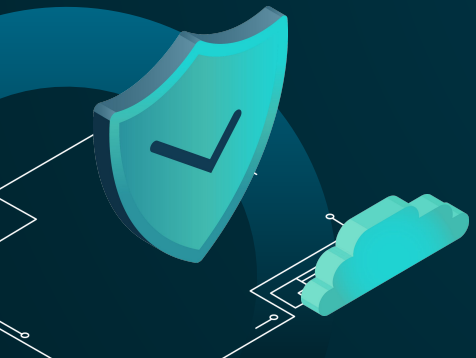
Firestick Under Fire

Exploiting Weaknesses to Access Sensitive Data



Presented by:

Krushonta Prude-Turner, Leondra Rascoe, Patrick Schultz, Gertrise Thomas, Shantel Varner, Tasheria Walls

Technical Background

(Jail)Breaking Boundaries:

- What is an APK file?
  - Android Package Kit a compressed archive containing all the necessary components for installing and executing an android app.
- By leveraging an APK file, we successfully exploited vulnerabilities within the Firestick system.
- While we have previously modified permissions in a controlled environment, this project allowed us to extend those capabilities to a broader range of file types and system systems.

The Burn Kit

- Malware Development & Analysis/ Penetration Testing Methodologies
  - Development & Staging
    - T1587.006 - Development Capabilities: Malware
    - T1608.001 - Stage Capabilities: Upload Malware
- Mobile Security & Sideloading Risks
  - Initial Access
    - T1476 - Deliver Malicious App
    - T1429 - Execution via Sideloading
- Data Collection & Exfiltration
  - Collection
    - T1412 - Capture Device Data
- Social Engineering
  - Credential Access
    - T1566.001 - Modify Authentication Process

Fueling the Fire: Our Research

- To exploit our Firestick, we first identified the device's operating system version. This allowed us to use ChatGPT to research and focus on the known vulnerabilities specific to our version. We targeted Firestick OS 8.1.3.3 for exploitation.
- ChatGPT provided information on enabling the developer option on the Firestick, which explained how ADB (Android Debug Bridge) grants command-line access to the device's OS, allowing for file transfers, APK (Android Package Kit) app installation, and other advanced operations.
- White Rabbit Neo was used to generate a script to extract personal information from our Firestick and transfer it to a designated download directory. The White Rabbit Neo script we received required modifications due to an error, which we resolved with the help of ChatGPT.

