

DFIR & malware analysis setup

HelSec 2021-01-07T161500





Ex's

elisa

nixu

■ ~\$ whoami

Juho "whois" Jauhiainen

30-years-old // Father of three

BEng, almost MSc in Tech.

CISSP, OSCP, GREM, GCFA, GMON

~7 years in the field (DFIR)

TRAFICOM

>HelSec



KyberVPK
COMMUNITY CYBER
RESPONSE FORCE



@JuhoJauhiainen

Disclaimers

This presentation **does not** represent views of my employer

This presentation presents stuff that can harm your [and others'] computer[s] → Repeat on **your own** responsibility



What to expect?

This presentation is about my setup and tooling
NOT how to use them

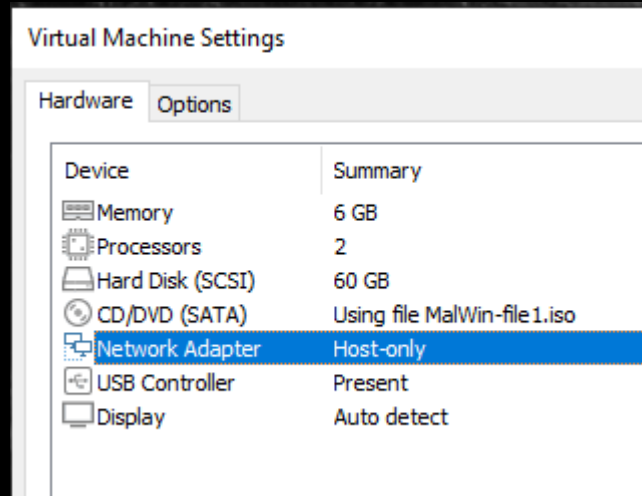


Adapt



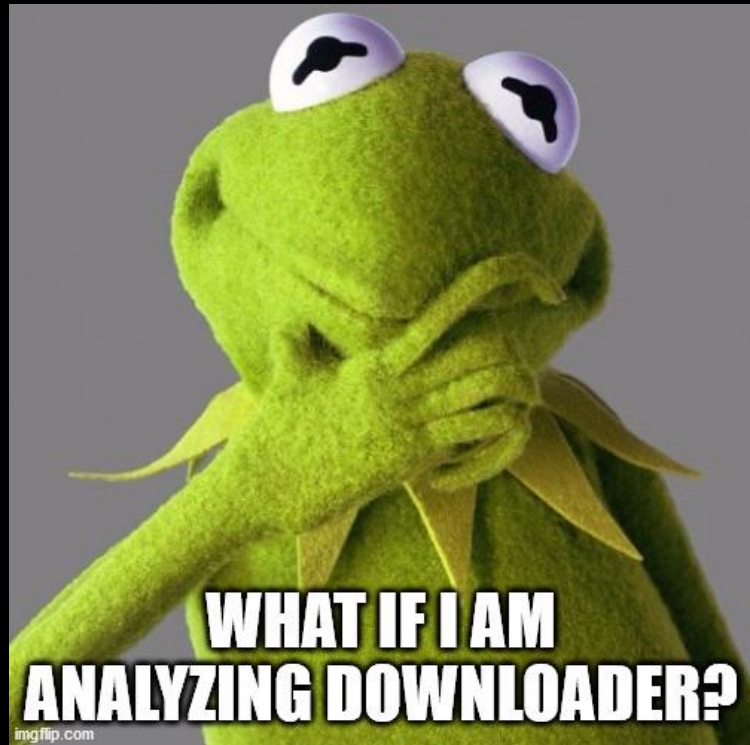
Basic setup

- » VMWare workstation
 - » You can use Vbox, Qemu, Parallels, etc. ofc
 - » Host-only network
- » Windows 10
 - » Easy setup → Use FlareVM
 - » <https://github.com/fireeye/flare-vm>
 - » Or build one yourself
- » Linux
 - » Easy setup → Use REMnux
 - » <https://remnux.org/>
 - » Or build one yourself



Yes – No internet





Cloud services are nice

1. Choose your favorite service
2. Create temporary host there
3. Interact with C2 like the malware does
4. Profit



Snapshots



**DIDN'T TAKE A
SNAPSHOT.**

DID YOU...?

