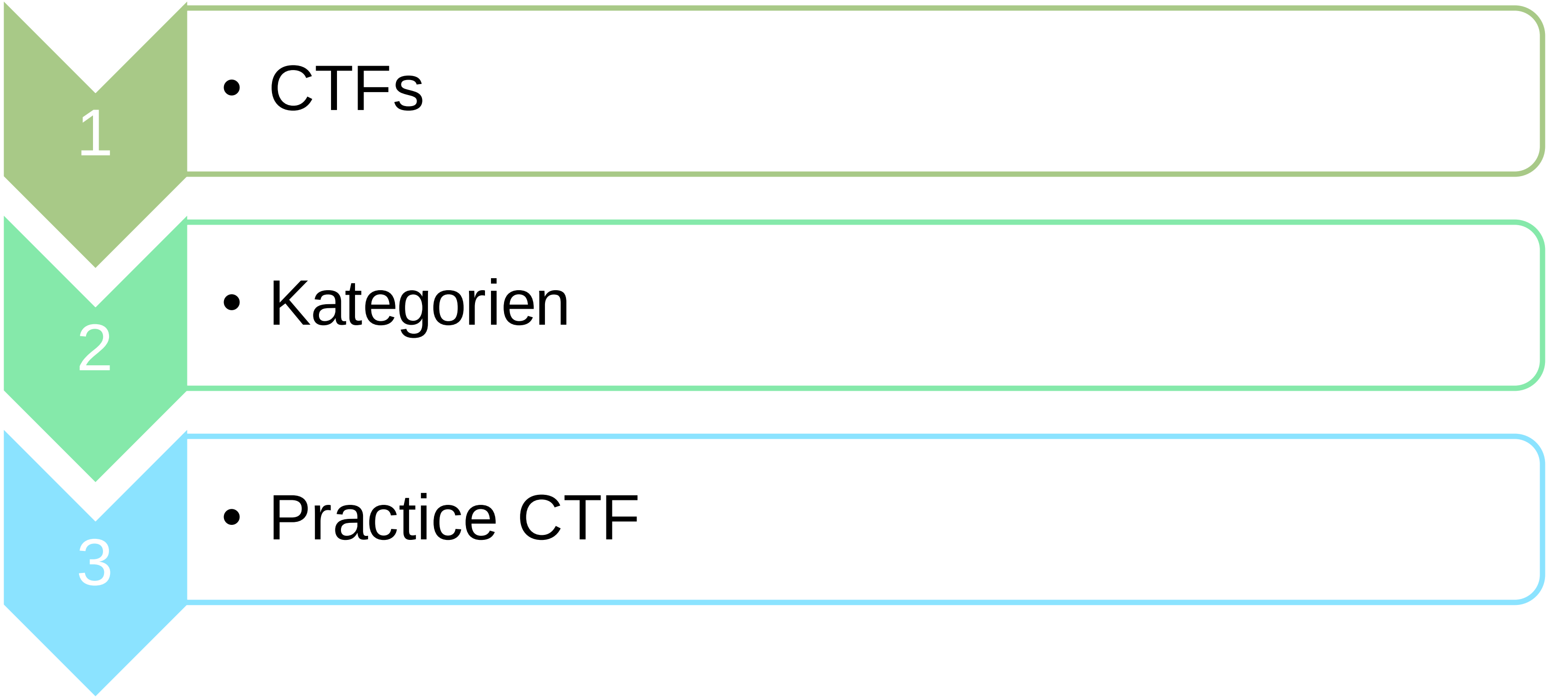
CAPTURE THE FLAG

CTF @ DHBW

Agenda



1

- CTFs

2

- Kategorien

3

- Practice CTF

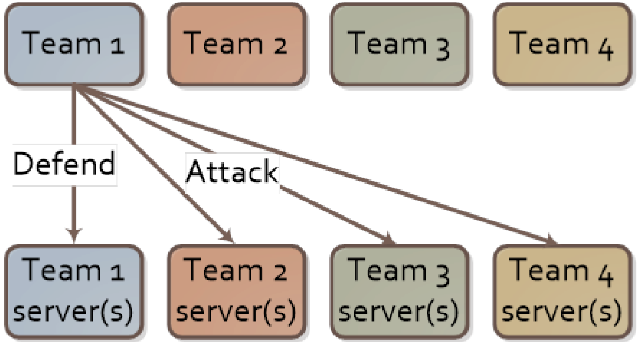
01
Jeopardy vs. Attack-Defense

Attack – Defense

Schwachstellen finden und absichern

Wird oft als Format für ein Finale genutzt

Flags werden von Admins vergeben, wenn erfolgreich exploited



Jeopardy

Crypto

Challenges

- Crypto 1 – 250
- Crypto 2 – 500
- Crypto 3 – 500

Revers

Challenges

- Revers 1 – 250
- Revers 2 – 500
- Revers 3 – 500

Misc

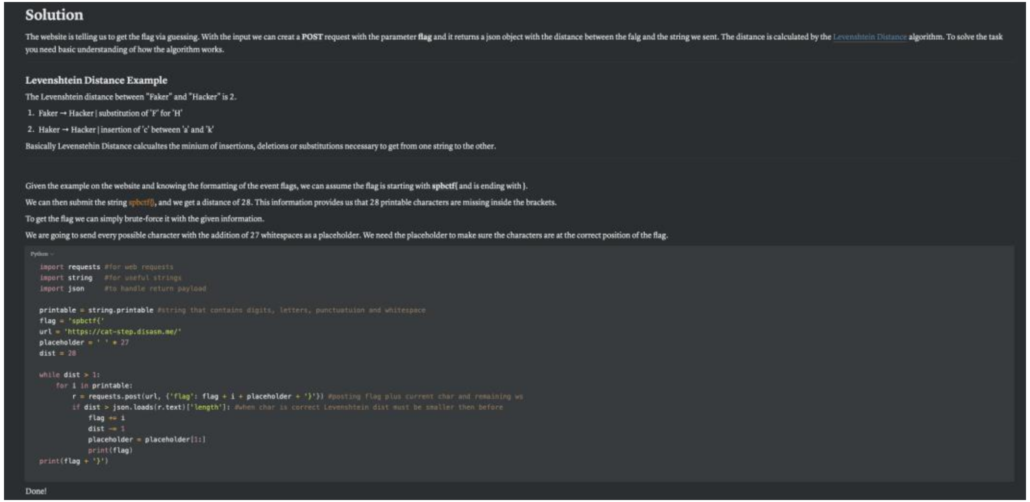
Challenges

- Misc 1 – 250
- Misc 2 – 500
- Misc 3 – 500

Jeopardy

- Kategorien
- Viele Challenges
- Punkte können dynamisch sein
- Häufigstes Event Format

Writeups



Solution

The website is telling us to get the flag via guessing. With the input we can creat a **POST** request with the parameter **flag** and it returns a json object with the distance between the falg and the string we sent. The distance is calculated by the Levenshtein Distance algorithm. To solve the task you need basic understanding of how the algorithm works.

Levenshtein Distance Example

The Levenshtein distance between "Faker" and "Hacker" is 2.

1. Faker → Hacker | substitution of 'F' for 'H'
2. Haker → Hacker | insertion of 'c' between 'a' and 'k'

Basically Levenstehin Distance calcualtes the minium of insertions, deletions or substitutions necessary to get from one string to the other.

Given the example on the website and knowing the formatting of the event flags, we can assume the flag is starting with **spbctf{** and is ending with }.

We can then submit the string spbctf{}, and we get a distance of 28. This information provides us that 28 printable characters are missing inside the brackets.

To get the flag we can simply brute-force it with the given information.

We are going to send every possible character with the addition of 27 whitespaces as a placeholder. We need the placeholder to make sure the characters are at the correct position of the flag.

