Mitigating Hackers with Hardening on Linux - an Overview for Developers, focus on BoF
07 Oct 2021
OSI virtual conference

Kaiwan N Billimoria
kaiwan@kaiwantech.com

https://github.com/kaiwan/hacksec

A few notes

sysad's... dev-ops...
ALWAYS update their systems..

esp security patches...

TESTING !!!

Buffer Errors

ptr = malloc(1024);
...

r|w underflow r|w overflow

Buffer OverFlow (BOF)

DoS:

```
while(1) {
    p = malloc(1024);
     memset(...);
}
```

```
// fork bomb !
while (1)
        fork();
```

set resource limits !
see ulimit / prlimit

make use of systemd ! can set resouce limits on processes being exec-ed at boot.....

If you're not running the latest stable kernel (preferably an LTS kernel), you're asking for security headaches !

Passwords-
use a passwd manager app
use a (master) passwd that's RANDOM and 16 chars...

Encryption:
- at rest : storage
- in motion: over the n/w

FOTA updates that are secure

madvise(2)
eg. prevent core-dumping a certain region of mem that has secrets...

32-bit:
2^32 = 4 GB

VAS (every process) is 4 GB

illusion...

map virtual pages (4 KB) to physical pages (pf: page frames); OS does the mapping for every process alive...

user:kernel VAS

3:1 (GB) :: u:k IA-32
2:2 (GB) :: u:k ARM32
3:1

by now the RET addr is : 0xAAAA !
  bogus...

(and we’re out of time! ....)
