Hardware challenges

Some advices to carefully read and follow!

- Manipulate GENTLY the boards: everything is hand soldered and somehow fragile
- The challenge is self-contained, no need for extra complex tools
- You'll need a logic analyser. Go to the desk, we'll lend one
- Never short-cut the VCC and the GND
- If you are not sure of what you are doing when adding wires, just ask
- No components have lock fuse, everything is dumpable (and should be!)
- NEVER set a lock fuse protection on the microcontroller's flash, it'll kill the game
- Dump everything before starting serious things
- Official Arduino IDE have been used for building the challenges
- No stegano, weird puzzle etc.
- Everything written in the subject is important and TRUE, no guessing (or not that much :))
- The fuse for ATtiny are l:0xE2 h:0xDF e:0xFF
- You need to dump the Arduino and ATtiny if you program or patch them
- Mandatory: Before you start to solve a challenge the board needs to be in the initial state
- "Jumper" is not needed, but it can gives you a valuable hint

The goals of all those challenges:

The main purpose of this year's hardware challenges is to show that everyone can do it. The first 3 challenges are solvable by anyone in a CTF context. Please keep in mind that the staff will help you if the task is not clear or if you think you'll fry the board. If you think you've broken something, erased the components, just please go to desk, we'll reload software inside for a beer.

Challenge 1: Power up!
Required tools: none
Sub category: hardware
This first step requires you to assemble the board. Don't be afraid: no soldering is needed. Just follow carefully the indication for inserting the Arduino the RIGHT WAY in its socket! And power it up to get the flag. See picture included.

Challenge 2: Logic analyser
Required tools: logic analyser, sigrok (or other), multimeter
Sub category: hardware
Now, you need to intercept your first signals on an electronic board. The flag is hidden in the 24C64 memory. During the main screen, the flag is periodically polled by the Arduino. You need to figure out how to connect a logic analyser to the board and where to connect it. You need to start by connecting the GND wire from the logic analyzer to the board. Then you need to run Sigrok, acquire the signals. Finally you must use the available protocols analysis functions to decode the signals for you. You'll have to isolate the correct component as the memory is not the only one shouting on the bus.

Challenge 3: Logic analyser - round 2
Required tools: logic analyser, sigrok (or other), multimeter
Sub category: hardware
This is the same task as challenge 2, but this time you'll need to do it on an unknown component, labelled "?????" on the board. You'll need to read the signals by yourself once it's acquired by the logic analyser. The flag is polled up periodically by the ATtiny.

Challenge 4: Dump it
Required tool: Arduino development environment
Sub category: hardware, coding
Extract the full content of the 24C64. You'll need to program the Arduino by yourself to achieve this task. Then figure out what to do with the dump. WARNING: save the content of the Arduino before flashing it!

Challenge 5: https://www.imdb.com/title/tt0091949/
Required tool: R2, Ghidra etc.
Sub category: hardware, reverse engineering OR guessing :-)
The board has a hidden mode at start-up. Find it and get your flag! You must reverse the firmware and find how to enable it. If you want to get it the guessing way it's also possible, just by looking at the board.

Challenge 6: Play with the I/O
Required tool: R2, Ghidra etc.
Sub category: hardware, reverse engineering
Put the right things on the I/O pin-header and get your flag.

Challenge 7: Serious things
Required tool: R2, Ghidra etc.
Sub category: hardware, reverse engineering
The ATtiny read & hash 32 bytes from the unknown component. What happened if you patch it and go for 64 bytes? Do it and get your flag. A little hint: no need to remove the ATtiny85 from the board, you have everything required, just figure out how.

Challenge 8: Other serious things
Required tool: R2, Ghidra etc.
Sub category: hardware, reverse engineering
There is a hidden firmware somewhere. Find it, run it and guess what it does and ... what to do. What? Guessing? No...

Credits:

All artworks, nice looking PCB and onscreen bitmaps by Lex (@superlexsec)
Electronics stuff by Phil (@PagetPhil) with the help of dok (@dokthar)
Sweat shop during a full day by Florent (@ZeNetPlumber), Jeanmi, dok, Dirval and Phil

Big thanks to:

Joe Grand (@joegrand) for his valuable advices on how to build a conference badge
Nicolas Oberli (@Baldanos) for all his good ideas and nice hardware trixxx
Marc Olanié (@marcolanie) for always sharing his valuable knowledge without counting his time
Larry Bank (@fast_code_r_us) for his nice ss_OLED library and all the last minute patches asked