe, identify, and alert on network scanning and enumeration activities using log analysis.

- TryHackMe | Introductory Networking – Build a fundamental understanding of the OSI Model, TCP/IP stack, IP addressing, port numbers, and subnetting.
- TryHackMe | What is Networking? – Get an introductory overview of computer networks, physical connections, and baseline internet operations.
- TryHackMe | Networking – Reinforce key networking concepts, common administrative commands, and protocol functions in a practical scenario.
- TryHackMe | Intro to LAN – Explore Local Area Network concepts, including Ethernet connections, MAC addressing, switches, and routers.
- Khan Academy | The Internet Resource – A comprehensive free alternative covering how the internet works, covering packet routing, HTTP, DNS, and encryption protocols.
- TryHackMe | HTTP in detail – Dive deep into the HTTP protocol, studying request/response cycles, headers, status codes, cookies, and session management.
- TryHackMe | DNS in detail – Understand the Domain Name System, studying how domain queries are resolved, different record types (A, AAAA, MX, TXT), and DNS hierarchy.
- TryHackMe | Dumping Router Firmware – Learn how physical network device firmware is extracted, analyzed, and audited for hidden vulnerabilities.

Tooling

Equipping yourself with the right software is essential for performing security audits, vulnerability scanning, and penetration testing efficiently. Cybersecurity tools help automate discovery, analyze web traffic, brute-force access credentials, and manage remote command-line sessions. This section covers industry-standard tools—ranging from scanning and proxy utilities like Nmap, Burp Suite, OWASP ZAP, and Nessus to terminal productivity applications like tmux and Vim—designed to streamline your workflow and expand your operational capabilities.

- TryHackMe | Snyk Open Source – Learn how to scan open-source dependencies in software projects to identify and remediate known security vulnerabilities.
- TryHackMe | Snyk Code – Explore Static Application Security Testing (SAST) techniques to uncover security flaws directly within source code repositories.
- TryHackMe | Intro to IaC – Understand Infrastructure as Code (IaC) principles and learn how to scan Terraform and CloudFormation templates for misconfigurations.
- TryHackMe | Metasploit: Introduction – Master the fundamentals of the Metasploit Framework, including exploring modules, configuring payloads, and managing sessions.
- TryHackMe | Metasploit: Introduction (RP) – Reinforce your understanding of Metasploit by learning how to use msfconsole for host scanning and automated exploitation.
- TryHackMe | tmux – Learn how to use tmux, a terminal multiplexer that allows you to manage multiple command-line sessions within a single window.
- TryHackMe | REmux The Tmux – Practice advanced terminal session management, pane splitting, and custom window configurations using tmux.

- TryHackMe | Hydra – Learn how to perform fast online password brute-forcing against various network protocols such as SSH, FTP, HTTP, and SMB.
- TryHackMe | Toolbox: Vim – Get comfortable with Vim, a powerful terminal-based text editor essential for managing configuration files on remote servers.
- TryHackMe | Introduction to OWASP ZAP – Discover OWASP ZAP, an open-source web application security scanner used for intercepting web requests and automated vulnerability scanning.
- TryHackMe | Phishing: HiddenEye – Examine how phishing frameworks function to raise awareness about credential harvesting and social engineering defenses.
- TryHackMe | RustScan – Learn how to leverage RustScan, a modern, ultra-fast port scanner designed to quickly discover open ports and pipe results into Nmap.
- TryHackMe | Nessus – Configure and run vulnerability scans with Tenable Nessus, an enterprise-grade automated assessment tool.
- TryHackMe | Nmap Live Host Discovery – Master active network sweeps using Nmap ICMP, ARP, and TCP probes to discover online hosts across subnets.
- TryHackMe | Nmap – Deep dive into Nmap port scanning types, service version detection, OS fingerprinting, and Nmap Scripting Engine (NSE) scripts.
- TryHackMe | TShark – Learn to perform command-line packet capture and deep network traffic analysis using TShark, the CLI version of Wireshark.
- TryHackMe | ffuf – Master ffuf (Fuzz Faster Fool), a high-speed web fuzzing utility used for directory discovery, virtual host enumeration, and parameter fuzzing.
- TryHackMe | Burp Suite: The Basics – Set up and configure Burp Suite to intercept, inspect, and modify HTTP/HTTPS traffic passing through an intercepting proxy.
- TryHackMe | Burp Suite: Repeater – Learn how to use Burp Repeater to manually modify HTTP requests, reissue them to a target server, and analyze the responses.

Container Security

Containers have revolutionized modern application deployment, but they also introduce unique security challenges if misconfigured. Container security spans the entire lifecycle—from securing container images and runtime environments to hardening orchestration platforms like Kubernetes. This section focuses on container isolation, Kubernetes runtime security monitoring, cluster hardening strategies, and practical techniques to prevent container breakout vulnerabilities.

- TryHackme | Containers - DoorDasher's Demise – Explore container security fundamentals and common misconfigurations through a scenario-based challenge.
- TryHackMe | K8s Runtime Security – Learn how to monitor container behavior, detect malicious activities, and enforce security policies in running Kubernetes workloads.
- TryHackMe | K8s Best Security Practices – Study core Kubernetes defense patterns, including Role-Based Access Control (RBAC), network policies, and Pod Security Standards.

- TryHackMe | Cluster Hardening – Master practical techniques to secure Kubernetes control planes, worker nodes, and internal API communication.

Cryptography & Hashes

Cryptography is the foundation of digital security, ensuring privacy, data integrity, and authentic communications across systems. However, implementing crypto incorrectly can lead to severe vulnerabilities that attackers can exploit to recover secrets or bypass authentication. This section introduces encryption algorithms, hash functions, cryptographic flaws, breaking RSA encryption, and practical hash cracking techniques using dictionary attacks and brute-force methods.

- TryHackMe | Cryptography Concepts – Build a fundamental understanding of symmetric/asymmetric encryption, hash algorithms, digital signatures, and Public Key Infrastructure (PKI).
- TryHackMe | Breaking Crypto the Simple Way – Learn how to identify and exploit common cryptographic implementation errors using mathematical and logical shortcuts.
- TryHackMe | Crypto Failures – Explore real-world cryptographic mistakes, including weak initialization vectors, reused keys, and insecure padding.
- TryHackMe | Breaking RSA – Dive into the mathematics behind RSA public-key cryptography and learn how to exploit weak keys and small prime numbers.
- TryHackMe | Cryptography for Dummies – A beginner-friendly introduction to ciphers, encoding standards, and classical cryptography methods.
- TryHackMe | Crack the hash – Practice identifying and cracking various hash types (MD5, SHA1, NTLM, bcrypt) using online databases and offline tools.
- TryHackMe | Crack The Hash Level 2 – Tackle advanced hash cracking challenges requiring custom wordlists, rulesets, and specialized cracking tools like Hashcat and John the Ripper.
- TryHackMe | Agent Sudo – Solve a beginner CTF machine involving User-Agent manipulation, steganography, hash cracking, and privilege escalation.
- TryHackMe | Brute It – Practice web directory enumeration, admin panel brute-forcing, hash cracking, and escalation techniques on a target machine.
- TryHackMe | Introduction to Cryptography – Learn how modern cryptographic algorithms protect sensitive data at rest and in transit across networks.

Steganography

Steganography is the art and science of concealing secret information within non-secret media, such as images, audio files, or video streams. Unlike cryptography—which obfuscates the content of a message so it cannot be read—steganography hides the very existence of the message itself. This section covers fundamental steganographic tools, data extraction techniques, hidden payload detection, and multi-layered cryptographic puzzles embedded inside digital media.

- TryHackMe | CC: Steganography – Learn how to use core steganography utilities like `steghide`, `exiftool`, `stegverify`, and `zsteg` to analyze and extract hidden data.
- TryHackMe | Cicada-3301 Vol:1 – Dive into an immersive mystery puzzle inspired by the famous Cicada 3301 internet ARG, focusing on advanced steganography and cryptography.
- TryHackMe | Musical Stego – Discover techniques for extracting hidden messages, embedded archives, and spectrograph images concealed inside audio files.
- TryHackMe | Madness – Solve a CTF machine that requires repairing corrupted image headers, uncovering hidden text, and conducting steganographic analysis.
- TryHackMe | Psycho Break – Practice image analysis, hash extraction, and multi-stage steganography challenges to retrieve credentials and compromise the target system.
- TryHackMe | Unstable Twin – Tackle a specialized CTF focused on image comparison, metadata inspection, and payload extraction from closely matching media files.

Web

Web applications form the primary attack surface for modern enterprise networks. Securing web platforms requires a deep understanding of HTTP request-response cycles, server-side logic, and client-side processing. This section explores classic and modern web vulnerabilities—including the OWASP Top 10, SQL Injection, Cross-Site Scripting (XSS), Server-Side Request Forgery (SSRF), Request Smuggling, and recent CVEs—alongside vulnerable practice labs like DVWA, WebGoat, and OWASP Juice Shop.

- TryHackme | CyberChef - Hoperation Save McSkidy – Use CyberChef to decode, transform, and analyze obfuscated web payloads in a Christmas-themed challenge.
- TryHackme | IDOR - Santa's Little IDOR – Learn to identify and exploit Insecure Direct Object Reference (IDOR) flaws by manipulating object identifiers in web requests.
- TryHackme | AWS Security - S3cret Santa – Discover techniques to enumerate public cloud assets, inspect misconfigured S3 buckets, and extract web secrets.
- TryHackme | React2Shell: CVE-2025-55182 – Analyze and exploit a remote code execution vulnerability affecting modern React server component environments.
- TryHackme | Race Conditions - Toy to The World – Understand business logic flaws where asynchronous concurrent requests create exploitable state windows.
- TryHackme | n8n: CVE-2025-68613 – Explore the impact of unauthenticated remote code execution in self-hosted workflow automation tools.
- TryHackme | Exploitation with cURL - Hoperation Eggsploit – Master using the `curl` command line utility to construct custom HTTP requests, headers, and payload injections.
- TryHackme | XSS - Merry XSSMas – Understand client-side JavaScript execution, session hijacking, and payload construction through a hands-on XSS challenge.
- TryHackMe | OWASP Top 10 2025: IAAA Failures – Examine modern Identification, Authentication, Authorization, and Accountability (IAAA) failure patterns in web services.

