Uygulama Laboratuvarı: Güvenlik Açığı Tarama ve Düzeltme

Tahmini süre: 30 dakika

Bu laboratuvarda, GitHub'daki kodu ciddiyet sırasına göre güvenlik açıkları için nasıl kontrol edeceğinizi ve bu açıkları nasıl düzelteceğinizi öğreneceksiniz.

Öğrenme Hedefleri

Bu laboratuvarı tamamladıktan sonra şunları yapabileceksiniz:

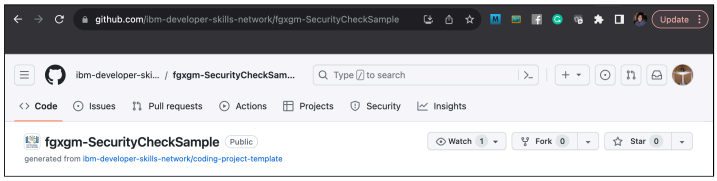
- Kodunuzda bir güvenlik açığı taraması gerçekleştirmek
- Güvenlik açığı riskini azaltma konusunda en iyi uygulamaları içselleştirmek
- Koddaki güvenlik açıklarını düzeltmek

Ön Koşullar:

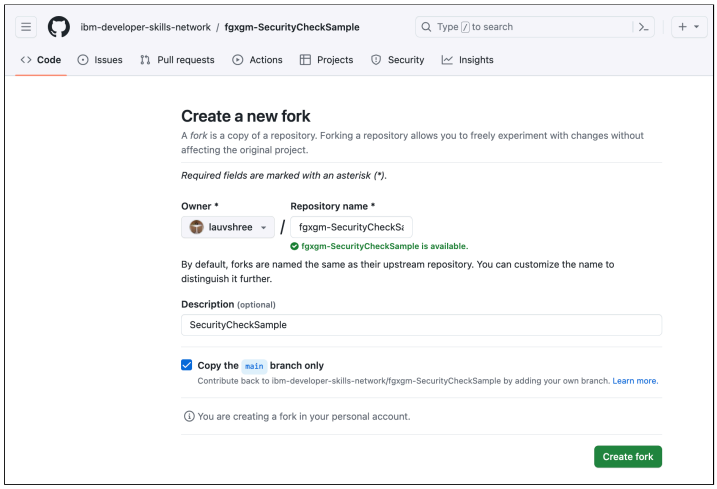
Bir GitHub hesabınızın yanı sıra GitHub hesabınızı kullanmak için kimlik doğrulaması yapılmış bir Snyk hesabınız olmalıdır. Lütfen devam etmeden önce bu laboratuvarı tamamladığınızdan emin olun.

Görev 1: Bir depo kopyası alın

1. https://github.com/ibm-developer-skills-network/fgxgm-SecurityCheckSample.git adresine gidin.
2. Depoyu (veya repo) çatal (fork) yaparak kendi kopyanızı alın. Sadece kendi depolarınızı veya herkese açık depoları tarayabileceğinizi unutmayın. Bu depo, dockerize edilip bulutta dağıtılması gereken basit bir sunucu tarafı uygulamasına sahiptir.



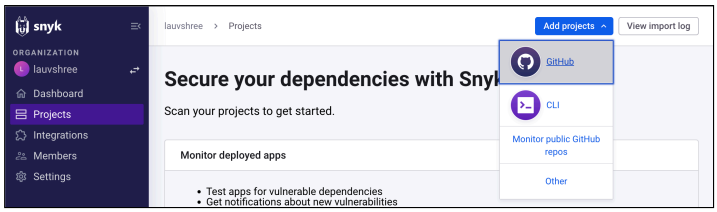
3. Çatal oluşturmayı onaylamanız isteniyor. Ayrıntıları okuyun ve `Create Fork` butonuna tıklayarak işlemi onaylayın.