S

S



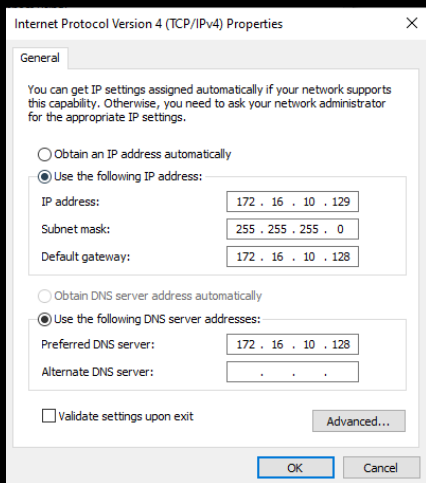
VM configuration

- » Remember **OPSEC** – mistakes happen
- » Use generic hostnames and usernames, **avoid** following:
 - » Analyst-PC
 - » FlareVM
 - » REM Workstation
 - » Sandbox
- » Use **VPN** on host machine while analyzing – If you accidentally connect your machine to internet, there's a chance you did not burn yourself



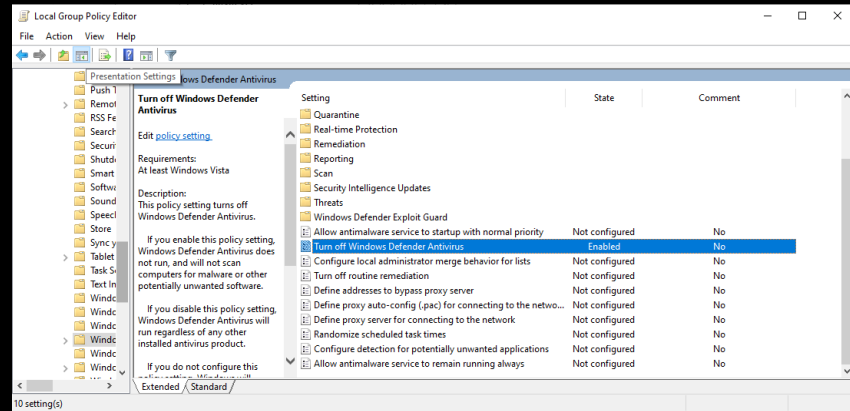
VM configuration

- » Set static IP addresses to both
 - » I tend to use non-default VMWare networks like 172.16.13.0/24
- » Use Linux VM IP address as a gateway AND DNS server on Windows machine



Windows configuration

- » Disable AV
 - » Might need some GPO adjusting
- » Install office tools
 - » Macros are still a thing !!!
- » Disable UAC
- » Disable passwords
- » Allow PowerShell scripts
 - » Enabling PowerShell transcript logging might ease your life



Tactics



Memory forensics

- » Super good and effective [and fun]
- » However, the cases where you actually get usable memory dump are rare [from my opinion]
- » Volatility is way to go...
 - » <https://www.volatilityfoundation.org>
- » ...BUT MemProcFS will soon be better than Volatility - go, check and try it out already
 - » <https://github.com/ufrisk/MemProcFS>



Memory forensics

Updated README (#553)

[Browse files](#)

* Update README.md

* Update README.md

master (#553)

joachimmetz committed on 18 Oct 2020 Verified

1 parent f6dd536 commit 55d1925f2df9759a989b35271b4fa48fc54a1c86

Showing 1 changed file with 19 additions and 0 deletions.

Unified Split

19 README.md

<> ...

@@ -1,3 +1,22 @@

1 + # **Rekall discontinuation**

2 +

3 + This project is no longer maintained.

4 +

5 + In December 2011, a new branch within the Volatility project was created to explore how to make the code base more modular, improve performance, and increase usability. This branch was later forked to become Rekall. The modularity allowed physical memory analysis functionality to be used in [GRR](https://github.com/google/grr) to enable remote live in-memory analysis.

