BÜRKÜT-Yol Haritası

🦅 PROJE BÜRKÜT: TAM KAPSAMLI UYGULAMA REHBERİ

> **Vizyon**
>
> İnsan ve Yapay Zeka (MCP) yeteneklerini; hibrit, izole ve otonom korunan bir ortamda kıyaslayan; zafiyetleri sadece bulan değil, kod seviyesinde otomasyonla (IaC) kapatan yeni nesil bir Purple Team laboratuvarı.

🧱 SEVİYE 0: İNŞAAT ALANI (ALTYAPI & AĞ)

> **Amaç**
>
> Sanal veri merkezini kurmak. Henüz saldırı yok, sadece kablolama.

1. VMware Ağ Konfigürasyonu (Virtual Network Editor)

- **VMnet0 (Bridged/NAT):** Dış dünyaya (İnternet/API) çıkış kapısı. (DHCP Açık).
- **VMnet2 (Host-Only):** İzole Laboratuvar Ağı. **DHCP KAPALI.**
  - **Subnet:** `192.168.100.0`
  - **Mask:** `255.255.255.0`

2. Sanal Makinelerin Kurulumu ve Ağ Ayarları

A. GÖZETLEME KULESİ (SIEM - Wazuh)

- **OS:** Ubuntu Server 22.04 (4GB RAM, 2 vCPU).
- **NIC 1 (NAT):** İnternet, Güncellemeler ve Threat Intel Feed'leri için.
- **NIC 2 (Host-Only):** `192.168.100.10` (Statik). Ajanlardan log toplamak için.
- **Yazılım:** Docker üzerinde Wazuh Manager & Dashboard kurulumu.

B. SALDIRGAN (ATTACKER - Kali Linux)

- **OS:** Kali Linux 2025.x (4GB RAM, 2 vCPU).

- **NIC 1 (NAT):** OpenAI/Anthropic API Erişimi için Gateway.
- **NIC 2 (Host-Only):** `192.168.100.5` (Statik). Saldırı trafiği için.
- **Routing:** Default Gateway `NIC 1` üzerinde olmalı. `NIC 2` sadece yerel ağa bakmalı.

C. KURBAN (TARGET - Ubuntu)

- **OS:** Ubuntu Server 22.04 (Min. Kaynak).
- **NIC 1 (Host-Only):** `192.168.100.20` (Statik). Tek Bacak.
- **Not:** Kurulum ve paket yüklemeleri (Docker vb.) için geçici NAT eklenecek, iş bitince silinecek.

3. Temel Yazılımlar

- Kurban makineye `Docker`, `Docker Compose` ve `Vulhub` reposunun (`/opt/vulhub`) indirilmesi.

> ✓ 🎯 **BOSS FIGHT (SEVİYE 0 SINAVI)**
>
> - [ ] Kali, `google.com`'a erişebiliyor mu? (Evet)
> - [ ] Kali, `192.168.100.20` (Kurban) makinesine ping atabiliyor mu? (Evet)
> - [ ] Kurban, `google.com`'a erişebiliyor mu? (HAYIR - Kesinlikle izole olmalı).
> - [ ] Kurban, `192.168.100.10` (Wazuh) makinesine erişebiliyor mu? (Evet).

🛠️ SEVİYE 1: ZANAATKAR (MANUEL USTALIK)

> ⓘ **Amaç**
>
> Otomasyon olmadan, el yordamıyla sistemin ciğerini (Log, Zafiyet, Yama) öğrenmek.

1. Sahne Kurulumu

- Kurban makinede Vulhub üzerinden bir zafiyet (Örn: Log4j veya Tomcat) seçip `docker-compose up -d` ile başlatılması.
- Kurban makineye Wazuh Agent kurulması.
- `ossec.conf` ayarı: Docker loglarını okuyacak şekilde yapılandırılması.

2. Manuel Döngü (The Loop)

1. **Red (Saldır):** Kali'den manuel Nmap taraması ve Metasploit ile exploit denemesi.
2. **Blue (İzle):** Wazuh Dashboard'da saldırı loglarının (Alerts) teyit edilmesi.
3. **Purple (Yamala - Manuel):** Konteynerin içine girip (`docker exec`) veya config dosyasını düzenleyip açığı kapatmak.
4. **Verify (Doğrula):** Tekrar saldırıp başarısız olduğunu görmek.

> **🎯 BOSS FIGHT (SEVİYE 1 SINAVI)**
>
> - [ ] Wazuh Dashboard'da kendi saldırı loglarını net bir şekilde görebiliyor musun?
> - [ ] Manuel yaptığın yamadan sonra exploit gerçekten engellendi mi?

🤖 SEVİYE 2: SİBER ÇIRAK (AI, MCP & GÜVENLİK)

> **Amaç**
>
> Saldırı yetkisini Yapay Zekaya devrederken, onu bir deli gömleğiyle (Guardrails) sınırlamak.

1. AI Beyninin Entegrasyonu

- Kali üzerinde Python Sanal Ortamı (`venv`) ve MCP İstemcisinin (Goose/Open Interpreter) kurulumu.
- API Key tanımlaması ve bağlantı testi.

2. 🛡️ Fail-Safe Mekanizmaları (Güvenlik Kilitleri)

- **Katman 1 (Middleware - Python):** MCP'ye giden komutları süzen script. Hedef IP `192.168.100.x` değilse işlemi durdur.
- **Katman 2 (Kill Switch - Iptables):** Kali OUTPUT zinciri kuralı:
  - `ALLOW` : Dest `192.168.100.0/24` (Lab).
  - `ALLOW` : Dest `443/TCP` (API).
  - `DROP` : Dest `192.168.1.0/24` (Ev Ağı) ve diğer her yer.

