CTF Primer

2013-09-24

Agenda

- CTF？
- CTF！

CTF ?

CTF?

What is CTF?

- Capture The Flag

> „Capture the Flag (CTF) contests are designed to serve as an educational exercise to give participants experience in security related real world problems“

- It‘s all about collecting „flags“
- ... and having fun!

http://net.cs.uni-bonn.de/wg/cs/staff/raphael-ernst/student-ctf-group/

CTF?

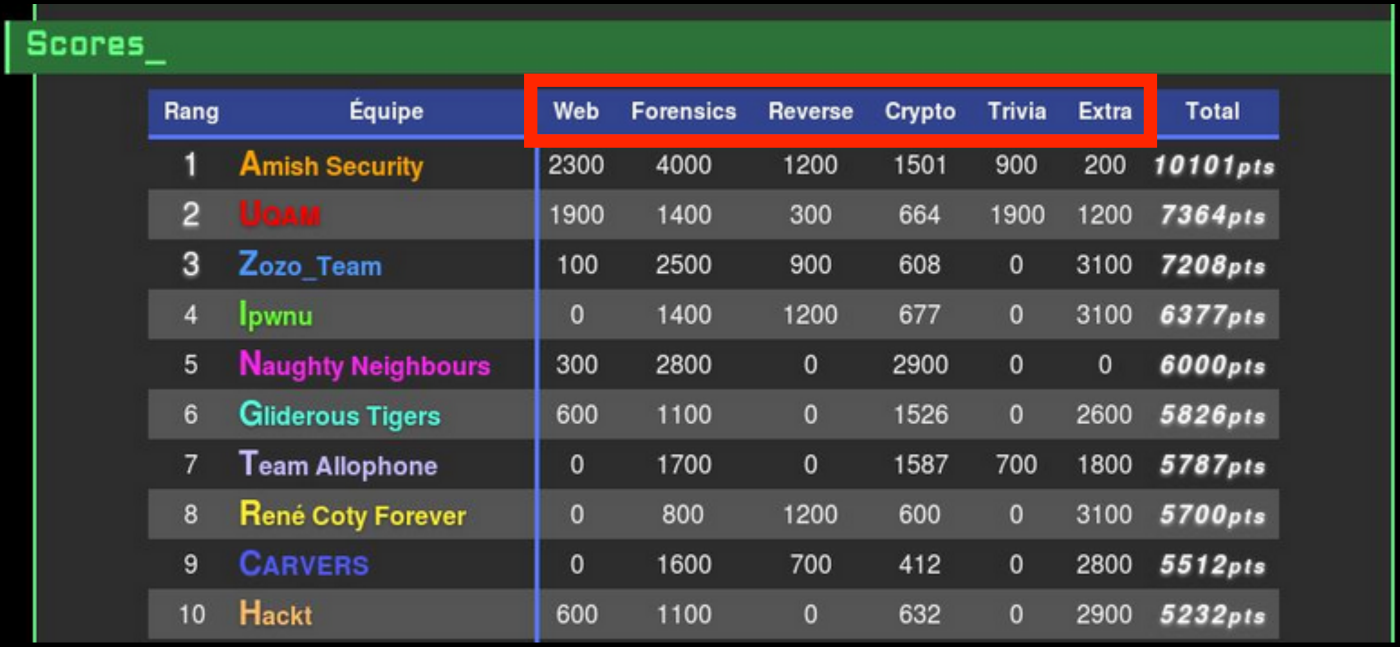
What are the Game Modes?

- Attack / Defense
  - Every participating team receives the same server image
    - Harden the services in your instance (!= port blocking)
    - Attack vulnerabilities in the other teams‘ services
  - Mostly pentesting oriented
- Hacker Jeopardy
  - Solve challenges to collect flags
    - Different disciplines
    - Various complexity levels
  - Often: First solver receives a bonus
  - More popular version

CTF?

What is a flag?

