CS3612-2 Lab4

> In this lab you will select one project from the following two topics and finish it **by yourself.** Our available topics are networking and virus.

Overview

(choose 1 from the following 3 projects)

- Networking
  - Socket programming: chat server
- Security
  - CTF
  - CVE exploitation

Topic 1. Networking: A Chat Server

Target

- Learn basic knowledge about socket programming and async programming.
- Learn essential skills in the usage of boost asio library.
- Learn basic operations of linking dynamic/static libraries.
- Learn basic skills in handling network traffic problems.

Requirement

In this topic, you should use boost library or derived boost.beast library to build a socket-level robust peer-to-peer chat server.

1. Code Implementation (70%)
   - chat functionality (40%)

     Implement basic chat functionality by using boost library, instead of known socket syscalls.

     Feel free to use the `boost` library directly or use the `boost.beast`.

     Feel free to use multi-threading or co-routine to implement your server! However, sync-io is not permitted in this lab.
   - security protection (30%)

     Implement some basic DDoS protection methods on your server.

     Feel free to attack your classmates' poor servers (networking-level only)!
2. Code Review (30%)
   - A pdf-format report. (optional, just to understand your project design better)
   - Show me your server design.
   - Prove your code's functionality.
   - Prove your code's reliability.

In your demo, **feel free to use network analyzer tools** and any other third-party tools to measure your code reliability!

Bonus

1. More advanced chat support.
    - Support size-limited chat groups.
    - Support message synchronization among different users' views. (2 or more)
    - ...
2. More advanced robustness protection.
    - Provide confidentiality to your chat channel, instead of using a plain-text channel.
    - Build a distributed server system to defend against attacks, and send back attack packages to the attacker.
    - ...

Misc

Some maybe-useful links and references.

- Overview of Boost lib
- Examples provide by boost 1.78.0 in cpp11 standard.

Submit

C++ files support client and server functionality.

Topic 2. Security 1: A try to CTF (capture the flag)

Have a guess!

Target

- Learn basic reverse engineering operations.
- Learn to hijack control flow by using stackoverflow (not stack**overflow...)**

Requirement

1. Code Implementation (70%)

   Login successfully in the `guess` binary file, and launch a bash in the guess prorgam successfuly.

2. Code Review (30%)

   Explain your hack idea.

Bonus

none.

Misc

Some recommended tools.

- IDA

  Generate pseudo codes for analyze guess code.

- pwn packages

```
pip3 install pwn
```

Submit

A hack program file.

(Frankly, it's nearly impossble for you to login the `guess` without a hack program.)

Topic 2. Security 2: A try to reproduce CVE-2017-16995

Pwn!

Target

- Learn to build and run a Linux kernel.
- Obtain essential skills in the usage of gdb/lldb.
- Obtain essential skills in ebpf programming.
- Be familiar with ebpf architecture and execution sequence.
- Learn the BPF instruction set.
- Be familiar with the Linux kernel's process management structures.

Requirement

In this project, you are encouraged to **reproduce** the famous **CVE-2017-16995** kernel vulnerability to pwn the Linux 4.4.0 kernel in Ubuntu 16.04, **bypass the ebpf verifier**, and **achieve privilege escalation** by writing into the process's `cred` structure.

1. Code Implementation (80%)
   - Your bpf virus bypasses the bpf verifier. (40%)
   - Your bpf virus escalates its into a privileged process and launches a shell. (40%)
2. Code Review (20%)
   - Explain your code execution logic.
   - Answer reviewer's questions in the program details.
   - ...

Bonus

None.

Misc

1. A detailed report about **CVE-2017-16995**
2. Available Linux kernel versions for exploiting this vulnerability (Ubuntu 16.04):

   4.4.0-62-generic

   4.4.0-62-generic

   4.4.0-81-generic

   4.4.0-116-generic

   4.8.0-58-generic

   4.10.0.42-generic

   4.13.0-21-generic
3. EBPF instruction set documentation.
4. Other discussions on this CVE

Feel free to search for any other thing you need on the Internet!

Submit

A bug-exploitation program file. Designate your kernel version, and reproduce the bug by your program in your code review.

> This page is written by Ganxiang Yang @ Apr 11, 2023.