UiO-CTF

IN2120: Ethical hacking

$ who

```
MaritIrenRognliTokle :0     2019-10-31 14:15 (:0)

      UiO-CTF Team Captain
      Leader TG:Hack
      Senior Software Engineer Sopra Steria
      Member of bootplug and pwnrpuffgirls
      Pwn, mobile and web

DanielHeinesen :1           2019-10-31 14:15 (:1)

      UiO-CTF Team Captain
      Student at Institute of Theoretical Astrophysics, UiO
      Reversing and Crypto
```

$ agenda - del 1

> UiO-CTF intro

> Hacking

> Hack yourself a job!

> CTF?! WTF??

> Categories of hacking

$ agenda - del 2

> Reverse engineering

> Cryptography

> Web exploitation

> Binary exploitation

What does UiO-CTF do?

- **Practical workshops:** Going through tasks from hacking competitions
- **Presentations:** Security related issues
- **CTF nights:** Pizza and hacking.
- **CTF participation:** Online competitions

UiO-CTF's goal

- Teach security
- **make people aware of the importance of security!**
- Get people interested in CTFs (hacking competitions) - Compete with us!

Participate in UiO-CTF events

- Awesome and very useful knowledge about programming, hacking and security
- More attractive to employers(!)
- It's super fun and addictive!

We need more UiO-CTFers! Sounds fun? Join us!

What is hacking?

Hacking

$ steps of hacking:

1. Do reconnaissance and find vulnerability in system

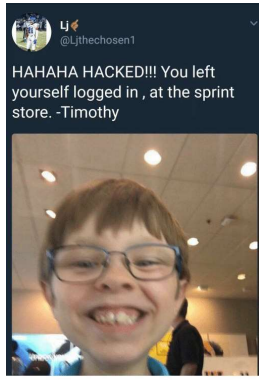
2. Use vulnerability against the system

Hacking

$ steps of hacking:

1. Do reconnaissance and find vulnerability in system

2. Use vulnerability against the system



Elite hacker

Ethical hacking

$ ethical (fun) hacking vs. criminal (boring) hacking
$ black hat, white hat and grey hat hacking

Ethical hacking

$ what to do when finding vulnerabilities?

1. Don't exploit the vulnerability
2. Report it - Responsible disclosure

Ethical hacking

$ responsible disclosure

- ❏ I noticed a vulnerability while surfing your webpage...
- ❏ Your webpage has piece of shit security...

Ethical hacking

Den trettenårige eleven skal ha fulgt en mappestruktur i systemet og funnet informasjon om brukernavn og passord for elever og ansatte. Han varslet om sikkerhetshullet i kommunens datasystem da han først oppdaget det, og skal ha gitt tydelig beskjed til skolen.

Skoleelev varslet om datahull i Bergen

https://www.vg.no/nyheter/innenriks/i/EoAnPA/skoleelev-varslet-om-datahull-i-bergen
https://www.digi.no/artikler/politet-mener-eleven-som-varslet-om-sikkerhetshull-har-begatt-straffbar-handling/462350

Ethical hacking

Varselet ble ikke tatt tak i, og da han et halvt år senere oppdaget at hullet fortsatt var der, logget han seg inn som administrasjonsbruker og sendte ut en epost i rektors navn hvor han latterliggjorde sikkerheten i datasystemet til kommunen. Han skal også ha lastet ned og lagret informasjon om brukernavn og passord.

Politiet mener eleven som varslet om sikkerhetshull har begått straffbar handling

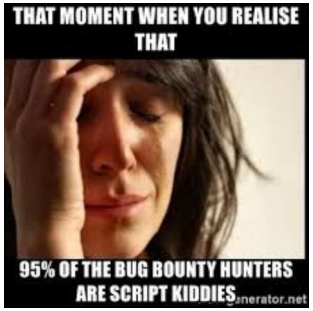
https://www.digi.no/artikler/politet-mener-eleven-som-varslet-om-sikkerhetshull-har-begatt-straffbar-handling/462350

Hack yourself a job!

Hack yourself a job!

- Reverse engineering malware
- Memory, disk and network forensics
- Incident Response Team (IRT)
- Security researching
- Bug bounties

- Pentesting
- Secure development
- Infrastructure
- Security Operations Center (SOC)



CTF!? WTF?