- TryHackMe | OWASP Top 10 2025: Application Design Flaws – Explore structural application design vulnerabilities and business logic flaws that bypass traditional code audits.
- TryHackMe | OWASP Top 10 2025: Insecure Data Handling – Learn how improper data validation, sensitive data exposure, and unsafe deserialization impact modern web applications.
- TryHackMe | WAF: Introduction – Understand Web Application Firewall (WAF) mechanics, inspection rulesets, and basic evasion concepts.
- TryHackMe | Chaining Vulnerabilities – Practice combining low-severity web bugs (e.g., self-XSS + CSRF) to achieve high-impact exploitation like Remote Code Execution (RCE).
- TryHackMe | Detecting Web Attacks – Analyze access logs and HTTP traffic patterns to spot malicious scanning, SQLi attempts, and path traversal probes.
- TryHackMe | Web Security Essentials – Build a fundamental understanding of web architecture, cookie flags, CORS policies, and browser security controls.
- TryHackMe | Microservices Architectures – Explore how web APIs interact across distributed microservice architectures and how security perimeters shift.
- TryHackMe | NoSQL Injection – Learn to manipulate non-relational database queries (such as MongoDB) to bypass login screens and extract data.
- TryHackMe | Advanced SQL Injection – Master second-order, error-based, time-based blind, and out-of-band (OOB) SQL injection techniques.
- TryHackMe | XSS – Deep dive into Stored, Reflected, and DOM-based Cross-Site Scripting vulnerabilities and their remediation strategies.
- TryHackMe | CSRF – Understand Cross-Site Request Forgery mechanics, token validation mechanisms, and SameSite cookie attributes.
- TryHackMe | File Inclusion, Path Traversal – Exploit Local File Inclusion (LFI), Remote File Inclusion (RFI), and path traversal to read sensitive system files.
- TryHackMe | HTTP Request Smuggling – Learn how discrepancies between front-end and back-end servers handling Content-Length and Transfer-Encoding lead to request smuggling.
- TryHackMe | HTTP/2 Request Smuggling – Understand advanced request smuggling attack vectors targeting modern HTTP/2 protocol implementations and downdraft translations.
- TryHackMe | SSRF – Learn how Server-Side Request Forgery vulnerabilities allow attackers to force web applications to make internal network requests.
- TryHackMe | OWASP Broken Access Control – Identify authorization enforcement failures, path manipulation bugs, and privilege escalation vulnerabilities.
- TryHackMe | HTTP in detail – Study web architecture, HTTP methods, response codes, parameters, headers, and cookies.
- TryHackMe | Vulnerabilities 101 – Understand software vulnerability scoring, CVE classifications, and how public exploits are documented.
- TryHackMe | Walking An Application – Manually inspect web application source code, developer comments, network tabs, and assets to discover hidden security details.

- TryHackMe | OWASP Top 10 - 2021 – Explore the critical web vulnerability categories outlined in the OWASP Top 10 2021 framework.
- TryHackMe | OWASP Top 10 – A practical walk-through of core web vulnerabilities, including SQLi, XSS, IDOR, and broken authentication.
- TryHackMe | OWASP Juice Shop – Practice finding and exploiting web vulnerabilities inside OWASP's intentionally insecure JavaScript web application.
- TryHackMe | OWASP Mutillidae II – Perform hands-on web application penetration testing on a PHP-based vulnerable target platform.
- TryHackMe | WebGOAT – Work through interactive web security lessons hosted on OWASP's Java-based training platform.
- TryHackMe | DVWA – Test your skills against Damn Vulnerable Web Application across different security difficulty settings.
- TryHackMe | VulnNet – Solve a practical web-focused CTF machine involving initial footprinting, vulnerability exploitation, and privilege escalation.
- TryHackMe | Juicy Details – Analyze web access log files to investigate past web application security breaches and trace attacker activity.
- TryHackMe | Vulnversity – Learn active reconnaissance, web directory scanning, upload form bypass, and system privilege escalation on a target host.
- TryHackMe | SQL Injection Lab – Practice extracting database contents manually through interactive, hands-on SQL injection scenarios.
- TryHackMe | SSTI – Learn to identify and exploit Server-Side Template Injection vulnerabilities across Jinja2, Twig, and Node.js engines.
- TryHackMe | SQL Injection – Understand the mechanics of SQL queries, input sanitization failures, and techniques to read or alter backend databases.
- TryHackMe | Basic Pentesting – Practice fundamental pentesting steps: enumeration, brute-forcing, web vulnerability exploitation, and hash cracking.
- TryHackMe | Ignite – Compromise a vulnerable Content Management System (CMS) target and elevate your privileges to root.
- TryHackMe | Overpass – Exploit weak web authentication logic, crack SSH keys, and elevate privileges via a vulnerable cron job.
- TryHackMe | Year of the Rabbit – Solve a multi-stage web CTF involving hidden web directories, file analysis, and local privilege escalation.
- TryHackMe | Develpy – Hack a vulnerable Python-based web service, obtain user flags, and abuse misconfigurations for root access.
- TryHackMe | Jack-of-All-Trades – Practice web enumeration, steganography, encoding decoding, and local privilege escalation on a beginner machine.
- TryHackMe | Bolt – Perform web reconnaissance, exploit default CMS configurations, and compromise an exposed machine.

Android

Mobile devices handle immense amounts of sensitive personal and corporate data, making mobile security a vital branch of cybersecurity. Android application security requires analyzing compiled application packages (APKs), identifying hardcoded secrets, bypassing security controls, and auditing IPC components. This section covers fundamental reverse engineering, static/dynamic analysis, and mobile exploitation concepts.

- TryHackMe | Android Hacking 101 – Learn the basics of Android architecture, APK file structure, reverse engineering with tools like `jadx` and `apktool`, and mobile app security risks.

Forensics

Digital Forensics and Incident Response (DFIR) focuses on investigating security breaches, analyzing digital evidence, and reconstructing attack timelines to understand how a system was compromised. When a security incident occurs, DFIR professionals examine volatile memory, disk images, system event logs, and registry keys to isolate malicious activity and limit operational damage. This section covers core forensic concepts across Windows, Linux, and macOS environments, memory analysis, log triage, and incident response frameworks.

- TryHackMe | DFIR: An Introduction – Understand the core principles, methodologies, and phases of Digital Forensics and Incident Response.
- TryHackMe | Critical – Investigate a critical security incident by analyzing disk artifacts and tracking attacker actions step-by-step.
- TryHackMe | Windows Incident Surface – Learn where Windows stores forensically rich artifacts, including event logs, prefetched files, and registry hives.
- TryHackMe | Introduction To Honeypots – Discover how decoy systems are deployed to lure attackers, record threat tactics, and gather early threat intelligence.
- TryHackMe | Geolocating Images – Practice analyzing image EXIF metadata and visual cues to determine the exact location where a photo was taken.
- TryHackMe | Compromised Windows Analysis – Perform a post-compromise investigation on a Windows endpoint using forensic tools to reconstruct the adversary's actions.
- TryHackMe | AppSec IR – Explore how incident responders handle security incidents originating from web application attacks.
- TryHackMe | IR Playbooks – Learn how standardized incident response playbooks guide SOC teams through containment, eradication, and recovery.
- TryHackMe | Intro to Endpoint Security – Discover how host-based protection tools, EDR solutions, and logging mechanisms monitor endpoint health.
- TryHackMe | IR Timeline Analysis – Master techniques to build chronological event timelines from system logs to trace an attack's progression.
- TryHackMe | macOS Forensics: Artefacts – Examine system logs, plist files, and unified logs to conduct forensic investigations on macOS devices.

- TryHackme | Forensics - Registry Furensics – Inspect Windows Registry keys to uncover persistence mechanisms, user activity, and recently connected USB devices.
- TryHackme | Phishing - Phishmas Greetings – Analyze email headers, attached files, and embedded URLs to dissect malicious phishing campaigns.
- TryHackme | YARA Rules - YARA mean one! – Learn to write custom YARA rules to detect, classify, and hunt for malware patterns across file systems.
- TryHackme | Web Attack Forensics - Drone Alone – Analyze web server access logs and database queries to reconstruct web-based intrusion attempts.
- TryHackMe | Linux Threat Detection 1 – Learn how to identify malicious behavior, unauthorized processes, and persistence on Linux hosts.
- TryHackMe | Linux Logging for SOC – Understand Linux system log formats (`/var/log`), syslog configurations, and log monitoring methods.
- TryHackMe | SOC Role in Blue Team – Explore the responsibilities, daily workflows, and collaborative functions of Security Operations Center (SOC) analysts.
- TryHackMe | Session Forensics – Investigate network session captures, cookie tokens, and active connections to trace hijacked sessions.
- TryHackMe | Windows Logging for SOC – Master Windows Event Log IDs (Security, System, Application) critical for monitoring security incidents.
- TryHackMe | Mobile Acquisition – Learn logical and physical forensic acquisition techniques to extract data from iOS and Android devices.
- TryHackMe | Volatility Essentials – Get started with Volatility, the industry-standard framework for extracting artifacts from raw RAM dumps.
- TryHackMe | Memory Analysis Introduction – Understand volatile memory structures and why RAM capture is essential for uncovering malware hiding in execution space.
- TryHackMe | MS Sentinel: Just Looking – Explore Microsoft Sentinel SIEM/SOAR platform basics, running queries to hunt for security threats.
- TryHackMe | SOC L1 Alert Triage – Practice first-line alert evaluation, distinguishing true positives from benign false alarms.
- TryHackMe | SOC L1 Alert Reporting – Learn how to write clear, actionable incident reports for technical leads and management following an alert investigation.
- TryHackMe | macOS Forensics: The Basics – Get introduced to Apple system architecture, APFS file system nuances, and basic macOS evidence collection.
- TryHackMe | FAT32 Analysis – Examine the structure of the FAT32 file system, analyzing file allocation tables and unallocated space.
- TryHackMe | MBR and GPT Analysis – Understand disk partitioning structures (Master Boot Record vs. GUID Partition Table) during low-level disk analysis.
- TryHackMe | Supply Chain Attack: Lottie – Investigate how third-party dependencies and supply chain compromises insert malicious code into trusted software.

- TryHackMe | Incident Response Process – Learn the standardized NIST and SANS Incident Response lifecycle stages: Preparation, Detection, Containment, Eradication, Recovery, and Lessons Learned.
- TryHackMe | Linux Incident Surface – Identify critical Linux system locations, cron jobs, environment variables, and startup scripts frequently targeted for exploitation.
- TryHackMe | Intro to Cold System Forensics – Learn static, dead-box forensic acquisition techniques on powered-off storage media.
- TryHackMe | Forensic Imaging – Master creating bit-stream forensic disk images (E01, RAW/DD) while maintaining chain of custody and hash integrity.
- TryHackMe | IR Philosophy and Ethics – Explore ethical principles, legal standards, and professional integrity required during forensic investigations.
- TryHackMe | Windows Applications Forensics – Inspect application-specific artifacts like web browser histories, SQLite databases, and communication logs.
- TryHackMe | Legal Considerations in DFIR – Understand data privacy laws, regulatory compliance requirements, and courtroom admissibility standards for digital evidence.
- TryHackMe | Servidae: Log Analysis in ELK – Practice ingesting, searching, and analyzing system event logs using the Elasticsearch, Logstash, and Kibana (ELK) stack.
- TryHackMe | Identification & Scoping – Learn how to define the blast radius and operational scope of a security compromise during early IR phases.
- TryHackMe | Digital Forensics Case B4DM755 – Solve an end-to-end digital forensics case by putting disk image analysis and artifact recovery skills into practice.
- TryHackMe | Linux Server Forensics – Investigate compromised Linux server infrastructure, examining web server logs, SSH connections, and modified binaries.
- TryHackMe | Forensics – Apply fundamental digital forensics tools to extract hidden artifacts and analyze unknown file types.
- TryHackMe | Memory Forensics – Analyze volatile memory captures to extract active network sockets, injected DLLs, and unencrypted passwords.
- TryHackMe | Volatility – Reinforce memory analysis skills using core Volatility plugins to inspect process trees and command-line arguments.
- TryHackMe | Disk Analysis & Autopsy – Master Autopsy, a popular open-source digital forensics platform, to analyze drive images and recover deleted files.