- Flag == <n> points
- Often a (prefixed) hash
    - C0ffeeb7ce3cb82a05b1b4d57e70f349
    - ^^^^ Dimva CTF, misc100_2

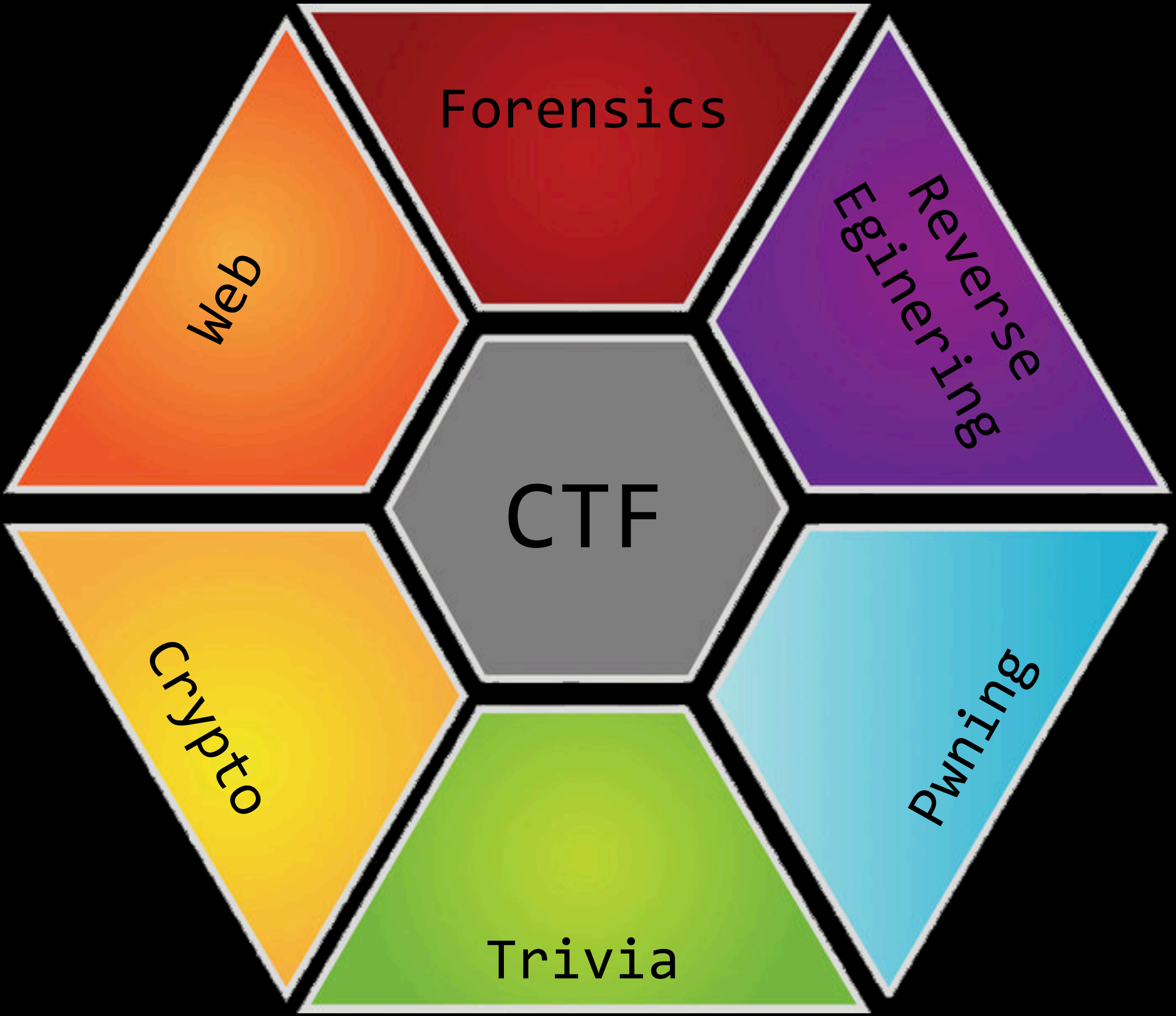


Scores_

Rang	Équipe	Web	Forensics	Reverse	Crypto	Trivia	Extra	Total
1	Amish Security	2300	4000	1200	1501	900	200	10101pts
2	UQAM	1900	1400	300	664	1900	1200	7364pts
3	Zozo_Team	100	2500	900	608	0	3100	7208pts
4	Ipwnu	0	1400	1200	677	0	3100	6377pts
5	Naughty Neighbours	300	2800	0	2900	0	0	6000pts
6	Gliderous Tigers	600	1100	0	1526	0	2600	5826pts
7	Team Allophone	0	1700	0	1587	700	1800	5787pts
8	René Coty Forever	0	800	1200	600	0	3100	5700pts
9	CARVERS	0	1600	700	412	0	2800	5512pts
10	Hackt	600	1100	0	632	0	2900	5232pts

CTF?

What are these disciplines?



CTF?

What are these disciplines?

- Web
    - (In)Security of web applications
        - SQL injection
        - directory traversal
        - exploiting logical flaws
        - ...
- Forensics
    - Needle in a haystack
    - „Broken“ data / files
    - Image recovery

CTF?

What are these disciplines?

- Reverse Engineering
  - Understanding algorithms
  - Defeating anti-analysis mechanisms
  - „crackmes“
- Crypto
