Project Report

Aaron Greenberg, Allie Duncan, Cypress Frankenfeld, Geoff Pleiss

Our goal is to better understand cyber security by implementing our own worm. Ideally, this worm will be for an documented exploit of a current or old mobile OS. We choose to work with mobile devices because mobile cyber security is a new field. This worm will contain a payload to demonstrate that it has infected a device, and potentially will install a backdoor to root the device. Through this project, we aim to understand malware propagation, firewalls, and network security.

Plan B

If, for some unforeseen reason, we are unable to complete this goal, our secondary project would be to build a compiler Trusting Trust compiler. Such a compiler would be based on a simple compiler found online. Then, from our understanding of compilers and computer security, we would install a backdoor. This is a good fail-safe project because the compiler's backdoor can be as simple or complicated as time and resources allow.

Techniques

Research and implementation.

Minimum Deliverable

The minimum deliverable will be a worm that spreads across devices (computers or mobile phones). It will be able to replicate itself and spread to other devices (possible via email, network, etc.), and will carry a simple payload.

Maximum Deliverable

The maximum deliverable will be a worm that spreads across *mobile* devices. It will create a backdoor in the device that will give a remote user root access to execute programs. Ideally, this worm will be spread in a discrete mobile-specific manner (e.g. via bluetooth).

Starting up

Our first step was to begin researching the design and implementation of famous computer worms. This was to lay the foundation of knowledge to be able to understand how we want to create a worm for mobile devices. We will also need to research security of mobile devices to figure out how to discover vulnerabilities that can be targeted by the worm we build.

Research

For the research phase of our project, we searched for useful sources on the web. We have documents that analyze the functionality of the Morris worm, SQL Slammer worm, Cabir worm and several others. Additionally, we have a list of some well-known vulnerabilities on multiple operating systems.

Additionally, we have reached out to three Olin resources with knowledge about computer security and exploits: Mark Chang and Noah Tye, who co-taught an independent study in computer security and both have a solid working knowledge of mobile development; and Kevin Mehall, an Olin student who outperformed everyone at the last MIT Capture the Flag competition.

Kevin Mehall gave a number of suggestions about how to choose an exploit---for the worm to self-propagate, we will need to make use of a “remotely exploitable code execution vulnerability.” He recommends the CVE database for its collection of patches, and then discussed different platforms for remote exploits: OS/kernel level (tricky and difficult to debug), code that runs as a server, or Wordpress (written in PHP, less “involved” exploit). He also gave recommendations about setting up a safe environment in which to test our code.

Noah Tye recommends treating the mobile device as a Unix machine rather than a cell phone to avoid “too much of an additional hassle.” He also pointed to two resources: a set of slides about Android security from a talk given at DEFCON 18 and a description of a “purported one-click root” for Android devices.

Mark Chang suggests that our best chance is to read the XDA forums on Android device rooting work (usually via a kernel exploit) done by XDA developers. He then wished us luck and gave us his blessing:

> *“These are hard to do and nearly impossible on current devices to accomplish remotely.”*

Preliminary Test

It is possible for a mobile device to find another mobile device through the private IP address space used by a Wi-Fi network. To test this, we used `ping` (a network utility command) to contact a several mobile devices from our computer. Both android and iphone devices were `ping`-able.

Challenges

The main challenges we foresee are:

- Finding a vulnerability on a mobile device that we can exploit to gain root permissions

    - We have contacted Mark Chang (our resident mobile expert) to enlist help in finding this.
    - It may be easier to find exploits in older OSs, which have more known problems.
- Spreading a worm safely without it contaminating devices other than our own
    - To make sure that we are not running a risk of unintentionally infecting devices, we will need to enlist the help of Allen or IT in setting up a safe working environment.

Future Work

Going forward, our plan is to do research into specific exploits that can be used effectively for our project. We would ideally like to choose 2 to 4 well-documented exploits and begin writing code to probe the vulnerabilities. We will attempt to start by using a mobile emulator to determine whether the exploit works, and then transition to actual, borrowed devices for test the worm’s propagation capabilities. We will begin to research specific vulnerabilities on Thursday, November 8 and will hopefully begin writing some code to exploit them by the end of the week.