eForensics Magazine

MAGAZINE

Ransomware attacks and investigations

Ransomware in insurance claims

Frogo ransomware memory analysis

Ransomware and Incident Response

VOL.08 NO.01
ISSUE 01/2019, (86) JANUARY
ISSN 2300 6986

eForensics Magazine

TEAM

Editor-in-Chief
Joanna Kretowicz
joanna.kretowicz@eforensicsmag.com

Managing Editor:

Dominika Zdrodowska
dominika.zdrodowska@eforensicsmag.com

Editors:

Marta Sienicka
sienicka.marta@hakin9.com

Marta Strzelec
marta.strzelec@eforensicsmag.com

Bartek Adach
bartek.adach@pentestmag.com

Senior Consultant/Publisher:
Paweł Marciniak

CEO:
Joanna Kretowicz
joanna.kretowicz@eforensicsmag.com

Marketing Director:
Joanna Kretowicz
joanna.kretowicz@eforensicsmag.com

DTP
Dominika Zdrodowska
dominika.zdrodowska@eforensicsmag.com

Cover Design
Hiep Nguyen Duc

Publisher
Hakin9 Media Sp. z o.o.
02-676 Warszawa
ul. Postępu 17D
Phone: 1 917 338 3631

www.eforensicsmag.com

All trademarks, trade names, or logos mentioned or used are the property of their respective owners.

The techniques described in our articles may only be used in private, local networks. The editors hold no responsibility for misuse of the presented techniques or consequent data loss.

word from the team

Dear Readers,

Let's kick ransomware out of our lives in 2019! We're extremely proud to present the the first issue of eForensics Magazine of this year - Ransomware attacks and investigations.

In this publication you can find an amazing article "Ransomware in insurance claims" by Alistair Ewing and Jason Bergson, a paper about Frogo ransomware memory analysis (prepared for you by Paulo Henrique Pereira, the instructor from eForensics' course - Ransomware Forensics), and some tips for Ransomware Investigations and Incident Response, written by John Fokker from McAfee.

In addition to this you'll want to check out "Obtaining your Certified Forensic Computer Examiner Certification - Tips and Tricks" by Matt Beers. Our reviewers said that after reading it they felt like going to register for the exams immediately. I'm sure you will have the same feeling! Also, we have for you articles about malicious mail attachments, Windows Live Forensics, Amazon Echo Forensics, a forensic analysis of the Electronic Point Record System and still… that's not all!

In this issue we also present a part of our course Digital Visual Media Anti–Forensics and Counter Anti–Forensics - in which you can learn about active non-blind tamper detection solutions.Thanks to all authors, reviewers and proofreaders for participating in this project.

Have a nice read!

Regards,

Dominika Zdrodowska

and the eForensics Magazine Editorial Team

Table of Contents

Ransomware Investigations and Incident Response tips

By John Fokker

It was at the end of 2018 when I sat down to write this article on Ransomware, and I can't help to think about "Operation Bakovia" that took place exactly one year earlier in Romania. Operation Bakovia was the arrest of individuals responsible for spreading CTB-Locker Ransomware. Please enjoy this article where I share some personal experiences from real ransomware investigations and share forensic and prevention tips when faced with ransomware.

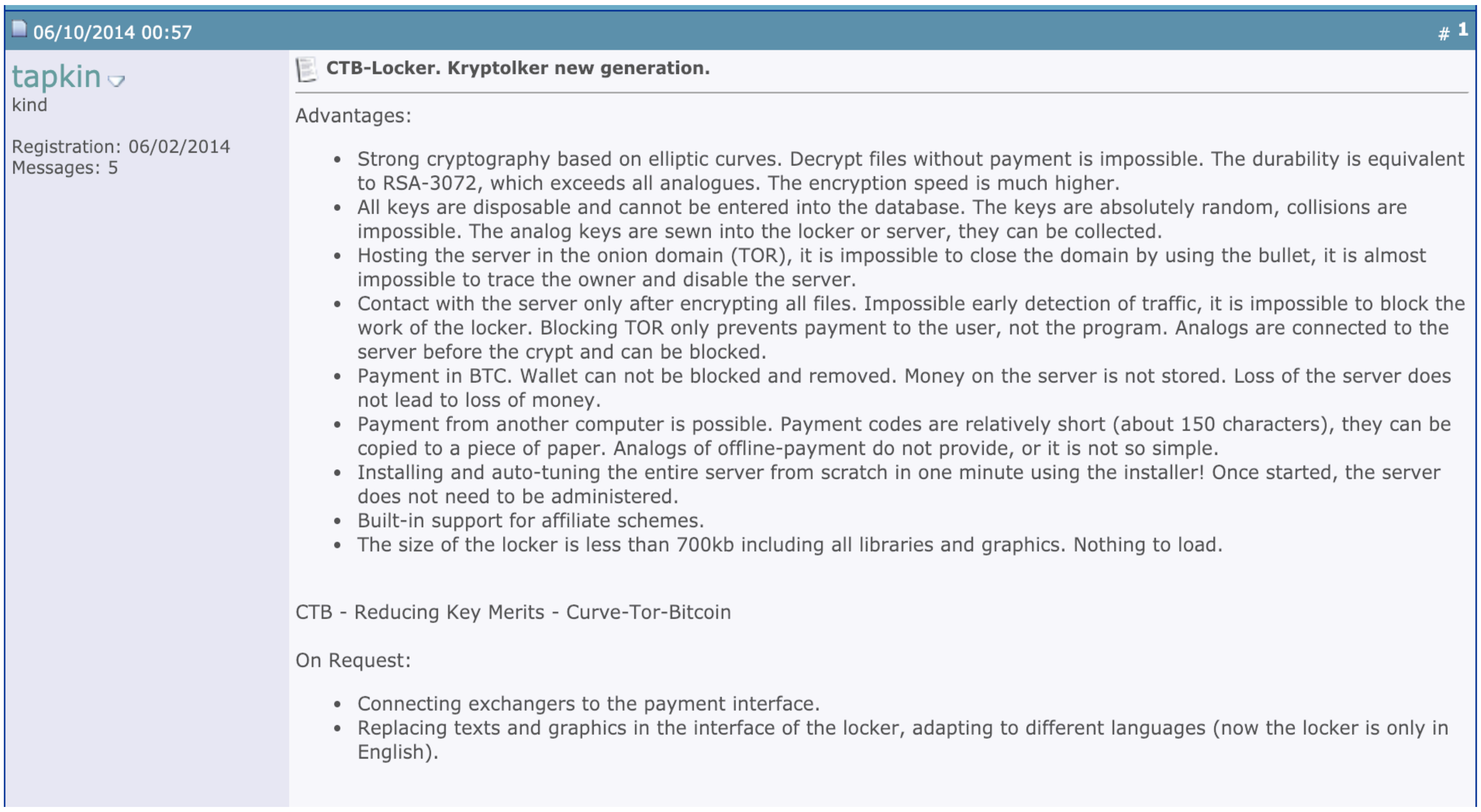
Intro

It was at the end of 2018 when I sat down to write this article on Ransomware, and I can't help to think about "Operation Bakovia[1]" that took place exactly one year earlier in Romania. Operation Bakovia was the arrest of individuals responsible for spreading CTB-Locker Ransomware. Please enjoy this article where I share some personal experiences from real ransomware investigations and share forensic and prevention tips when faced with ransomware.

