Tales of Cheap Duckies

Combining
Digispark Attiny 85,
TOR and
the Empire Framework

About
Me

Davide Meacci

Penetration tester, Ethical Hacker
Student at Sapienza for 5 years (Bachelor and Master degrees), external collaborator at CINI for more than 1 year
Now (?)
CTF and security challenges lover

DEFCON Rome

What the presentation IS about

Basics of micro controllers (MCU)

Basics of rubber duckies

Short rehearsal on TOR and hidden services

Short introduction to the EMPIRE framework

How integrating rubber duckies, TOR and the EMPIRE framework through a case study

What the presentation IS NOT about

How to hack in anyone’s computer

How to get better grades at University

How to disrupt private servers and public websites

How to get free goods on Amazon

Disclaimer

The contents of this talk are potentially

D A N G E R O U S

I do **not** take responsibility for your own actions

Micro Controller Units (MCU)

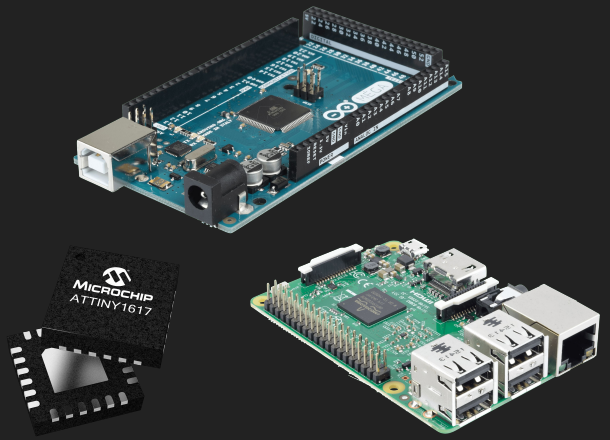
A MCU is a small computer on a single integrated circuit.

contains one or more CPUs (processor cores) along with memory and programmable input/output peripherals.

used in automatically controlled products and devices

Advantages: high integration, flexibility, easy to program, fast

Disadvantages: sometimes expensive, complex internal structure, challenging programming



Rubber Duckies

A Rubber Ducky is an input injection tool disguised as a generic flash drive (thanks to USB descriptors)

computers recognize it as a regular device and accept pre-programmed input payloads.

can be used to drop reverse shells, inject binaries, brute force pin codes, and more.

Sold by hak5 gear shop, but can we do better than that?

YES !

How It works

Write

Write payloads in a simple scripting language called DUCKYSCRIPT

Encode

Port the script to different platforms, depending on the MCU you are going to use

Load

Upload the script to your MCU

Deploy

Fire up your code by inserting your MCU into any available USB port

Everyone hates Teemo!

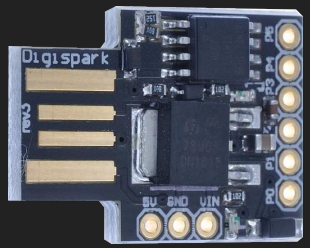
Our ducky:
Digispark Attiny 85

The Digispark Attiny 85 is a small development board that measures only about 18x18 mm;

similar to the one mounted on the Arduino series, but cheaper, smaller and a little less powerful.

can be extended by extended by many shields, and programmed through the Arduino IDE,

It s perfect when, for example, an Arduino UNO is too big (or too expensive)



D5/A0
D4/PWM4/A2/USB
D3/A3/USB+
D2/A1/SCK/SCL
D1/PWM1/MISO
D0/PWM0/AREF/MOSI/SDA

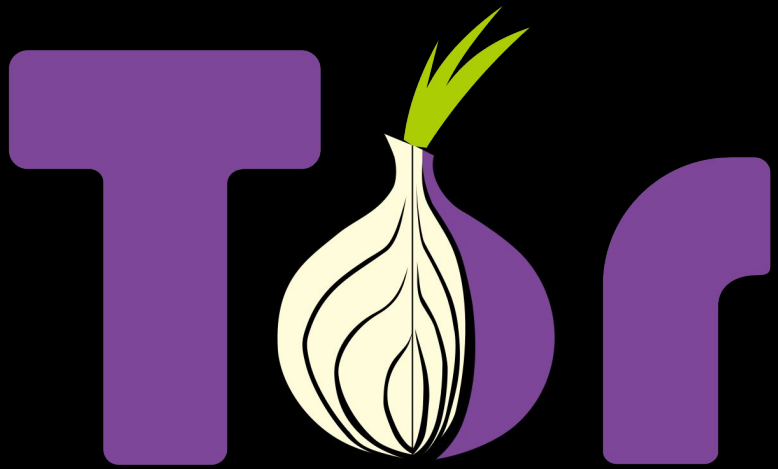
D: Digital Read/Write
A: Analog Read (ADC)
PWM: Analog Write
MOSI/MISO/SCK: SPI
SDA/SCL: I2C
USB+: USB Interface
AREF: Analog Reference

