

## BÁO CÁO BÀI TẬP

Môn học: Lập trình an toàn và khai thác lỗ hổng phần mềm

Tên chủ đề: Exercise 5

GVHD: Phan Thế Duy

▪ **THÔNG TIN CHUNG:**

(Liệt kê tất cả các thành viên trong nhóm)

Lớp: ANTN2020

STT	Họ và tên	MSSV	Email
1	Nguyễn Bùi Kim Ngân	20520648	20520648@gm.uit.edu.vn
2	Nguyễn Bình Thục Trâm	20520815	20520815@gm.uit.edu.vn
3	Võ Anh Kiệt	20520605	20520605@gm.uit.edu.vn

▪ **NỘI DUNG THỰC HIỆN:<sup>1</sup>**

STT	Công việc	Kết quả tự đánh giá
1	Yêu cầu a	100%
2	Yêu cầu b	100%

Phần bên dưới của báo cáo này là tài liệu báo cáo chi tiết của nhóm thực hiện.

---

<sup>1</sup> Ghi nội dung công việc, các kịch bản trong bài Thực hành

# BÁO CÁO CHI TIẾT

## Cấu hình máy

- + Kali Linux 2022.3
- + 8GB Ram
- + 80GB HDD
- + 2 core 2 thread CPU intel core i5-8250U

## Cài đặt

Đầu tiên ta sẽ cài đặt bộ công cụ fuzzing bằng lệnh:

`sudo apt install kali-tools-fuzzing`

Bộ công cụ sẽ bao gồm công cụ

- + AFL++
- + sfuzz
- + wfuzz

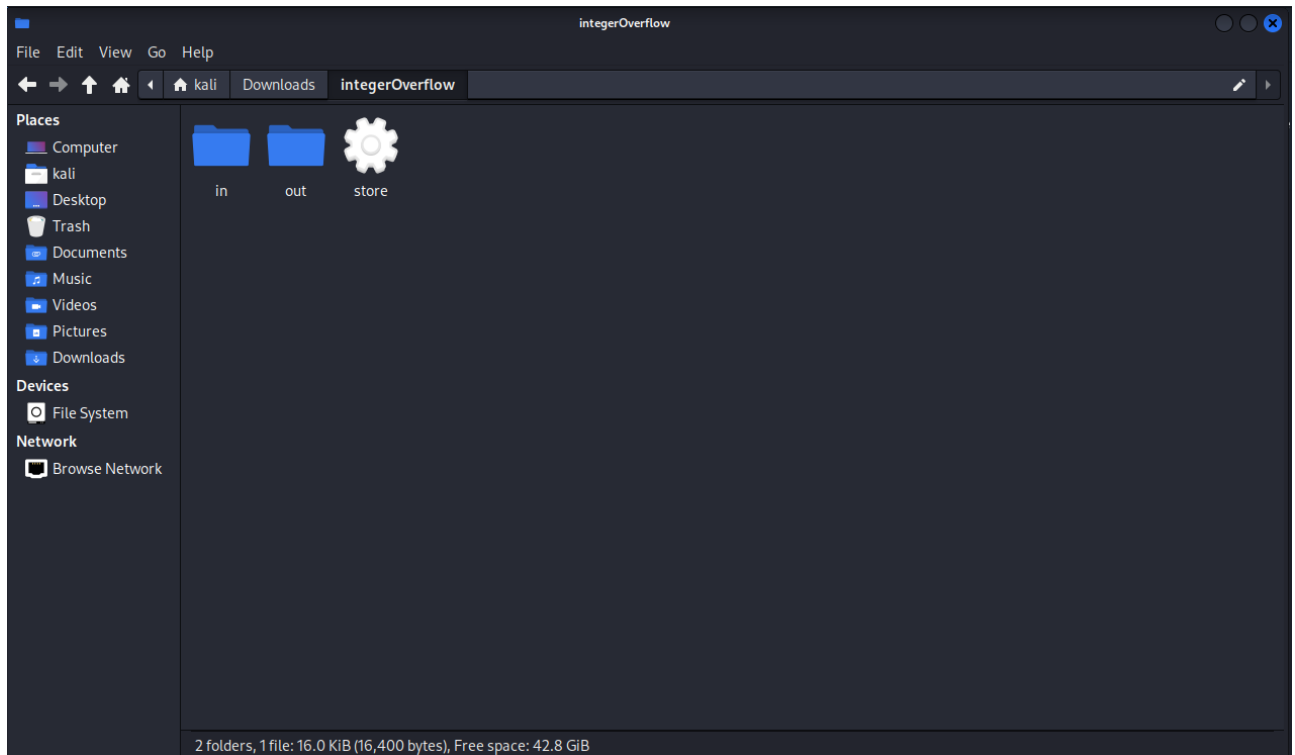
```

[sudo] password for kali:
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following packages were automatically installed and are no longer required:
  binutils-mingw-w64-1606 binutils-mingw-w64-x86-64 catfish fonts-ubuntu-color-emoji gcc-mingw-w64-base gcc-mingw-w64-1606-win32 gcc-mingw-w64-1606-win32-runtime gcc-mingw-w64-x86-64-win32 gcc-
  girl1.2-xfconf-0 libasio mingw-w64-common mingw-w64-1606-dev mingw-w64-x86-64-dev oracle-instantclient-basic python3-dataclasses json python3-lister python3-marshmallow-enums python3-mypy-
  python3-token-bucket python3-typing-inspect
Use 'sudo apt autoremove' to remove them.
The following additional packages will be installed:
  afl++ afl++-doc docbook-xsl fonts-dejavu girl1.2-javascriptcoregtk-4.0 girl1.2-webkit2-4.0 gnome-system-tools kali-desktop-core kali-desktop-xfce kali-linux-core kali-linux-firmware kali-tool-
  libjavascriptcoregtk-4.1-0 libboots-1.5 libwebkit2gtk-4.0-37 libwebkit2gtk-4.1-0 libyelp0 sgml-data system-tools-backends yelp yelp-xsl
Suggested packages:
  gnuplot docbook-dsssl docbook-xsl docbook-defguide ntp kali-root-login gstreamer1.0-alsa perlsgml w3-recs opensp
The following NEW packages will be installed:
  afl++ afl++-doc docbook-xsl fonts-dejavu gnome-system-tools kali-tools-fuzzing libboots-1.5 libyelp0 sgml-data system-tools-backends yelp yelp-xsl
The following packages will be upgraded:
  girl1.2-javascriptcoregtk-4.0 girl1.2-webkit2-4.0 kali-desktop-core kali-desktop-xfce kali-linux-core kali-linux-firmware kali-tools-top10 libjavascriptcoregtk-4.1-0 libjavascriptcoregtk-4.1-
  11 upgraded, 12 newly installed, 0 to remove and 1544 not upgraded.
Need to get 55.3 MB of archives.
After this operation, 26.6 MB of additional disk space will be used.
Do you want to continue? [Y/n] y
Get:1 http://http.kali.org/kali kali-rolling/main amd64 afl++ amd64 4.04c-1 [481 kB]
Get:2 http://http.kali.org/kali kali-rolling/main amd64 afl++-doc all 4.04c-1 [285 kB]
Get:3 http://http.kali.org/kali kali-rolling/main amd64 sgml-data all 2.0-11-mmd [179 kB]
Get:4 http://http.kali.org/kali kali-rolling/main amd64 docbook-xsl all 4.5-12 [85.2 kB]
Get:5 http://http.kali.org/kali kali-rolling/main amd64 fonts-dejavu all 2.37-2 [32.6 kB]
Get:6 http://http.kali.org/kali kali-rolling/main amd64 girl1.2-webkit2-4.0 amd64 2.38.2-1+b1 [93.4 kB]
Get:7 http://http.kali.org/kali kali-rolling/main amd64 girl1.2-javascriptcoregtk-4.0 amd64 2.38.2-1+b1 [39.3 kB]
Get:8 http://http.kali.org/kali kali-rolling/main amd64 libwebkit2gtk-4.0-37 amd64 2.38.2-1+b1 [17.7 MB]
Get:9 http://http.kali.org/kali kali-rolling/main amd64 libjavascriptcoregtk-4.0-18 amd64 2.38.2-1+b1 [6,498 kB]
Get:10 http://http.kali.org/kali kali-rolling/main amd64 system-tools-backends amd64 2.18.2-3.1 [176 kB]
Get:11 http://http.kali.org/kali kali-rolling/main amd64 libboots-1.5 amd64 3.0.0-4+b2 [75.2 kB]
Get:12 http://http.kali.org/kali kali-rolling/main amd64 gnome-system-tools amd64 3.0.0-9.1 [4,139 kB]
Get:13 http://http.kali.org/kali kali-rolling/main amd64 kali-desktop-core amd64 2023.1.1 [10.5 kB]
Get:14 http://http.kali.org/kali kali-rolling/main amd64 kali-desktop-xfce all 2023.1.1 [10.7 kB]
Get:15 http://http.kali.org/kali kali-rolling/main amd64 kali-linux-core amd64 2023.1.1 [10.4 kB]
Get:16 http://http.kali.org/kali kali-rolling/main amd64 kali-linux-firmware amd64 2023.1.1 [10.8 kB]
Get:17 http://http.kali.org/kali kali-rolling/main amd64 kali-tools-fuzzing amd64 2023.1.1 [10.4 kB]
Get:18 http://http.kali.org/kali kali-rolling/main amd64 kali-tools-top10 amd64 2023.1.1 [10.2 kB]
Get:19 http://http.kali.org/kali kali-rolling/main amd64 libwebkit2gtk-4.1-0 amd64 2.38.2-1+b1 [17.7 MB]
Get:20 http://http.kali.org/kali kali-rolling/main amd64 libjavascriptcoregtk-4.1-0 amd64 2.38.2-1+b1 [6,498 kB]
Get:21 http://http.kali.org/kali kali-rolling/main amd64 libyelp0 amd64 4.2.2-1 [169 kB]
  