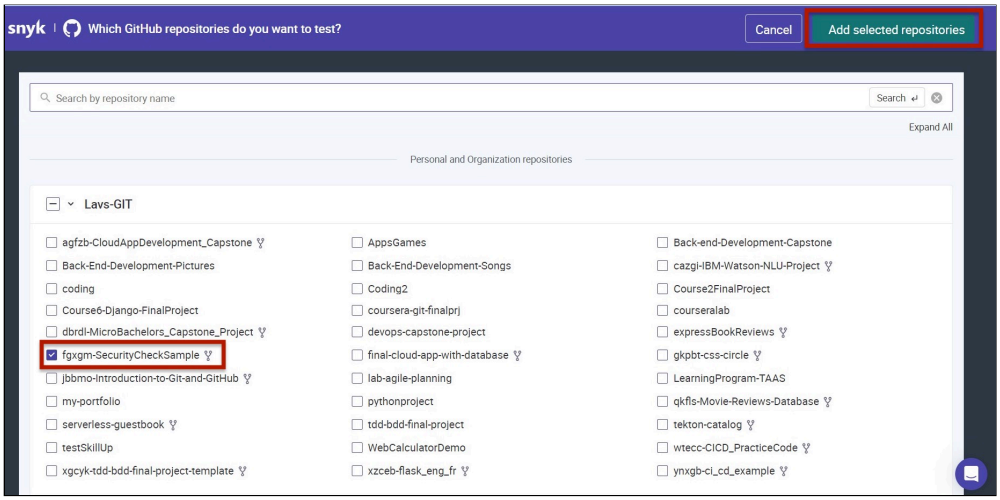
Artık kendi depo kopyanıza sahipsiniz ve bu depoda yaptığınız değişiklikler, kopyaladığınız kaynağı değiştirmeyecektir.

Görev 2: Repo'yu Tarayın

1. Artık kendi kopyanız olan depoya sahip olduğunuza göre, https://app.snyk.io/login adresine gidin ve GitHub kimlik bilgilerinizi kullanarak giriş yapın.
2. Snyk'te, `Projeleri Ekle` butonuna tıklayın ve eklemek istediğiniz depoyu kaynak olarak `GitHub` seçin.



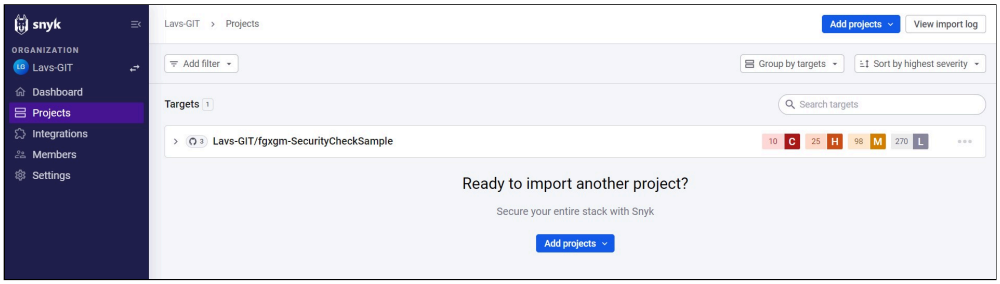
3. Forkladığınız depoyu seçin ve `Seçilen Depoları Ekle` butonuna tıklayın.



4. Proje eklendikten sonra, Snyk tarafından içe aktarılır ve taranır.

Bu birkaç saniye sürebilir. Tarama tamamlandığında, taramanın raporunu göreceksiniz.

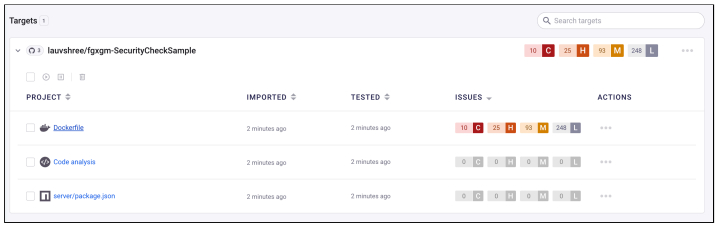
5. Tarama tamamlandığında, aşağıdaki gibi bir sorun raporu göreceksiniz.



