The Container Security in Healthcare Data Exchange System

Bachelor's degree graduation project

Chih-Hsuan Yang

National Sun Yat-sen University

March 26, 2021
v1.0

Outline

1. Outcome
2. Related works
3. Current progress
4. Reference

Outcome

Outcome

No outcome.

Related works

Two papers

- A Measurement Study on Linux Container Security: Attacks and Countermeasures[1]
- Container-Based Cloud Platform for Mobile Computation Offloading[2]

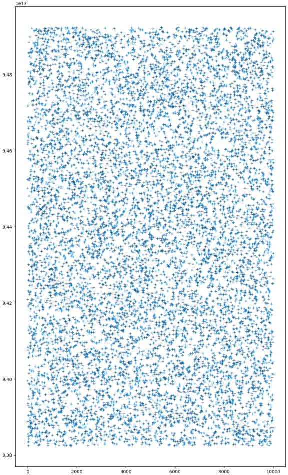
Some Golang/Rust

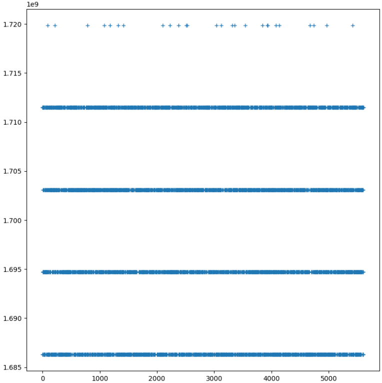
1. The next generation of C/C++
2. High Concurrency, Memory Safe, Traits

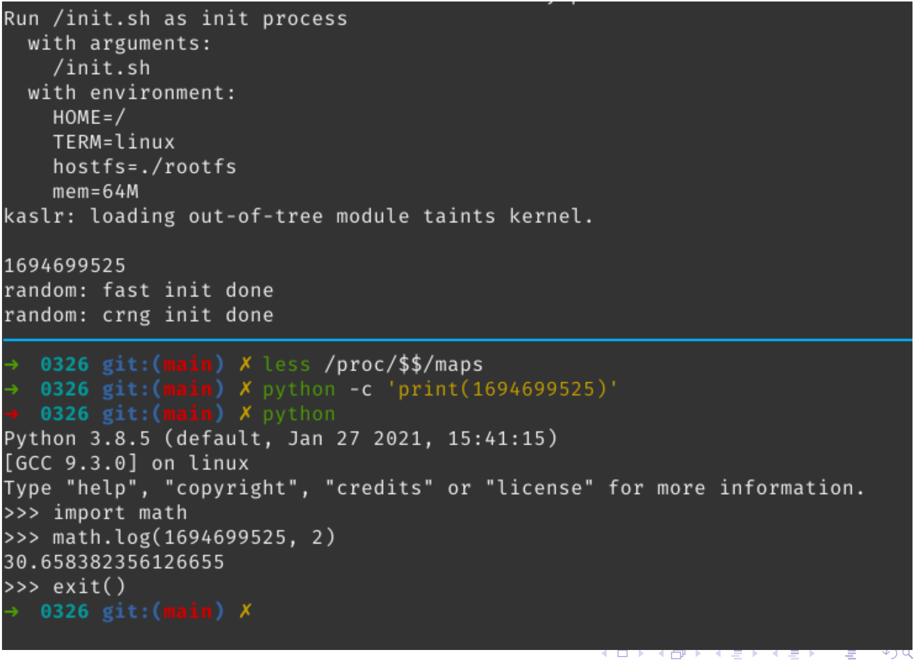
Why?

High performance and secure server.
The docker-engine is written by Golang.