  - Implementation flaws
  - Efficient attacks

CTF?

What are these disciplines?

- Pwning
  - Exploitation of services
  - Remote Code Execution (RCE)
  - Finding the vector + planting a shell
- Trivia
  - Guessing / inference
  - Recognizing network protocols, file formats, ...

CTF！

CTF!

Why should I play?

- Meet nice people & have a good time :)
- Great opportunity to learn
- Extend your horizon
- Fame & Glory

CTF!

Skills needed.

- Creativity!
- Endurance!!
- Communication!

- Previous knowledge welcome :)
    - Coding + Reversing

CTF!

Infrastructure.

- We can provide:
  - Locality
  - Infrastructure (Net access, some boxes)
  - Some experience in CTFs

CTF!

Our First Gig!

- Hack.lu CTF
  - 22nd – 24th October 2013
    - Most likely 11am – 11am (as in previous years)
  - Hacker Jeopardy, run by FluxFingers
  - Lots of challenges with varying difficulty

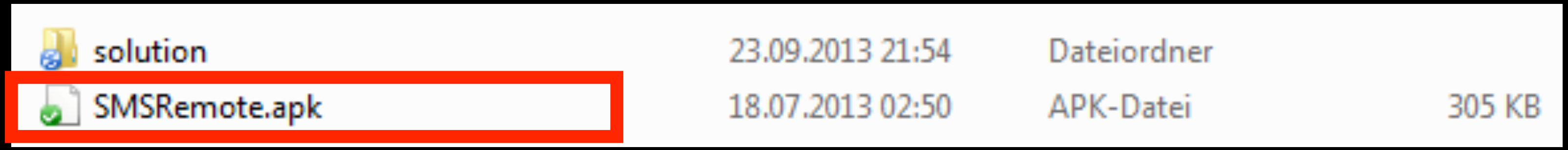
Teaser!

A challenge example

CTF!

A challenge example.

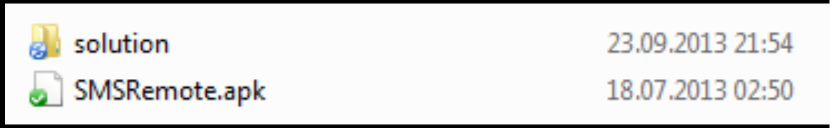
- Dimva CTF, misc100_2



solution	23.09.2013 21:54	Dateiordner	
SMSRemote.apk	18.07.2013 02:50	APK-Datei	305 KB

CTF!

A challenge example.