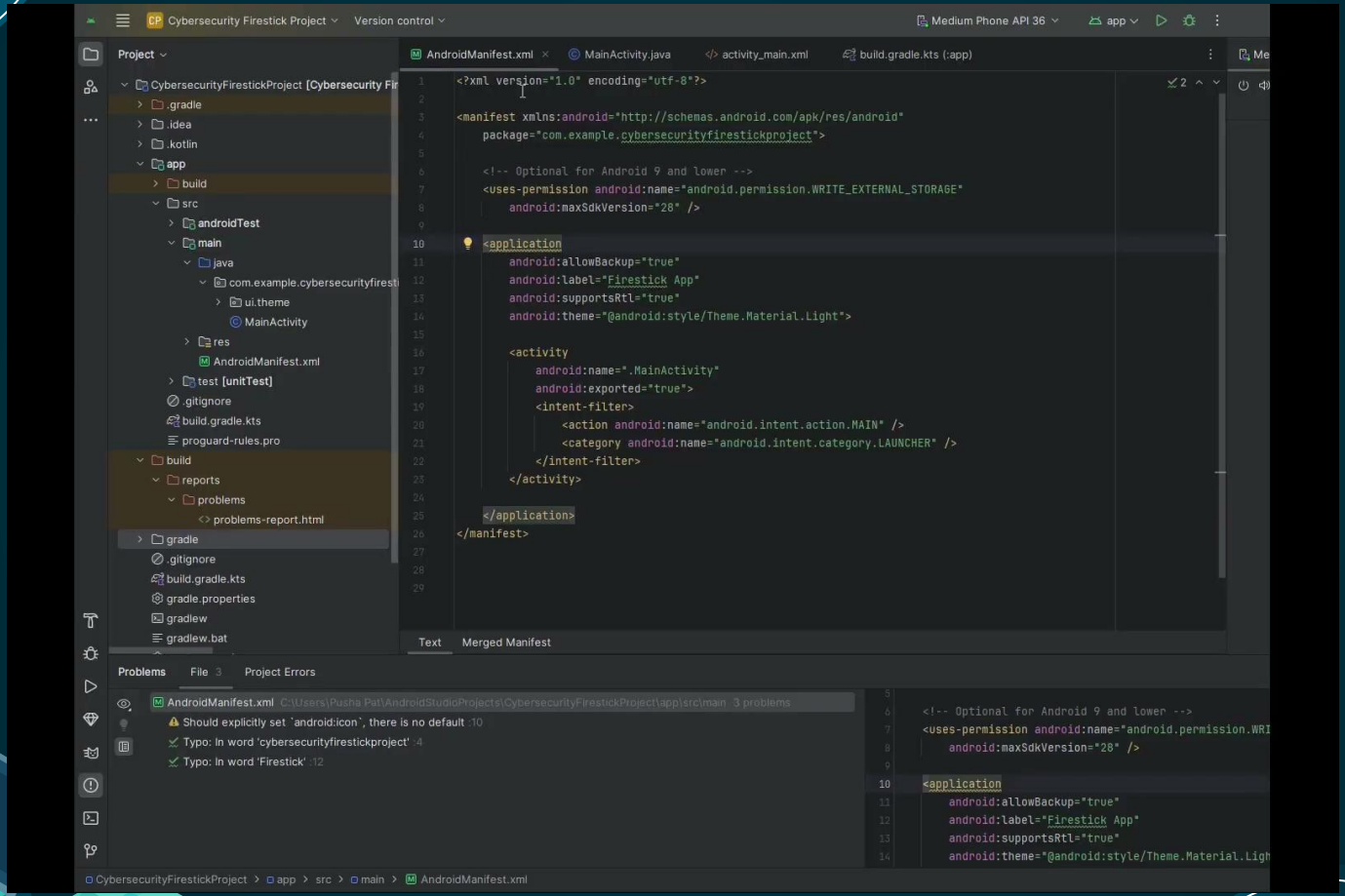
Demonstration Preview

The Burn Path:

The main goal of this exploiting process is to trick the user into giving us his sensitive credit card information.

- First we downloaded ES File Explorer and the X-plore app from the Amazon store.
- We used the White Rabbit Neo application to locate Firestick vulnerabilities.
    - This tool helps with both offensive and defensive cybersecurity tasks. It can create code for detecting weaknesses, testing security, analyzing threats, and protecting against cyber-attacks.
- We enabled the developer options within the Firestick's settings which allowed us to activate ADB debugging to install apps (the APK file) not located in the app store.
- Next we worked on generating our code using White Rabbit Neo, and ChatGPT to help fix errors.
- We generated our APK file and moved it to the firestick using ES File Explorer.
    - Once the APK is installed and opened, there is a pop-up for the user to input his credit card info for a one time fee to gain access to numerous apps for pirating movies and TV shows.
- Once Jim enters his credit card info, the information is copied to the Firestick's download folder where we then use the FTP server option in the X-plore app to copy the information to our computer.