CTB-Locker – Operation Bakovia

CTB-locker (Curve-Tor-Bitcoin) was one of the most prolific ransomware families that reigned from 2014 till 2016. CTB-locker made victims across the globe but was mostly targeted at western countries where victims could more likely afford to pay the ransom.

CTB-locker was offered through an affiliate model, this means that the developers sold the ransomware to partners, known as affiliates, who were responsible for spreading the ransomware. The affiliates, in return, had to share a percentage of their received ransoms with the developers. By using this model, the developers minimized their own risk of prosecution and at the same time offered a cyber-criminal career opportunity to less technical hackers, or botnet herders.



06/10/2014 00:57 # 1

tapkin
kind

Registration: 06/02/2014
Messages: 5

CTB-Locker. Kryptolker new generation.

Advantages:

- Strong cryptography based on elliptic curves. Decrypt files without payment is impossible. The durability is equivalent to RSA-3072, which exceeds all analogues. The encryption speed is much higher.
- All keys are disposable and cannot be entered into the database. The keys are absolutely random, collisions are impossible. The analog keys are sewn into the locker or server, they can be collected.
- Hosting the server in the onion domain (TOR), it is impossible to close the domain by using the bullet, it is almost impossible to trace the owner and disable the server.
- Contact with the server only after encrypting all files. Impossible early detection of traffic, it is impossible to block the work of the locker. Blocking TOR only prevents payment to the user, not the program. Analogs are connected to the server before the crypt and can be blocked.
- Payment in BTC. Wallet can not be blocked and removed. Money on the server is not stored. Loss of the server does not lead to loss of money.
- Payment from another computer is possible. Payment codes are relatively short (about 150 characters), they can be copied to a piece of paper. Analogs of offline-payment do not provide, or it is not so simple.
- Installing and auto-tuning the entire server from scratch in one minute using the installer! Once started, the server does not need to be administered.
- Built-in support for affiliate schemes.
- The size of the locker is less than 700kb including all libraries and graphics. Nothing to load.

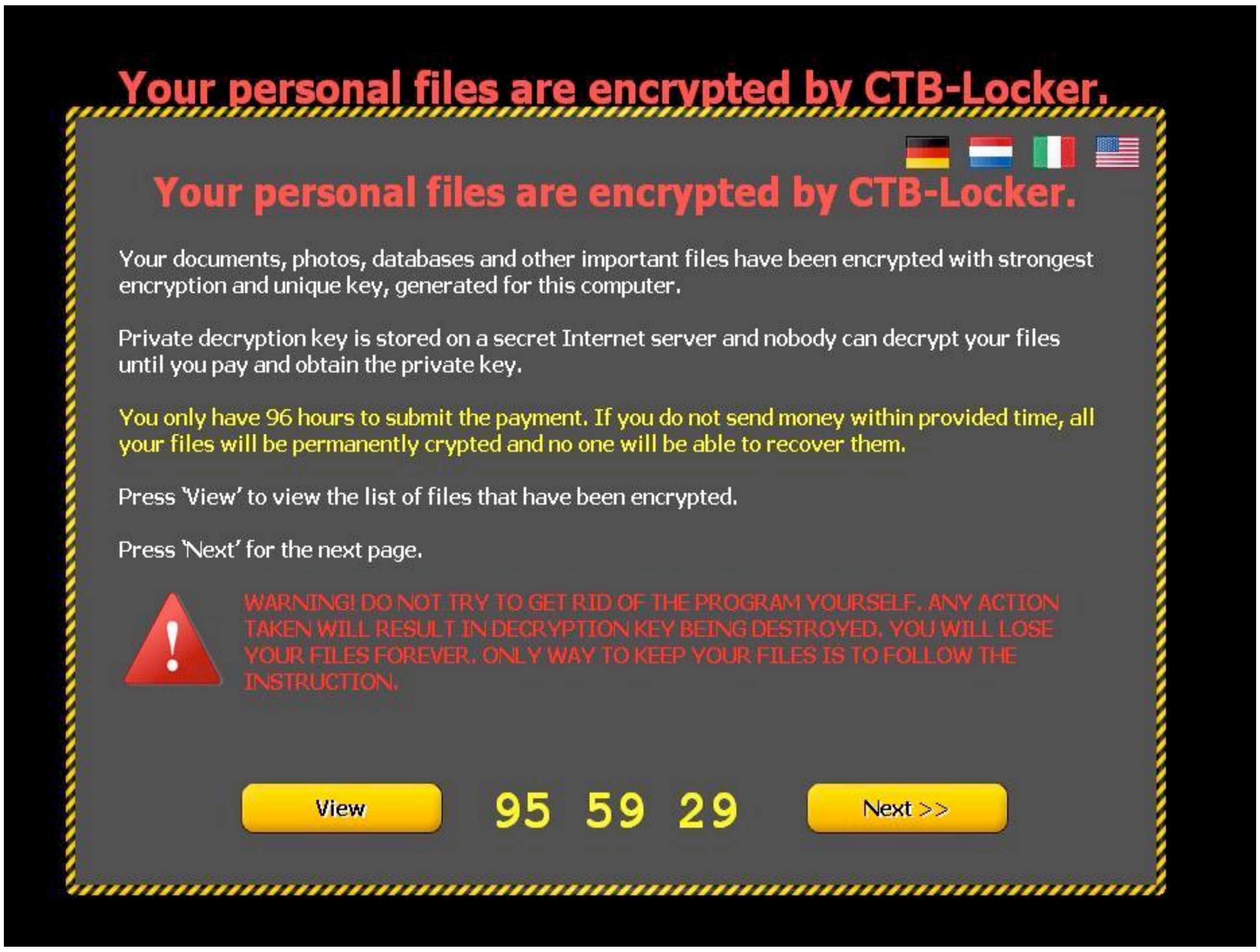
CTB - Reducing Key Merits - Curve-Tor-Bitcoin

On Request:

- Connecting exchangers to the payment interface.
- Replacing texts and graphics in the interface of the locker, adapting to different languages (now the locker is only in English).

The original advertisement for CTB-locker on zloy(.)bz underground forum

In 2016, the Netherlands was struck by a large spam campaign that impersonated one of the largest Telco providers. The spam mails included an invoice as attachment and were signed by an executive that actually worked at the Telco provider but in a different department. Obviously, the mails didn't include a real invoice, but in fact had the CTB-locker ransomware included. Once a victim opened the attachment, their computer system was encrypted and held for ransom.



Your personal files are encrypted by CTB-Locker.

Your personal files are encrypted by CTB-Locker.

Your documents, photos, databases and other important files have been encrypted with strongest encryption and unique key, generated for this computer.

Private decryption key is stored on a secret Internet server and nobody can decrypt your files until you pay and obtain the private key.

You only have 96 hours to submit the payment. If you do not send money within provided time, all your files will be permanently crypted and no one will be able to recover them.

Press 'View' to view the list of files that have been encrypted.

Press 'Next' for the next page.

WARNING! DO NOT TRY TO GET RID OF THE PROGRAM YOURSELF. ANY ACTION TAKEN WILL RESULT IN DECRYPTION KEY BEING DESTROYED. YOU WILL LOSE YOUR FILES FOREVER. ONLY WAY TO KEEP YOUR FILES IS TO FOLLOW THE INSTRUCTION.