Wi-Fi Hacking

Wireless networks rely on radio waves rather than physical cables, extending the network perimeter beyond physical building boundaries. Securing Wi-Fi environments requires understanding wireless frames, encryption protocols, and authentication schemes. This section covers wireless packet sniffing, rogue access points, and common vulnerabilities present in Wi-Fi security standards.

- TryHackMe | Wifi Hacking 101 – Learn the fundamentals of IEEE 802.11 wireless protocols, packet interception, WPA/WPA2 handshakes, and cracking methods using the Aircrack-ng suite.

Reverse Engineering

Reverse engineering is the process of deconstructing software binaries to understand their internal logic, control flow, and functionality without access to original source code. It is an indispensable skill for vulnerability research, exploit development, software auditing, and malware analysis. This section introduces low-level assembly language, CPU architecture, executable file formats, and industry-standard reverse engineering frameworks like Ghidra and Radare2.

- TryHackMe | Intro to x86-64 – Master x86-64 CPU architecture fundamentals, exploring CPU registers, instruction sets, and memory stack execution.
- TryHackMe | Windows x64 Assembly – Learn 64-bit Windows assembly syntax, function calling conventions, and register usage required for binary analysis.
- TryHackMe | Reverse Engineering – Get introduced to reverse engineering methodologies, dynamic debugging, and static code disassembly techniques.
- TryHackMe | Reversing ELF – Analyze Linux Executable and Linkable Format (ELF) binaries to understand execution flow and extract hidden logic.
- TryHackMe | JVM Reverse Engineering – Learn how to decompile and analyze Java Virtual Machine (JVM) bytecode and `.class` files.
- TryHackMe | CC: Radare2 – Master Radare2, a powerful command-line framework for disassembling, debugging, and inspecting binary files.
- TryHackMe | CC: Ghidra – Learn how to navigate Ghidra, the open-source software reverse engineering framework created by the NSA, using its decompiler and code browser.
- TryHackMe | Aster – Reverse engineer a compiled binary challenge to uncover hidden validation logic and retrieve the flag.
- TryHackMe | Classic Passwd – Analyze a password validation binary using static and dynamic reverse engineering tools to crack its logic.
- TryHackMe | REloaded – Tackle a series of progressive binary reverse engineering challenges designed to build disassembly fluency.

Malware Analysis

Malware analysis involves dissecting malicious software—such as ransomware, trojans, rootkits, and info-stealers—to understand its behavior, capability, origin, and impact. Security analysts use static analysis (examining code without running it) and dynamic analysis (executing sample files inside isolated sandboxes) to extract Indicators of Compromise (IoCs) and build detection rules. This section covers sandbox analysis, code obfuscation, EDR evasion, and advanced threat techniques.

- TryHackMe | Android Malware Analysis – Learn how to static-analyze malicious APK files, inspecting permissions, reverse-engineered Java code, and suspicious Android endpoints.
- TryHackMe | ret2libc – Explore advanced binary exploitation by bypassing Non-Executable (NX) memory protections using Return-to-Libc attack vectors.
- TryHackMe | Bypass Really Simple Security – Analyze software security controls and learn techniques to identify implementation bypasses.
- TryHackMe | Snyk Code – Perform automated code analysis to identify potential security vulnerabilities within software applications.
- TryHackMe | Attacking ICS Plant #1 – Explore Industrial Control Systems (ICS/SCADA) security, dissecting operational technology protocols and threat vectors targeting physical infrastructure.
- TryHackMe | Linux Function Hooking – Understand how malicious userland rootkits hijack library calls using dynamic linker capabilities (`LD_PRELOAD`).
- TryHackMe | Linux Process Analysis – Inspect running processes, process memory maps, and file descriptors on Linux systems to identify hidden malware execution.
- TryHackMe | ParrotPost: Phishing Analysis – Analyze email artifacts and malicious attachments to understand how initial access payloads deliver malware.
- TryHackMe | Hosted Hypervisors – Understand virtualization architecture and how sandbox isolation protects host environments during live malware execution.
- TryHackMe | TShark – Inspect command-line network captures to isolate malware Command and Control (C2) communication channels.
- TryHackMe | AttackerKB – Learn to leverage crowdsourced threat intelligence platforms to evaluate vulnerability exploitability and impact.
- TryHackMe | Hip Flask – Perform reverse engineering and memory inspection on a target host to trace persistent threat payloads.
- TryHackMe | Outlook NTLM Leak – Understand how email components and malicious OLE objects force Outlook clients to leak NTLM password hashes over the network.
- TryHackMe | Hacking Hadoop – Analyze misconfigurations and exploitation paths in Apache Hadoop big data cluster environments.
- TryHackme | Malware Analysis - Egg-xecutable – Execute and observe suspicious Windows executables inside isolated sandbox environments to extract dynamic IoCs.
- TryHackme | SOC Alert Triaging - Tinsel Triage – Investigate cloud security alerts and trace malware propagation across Azure environments.
- TryHackme | Obfuscation - The Egg Shell File – Deobfuscate heavily encoded PowerShell scripts, batch scripts, and command chains used by adversaries to evade static scanners.
- TryHackme | Malware Analysis - Malhare.exe – Analyze malicious HTML Applications (.HTA) and embedded PowerShell scripts used in initial access campaigns.
- TryHackMe | Introduction to EDR – Explore Endpoint Detection and Response (EDR) architecture, user-mode API hooking, and endpoint telemetry collection.

- TryHackMe | Malware Classification – Learn how security teams classify samples into malware families (Ransomware, Spyware, Keyloggers, Trojans) based on behavior and code similarity.
- TryHackMe | File and Hash Threat Intel – Query threat intelligence platforms like VirusTotal and Hybrid Analysis using cryptographic hashes (MD5, SHA256) to identify known malware files.
- TryHackMe | APT28 Inception Theory – Study the Advanced Persistent Threat group APT28 (Fancy Bear), analyzing their custom malware toolsets, campaign tactics, and infrastructure.
- TryHackMe | Intro to Detection Engineering – Learn how security engineers build, test, and tune detection rules (Sigma, Snort, YARA) to catch malicious behavior.
- TryHackMe | History of Malware – Explore the evolution of malicious code, tracing notable historical outbreaks from early boot-sector viruses to modern wormable ransomware.
- TryHackMe | MAL: Malware Introductory – Understand foundational malware analysis methodologies, safety precautions, and isolated lab setups.
- TryHackMe | Basic Malware RE – Practice basic static reverse engineering on compiled PE files using tools like PEStudio, Strings, and disassemblers.
- TryHackMe | MAL: Researching – Learn how malware analysts research unknown samples, track threat actor groups, and write technical intelligence reports.
- TryHackMe | Mobile Malware Analysis – Examine specialized dynamic and static analysis techniques tailored for uncovering Android and iOS mobile malware.
- TryHackMe | Carnage – Investigate network packet captures to uncover Command and Control (C2) traffic generated by Cobalt Strike and other C2 frameworks.
- TryHackMe | Dunkle Materie – Perform a deep forensic and malware investigation on a heavily compromised enterprise system.

PrivEsc

Privilege Escalation (PrivEsc) occurs when an attacker exploits a bug, design flaw, or configuration oversight to gain elevated access to resources normally protected from an application or user. Once an initial foothold is secured on a Linux or Windows target, escalating privileges to `root` or `SYSTEM` is essential for full control, persistence, and lateral movement. This section covers enumeration scripts, kernel exploits, misconfigured SUID binaries, insecure `sudo` permissions, unquoted service paths, and token manipulation.

- TryHackMe | Linux Privilege Escalation – Learn fundamental Linux privilege escalation vectors, including SUID/SGID files, cron jobs, wildcard injection, and capabilities.
- TryHackMe | Linux PrivEsc – Explore practical privilege escalation techniques using automated enumeration scripts, misconfigured sudo rights, and writable `/etc/passwd`.
- TryHackMe | Linux PrivEsc Arena – Practice escalating privileges across a hands-on sandbox environment featuring diverse Linux misconfigurations.
- TryHackMe | Windows PrivEsc – Learn key Windows escalation pathways, including unquoted service paths, weak service permissions, DLL hijacking, and AlwaysInstallElevated.

- TryHackMe | Windows PrivEsc Arena – Put Windows privilege escalation theories into practice inside an arena tailored for local system exploitation.
- TryHackMe | Linux Agency – Investigate system misconfigurations and user permission flaws to escalate privileges across a multi-user Linux host.
- TryHackMe | Sudo Security Bypass – Analyze and exploit historic CVEs in the sudo utility that allowed local users to execute commands as root (CVE-2019-14287).
- TryHackMe | Sudo Buffer Overflow – Explore buffer overflow vulnerabilities in sudo (such as Baron Samedit / CVE-2021-3156) to gain root shell access.
- TryHackMe | Blaster – Exploit a vulnerable Windows target via Remote Desktop Protocol (RDP) and escalate privileges using CVE-2019-1388.
- TryHackMe | Ignite – Gain initial access through a vulnerable CMS and locate local system misconfigurations to escalate privileges to root.
- TryHackMe | Kenobi – Exploit ProFTPD vulnerabilities, gain initial access via SSH, and abuse a custom SUID binary for root privileges.
- TryHackMe | c4ptur3-th3-fl4g – Solve a multi-task beginner challenge covering encoding, steganography, hash cracking, and privilege escalation basics.
- TryHackMe | Pickle Rick – A beginner-friendly CTF requiring web enumeration, file inspection, and simple sudo exploitation to gain root access.

AI

As Artificial Intelligence (AI) and Machine Learning (ML) models become deeply integrated into software products, they introduce entirely new security attack surfaces. AI security covers both defending machine learning pipelines and auditing LLM-driven applications against adversarial attacks. This section explores prompt injection techniques, training data poisoning, model inversion, output manipulation, and the integration of AI tools into modern cybersecurity workflows.

- TryHackme | Prompt Injection - Sched-yule conflict – Learn how attackers manipulate Large Language Model (LLM) prompts to bypass system guardrails in a scenario-based challenge.
- TryHackme | AI in Security - old sAInt nick – Explore how AI and machine learning models assist threat detection and automate SOC triage workflows.
- TryHackMe | AI/ML Security Threats – Examine the OWASP Top 10 for LLMs, including data poisoning, model inversion, supply chain vulnerabilities, and sensitive information disclosure.
- TryHackMe | Input Manipulation & Prompt Injection – Master direct and indirect prompt injection attacks to hijack AI agent behaviors and bypass system context constraints.

