T.C.

ONDOKUZ MAYIS ÜNİVERSİTESİ

MÜHENDİSLİK FAKÜLTESİ

BİLGİSAYAR MÜHENDİSLİĞİ BÖLÜMÜ

Web Programlama Laboratuvarı

Güvenli Yazılım Geliştirme Ve Güvenlik Testleri

Deney Föyü-4

ALEYNA KAHRAMAN

20060355

1) Zafiyet testi ve sızma testinin farkları nelerdir?

Zafiyet testi (vulnerability assessment) ve sızma testi (penetration testing) bilgi güvenliği alanında iki farklı ancak birbirini tamamlayan güvenlik testi türüdür. İşte bu iki kavramın temel farkları:

- **Amaç:**
  - **Zafiyet Testi:** Zafiyet testi, bir sistemdeki veya ağdaki güvenlik açıklarını tespit etmeyi amaçlar. Bu test, genellikle otomatize edilmiş araçlar kullanılarak gerçekleştirilir ve organizasyonun hangi noktalarda savunmasız olduğunu belirlemek için taranır.
  - **Sızma Testi:** Sızma testi, saldırganın bakış açısını simüle etmeyi amaçlar. Bu test, saldırganın gerçekleştirebileceği saldırı senaryolarını, saldırı vektörlerini ve potansiyel tehlikeleri değerlendirmeyi içerir.
- **Yaklaşım:**
  - **Zafiyet Testi:** Zafiyet testi genellikle daha pasif bir yaklaşımı benimser. Otomatize edilmiş araçlar kullanılarak sistemdeki zafiyetler belirlenir ve raporlanır.
  - **Sızma Testi:** Sızma testi, aktif bir saldırıcı gibi davranır. Bu test, manuel test tekniklerini içerir ve gerçek dünya saldırılarını simüle etmeyi amaçlar.
- **Zaman Çerçevesi:**
  - **Zafiyet Testi:** Zafiyet testleri genellikle periyodik olarak yapılır, belirli aralıklarla sistemdeki güvenlik durumunu kontrol etmek için kullanılır.
  - **Sızma Testi:** Sızma testleri genellikle belirli bir hedef veya uygulama için belirli bir zaman diliminde gerçekleştirilir. Daha spesifik ve odaklı bir yaklaşımdır.
- **Sonuçlar:**
  - **Zafiyet Testi:** Zafiyet testi sonuçları, sistemdeki güvenlik açıklarını ve zafiyetleri içerir. Bu raporlar, savunma eksikliklerini ve iyileştirmeleri belirlemek için kullanılır.
  - **Sızma Testi:** Sızma testi sonuçları, gerçek saldırı senaryolarına dayanarak organizasyonun ne kadar dirençli olduğunu gösterir. Bu raporlar, belirli saldırı vektörlerine karşı alınması gereken önlemleri ortaya koyar.

Her iki test de bir organizasyonun bilişim güvenliği stratejisinin önemli bir parçasıdır ve birbirini tamamlar. Zafiyet testi, genel savunma durumunu değerlendirmeye yardımcı olurken, sızma testi ise gerçek dünya saldırı senaryolarını simüle ederek organizasyonun gerçek direncini ölçmeye yardımcı olur.

2) Nmap aracı ile yerel ağ üzerinde şu işlemleri gerçekleştiriniz:

a. Ağda bulunan bilgisayarların keşfini gerçekleştiriniz.

```
ubuntu@DESKTOP-VMO3C7M:~$ nmap -sn 192.168.1.0/24
Starting Nmap 7.80 ( https://nmap.org ) at 2023-12-14 22:59 +03
Nmap scan report for 192.168.1.2
Host is up (0.087s latency).
Nmap scan report for 192.168.1.3
Host is up (0.11s latency).
Nmap scan report for 192.168.1.4
Host is up (0.047s latency).
Nmap scan report for 192.168.1.5
Host is up (0.047s latency).
Nmap scan report for 192.168.1.6
Host is up (0.11s latency).
Nmap scan report for 192.168.1.7
Host is up (0.16s latency).
Nmap scan report for 192.168.1.8
Host is up (0.050s latency).
Nmap scan report for 192.168.1.9
Host is up (0.047s latency).
Nmap scan report for 192.168.1.10
Host is up (0.050s latency).
Nmap scan report for 192.168.1.37
Host is up (0.034s latency).
Nmap scan report for 192.168.1.50
Host is up (0.11s latency).
Nmap scan report for 192.168.1.51
Host is up (0.048s latency).
Nmap scan report for 192.168.1.57
Host is up (0.071s latency).
```

...

```
Nmap scan report for 192.168.1.217
Host is up (0.052s latency).
Nmap scan report for 192.168.1.219
Host is up (0.040s latency).
Nmap scan report for 192.168.1.220
Host is up (0.040s latency).
Nmap scan report for 192.168.1.223
Host is up (0.047s latency).
Nmap scan report for 192.168.1.234
Host is up (0.048s latency).
Nmap done: 256 IP addresses (114 hosts up) scanned in 166.23 seconds
ubuntu@DESKTOP-VMO3C7M:~$
```

Bu **nmap** tarama çıktısı, 192.168.1.0/24 IP aralığındaki cihazların durumlarını ve yanıt sürelerini göstermektedir. İşte bu çıktıdaki temel bilgiler ve yorumları:

- **Starting Nmap 7.80 (** https://nmap.org **) at 2023-12-14 22:59 +03:** Bu kısım, **nmap** taramasının başlatıldığı sürüm ve tarih bilgisini gösterir.

- **Nmap scan report for ...:** Bu kısımlar, tarama sonucunda elde edilen IP adreslerini gösterir. Ardından, "Host is up" ifadesi, bu IP adreslerinin aktif olduğunu ve yanıt verdiğini gösterir. "Latency" değerleri, her bir cihaza olan yanıt sürelerini milisaniye cinsinden ifade eder.
- **Nmap done: 256 IP addresses (114 hosts up) scanned in 166.23 seconds:** Bu kısım, taramanın tamamlandığını ve kaç adet IP adresinin tarandığını, kaç adet hostun aktif olduğunu ve taramanın toplamda ne kadar sürede tamamlandığını gösterir.

Yorumlar:

- 256 IP adresi taranmıştır.
- 114 cihaz aktiftir ve yanıt vermektedir. "Host is up" ifadesiyle belirtilmiştir.
- Yanıt süreleri, cihazlara ulaşma sürelerini gösterir. Düşük latency değerleri, cihazların hızlı bir şekilde yanıt verdiğini gösterir.
- Bu tarama sadece cihazların durumlarını ve yanıt sürelerini gösterir, ancak açık olan portlar veya çalışan servisler hakkında bilgi içermez.

Bu tür bir tarama, ağdaki cihazların durumu hakkında genel bir bakış sağlar, ancak daha fazla detay için belirli bir IP adresine yönlendirilen port taraması yapmanız gerekebilir.

b. Bilgisayarınızda çalışan servislerin kullandığı port numaralarını bularak bu servislerin ne işe yaradıklarını anlatınız. Bilgisayarınızda herhangi bir port kullanan servis açık değilse (örneğin SSH) en az 2 servis açmaya çalışınız ve bu servisleri anlatınız.

```
ubuntu@DESKTOP-VMO3C7M:~$ netstat -tulpn
(Not all processes could be identified, non-owned process info
 will not be shown, you would have to be root to see it all.)
Active Internet connections (only servers)
Proto Recv-Q Send-Q Local Address           Foreign Address         State       PID/Program name
tcp        0      0 127.0.0.53:53           0.0.0.0:*               LISTEN      -
udp        0      0 127.0.0.53:53           0.0.0.0:*                           -
udp        0      0 127.0.0.1:323           0.0.0.0:*                           -
udp6       0      0 ::1:323                 :::*                                -
ubuntu@DESKTOP-VMO3C7M:~$ nmap -p 22 localhost
Starting Nmap 7.80 ( https://nmap.org ) at 2023-12-14 23:21 +03
Nmap scan report for localhost (127.0.0.1)
Host is up (0.000049s latency).

PORT   STATE  SERVICE
22/tcp closed ssh

Nmap done: 1 IP address (1 host up) scanned in 0.05 seconds
ubuntu@DESKTOP-VMO3C7M:~$
```