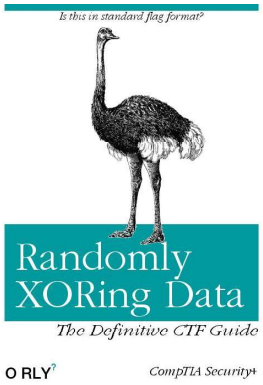
"CTF is a **hacking competition** with a wide range of security related **challenges** where the goal is to find a **flag**."

What's a flag?

UiO-CTF{fancy_text123}

CTF

- Capture the Flag
- Ethical hacking
- Team based competition
- Challenges in the computer security domain
- Two modes:
  - Attack & Defense
  - Jeopardy Style



Jeopardy style

- Different categories
- How to play the game
- How to win

Gains of CTFing

- Awesome way to learn programming, security and hacking.
  - Writing/reading writeups
- Team work & friendship
- Fun and addictive
  - Accomplishment
  - Dueling with known teams in scoreboards
  - Global team rating
- Attractive for employers

We would argue that most CTFers usually has a higher level of knowledge than the common developer/security person.

$ agenda - del 2

> Reverse engineering

> Cryptography

> Web exploitation

> Binary exploitation

Reverse engineering

Reverse Engineering

$ Definition

"Reverse Engineering in a CTF is typically the process of taking a compiled (machine code, bytecode) program and converting it back into a more human readable format.
Very often the goal of a reverse engineering challenge is to understand the functionality of a given program such that you can identify deeper issues."

https://ctf101.org/reverse-engineering/overview/

Reverse Engineering

$ What does this program do?

- What is the password?
- How does encrypt a string/file?
- How does the program influence other files/programs?

Reverse Engineering

$ Kinds of program to reverse

- Source file in C, python, rust, etc
- Compiled binary, mostly written in C or .NET. This is the most common.

Reverse Engineering - Methods

$ How to reverse a binary?

- Disassemble: Need to be able to read assembly
- Decompile: Difficult, few programs that do this
- Debug: Edit program flow during runtime

$ Static vs Dynamic Analysis

- Static: Analysing the code without running it
- Dynamic: Running the program, and analysing what is does.

Reverse Engineering - Tools

$ IDA Free (or Pro if you're wealthy...)

- Disassembles into assembly code.
- Nice User Interface, with plugins

$ Ghidra

- The new (scary) game changer
- Decompiles to C

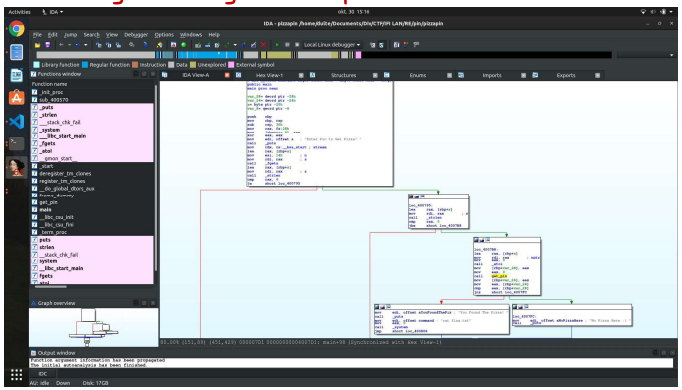
$ GDB (GNU Debugger)

- Assembly code, but can interact during runtime

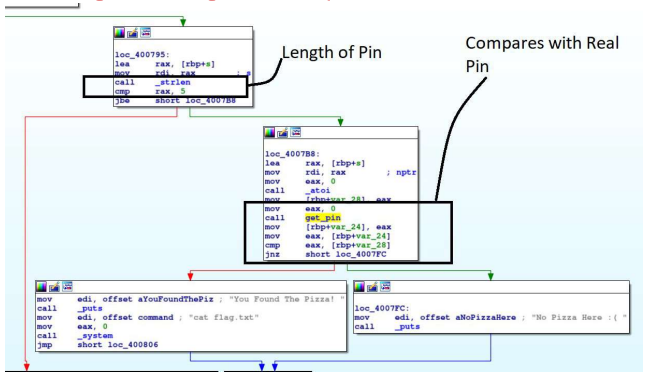
Reverse Engineering - Some Assembly

- **mov a, b;** Moves value of b into a
- **add a, b;** Adds a to b, and stores result in a.
- **sub a, b;** Subtracts a from b, and stores result in a.
- **cmp a,b;** Compares a and b
- **jne/jle add;** Jumps to address on condition
- **call func;** Calls function
- **eax, edx, etc;** Registers. Return values are stored in eax.

