🛡️ CyberRange - Ethical Hacking Training Platform

A modern, Apple-inspired web-based cyber range for ethical hacking and DFIR (Digital Forensics & Incident Response) training. Packaged as a single Docker container for easy deployment.

✨ Features

🎯 Training Scenarios

1. **Web Application Vulnerabilities** (Beginner)
   - SQL Injection, XSS, CSRF exploitation
   - Tools: Burp Suite, SQLMap, OWASP ZAP
2. **Network Reconnaissance** (Intermediate)
   - Port scanning, service enumeration, network mapping
   - Tools: Nmap, Wireshark, Netcat, Masscan
3. **Password Cracking & Hash Analysis** (Intermediate)
   - Hash cracking, brute force attacks
   - Tools: John the Ripper, Hashcat, Hydra
4. **Wireless Network Security** (Advanced)
   - WiFi hacking, WPA2 cracking, Evil Twin attacks
   - Tools: Aircrack-ng, Reaver, Wifite, Kismet
5. **Digital Forensics Investigation** (Advanced)
   - Disk image analysis, memory forensics, incident investigation
   - Tools: Autopsy, Volatility, FTK Imager, Sleuth Kit
6. **Malware Analysis Lab** (Expert)
   - Reverse engineering, static/dynamic analysis
   - Tools: IDA Pro, Ghidra, x64dbg, Process Monitor
7. **Log Analysis & SIEM** (Intermediate)
   - Security event detection, anomaly analysis
   - Tools: Splunk, ELK Stack, Grep, Regex
8. **Exploit Development** (Expert)
   - Buffer overflows, ROP chains, binary exploitation
   - Tools: GDB, pwntools, ROPgadget, Metasploit
9. **Incident Response Simulation** (Advanced)
   - Real-time incident response, NIST framework
   - Tools: Velociraptor, KAPE, Sysmon, TheHive

🎨 Design Features

- Clean, minimalist Apple-inspired interface
- Responsive design for desktop and mobile
- Interactive terminal emulator
- Difficulty-based color coding
- Smooth animations and transitions
- Modern card-based layout

🚀 Quick Start

Prerequisites

- Docker installed on your system
- No other software required!

Build and Run

1. **Save the Dockerfile** to a new directory
2. **Build the Docker image:**

```bash
docker build -t cyberrange:latest .
```

3. **Run the container:**

```bash
docker run -d -p 8080:80 --name cyberrange cyberrange:latest
```

4. **Access the platform:**

Open your browser and navigate to: `http://localhost:8080`

Alternative: One-liner Deployment

```bash
docker build -t cyberrange . && docker run -d -p 8080:80 --name cyberrange cyberrange:latest
```

🔧 Advanced Configuration

Custom Port

Run on a different port (e.g., 3000):

```bash
docker run -d -p 3000:80 --name cyberrange cyberrange:latest
```

Environment Variables

```bash
docker run -d \
  -p 8080:80 \
  -e NGINX_HOST=localhost \
  -e NGINX_PORT=80 \
  --name cyberrange \
  cyberrange:latest
```

Persistent Data (Optional)

If you want to add persistent storage for user progress:

```bash
docker run -d \
  -p 8080:80 \
  -v $(pwd)/data:/data \
  --name cyberrange \
  cyberrange:latest
```

🛠️ Docker Commands

Start the container

```bash
docker start cyberrange
```

Stop the container

```bash
docker stop cyberrange
```

View logs

bash

```
docker logs cyberrange
```

Remove the container

bash

```
docker rm -f cyberrange
```

Rebuild after updates

bash

```
docker rm -f cyberrange
docker rmi cyberrange:latest
docker build -t cyberrange:latest .
docker run -d -p 8080:80 --name cyberrange cyberrange:latest
```

📱 Usage Guide

1. **Browse Scenarios**: View all available training scenarios on the main page
2. **Select a Scenario**: Click on any card to open the scenario details
3. **Explore Tools**: Review available tools and challenges
4. **Use Terminal**: Interact with the terminal emulator to practice commands
5. **Start Training**: Begin your ethical hacking journey

Terminal Commands

The interactive terminal supports these commands:

- `help` - Display available commands
- `nmap` - Simulate network scanning
- `sqlmap` - Simulate SQL injection testing
- `hydra` - Simulate password attacks
- `clear` - Clear terminal output

🎯 Target Audience

- **Ethical Hackers**: Practice offensive security techniques
- **Security Professionals**: Enhance penetration testing skills

- **DFIR Analysts**: Train on forensics and incident response
- **Students**: Learn cybersecurity fundamentals
- **Bug Bounty Hunters**: Sharpen vulnerability discovery skills

🔒 Security & Ethics

This platform is designed for **educational purposes only**. Always follow these guidelines:

- ✅ Only test systems you own or have explicit permission to test
- ✅ Practice responsible disclosure for any vulnerabilities found
- ✅ Respect privacy and data protection laws
- ✅ Use knowledge for defensive purposes
- ❌ Never use these skills for malicious purposes
- ❌ Do not attack systems without authorization

📊 System Requirements

Development

- Docker 20.10+
- 2GB RAM minimum
- 1GB free disk space

Production

- Docker 20.10+
- 512MB RAM minimum
- 500MB free disk space
- Any modern browser (Chrome, Firefox, Safari, Edge)

🌐 Browser Compatibility

- ✅ Chrome 90+
- ✅ Firefox 88+
- ✅ Safari 14+
- ✅ Edge 90+
- ✅ Mobile browsers (iOS Safari, Chrome Mobile)

🔄 Updates & Maintenance

Update the application

```bash
# Pull latest changes
git pull origin main

# Rebuild container
docker rm -f cyberrange
docker rmi cyberrange:latest
docker build -t cyberrange:latest .
docker run -d -p 8080:80 --name cyberrange cyberrange:latest
```

🐛 Troubleshooting

Port already in use

```bash
# Check what's using port 8080
lsof -i :8080

# Use a different port
docker run -d -p 3000:80 --name cyberrange cyberrange:latest
```

Container won't start

```bash
# Check logs
docker logs cyberrange

# Remove and recreate
docker rm -f cyberrange
docker run -d -p 8080:80 --name cyberrange cyberrange:latest
```

Build fails

```bash
```

```
# Clean Docker cache
docker builder prune -a

# Rebuild from scratch
docker build --no-cache -t cyberrange:latest .
```

📚 Additional Resources

- OWASP Top 10
- NIST Cybersecurity Framework
- SANS Reading Room
- Hack The Box
- TryHackMe

🤝 Contributing

Contributions are welcome! To add new scenarios or features:

1. Fork the repository
2. Create a feature branch
3. Add your scenario to the `scenarios` array
4. Test thoroughly
5. Submit a pull request

📄 License

This project is for educational purposes. Always ensure compliance with local laws and regulations when using security testing tools.

⚠️ Disclaimer

The tools and techniques demonstrated in this platform are for educational purposes only. The creators and maintainers are not responsible for any misuse or damage caused by this software. Always obtain proper authorization before conducting security testing.

🎓 Learning Path

Beginner Track

1. Web Application Vulnerabilities
2. Password Cracking & Hash Analysis
3. Log Analysis & SIEM

Intermediate Track

1. Network Reconnaissance
2. Wireless Network Security
3. Digital Forensics Investigation

Advanced Track

1. Malware Analysis Lab
2. Exploit Development
3. Incident Response Simulation

📞 Support

For issues, questions, or feature requests, please open an issue on the project repository.

Happy Hacking! 🚀🛡️

Remember: With great power comes great responsibility. Always hack ethically.