Bu çıktılar, bilgisayarınızda çalışan servisleri ve belirli bir portun durumunu kontrol etmek için kullanılan komutların sonuçlarını göstermektedir. İşte çıktılarla ilgili açıklamalar:

- **netstat -tulpn Çıktısı:**

Bu çıktı, TCP ve UDP protokollerini kullanan, dinleme modunda olan (LISTEN) bağlantıları gösterir. Ancak, süreç bilgileri tamamen görüntülenemiyor çünkü komutun root (süper kullanıcı) yetkileriyle çalıştırılmadığı belirtilmiş.

- **nmap -p 22 localhost Çıktısı:**conds
  Bu çıktı, bilgisayarınızdaki localhost (127.0.0.1) adresindeki 22 numaralı portu (SSH'nin varsayılan portu) kontrol eder. Ancak, bu portun kapalı olduğunu gösterir (closed).

Açıklamalar:

- **netstat** çıktısı, belirli bir IP adresi ve port kombinasyonunu gösterir. Burada DNS servisi (53 numaralı port) ve NTP (Network Time Protocol) servisi (323 numaralı port) dinleme modunda görünüyor.
- **nmap** çıktısı, belirli bir IP adresindeki belirli bir portun durumunu gösterir. 22 numaralı portun "closed" olduğunu belirtir. Bu, SSH servisinin bu portta çalışmadığını gösterir.

c. Tek bir IP (yerel makine IP adresi) kullanarak port taraması yapınız ve elde edilen sonuçları düzgün bir şekilde yorumlayınız.

```
ubuntu@DESKTOP-VMO3C7M:~$ nmap -Pn 192.168.1.100
Starting Nmap 7.80 ( https://nmap.org ) at 2023-12-14 23:34 +03
Nmap scan report for 192.168.1.100
Host is up (0.034s latency).
Not shown: 999 closed ports
PORT     STATE SERVICE
2909/tcp open  funk-dialout

Nmap done: 1 IP address (1 host up) scanned in 48.02 seconds
```

Açıklamalar:

- **Host is up (0.034s latency)**: Belirtilen IP adresine bir yanıt alındığını ve bu IP adresine düşük bir gecikme süresiyle ulaşılabildiğini gösterir.
- **Not shown: 999 closed ports**: Bu, tarama sonucunda 1000 portun sadece bir tanesinin görüntülendiğini belirtir.
- **PORT STATE SERVICE**: Bu bölümde, tarama sonucunda bulunan portların durumları ve kullanılan servisler listelenir.
- **2909/tcp open funk-dialout**: Bu, 2909 numaralı TCP portunun açık olduğunu ve bu port üzerinde "funk-dialout" adında bir servisin çalıştığını gösterir.
- **Nmap done: 1 IP address (1 host up) scanned in 48.02 seconds**: Taramanın tamamlandığını, kaç IP adresinin tarandığını, kaç adet hostun aktif olduğunu ve taramanın ne kadar sürede tamamlandığını gösterir.

