Ders 8 - Input Validation Issues - Part 2

Dersin Hedefi

Diva uygulamasındaki “7. Input Validation Issues - Part 2” seçeneğine tıklayın. Bu seçenek ile açılan website URL adresi girme ve içerik görüntüleme sayfasındaki mekanizmada yer alan bir açıklıktan faydalanarak hassas bir içeriğe erişmeye çalışın.

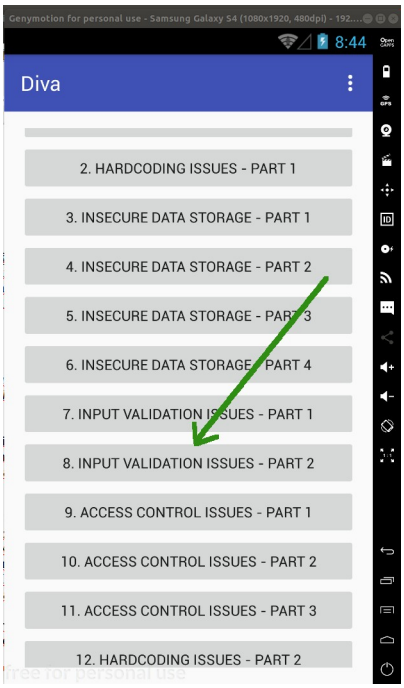
Dersin Açıklaması

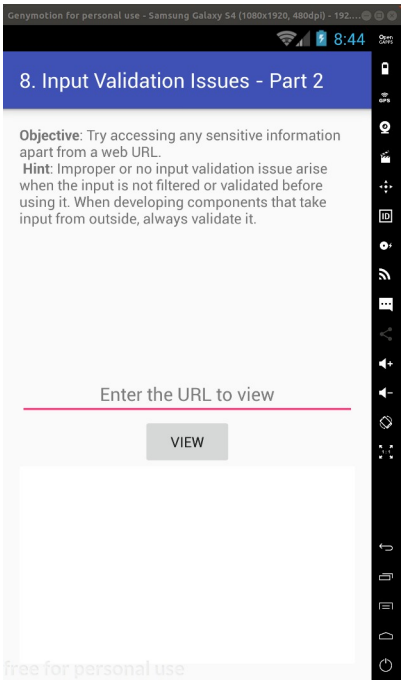
Uygulamlardaki girdi denetleme eksikliği birçok açıklığa sebebiyet verebilir. Örneğin girdi veritabanı sorgularında kullanılıyorsa sql enjeksiyonu açığına, sunucuda komut satırında komut çalıştırmada kullanılıyorsa komut enjeksiyonu açığına, uygulama sayfalarında görüntülemek için depolanıyorsa cross site scripting açığına,… sebebiyet verebilir. Dolayısıyla her uygulama girdi aldığı noktaya bir denetleme mekanizması kurmalıdır.

Bu derste sunulan sayfa uzaktan website sayfasını (dosyasını) dahil ettiğinden girdi kontrolü eksikliği varsa yerel bir dosya dahil edeceğiz ve cihaz içerisindeki gömülü hassas bir dosya içeriğini görüntüleyebileceğiz.

Dersin Çözümü

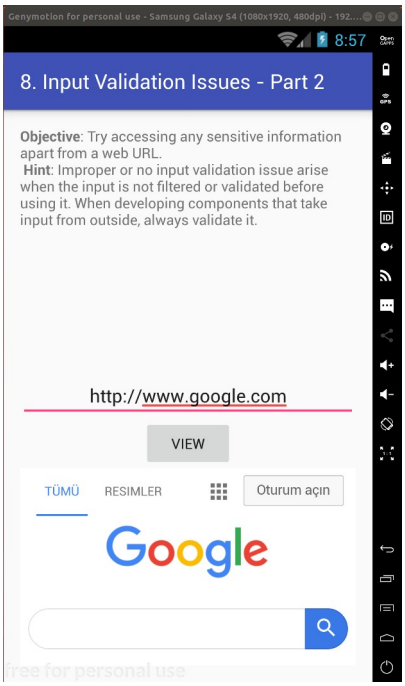
Ders ekranına bir göz atalım.