6 +

REST IN PEACE

IN PIECES

memegenerator.net

Updated README (#55)

* Update README.md

* Update README.md

👤 master (#553)

✈ joachimmetz committed

📄 Showing 1 changed file with

▼ 19 █████ README.md

```
... @@ -1,3 +1,22 @
1 + # Rekall dis
2 +
3 + This project
4 +
5 + In December 2
  increase usabi
  [GRR](https://
6 +
```

Browse files

59a989b35271b4fa48fc54a1c86

Unified

Split

<>

📄

...

e performance, and
be used in

Plaso everything

- » In my experience, you should always run `plaso/log2timeline` against everything
 - » Eases up setting up incident timeline
 - » Might help you “find evil”
- » Even though you would not investigate the timeline immediately, you might need it later and plaso takes some time...
- » <https://github.com/log2timeline/plaso>



Tooling

REMnux



Almost all you need

- » Lenny's cheat sheets → <https://zeltser.com/cheat-sheets/>
- » When you run dynamic analysis on your Windows machine, run these [on your Linux]:
 - » inetsim
 - » fakedns
 - » Wireshark
 - » BurpSuite [optional]
 - » Needs some proxy configuration on Windows



Tooling

Windows / Forensics



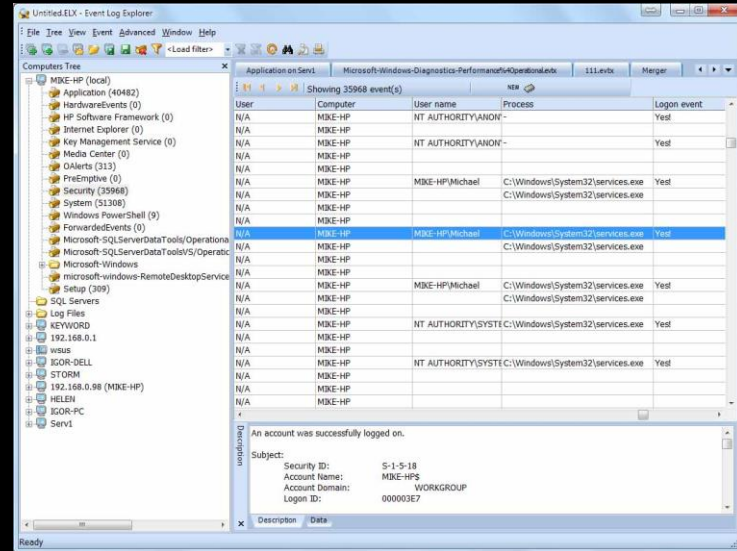
Zimmerman tools

- » <https://ericzimmerman.github.io/#!index.md>
- » Great set of FOSS tools
- » My personal favorites:
 - » Registry Explorer/RECmd
 - » RBCmd
 - » PECmd
 - » JumpList Explorer
 - » LECmd
 - » Timeline Explorer
 - » KAPE [***]



Event Log Explorer™

- » <https://eventlogxp.com/>
- » 199 USD for Standard Edition
- » Bang for the bucks
- » Make custom searches and views for Eventlogs



<https://eventlogxp.com/sshots/customcolumns.jpg>

Tooling

Windows / Malware analysis



pestudio

» <https://www.winitor.com/>

» Triage binary file

» Metadata

» Embedded files

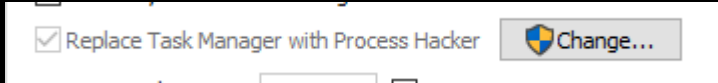
» Imports, exports,
strings

property	value
md5	AE12B854F31227017EFFD9598A6F5E
sha1	F397A1CC16D4287F0E077698E067CD3030A06D9
sha256	C05E2DAB77349CD639AA837E7E121710B8A0718D8FC93FB4CC6458AE90ESC97
md5-without-overlay	693E9AF84D3DFCC71E640E005BDC3E2E
sha1-without-overlay	29E2DCFB816F638B80540F7585A158B8F83E927D
sha256-without-overlay	706E80CB8497A2411E1E4DF89F22A861492D20C4765150C794ABD70F8147C
first-bytes-hex	4D 5A 90 00 03 00 00 00 04 00 00 00 FF FF 00 00 B8 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
first-bytes-text	M Z
file-size	5267499 (bytes)
size-without-overlay	5267496 (bytes)
entropy	6.413
imphash	7C308AE5FED0403E8117C645F823E5B
signature	Microsoft Visual C++ 6.0.DLL (Debug)
entry-point	55 8B EC 53 8B 5D 08 56 8B 75 0C 57 8B 7D 10 85 F6 75 09 83 3D 40 31 00 10 00 EB 26 83 FE 01 74 05
file-version	n/a
description	n/a
file-type	dynamic-link-library
cpu	32-bit
subsystem	GUI
compiler-stamp	0x59145751 (Thu May 11 12:21:37 2017 - UTC)
debugger-stamp	n/a
resources-stamp	empty
exports-stamp	0x59145751 (Thu May 11 12:21:37 2017)
version-stamp	n/a
certificate-stamp	n/a