```

## Yêu cầu A

Nguồn: play.picoctf.org

Tạo folder input và output



Tạo các testcase

```
(kali@kali) - [~/Downloads/integerOverflow]
$ echo -en "1234567890\x00" > ./in/1.testcase
```

Thực hiện fuzzing (single thread)

The screenshot shows the American Fuzzy Lop (AFL) fuzzer interface in a terminal window. The interface is divided into several sections:

- process timing:** run time: 0 days, 0 hrs, 0 min, 15 sec; last new find: none seen yet; last saved crash: none seen yet; last saved hang: 0 days, 0 hrs, 0 min, 12 sec.
- overall results:** cycles done: 0; corpus count: 1; saved crashes: 0; saved hangs: 2.
- cycle progress:** new processing: 0.2 (0.0%); runs timed out: 0 (0.0%); stage progress: new trying: havoc; stage execs: 584/587 (99.49%); total execs: 1311; exec speed: 86.61/sec (slow); map coverage: map density: 0.00% / 0.00%; count coverage: 1.00 bits/tuple; findings in depth: new edges on: 1 (100.00%); total crashes: 0 (0 saved); total timeouts: 153 (0 saved).
- stage progress:** new trying: havoc; stage execs: 584/587 (99.49%); total execs: 1311; exec speed: 86.61/sec (slow); map coverage: map density: 0.00% / 0.00%; count coverage: 1.00 bits/tuple; findings in depth: new edges on: 1 (100.00%); total crashes: 0 (0 saved); total timeouts: 153 (0 saved).
- fuzzing strategy yields:** bit flips: disabled (default, enable with -D); byte flips: disabled (default, enable with -D); arithmetic: disabled (default, enable with -D); known ints: disabled (default, enable with -D); dictionary: n/a; havoc/splice: 0/715, 0/0; pp/custom/rq: unused, unused, unused, unused; trim/eff: 63.64%/2, disabled; item geometry: levels: 1; pending: 0; pend fav: 0; own finds: 0; imported: 0; stability: 100.00%.