Uygulama sayfasına göz attığımızda bizden bir website URL adresi istemekte ve View (Görüntüle) butonu ile website içeriğini görüntüleyeceği gözükmektedir.

Örneğin http://google.com website URL adresini girdiğimizde ve View butonuna bastığımızda sayfa bize google web sayfa içeriğini görüntülemektedir.

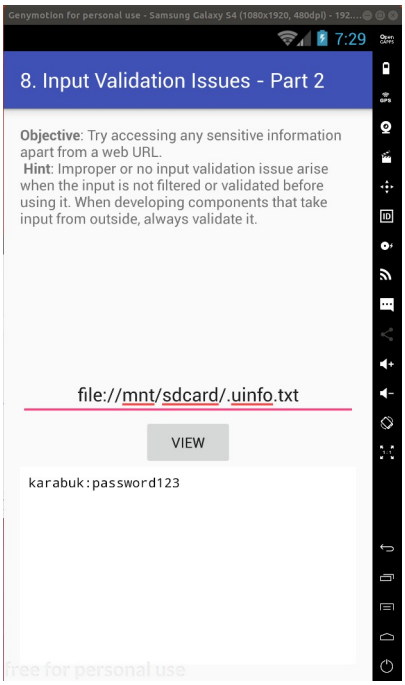


Uygulama sayfasının davranışı gereği belirtilen uzak adresteki web sayfasını dahil ettiğini ve içeriğini görüntülediğini söyleyebiliriz. Bu uygulama sayfasında dosya yolu belirttiğimiz girdi kutusunda denetleme olmazsa dosya dahil etme mekanizmasına uzak bir web sayfasını (dosyasını)

gösterme yerine yerel bir dosyayı gösterme yapılabilir ve yerel bir dosya dahil edilerek içeriği okunabilir.

Uzak bir dosya dahil edileceği zaman http:// veya https:// ön eki kullanılmaktadır. Bu sayede uzaktaki bir websitenin sayfası dahil edilmekte ve görüntülenmektedir. Yereldeki bir dosyayı göstermek ve görüntülemek için file:// ön eki kullanılabilir.

Örneğin daha önceki derslerde diva uygulamasının üçüncü taraf servis hesap bilgilerini cihazda yerel olarak bir dosyada depolamıştık: .uinfo.txt. Bu dosyayı görüntülemeyi deneyelim.



Görüldüğü üzere sayfada sunulan dosya dahil etme mekanizmasına sayfada istenildiği gibi uzak adres girmek yerine yerel adres girerek cihaz içerisindeki bir hassas dosyayı okuyabildik.