Concerns on C&C interactions

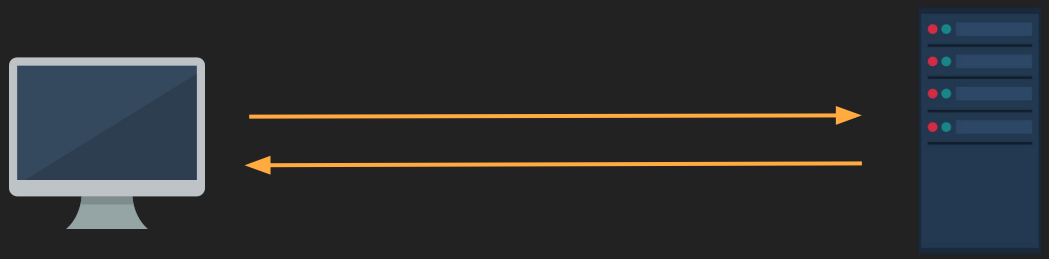
Concerns on C&C interactions

- Is it encrypted?
- Is it traceable?
- Is it anonymous?
- Is it secure to other attacks?
- … more

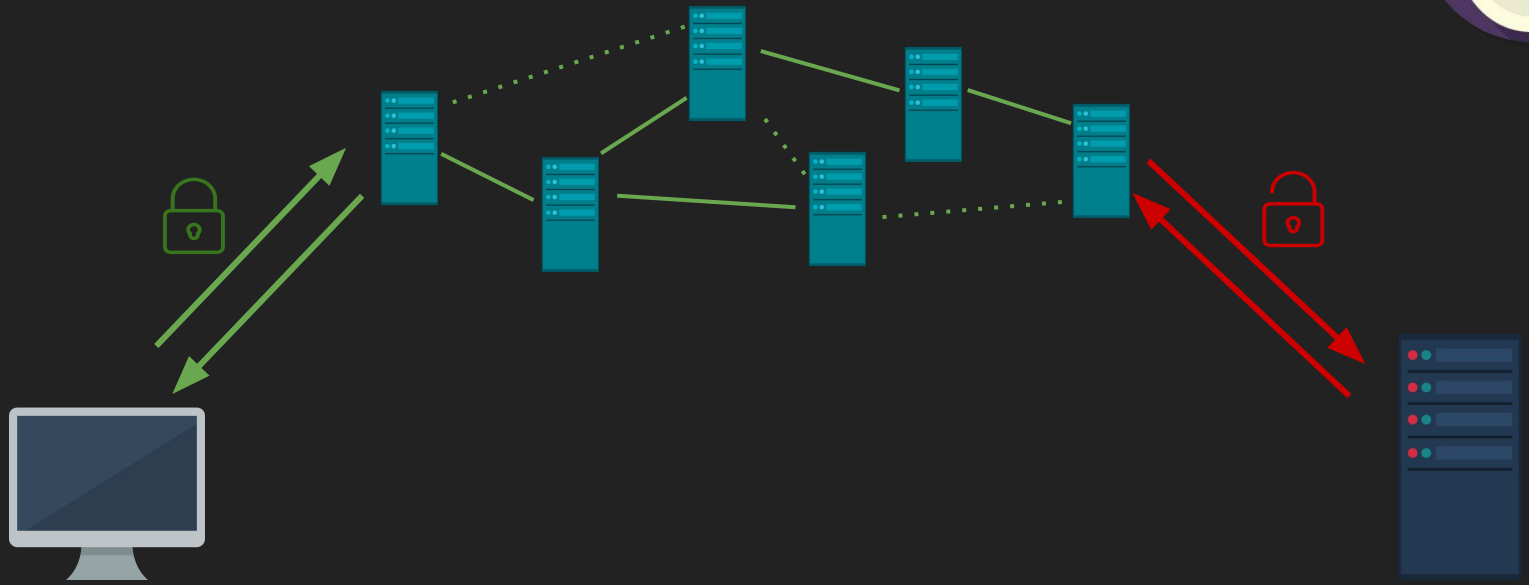
A possible solution:



