

The Container Security in Healthcare Data Exchange System

Bachelor's degree graduation project

Chih-Hsuan Yang

National Sun Yat-sen University

Advisor: Chun-I Fan

April 13, 2021

v1.2

Outline

- 1 Outcome
- 2 Next

- 3 Issues
- 4 Q&A

Not expecting:
Realize what will I say.
Too many difficult details.

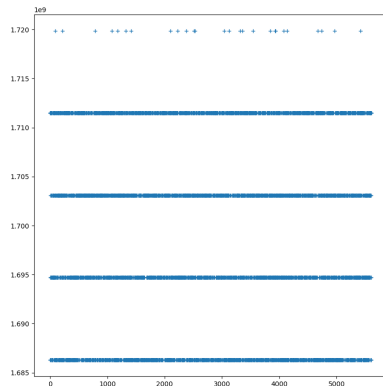
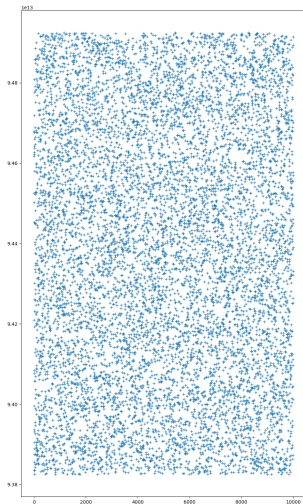
Outcome - light weight container

```

→ container git:(main) X gcc *.c -o c
→ container git:(main) X sudo ./c "bash"
Success on creating container
Start container: bash with clone id: 193761
In container PID: 1
bash-5.0# ./test.sh
This is the self test script in container!
Support bash cat echo ls rm hostname, 5 commands.
./test.sh
-----FILE: test.sh -----
1  #!/bin/bash
2
3  echo "This is the self test script in container!"
4  echo "Support bash cat echo ls rm hostname, 5 commands."
5
6  echo $0
7
8  echo "-----FILE: test.sh -----"
9  cat -n test.sh
10 echo "-----"
11
12 echo $(hostname) >天竺鼠車車
13 cat 天竺鼠車車
14 rm 天竺鼠車車
15 ls
-----
container
bin dev etc home lib lib64 mnt opt proc root run sbin sys test.sh tmp usr var
bash-5.0# exit
exit
→ container git:(main) X

```

Outcome - (k)ASLR



Outcome - kernel module

```
Run /init.sh as init process
  with arguments:
    /init.sh
  with environment:
    HOME=/
    TERM=linux
    hostfs=./rootfs
    mem=64M
kaslr: loading out-of-tree module taints kernel.
```

```
1694699525
random: fast init done
random: crng init done
```

```
→ 0326 git:(main) X less /proc/$$/maps
→ 0326 git:(main) X python -c 'print(1694699525)'
→ 0326 git:(main) X python
```

```
Python 3.8.5 (default, Jan 27 2021, 15:41:15)
[GCC 9.3.0] on linux
Type "help", "copyright", "credits" or "license" for more information.
>>> import math
>>> math.log(1694699525, 2)
30.658382356126655
>>> exit()
→ 0326 git:(main) X
```

Outcome - chroot

```

→ cont git:(main) X sudo ./c ./picosh
Success on creating container
Start container: ./picosh with clone id: 2106853
In container PID: 1
$ ls
bin      dev  home  lib    mnt  picosh  root  sbin  test.sh  usr
chroot-dir  etc  jb    lib64  opt  proc    run   sys   tmp      var
$ ./jb
root@container:/# ls /media/d/git/nsysu/cs/report/presentation/0409/
images      main.bib      main.fdb_latexmk  main.nav  main.run.xml  main.tex  methFlow.pdf
main.aux    main.blg      main.fls          main.out  main.snm      main.toc  src
main.bbl    main-blx.bib  main.log          main.pdf  main.synctex.gz  Makefile
root@container:/#

→ 0409 git:(main) X ls
images      main.bib      main.fdb_latexmk  main.nav  main.run.xml  main.tex  methFlow.pdf
main.aux    main.blg      main.fls          main.out  main.snm      main.toc  src
main.bbl    main-blx.bib  main.log          main.pdf  main.synctex.gz  Makefile
→ 0409 git:(main) X

```

Next steps

- ① Studies
 - ① Linux features
 - ② Papers
- ② Formal Verification
- ③ The Healthcare Data Exchange System part

The 3Ws

- ① What this issue?
- ② Why this issue?
- ③ How this issue?

What this issue

The medical cyber security issue of
the next-generation.

Why this issue

Microservice Scalability Manageability

How this issue

Exactly microservice Security Performance

FAQ

- 1 Docker, container and virtual machine
- 2 FHIR? Why need container?
- 3 User Mode Linux
- 4 clone, io_uring

Q&A

Feel free to give your comments.