### **Grype**

**Grype**, konteyner imajları ve dosya sistemlerinde bulunan yazılım bileşenlerini tarayarak **bilinen güvenlik açıklarını** (CVE) tespit eden açık kaynaklı bir araçtır. Anchore tarafından geliştirilmiştir. Docker, OCI imajları veya yerel dosya sistemlerinde çalışan projelerde açık taraması yapmak için idealdir.

Grype; veritabanı olarak NVD (National Vulnerability Database), GitHub Advisory Database, Alpine SecDB gibi güvenlik veri kaynaklarını kullanır. Ayrıca Syft adlı bağımlılık analiz aracını arka planda kullanarak hangi bileşenlerin yüklü olduğunu analiz eder.

### Grype ne işe yarar?

Bir konteyner imajı, içinde yüzlerce paket barındırabilir. Bu paketlerden biri güncel değilse ve bilinen bir CVE içeriyorsa, sistem güvenliği riske girebilir. Grype bu paketleri tarar ve şu soruları yanıtlar:

* İmaj içinde bilinen zafiyet içeren bir yazılım var mı?
* Açık bulunan bileşen hangi sürümde?
* Güvenli sürüm hangisi?
* Zafiyetin şiddet seviyesi ne?

Bu tür analizler, yazılım geliştirme sürecinde CI/CD pipeline’larına dahil edilerek erken aşamada güvenlik sorunlarını fark etmek için kullanılır.

### **Kurulum**

Grype'i kurmak için sisteminize uygun binary’yi indirerek ya da brew, apt, docker gibi yöntemlerle kurabilirsiniz. Örneğin:

* brew install grype

### **Örnek Komut**

aşağıdaki komut, bir Docker imajı olan python:3.8-slim üzerinde güvenlik açığı taraması yapar:

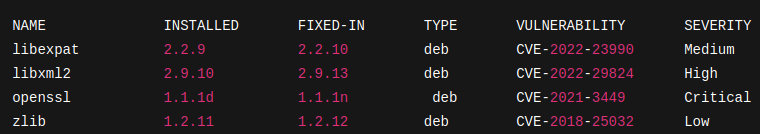
* grype python:3.8-slim

ya da yerel bir dizini taramak için:

* grype dir:/home/kullanici/proje

### **Örnek Çıktı**

tarama sonucunda terminalde aşağıdakine benzer bir tablo çıkar:



bu çıktıya göre:

* libxml2 2.9.10 sürümü yüksek seviyeli bir açık içeriyor.
* openssl bileşeninde kritik düzeyde bir CVE var.
* hangi sürümde düzeltildiği “FIXED-IN” kolonunda belirtiliyor.

### **Örnek JSON Çıktı**

makineler tarafından işlenebilir formatta çıktı almak için -o json kullanılır:

* grype python:3.8-slim -o json > output.json

### **Sık Kullanılan Parametreler**

* -o <format>  
  Çıktı formatını belirler.  
  Desteklenenler: table (varsayılan), json, cyclonedx, sarif, template
* -q  
  Sessiz mod. Bilgi mesajlarını gizler.
* --only-fixed  
  Yalnızca çözümü olan açıkları listeler.
* --fail-on <seviye>  
  Belirtilen seviyeden daha yüksek açık bulunduğunda hata kodu döner (CI/CD kullanımı için).  
  Örnek: --fail-on high
* --add-cpes-if-none  
  Eksik CPE (Common Platform Enumeration) varsa otomatik olarak tahmin eder.
* --config <config.yaml>  
  Özel yapılandırma dosyası kullanır (proxy, veri kaynakları, vb.).

**Grype**, özellikle konteyner güvenliği ve yazılım bileşenlerinin CVE taraması açısından en hafif ve etkili araçlardan biridir. Docker imajları, dosya sistemleri ve derlenmiş projelerin içinde gömülü açıkları tespit eder. CI/CD süreçlerine kolayca entegre edilebilir. Yazılım geliştirme ve DevSecOps ekipleri için vazgeçilmezdir.