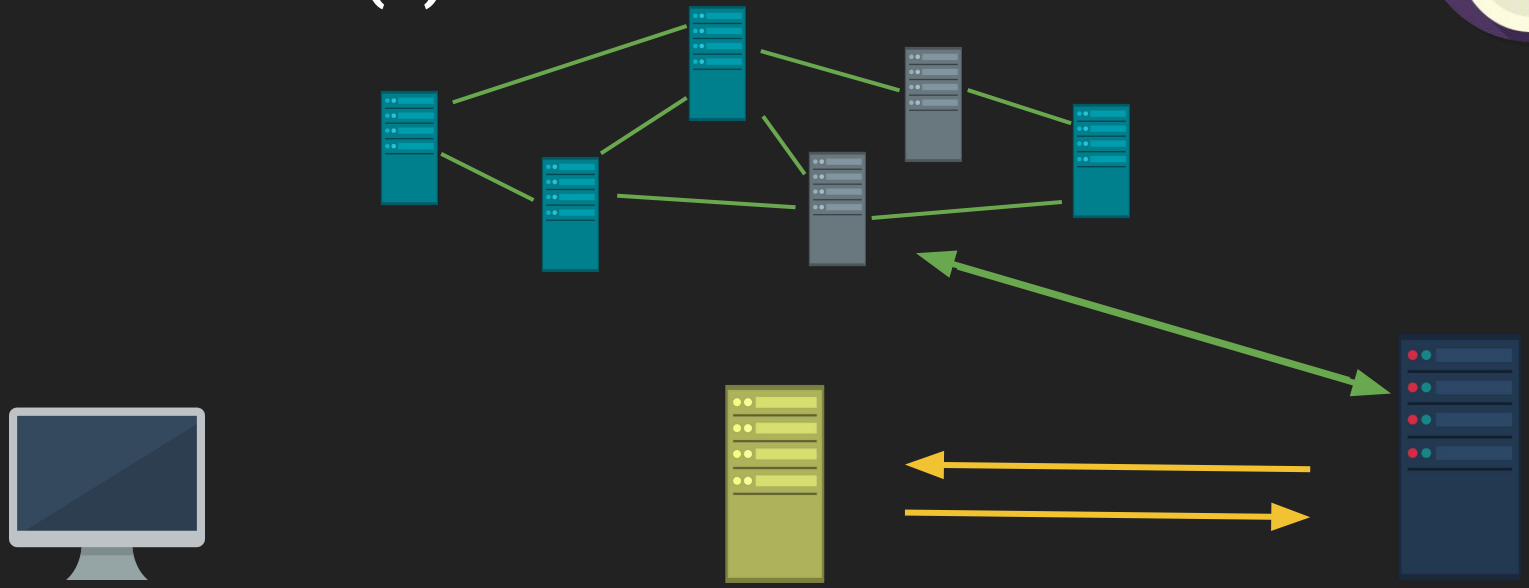
Refreshing the Tor Network



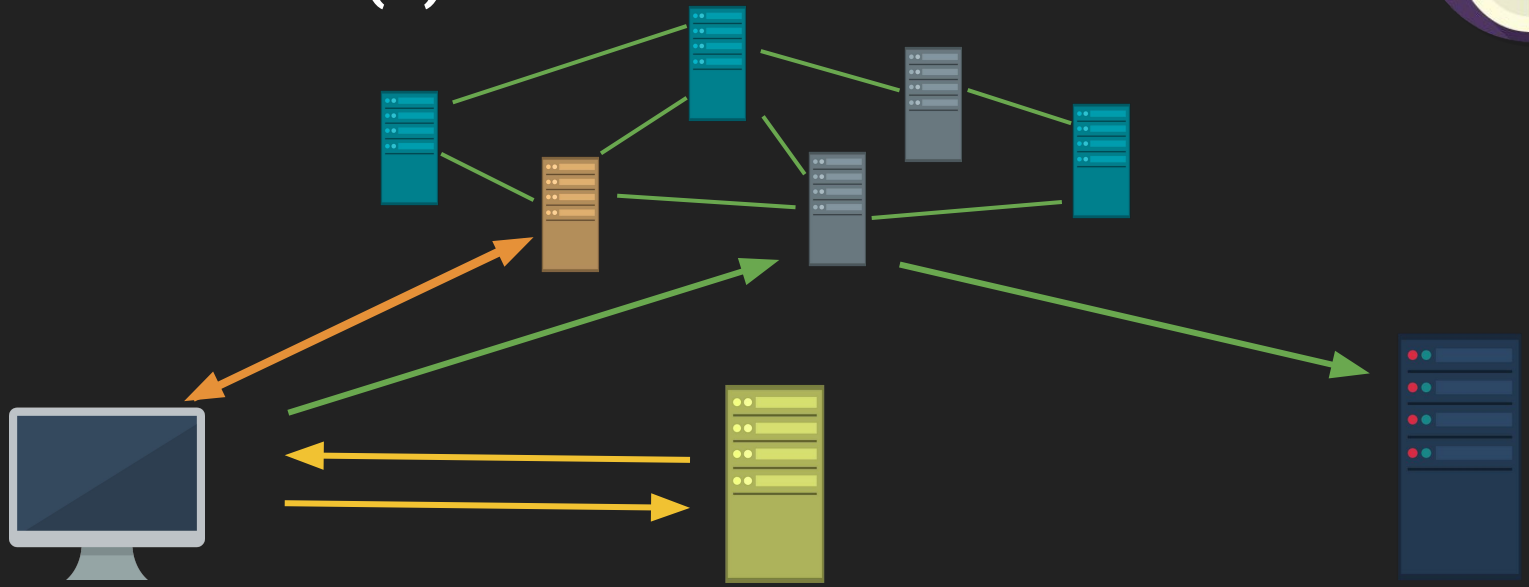