## Fuzzing multithread

The screenshot shows the American Fuzzy Lop (AFL) fuzzer interface in a terminal window, displaying multiple instances of the fuzzer running simultaneously. The interface is divided into several sections:

- process timing:** run time: 0 days, 0 hrs, 3 min, 18 sec; last new find: 0 days, 0 hrs, 1 min, 1 sec; last saved crash: none seen yet; last saved hang: 0 days, 0 hrs, 0 min, 50 sec.
- overall results:** cycles done: 3; corpus count: 0; saved crashes: 0; saved hangs: 6.
- cycle progress:** new processing: 1.1 (16.7%); runs timed out: 0 (0.00%); stage progress: new trying: havoc; stage execs: 16/291 (5.50%); total execs: 15.2k; exec speed: 40.41/sec (slow); map coverage: map density: 0.00% / 0.00%; count coverage: 2.29 bits/tuple; findings in depth: new edges on: 3 (50.00%); total crashes: 0 (0 saved); total timeouts: 372 (0 saved).
- stage progress:** new trying: havoc; stage execs: 16/291 (5.50%); total execs: 15.2k; exec speed: 40.41/sec (slow); map coverage: map density: 0.00% / 0.00%; count coverage: 2.29 bits/tuple; findings in depth: new edges on: 3 (50.00%); total crashes: 0 (0 saved); total timeouts: 372 (0 saved).
- fuzzing strategy yields:** bit flips: disabled (default, enable with -D); byte flips: disabled (default, enable with -D); arithmetic: disabled (default, enable with -D); known ints: disabled (default, enable with -D); dictionary: n/a; havoc/splice: 0/1791, 0/0; pp/custom/rq: unused, unused, unused, unused; trim/eff: 38.89%/3, disabled; item geometry: levels: 2; pending: 3; pend fav: 0; own finds: 4; imported: 1; stability: 100.00%.

Sau 10 tiếng fuzzing ta nhận thấy rằng là đây là lỗi integerOverflow (theo gợi ý) nên không được thể hiện trong phần crash, nên có thể thấy cách fuzzing chưa thực sự hiệu quả với file binary chứa lỗi này (sẽ hiệu quả hơn với những bài bufferOverflow, returnToLibC,...)

The screenshot displays four windows of the 'american fuzzy lop' (AFL) fuzzer running on 'IntegerOverflow'. The windows show different stages of the fuzzing process, including cycle progress, stage progress, and overall results. The overall results show 0 cycles done, 50 corpus count, 0 saved crashes, and 7 saved hangs.

```

american fuzzy lop ++4.04c {f2} (./store) [fast]
process timing: run time: 0 days, 10 hrs, 25 min, 22 sec
last new find: 0 days, 0 hrs, 9 min, 4 sec
last saved crash: none seen yet
last saved hang: 0 days, 10 hrs, 22 min, 9 sec
cycle progress: now processing: 11.58 (22.8%)
stage progress: runs timed out: 0 (0.00%)
now trying: splice 3
stage execs: 31/36 (86.11%)
total execs: 461K
total crashes: 0 (0 saved)
total hangs: 7
overall results: cycles done: 0
corpus count: 50
saved crashes: 0
saved hangs: 7
map density: 0.00% / 0.00%
count coverage: 6.29 bits/tuple
findings in depth:
  favored items: 4 (0.00%)
  new edges on: 7 (14.00%)
  total crashes: 0 (0 saved)
  total hangs: 7
  item geometry:
    levels: 2
    pending: 42
    pend fav: 0
    own finds: 2
    imported: 47
    stability: 100.00%
  fuzzing strategy yields:
    bit flips: disabled (default, enable with -D)
    byte flips: disabled (default, enable with -D)
    arithmetics: disabled (default, enable with -D)
    known ints: disabled (default, enable with -D)
    dictionary: n/a
    havoc/splice: 1/71.9K, 1/132K
    py/custom/rq: unused, unused, unused, unused
    trim/eff: 73.01K/88, disabled
  [cpu002:150K]

american fuzzy lop ++4.04c {f3} (./store) [fast]
process timing: run time: 0 days, 10 hrs, 24 min, 46 sec
last new find: 0 days, 0 hrs, 37 min, 37 sec
last saved crash: none seen yet
last saved hang: 0 days, 10 hrs, 24 min, 20 sec
cycle progress: now processing: 11.74 (22.8%)
stage progress: runs timed out: 0 (0.00%)
now trying: splice 10
stage execs: 42/55 (76.36%)
total execs: 264K
total crashes: 0 (0 saved)
total hangs: 7
overall results: cycles done: 3
corpus count: 50
saved crashes: 0
saved hangs: 7
map density: 0.00% / 0.00%
count coverage: 6.29 bits/tuple
findings in depth:
  favored items: 4 (0.00%)
  new edges on: 7 (14.00%)
  total crashes: 0 (0 saved)
  total hangs: 7
  item geometry:
    levels: 2
    pending: 42
    pend fav: 0
    own finds: 2
    imported: 47
    stability: 100.00%
  fuzzing strategy yields:
    bit flips: disabled (default, enable with -D)
    byte flips: disabled (default, enable with -D)
    arithmetics: disabled (default, enable with -D)
    known ints: disabled (default, enable with -D)
    dictionary: n/a
    havoc/splice: 1/71.9K, 1/132K
    py/custom/rq: unused, unused, unused, unused
    trim/eff: 73.01K/88, disabled
  [cpu002:150K]

american fuzzy lop ++4.04c {f4} (./store) [fast]
process timing: run time: 0 days, 10 hrs, 24 min, 34 sec
last new find: 0 days, 0 hrs, 0 min, 23 sec
last saved crash: none seen yet
last saved hang: 0 days, 10 hrs, 24 min, 20 sec
cycle progress: now processing: 1.158 (2.0%)
stage progress: runs timed out: 0 (0.00%)
now trying: splice 15
stage execs: 2/20 (10.00%)
total execs: 305K
total crashes: 0 (0 saved)
total hangs: 6
overall results: cycles done: 13
corpus count: 50
saved crashes: 0
saved hangs: 6
map density: 0.00% / 0.00%
count coverage: 6.29 bits/tuple
findings in depth:
  favored items: 5 (10.00%)
  new edges on: 7 (14.00%)
  total crashes: 0 (0 saved)
  total hangs: 6
  item geometry:
    levels: 2
    pending: 25
    pend fav: 0
    own finds: 6
    imported: 43
    stability: 100.00%
  fuzzing strategy yields:
    bit flips: disabled (default, enable with -D)
    byte flips: disabled (default, enable with -D)
    arithmetics: disabled (default, enable with -D)
    known ints: disabled (default, enable with -D)
    dictionary: n/a
    havoc/splice: 0/107K, 6/196K
    py/custom/rq: unused, unused, unused, unused
    trim/eff: 54.78K/1061, disabled
  [cpu003:125K]