Windows

Windows environments underpin enterprise IT infrastructure, making Windows monitoring, logging, and threat detection vital skills for security analysts. Defending Windows hosts requires understanding telemetry sources, registry mechanics, cloud integration (Entra ID, Microsoft

365, Intune), and forensic investigation of compromised systems. This section covers defensive monitoring, registry persistence detection, incident investigation, and classic exploitation scenarios on Windows systems.

- TryHackMe | M365 Monitoring Basics – Learn how to track user activity, configure security alerts, and inspect audit logs within Microsoft 365 tenant environments.
- TryHackMe | Monitoring Active Directory – Explore event logging and auditing mechanisms used to detect suspicious Kerberos, NTLM, and domain changes in Active Directory.
- TryHackMe | Entra ID Monitoring – Monitor identity events, conditional access logs, and sign-in activities across cloud-based Entra ID (formerly Azure AD).
- TryHackMe | Microsoft Intune Monitoring – Understand device management telemetry, compliance policies, and endpoint health monitoring using Microsoft Intune.
- TryHackMe | Windows Logging for SOC – Master key Windows Event Log IDs (Security, System, Sysmon) required for effective threat hunting and incident triage.
- TryHackMe | Windows Threat Detection 1 – Learn how to identify malicious process creation, suspicious network connections, and command-line execution on Windows.
- TryHackMe | XDR: Introduction – Understand Extended Detection and Response (XDR) architecture, unifying endpoint, network, and cloud telemetry.
- TryHackMe | Windows Incident Surface – Inspect core Windows system locations and registry hives commonly abused by threat actors during an attack.
- TryHackMe | Registry Persistence Detection – Analyze Windows Registry Run keys, Services, and Startup locations to spot attacker persistence mechanisms.
- TryHackMe | Investigating Windows – Conduct a live-response forensic investigation on a compromised Windows host to identify malicious artifacts.
- TryHackMe | Investigating Windows 2.0 – Deepen your threat hunting skills by analyzing event logs, scheduled tasks, and rogue network connections.
- TryHackMe | Investigating Windows 3.x – Solve an advanced Windows digital forensics scenario, tracking sophisticated attacker lateral movement and data exfiltration.
- TryHackMe | Blueprint – Exploit a vulnerable web service hosted on Windows, extract password hashes, and crack them to gain administrator access.
- TryHackMe | VulnNet: Active – Practice Active Directory enumeration, GPO exploitation, and ticket attacks on a vulnerable enterprise network environment.
- TryHackMe | Anthem – Perform OSINT and web reconnaissance on a Windows target, extract credentials, and gain access via RDP.
- TryHackMe | Blue – Exploit the infamous EternalBlue vulnerability (CVE-2017-0144 / MS17-010) to gain immediate `NT AUTHORITY\SYSTEM` shell access.

Active Directory

Active Directory (AD) is the identity backbone used by over 90% of Fortune 500 companies to manage users, computers, and access rights across networks. Consequently, AD security is a primary focus for both Red Teams and Blue Teams. This section explores Active Directory architecture, domain enumeration, certificate services (AD CS) misconfigurations, Kerberos attacks (AS-REP Roasting, Kerberoasting), domain hardening, and privilege escalation to Domain Admin.

- TryHackMe | AD Certificate Templates – Learn to identify and exploit misconfigured Active Directory Certificate Services (AD CS) templates (ESC1, ESC2, ESC3) for domain takeover.
- TryHackMe | Active Directory Basics – Master fundamental AD concepts, including Domain Controllers, Forests, Trees, OUs, Users, Groups, and Group Policy Objects (GPOs).
- TryHackMe | AD: Basic Enumeration – Learn how to query Active Directory domain objects using PowerView, BloodHound, and native PowerShell commands.
- TryHackMe | Active Directory Hardening – Explore defensive strategies, secure GPO configurations, and Tiered Administration models to protect AD infrastructure.
- TryHackMe | Attacktive Directory – Perform full-scope Active Directory exploitation using Kerbrute, Impacket, BloodHound, and pass-the-hash attacks.
- TryHackMe | Post-Exploitation Basics – Learn post-compromise enumeration, credential dumping (lsass/Mimikatz), and lateral movement tools on Windows networks.
- TryHackMe | USTOUN – Practice enumerating DC services, cracking passwords, and elevating privileges on a domain-joined machine.
- TryHackMe | Enterprise – Hack a corporate enterprise environment by exploiting web services, pivoting through internal networks, and compromising the Active Directory Domain Controller.
- TryHackMe | RazorBlack – Perform Kerberos roasting, AS-REP roasting, NFS share enumeration, and domain privilege escalation on an Active Directory CTF machine.

PCAP Analysis

Packet Capture (PCAP) analysis involves inspecting recorded network traffic to identify suspicious communications, investigate security breaches, and reconstruct attacker behavior. When an incident occurs, network captures reveal unencrypted credentials, Command and Control (C2) channels, data exfiltration, and lateral movement across hosts. This section focuses on using tools like Wireshark and TShark to analyze traffic captures, inspect protocol headers, and trace malicious network flows.

- TryHackMe | h4cked – Analyze a PCAP file from a compromised web server to trace how an attacker gained access, executed commands, and transferred files.
- TryHackMe | Carnage – Inspect network traffic logs to identify Cobalt Strike C2 beacons, extract malicious IP addresses, and analyze exfiltrated data.

- TryHackMe | CCT2019 – Practice packet analysis techniques to solve network-based forensic challenges from a capture file.
- TryHackMe | Overpass 2 - Hacked – Investigate a full network intrusion by analyzing a PCAP capture file to discover how the server was breached and backdoored.

Buffer Overflow

Buffer Overflow vulnerabilities occur when a program writes more data to a memory buffer than it was allocated to hold, overflowing into adjacent memory space. This can corrupt control flow pointers (such as the Instruction Pointer `EIP/RIP`) and allow an attacker to execute arbitrary shellcode. Understanding memory stack layout, offset calculation, bad character identification, and Return Oriented Programming (ROP) is essential for low-level exploit development. This section covers step-by-step buffer overflow methodologies across 32-bit and 64-bit binaries.

- TryHackMe | Buffer Overflow Prep – Master the standardized 10-step process for 32-bit Windows stack buffer overflow exploitation using Immunity Debugger and Mona.py.
- TryHackMe | Gatekeeper – Practice discovering and exploiting a stack-based buffer overflow vulnerability in a vulnerable Windows service.
- TryHackMe | Chronicle – Tackle a challenging binary exploitation room requiring memory corruption analysis and custom exploit crafting.
- TryHackMe | Intro To Pwntools – Learn to leverage the Python `pwntools` framework for rapid buffer overflow development, dynamic shellcode generation, and remote service interaction.

Easy CTF

Capture The Flag (CTF) challenges offer a practical, gamified environment to test offensive security and threat analysis skills against real target environments. The **Easy CTF** collection features beginner-friendly machines designed to reinforce fundamental concepts: basic port scanning, web directory enumeration, exploiting known CVEs, hash cracking, and introductory privilege escalation. Completing these machines helps build the problem-solving mindset needed for real-world security assessments.

- TryHackMe | Toolbox: Vim – Learn essential terminal text editing skills using Vim to quickly modify configuration files and scripts during CTF challenges.
- TryHackMe | DFIR: An Introduction – Explore foundational concepts of digital forensics and incident response applied within forensic CTF scenarios.
- TryHackMe | The Phishing Pond – Analyze email artifacts and headers to investigate phishing attempts and uncover hidden flags.
- TryHackMe | Oracle 9 – Perform network enumeration, exploit exposed services, and elevate privileges on an Oracle-themed target.
- TryHackMe | Soupedecode 01 – Solve encoding and decoding puzzles to extract hidden flags from multi-layered data streams.

- TryHackMe | Billing – Exploit vulnerabilities in a web-based billing platform to obtain initial access and escalate privileges.
- TryHackMe | Light – Perform web application enumeration and exploit basic injection flaws on a lightweight target host.
- TryHackMe | Lo-Fi – Identify and exploit Local File Inclusion (LFI) vulnerabilities to read sensitive files and compromise the server.
- TryHackMe | Silver Platter – Practice beginner-friendly web enumeration and exploit exposed software misconfigurations.
- TryHackMe | The Sticker Shop – Audit an e-commerce platform for web security vulnerabilities to retrieve hidden flags.
- TryHackMe | Lookup – Exploit domain name lookup utilities and command injection bugs to gain access to the host machine.
- TryHackMe | Threat Hunting With YARA – Construct custom YARA rules to detect, classify, and isolate malicious file samples.
- TryHackMe | Whiterose – Solve an interactive, web-centric CTF challenge inspired by Mr. Robot themes.
- TryHackMe | Pyrat – Explore remote Python execution services and exploit script flaws to capture user and root flags.
- TryHackMe | Cheese CTF – Practice basic web exploitation, file manipulation, and local privilege escalation.
- TryHackMe | U.A. High School – Enumerate hidden web endpoints, exploit web parameters, and escalate privileges on an anime-themed machine.
- TryHackMe | Joomify – Audit a Joomla CMS installation to exploit known web vulnerabilities (CVE-2023-23752) and extract credentials.
- TryHackMe | Critical – Investigate critical target misconfigurations to compromise web services and gain administrative control.
- TryHackMe | Publisher – Exploit vulnerable publishing software, inspect system binaries, and achieve root access.
- TryHackMe | W1seGuy – Solve cryptographic challenges involving XOR ciphers to recover encryption keys and flags.
- TryHackMe | mKingdom – Compromise a target running a vulnerable CMS, elevate privileges, and explore internal misconfigurations.
- TryHackMe | Linux Process Analysis – Inspect running processes, memory spaces, and environment variables to uncover hidden malicious activity.
- TryHackMe | CyberLens – Exploit a vulnerable image processing service hosted on Windows to gain remote shell access.
- TryHackMe | TryHack3M: Bricks Heist – Investigate a compromised WordPress site and perform incident response steps to uncover attacker activities.