CTB-Locker Lock screen

Mid 2016 the Dutch National High-Tech Crime Unit (NHTCU) received an anonymous tip that a certain server hosted in the Netherlands was sending out spam mails that had CTB-locker attached. At that time, I was the technical supervisor of the team that received the information and we decided to start a formal investigation into CTB-locker not knowing that this investigation would lead to eastern Romania.

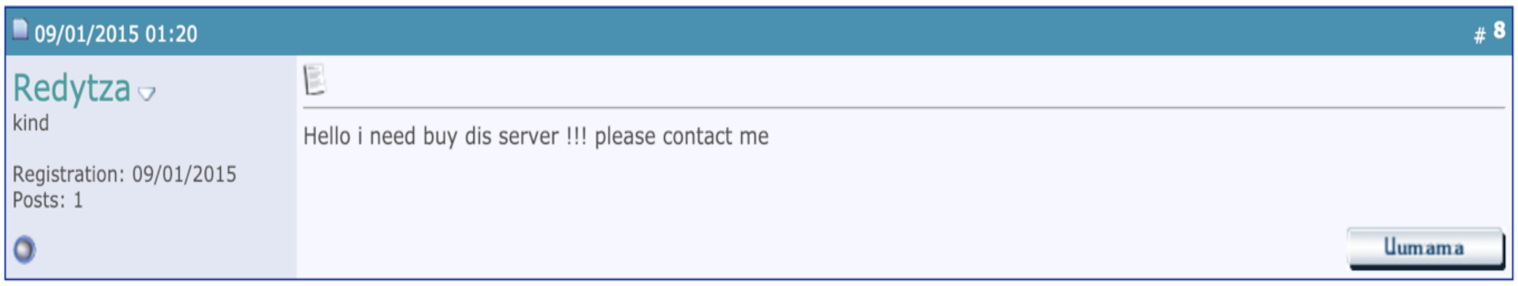
The server from the anonymous tip was eventually seized and imaged for forensic examination. This server acted as a command and control (C2) server for a botnet from which several spam campaigns were launched, not only directed at the Netherlands but also Italy and the United Kingdom. Forensic examination of the server revealed that the criminals controlled their botnet via Internet Relay Chat (IRC), however they made the mistake to chat with each other via the same chat channel that controlled the bots, so every bot could listen in on their conversations. Subsequently, they set up different folders for every spam campaign containing the fake mail body, company logos, version of CTB-locker and a script for spoofing the mail headers. At the time, we had requested that cyber security company McAfee analyze the samples we found and confirm that the viruses found were indeed CTB-locker.

Besides from the viruses, the spoofing script contained a major clue. The script contained a commented-out email address.

Spammers often test their campaigns with an email address that is under their control, that way they can make sure that everything works properly before they mass mail.

In this case, the mail address didn't belong to an anonymous mailing service but belong to a major American Internet Service Company. This meant that more information on the mail account could be obtained by sending out a subpoena via the formal MLAT (Mutual Legal Assistance Treaty) process. Unfortunately, the MLAT process is a long one and not quite equipped to deal with today's speed with which information travels. Eventually the information returned, and it proved to be full of useful evidence. The mail account had chatted extensively in Romanian with a co-conspirator via the Internet Service Company's Chat messenger. During these chats, they spoke about a new business opportunity in ransomware. They have been doing other forms of cybercrime, such as phishing and SMS fraud, but being a partner for CTB-locker seemed pretty lucrative.

Around the same time that the chats took place between the two, an account linked to one of the suspects reached out to the CTB-locker developers on the underground forum Zloy to ask if they could join as an affiliate.



09/01/2015 01:20 # 8

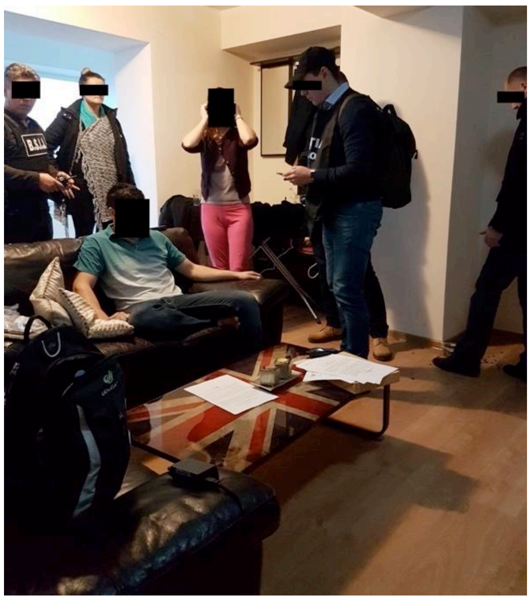
Redytza
kind

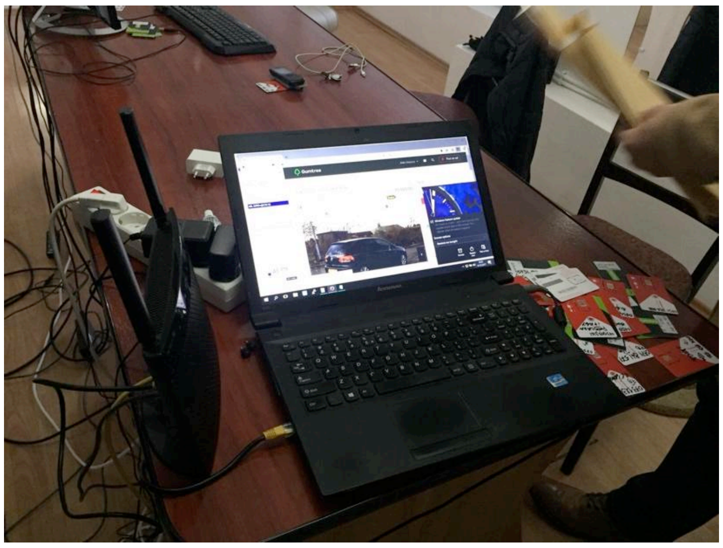
Registration: 09/01/2015
Posts: 1

Hello i need buy dis server !!! please contact me

Additional account data obtained via the subpoena, combined with open-source research, eventually led to the identification of all the individuals involved and enough evidence to hand over the case to the Romanian authorities. This information package eventually led to the international law enforcement operation "Bakovia" at the end of 2017. During the operation, six locations were searched, three people detained, and several devices were seized. The suspects are awaiting their trial in Romania.

Pictures of Operation Bakovia:





Coinvault

Coinvault was my first ransomware investigation at NHTCU. Till this day, it remains a special one, not because it was tremendously complicated, but it did show how the security industry and law enforcement can work together effectively.

The case started when an owner of a telco company reached out with the message that part of his website was hacked and that the website database was used to store almost a thousand decryption keys. This gave us a unique opportunity because we were able to help ransomware victims.

By teaming up with the security industry, a decryptor for CoinVault was quickly built based on the keyset. This decryptor helped hundreds of victims who were able to get their files back, without paying the ransom. It was a great success and we felt that we had to do more… eventually this investigation was the start of NoMoreRansom.