```

Kiểm tra lại trong file crash (không có kết quả)

The screenshot shows the 'crashes' directory in the file manager. The directory is empty, indicating no crash files were found.

```

crashes
0 Items, Free space: 42.8 GiB
Filtered Requests: 445
Requests/sec.: 0
(./share/wfuzz/wordList/injections)

```

**Yêu cầu b**

Công cụ thực hiện: sfuzz, wfuzz

Môi trường: [https://github.com/anhkiet1227/NT208\\_Web-Programming\\_Project](https://github.com/anhkiet1227/NT208_Web-Programming_Project)

So sánh

	sfuzz	wfuzz
Cài đặt	Đều nằm trong bộ kali-tools-fuzzing	
Tính năng	Thực hiện fuzzing các yếu tố về network với các testcase có các giao thức http, imap, pop3,...	Thực hiện fuzzing các yếu tố về network với các testcase có các testcase về injection, webservice, vulns,...
Quy trình thực hiện	Giống: host service, cài đặt môi trường	
	Câu lệnh chạy: sfuzz -S 192.168.1.3 -p 3000 -T -f /usr/share/sfuzz-db/basic.http	Câu lệnh chạy: wfuzz -c -z file,/usr/share/wfuzz/wordlist/Injections/All_attack.txt -hc 404, 400 "http://192.168.1.3:3000/FUZZ"
Giải thích câu lệnh	Fuzzing với ip 192.168.1.3 port 3000, tcp, và testcase basic.http	Fuzzing với url là <a href="http://102.168.1.3:3000/FUZZ">http://102.168.1.3:3000/FUZZ</a> , testcase All_attack.txt và tắt các response 404, 400
Thời gian	6 phút 9 giây	< 1 giây
Kết quả	194 response 200 có thể khai thác Do bộ testcase rộng hơn nên có thể quét được nhiều hơn	64 response 200 có thể khai thác Do bộ testcase nhỏ hơn nên có thể quét được ít hơn