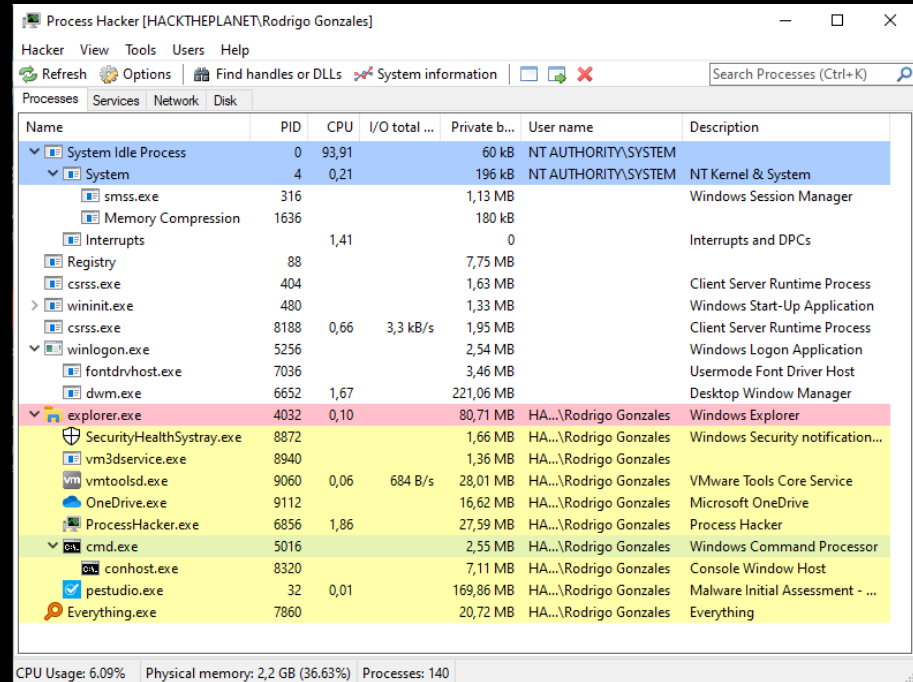
sha256: C05E2DAB77349CD639AA837E7E121710B8A0718D8FC93FB4CC6458AE90ESC97 cpu: 32-bit file-type: dynamic-link-library subsystem: GUI entry-point: 0x00011E9 signature: Micr

Process Hacker

- » <https://processhacker.sourceforge.io/>
- » Task manager on steroids



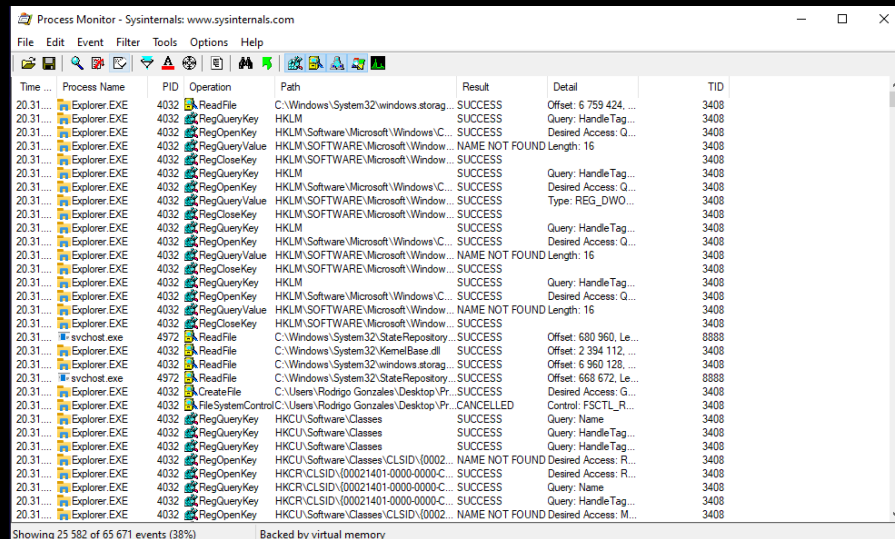
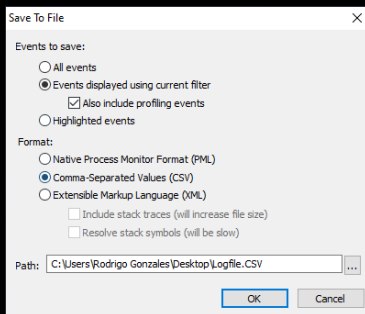
- » You can also replace Task Manager with Process Hacker
- » Many malware check if Process Hacker is running though



