



دانشکده مهندسی کامپیوتر

پروژه ایریدیوم درس امنیت سیستم‌های کامپیوتری

گردآورندگان: مهدی قضاوی - ۹۹۵۲۲۰۱۴ احسان احمدپور - ۹۹۵۲۱۰۱۹

نیم‌سال دوم

سال تحصیلی ۱۴۰۲-۱۴۰۳

## ۱.۰ مقدمه

در این پروژه قصد داریم بدافزار میرای (Mirai) را شبیه‌سازی کنیم. میرای با دسترسی به تعداد کثیری از سیستم‌ها و ارسال درخواست DNS، Dyn، سرورها را از کار می‌انداخت و باعث قطع دسترسی بسیاری از وبسایت‌های معروف دنیا می‌شد.

هدف این پروژه ساخت برنامه‌ای است که پورت‌های باز موجود در شبکه که ssh در آن‌ها اجرا می‌شود را پیدا کند و سپس با تست کردن رمز عبورهای معروف که در تعداد زیادی از دستگاه‌ها استفاده می‌شوند به این سیستم‌ها دسترسی پیدا کرده و با پیاده‌سازی یک بدافزار روی آن‌ها اطلاعات امنیتی این سیستم‌ها را به یک سرور مشخص ارسال کند.

## ۲.۰ ساختار شبکه

در این پروژه با استفاده از داکر یک شبکه ایجاد کرده و تعدادی کاننتینر به آن اضافه کرده و در نهایت یک حمله را در این شبکه شبیه سازی کرده‌ایم.

این شبکه دارای سه نوع image است: وب‌سرور، حمله‌کننده و سرور هدف. برای هر image یک Dockerfile پیاده‌شده که از image‌های سرورهدف چند و از image‌های سیستم‌های وب‌سرور و حمله‌کننده تنها یک کاننتینر ایجاد می‌شوند.

## ۳.۰ وب‌سرور

از این وب‌سرور که با جنگو پیاده‌سازی شده برای اهداف مختلفی مانند ارسال اطلاعات امنیتی جمع‌آوری شده از سیستم‌های مورد حمله قرار گرفته و ارسال به این سرور و ذخیره اطلاعات در این دیتابیس و دانلود بدافزار توسط سیستم‌های قربانی از این سرور استفاده می‌شود.

اسکرپت infocrawler.sh در این وب‌سرور ذخیره‌شده است که روی سیستم مورد هدف دانلود شده و اطلاعات امنیتی سیستم هدف را به صورت json به وب‌سرور ارسال می‌کند.

Change host info

172.26.0.3

|                   |  |
|-------------------|--|
| Total memory:     | 31.1G  |
| Cpu model:        | 11th Gen Intel(R) Core(TM) i7-11800H @ 2.30i |
| Os:               | Alpine Linux v3.18                           |
| Disk space:       | 280.3G                                       |
| Hostname:         | 431281b84f6a                                 |
| MACs:             | 02:42:ac:1a:00:03                            |
| Users:            |  |
| Available memory: | 24.8G  |
| Disk usage:       | 34%  |
| IPs:              | 172.26.0.3                                   |
| Kernel:           | 5.19.0-kali2-amd64                           |
| Free space:       | 177.3G                                       |
| Open ports:       | 22 34355 52510                               |

SAVE Save and add another Save and continue editing

شکل ۱: نمونه اطلاعات استخراج شده از سیستم هدف

## ۴.۰ Image Attacker

برای حمله به سیستم‌های قربانی یک attacker image ساخته شده است. سه فایل موجود در این ایمیج `scan.sh`، `userpass.csv` و `hack.sh` می‌باشند.

اسکرپت `hack.sh` با خواندن `openports.csv` به پورت‌های `ssh` پیدا شده حمله کرده و رمزعبورهای معروف با `Brute-force` تست کرده و در صورت اتصال، بدافزار را از وب سرور بر روی سرور قربانی دانلود و اجرا می‌کند.

فایل `userpass.csv` دارای `username`ها و `password`های معروف و پرتکرار می‌باشد و در حمله به `ssh` استفاده می‌شود.

اسکرپت `scan.sh` هاست‌های فعال در رنج ورودی را تست کرده و پورت‌های باز آن‌ها را پیدا می‌کند

که در نهایت اطلاعات پیدا شده را در فایل openports ذخیره می‌کند.

## ۵.۰ سرور هدف

image های سرورهای هدف که مورد حمله قرار می‌گیرند روی لینوکس alpine قرار گرفته اند تا سبک باشند.