But besides helping the victims, the investigation eventually led us to the criminals behind CoinVault. A world first that the developers and spreaders of ransomware were caught, so how did we find the criminals behind CoinVault?

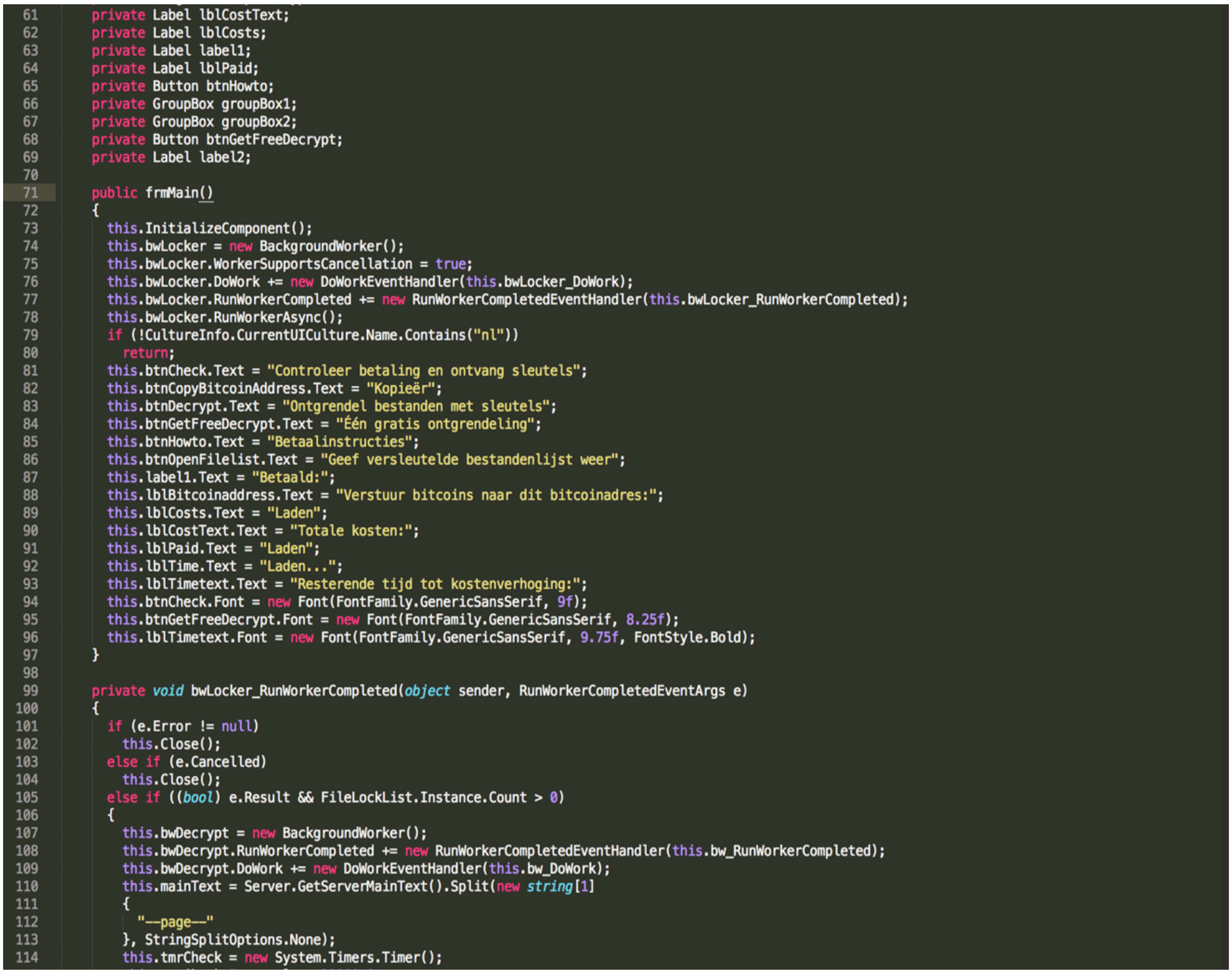
One of our first steps was examining the malware itself.

The malware was coded in Microsoft's .NET framework - this was easily determined using a tool called "Detect It Easy".

The next step was to use the Il-Spy de-compiler for .NET so see if the ransomware was easily de-complied. This was the case and we ended up with clearly readable source-code. Most professional ransomware isn't coded in .NET and easily de-compiled. Cyber-criminals do this on purpose because the original source code can contain a lot of forensic evidence.

An excellent write-up on the malware analysis of CoinVault is available online and can be found here.

One of the interesting findings that came out of the malware analysis was that the Dutch language instructions of the ransomware were hard-coded and absolutely flawlessly written with no spelling errors. Dutch isn't a common language at all, and it showed no signs of the use of an automatic translation service. So, whoever wrote the malware must have written the Dutch himself or had access to someone who could provide the right text.

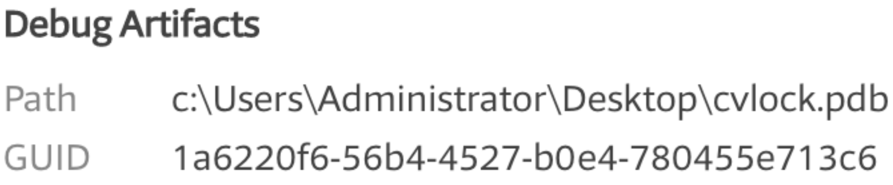


```
private Label lblCostText;
private Label lblCosts;
private Label label1;
private Label lblPaid;
private Button btnHowto;
private GroupBox groupBox1;
private GroupBox groupBox2;
private Button btnGetFreeDecrypt;
private Label label2;

public frmMain()
{
  this.InitializeComponent();
  this.bwLocker = new BackgroundWorker();
  this.bwLocker.WorkerSupportsCancellation = true;
  this.bwLocker.DoWork += new DoWorkEventHandler(this.bwLocker_DoWork);
  this.bwLocker.RunWorkerCompleted += new RunWorkerCompletedEventHandler(this.bwLocker_RunWorkerCompleted);
  this.bwLocker.RunWorkerAsync();
  if (!CultureInfo.CurrentUICulture.Name.Contains("nl"))
    return;
  this.btnCheck.Text = "Controleer betaling en ontvang sleutels";
  this.btnCopyBitcoinAddress.Text = "Kopieër";
  this.btnDecrypt.Text = "Ontgrendel bestanden met sleutels";
  this.btnGetFreeDecrypt.Text = "Één gratis ontgrendeling";
  this.btnHowto.Text = "Betaalinstructies";
  this.btnOpenFilelist.Text = "Geef versleutelde bestandenlijst weer";
  this.label1.Text = "Betaald:";
  this.lblBitcoinaddress.Text = "Verstuur bitcoins naar dit bitcoinadres:";
  this.lblCosts.Text = "Laden";
  this.lblCostText.Text = "Totale kosten:";
  this.lblPaid.Text = "Laden";
  this.lblTime.Text = "Laden...";
  this.lblTimetext.Text = "Resterende tijd tot kostenverhoging:";
  this.btnCheck.Font = new Font(FontFamily.GenericSansSerif, 9f);
  this.btnGetFreeDecrypt.Font = new Font(FontFamily.GenericSansSerif, 8.25f);
  this.lblTimetext.Font = new Font(FontFamily.GenericSansSerif, 9.75f, FontStyle.Bold);
}

private void bwLocker_RunWorkerCompleted(object sender, RunWorkerCompletedEventArgs e)
{
  if (e.Error != null)
    this.Close();
  else if (e.Cancelled)
    this.Close();
  else if ((bool) e.Result && FileLockList.Instance.Count > 0)
  {
    this.bwDecrypt = new BackgroundWorker();
    this.bwDecrypt.RunWorkerCompleted += new RunWorkerCompletedEventHandler(this.bw_RunWorkerCompleted);
    this.bwDecrypt.DoWork += new DoWorkEventHandler(this.bw_DoWork);
    this.mainText = Server.GetServerMainText().Split(new string[1]
    {
      "—page—"
    }, StringSplitOptions.None);
    this.tmrCheck = new System.Timers.Timer();
```

Flawless Dutch hard-coded in the Coinvault source code

Another crucial mistake the Coinvault authors made was leaving the compiler debugging mode on while compiling their malware, this left a Program Database (PDB) directory path hardcoded in the malware.