procmom

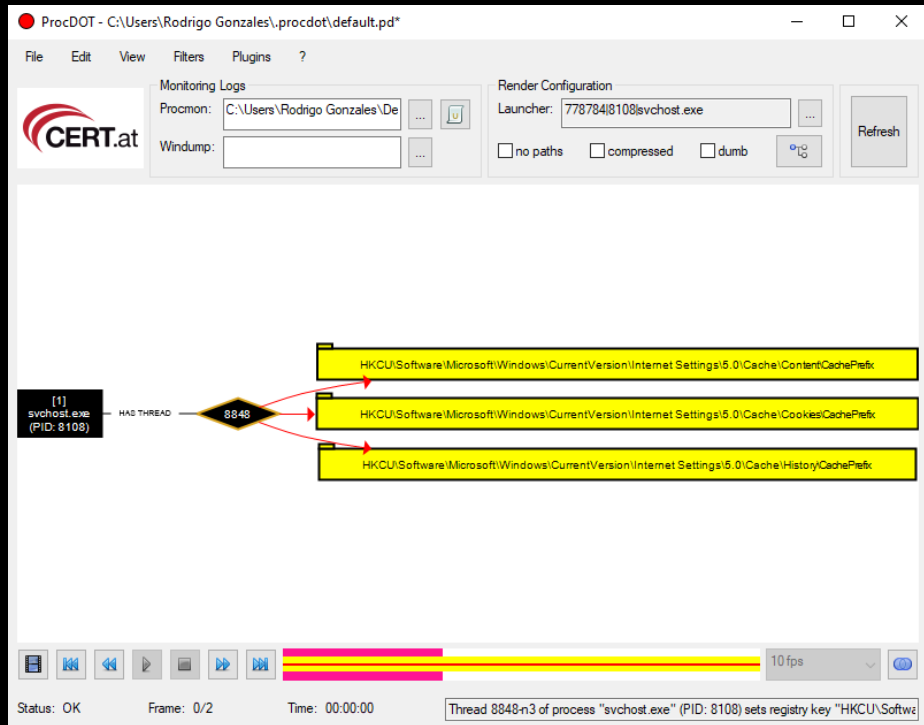
» <https://docs.microsoft.com/en-us/sysinternals/downloads/procmom>

» Part of SysInternals, which in general is good system utility collection for Windows



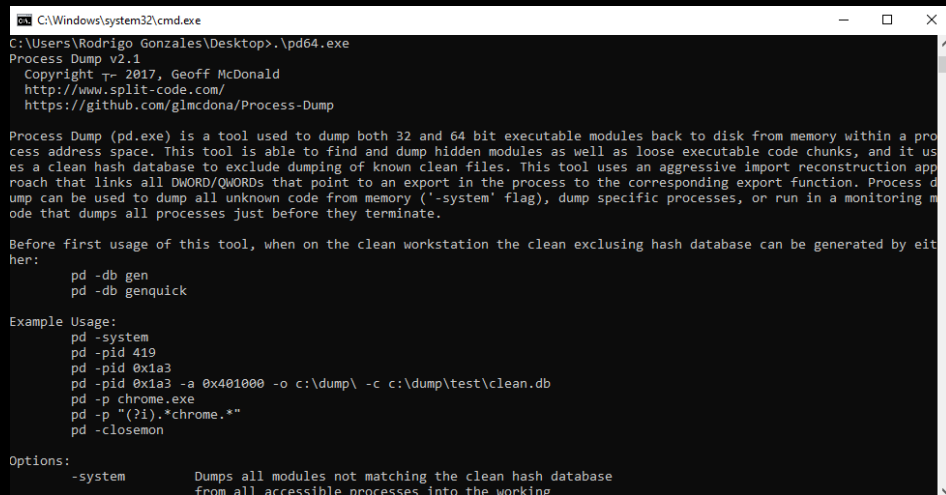
procdot

- » <https://www.procdot.com/>
- » Created by CERT.at's Christian Wojner
- » Visualization for procmon output AND packet captures