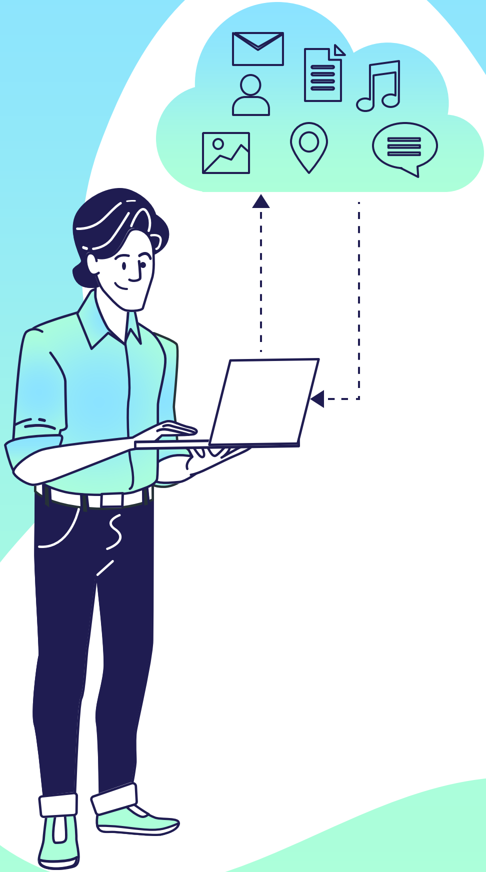
```python
import requests #for web requests
import string   #for useful strings
import json     #to handle return payload

printable = string.printable #string that contains digits, letters, punctuatuion and whitespace
flag = 'spbctf{'
url = 'https://cat-step.disasm.me/'
placeholder = ' ' * 27
dist = 28

while dist > 1:
    for i in printable:
        r = requests.post(url, {'flag': flag + i + placeholder + '}'}) #posting flag plus current char and remaining ws
        if dist > json.loads(r.text)['length']: #when char is correct Levenshtein dist must be smaller then before
            flag += i
            dist -= 1
            placeholder = placeholder[1:]
            print(flag)
print(flag + '}')
```

Done!

02

Kategorien

In Jeopardy



OSINT 01

Forensic 02

Exploit 03

04 Crypto

05 Revers

06 Misc

2.01

OSINT

OSINT

Flag in Social-Media finden

Gegeben: Username, Cloudspeicher, Location

Flagformat manchmal abnormal

z.B. E-Mail

Tools

Who.is, instantusername.com, Google

2.02

Forensic

Forensic

Flag in Dateien finden

Gegeben: Files, Netzwerk mitschnitte, Archive

File format analysis

Suche nach versteckten „etwas" in einer Datei

Methoden

Steganography, memory dump analysis, network capture analysis

Exploit

- Enthält oft Crythography
- Oft täuschen Dateiendungen
- Suche nach dem „FIngerabdruck“

Vorgehensweise

1. Files analysieren
2. Hinweise suchen
3. Flag auslesen

Forensic - Tooling

strings

- String Charactern in Dateien anzeigen
- Text aus binary extrahieren

WireShark

- Netzwerkverkehr überprüfen

exif

- Original Informationen
- Metadatenanalyse

binwalk

- Sucht binary in Audio und Bild

zsteg

- Versteckte Daten in Bildern (png & bmp)

stegsolve

- Color filters, invert etc...
- Tools: Photoshop, GIMP

2.03

(Web-)Exploit

Exploit

- Gegeben: (Web) Anwendung
- Ziel: Webseite kompromittieren, Zugriff erlangen

Vorgehensweise

1. Webanwendung analysieren
2. Sicherheitslücke(n) finden und ausnutzen
3. Manchmal: Auf Webserver Rechte erhöhen
4. Flag auslesen