By simply using the Strings command in Linux, we were able to display all readable strings in the malware and also display the PDB paths, which looked something like this:



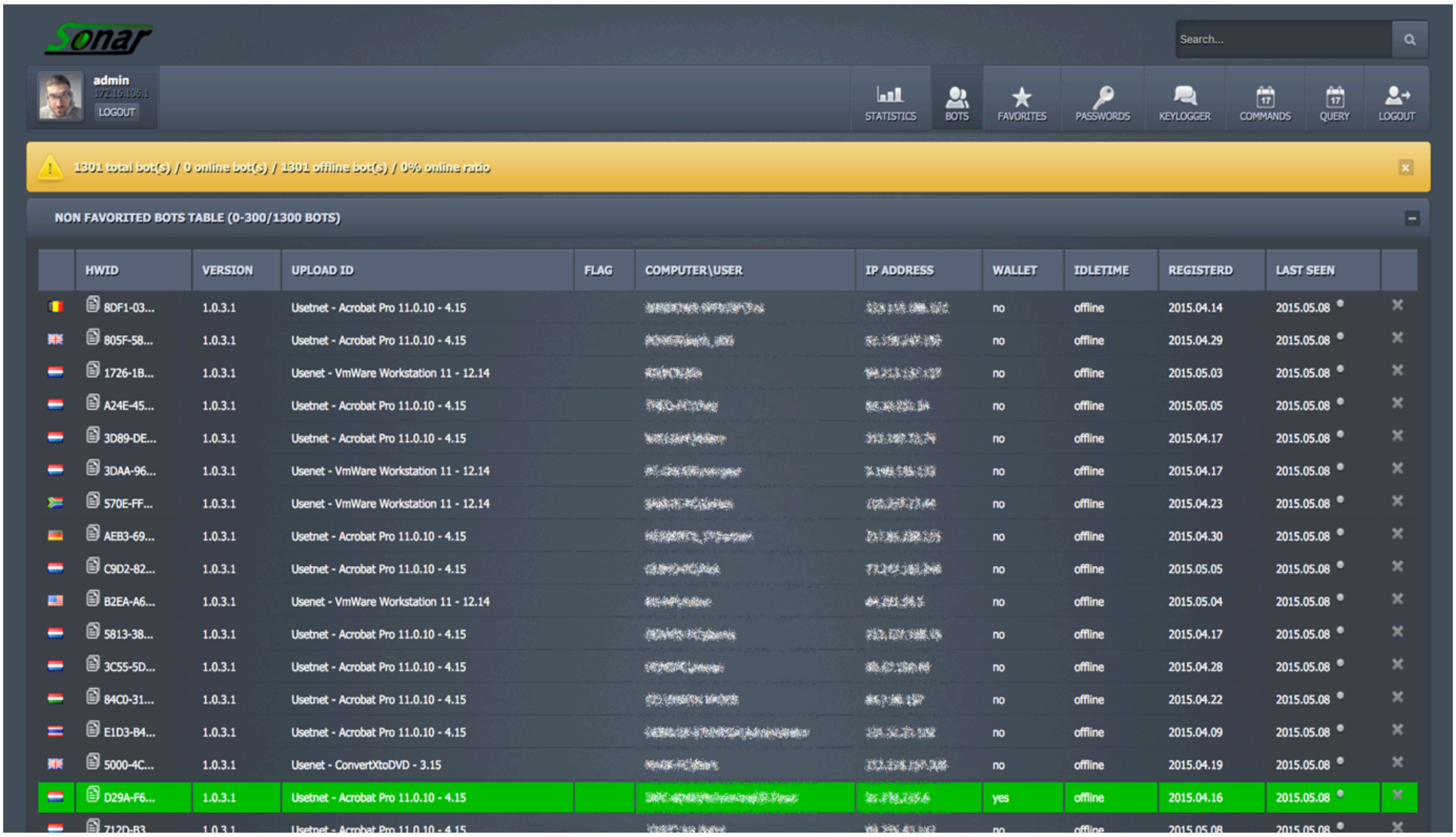
Debug Artifacts

Path	c:\Users\Administrator\Desktop\cvlock.pdb
GUID	1a6220f6-56b4-4527-b0e4-780455e713c6

The debug artifacts for Coinvault sample e6227eaefc147e66e3c7fa87a7e90fd6.

In the example above, the PDB path C:\Users\Administrator\.. is rather generic, however the earlier samples of Coinvault contained the first and last names of the suspects. It might seem like a done deal from now on, but until we had additional evidence that could link the names to the malware, we would have to consider that it could be a false flag.

Eventually, additional evidence came from one of the C2 servers they used in their campaign to launch their ransomware. This C2 server happened to be hosted on the website of an art-house cinema in the Netherlands. The location of this server was also established from the malware analysis.



1301 total bot(s) / 0 online bot(s) / 1301 offline bot(s) / 0% online ratio

NON FAVORITED BOTS TABLE (0-300/1300 BOTS)

