Midterm Exam Review Guide

EE382V/EE379K Enterprise Network Security

Exam Topics

- osquery
- sysdig
- Kubernetes
- SELinux policies
- Wireshark + analyzing pcap files
- Containers, Linux namespaces, chroot, cgroups
- Networking basics (IPv4, Tor, VPN, SSL, DNS)
- Networking tools (whois, nslookup)
- Web application vulnerabilities (e.g. CSP Bypass, SQL Injection, etc)

Tools for programming portion

Please have the following tools ready on your machine for the programming portion of the exam:

- Sysdig
- OSQuery
- Wireshark
- Kubernetes (microk8s)

Practice Exercises

Osquery task:

Describe the following tables found in osquery:

- routes
- dns
- users
- processes
- groups

Be able to use osquery to:

- Find the command used to start a process
- Find the shell used by username `class`.
- Find the memory location of the kernel module ip_tables.
- Find all IP address associated with a docker container.
  - This address can be used to access the docker service without having to map the port.
  - i.e., if you run `docker run --rm -it vulnerables/web-dvwa` and find the IP address associated with the container you don't have to run the container with the `-p` flag as suggested `docker run --rm -it -p 80:80 vulnerables/web-dvwa`, in order to access dvwa.

Sysdig

Be able to read a file with sysdig-inspect.

- Record, with sysdig running on your VM, the sql injection attack on your DVWA setup from lab 2 part 2.
- Find evidence of the injection attack in your sysdig file.
- Find the system call associated with your attack.
- Find the network traffic associated with your attack.
- Use filters with sysdig (ex `proc.pid=4594`).
- Use `sysdig -l | less` to see other filters apply filters.
- `docker run -it -v $(pwd):/captures -p8085:3000 sysdig/sysdig-inspect:latest`

Networking

- What does each networking layer do(Physical, Ethernet, IP, TCP, HTTP)?
- What is happening in the following pcap file.
  - https://drive.google.com/open?id=1qAbSRA5Y1zdDlZN-1k1bmDMfRVZyoGCg

SELinux

You are given a sample wikipedia policy file wiki_messed.te. Wiki files are allowed to view, read, write, and create, only wiki_var_t type files. The wikipedia application has a label of wiki_t and hence whenever we create a new file they are created with wiki_var_t type. Look at the given policy file and find out all the mistakes that has been made in the given file and correct them. Note: You can assume all lines of code before the comment are fine. Init and socket lines of code are assumed to be correctly configured.

HINT: There are 3 lines of mistakes.

```
policy_module(wiki, 1.0)
userdom_unpriv_user_template(wiki)
type wiki_var_t;
files_type(wiki_var_t)
require {
  type home_root_t;
    type user_home_t;
    type init_t;
    class file { create execute open read write getattr execute_no_trans};
    type http_cache_port_t;
}

########################################

# messed up wiki local policy
type_transition wiki_t wiki_exec_t:file wiki_exec_t;
type_transition wiki_t wiki_var_t:dir wiki_var_t;

#============= socket ==============
allow wiki_t http_cache_port_t:tcp_socket name_bind;

#============= init_t ==============
allow init_t wiki_var_t:file execute;
allow init_t home_root_t:file execute;
allow init_t user_home_t:file execute;

#============= wiki_t ==============
allow wiki_t wiki_var_t:file {execute read write getattr};
allow wiki_t wiki_var_t:dir {search add_name};
```