- TryHackMe | Creative – Enumerate subdomains, exploit local file inclusions, and abuse local path configurations for privilege escalation.
- TryHackMe | Eviction – Analyze threat telemetry and event logs to trace an attacker's steps during an eviction investigation.
- TryHackMe | Probe – Audit exposed network ports and services to find exploitation vectors on a target host.
- TryHackMe | Dreaming – Inspect database records and Python scripts to pivot across multiple user accounts to root.
- TryHackMe | The Witch's Cauldron – Solve multi-stage riddles combining web vulnerability exploitation, steganography, and cryptography.
- TryHackMe | Bulletproof Penguin – Strengthen Linux system hardening awareness by auditing common operational misconfigurations.
- TryHackMe | Hijack – Exploit weak service configurations, hijack dynamic libraries, and gain root access.
- TryHackMe | Compiled – Practice decompiling binary executables to locate hardcoded credentials and key verification logic.
- TryHackMe | Super Secret Tip – Decode hidden messages, bypass web forms, and complete a multi-step CTF puzzle.
- TryHackMe | Lesson Learned? – Analyze security post-mortems and investigate logs to discover how system flaws were exploited.
- TryHackMe | Grep – Practice using `grep` and regular expressions to search through massive log files for sensitive data and flags.
- TryHackMe | Red – Exploit a misconfigured Redis database instance to achieve Remote Code Execution (RCE) on the server.
- TryHackMe | Snapped "Phish"-ing Line – Conduct incident response on a phishing campaign by analyzing email attachments and malicious URLs.
- TryHackMe | Cat Pictures 2 – Perform web enumeration, inspect image metadata, and escalate local privileges on a Linux host.
- TryHackMe | Flip – Learn to perform cryptographic bit-flipping attacks against CBC mode encryption to bypass authentication.
- TryHackMe | Valley! – Enumerate hidden directories, inspect exposed API endpoints, and escalate privileges via misconfigured system scripts.
- TryHackMe | Capture! – Bypass web authentication forms using custom brute-force scripts and parameter manipulation.
- TryHackMe | Opacity – Bypass web upload restrictions to achieve a reverse shell and exploit local file permissions for root.
- TryHackMe | LookBack – Audit Windows event logs and IIS web logs to trace an attacker's initial access vector.

- TryHackMe | Bugged – Intercept and decode IoT communication protocols (MQTT) to extract sensitive flags.
- TryHackMe | GamingServer – Practice web directory enumeration, SSH key cracking, and abusing `lxd` container group permissions for root.
- TryHackMe | Confidential – Inspect confidential PDF documents and extract embedded flags using forensic tools.
- TryHackMe | OverlayFS - CVE-2021-3493 – Learn about and exploit the Linux Kernel OverlayFS vulnerability to instantly escalate local privileges to root.
- TryHackMe | Psycho Break – Solve steganography puzzles, crack password hashes, and escalate privileges on a horror-themed CTF host.
- TryHackMe | Bounty Hacker – Enumerate exposed FTP services, brute-force SSH logins, and abuse sudo permissions on a Cowboy Bebop-themed machine.
- TryHackMe | Fowsniff CTF – Perform OSINT, gather leaked credentials, brute-force POP3 email accounts, and exploit local misconfigurations.
- TryHackMe | RootMe – A classic beginner CTF machine involving web directory fuzzing, file upload bypass, and SUID privilege escalation.
- TryHackMe | AttackerKB – Learn to leverage public vulnerability intelligence to find and execute exploits against vulnerable software.
- TryHackMe | Pickle Rick – Exploit a web application to execute system commands and locate hidden ingredients across the file system.
- TryHackMe | c4ptur3-th3-fl4g – A beginner challenge covering binary conversion, ciphers, steganography, and basic file analysis.
- TryHackMe | Library – Brute-force SSH credentials, analyze Python scripts, and abuse sudo rights for root escalation.
- TryHackMe | Thompson – Exploit an exposed Apache Tomcat manager application to upload a malicious WAR web shell.
- TryHackMe | Simple CTF – Exploit an outdated CMS vulnerability (CMS Made Simple), crack password hashes, and escalate privileges via sudo.
- TryHackMe | LazyAdmin – Discover exposed admin panels, exploit CMS backup files, and abuse sudo permissions to execute arbitrary scripts.
- TryHackMe | Anonforce – Inspect anonymous FTP shares, extract encrypted password hashes (`/etc/shadow`), and crack them offline.
- TryHackMe | Ignite – Exploit a remote code execution bug in Fuel CMS 1.4.1 to gain a shell and locate the root flag.
- TryHackMe | Wgel CTF – Discover hidden SSH keys via web directory fuzzing and abuse sudo `wget` capabilities for root access.
- TryHackMe | Kenobi – Enumerate Samba shares, exploit ProFTPD version 1.3.5, manipulate NFS mounts, and abuse SUID binaries.

- TryHackMe | Dav – Exploit default credentials on a WebDAV server to upload a PHP web shell and compromise the host.
- TryHackMe | Ninja Skills – Practice Linux command-line mastery by searching, filtering, and locating hidden files across a file system.
- TryHackMe | Ice – Exploit a vulnerable Icecast streaming server on Windows, run post-exploitation modules, and dump passwords via Mimikatz.
- TryHackMe | Lian_Yu – Perform web enumeration, steganography extraction, and file analysis on an Arrow-themed CTF machine.
- TryHackMe | The Cod Caper – Exploit a web vulnerability, recover SSH credentials, and execute a local buffer overflow exploit.
- TryHackMe | Blaster – Perform web reconnaissance, gain access via RDP, and escalate privileges using a Windows GUI vulnerability.
- TryHackMe | Encryption - Crypto 101 – Master foundational cryptographic concepts, ciphers, key exchanges, and hash verification algorithms.
- TryHackMe | Brooklyn Nine Nine – Exploit basic web vulnerabilities, anonymous FTP access, or weak SSH logins on a TV-themed room.
- TryHackMe | Year of the Rabbit – Fuzz web directories, inspect hidden JavaScript files, analyze hydra outputs, and elevate local user privileges.
- TryHackMe | Jack-of-All-Trades – Solve a multi-discipline challenge involving base64 decoding, steganography, and local privilege escalation.
- TryHackMe | Madness – Fix corrupted image headers, uncover hidden steganography keys, and elevate privileges on a Linux host.
- TryHackMe | KoTH Food CTF – Practice King of the Hill (KoTH) style fast-paced machine exploitation and system hardening techniques.
- TryHackMe | Easy Peasy – Practice web directory enumeration, hidden port scanning, cracked hash analysis, and cron job exploitation.
- TryHackMe | Tony the Tiger – Exploit Java deserialization vulnerabilities, decode secret payloads, and elevate privileges to root.
- TryHackMe | CTF collection Vol.1 – Complete a collection of beginner micro-challenges spanning decoding, steganography, and reverse engineering.
- TryHackMe | Smag Grotto – Intercept network traffic, analyze pcap logs, extract credentials, and exploit cron jobs for root access.
- TryHackMe | Couch – Exploit an unauthenticated Apache CouchDB database server to achieve command execution and system access.
- TryHackMe | Source – Exploit a remote code execution vulnerability in Webmin (CVE-2019-15107) to instantly gain root privileges.
- TryHackMe | Overpass – Bypass front-end JavaScript authentication, crack encrypted SSH keys, and exploit misconfigured cron jobs.

- TryHackMe | Gotta Catch'em All! – Solve a Pokémon-themed CTF by exploiting web applications and searching the Linux file system for hidden flags.
- TryHackMe | Bolt – Conduct web reconnaissance on a Bolt CMS target, exploit default settings, and capture user flags.
- TryHackMe | kiba – Exploit a prototype pollution vulnerability in Kibana (CVE-2019-7609) to achieve remote code execution.
- TryHackMe | Poster – Enumerate exposed PostgreSQL database instances, dump database credentials, and execute system commands.
- TryHackMe | Chocolate Factory – Solve a Willy Wonka-themed room involving web form exploitation, key decoding, and SUID privilege escalation.
- TryHackMe | Startup – Abuse writable FTP shares to upload a reverse shell, analyze pcap files, and hijack execution scripts.
- TryHackMe | Chill Hack – Exploit command injection vulnerabilities on a web application, pivot through user accounts, and abuse Docker group permissions.
- TryHackMe | ColddBox: Easy – Enumerate a WordPress site, brute-force admin logins, upload a PHP web shell, and exploit sudo permissions.
- TryHackMe | GLITCH – Inspect client-side JavaScript, exploit NodeJS execution, and elevate privileges on a glitch-themed target.
- TryHackMe | All in One – Practice web enumeration, WordPress exploitation, and local privilege escalation on a single target machine.
- TryHackMe | Archangel – Exploit LFI vulnerabilities, poison log files to achieve RCE, and hijack environment PATH variables for root access.
- TryHackMe | Cyborg – Perform web fuzzing, extract Borg backup archives, crack passwords, and abuse misconfigured admin scripts.
- TryHackMe | Lunizz CTF – Enumerate web ports, extract hidden SQL database contents, and elevate system access.
- TryHackMe | Badbyte – Perform port scanning, analyze dynamic malware samples, and exploit local misconfigurations.
- TryHackMe | Team – Exploit LFI vulnerabilities in domain headers, leverage SSH keys, and abuse SUID scripts for root access.
- TryHackMe | VulnNet: Node – Exploit insecure deserialization flaws in a Node.js web application and elevate privileges via npm.
- TryHackMe | VulnNet: Internal – Audit internal network services (Samba, Redis, Rsync) to extract credentials and compromise the machine.
- TryHackMe | Atlas – Enumerate exposed services, crack credential hashes, and elevate privileges on a Linux CTF host.
- TryHackMe | VulnNet: Roasted – Exploit Active Directory misconfigurations, perform Kerberoasting/AS-REP roasting, and compromise the domain.

- TryHackMe | Cat Pictures – Exploit an open-source web application, pivot to local system services, and gain root access.
- TryHackMe | Mustacchio – Extract HSQL database files, crack admin hashes, inspect SSH keys, and exploit custom SUID binaries.

Medium CTF

The **Medium CTF** challenges bridge the gap between basic vulnerability exploitation and complex, multi-stage network compromise. These rooms move beyond straightforward flaws, requiring you to chain multiple vulnerabilities together—such as combining blind SQL injection, custom API bypasses, internal network pivoting, binary exploitation, and advanced privilege escalation. Mastering Medium rooms develops the analytical persistence and methodology needed for professional penetration testing.

- TryHackMe | APIWizards Breach – Investigate an API-centric security breach, analyzing endpoint logs and web request payloads to reconstruct the attack.
- TryHackMe | TryHack3M: Sch3Ma D3Mon – Audit complex database schemas and web endpoints to discover hidden injection vulnerabilities and escalate access.
- TryHackMe | Crylo – Crack custom cryptographic mechanisms, inspect web parameters, and escalate privileges on a Linux target.
- TryHackMe | Industrial Intrusion – Investigate a cyber incident targeting Operational Technology (OT) and Industrial Control Systems (ICS).
- TryHackMe | Volt Typhoon – Trace living-off-the-land (LotL) tactics and state-sponsored adversary techniques inspired by real-world threat actors.
- TryHackMe | Logless Hunt – Practice threat hunting and digital forensic reconstruction on a compromised system where logging was disabled.
- TryHackMe | Security Footage – Analyze video feeds, image metadata, and network traffic to solve a multi-disciplinary security puzzle.
- TryHackMe | Mayhem – Navigate complex web logic flaws, pivot through local user accounts, and achieve root privileges.
- TryHackMe | Robots – Inspect web crawler directives (`robots.txt`), hidden endpoints, and server misconfigurations to breach the host.
- TryHackMe | Hackfinity Battle – Test your offensive skills in a competitive CTF environment featuring web, crypto, and system exploitation.
- TryHackMe | Rabbit Store – Audit an e-commerce platform for business logic flaws, parameter tampering, and server-side vulnerabilities.
- TryHackMe | Smol – Perform web enumeration, exploit subtle application bugs, and escalate privileges on a lightweight target.
- TryHackMe | Backtrack – Trace attacker activity in reverse, utilizing log files and system artifacts to uncover initial access vectors.