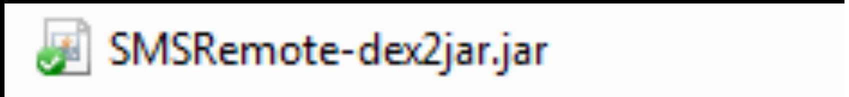
solution	23.09.2013 21:54
SMSRemote.apk	18.07.2013 02:50

- APK
  - (Android) application package format
  - Zip-file / *.jar
  - Contains Dalvik VM code (*.dex)
- DEX2JAR!

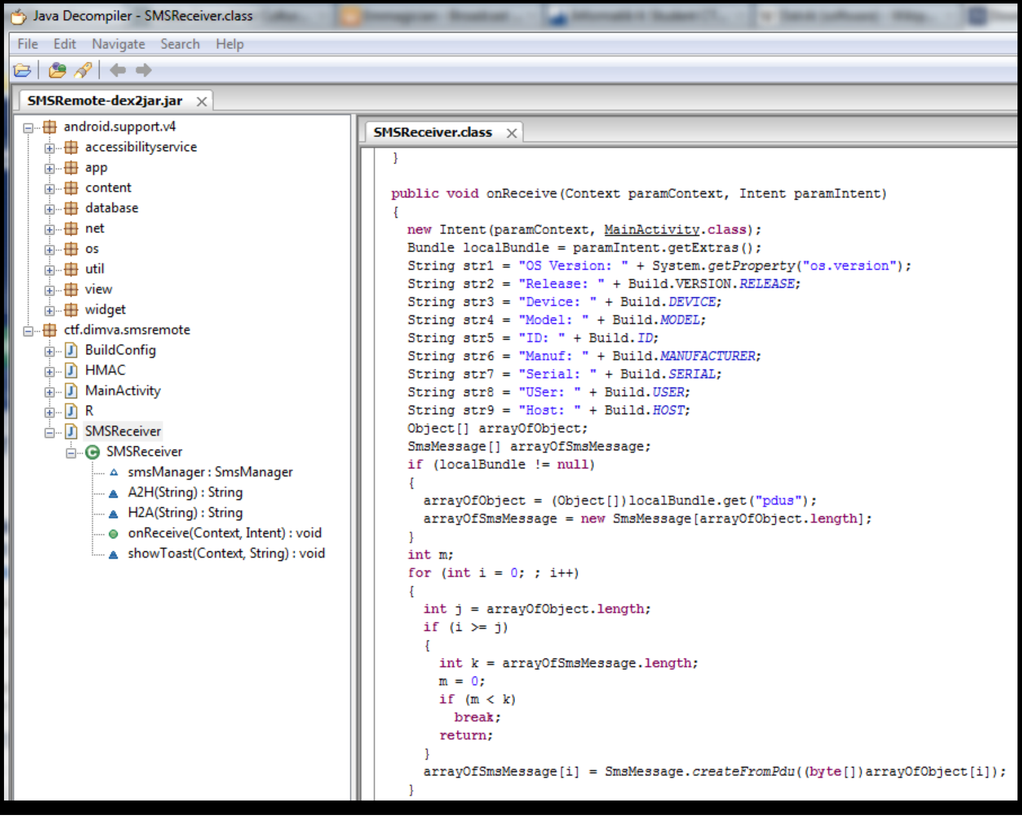
SMSRemote-dex2jar.jar	18.07.2013 03:01

CTF!

A challenge example.



- JAR = Java?
  - Yes, but only (bytecode-compiled) class files
- JDGUI! (Java Decompiler)