Sonuç olarak, 192.168.1.100 IP adresine yapılan taramada sadece 2909 numaralı TCP portunun açık olduğu ve bu port üzerinde "funk-dialout" adında

bir servisin çalıştığı görülmektedir. Açık olan port ve servis isimleri, hedef sistemdeki belirli bir servise işaret edebilir. "funk-dialout" servisinin tam olarak ne olduğunu belirlemek için daha fazla bilgiye ihtiyaç duyabilirsiniz, çünkü bu isim genel bir tanımlama olmayabilir.

d. Yerel makine üzerindeki 1 ve 1000 nolu portlar arasındaki bütün portları tarayan komutu yazınız ve çıkan sonuçları yorumlayınız.

```
ubuntu@DESKTOP-VMO3C7M:~$ nmap -p 1-1000 172.18.143.194
Starting Nmap 7.80 ( https://nmap.org ) at 2023-12-14 18:42 +03
Nmap scan report for 172.18.143.194
Host is up (0.00022s latency).
All 1000 scanned ports on 172.18.143.194 are closed

Nmap done: 1 IP address (1 host up) scanned in 1.19 seconds
```

Bu **nmap** tarama çıktısı, 172.18.143.194 IP adresine yapılan port taramasının sonuçlarını göstermektedir. İşte bu çıktıdaki temel bilgiler ve yorumları:

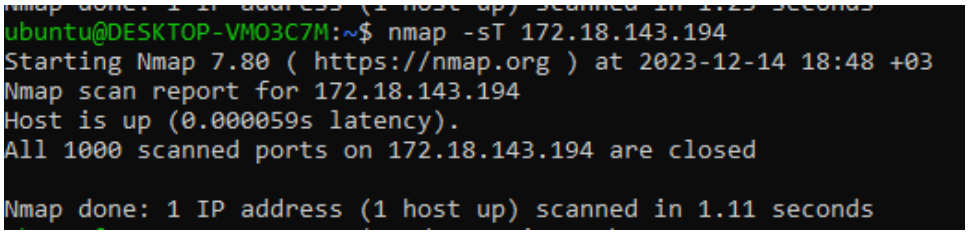
- **Starting Nmap 7.80 (** https://nmap.org **) at 2023-12-14 18:42 +03:** Bu kısım, **nmap** taramasının başlatıldığı sürüm ve tarih bilgisini gösterir.
- **Nmap scan report for 172.18.143.194:** Bu kısım, tarama sonucunda elde edilen IP adresini gösterir. Ayrıca, hedef IP adresine başarıyla bir bağlantı kurulduğunu ifade eder.
- **Host is up (0.00022s latency):** Hedef IP adresinin aktif olduğunu ve yanıt verdiğini gösterir. "Latency" değeri, hedefe olan yanıt süresini milisaniye cinsinden ifade eder.
- **All 1000 scanned ports on 172.18.143.194 are closed:** Bu kısım, tarama sonucunda elde edilen bilgilerin ana noktasını içerir. Bu durumda, 172.18.143.194 IP adresinde taranan 1000 portun tamamının kapalı olduğunu gösterir.
- **Nmap done: 1 IP address (1 host up) scanned in 1.19 seconds:** Bu kısım, taramanın tamamlandığını, kaç adet IP adresinin tarandığını, kaç adet hostun aktif olduğunu ve taramanın toplamda ne kadar sürede tamamlandığını gösterir. Bu örnekte, 1 IP adresi taranmış ve bu IP adresinin üzerindeki hedefin tüm portları 1.19 saniyede taranmıştır.

Yorumlar:

- Tarama sonuçlarına göre, 172.18.143.194 IP adresindeki tüm 1000 port kapalıdır.
- Bu durum, bu IP adresindeki hedef sistemin belirtilen portlarda aktif bir servis çalıştırmadığını veya güvenlik önlemleri nedeniyle bu portlara erişime izin verilmediğini gösterebilir.

- Portların kapalı olması, sistemdeki servislerin güvenlik duvarı veya konfigürasyon nedeniyle erişilemez olduğunu düşündürebilir.

e. TCP bağlantı taraması nedir? TCP bağlantı taraması gerçekleştiriniz.