- TryHackMe | Extracted – Extract hidden payloads from custom file formats and reverse engineer validation logic.
- TryHackMe | The London Bridge – Exploit web application vulnerabilities, pivot across local services, and escalate privileges.
- TryHackMe | Breakme – Identify logical bypasses in restricted shells and elevate execution access on a Linux target.
- TryHackMe | Block – Analyze blockchain structures, smart contract logic, or network block configurations to retrieve flags.
- TryHackMe | New York Flankees – Exploit padding oracle vulnerabilities in web applications to decrypt sensitive cookie sessions.
- TryHackMe | Airplane – Exploit LFI bugs, inspect running processes via `/proc`, and hijack internal service binaries.
- TryHackMe | Profiles – Audit web user profile handlers for IDOR and deserialization bugs to gain unauthorized system shells.
- TryHackMe | Clocky – Reverse engineer time-based token generation algorithms to reset administrator passwords.
- TryHackMe | Hack Smarter Security – Analyze misconfigured security tools, pivot across internal subnets, and compromise target servers.
- TryHackMe | Kitty – Exploit blind SQL injection flaws in a web application to extract passwords and gain SSH access.
- TryHackMe | Umbrella – Practice network enumeration, Docker breakout vectors, and system privilege escalation.
- TryHackMe | AVenger – Practice Antivirus (AV) evasion techniques to execute custom payloads on a monitored host.
- TryHackMe | WhyHackMe – Solve a multi-stage CTF featuring web exploitation, hash cracking, and local system pivoting.
- TryHackMe | Stealth – Evade basic endpoint logging and detection rules while enumerating and compromising a target host.
- TryHackMe | Hunt Me I: Payment Collectors – Conduct threat hunting on payment gateway infrastructure to identify malicious credit card scrapers.
- TryHackMe | Hunt Me II: Typo Squatters – Investigate malicious domain squatting campaigns and malicious infrastructure targeting corporate brands.
- TryHackMe | Athena – Exploit command injection flaws, analyze network shares, and abuse internal Linux capabilities.
- TryHackMe | Forgotten Implant – Locate and analyze an active C2 malware implant left behind on a enterprise network host.
- TryHackMe | Race Conditions – Exploit asynchronous file handling and memory state race conditions (TOCTOU) to escalate privileges.

- TryHackMe | Weasel – Exploit Jupyter Notebook services, pivot through Windows Subsystem for Linux (WSL), and compromise the host.
- TryHackMe | Prioritise – Analyze web application sorting logic to execute blind SQL injection and dump database contents.
- TryHackMe | Boogeyman 1 – Investigate a full phishing incident response lifecycle, inspecting malicious macros, ISO files, and C2 traffic.
- TryHackMe | Mr Robot CTF – A popular CTF inspired by the TV show: perform web fuzzing, crack WordPress hashes, and escalate privileges.
- TryHackMe | Unattended – Exploit LFI vulnerabilities through web log poisoning to achieve remote code execution.
- TryHackMe | GoldenEye – Perform web enumeration, intercept POP3/Hydra credentials, and exploit vulnerable pop3/spip services.
- TryHackMe | StuxCTF – Exploit serialisation flaws, decode complex parameter chains, and achieve root on a Stuxnet-themed machine.
- TryHackMe | Boiler CTF – Enumerate hidden web subdirectories, exploit misconfigured internal services, and crack local hashes.
- TryHackMe | HA Joker CTF – Audit Joomla CMS installations, analyze secret archives, and execute local privilege escalation.
- TryHackMe | Biohazard – Solve a Resident Evil-themed room combining web form bypasses, steganography, and multi-user privilege escalation.
- TryHackMe | Break it – Audit web logic controls, bypass authentication headers, and gain access to internal server files.
- TryHackMe | Willow – Solve RSA decryption challenges, inspect hidden file systems, and elevate privileges on Linux.
- TryHackMe | The Marketplace – Exploit Blind XSS bugs, forge admin cookies, execute system commands, and abuse SUID Docker permissions.
- TryHackMe | Nax – Exploit Nagios XI vulnerabilities (CVE-2019-15846) to obtain remote code execution and root access.
- TryHackMe | Mindgames – Decrypt esoteric programming code (Brainfuck), obtain a Python shell, and abuse Linux capabilities.
- TryHackMe | Anonymous – Exploit misconfigured anonymous FTP shares, inject malicious shell scripts into cron jobs, and gain root.
- TryHackMe | Blog – Exploit CVE-2019-8942 in WordPress (Crop-image RCE), inspect SMB shares, and escalate privileges.
- TryHackMe | Wonderland – Solve an Alice in Wonderland-themed room involving directory fuzzing, Python library hijacking, and capabilities.
- TryHackMe | 0day – Exploit web application vulnerabilities, locate Shellshock (CVE-2014-6271), and escalate to root via kernel exploits.

- TryHackMe | CTF collection Vol.2 – Tackle a series of intermediate puzzles covering reverse engineering, cryptography, and forensic analysis.
- TryHackMe | CMesS – Discover subdomains, exploit Gila CMS vulnerabilities, extract cron job credentials, and gain root.
- TryHackMe | Deja Vu – Exploit web file upload vulnerabilities, analyze EXIF metadata processing bugs, and elevate local permissions.
- TryHackMe | hackerNote – Exploit web parameter manipulation, bypass authentication logic, and achieve internal command execution.
- TryHackMe | dogcat – Exploit LFI via PHP wrappers, poison Apache access logs, escape Docker containers, and capture all flags.
- TryHackMe | ConvertMyVideo – Exploit unvalidated command injection in a video converter site, sniff local traffic, and abuse cron jobs.
- TryHackMe | KoTH Hackers – Practice offensive speed-hacking strategies and system persistence mechanisms on a shared target.
- TryHackMe | Revenge – Exploit SQL injection flaws, crack system passwords, and abuse misconfigured system services.
- TryHackMe | harder – Bypass IP restrictions, exploit Git repository leaks, manipulate HMAC headers, and execute custom binary exploits.
- TryHackMe | HaskHell – Exploit a Haskell-based web submission platform to execute remote commands and recover root keys.
- TryHackMe | Undiscovered – Perform deep web directory fuzzing, exploit CMS flaws, and elevate privileges across local user accounts.
- TryHackMe | Break Out The Cage – Decrypt custom Vigenère ciphers, analyze entropy logs, and exploit Python script execution.
- TryHackMe | The Impossible Challenge – Solve complex reverse engineering puzzles and bypass anti-debugging protections to capture the flag.
- TryHackMe | Looking Glass – Port scan high-range ports, solve network routing riddles, and exploit local privilege escalation scripts.
- TryHackMe | Recovery – Perform post-incident recovery on a compromised host, reversing ransomware encryption and fixing broken system files.
- TryHackMe | Relevant – A realistic Windows pentest scenario involving SMB share enumeration, ASPX web shell execution, and token impersonation.
- TryHackMe | Ghizer – Exploit vulnerabilities in Ghidra network servers and local web administration utilities.
- TryHackMe | Mnemonic – Reverse engineer memory structures, crack mnemonic seeds, and bypass input validation.
- TryHackMe | WWBuddy – Exploit web application API endpoints, extract user database records, and elevate system access.

- TryHackMe | The Blob Blog – Audit blog CMS logic, reverse engineer binary validation files, and escalate system permissions.
- TryHackMe | Cooctus Stories – Solve multi-stage web exploitation puzzles, manipulate cookies, and abuse sudo script execution.
- TryHackMe | One Piece – Enumerate custom API ports, exploit command injection bugs, and elevate privileges on an anime-themed machine.
- TryHackMe | toc2 – Exploit Time-of-Check to Time-of-Use (TOCTOU) race condition bugs to overwrite sensitive system files.
- TryHackMe | NerdHerd – Perform OSINT, enumerate Samba shares, extract hidden steganography keys, and exploit sudo configurations.
- TryHackMe | Kubernetes Chall TDI 2020 – Audit misconfigured Kubernetes pods, perform cluster enumeration, and achieve node breakout.
- TryHackMe | The Server From Hell – Port scan thousands of open network ports, banner grab services, and solve host-based puzzles.
- TryHackMe | Jacob the Boss – Exploit an outdated JBoss application server via deserialization (DotCMS/JBoss RCE) to gain root access.
- TryHackMe | Unbaked Pie – Exploit Python pickle deserialization vulnerabilities in web cookie handlers to achieve remote execution.
- TryHackMe | Bookstore – Exploit REST API parameter flaws, analyze Werkzeug pin generators, and perform local privilege escalation.
- TryHackMe | Overpass 3 - Hosting – Exploit web FTP uploads, pivot through internal network interfaces, and hijack GPG key execution.
- TryHackMe | battery – Exploit XML External Entity (XXE) vulnerabilities, bypass login screens, and abuse SUID binary execution.
- TryHackMe | Madeye's Castle – Solve a Harry Potter-themed CTF featuring SQL injection, steganography, and custom privilege escalation.
- TryHackMe | En-pass – Exploit web password managers, bypass authentication forms, and escalate access on a target machine.
- TryHackMe | Sustah – Brute-force web forms using custom rate-limit bypass headers and exploit local privilege escalation.
- TryHackMe | KaffeeSec - SoMeSINT – Perform deep Social Media Intelligence (SoMeSINT) investigations to uncover hidden identities and assets.
- TryHackMe | Tokyo Ghoul – Exploit web parameter vulnerabilities, extract steganography keys, and hijack Python libraries.
- TryHackMe | Watcher – Practice LFI exploitation, log poisoning, FTP share manipulation, and multi-user privilege escalation.
- TryHackMe | broker – Exploit default credentials on ActiveMQ services and achieve RCE via CVE-2023-46604.

- TryHackMe | Inferno – Audit web login forms, brute-force HTTP basic auth, and abuse local SUID binaries.
- TryHackMe | VulnNet: dotpy – Exploit SSTI vulnerabilities in Python Flask apps, break out of restricted shells, and abuse cron jobs.
- TryHackMe | Wekor – Exploit SQL injection in WordPress plugins, pivot to internal memcached services, and gain root access.
- TryHackMe | pyLon – Audit Python web frameworks, analyze exposed SQLite databases, and achieve local command execution.
- TryHackMe | The Great Escape – Audit git repositories, escape restricted Docker containers, and compromise host system root.
- TryHackMe | SafeZone – Exploit web buffer overflows, inspect local services, and escalate access on a Linux target.
- TryHackMe | NahamStore – Audit a large e-commerce platform for web security bugs, including IDOR, XSS, CSRF, and command injection.
- TryHackMe | Sweettooth Inc. – Exploit Docker engine APIs, pivot through internal networks, and escalate privileges on Windows/Linux targets.
- TryHackMe | CMSpit – Exploit vulnerabilities in Cockpit CMS to reset admin credentials and achieve remote code execution.
- TryHackMe | Super-Spam – Analyze email logs, exploit web injection flaws, and elevate access on a spam-themed machine.
- TryHackMe | That's The Ticket – Exploit ticketing platform logic bugs, forge authentication tokens, and elevate system access.
- TryHackMe | Debug – Exploit PHP deserialization vulnerabilities in debugging modules to achieve remote shell access.
- TryHackMe | Red Stone One Carat – Perform active network sweeps, audit web endpoints, and escalate privileges on a Linux host.
- TryHackMe | Cold VVars – Exploit SMB share misconfigurations, manipulate web sockets, and hijack environment variables.
- TryHackMe | Metamorphosis – Exploit internal network services, leverage local file inclusions, and execute root privilege escalation.
- TryHackMe | SQHell – Solve a multi-lab room covering diverse SQL injection scenarios (in-band, blind, time-based, header-based).
- TryHackMe | Fortress – Perform network pivoting, breach hardened perimeters, and escalate privileges across multiple subnets.
- TryHackMe | CyberCrafted – Audit Minecraft server plugins, exploit web admin portals, and execute system commands.
- TryHackMe | Road – Exploit web profile picture upload forms, manipulate server background tasks, and abuse Shadow group permissions.