ASLR/KASLR/Finer-grained KASLR







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

Finer-grained KASLR

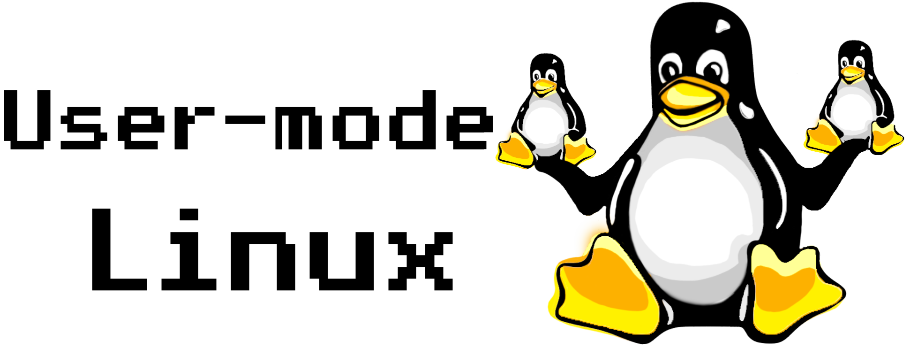
Finer-grained kernel address-space layout randomization[3]

Not merged on mainline yet.

New idea

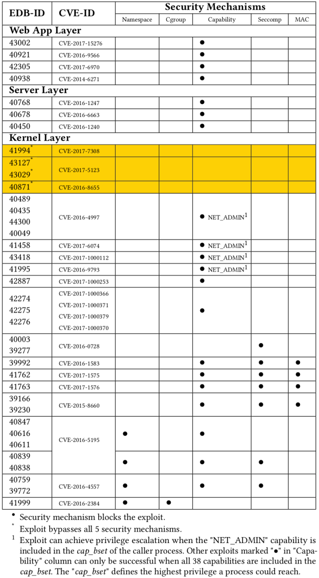
Run container in UML?

https://github.com/weber-software/diuid



Capabilities

Table 3: Function of Security Mechanisms in Preventing Privilege Escalation Attacks



EDB-ID	CVE-ID	Security Mechanisms				
		Namespace	Cgroup	Capability	Seccomp	MAC
Web App Layer						
43002	CVE-2017-15276			●		
40921	CVE-2016-9566			●		
42305	CVE-2017-6970			●		
40938	CVE-2014-6271			●		
Server Layer						
40768	CVE-2016-1247			●		
40678	CVE-2016-6663			●		
40450	CVE-2016-1240			●		
Kernel Layer						
41994[*]	CVE-2017-7308					
43127[*] 43029[*]	CVE-2017-5123					
40871[*]	CVE-2016-8655					
40489 40435 44300 40049	CVE-2016-4997			● NET_ADMIN[1]		
41458	CVE-2017-6074			● NET_ADMIN[1]		
43418	CVE-2017-1000112			● NET_ADMIN[1]		
41995	CVE-2016-9793			● NET_ADMIN[1]		
42887	CVE-2017-1000253			●		
42274 42275 42276	CVE-2017-1000366 CVE-2017-1000371 CVE-2017-1000379 CVE-2017-1000370			●		
40003 39277	CVE-2016-0728				●	
39992	CVE-2016-1583			●	●	●
41762	CVE-2017-1575			●	●	●
41763	CVE-2017-1576			●	●	●
39166 39230	CVE-2015-8660			●	●	●
40847 40616 40611	CVE-2016-5195	●		●		
40839 40838		●		●	●	
40759 39772	CVE-2016-4557	●		●	●	
41999	CVE-2016-2384	●	●			

[•] Security mechanism blocks the exploit.

[*] Exploit bypasses all 5 security mechanisms.

[1] Exploit can achieve privilege escalation when the "NET_ADMIN" capability is included in the *cap_bset* of the caller process. Other exploits marked "•" in "Capability" column can only be successful when all 38 capabilities are included in the *cap_bset*. The "*cap_bset*" defines the highest privilege a process could reach.