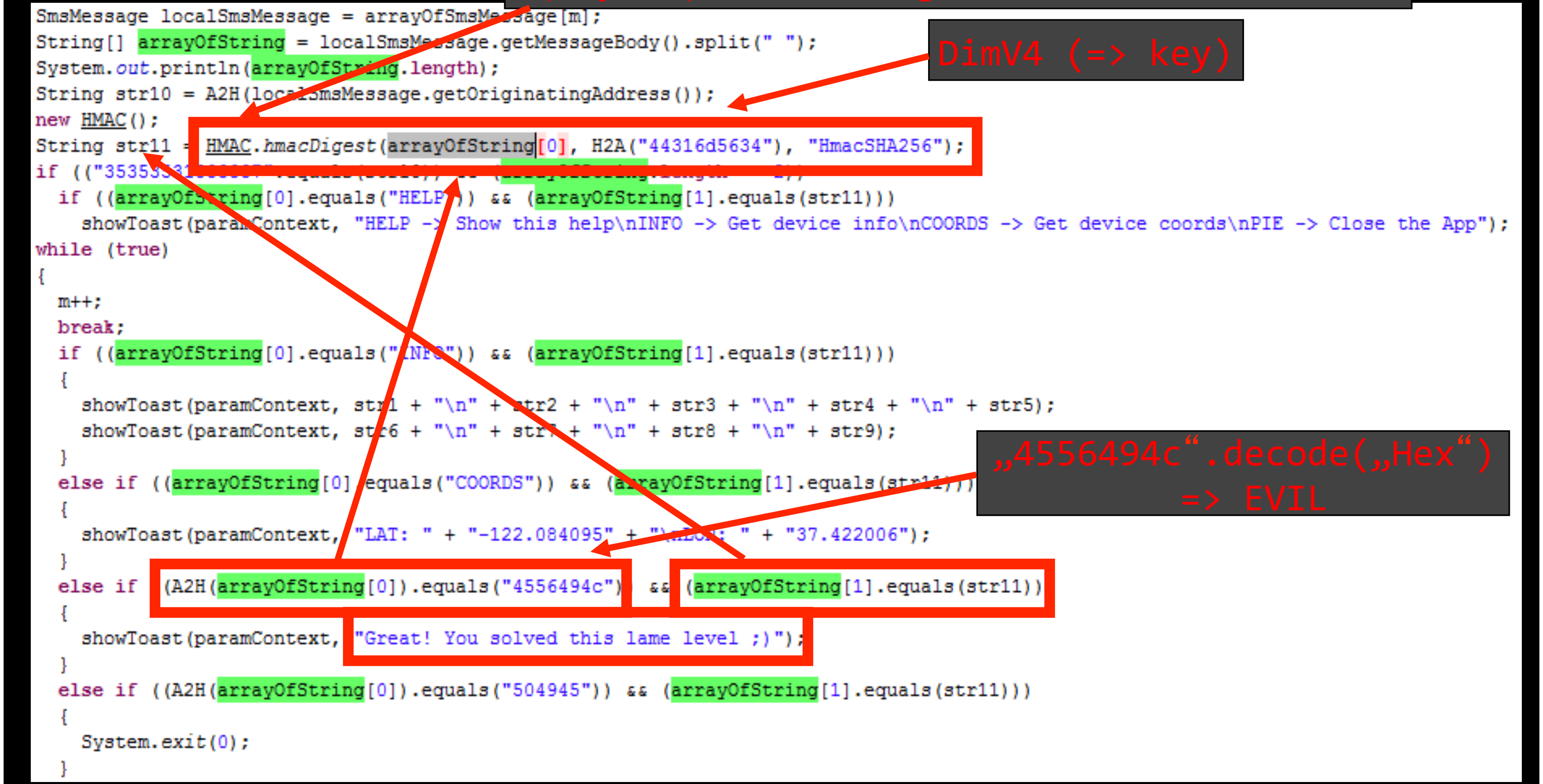
```java
}

public void onReceive(Context paramContext, Intent paramIntent)
{
  new Intent(paramContext, MainActivity.class);
  Bundle localBundle = paramIntent.getExtras();
  String str1 = "OS Version: " + System.getProperty("os.version");
  String str2 = "Release: " + Build.VERSION.RELEASE;
  String str3 = "Device: " + Build.DEVICE;
  String str4 = "Model: " + Build.MODEL;
  String str5 = "ID: " + Build.ID;
  String str6 = "Manuf: " + Build.MANUFACTURER;
  String str7 = "Serial: " + Build.SERIAL;
  String str8 = "USer: " + Build.USER;
  String str9 = "Host: " + Build.HOST;
  Object[] arrayOfObject;
  SmsMessage[] arrayOfSmsMessage;
  if (localBundle != null)
  {
    arrayOfObject = (Object[])localBundle.get("pdus");
    arrayOfSmsMessage = new SmsMessage[arrayOfObject.length];
  }
  int m;
  for (int i = 0; ; i++)
  {
    int j = arrayOfObject.length;
    if (i >= j)
    {
      int k = arrayOfSmsMessage.length;
      m = 0;
      if (m < k)
        break;
      return;
    }
    arrayOfSmsMessage[i] = SmsMessage.createFromPdu((byte[])arrayOfObject[i]);
  }
```

