Grenoble INP **UGA**
Ensimag **UFR IM2AG**

M2 Cybersecurity

Advanced Security

Practical exercises on binary code Reverse Engineering

Download the examples provided on the Moodle page for this Lab.
They consist in 4 « CTF like » challenges, sorted by increasing difficulty.

You can execute/solve them directly on the Ensimag machines using no more tools than objdump, gdb en Ghidra.

Chal1: crakme0 (x86) [very very easy]

The goal is to get a congratulation message …

Easy to obtain if we decompile this code with Ghidra and look at the expected input string (available in plain text).

Chal2: crakme2 (x86) [similar to Chal2, but a bit more difficult]

The goal is to get a congratulation message …

The expected input string is not readily available in plain text in the excecutable code but slightly transformed. Using Ghidra we can easily retrieve the tranformation algorithm (and write a small program to reproduce its behavior).

Chal2: crackme1 (x86_64) [easy]

The goal is to get the magic word printed on the screen …

Similar than crackme2, but the transformation algorithm is a bit more complex.
You can skip this example if your are late ...

A break: foo (x86) [something different to take a break …]

- Run this program
- Load it into gdb and put a breakpoint at function foo (`b foo`)
- run it and disassemble foo when the breakpoint is hit (`disass`)
- continue the execution (`continue`) until the breakpoint is hit again
- disassemble foo, what do you observe?

Chal4: crakme3 (x86) [difficult!!]

The goal is to find the correct password ...