Hard CTF

The **Hard CTF** category represents the pinnacle of offensive security challenges on TryHackMe. Designed for experienced practitioners, these rooms demand deep technical mastery across multiple domains: advanced binary reverse engineering, custom exploit development, Active Directory forest compromises, perimeter defense evasion, kernel-level memory corruption, and multi-network pivoting. Expect minimal guidance, realistic enterprise defense configurations, and complex multi-vector exploitation paths.

- TryHackMe | Elevating Movement – Exploit complex Windows active directory misconfigurations and internal service bugs to achieve domain admin.
- TryHackMe | Initial Access Pot – Analyze honeypot environments and exploit zero-day style initial access vectors on enterprise systems.
- TryHackMe | Contrabando – Breach a heavily fortified enterprise target, bypass detection controls, and pivot across internal network segments.
- TryHackMe | Event Horizon – Reverse engineer complex compiled binaries, bypass memory protections, and escalate privileges.
- TryHackMe | Directory – Conduct advanced digital forensics and incident response on a compromised Active Directory infrastructure.
- TryHackMe | Honeynet Collapse CTF – Analyze compromised honeynet infrastructure, trace threat actor movements, and reconstruct attack paths.
- TryHackMe | Moebius – Solve complex cryptographic puzzles, exploit custom web applications, and break out of restricted execution environments.
- TryHackMe | Rabbit Hole – Navigate a deep maze of rabbit holes, anti-analysis traps, and obfuscated binaries to compromise the host.
- TryHackMe | Mountaineer – Exploit complex Linux kernel misconfigurations, custom SUID binaries, and internal network services.
- TryHackMe | CERTain Doom – Audit digital certificate infrastructures, exploit PKI misconfigurations, and achieve domain escalation.
- TryHackMe | Capture Returns – Bypass network authentication mechanisms and execute multi-stage exploits against hardened infrastructure.
- TryHackMe | Chrome – Inspect browser memory dumps, extract encrypted credentials, and exploit browser extension storage.
- TryHackMe | Reset – Exploit complex Active Directory password reset mechanics and ticket granting flaws.
- TryHackMe | Motunui – Reverse engineer custom software binaries, exploit web API logic, and achieve root access.
- TryHackMe | Spring – Exploit critical vulnerabilities in Java Spring Framework applications (Spring4Shell / CVE-2022-22965).

- TryHackMe | Brainpan 1 – A classic OSCP-style machine requiring 32-bit Windows buffer overflow exploitation and Linux privilege escalation.
- TryHackMe | Borderlands – Perform multi-network pivoting, exploit API microservices, and compromise hardened enterprise perimeters.
- TryHackMe | hc0n Christmas CTF – Solve an advanced multi-stage CTF covering reverse engineering, cryptography, and network exploitation.
- TryHackMe | Daily Bugle – Exploit Joomla SQL injection (CVE-2023-23752 / SQLi), crack password hashes, and exploit yum SUID privileges.
- TryHackMe | Retro – Exploit CVE-2019-1388 via Windows GUI elevation of privilege on a retro-themed target.
- TryHackMe | Jeff – Exploit WordPress vulnerabilities, intercept web traffic, break out of backup archives, and abuse sudo permissions.
- TryHackMe | Racetrack Bank – Exploit complex financial logic flaws and race conditions in a banking web application.
- TryHackMe | Dave's Blog – Audit Node.js source code, bypass security filters, and execute local binary exploits.
- TryHackMe | CherryBlossom – Exploit custom router firmware, analyze network traffic, and compromise IoT infrastructure.
- TryHackMe | CCT2019 – Tackle an advanced cybersecurity competition environment covering network forensics and binary exploitation.
- TryHackMe | Iron Corp – Perform Active Directory enumeration, exploit custom internal web utilities, and achieve Domain Admin.
- TryHackMe | Carpe Diem 1 – Breach an enterprise network perimeter, pivot through internal subnets, and execute domain compromise.
- TryHackMe | Ra – Exploit misconfigured Windchill services, dump Active Directory hashes, and compromise Domain Controllers.
- TryHackMe | Year of the Fox – Perform web directory brute-forcing, exploit SMB null sessions, bypass restricted shells, and gain root.
- TryHackMe | For Business Reasons – Exploit Windows enterprise misconfigurations, manipulate Kerberos tokens, and escalate access.
- TryHackMe | Anonymous Playground – Decrypt custom ciphers, reverse engineer binary files, and exploit custom SUID binaries.
- TryHackMe | Misguided Ghosts – Solve multi-stage forensic and reverse engineering challenges to discover exploitation vectors.
- TryHackMe | Theseus – Exploit web application logic flaws, break out of restricted containers, and execute root privilege escalation.
- TryHackMe | Internal – Perform a full internal penetration test: exploit WordPress, pivot via SSH, crack Jenkins credentials, and compromise Active Directory.

- TryHackMe | Year of the Dog – Exploit web authentication bypass bugs, poison internal API requests, and escalate system permissions.
- TryHackMe | You're in a cave – Solve a maze of low-level system puzzles, binary disassemblies, and memory exploitation steps.
- TryHackMe | Year of the Owl – Exploit Windows Active Directory environments, manipulate GPOs, and execute ticket attacks.
- TryHackMe | Year of the Pig – Bypass complex web application firewalls (WAF), perform parameter fuzzing, and achieve root shell access.
- TryHackMe | envizon – Exploit vulnerabilities in network mapping dashboards and execute system command injections.
- TryHackMe | GameBuzz – Exploit web application logic bugs, deobfuscate source code, and hijack system execution paths.
- TryHackMe | Fusion Corp – Perform Kerberoasting, AS-REP roasting, and pass-the-hash attacks against an Active Directory Domain Controller.
- TryHackMe | Crocc Crew – Exploit custom web APIs, manipulate database tokens, and achieve root access across internal servers.
- TryHackMe | Uranium CTF – Intercept malicious email attachments, extract SSH keys, crack passwords, and exploit custom SUID binaries.
- TryHackMe | Year of the Jellyfish – Exploit web applications, bypass security controls, and escalate privileges on a hardened host.
- TryHackMe | Rocket – Audit Chat platforms (Rocket.Chat RCE), exploit MongoDB injection, and execute privilege escalation.
- TryHackMe | Squid Game – Analyze malicious document files (.doc/.xls macros), extract shellcode, and reverse engineer C2 payloads.
- TryHackMe | EnterPrize – Exploit enterprise Java applications, pivot across internal subnets, and compromise Active Directory.
- TryHackMe | Different CTF – Solve non-standard exploitation challenges requiring custom script development and low-level analysis.
- TryHackMe | VulnNet: dotjar – Exploit Java deserialization vulnerabilities in JAR applications, break out of restricted execution, and gain root.
- TryHackMe | M4tr1x: Exit Denied – Solve a Matrix-themed room involving port knocking, custom cryptography, and binary exploitation.
- TryHackMe | Shaker – Audit web application backend code, exploit deserialization flaws, and elevate local system privileges.

Misc

The **Miscellaneous** collection brings together specialized topics, emerging hardware/IoT security concepts, frameworks, and real-world Common Vulnerabilities and Exposures (CVEs).

This section covers iconic software vulnerabilities (such as Log4j, PrintNightmare, Spring4Shell, and Dirty Pipe), framework methodologies like MITRE ATT&CK, hardware and ICS/SCADA security, and niche evasion concepts that don't fit into single traditional categories.

- TryHackMe | GeoServer: CVE-2025-58360 – Analyze and exploit a critical remote code execution vulnerability in GeoServer geospatial software.
- TryHackMe | Next.js: CVE-2025-29927 – Explore routing and server-side rendering vulnerabilities affecting modern Next.js web applications.
- TryHackMe | Spring4Shell: CVE-2022-22965 – Understand and exploit the critical RCE flaw in the Java Spring Framework via Data Binder parameter binding.
- TryHackMe | PrintNightmare – Exploit the Windows Print Spooler vulnerability (CVE-2021-34527) to gain remote code execution and local system privileges.
- TryHackMe | GitLab CVE-2023-7028 – Learn how unauthenticated password resets allowed attackers to hijack GitLab accounts via email parameter manipulation.
- TryHackMe | CVE-2023-38408 – Analyze remote code execution vulnerabilities in OpenSSH PKCS#11 provider library loading.
- TryHackMe | Joomify: CVE-2023-23752 – Exploit an unauthenticated information disclosure vulnerability in Joomla web applications to retrieve database credentials.
- TryHackMe | OverlayFS - CVE-2021-3493 – Learn how user namespaces in Ubuntu's OverlayFS implementation allowed local privilege escalation to root.
- TryHackMe | Intrusion Detection – Discover techniques used by attackers to bypass Snort and Suricata Intrusion Detection System (IDS) rules.
- TryHackMe | Shodan.io – Master using the Shodan search engine to discover exposed infrastructure, open ports, and vulnerable IoT devices.
- TryHackMe | Insekube – Practice auditing misconfigured Kubernetes clusters and breaking out of insecure pod configurations.
- TryHackMe | Solar, exploiting log4j – Exploit Log4Shell (CVE-2021-44228) in Apache Log4j using JNDI lookup payloads to achieve instant remote code execution.
- TryHackMe | CVE-2022-26923 – Explore Active Directory Domain Services privilege escalation via computer account dNSHostName manipulation.
- TryHackme | ICS/Modbus - Claus for Concern – Analyze Industrial Control Systems (ICS) protocols like Modbus to detect and manipulate operational technology traffic.
- TryHackme | Passwords - A Cracking Christmas – Practice extracting and cracking passwords from encrypted file archives (ZIP, RAR, 7z) using John the Ripper.
- TryHackMe | Django: CVE-2025-64459 – Examine and exploit security flaws in Django backend handling logic.
- TryHackMe | Roundcube: CVE-2025-49113 – Analyze remote code execution vulnerabilities targeting Roundcube webmail applications.