## ۶.۰ اجرای حمله

ابتدا با اسکریپت buildimages.sh شبکه را راه اندازی می‌کنیم و image ها را می‌سازیم و سپس اسکریپت setupsimulator.sh را برای اجرا کردن کانتینرها اجرا می‌کنیم. برای مشاهده دیتابیس، می‌توان به پورت 8000 و قسمت ادمین مراجعه کرد ( 127.0.0.1:8000/admin ، نام کاربری و پسورد : superuser:superuser )

```

(mahdi@kali)~/root/UniCodes/CS4023/Projects/Iridium Project
$ ./image_builder.sh
-----
Building the attacker image
-----
Sending build context to Docker daemon 6.144kB
Step 1/9 : FROM alpine:3.18
--> d3782b16ccc9
Step 2/9 : USER root
--> Using cache
--> 16f86d721ba2
Step 3/9 : RUN echo -e "root\nroot" | passwd
--> Using cache
--> a51085a24a1d
Step 4/9 : RUN apk update && apk add busybox-extras openssh openssh
--> Using cache
--> 23a60d4c34e4
Step 5/9 : RUN echo "PermitRootLogin yes" >> /etc/ssh/sshd_config &
--> Using cache
--> 9a174031e637
Step 6/9 : RUN apk add vsftpd
--> Using cache
--> ea094fe17167
Step 7/9 : RUN rc-update add vsftpd default
--> Using cache
--> 62dd31493fec
Step 8/9 : COPY .//* /root/
--> Using cache
--> 21a7f1985f4b
Step 9/9 : CMD ["/bin/sh"]
--> Using cache
--> 7c5bea8442f1
Successfully built 7c5bea8442f1
Successfully tagged attacker-machine:1.0.0

```

شکل ۲: راه اندازی image ها

## ۷.۰ حمله به سیستم های هدف

ابتدا درس network شبکه داکر که کانتینرها در آن در حال اجرا هستند را پیدا کرده و سپس با اجرای فایل scan.sh نتایج مورد نظر در فایل openports.csv ذخیره می‌شوند. سپس با اجرای hack.sh حمله آغاز می‌شود و در هر دقیقه یک بار اطلاعات امنیتی سیستم های هدف در دیتابیس ذخیره می‌شوند.

```
(mahdi@kali)-[/root/UniCodes/CS4023/Projects/Iridium Project]
$ ./setup_simulator.sh

-----
Creating the docker network...
70e1f5d0e71b36705575e1f2c4d865716d65031abd981de4dad74e17d72e3be0
-----
Creating the target servers...
60c2f1ca72222ecaab01677be7c98daf6611a0064ecc773e576f418a364acd3
431281b84f6a3e02be63dc57430a35e180cbde1fb55a26bbab2fe1ff656afbb
eec4307fc6246e2b33e6faab6382ef907bedbf70fd1fbbf2d0804cd12cd660a9
-----
Starting ssh and ftp services on the target servers...
-----
Creating the web server...
fa5f3e7665ebec175ff88b06327749f7607f6bcc3cca1cc7f9a798b970f5cca
-----
Creating the attacker machine...
-----
```

شکل ۳: اجرای کانتنرها

```
/ # ifconfig
eth0      Link encap:Ethernet  HWaddr 02:42:AC:1A:00:06
          inet addr:172.26.0.0  Bcast:172.26.255.255  Mask:255.255.
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:6 errors:0 dropped:0 overruns:0 frame:0
          TX packets:0 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:0
          RX bytes:516 (516.0 B)  TX bytes:0 (0.0 B)

lo        Link encap:Local Loopback
          inet addr:127.0.0.1  Mask:255.0.0.0
          UP LOOPBACK RUNNING  MTU:65536  Metric:1
          RX packets:0 errors:0 dropped:0 overruns:0 frame:0
          TX packets:0 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:0 (0.0 B)  TX bytes:0 (0.0 B)
```

شکل ۴: اسکن کردن شبکه

```
- # cat open_ports.csv
172.26.0.1,8000/tcp,open
(172.26.0.2),21/tcp,open
(172.26.0.2),22/tcp,open
(172.26.0.3),22/tcp,open
(172.26.0.4),21/tcp,open
(172.26.0.5),8000/tcp,open
```

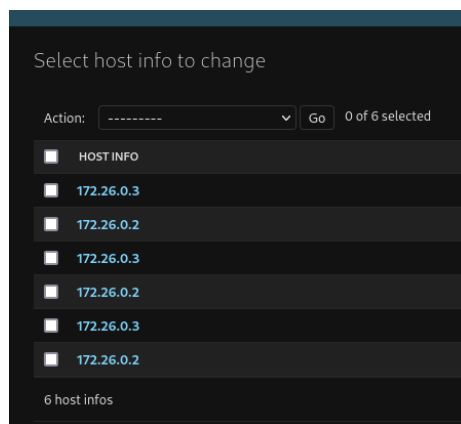
شکل ۵: پورت‌های باز پیدا شده

```
~ # source hack.sh 172.26.0.5
Brute-forcing SSH on 172.26.0.2:22
SSH Successful\!! root:root
Brute-forcing SSH on 172.26.0.3:22
SSH Successful\!! root:root
" 172.26.0.4
```

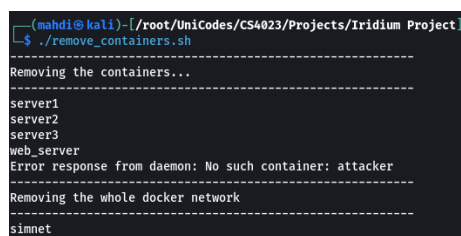
شکل ۶: اجرای اسکریپت hack

## ۸.۰ پایان حمله

در نهایت با استفاده از اسکریپت removecontainers.sh کانتنرهای در حال اجرا را متوقف کرده و شبکه داکر را حذف می‌کنیم.



شکل ۷: هاست‌های ذخیره‌شده در دیتابیس



شکل ۸: حذف کانتینرها و پایان حمله

## ۹.۰ داکرهاب و گیت‌هاب

Image هریک از بخش‌های پروژه در [در این لینک](#) و کد پروژه پیاده‌شده در [این لینک](#) گیت‌هاب قرار دارند.