	HWID	VERSION	UPLOAD ID	FLAG	COMPUTER\USER	IP ADDRESS	WALLET	IDLETIME	REGISTERD	LAST SEEN	
🇧🇪	8DF1-03...	1.0.3.1	Usetnet - Acrobat Pro 11.0.10 - 4.15		[redacted]	[redacted]	no	offline	2015.04.14	2015.05.08 ●	✕
🇬🇧	805F-58...	1.0.3.1	Usetnet - Acrobat Pro 11.0.10 - 4.15		[redacted]	[redacted]	no	offline	2015.04.29	2015.05.08 ●	✕
🇳🇱	1726-1B...	1.0.3.1	Usenet - VmWare Workstation 11 - 12.14		[redacted]	[redacted]	no	offline	2015.05.03	2015.05.08 ●	✕
🇳🇱	A24E-45...	1.0.3.1	Usetnet - Acrobat Pro 11.0.10 - 4.15		[redacted]	[redacted]	no	offline	2015.05.05	2015.05.08 ●	✕
🇳🇱	3D89-DE...	1.0.3.1	Usetnet - Acrobat Pro 11.0.10 - 4.15		[redacted]	[redacted]	no	offline	2015.04.17	2015.05.08 ●	✕
🇳🇱	3DAA-96...	1.0.3.1	Usenet - VmWare Workstation 11 - 12.14		[redacted]	[redacted]	no	offline	2015.04.17	2015.05.08 ●	✕
🇿🇦	570E-FF...	1.0.3.1	Usenet - VmWare Workstation 11 - 12.14		[redacted]	[redacted]	no	offline	2015.04.23	2015.05.08 ●	✕
🇩🇪	AEB3-69...	1.0.3.1	Usetnet - Acrobat Pro 11.0.10 - 4.15		[redacted]	[redacted]	no	offline	2015.04.30	2015.05.08 ●	✕
🇳🇱	C9D2-82...	1.0.3.1	Usetnet - Acrobat Pro 11.0.10 - 4.15		[redacted]	[redacted]	no	offline	2015.05.05	2015.05.08 ●	✕
🇺🇸	B2EA-A6...	1.0.3.1	Usenet - VmWare Workstation 11 - 12.14		[redacted]	[redacted]	no	offline	2015.05.04	2015.05.08 ●	✕
🇳🇱	5813-38...	1.0.3.1	Usetnet - Acrobat Pro 11.0.10 - 4.15		[redacted]	[redacted]	no	offline	2015.04.17	2015.05.08 ●	✕
🇳🇱	3C55-5D...	1.0.3.1	Usetnet - Acrobat Pro 11.0.10 - 4.15		[redacted]	[redacted]	no	offline	2015.04.28	2015.05.08 ●	✕
🇭🇺	84C0-31...	1.0.3.1	Usetnet - Acrobat Pro 11.0.10 - 4.15		[redacted]	[redacted]	no	offline	2015.04.22	2015.05.08 ●	✕
🇹🇭	E1D3-B4...	1.0.3.1	Usetnet - Acrobat Pro 11.0.10 - 4.15		[redacted]	[redacted]	no	offline	2015.04.09	2015.05.08 ●	✕
🇬🇧	5000-4C...	1.0.3.1	Usenet - ConvertXtoDVD - 3.15		[redacted]	[redacted]	no	offline	2015.04.19	2015.05.08 ●	✕
🇳🇱	D29A-F6...	1.0.3.1	Usetnet - Acrobat Pro 11.0.10 - 4.15		[redacted]	[redacted]	yes	offline	2015.04.16	2015.05.08 ●	✕
🇳🇱	712D-B3	1.0.3.1	Usetnet - Acrobat Pro 11.0.10 - 4.15		[redacted]	[redacted]	no	offline	2015.05.08	2015.05.08 ●	✕

Screenshot of the botnet control panel, from where the criminals launched their ransomware.

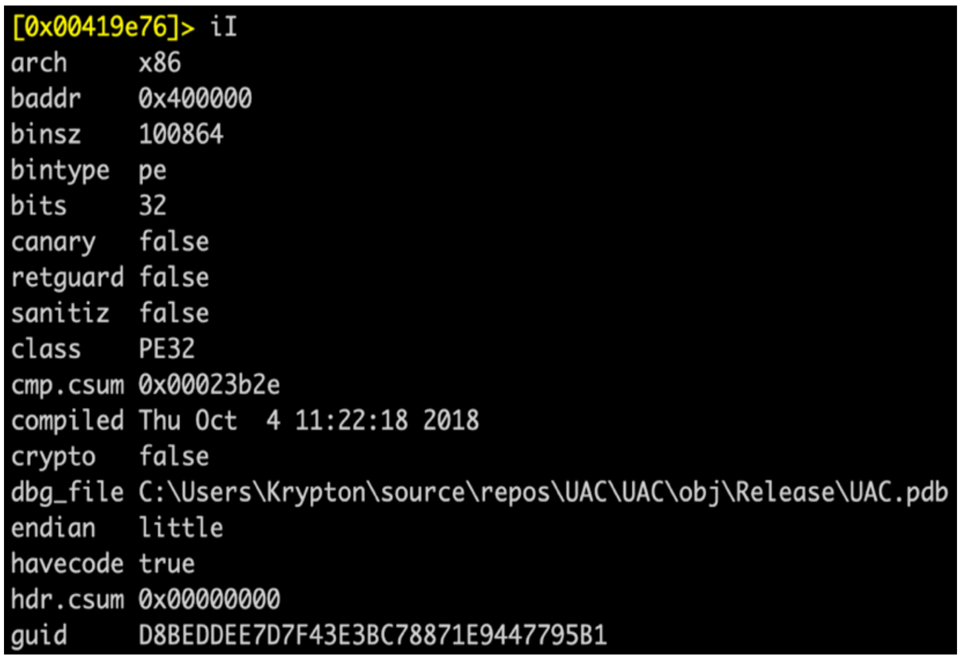
Upon this discovery, we requested a court-order for a wiretap on that specific server, this way we could effectively monitor their activity and determine the size of their botnet.

Eventually, this wiretap paid off and the criminals made a mistake by connecting to the C2 server without the use of anonymization such as TOR or a VPN. They left four HTTP requests that we could trace back to a residential landline in the Netherlands.

After checking the subscriber info of that specific landline, we discovered that the family living at that address had two sons, whose names matched up exactly with the names in the PDB path. Eventually, this case led to the arrest and conviction of the two brothers - more on the court case and what drove the criminals to commit their acts can be read here.

Kraken Ransomware

PDB paths aren't a thing of the past. In one McAfee ATR's ransomware research reports from the end of 2018, we took a closer look at Kraken ransomware, a new ransomware family that appeared in the summer of 2018. Kraken was also written in .NET and contained a PDB path with the name Krypton. This was done with the free reversing tool Radare2.



```
[0x00419e76]> iI
arch     x86
baddr    0x400000
binsz    100864
bintype  pe
bits     32
canary   false
retguard false
sanitiz  false
class    PE32
cmp.csum 0x00023b2e
compiled Thu Oct  4 11:22:18 2018
crypto   false
dbg_file C:\Users\Krypton\source\repos\UAC\UAC\obj\Release\UAC.pdb
endian   little
havecode true
hdr.csum 0x00000000
guid     D8BEDDEE7D7F43E3BC78871E9447795B1
```

