How-To Survive an A/D CTF

https://github.com/ENOFLAG/ENOINTRO

Structure

- Attack/Defense CTFs
  - Overview
  - How to Earn Points
- Your Machine
  - Exploring the Server
  - Docker Basics
- Traffic Analysis
- Exploit Automation

Attack/Defense CTFs

https://www.youtube.com/watch?v=RXgp4cDbiq4&t=85

How to Earn Points

- Attacking other teams
  - Find vulnerabilities in your services
  - Write exploits to attack other teams and capture their flags
  - Hand flags to the game server and earn points!
- Defending your team
  - Patch found vulnerabilities
  - Other teams cannot exploit your service and you don’t lose points!
- Service Level Agreement
  - If the game server detects a service to be offline / to not work as intended, you will lose points!

Your Machine

Exploring the Server

- First hour of CTF: Recon (network is closed)
    - Check defaults for misconfigurations, weak passwords etc.
    - Find services and exposed ports
    - Learn how to interact with services, how to patch and restart
    - Build teams, start analyzing

```
# interact with systemd services
systemctl status <SERVICE>
systemctl start <SERVICE>
          restart <SERVICE>
          stop <SERVICE>

# check logs
journalctl -xe # start at bottom, add explanations
journalctl -fu <SERVICE> # live updates

# show programs and ports they are listening on
netstat -tulpen
ss -tulpen

# search for service (or their source code) on system
find / -name '*<SERVICE_NAME>*' 2>/dev/null
```

Docker Basics

- Container Virtualization: easily™ ship and deploy software
- Defined through ‘Dockerfile’s

- docker-compose: configure and run services consisting of multiple docker containers
  - E.g.:
    - Web Server
    - Backend
    - Database

```
$ cat Dockerfile

FROM ubuntu:latest

RUN apt update -y && apt install python3 -y && apt
install socat -y

COPY ./service.py /root

ENTRYPOINT python3 /root/service.py'
```

```
# show show running and inactive containers
docker ps --all
docker stats

# build container image from Dockerfile in CWD
docker build .
docker-compose build

# add -d to run in background
docker run -p 5000:5000 <IMAGE_ID>
docker-compose up

# attach to running container
docker exec -it <CONTAINER_ID> /bin/bash

docker kill <CONTAINER_ID>
```

Traffic Analysis

Traffic Analysis

- Vital Tool
    - Monitor outgoing traffic
        - See when you're losing flags
    - Monitor incoming traffic
        - Analyze attacks of other teams to replicate them

- Our setup: - capture all traffic on game router
  - analyze with ENOMIND / Moloch / Wireshark

```
# listen on eth0 interface with human-readable timestamp
tcpdump -i eth0 -t

# capture packets greater than 1024 bytes and write to file
tcpdump -w tmp.pcap greater 1024

# capture packets for particular dest IP addr and port
tcpdump dst 10.10.10.69 and port 80

# only show packets of specific protocol
# (other examples: ip6, arp, udp)
tcpdump -i any arp

# [n]o address resolution, [e]thernet frames,
# skip tcp checksum verification
tcpdump -neK
```

Live Demo

Exploit Automation

```python
#!/usr/bin/env python3

import telnetlib as tl
import requests

cheatsheet = requests.get(
"https://github.com/ENOFLAG/enointro/blob/master/survival/cheatsheet.py")

flag_submission = requests.get("http://flag.stronk.pw:1338/")

service = tl.Telnet("service.stronk.pw", 1337)
service.read_some()
```

ENOTRAINING

https://training.enoflag.de

Auth-Key: ZW5vZmxhZw==