Building a modern network security lab requires moving beyond simple firewall rules and into the territory of **Zero Trust**, **Cloud integration**, and **Automated Defense**.

Here are 10 modern network security labs you can implement to bridge the gap between "entry-level" and "professional" cybersecurity skills.

---

1. Zero Trust Remote Access (ZTNA)

Instead of a traditional VPN, implement a Zero Trust access model using tools like **Tailscale** or **Twingate**.

- **The Task:** Set up a private resource (like a file server) that is invisible to the public internet. Configure identity-based access so only authenticated users on "healthy" devices can connect.
- **Skill Gained:** Understanding the "Trust No One" principle and move away from perimeter-based security.

2. Network Segmentation with VLANs and Docker

Don't just put everything on one subnet. Use a hypervisor like **Proxmox** or **ESXi** to segment your traffic.

- **The Task:** Create three distinct VLANs: Management, IoT, and Lab. Use a virtualized firewall (like **pfsense**) to write rules that prevent your IoT devices from "talking" to your lab servers.
- **Skill Gained:** Preventing lateral movement—the primary way attackers spread after an initial breach.

3. SIEM & Centralized Logging (Wazuh/ELK)

A network you can't see is a network you can't defend.

- **The Task:** Deploy a **Wazuh** or **Elastic Stack (ELK)** manager. Install agents on your Windows and Linux VMs to stream logs. Create a custom dashboard that alerts you when a "Brute Force" attempt is detected on your SSH port.
- **Skill Gained:** Security Monitoring and Log Analysis—essential for SOC (Security Operations Center) roles.

4. Web Application Firewall (WAF) Integration

Protect a vulnerable application (like **OWASP Juice Shop**) from common attacks.

- **The Task:** Deploy Juice Shop behind a reverse proxy like **Nginx** or **Traefik**. Configure a WAF (like **ModSecurity**) to block SQL Injection and Cross-Site Scripting (XSS) attacks before they reach the app.
- **Skill Gained:** Layer 7 (Application Layer) security and traffic filtering.

5. Network-Wide Ad and Malware Blocking

Use DNS-level filtering to secure every device in your lab.

- **The Task:** Set up **Pi-hole** or **AdGuard Home**. Configure it to use **DNS-over-HTTPS (DoH)** for privacy. Add community-maintained malware blacklists to sinkhole known malicious domains.

- **Skill Gained:** DNS security and reducing the "attack surface" by eliminating telemetry and ad-based tracking.

6. Automated Intrusion Prevention (IPS)

Move from passive detection to active prevention.

- **The Task:** Configure **Suricata** or **Snort** on your pfsense/opnsense firewall. Set up "Block" rules that automatically drop traffic from IPs that trigger specific high-severity signatures.
- **Skill Gained:** Real-time threat mitigation and signature management.

7. Cloud Security Posture Management (CSPM)

Modern networking happens in the cloud.

- **The Task:** In a free-tier AWS or Azure account, intentionally misconfigure an S3 bucket to be "Public" and create a Security Group with port 22 open to 0.0.0.0/0. Use a tool like **Prowler** or **Checkov** to scan and "find" these vulnerabilities.
- **Skill Gained:** Identifying cloud misconfigurations, which are the #1 cause of modern data breaches.

8. Honey Pot Deployment

Learn how attackers behave by giving them a fake target.

- **The Task:** Deploy a low-interaction honeypot like **Cowrie** (which mimics a vulnerable SSH/Telnet server). Expose it to the internet (securely, in a DMZ) and watch the "attacks" roll in via your SIEM.
- **Skill Gained:** Threat Intelligence and understanding attacker TTPs (Tactics, Techniques, and Procedures).

9. Infrastructure as Code (IaC) Security

Security should be "baked in" from the start.

- **The Task:** Write a **Terraform** script to deploy a small network. Use a linter like **tfsec** to scan your code for security flaws (like unencrypted disks) before the infrastructure is even built.
- **Skill Gained:** DevSecOps—the practice of integrating security into the software development lifecycle.

10. Incident Response Simulation

Test your ability to clean up after a "breach."

- **The Task:** Use a script to simulate a ransomware infection (e.g., encrypting a specific folder) on a Linux VM. Use your logs to find the "Patient Zero" and follow a formal **Incident Response** plan to contain, eradicate, and recover.
- **Skill Gained:** "Blue Team" resilience and forensics.