Trigger warning: Powerpoint madness incoming!

CTF!

A challenge example.



```
SmsMessage localSmsMessage = arrayOfSmsMessage[m];
String[] arrayOfString = localSmsMessage.getMessageBody().split(" ");
System.out.println(arrayOfString.length);
String str10 = A2H(localSmsMessage.getOriginatingAddress());
new HMAC();
String str11 = HMAC.hmacDigest(arrayOfString[0], H2A("44316d5634"), "HmacSHA256");
if (("35353[illegible]").equals(str10)) && (arrayOfString.length == 2))
  if ((arrayOfString[0].equals("HELP")) && (arrayOfString[1].equals(str11)))
    showToast(paramContext, "HELP -> Show this help\nINFO -> Get device info\nCOORDS -> Get device coords\nPIE -> Close the App");
while (true)
{
  m++;
  break;
  if ((arrayOfString[0].equals("INFO")) && (arrayOfString[1].equals(str11)))
  {
    showToast(paramContext, str1 + "\n" + str2 + "\n" + str3 + "\n" + str4 + "\n" + str5);
    showToast(paramContext, str6 + "\n" + str7 + "\n" + str8 + "\n" + str9);
  }
  else if ((arrayOfString[0].equals("COORDS")) && (arrayOfString[1].equals(str11)))
  {
    showToast(paramContext, "LAT: " + "-122.084095" + "\nLON: " + "37.422006");
  }
  else if ((A2H(arrayOfString[0]).equals("4556494c")) && (arrayOfString[1].equals(str11)))
  {
    showToast(paramContext, "Great! You solved this lame level ;)");
  }
  else if ((A2H(arrayOfString[0]).equals("504945")) && (arrayOfString[1].equals(str11)))
  {
    System.exit(0);
  }
```

(Keyed-)Hash Message Authentication Code

DimV4 (=> key)

„4556494c".decode(„Hex")
=> EVIL

CTF!

A challenge example.

☼ Solution (Python):

```
import hmac
import hashlib

key = „D1mV4“
data = „EVIL“
digestmod = hashlib.sha256

hashed = hmac.new(key, data, digestmod).digest()
result = hashed.encode(„base64“)
print result
```

```
Result: vqwYfONxkebWk4SUsmpQenN6ik3uvvoJw3/oq7hQQYg=
```

```
$ cat result | netcat dimvactf.0x90.eu 5555
C0ffeeb7ce3cb82a05b1b4d57e70f349
```

Conclusion

Final words to wrap things up

CTF

Conclusion.

- Student CTF group
  - It‘s about you! We only offer the framework
- Expected Skills:
  - Motivation! Rest will come by itself!
- Time commitment:
  - One meeting per week + optional „homework“ + CTF participation
- Homework?
  - Voluntary tasks from old CTFs as practice
- Long-Run:
  - A platform to publish

CTF

Resources.

- CTFtime.org
  - Archive / Scores / Write-Ups
- Captf.com/practice-ctf
  - Collection of write-ups, challenges, etc.
- Blog.dragonsector.pl
  - Very active write-up blog of a leading team

CTF

Homework! :)

- Dimva CTF, crypto100
- A bunch of images
  - ... that look the same
  - ... but have a secret
- Hints:

α

https://dl.dropboxusercontent.com/u/1346415/dimva_crypto100.zip