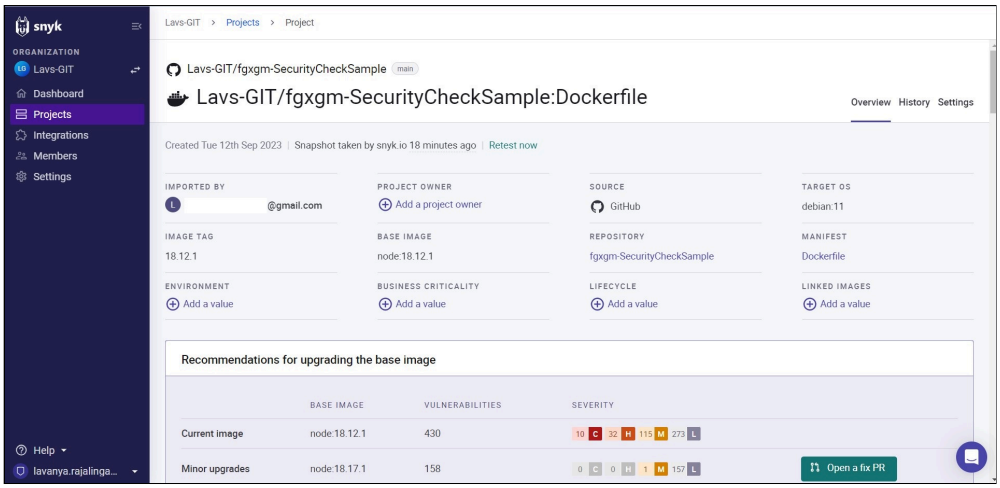
Görev 3: Tarama raporunu görüntüle

1. Güvenlik açıklarının nerede olduğunu ve ciddiyetini belirten ayrıntılı raporu görmek için repo adının önündeki ok simgesine tıklayın. Her dosya için gösterilen 4 farklı güvenlik açığı vardır:

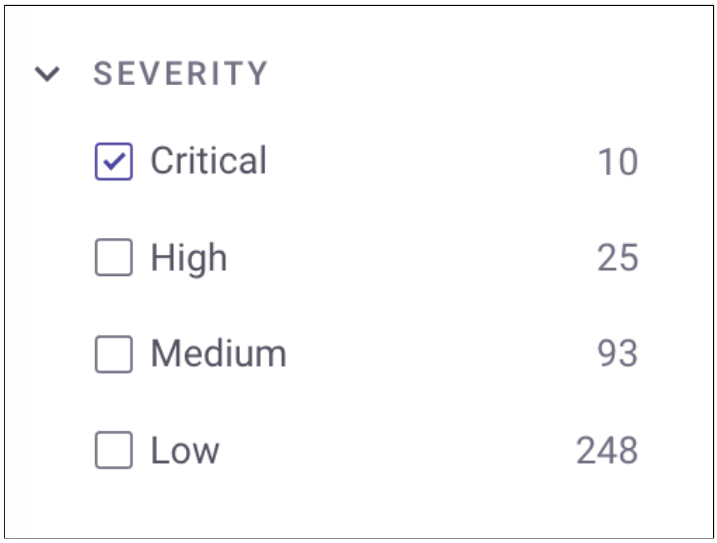
- Kritik
- Yüksek
- Orta
- Düşük



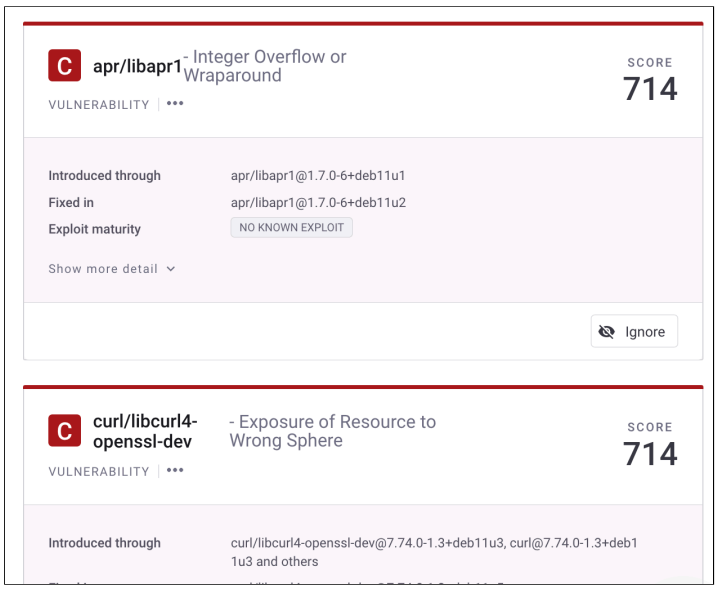
2. Tarama yapılmış güvenlik açıklarını görmek için `Dockerfile` dosyasına tıklayın.



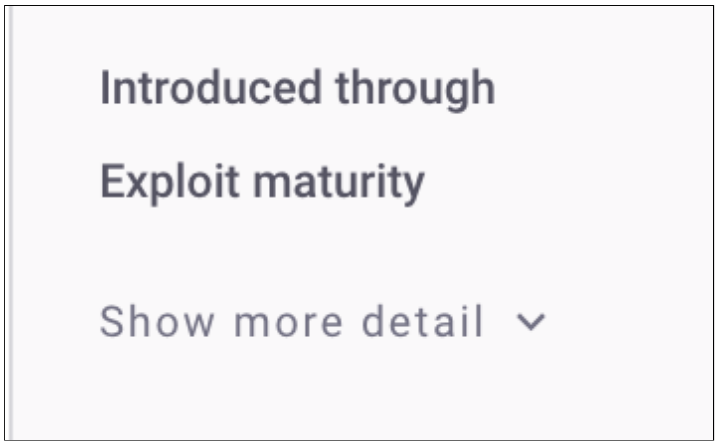
3. Sonuçlar, belirli bir tür güvenlik açığının ayrıntılı görünümünü elde etmek için ciddiyetine göre filtrelenebilir.



4. Listelenen hataları ve bunların hangi tür güvenlik açığı veya güvenlik riski olabileceğini görebilirsiniz.



5. Her hata için, hatanın sonuçları hakkında tavsiyelerde bulunan ve hatanın nasıl düzeltilebileceğini öneren ayrıntılı bir rapor görebilirsiniz.



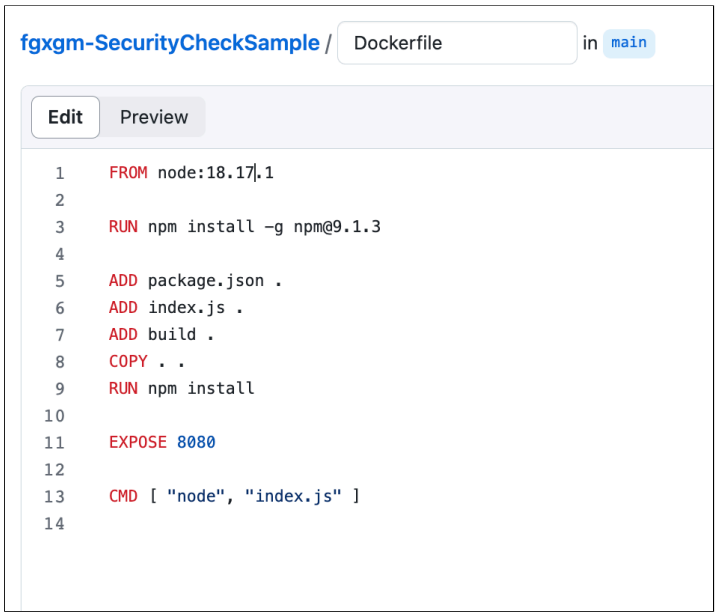
6. Sayfanın üst kısmına kaydırdığınızda, çoğunlukla en çok hatayı ortadan kaldırmak için takip edebileceğiniz büyük bir öneri olacaktır. Bu, çoğu sorunun düzeltildiği en son sürüme dayanacaktır.