Reverse Engineering - Example



Reverse Engineering - Example



Reverse Engineering - Example

Reverse Engineering - Example

```
dulte@dulte-HP-Pavilion-Notebook:~/Documents/Div/CTF/IFI LAN/RE/pin/test$ ./pizz
apin
Enter Pin to Get Pizza!
[redacted]
You Found The Pizza!
UiO-CTF{random_flag}
dulte@dulte-HP-Pavilion-Notebook:~/Documents/Div/CTF/IFI LAN/RE/pin/test$
```

Cryptography

Cryptography

$ Hiding messages

- Encoding
  - Hex
  - Base64
- Encryption
  - ROT13/Caesar Cipher
  - Vigenère Cipher
  - XOr
  - RSA

Cryptography

$ Encoding

- Hex
  - 0-9 and a-f.
  - Hello -> 48 65 6c 6c 6f
- Base64
  - A-Z, a-z, 0-9 and + and /
  - Hello -> SGVsbG8=

Cryptography

$ Encrypting

- ROT/Caesar Cipher
  - Rotates the letters by some number n.
  - n = 4: a->e , b->f, etc.
- Vigenère Cipher
  - Polyalphabetic substitution
  - Requires a key. Tell how much to shift each letter

Cryptography

$ Encrypting

- XOR
  - Bitwise xor the plain text with a key
  - 1101 xor 1011 -> 0110
- RSA
  - Modern public key encryption
  - Safety from the difficulty of factorization of large numbers.
  - If care is taken, this is practically impossible to break.

Cryptography - Example

- Caesar: HvB-PGS{rg_gh_oehghf}
- Xor: MwUuSiU4JxwDFAILEx8IEQMzBwsHCxw=

Web exploitation

Web exploitation

$ Exploiting web pages

- Various programming languages
- Issues fundamental to the internet

$ Examples

- SQL Injection
- Command Injection
- Directory Traversal
- Cross Site Request Forgery (CSRF)
- Cross Site Scripting (XSS)
- Server Side Request Forgery

OWASP Top 10

https://www.question-defense.com/2019/10/07/what-are-the-owasp-top-10-vulnerabilities

OWASP Top 10 2013 -› 2017

OWASP Top 10 - 2013		OWASP Top 10 - 2017
A1 – Injection	→	A1:2017-Injection
A2 – Broken Authentication and Session Management	→	A2:2017-Broken Authentication
A3 – Cross-Site Scripting (XSS)	↘	A3:2017-Sensitive Data Exposure
A4 – Insecure Direct Object References [Merged+A7]	U	A4:2017-XML External Entities (XXE) [NEW]
A5 – Security Misconfiguration	↘	A5:2017-Broken Access Control [Merged]
A6 – Sensitive Data Exposure	↗	A6:2017-Security Misconfiguration
A7 – Missing Function Level Access Contr [Merged+A4]	U	A7:2017-Cross-Site Scripting (XSS)
A8 – Cross-Site Request Forgery (CSRF)	☒	A8:2017-Insecure Deserialization [NEW, Community]
A9 – Using Components with Known Vulnerabilities	→	A9:2017-Using Components with Known Vulnerabilities
A10 – Unvalidated Redirects and Forwards	☒	A10:2017-Insufficient Logging&Monitoring [NEW,Comm.]

https://www.checkmarx.com/2017/12/03/closer-look-owasp-top-10-application-security-risks/

Now its..

DEMO TIME!

Binary exploitation

Binary exploitation

$ pwn

- binary files
- memory corruption
  - stack overflow
  - heap overflow
  - format string bugs
  - integer overflow