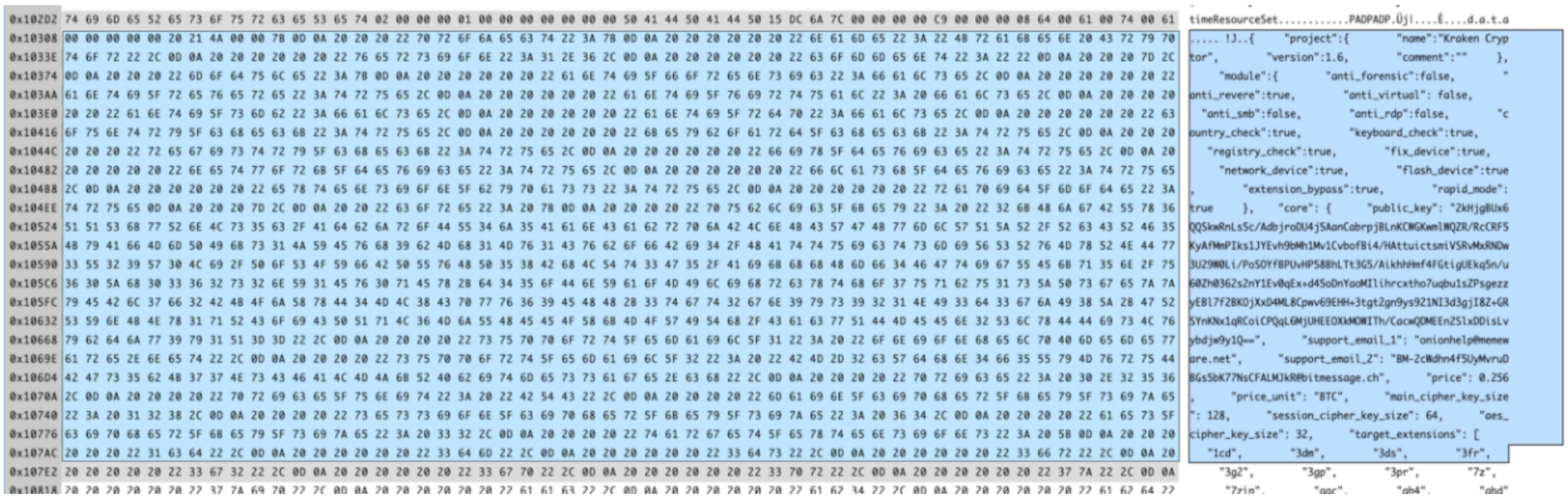
Debug information on Kraken ransomware, including the PDB path with the username Krypton

Now this isn't exactly a smoking gun, but it did help to hunt for other versions of Kraken based on the Krypton PDB path. Kraken ransomware is a typical Ransomware-as-a-service(RaaS) that is being offered to affiliates on one of the underground forums.

The Kraken Advertisement logo from one of the Underground hacker Forums

Kraken was clearly developed with affiliates in mind. The ransomware executable had a built-in configuration file which most likely could be used by affiliates to fine tune their version of the virus.

The details of the config file are clearly visible when the ransomware executable is examined with a simple hex-editor.