pd

- » <http://split-code.com/processdump.html>
- » Good tool to for malware unpacking
- » Remember to generate clean hash database before taking snapshot from your analysis machine!



```
C:\Windows\system32\cmd.exe
C:\Users\Rodrigo_Gonzales\Desktop>.\pd64.exe
Process Dump v2.1
Copyright © 2017, Geoff McDonald
http://www.split-code.com/
https://github.com/glmcdona/Process-Dump

Process Dump (pd.exe) is a tool used to dump both 32 and 64 bit executable modules back to disk from memory within a process address space. This tool is able to find and dump hidden modules as well as loose executable code chunks, and it uses a clean hash database to exclude dumping of known clean files. This tool uses an aggressive import reconstruction approach that links all DWORD/QWORDS that point to an export in the process to the corresponding export function. Process dump can be used to dump all unknown code from memory ('-system' flag), dump specific processes, or run in a monitoring mode that dumps all processes just before they terminate.

Before first usage of this tool, when on the clean workstation the clean excluding hash database can be generated by either:

pd -db gen
pd -db genquick

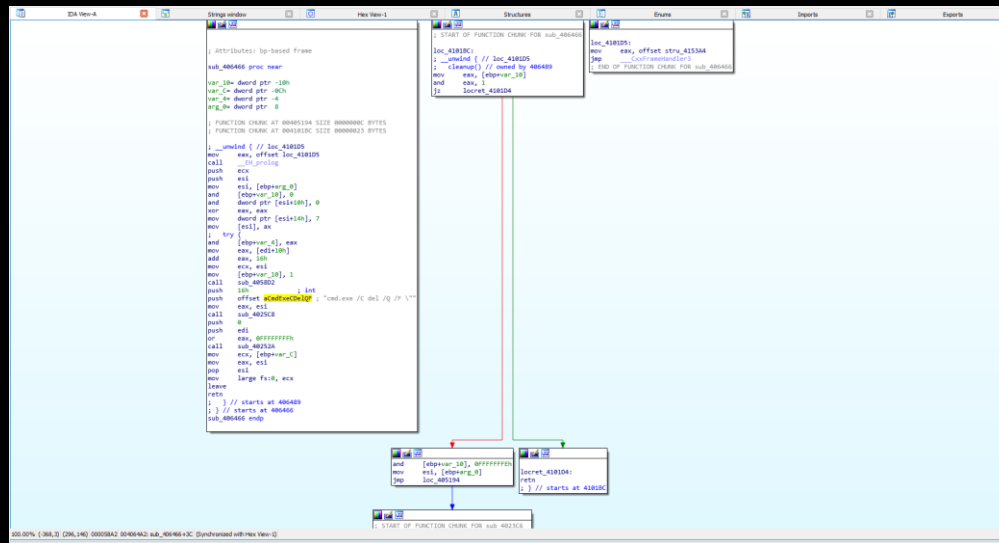
Example Usage:
pd -system
pd -pid 419
pd -pid 0x1a3
pd -pid 0x1a3 -a 0x401000 -o c:\dump\ -c c:\dump\test\clean.db
pd -p chrome.exe
pd -p "(?i).*chrome.*"
pd -closemon

Options:
-system          Dumps all modules not matching the clean hash database
                  from all accessible processes into the working
```