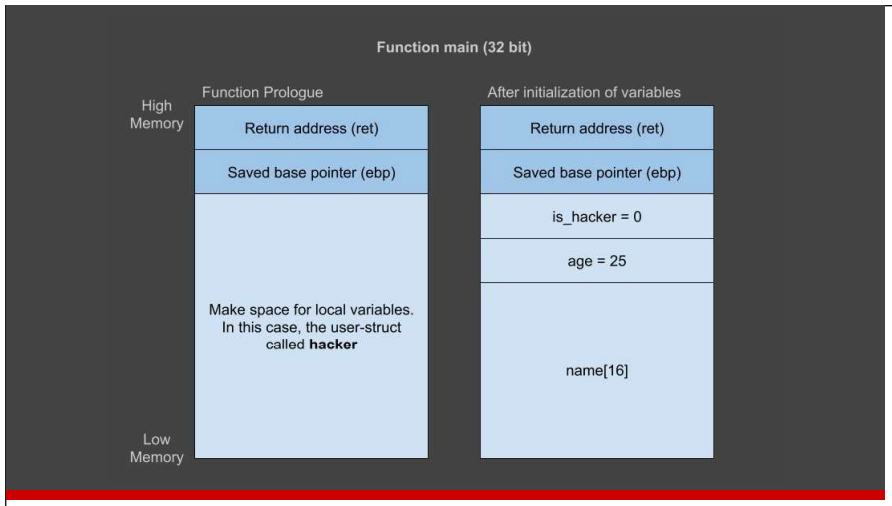
Stack overflow

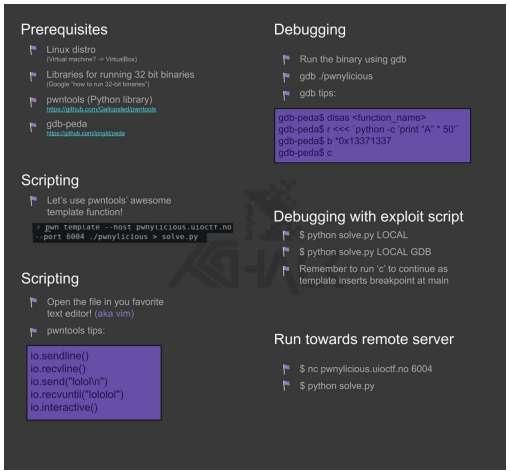
"... when a computer program tries to use more memory space than the call **stack** has available."

- https://whatis.techtarget.com/definition/stack-overflow

```c
#include <stdlib.h>
#include <stdio.h>
#include <unistd.h>
#include <stdbool.h>

struct user {
    char name[16];
    int age;
    bool is_hacker;
} __attribute__((packed));
/* the attribute makes sure that there is no padding between struct members */

int main()
{
    struct user hacker;
    hacker.is_hacker = 0;
    hacker.age = 25;

    setvbuf(stdout, NULL, _IONBF, 0);
    printf("Hey, hacker! What's your name?\n");

    read(STDIN_FILENO, hacker.name, 30);

    if (hacker.age == 25 && hacker.is_hacker == 1) {
        printf("You made it! Hackers dont have an age limit!\n");
        system("/bin/sh");
    } else {
        printf("Oh noes, you're not old enough to be a hacker!\n");
    }

     return 0;
}
```

```c
#include <stdlib.h>
#include <stdio.h>
#include <unistd.h>
#include <stdbool.h>

struct user {
    char name[16];
    int age;
    bool is_hacker;
}
/*

int main()
{
    struct user hacker;
    hacker.is_hacker = 0;
    hacker.age = 25;


    read(STDIN_FILENO, hacker.name, 30);

    if (hacker.age == 25 && hacker.is_hacker == 1) {
        printf("You made it! Hackers dont have an age limit!\n");
        system("/bin/sh");
    } else {
        printf("Oh noes, you're not old enough to be a hacker!\n");
    }

     return 0;
}
```





We need more UiO-CTFers!
Sounds fun? Join us!

Any questions? Ask us at

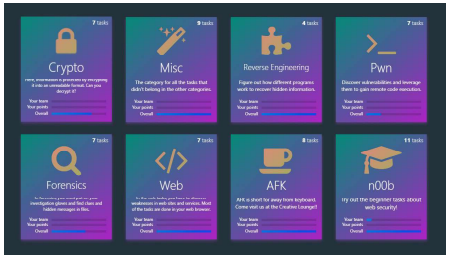
facebook.com/uioctf

Practical tasks are coming at

https://in2120.uioctf.no

Like CTFs? Checkout TG:Hack!

https://tghack.no



Sources

- https://www.vg.no/nyheter/innenriks/i/EoAnPA/skoleelev-varslet-om-datahull-i-bergen
- https://www.digi.no/artikler/politet-mener-eleven-som-varslet-om-sikkerhetshull-har-begatt-straffbar-handling/462350