Demonstration


Cybersecurity Firestick Project
Version control
Medium Phone API 36
app
Project
CybersecurityFirestickProject [Cybersecurity Fir
.gradle
.idea
.kotlin
app
build
src
androidTest
main
java
com.example.cybersecurityfirest
ui.theme
MainActivity
res
AndroidManifest.xml
test [unitTest]
.gitignore
build.gradle.kts
proguard-rules.pro
build
reports
problems
problems-report.html
gradle
.gitignore
build.gradle.kts
gradle.properties
gradlew
gradlew.bat
AndroidManifest.xml
MainActivity.java
activity_main.xml
build.gradle.kts (:app)
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
package="com.example.cybersecurityfirestickproject">
<!-- Optional for Android 9 and lower -->
<uses-permission android:name="android.permission.WRITE_EXTERNAL_STORAGE"
android:maxSdkVersion="28" />
<application
android:allowBackup="true"
android:label="Firestick App"
android:supportsRtl="true"
android:theme="@android:style/Theme.Material.Light">
<activity
android:name=".MainActivity"
android:exported="true">
<intent-filter>
<action android:name="android.intent.action.MAIN" />
<category android:name="android.intent.category.LAUNCHER" />
</intent-filter>
</activity>
</application>
</manifest>
Text
Merged Manifest
Problems
File 3
Project Errors
AndroidManifest.xml C:\Users\Pusha Pat\AndroidStudioProjects\CybersecurityFirestickProject\app\src\main 3 problems
Should explicitly set `android:icon`, there is no default :10
Typo: In word 'cybersecurityfirestickproject' :4
Typo: In word 'Firestick' :12
CybersecurityFirestickProject
app
src
main
AndroidManifest.xml

Demonstration Summary

The Afterburn:

A Firestick can provide a real-world environment to practice penetration testing, vulnerability scanning, and vulnerability exploitation techniques.. Even though it seems harmless, it's crucial to recognize the potential hazards involved.

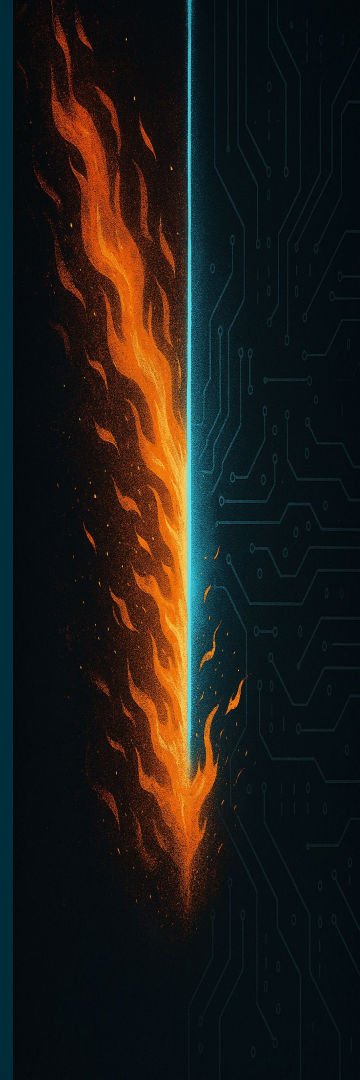
- Unauthorized modifications to the Firestick software, including bypassing security features,violating Amazon's Terms of Service. This can lead to account suspension, device bans, or loss of warranty.
- Experimenting with exploits can lead to data corruption or even "bricking" your device. You might lose personal data, settings, or even render the Firestick unusable.
- If your Firestick is connected to your home network, poorly designed exploits or network scanning tools could inadvertently expose other devices on your network.
- Even in a controlled environment, you might accidentally expose sensitive data during your testing. For example, log files or network traffic captures could contain personal information.
- If the exploitation is used to access paid content for free, such as bypassing Digital Rights Management (DRM), it could violate copyright laws under the Digital Millennium Copyright Act (DMCA).
- Exploits can cause system instability, app crashes, or unexpected behavior. This can make your Firestick unreliable for regular use.

To perform ethical Firestick penetration testing, ensure that you can secure proper permissions and operate within legal boundaries. Unauthorized exploitation carries substantial legal and financial risks.

Mitigation

Stopping The Spread:

- Purchase a Firestick from a trusted retailer.
- Enable multi-factor authentication for added security.
- If buying from a third party, perform a factory reset before use.
- Keep ADB debugging disabled to prevent unauthorized access.
- Avoid downloading apps from unverified sources.
- Connect your Firestick only to trusted, encrypted Wi-Fi networks
- Check for unknown devices connected to your Firestick under:
  Settings > My Fire TV > About > Network
- If you log into apps (like Netflix, Hulu, etc.), periodically log out to minimize stored session data.
- If you suspect unauthorized access, reset the Firestick to factory settings:
  Settings > My Fire TV > Reset to Factory Defaults



Thank you


WE MADE IT