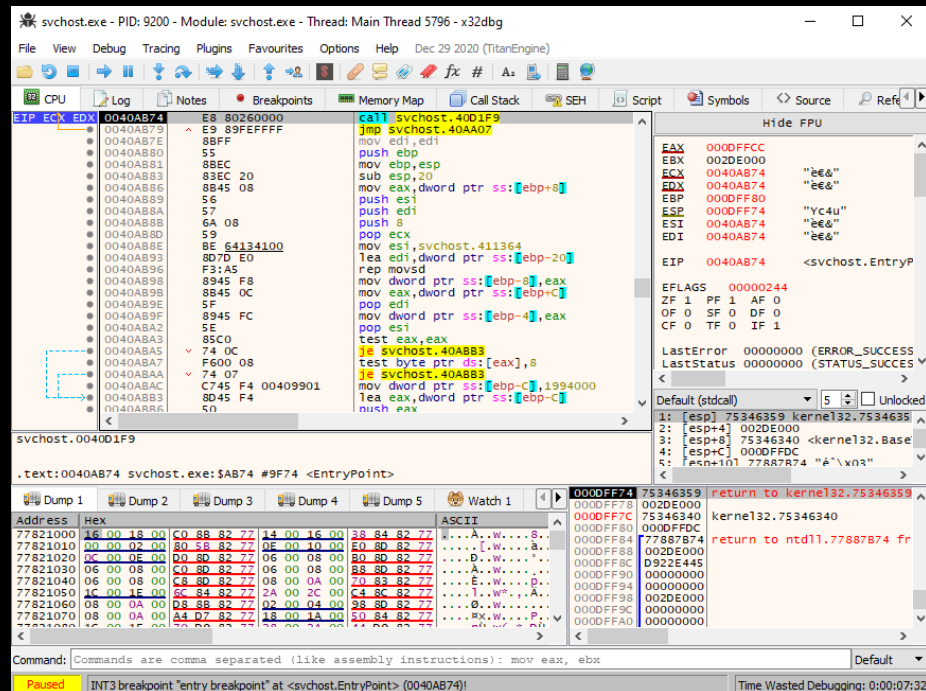
Ida Pro / Ida Free

- » <https://www.hex-rays.com/>
- » If you have sponsor for Ida Pro, good for you 👍👍👍
 - » Remember you can debug basically everything with this instance - <https://www.hex-rays.com/products/ida/support/idadoc/1463.shtml>
- » Otherwise, either use Ida Free and...



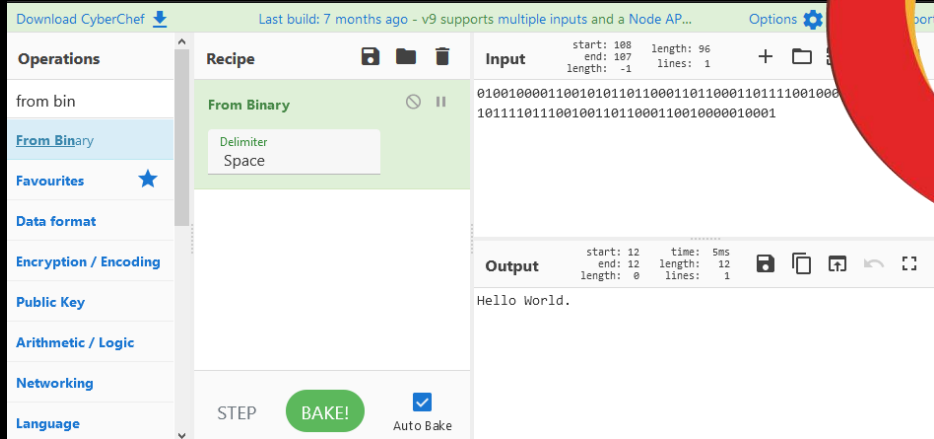
x64dbg / x32dbg

- » <https://x64dbg.com/#start>
- » Good open-source debugger, lot of plugins available
- » ...use this OR just go...



Ghidra

- » <https://ghidra-sre.org/>
- » ...to the future



GHIDRA

Other mentionable

- » dnSpy - <https://github.com/dnSpy/dnSpy>
 - » .NET binaries debugger / assembly editor
- » die - <http://ntinfo.biz/index.html>
 - » Detect packer
- » setdllcharacteristics -
<https://blog.didierstevens.com/2010/10/17/setdllcharacteristics/>
 - » Disable ASLR and DEP
- » UPX - <https://upx.github.io/>
 - » Unpack UPX
- » What ever you need...
 - » Usually you need to adapt and either install OR create (OR ask colleague to create) tool required for the task



Demo



T. Hanks!

Questions?

@JuhoJauhiainen

whois [at] helsec.fi

CREDITS:

This presentation template was created
by Slidesgo, including icons by FlatIcon,
infographics & images by Freepik