Exploit - Tooling

Burp Suite

- HTTP Proxy für Netzwerkanalyse
- Attacken ausführen

nmap

- Port + Vulnerability Scanner

gobuster

- HTTP brute-forcing (URLs / DNS)

exploit-db

- Datenbank mit Sicherheitslücken

metasploit

- Penetration testing Framework
- Viele vordefinierte Skripe

weitere Tools

- Hydra – bruteforcing
- Zed Attack Proxy – OWASP web scanner

Burp Suite Demo

2.04

Crypto

Crypto

Challenges:

Textdateien mit verschlüsselten Text

Zip-Dateien

Verschlüsselungsalgorithmus

Tools:
CyberChef
Cryptii

2.05

Revers

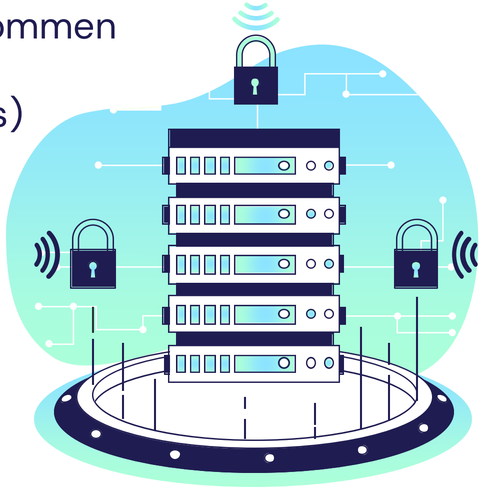
Revers

- Gegeben: Eine oder mehrere Binaries
- Ziel: Flag finden, entweder in der Binary oder Datei auf einem Server

- crackme – Korrektes Passwort / Key finden oder generieren
- pwn – Sicherheitslücke finden und ausnutzen, um Zugriff zu bekommen
- Flag liegt oft in einer Textdatei (flag.txt)
- Verbindung mit Challenge-Server oft über netcat (oder pwntools)

Vorgehensweise

1. Lokale Datei reversen und Exploit erstellen
2. Per netcat (oder pwntools) mit Challenge-Server verbinden
3. Exploit auf dem Challenge Server anwenden



Revers - Tooling

Ghidra (Win/Unix)

- Software Reverse Engineering (SRE) Framework

GDB + GEF (Unix)

- GNU-Debugger
- GEF = GDB Enhanced Features

x64dbg (Win)

- x86/64 Debugger für Windows

pwntools (Win/Unix)

- CTF Framework
- Exploit development library

radare2 (Unix)

- Reverse Engineering Framework

weitere Tools

- ApkTool (Android)
- Fiddler (Netzwerk)
- Detect It Easy (Win/Unix)

Ghidra Demo

2.06

Miscellaneous (Misc)