Refreshing the Tor Network



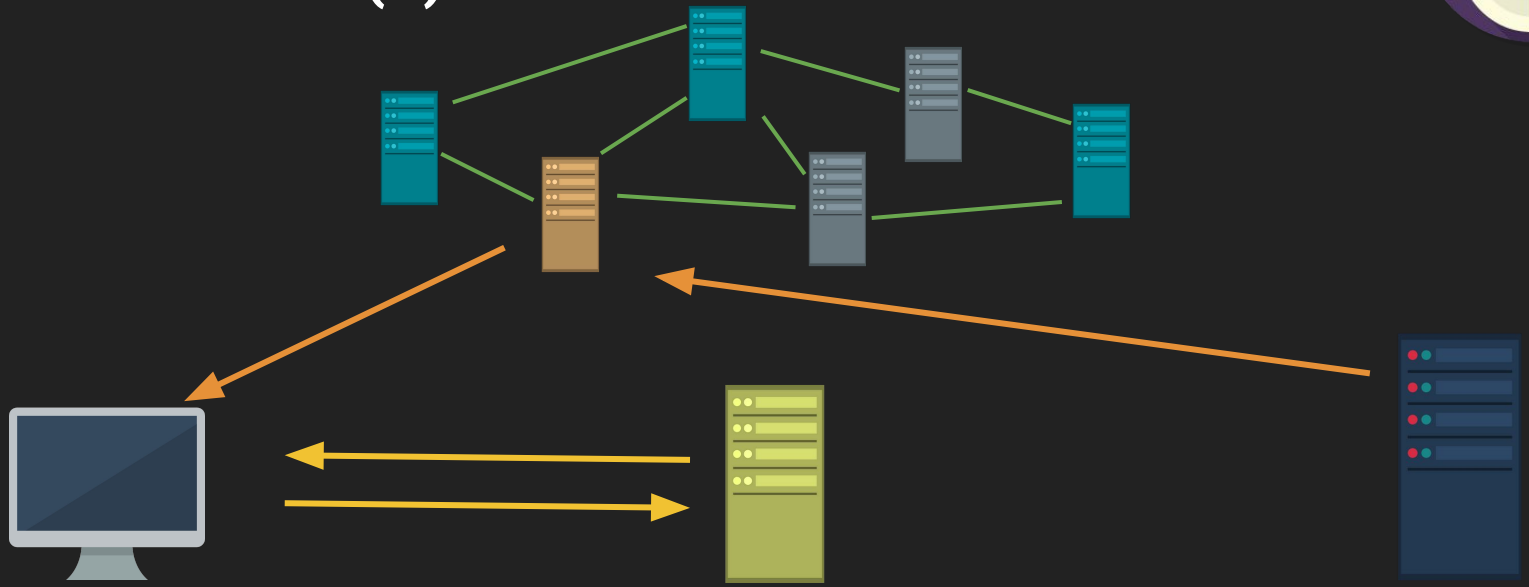
Refreshing the Tor Network (2)