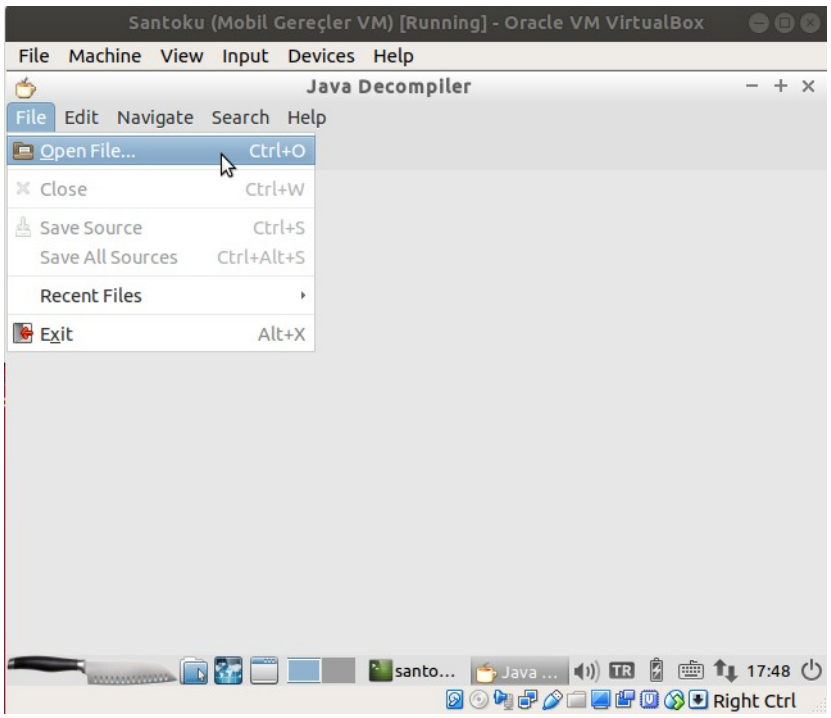
Şimdi bu uygulama sayfasında filtreleme / denetleme eksikliği hangi kod satırında mevcut tespit edelim. Bunun için mobil cihazdan çalışan uygulamanın kaynak kodlarını elde edelim.

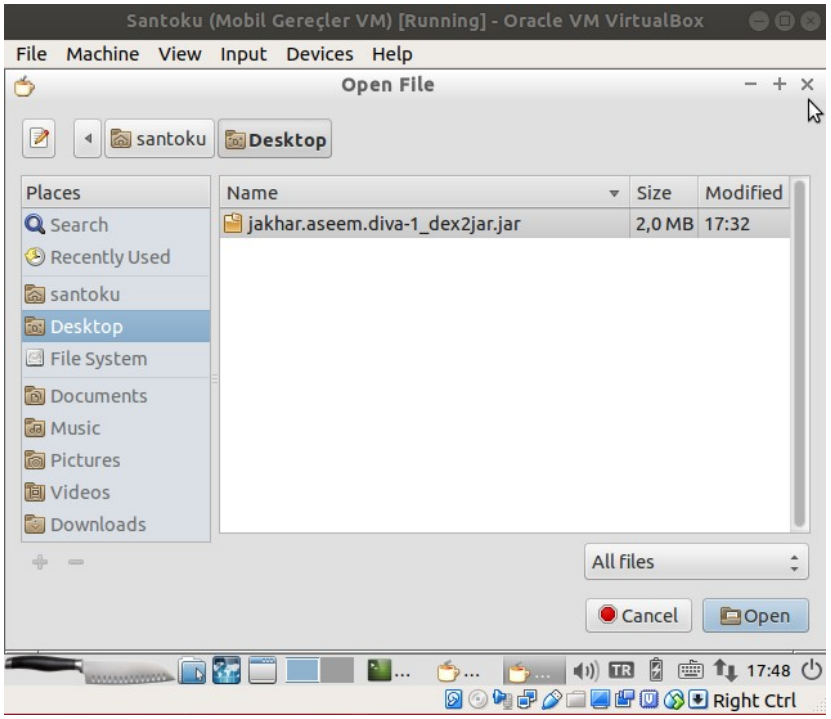
Bu makale dizisinin en başında yaptığımız uygulama binary dosyasını (.apk dosyasını) bilgisayara çekme, üzerinde tersine mühendislik yapma ve uygulamanın okunabilir java dosyalarını elde etme işini tekrarladığımızı varsayalım. Uygulamanın okunabilir java dosyalarını JD-GUI editörü ile inceleyerek bu uygulama sayfasını (“Input Validation Issues > Part 1” sayfasını) sunan / kontrol eden java dosyasını tespit edelim. Bahsedilen işlemlerin ayrıntısı için bkz. Başlangıç: Android Uygulamalarda Tersine Mühendislik ile Okunabilir Java Kaynak Kodları Elde Etme (Dex2Jar, JD-GUI, ApkTool)