Bu **nmap** tarama çıktısı, 172.18.143.194 IP adresine yapılan TCP bağlantı taramasının sonuçlarını göstermektedir. İşte bu çıktıdaki temel bilgiler ve yorumları:

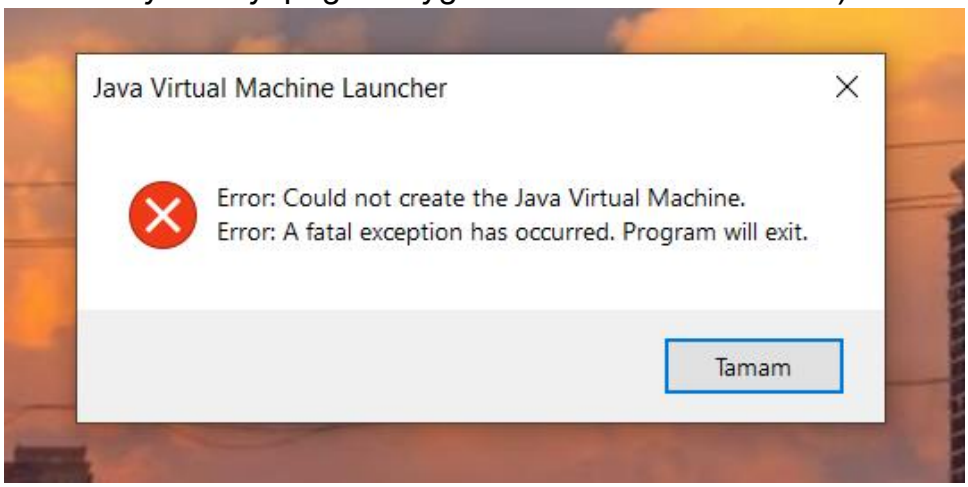
- **Starting Nmap 7.80 (** https://nmap.org **) at 2023-12-14 18:48 +03:** Bu kısım, **nmap** taramasının başlatıldığı sürüm ve tarih bilgisini gösterir.
- **Nmap scan report for 172.18.143.194:** Bu kısım, tarama sonucunda elde edilen IP adresini gösterir. Ayrıca, hedef IP adresine başarıyla bir bağlantı kurulduğunu ifade eder.
- **Host is up (0.000059s latency):** Hedef IP adresinin aktif olduğunu ve yanıt verdiğini gösterir. "Latency" değeri, hedefe olan yanıt süresini milisaniye cinsinden ifade eder.
- **All 1000 scanned ports on 172.18.143.194 are closed:** Bu kısım, tarama sonucunda elde edilen bilgilerin ana noktasını içerir. Bu durumda, 172.18.143.194 IP adresinde taranan 1000 portun tamamının kapalı olduğunu gösterir.
- **Nmap done: 1 IP address (1 host up) scanned in 1.11 seconds:** Bu kısım, taramanın tamamlandığını, kaç adet IP adresinin tarandığını, kaç adet hostun aktif olduğunu ve taramanın toplamda ne kadar sürede tamamlandığını gösterir. Bu örnekte, 1 IP adresi taranmış ve bu IP adresinin üzerindeki hedefin tüm portları 1.11 saniyede taranmıştır.

Yorumlar:

- Tarama sonuçlarına göre, 172.18.143.194 IP adresindeki tüm 1000 port kapalıdır.
- TCP bağlantı taraması, belirli portların TCP bağlantısı kurularak durumlarını kontrol eder. Tüm portların kapalı olması, bu IP adresindeki hedef sistemin belirtilen portlarda aktif bir servis çalıştırmadığını veya güvenlik önlemleri nedeniyle bu portlara erişime izin verilmediğini gösterir.

- Portların kapalı olması, sistemdeki servislerin güvenlik duvarı veya konfigürasyon nedeniyle erişilemez olduğunu düşündürebilir.

3) Vega aracı ile localhost üzerinde çalışan bir web uygulamasına zafiyet testi gerçekleştiriniz ve çıkan sonuçları yorumlayınız. Çıkan hatalardan bir tanesini nasıl kullanabileceğiniz ile ilgili bir örnek veriniz (Kullanacağınız yerel uygulama önceki föylerde yaptığınız uygulamalardan birisi olabilir).