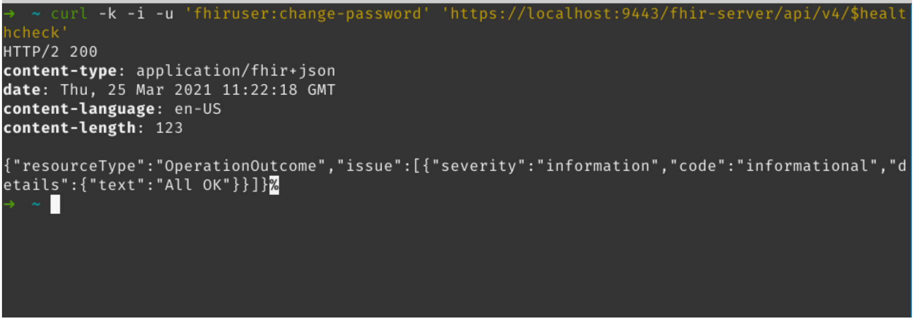
Tim Hsu

1. 「你研究了 capabilities 了嗎？」
   - 有試過，不能算研究。
2. 「如果有 kernel exploit 能打穿 container 那你有那些 anti-exploit 的方式？」
   - Not to kernel: KASLR, SECCOMP, capabilities
   - Landed kernel: Encrypt container database, alert
   - UML? Hypervisor.

Tim Hsu

1. 「先從只用現有機制 user land 的方式再往 kernel land 的方向」
2. 「擋掉 kernel exploit 『可以從無法執行』或『可執行但不會成功』或『可執行但會被限制』等想法」
    - Capabilities, SECCOMP, AppArmor(Docker only)
    - No idea
    - Network Configuration, LSM

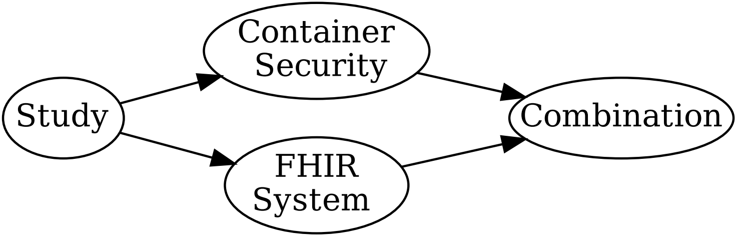
Interact with FHIR in docker



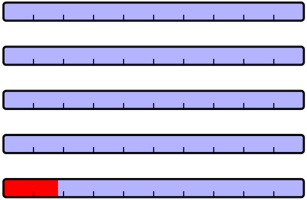
```
→ ~ curl -k -i -u 'fhiruser:change-password' 'https://localhost:9443/fhir-server/api/v4/$healt
hcheck'
HTTP/2 200
content-type: application/fhir+json
date: Thu, 25 Mar 2021 11:22:18 GMT
content-language: en-US
content-length: 123

{"resourceType":"OperationOutcome","issue":[{"severity":"information","code":"informational","d
etails":{"text":"All OK"}}]}%
→ ~
```

Current progress

Map-reduce



Map-reduce



- Study: $\frac{10}{\infty}$
- Container Security: read some
- FHIR system: Configured, can run.
- Combination: Of course: 0
- 專題競賽暨成果展: $\frac{48}{265} \sim 0.181$

Reference

References I

[1] Xin Lin et al. "A Measurement Study on Linux Container Security: Attacks and Countermeasures". In: ACSAC '18. San Juan, PR, USA: Association for Computing Machinery, 2018, 418–429. ISBN: 9781450365697. DOI: 10.1145/3274694.3274720. URL: https://doi.org/10.1145/3274694.3274720.

[2] S. Wu et al. "Container-Based Cloud Platform for Mobile Computation Offloading". In: *2017 IEEE International Parallel and Distributed Processing Symposium (IPDPS)*. 2017, pp. 123–132. DOI: 10.1109/IPDPS.2017.47.

[3] Jake Edge. 2020. URL: https://lwn.net/Articles/812438/.