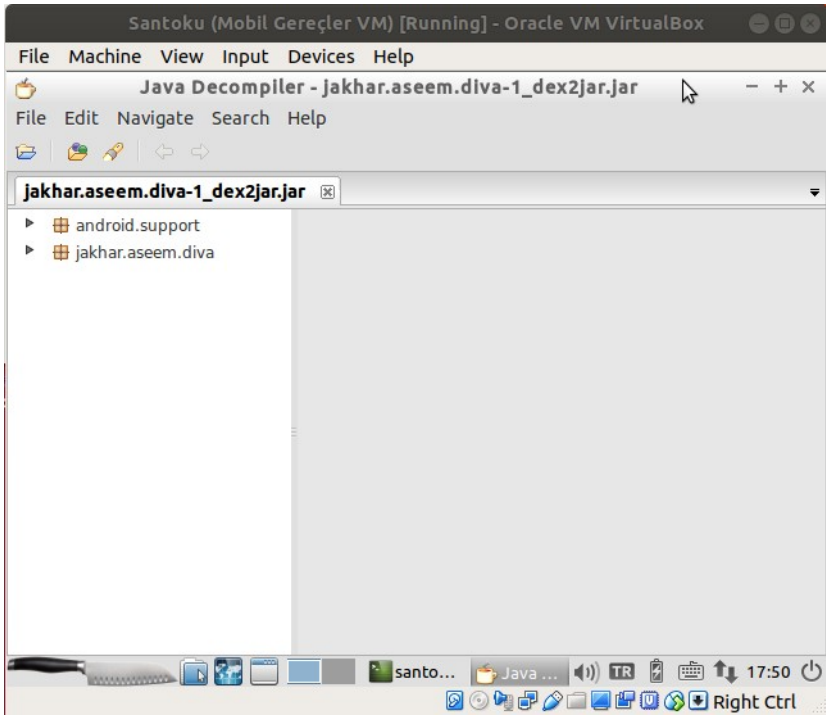
Santoku Linux Terminal:

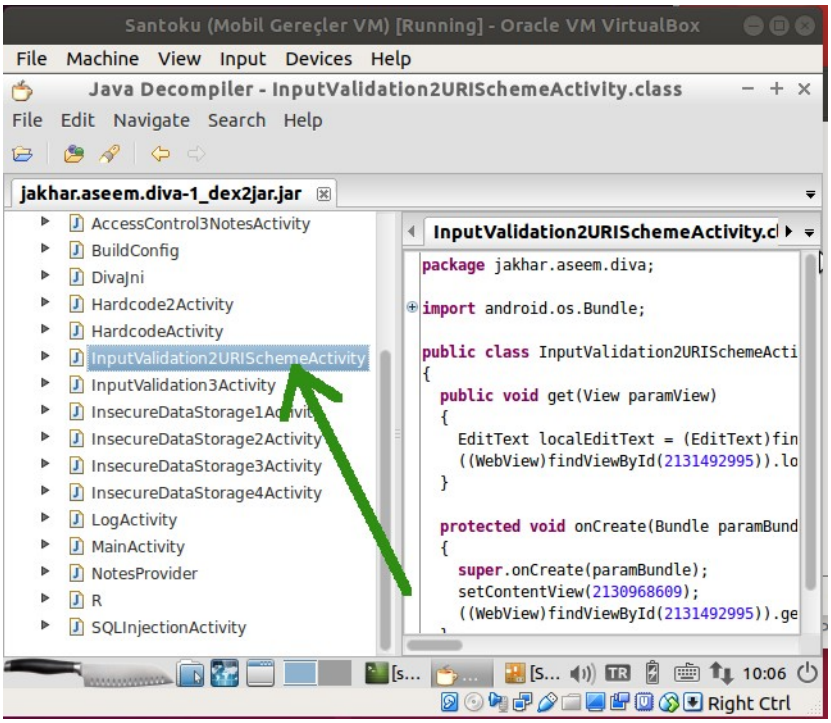
> jd-gui

Çıktı:


Santoku (Mobil Gereçler VM) [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
Java Decompiler
File Edit Navigate Search Help
Open File... Ctrl+O
Close Ctrl+W
Save Source Ctrl+S
Save All Sources Ctrl+Alt+S
Recent Files
Exit Alt+X
santo...
Java ...
TR
17:48
Right Ctrl


Santoku (Mobil Gereçler VM) [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
Open File
santoku
Desktop
Places
Search
Recently Used
santoku
Desktop
File System
Documents
Music
Pictures
Videos
Downloads
Name
Size
Modified
jakhar.aseem.diva-1_dex2jar.jar
2,0 MB
17:32
All files
Cancel
Open
TR
17:48
Right Ctrl





Java kaynak kod dosyaları incelendiğinde uygulamada görüntülüyor olduğumuz ekranın hangi java dosyası tarafından sunulduğu / kontrol edildiği yukarıdaki gibi görülebilir. Bu örnek için java dosyasını bulmak uygulama ekranındaki sayfa ismi ile benzerliği dolayısıyla kolay olacaktır ama gerçek bir senaryoda java dosyaları arasında inceleme yapmak ve doğru java dosyasını bulmak için okumalar yapmak gerekecektir.

Mobil cihaz ekranında görüntülemekte olduğumuz “Input Validation Issues > Part 2” sayfasını sunan / kontrol eden InputValidation2URISchemaActivity.class java dosyası içeriğini

incelediğimizde get() isimli (yani Türkçe ifadeyle elde et / getir isimli) bir metot kullanımı görünecektir.

InputValidation2URISchemeActivity.class:

```
public void get (View paramView)
{
        EditText localEditText = (EditText)findViewById(2131492993);

((WebView)findViewById(2131492995)).loadURL(localEditText.getText().toString());
}
```

Metot içerisi incelendiğinde uygulama sayfasındaki metin kutusu nesne halinde, içerdiği verilerle beraber localEditText'e atanmaktadır. Sonra WebView nesnesinin loadURL metodu ile metin kutusundaki adresten dosya dahil edilmektedir ve içerik yüklemesi arayüze yapılmaktadır.

Bu uygulama sayfasını sunan / kontrol eden java kaynak kodundaki kullanıcı girdisini filtreleme / denetleme eksikliği barındıran satır dosya dahil etme / yükleme satırındadır:

```
>>> Girdi Denetleme / Kontrol Eksikliği

((WebView)findViewById(2131492995)).loadURL(localEditText.getText().toString());
```

Kaynak koddaki metin kutusundaki adres verisi filtrelenmeden / denetlenmeden dosya dahil etme / yükleme metoduna (loadURL()'e) verildiğinden uzak adres yerine yerel adres girerek yereldeki dosyaları dahil edebilmekteyiz.

Sonuç

Girdi kontrolü yapılması her uygulama için zaruridir. Aksi takdirde birçok açıklığa meydan verilebilir. Bu derste girdi kontrolü yapılmadığı için normalde sayfa içerik sunmak içi uzak adres beklerken yerel adres girilebilmiştir ve cihazın içerisinde gömülü bir hassas dosya okunabilmiştir.

Kaynaklar

https://blog.secureideas.com/2014/10/sqlite-good-bad-embedded-database.html
https://www.sqlite.org/whentouse.html
https://stackoverflow.com/questions/52201959/can-another-person-access-my-sqlite-data
https://en.wikipedia.org/wiki/Embedded_database
https://en.wikipedia.org/wiki/SQLite
https://dba.stackexchange.com/questions/21/is-it-possible-to-use-sqlite-as-a-client-server-database
https://stackoverflow.com/questions/31695766/no-credentials-needed-for-sqlite-database
https://dev.to/lefebvre/sqlite-is-not-a-server-56il