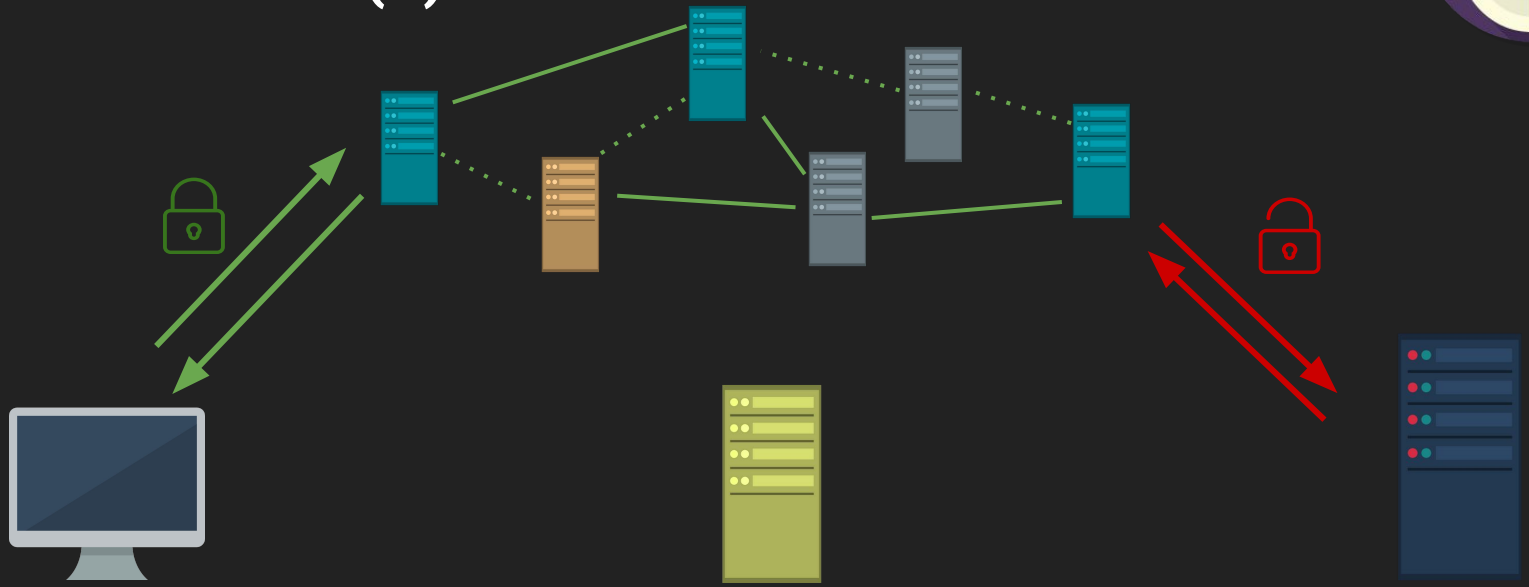
Refreshing the Tor Network (2)



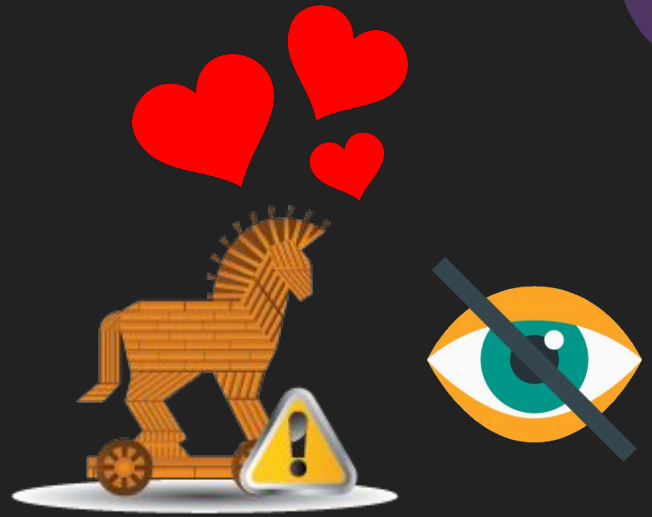
Refreshing the Tor Network (2)



Refreshing the Tor Network (2)



Tor Network and C&C interactions



Given our ducky, why not create a C&C server as an hidden service?

This would give us anonimity, as well as avoiding country restrictions, traffic analysis, etc.

Problem: we need to install TOR on the attacked host (as many other malwares)

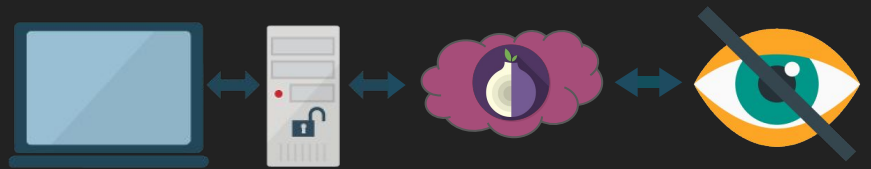
… don’t we?

Tor Network and C&C interactions (2)

We do **not** actually.

Thanks to proxying service as TOR2WEB, we can access .onion websites, web apps and services without the need of having the TOR software installed!

Some concerns (end communication is not inside TOR, URL filtering, etc.) but it works!



Hands on: the Empire Framework

Empire is a post-exploitation framework that includes a pure-PowerShell2.0 Windows agent, and a pure Python 2.6/2.7 Linux/OS X agent.