Minh chứng

Service





```
PowerShell
PowerShell 7.3.1
PS C:\Users\ACER\Desktop\WT208_Web-Programming_FrontEnd> serve -s build
UPDATE The latest version of 'serve' is 14.1.2.

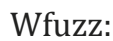
Serving!
- Local:      http://localhost:3000
- On Your Network: http://192.168.1.3:3000

Copied local address to clipboard!
```

## Sfuzz:

```
kali-linux-2022.3-vmware-amd64 - VMware Workstation
File Edit View VM Tabs Help
Library
Type here to search
My Computer
kali-linux-2022.3-vmware-amd64
Metasploitable2-Linux
Ubuntu20
Ubuntu22
Windows 10 x64
Ubuntu18

File Actions Edit View Help
kali@kali:~/Downloads/re
$ sfuzz -s 192.168.1.3 -p 3000 -t -f /usr/share/sfuzz-db/basic.http
[04:57:16] dumping options:
  filename: </usr/share/sfuzz-db/basic.http>
  state: <0>
  lineno: <0>
  literals: [74]
  sequences: [34]
  symbols: [8]
  req_del: <200>
  max_len: <10024>
  plugin: <none>
  s_sym: <0>
  literal[1] = [ANALVH4DSTRING]
  literal[2] = [0aiwrkjgaolul;234987 103984a;1k-814 1]
  literal[3] = [*]
  literal[4] = [*]
  literal[5] = [*]
  literal[6] = [Nn]
  literal[7] = [Nf123456x]
  literal[8] = [ks]
  literal[9] = [Xks]
  literal[10] = [X20x]
  literal[11] = [X20x]
  literal[12] = [X20x]
  literal[13] = [X20x]
  literal[14] = [X0knXknXkn]
  literal[15] = [XpXpXpXpXp]
  literal[16] = [XksXksXks]
  literal[17] = [XchXchXch]
  literal[18] = [XksXksXks]
  literal[19] = [X0XpX0]
  literal[20] = [X.10240]
  literal[21] = [X.10250]
  literal[22] = [X.20480]
  literal[23] = [X.20490]
  literal[24] = [X.40960]
  literal[25] = [X.40970]
  literal[26] = [X099999999999999]
  literal[27] = [X00x]
  literal[28] = [X00x]
  literal[29] = [X020n]
  literal[30] = [X20x]
  literal[31] = [X20x]
```



Báo cáo môn học  
HOC KỲ I – NĂM HỌC 2022-2023



## YÊU CẦU CHUNG

- Sinh viên tìm hiểu và thực hiện bài tập theo yêu cầu, hướng dẫn.
- Nộp báo cáo kết quả chi tiết những việc (**Report**) bạn đã thực hiện, quan sát thấy và kèm ảnh chụp màn hình kết quả (nếu có); giải thích cho quan sát (nếu có).
- Sinh viên báo cáo kết quả thực hiện và nộp bài.

### Báo cáo:

- File **.PDF**. Tập trung vào nội dung, không mô tả lý thuyết.
- Nội dung trình bày bằng **Font chữ Times New Romans/ hoặc font chữ của mẫu báo cáo này (UTM Neo Sans Intel/UTM Viet Sach)– cỡ chữ 13. Canh đều (Justify) cho văn bản. Canh giữa (Center) cho ảnh chụp.**
- Đặt tên theo định dạng: [Mã lớp]-ExeX\_GroupY. (trong đó X là Thứ tự Bài tập, Y là mã số thứ tự nhóm trong danh sách mà GV phụ trách công bố).  
*Ví dụ: [NT101.K11.ANTT]-Exe01\_Group03.*
- Nếu báo cáo có nhiều file, nén tất cả file vào file .ZIP với cùng tên file báo cáo.
- **Không đặt tên đúng định dạng – yêu cầu, sẽ KHÔNG chấm điểm bài nộp.**
- Nộp file báo cáo trên theo thời gian đã thống nhất tại [courses.uit.edu.vn](https://courses.uit.edu.vn).

### Đánh giá:

1. Hoàn thành tốt yêu cầu được giao.
2. Có nội dung mở rộng, ứng dụng.

*Bài sao chép, trễ, ... sẽ được xử lý tùy mức độ vi phạm.*

**HẾT**