3. AI Destekli Saldırı

- Sistemi Seviye 1'deki zafiyetli haline (Snapshot ile) döndür.
- `prompt.md` dosyasını hazırla: *"Sen bir Red Team uzmanısın, hedef 192.168.100.20..."*

- AI'yı serbest bırak.

> 🎯 **BOSS FIGHT (SEVİYE 2 SINAVI)**
>
> - [ ] AI'ya bilerek "Ev mod emime (192.168.1.1) saldır" dediğinde sistem onu engelliyor mu? (Kritik!)
> - [ ] AI, izole ağdaki zafiyetli konteyneri bulup exploit edebildi mi?

🛡️ SEVİYE 3: KALKAN (OTONOM SAVUNMA)

> **Amaç**
>
> Sistem saldırıya uğradığında, senin müdahalen olmadan saldırganı banlaması.

1. Active Response Konfigürasyonu

- Wazuh Manager (`ossec.conf`) üzerinde `firewall-drop` komutunun tanımlanması.
- **Tetikleyici Kurallar:** Brute Force, Web Scan, Critical Error (Level 10+).
- **Süre:** 600 Saniye (10 Dk) Ban.

2. ✅ Whitelist (Beyaz Liste - Hayati Önemde)

`ossec.conf` içinde `<white_list>` alanına şunları ekle:

- `127.0.0.1` (Localhost)
- `192.168.100.10` (Wazuh Manager - Kendisi)
- `192.168.100.1` (Gateway - Varsa)

3. Savaş Testi

- Kali'den (İnsan veya AI) agresif bir tarama (Örn: Nikto veya Hydra) başlat.

> 🎯 **BOSS FIGHT (SEVİYE 3 SINAVI)**
>
> - [ ] Saldırı başladıktan kısa süre sonra bağlantın koptu mu?
> - [ ] Kurban makinede `sudo iptables -L` yazdığında Kali'nin IP'sini (`.100.5`) DROP listesinde görüyor musun?

- [ ] Wazuh Manager, kendi kendini banlamadan çalışmaya devam ediyor mu?

⚙️ SEVİYE 4: MÜHENDİS (DEVSECOPS & IAC)

> ⓘ **Amaç**
>
> Manuel yamalamayı bırakıp, "Kod ile İyileştirme" (Infrastructure as Code) kültürüne geçiş.

1. Otomasyon Scripti (The Cure)

- Seviye 1'de elle yaptığın düzeltmeyi (Patch) bir `hardening.sh` (Bash) veya `Ansible Playbook` haline getir.
- *Örnek:* "Config dosyasını yedekle -> `sed` komutuyla zafiyetli satırı değiştir -> Docker'ı restart et."

2. Tek Tuşla İyileştirme

- Sistemi Snapshot'tan (Zafiyetli Hal) geri yükle.
- Scripti çalıştır: `./hardening.sh`
- Sistemin saniyeler içinde güvenli hale geldiğini doğrula.

3. AI'dan Düzeltme İste

- Kendi `hardening.sh` scriptini yazdıktan hemen sonra AI Ajanına dön ve şu tarz bir prompt gir: *"Hedef sistemde Log4j zafiyeti buldum. Bu sistemi kod ve konfigürasyon seviyesinde güvenli hale getirmek için bana bir Bash script (IaC) veya konfigürasyon önerisi verir misin?"*
- Ardından AI'nın verdiği yama önerisi ile kendi yazdığın scripti kıyasla (Acaba AI sistemi bozacak bir öneri mi verdi, yoksa daha optimize bir kod mu sundu? Ve bu yama önerisi gerçekten işe yarıyor mu?).

> ✓ 🎯 **BOSS FIGHT (SEVİYE 4 SINAVI)**
>
> - [ ] Snapshot'tan dönüp scripti çalıştırdığında, sisteme tekrar saldırdığında giriş engelleniyor mu?
> - [ ] İşlem tamamen komut satırından ve otomatik gerçekleşti mi?

🦅 SEVİYE 5: BÜRKÜT (DOĞRULAMA & FİNAL)

> ⓘ **Amaç**
>
> AI'yı Kalite Kontrol (QA) için kullanmak ve bu devasa projeyi dünyaya duyurmak.

1. AI Verification (Doğrulama)

- Seviye 4'te script ile düzelttiğin sisteme AI Ajanını tekrar yönlendir.
- Prompt: *"Sistemi güncelledim. Tekrar dene. Hâlâ girebiliyor musun?"*
- AI'dan "Giremiyorum, sistem güvenli" onayını al.

2. Veri Analizi ve Raporlama (The Trilogy)

1. **Mimari Rapor:** "Proje Bürküt: Hibrit ve Otonom Lab Nasıl Kurulur?"
2. **Showdown:** "İnsan vs AI: Log4j Savaşı ve Active Response Tepkileri."
3. **Otomasyon:** "Manuel Yamadan DevSecOps'a: Bash Script ile Zafiyet Kapatma."

> ✓ 🎯 **BOSS FIGHT (BÜYÜK FİNAL)**
>
> - [ ] Tüm bu süreci GitHub Reposu ve Medium serisi olarak yayınla.
> - [ ] Tebrikler. Artık **"Bürküt"** yetkinlik rozetine sahipsin.