Vega kurulumunda hata aldım. Çözemediğimden dolayı soruyu yapamadım.

4) Kara-kutu ve beyaz-kutu test yöntemleri nedir? Birbirlerinden farklılıkları nelerdir?

Kara kutu test yöntemleri, bir yazılımın işlevselliğini ve performansını dışardan inceleyen ve iç yapısı hakkında bilgi sahibi olmayan bir test yaklaşımını ifade eder. Kara kutu testleri, genellikle kullanıcı perspektifinden yazılımın doğru çalışıp çalışmadığını ve belirli gereksinimlere uyup uymadığını değerlendirmek için kullanılır. İşte kara kutu test yöntemlerinden bazıları:

- **Eşdeğer Sınıf Analizi (Equivalence Class Testing):**
  - İlgili girdi verilerini sınıflara ayırarak her sınıftan bir temsilci kullanma prensibine dayanır. Bu sayede her sınıfın içeriğini temsil eden girdilerle testler yapılır.
- **Sınır Değer Analizi (Boundary Value Analysis):**
  - Sınır değerleri ve sınırların hemen öncesinde ve sonrasındaki değerlerle test yapar. Bu, genellikle hataların sınırlarda ortaya çıkma eğiliminde olduğu varsayımına dayanır.
- **Durum Tablosu Testi (Decision Table Testing):**
  - Farklı durum kombinasyonlarına göre geliştirilen tablolar üzerinden test yapar. Her durum kombinasyonunu içeren tablo, yazılımın doğru davranıp davranmadığını kontrol etmek için kullanılır.

- **Zaman Çizelgesi Testi (Time Box Testing):**
    - Zaman ile ilgili özellikleri test etmek amacıyla yapılır. Örneğin, bir işlem belirli bir süre içinde tamamlanmalıdır. Bu süre içindeki çeşitli durumlar ve senaryolar test edilir.
- **Nesne ve Veri Yapısı Testi (Object and Data Structure Testing):**
    - Yazılımın nesne ve veri yapılarını test eder. Bu, özellikle nesne yönelimli programlamaya dayanan yazılımlar için önemlidir.
- **Senaryo Testi (Scenario Testing):**
    - Gerçek kullanım senaryolarına dayanarak testler yapar. Bu, yazılımın gerçek dünya koşullarında nasıl davrandığını anlamak için kullanılır.
- **Fonksiyonel ve İşlevsel Testler (Functional Testing):**
    - Yazılımın belirli bir işlevselliği sağlayıp sağlamadığını kontrol eder. Kullanıcının beklentilerini karşılayıp karşılamadığını değerlendirir.
- **Performans Testi (Performance Testing):**
    - Yazılımın performansını, yanıt sürelerini, yüksek kullanıcı trafiği altındaki tepkilerini değerlendirir.
- **Güvenlik Testi (Security Testing):**
    - Yazılımın güvenlik açıklarını tespit etmek ve bu açıkları kapatmak için testler yapar.
- **Uyumluluk Testi (Compatibility Testing):**
    - Yazılımın farklı platformlarda, işletim sistemlerinde veya tarayıcılarda uyumlu bir şekilde çalışıp çalışmadığını kontrol eder.

Bu yöntemler, kara kutu testlerini daha sistematik ve kapsamlı hale getirmek için kullanılan yaygın test teknikleridir. Hangi yöntemin kullanılacağı, test edilen sistemin gereksinimlerine, kullanım senaryolarına ve diğer faktörlere bağlı olarak değişebilir.

