Extra Credit Homework
CS 642: Information Security

April 30, 2014

This homework assignment is for extra credit. You *may* NOT work with a partner. It is due **May 15, 2014** by midnight local time. Big thanks to Heming Shou for putting together the materials for this homework!

1 Project Introduction

Heartbleed bug, an OpenSSL encryption flaw, was made public in April 2014. The bug had huge consequence, the vulnerable software being widely used among website servers and put millions of servers in danger of information leaks. Heartbleed allows hackers to remotely retrieve swathes of process memory from the openssl. Although the retrieved chunk of memory is only up to 64 KB for each time, it can be exploited repeatedly to attempt to obtain different memory chunks. The attacker exploits this by sending malformed heartbeat requests to the server.

2 Environment Setup

For this project, you are going to use a VirtualBox virtual machine for development and test. Set up VM on your own machine: Download the Heartbleed virtual machine tarball, heartbleed.tar.bz2 (approximately 1GB!). Decompress this tarball and import it using VirtualBox, as you did for the previous projects. This virtual machine contains an Ubuntu 12.04(LTS) system with a vulnerable openssl version. Once the HeartBleed VM is running, you can login to the ubuntu unity desktop using user "user". There are two users in the system: "root" with password "root" and "user" with password "user". You cannot simply login as a "root", so it is suggested to do all the root actions using "sudo". There is an apache server running under the VM using address localhost (127.0.0.1) and its SSL connection (port 443) is open. If you have trouble running the VM image, email the TA for support.

3 Project Requirements

In this project, you need to finish the following tasks:

1. (Description) First you should read up on the Heartbleed bug and what happens in the bug source code of OpenSSL. Provide a brief description (at most a few paragraphs) clearly explaining the vulnerability, including where it arises in the OpenSSL code base. Describe how to fix the bug.

2. (Detection) Write a script (Python/bash) or program (C/Java) to detect whether a system is vulnerable to Heartbleed. This program should be designed to do *no* harm against the target system. This means that the script can confirm the bug is exploitable, but ensure minimize the amount of information extracted from the target system. The program/script should take as input on the command line an IP address and output a short message indicating vulnerability.

3. (Exploitation) Write a script (Python/bash) or program (C/Java) to extract memory contents from the local vulnerable web server. You must dump at least 64 KB each time. The output format should be a hex dump of the 64 KB buffer. Optionally you might have the script/program scan the data programmatically for potentially sensitive information, such as portions of HTTP GET messages, and output this data with a description of what type of data it is.

The deliverables should be a single TXT or PDF file with the writeup for part 1, as well as short explanations of your approaches for parts 2 and 3. The TA will grade by reading your code and supplemental file as well as running your code against the vulnerable server a couple of times. You can test your own code by dumping large amount of memory for running it lots of times and check there is some common string like pieces of HTTP GET results or other common strings.

4 Supplemental Materials

Here are some materials for Heartbleed:

- http://heartbleed.com/
- http://en.wikipedia.org/wiki/Heartbleed
- http://arstechnica.com/security/2014/04/how-i-used-heartbleed-to-steal-a-sites-private-crypto-key/

You may use exploit code you find on the Internet to help understand the vulnerability, but do *not* plagiarize exploit code or scripts. If in doubt ask the TA or instructor for permission before using any source you are unsure about.

There are still many vulnerable servers on the Internet. Do not run your exploit code against any server other than the local VM or other systems that you own. Running it against another user's system could be considered a crime.

5 Grading

Each part is worth up to 2 points. If the code works and the description is comprehensive, then one will receive full credit. Partial credit will be given for a good description of the vulnerability when the code fails to run.