Offers cryptologically-secure communications and a flexible architecture

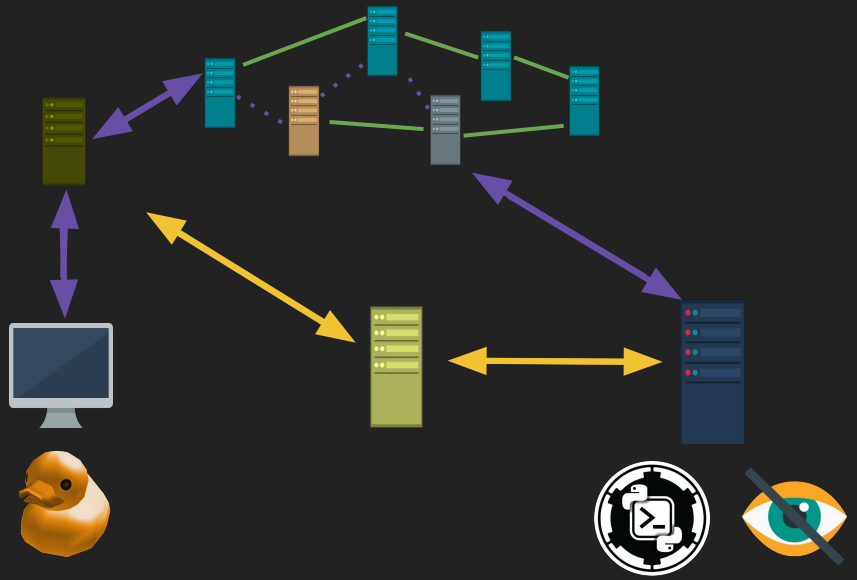
Rapidly deployable post-exploitation modules ranging from key loggers to Mimikatz, and adaptable communications to evade network detection, all wrapped up in a usability-focused framework

Hands on: the Empire Framework (2)

By using the framework, we can set up an Empire listener as a hidden service

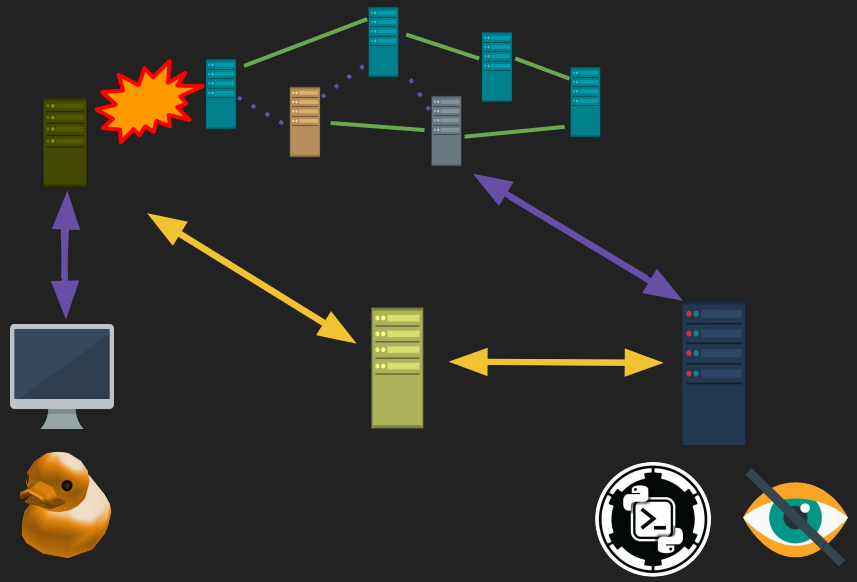
Then our ducky will connect to it through the TOR network without TOR software, thanks to TOR2WEB relays

Our C&C communication is now **encrypted** and **anonymous**.



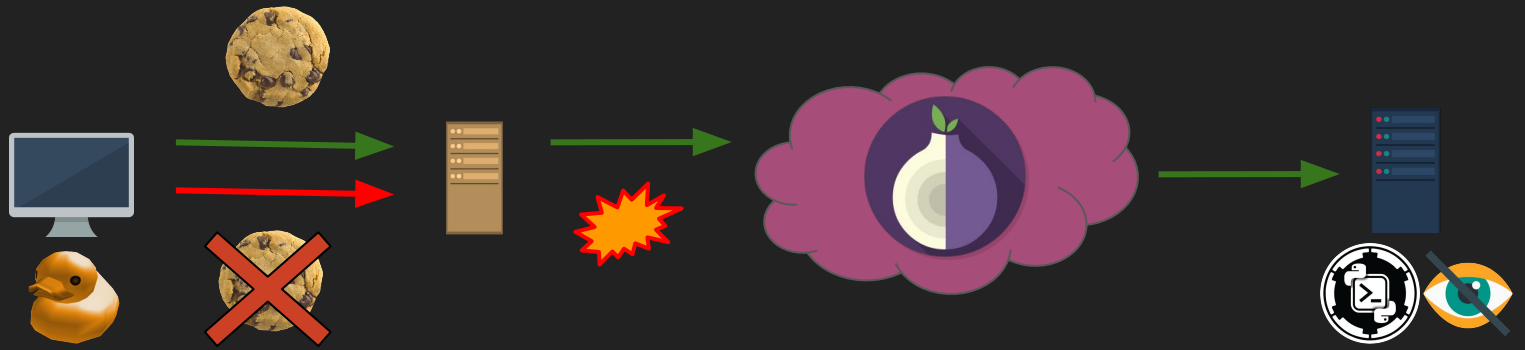
Wait, is it working or not?