Beyaz kutu test yöntemleri, yazılımın iç yapısını inceleyen ve kodun her bir bileşeninin doğru çalışıp çalışmadığını değerlendiren test yaklaşımlarıdır. Bu testler genellikle yazılım geliştiricileri veya test mühendisleri tarafından gerçekleştirilir. Beyaz kutu test yöntemleri, kodun içindeki mantık hatalarını, döngü hatalarını, veri akış hatalarını ve diğer programlama hatalarını ortaya çıkarmak için kullanılır. İşte beyaz kutu test yöntemlerinden bazıları:

- **Dallanma Analizi (Branch Coverage):**
    - Yazılımın her bir kod dallanmasının (if, else, switch, vb.) en az bir kez test edildiğini doğrulamayı amaçlar.
- **Yolu İzleme Analizi (Path Coverage):**

    - Kodun farklı yollarının test edildiğinden emin olmak için kullanılır. Her bir fonksiyonun ve kontrol yapısının tüm yollarını içermesi beklenir.
- **Durum Makinesi Tabanlı Testler (State Transition Testing):**
    - Yazılımın durum makinesine dayanarak farklı durum geçişlerini ve durumları test eder.
- **Döngü Testi (Loop Testing):**
    - Yazılımdaki döngülerin doğru çalışıp çalışmadığını kontrol eder. Bu, döngülerin minimum, maksimum ve orta değerleri içerecek şekilde test edilmesini içerir.
- **Veri Akışı Testi (Data Flow Testing):**
    - Yazılımda veri akışını analiz ederek, değişkenlerin doğru bir şekilde kullanılıp kullanılmadığını ve değişken değerlerinin beklenen değerlere ulaşıp ulaşmadığını kontrol eder.
- **Kod İncelemesi (Code Review):**
    - Yazılım kodunu dikkatlice inceleyerek hataları, mantık hatalarını ve hatalı kullanımları tespit etmeyi amaçlar. Bu bir test yöntemi olarak kabul edilmese de, yazılım kalitesini artırmak için sıkça kullanılır.
- **Hata Enjeksiyonu (Fault Injection):**
    - Bilerek hatalı kodlar ekleyerek yazılımın bu hatalara nasıl tepki verdiğini test eder. Bu, yazılımın dayanıklılığını ve hata kurtarma mekanizmalarını kontrol etmek için kullanılır.
- **Yerine Koyma Testi (Mutation Testing):**
    - Kod içinde küçük değişiklikler yaparak (mutasyonlar ekleyerek), yazılımın bu değişikliklere nasıl tepki verdiğini değerlendirir. Başarılı bir test, bu değişikliklere karşı duyarlılığı tespit edebilmelidir.
- **Kod Analizi (Code Analysis):**
    - Otomatik analiz araçları kullanarak, yazılım kodunu derinlemesine inceleyerek olası hataları tespit etmeye çalışır.

Bu beyaz kutu test yöntemleri, yazılımın iç yapısını anlamak ve hataları tespit etmek için kullanılır. Genellikle yazılımın güvenilirliğini artırmak, performansını optimize etmek ve genel kalitesini yükseltmek amacıyla uygulanırlar.

Farkları:

- **Bakış Açısı:**
    - **Kara Kutu Testi (Black-Box Testing):** Yazılımın dış davranışına odaklanır. Test eden, iç yapıyı ve kodu bilmez; sadece girdileri girer, çıktıları kontrol eder ve yazılımın belirli bir işlevselliği veya gereksinimleri nasıl karşıladığını değerlendirir.

