**Project Description and Goals**

*Provide a description of your research proposal, written in terms easy to understand for someone outside your field. Include your project’s overall objectives as well as specific goals you plan to accomplish this semester.*

Apple’s iOS is one of the most popular mobile operating systems, driving over a billion devices worldwide. The Apple App Store is the only legitimate means for millions of people to install their favorite applications on their iOS devices. Users can download applications directly to their phones or use iTunes, a desktop application for syncing music, apps, and other information with iOS devices. Unlike competing mobile operating systems such as Android, iOS is considerably more locked down.

While there are analyses of iOS applications, there is little research on the interaction between iTunes and iOS beyond the limited documentation offered by Apple. This project evaluates the extent to which iTunes and iOS verify the integrity of downloaded applications. To install an application on iOS using desktop iTunes, the application is first downloaded to the computer from Apple servers. The application is downloaded in a format called IPA, which is a compressed archive containing code and resources such as images or multimedia content. iTunes then embeds metadata into the IPA, and then transfers the IPA to the iOS device. The goal of our assessment is to identify the extent to which the IPA can be modified and still recognized and transferred by iTunes. To this end, it is necessary to investigate whether images or other assets can be changed without triggering integrity protection.

In addition to investigating iTunes’ integrity protection, we are interested in developing a system to automatically download and analyze iOS applications for certain characteristics. While a similar project called *Playdrone* was developed for Android, no such attempts have been made on iOS because of the difficulty in circumventing Apple’s IPA encryption. The advantage of developing an automatic analysis tool is the ability to analyze many different applications at once in a short amount of time. If we can check hundreds of applications for a certain characteristic, we can discover trends that might have been overlooked by the manual analysis applied during Apple's vetting process.

**Project Significance**

*Explain in general terms why the information gained from this research project will be beneficial.*

By analyzing the iTunes’ integrity check for IPA files, we could identify potential security holes, where a malicious attacker could modify his Apple-approved application to look like a bank application, install it on a device, and capture user credentials. Using our automatic analysis tool, we can spot certain trends in applications. For example, we can check which applications are duplicates of each other, either through sharing the same graphical user interface or the same codebase; duplicates raise a red flag because of the possibility that they are fakes created to steal passwords from the user. We can also calculate the percentage of applications that use social media libraries (like Facebook, Twitter) for integration for analytical purposes. Most importantly, the quick speed of automatic analysis allows us to rerun the tool, checking for as many different characteristics as desired. The knowledge gained from these analyses could be used as the basis for further investigation into Apple’s application security.

**Methodology**

*Clearly state how you plan to accomplish the goals listed in your project description. Identify the specific steps necessary to perform your research.*

Using an open source tool called *mitmproxy*, we can analyze network traffic as iTunes sends a request to Apple servers to download an application IPA. Not only can the web request be examined, it is also possible to intercept an IPA and dump it for analysis. This tool allows us to swap the application IPA that iTunes requested with one that we modified. Using these techniques, we can start by modifying filenames in known IPAs to check how iTunes responds to changes. Depending on the former, we can make larger changes, such as substituting images and other resources while checking to see whether iTunes accepts the IPA. Therefore, we can verify the extent to which iTunes checks for integrity.

For the analysis of applications, we will leverage both dynamic runtime analysis and static analysis. Under dynamic analysis, we will use a jailbroken iOS device (a device with the ability to install unsigned applications) to execute the applications, then use a debugger or internal iOS tools to collect information while the applications are running on the device. We will use dynamic analysis to explore detailed characteristics, such as network activity, of a small set of applications. For static analysis, we will examine the IPA file on a computer using disassembling tools like IDA Pro or the Hopper framework to identify libraries, shared code, and other interesting characteristics in a large set of applications.

**Time Line**

*Provide a detailed timeline of goals of your project. Be sure to include the goals listed above.*

For the first week, we will try intercepting an IPA download as it occurs, making small changes, and feeding our intercepted version back to iTunes. Depending on how iTunes reacts to these changes (e.g., whether it states that downloaded IPA is corrupted), we can perform larger changes, such as replacing images. Depending on the outcome of this experiment, we can draw conclusions and move onto our analysis tool. To start developing our tool, we will manually analyze applications from the CRIOS dataset. Depending on our choices for characteristics to look for, we will switch between runtime and static analysis to find which works best. If we choose static analysis, we will need to circumvent Apple’s encryption to disassemble the code; this process can be time consuming. Once we have access to a disassembled version of the code, we can start looking for characteristics of those applications (e.g., what kind of libraries it uses). Following our manual analysis, we will proceed to write a script for IDA Pro’s built-in scripting language to find characteristics like shared code and common libraries. In addition to automatic analysis, we will also create an application to automatically download applications from the App Store for our IDA Pro script. Once we have a working prototype of the tool, we can expand its capabilities to explore different trends.

**Background Experience**

*List any previous research experience, applicable course work, or other relevant experience you may have.*

In order to prepare myself for this research, I have taken EC440: Introduction to Operating Systems to gain insight into the inner workings of applications and operating systems. I have also taken EC327: Introduction to Software Engineering, where my knowledge of C/C++ was greatly expanded. In addition to my classes, I have done some programming projects of my own, such as *PE Section*, a library for creating and editing file headers in the (Windows) portable executable format. I have also done some work with reverse engineering eventually creating *.NET Obfuscator*, a tool for programmers to protect their software by making decompiling more difficult.

**Bibliography**

*List the sources you have consulted in preparation of this proposal, as well as any references you have cited within this application. If you need additional space, use the Appendix.*

<http://megele.io/crios.pdf>

http://seclab.nu/static/publications/ndss2011pios.pdf

https://www.cs.purdue.edu/homes/bsaltafo/pubs/CCS\_15\_iRiS.pdf