Wait, is it working or not?



Nope, let's find out what happened...

Wait, is it working or not?

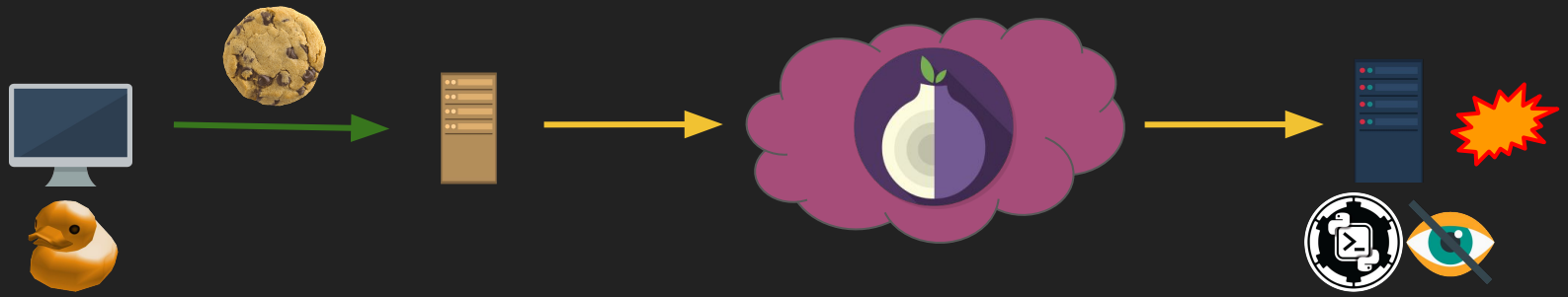


Wait, is it working or not?

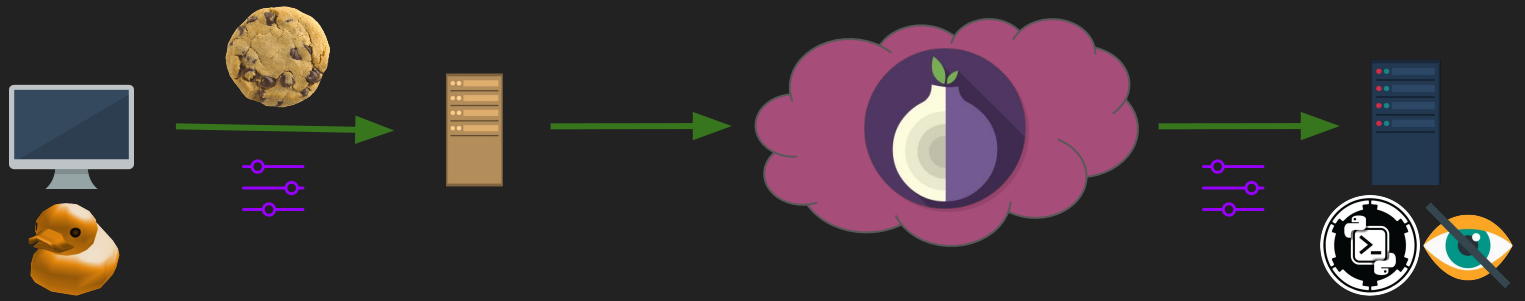
9 data/agent/stagers/http.py — View

```
@@ -48,12 +48,9 @@ def post_message(uri, data):
48 48      try:
49 49          headerKey = headerRaw.split(":")[0]
50 50          headerValue = headerRaw.split(":")[1]
51    -         if headerKey.lower() == "cookie":
52    -             headers['Cookie'] = "%s;%s" % (headers['Cookie'], headerValue)
53    -         else:
54    -             headers[headerKey] = headerValue
55    -     except:
56    -         pass
   51 +         headers[headerKey] = headerValue
   52 +     except Exception as e:
   53 +         print e
57 54
58 55  # stage 3 of negotiation -> client generates DH key, and POSTs HMAC(AESn(PUBc)) back to server
59 56  clientPub = DiffieHellman()
```