    - **Beyaz Kutu Testi (White-Box Testing):** Yazılımın iç yapısına odaklanır. Test eden, kodu ve yazılımın iç işleyişini bilmekte ve bu bilgiyi kullanarak test senaryolarını oluşturmaktadır.
- **Bilgi Seviyesi:**
    - **Kara Kutu Testi:** Test eden, iç yapı hakkında bilgi sahibi olmaz. Test senaryolarını, belgeleri ve gereksinimleri kullanarak hazırlar.
    - **Beyaz Kutu Testi:** Test eden, iç yapı ve kod hakkında bilgi sahibidir. Bu bilgi, test senaryolarını ve stratejilerini oluştururken kullanılır.
- **Test Odak Noktası:**
    - **Kara Kutu Testi:** Yazılımın işlevselliği, kullanılabilirlik, performans gibi dış özelliklere odaklanır.
    - **Beyaz Kutu Testi:** Kodun doğruluğu, iç kontrol yapıları, algoritmalar gibi iç özelliklere odaklanır.
- **Giriş ve Çıktı:**
    - **Kara Kutu Testi:** Test eden, yazılımın girdilerini belirler ve çıktıları değerlendirir. İç yapının nasıl çalıştığına dair bilgiye sahip değildir.
    - **Beyaz Kutu Testi:** Test eden, girdilerin ve çıktıların yanı sıra kodun içindeki işleyişi de göz önünde bulundurur.
- **Bilgi Dağılımı:**
    - **Kara Kutu Testi:** Test ekibi, yazılımın geliştirilmesi sürecinde genellikle dıştandır ve geliştiricilerle sınırlı bir etkileşime sahiptir.
    - **Beyaz Kutu Testi:** Test ekibi, geliştirme ekibiyle daha sıkı bir etkileşime sahiptir, çünkü iç yapıyı anlamak ve kodu test etmek için bu bilgiye ihtiyaçları vardır.

Bu farklar, her iki test türünün avantajlarını ve dezavantajlarını belirler. Genellikle bir proje, her iki test türünü de kapsayabilir ve birbirini tamamlayıcı olarak kullanabilir.

5) (Bonus) Shodan.io aracını araştırınız. Örnek bir kullanım senaryosu gösteriniz. Elde ettiğiniz sonuçları yorumlayınız.

Shodan, İnternet'e bağlı cihazlar için bir arama motorudur. Google ve Bing gibi web arama motorları web sitelerini bulmak için mükemmeldir. Peki ya hangi ülkelerin daha fazla bağlantılı hale geldiğini ölçmekle ilgileniyorsanız? Veya Microsoft IIS'nin hangi sürümünün en popüler olduğunu bilmek mi istiyorsunuz? Veya kötü amaçlı yazılımlara karşı kontrol sunucularını mı bulmak istiyorsunuz? Belki yeni bir güvenlik açığı ortaya çıktı ve bunun kaç ana bilgisayarı etkileyebileceğini görmek istiyorsunuz? Geleneksel web arama motorları bu soruları yanıtlamanıza izin vermez.

Shodan, internete doğrudan bağlı tüm cihazlar hakkında bilgi toplar. Bir cihaz doğrudan İnternet'e bağlıysa Shodan, kamuya açık çeşitli bilgiler için cihazı sorgular. İndekslenen cihaz türleri büyük ölçüde farklılık gösterebilir: küçük masaüstü bilgisayarlardan nükleer enerji santrallerine ve aradaki her şeye kadar.

Verilerin büyük kısmı , cihazda çalışan bir yazılımla ilgili meta veriler olan banner'lardan alınır. Bu, sunucu yazılımı hakkında bilgi, hizmetin hangi seçenekleri desteklediği, bir karşılama mesajı veya müşterinin sunucuyla etkileşime girmeden önce bilmek isteyeceği herhangi bir şey olabilir.

Google'dan ne farkı var?
En temel fark, Shodan'ın İnternet'i taraması, Google'ın ise World Wide Web'i taramasıdır. Ancak World Wide Web'e güç sağlayan cihazlar, gerçekte İnternet'e bağlı olanların yalnızca küçük bir kısmını oluşturur. Shodan'ın amacı İnternet'in tam bir resmini sunmaktır.

Google'dan bir diğer fark ise Shodan'ın arama sorgusu sözdizimini anlamanızı gerektirmesidir. Örneğin, Shodan'a enerji santraline girip doğru sonuçları almayı bekleyemezsiniz. Shodan'ı mühendisler/geliştiriciler için ve arama sorgusu sözdizimini anlamanız için ihtiyaç duyduğunuz verilerden en iyi şekilde yararlanmanız için tasarladık.