- TryHackMe | Erlang/OTP SSH: CVE_2025_32433 – Audit SSH daemon implementations written in Erlang/OTP for authentication bypass bugs.
- TryHackMe | Training Impact on Teams – Understand how continuous practical cybersecurity training improves SOC response times and team effectiveness.
- TryHackMe | PaperCut: CVE-2023-27350 – Exploit an unauthenticated remote code execution vulnerability in PaperCut print management software.
- TryHackMe | Moniker Link (CVE-2024-21413) – Analyze how Outlook Moniker Link handling flaws bypass Protected View and leak NTLM credentials.
- TryHackMe | Confluence CVE-2023-22515 – Exploit a critical broken authentication flaw in Atlassian Confluence Data Center to create unauthorized admin accounts.
- TryHackMe | Cactus – Analyze ransomware infection vectors, persistence mechanisms, and incident response mitigation steps.
- TryHackMe | Looney Tunables – Exploit CVE-2023-4911 in GNU C Library's (`glibc`) `ld.so` dynamic loader to gain root privileges on Linux systems.
- TryHackMe | Threat Intel & Containment – Learn to transform raw security indicators into actionable threat intelligence to contain active network breaches.
- TryHackMe | Introduction to Django – Explore how Django web applications function, including routing, ORM models, and template rendering.
- TryHackMe | Git Happens – Recover source code and secret credentials from exposed `.git` version control repositories.
- TryHackMe | Meltdown Explained – Understand hardware speculative execution side-channel vulnerabilities (Meltdown/Spectre) that leak kernel memory.
- TryHackMe | Splunk – Learn the basics of Splunk SIEM architecture, log indexing, and searching security telemetry.
- TryHackMe | Linux Backdoors – Discover common techniques attackers use to maintain persistent root access on compromised Linux systems (cron, SSH keys, web shells).
- TryHackMe | Jupyter 101 – Learn how data science environments like Jupyter Notebooks are used in security analytics and threat research.
- TryHackMe | Geolocating Images – Practice image analysis and OSINT geolocation techniques to identify real-world physical locations from photographs.
- TryHackMe | Tor – Explore the mechanics of the Tor network, onion routing, anonymous web browsing, and dark web operational security.
- TryHackMe | tomghost – Exploit the Ghostcat vulnerability (CVE-2020-1938) in Apache Tomcat AJP connectors to read secret configuration files.
- TryHackMe | DLL HIJACKING – Understand how Windows applications load Dynamic Link Libraries (DLLs) and exploit unsafe search order hijacking for privilege escalation.
- TryHackMe | Intro to IoT Pentesting – Learn how to audit Internet of Things (IoT) hardware firmware, inspect network protocols, and identify embedded vulnerabilities.

- TryHackMe | Attacking ICS Plant #1 – Explore Programmable Logic Controller (PLC) operations and manipulate physical plant controls in a simulated SCADA environment.
- TryHackMe | Attacking ICS Plant #2 – Practice advanced industrial control network sniffing, logic coil manipulation, and OT attack vectors.
- TryHackMe | Printer Hacking 101 – Audit network printing protocols (PJL, PostScript) to extract cached documents, print jobs, and credentials.
- TryHackMe | DNS Manipulation – Learn techniques like DNS tunneling, exfiltration, and cache poisoning used to bypass network firewalls.
- TryHackMe | Introduction to Flask – Understand web routes, templates, and backend processing using the Python Flask framework.
- TryHackMe | MITRE – Master the MITRE ATT&CK framework to categorize adversary Tactics, Techniques, and Procedures (TTPs).
- TryHackMe | magician – Exploit ImageMagick vulnerabilities (ImageTragick / LFI), manipulate port proxies, and gain root access.
- TryHackMe | JPGChat – Exploit unvalidated command execution in Python custom chat scripts to obtain initial system access.
- TryHackMe | Baron Samedit – Exploit the Baron Samedit heap-based buffer overflow in sudo (CVE-2021-3156) for root access.
- TryHackMe | CVE-2021-41773/42013 – Exploit path traversal and remote code execution vulnerabilities in Apache HTTP Server 2.4.49 and 2.4.50.
- TryHackMe | Binary Heaven – Reverse engineer binary executables, analyze disassembly logic, and bypass license check routines.
- TryHackMe | Git and Crumpets – Audit self-hosted Gitea instances, inspect commit histories, and execute local privilege escalation.
- TryHackMe | Polkit: CVE-2021-3560 – Exploit local privilege escalation flaws in Polkit (PolicyKit) via timed parameter termination.
- TryHackMe | Hip Flask – Inspect local web execution environments, extract system tokens, and escalate privileges.
- TryHackMe | Bypass Disable Functions – Learn techniques to execute system commands in PHP web applications when `exec()` and `system()` functions are disabled.
- TryHackMe | Wordpress: CVE-2021-29447 – Exploit XML External Entity (XXE) vulnerabilities in WordPress media upload parsing routines.
- TryHackMe | Linux Function Hooking – Learn how shared library function hooking operates to intercept userland function calls.
- TryHackMe | REvil Corp – Perform threat analysis and digital forensics on an enterprise network infected by the REvil ransomware group.
- TryHackMe | Sudo Buffer Overflow – Analyze heap memory corruption vulnerabilities within older versions of the sudo binary.

- TryHackMe | Sudo Security Bypass – Exploit historic configuration parsing bugs in `sudo` to run commands as root without valid credentials.
- TryHackMe | Conti – Conduct incident response on a target infected with Conti ransomware, inspecting event logs, shadow copies, and network telemetry.
- TryHackMe | Dirty Pipe: CVE-2022-0847 – Exploit a vulnerability in the Linux Kernel pipe mechanism to overwrite read-only arbitrary file contents and elevate to root.
- TryHackMe | The find command – Master using the Linux `find` command for security auditing, permission auditing, and SUID file discovery.

Cloud

As organizations migrate workloads away from traditional on-premise datacenters, understanding cloud security architecture becomes essential. Cloud security covers identity management, misconfigured storage containers, overly permissive IAM roles, and shared responsibility defense models. This section introduces foundational cloud computing concepts across AWS, Azure, and Google Cloud Platform (GCP) alongside common architectural security pitfalls.

- TryHackMe | Cloud Computing Fundamentals – Learn core cloud concepts, deployment models (IaaS, PaaS, SaaS), virtualization structures, and shared responsibility frameworks.
- TryHackMe | Cloud Security Pitfalls – Explore common cloud misconfigurations, publicly exposed storage buckets, overly permissive IAM policies, and credential leaks.

Special Events

TryHackMe regularly hosts time-limited challenges, community competitions, and seasonal learning events designed to showcase diverse security disciplines. These events—such as the annual *Advent of Cyber* series—deliver bite-sized daily tasks that introduce beginner and intermediate learners to web hacking, forensics, malware analysis, cloud defense, and log investigation through gamified storylines.

- TryHackMe | 25 Days of Cyber Security – A 25-task beginner course covering fundamental topics across offensive, defensive, and general security fields.
- TryHackMe | Advent of Cyber 1 2019 – The inaugural Advent of Cyber event featuring 25 daily beginner-friendly hands-on security rooms.
- TryHackMe | Advent of Cyber 2 2020 – Practice fundamental web application hacking, network scanning, and log analysis in a holiday-themed event.
- TryHackMe | Advent of Cyber 3 (2021) – Tackle daily scenarios covering Blue Team triage, web exploitation, memory forensics, and script automation.
- TryHackMe | Advent of Cyber 2022 – Explore Christmas-themed challenges focused on OSINT, malware analysis, web security, and cloud basics.
- TryHackMe | Advent of Cyber 2023 – Solve beginner-friendly defensive and offensive tasks, covering SQLi, log parsing, secure coding, and threat intelligence.

- TryHackMe | Advent of Cyber 2024 – Work through interactive daily challenges introducing modern security topics, including AI security, cloud triage, and log investigation.
- TryHackMe | Advent of Cyber '23 Side Quest – An advanced, hard-difficulty companion event featuring multi-stage CTF challenges and binary exploitation.
- TryHackMe | Cyber Scotland 2021 – Complete a collection of regional event challenges focusing on core cybersecurity skills and awareness.
- TryHackMe | Hacker of the Hill #1 – Test your machine takeover speed and persistence mechanisms in a competitive King of the Hill (KoTH) format.
- TryHackMe | Learn and win prizes – Complete fundamental security learning modules to earn event tickets and complete platform challenges.
- TryHackMe | Learn and win prizes #2 – Expand your hands-on skills through seasonal platform event rooms and interactive exercises.

CONCLUSION

Cybersecurity is a marathon, not a sprint. While the initial learning curve can feel steep and intimidating, resources like TryHackMe prove that financial barriers should never stand in the way of a high-quality technical education. Dedicating yourself to hands-on learning, active problem-solving, and continuous exploration will yield far greater long-term success than simply reading theoretical books or watching passive video tutorials. By systematically working through the curated collection of free rooms cataloged above, you are taking a decisive step toward mastering the fundamentals of both offensive and defensive security. Each room you complete builds real technical muscle memory, sharpens your analytical thinking, and brings you closer to operational competence. Navigating through diverse categories—from basic networking and scripting to complex web vulnerabilities and incident response—will help you connect theoretical concepts to actual system behavior. To get the most value out of this resource, structure your approach thoughtfully rather than rushing to collect flags. Bookmark this guide, integrate it into your weekly study routine, and aim for consistent progress—even completing just one room a day will lead to massive skill growth over time. Document your findings, take detailed notes, and try to understand *why* a specific vulnerability exists or *how* a defensive rule catches malicious traffic before moving on to the next task.

Ultimately, tools, platforms, and security methodologies will continuously evolve, but the core engine driving any successful cybersecurity professional remains the same: relentless curiosity and stubborn persistence. Embrace the inevitable challenges when an exploit fails or a forensic investigation hits a wall, because troubleshooting those frustrating moments is where true technical mastery happens. Trust the learning process, stay consistent, and enjoy every step of your cybersecurity journey. Happy Hacking!

SUPPORT & SPREAD THE WORD!

Putting together, testing, and categorizing this massive list of TryHackMe free rooms took a lot of time, research, and effort. My main goal was to create a truly valuable, zero-cost guide to help aspiring cybersecurity professionals kickstart their careers without getting lost in paywalls or tutorial hell.

If you found this guide helpful, **you can support this project in a few quick ways:**

- 💬 **Leave a Comment:** Drop a comment below! Let me know which room you're currently working on, suggest rooms I should add, or just say hi.
- 📢 **Share on Social Media:** If you know anyone trying to break into cybersecurity, share this article on **LinkedIn, X (Twitter), Reddit, or Discord**.
- 🏷️ **Tag Me:** When sharing your progress or posting about this guide on social platforms, feel free to tag me—I'd love to see your learning journey and re-share your posts!
- 🔖 **Bookmark & Pass It Along:** Keep this page bookmarked for your daily practice, and send it to your study groups, tech communities, or classmates.

Your support, shares, and feedback mean the world to me and keep projects like this alive. Thank you for reading, and happy hacking! 🚀