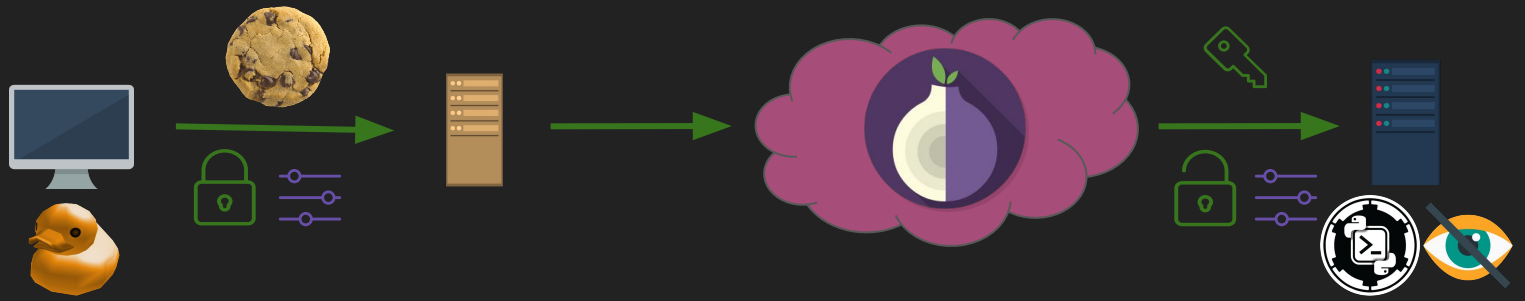
Wait, is it working or not? (2)



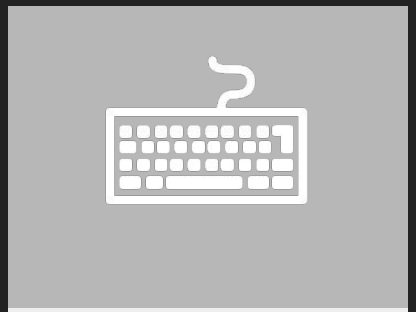
Wait, is it working or not? (2)



Yes, it FINALLY works!
(Python pre-installed distros)

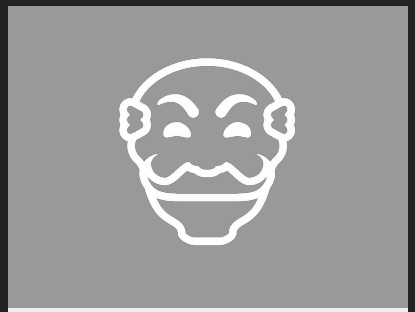


Results and wrapping up



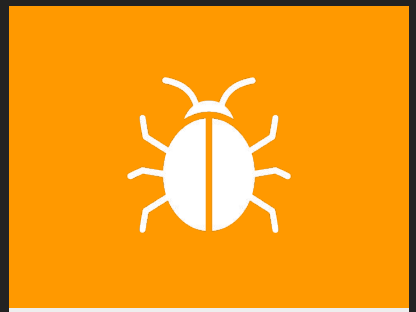
Automated Payload injection

By the use of the Digispark Attiny 85 MCU



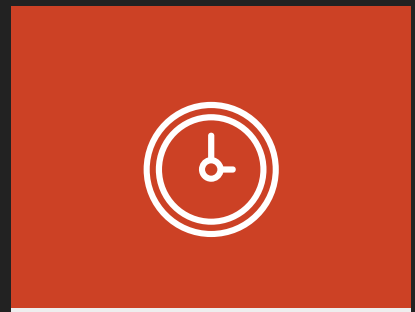
Anonymous C&C

Through the use of the TOR and Empire Frameworks



Bug tracking

By analyzing C&C interactions between Empire's clients and servers



… what's next?

Extend to Windows with Powershell code review

Mobile App remote control

BREAK TIME! Questions?

Demo

Demo (2)

The basics of DuckyScript

Getting .ino files and loading scripts

How to fire it up

DEFCON Rome

Thank You!

Feel free to stop by and have a chat :-)

DEFCON Rome

Useful links:

Rubber duckies from hak5:
https://hakshop.com/products/usb-rubber-ducky-deluxe
Digispark Attiny85 chipset:
https://www.adrirobot.it/arduino/digispark/digispark.htm
(ITA)
The Empire Framework:
https://github.com/EmpireProject/Empire

Tor2web: https://www.tor2web.org/
Issue about cookie on Empire framework:
https://github.com/EmpireProject/Empire/issues/1142
DuckyScript quickstart:
https://github.com/hak5darren/USB-Rubber-Ducky/wiki/Duckyscript
Duckyscript porting module:
https://github.com/hak5darren/USB-Rubber-Ducky/wiki/Downloads