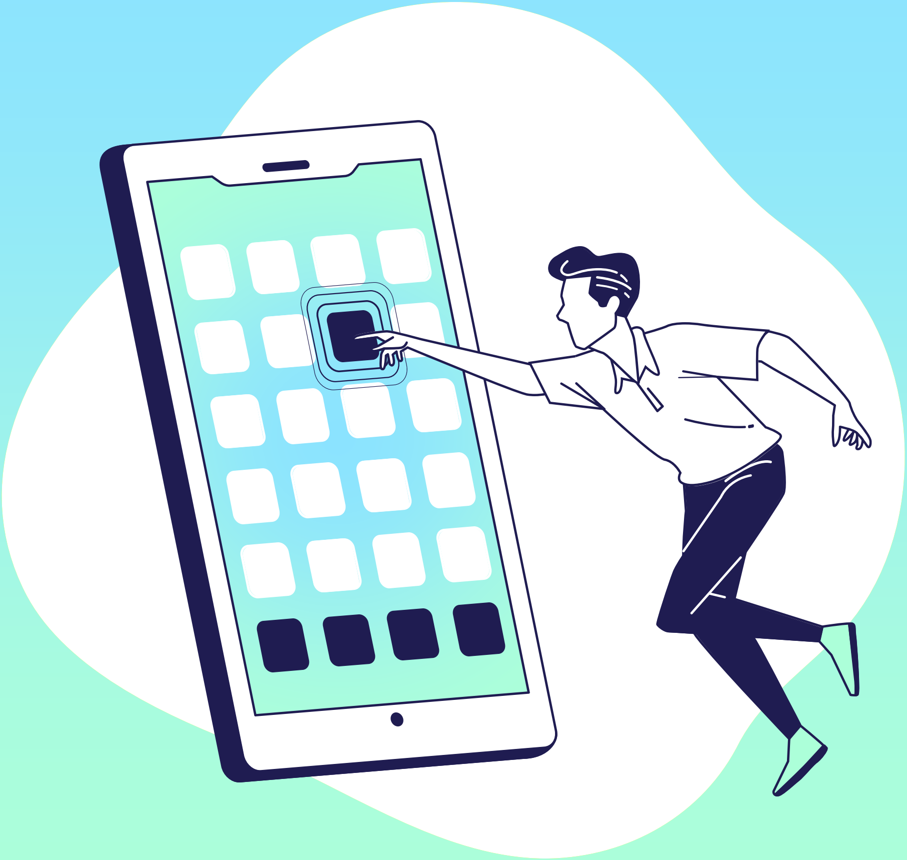
Same like forensic but different

Miscellaneous

- Völlig zufällige Aufgaben
- Oft logisches denken
- Wissen + Geduld nötig
- Vorbereitung schwer möglich