7. Yukarıdaki örnekte görüldüğü gibi, node sürümünü 18.12.1'den 18.17.1'e yükseltmenizi önerir.

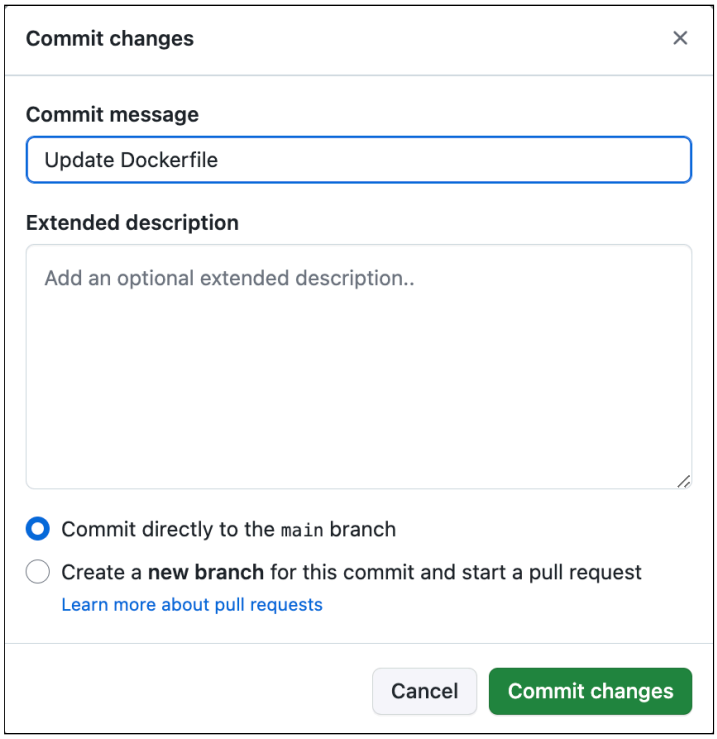
Görev 4: Güvenlik açıklarını düzeltin

1. GitHub'daki depoya geri dönün. `Dockerfile`'a tıklayarak açın ve görüntüleyin.

2. Sağ üstteki kalem simgesine tıklayarak düzenleyin ve node sürümünü önerildiği gibi `18.17.1` olarak değiştirin.

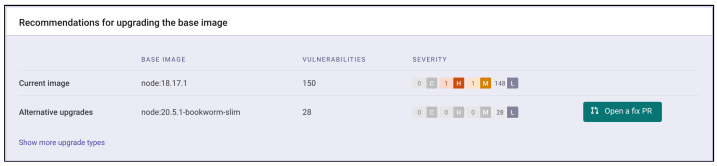


3. Değişiklikleri kalıcı hale getirmek için bunları deponuzda taahhüt edin.



Görev 5: Güvenlik açıklarının düzeltildiğini doğrulama

1. Snyk taramasını yaptığınız tarayıcı sekmesine geri dönün. Eğer sekmeyi kapattıysanız, yeni bir tane açabilirsiniz.
2. Tarama yeniden çalıştırılacak ve güncellenmiş bir rapor mevcut olacaktır.
3. Yeni öneriler olup olmadığını kontrol edin.



4. Git üzerinde değişiklik yapın ve commit edin.
5. Tekrar tekrar önerilen değişiklikleri yapmaya devam edin ve proje raporunu tekrar kontrol edin. C, H veya M sorunları kalmadığında, projenin hazır olduğu kabul edilir.

Görev 6 (İsteğe Bağlı): Düşük öncelikli sorunları düzeltin

1. Öneriye dayanarak, `L` ile işaretlenmiş düşük öncelikli sorunları düzeltip düzeltemeyeceğinizi kontrol edin.

Tebrikler! Bu laboratuvar çalışmasında, bir git kod deposunu güvenlik açıkları için taramayı ve güvenlik açıklarını düzeltmeyi öğrendiniz.

Yazar(lar)

Lavanya T S