- Übung macht den Meister

- Viele Ähnlichkeiten zu Forensic



03

Practice

Happy Hacking

Practice

TryHackMe

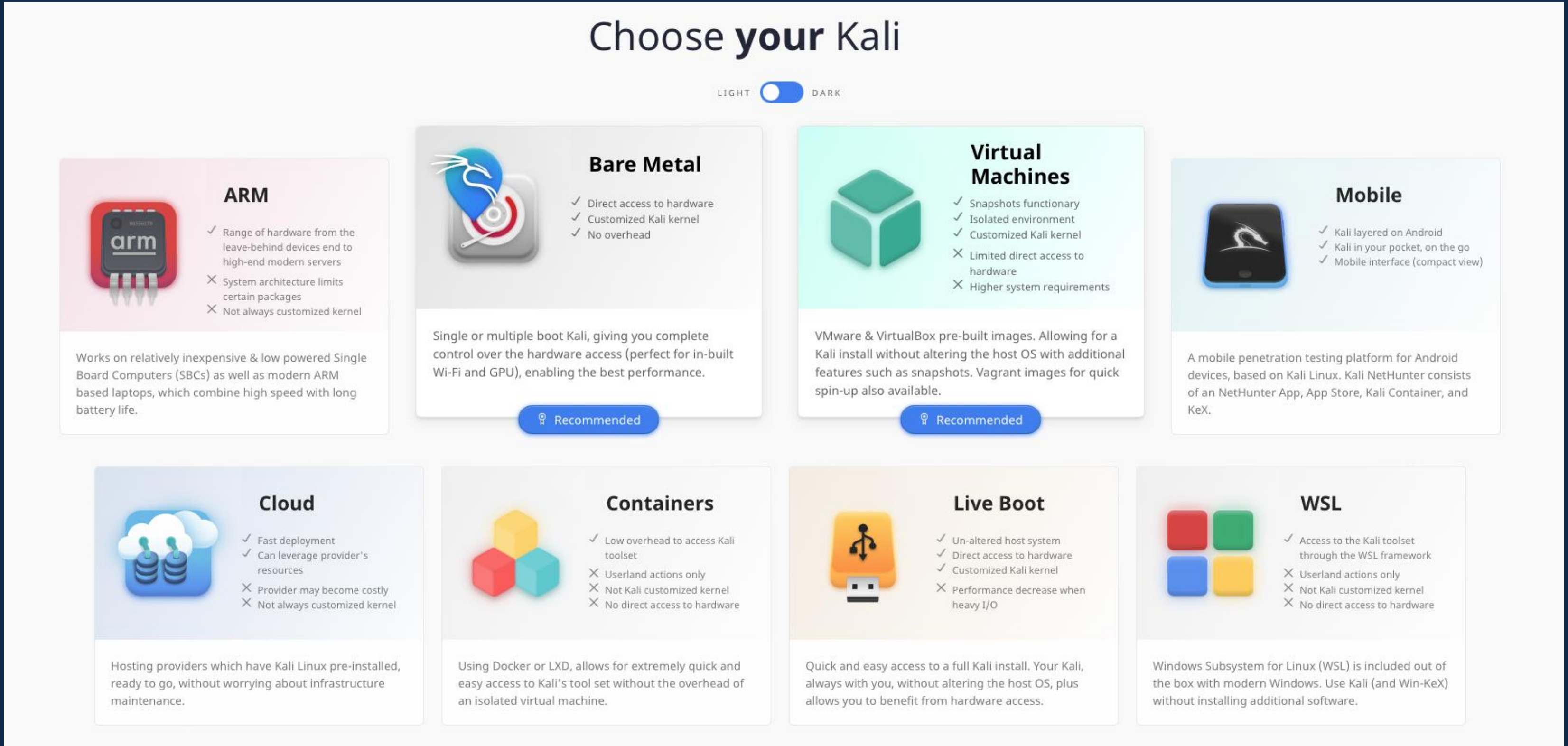
HackTheBox

WeChall

Enter CTFs

Balsn

Google
CTF



Choose **your** Kali

ARM

- ✓ Range of hardware from the leave-behind devices end to high-end modern servers
- ✗ System architecture limits certain packages
- ✗ Not always customized kernel

Works on relatively inexpensive & low powered Single Board Computers (SBCs) as well as modern ARM based laptops, which combine high speed with long battery life.

Bare Metal

- ✓ Direct access to hardware
- ✓ Customized Kali kernel
- ✓ No overhead

Single or multiple boot Kali, giving you complete control over the hardware access (perfect for in-built Wi-Fi and GPU), enabling the best performance.

Recommended

Virtual Machines

- ✓ Snapshots functionary
- ✓ Isolated environment
- ✓ Customized Kali kernel
- ✗ Limited direct access to hardware
- ✗ Higher system requirements

VMware & VirtualBox pre-built images. Allowing for a Kali install without altering the host OS with additional features such as snapshots. Vagrant images for quick spin-up also available.

Recommended

Mobile

- ✓ Kali layered on Android
- ✓ Kali in your pocket, on the go
- ✓ Mobile interface (compact view)

A mobile penetration testing platform for Android devices, based on Kali Linux. Kali NetHunter consists of an NetHunter App, App Store, Kali Container, and KeX.

Cloud

- ✓ Fast deployment
- ✓ Can leverage provider's resources
- ✗ Provider may become costly
- ✗ Not always customized kernel

Hosting providers which have Kali Linux pre-installed, ready to go, without worrying about infrastructure maintenance.

Containers

- ✓ Low overhead to access Kali toolset
- ✗ Userland actions only
- ✗ Not Kali customized kernel
- ✗ No direct access to hardware

Using Docker or LXD, allows for extremely quick and easy access to Kali's tool set without the overhead of an isolated virtual machine.

Live Boot

- ✓ Un-altered host system
- ✓ Direct access to hardware
- ✓ Customized Kali kernel
- ✗ Performance decrease when heavy I/O

Quick and easy access to a full Kali install. Your Kali, always with you, without altering the host OS, plus allows you to benefit from hardware access.

WSL

- ✓ Access to the Kali toolset through the WSL framework
- ✗ Userland actions only
- ✗ Not Kali customized kernel
- ✗ No direct access to hardware

Windows Subsystem for Linux (WSL) is included out of the box with modern Windows. Use Kali (and Win-KeX) without installing additional software.