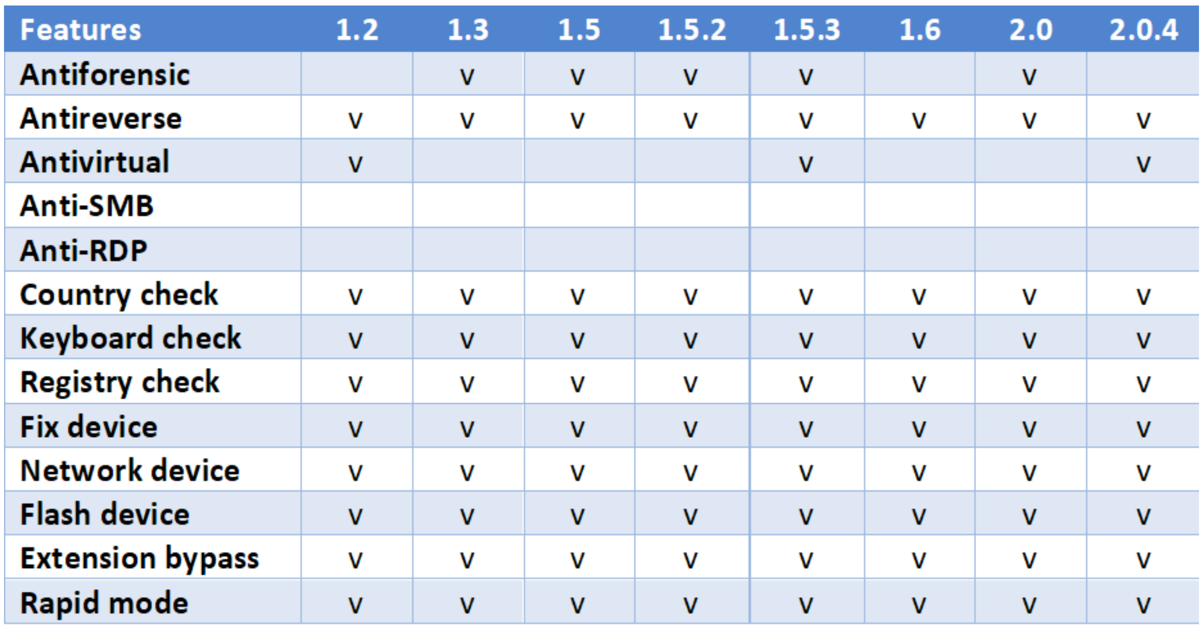
```
0x102D2 74 69 6D 65 52 65 73 6F 75 72 63 65 53 65 74 02 00 00 00 01 00 00 00 00 00 00 00 50 41 44 50 41 44 50 15 DC 6A 7C 00 00 00 00 C9 00 00 00 08 64 00 61 00 74 00 61 timeResourceSet............PADPADP.Üj|....É....d.a.t.a
0x10308 00 00 00 00 00 20 21 4A 00 00 7B 0D 0A 20 20 20 22 70 72 6F 6A 65 63 74 22 3A 7B 0D 0A 20 20 20 20 20 20 22 6E 61 6D 65 22 3A 22 4B 72 61 6B 65 6E 20 43 72 79 70 ..... !J..{     "project":{        "name":"Kraken Cryp
0x1033E 74 6F 72 22 2C 0D 0A 20 20 20 20 20 20 22 76 65 72 73 69 6F 6E 22 3A 31 2E 36 2C 0D 0A 20 20 20 20 20 20 22 63 6F 6D 6D 65 6E 74 22 3A 22 22 0D 0A 20 20 20 7D 2C tor",        "version":1.6,        "comment":""     },
0x10374 0D 0A 20 20 20 22 6D 6F 64 75 6C 65 22 3A 7B 0D 0A 20 20 20 20 20 20 22 61 6E 74 69 5F 66 6F 72 65 6E 73 69 63 22 3A 66 61 6C 73 65 2C 0D 0A 20 20 20 20 20 20 22      "module":{        "anti_forensic":false,        "
0x103AA 61 6E 74 69 5F 72 65 76 65 72 65 22 3A 74 72 75 65 2C 0D 0A 20 20 20 20 20 20 22 61 6E 74 69 5F 76 69 72 74 75 61 6C 22 3A 20 66 61 6C 73 65 2C 0D 0A 20 20 20 20 anti_revere":true,        "anti_virtual": false,
0x103E0 20 20 22 61 6E 74 69 5F 73 6D 62 22 3A 66 61 6C 73 65 2C 0D 0A 20 20 20 20 20 20 22 61 6E 74 69 5F 72 64 70 22 3A 66 61 6C 73 65 2C 0D 0A 20 20 20 20 20 20 22 63   "anti_smb":false,        "anti_rdp":false,        "c
0x10416 6F 75 6E 74 72 79 5F 63 68 65 63 6B 22 3A 74 72 75 65 2C 0D 0A 20 20 20 20 20 20 22 6B 65 79 62 6F 61 72 64 5F 63 68 65 63 6B 22 3A 74 72 75 65 2C 0D 0A 20 20 20 ountry_check":true,        "keyboard_check":true,
0x1044C 20 20 20 22 72 65 67 69 73 74 72 79 5F 63 68 65 63 6B 22 3A 74 72 75 65 2C 0D 0A 20 20 20 20 20 20 22 66 69 78 5F 64 65 76 69 63 65 22 3A 74 72 75 65 2C 0D 0A 20    "registry_check":true,        "fix_device":true,
0x10482 20 20 20 20 20 22 6E 65 74 77 6F 72 6B 5F 64 65 76 69 63 65 22 3A 74 72 75 65 2C 0D 0A 20 20 20 20 20 20 22 66 6C 61 73 68 5F 64 65 76 69 63 65 22 3A 74 72 75 65      "network_device":true,        "flash_device":true
0x104B8 2C 0D 0A 20 20 20 20 20 20 22 65 78 74 65 6E 73 69 6F 6E 5F 62 79 70 61 73 73 22 3A 74 72 75 65 2C 0D 0A 20 20 20 20 20 20 22 72 61 70 69 64 5F 6D 6F 64 65 22 3A ,        "extension_bypass":true,        "rapid_mode":
0x104EE 74 72 75 65 0D 0A 20 20 20 7D 2C 0D 0A 20 20 22 63 6F 72 65 22 3A 20 7B 0D 0A 20 20 20 20 22 70 75 62 6C 69 63 5F 6B 65 79 22 3A 20 22 32 6B 48 6A 67 42 55 78 36 true     },    "core": {      "public_key": "2kHjgBUx6
0x10524 51 51 53 6B 77 52 6E 4C 73 35 63 2F 41 64 62 6A 72 6F 44 55 34 6A 35 41 61 6E 43 61 62 72 70 6A 42 4C 6E 4B 43 57 47 4B 77 6D 6C 57 51 5A 52 2F 52 63 43 52 46 35 QQSkwRnLs5c/AdbjroDU4j5AanCabrpjBLnKCWGKwmlWQZR/RcCRF5
0x1055A 4B 79 41 66 4D 6D 50 49 6B 73 31 4A 59 45 76 68 39 62 4D 68 31 4D 76 31 43 76 62 6F 66 42 69 34 2F 48 41 74 74 75 69 63 74 73 6D 69 56 53 52 76 4D 78 52 4E 44 77 KyAfMmPIks1JYEvh9bMh1Mv1CvbofBi4/HAttuictsmiVSRvMxRNDw
0x10590 33 55 32 39 57 30 4C 69 2F 50 6F 53 4F 59 66 42 50 55 76 48 50 35 38 42 68 4C 54 74 33 47 35 2F 41 69 6B 68 68 48 6D 66 34 46 47 74 69 67 55 45 6B 71 35 6E 2F 75 3U29W0Li/PoSOYfBPUvHP58BhLTt3G5/AikhhHmf4FGtigUEkq5n/u
0x105C6 36 30 5A 68 30 33 36 32 73 32 6E 59 31 45 76 30 71 45 78 2B 64 34 35 6F 44 6E 59 61 6F 4D 49 6C 69 68 72 63 78 74 68 6F 37 75 71 62 75 31 73 5A 50 73 67 65 7A 7A 60Zh0362s2nY1Ev0qEx+d45oDnYaoMIlihrcxtho7uqbu1sZPsgezz
0x105FC 79 45 42 6C 37 66 32 42 4B 4F 6A 58 78 44 34 4D 4C 38 43 70 77 76 36 39 45 48 48 2B 33 74 67 74 32 67 6E 39 79 73 39 32 31 4E 49 33 64 33 67 6A 49 38 5A 2B 47 52 yEBl7f2BKOjXxD4ML8Cpwv69EHH+3tgt2gn9ys921NI3d3gjI8Z+GR
0x10632 53 59 6E 4B 4E 78 31 71 52 43 6F 69 43 50 51 71 4C 36 4D 6A 55 48 45 45 4F 58 6B 4D 4F 57 49 54 68 2F 43 61 63 77 51 44 4D 45 45 6E 32 53 6C 78 44 44 69 73 4C 76 SYnKNx1qRCoiCPQqL6MjUHEEOXkMOWITh/CacwQDMEEn2SlxDDisLv
0x10668 79 62 64 6A 77 39 79 31 51 3D 3D 22 2C 0D 0A 20 20 20 20 22 73 75 70 70 6F 72 74 5F 65 6D 61 69 6C 5F 31 22 3A 20 22 6F 6E 69 6F 6E 68 65 6C 70 40 6D 65 6D 65 77 ybdjw9y1Q==",      "support_email_1": "onionhelp@memew
0x1069E 61 72 65 2E 6E 65 74 22 2C 0D 0A 20 20 20 20 22 73 75 70 70 6F 72 74 5F 65 6D 61 69 6C 5F 32 22 3A 20 22 42 4D 2D 32 63 57 64 68 6E 34 66 35 55 79 4D 76 72 75 44 are.net",      "support_email_2": "BM-2cWdhn4f5UyMvruD
0x106D4 42 47 73 35 62 4B 37 37 4E 73 43 46 41 4C 4D 4A 6B 52 40 62 69 74 6D 65 73 73 61 67 65 2E 63 68 22 2C 0D 0A 20 20 20 20 22 70 72 69 63 65 22 3A 20 30 2E 32 35 36 BGs5bK77NsCFALMJkR@bitmessage.ch",      "price": 0.256
0x1070A 2C 0D 0A 20 20 20 20 22 70 72 69 63 65 5F 75 6E 69 74 22 3A 20 22 42 54 43 22 2C 0D 0A 20 20 20 20 22 6D 61 69 6E 5F 63 69 70 68 65 72 5F 6B 65 79 5F 73 69 7A 65 ,      "price_unit": "BTC",      "main_cipher_key_size
0x10740 22 3A 20 31 32 38 2C 0D 0A 20 20 20 20 22 73 65 73 73 69 6F 6E 5F 63 69 70 68 65 72 5F 6B 65 79 5F 73 69 7A 65 22 3A 20 36 34 2C 0D 0A 20 20 20 20 22 61 65 73 5F ": 128,      "session_cipher_key_size": 64,      "aes_
0x10776 63 69 70 68 65 72 5F 6B 65 79 5F 73 69 7A 65 22 3A 20 33 32 2C 0D 0A 20 20 20 20 22 74 61 72 67 65 74 5F 65 78 74 65 6E 73 69 6F 6E 73 22 3A 20 5B 0D 0A 20 20 20 cipher_key_size": 32,      "target_extensions": [
0x107AC 20 20 20 22 31 63 64 22 2C 0D 0A 20 20 20 20 20 20 22 33 64 6D 22 2C 0D 0A 20 20 20 20 20 20 22 33 64 73 22 2C 0D 0A 20 20 20 20 20 20 22 33 66 72 22 2C 0D 0A 20    "1cd",        "3dm",        "3ds",        "3fr",
0x107E2 20 20 20 20 20 22 33 67 32 22 2C 0D 0A 20 20 20 20 20 20 22 33 67 70 22 2C 0D 0A 20 20 20 20 20 20 22 33 70 72 22 2C 0D 0A 20 20 20 20 20 20 22 37 7A 22 2C 0D 0A      "3g2",        "3gp",        "3pr",        "7z",
0x10818 20 20 20 20 20 20 22 37 7A 69 70 22 2C 0D 0A 20 20 20 20 20 20 22 61 61 63 22 2C 0D 0A 20 20 20 20 20 20 22 61 62 34 22 2C 0D 0A 20 20 20 20 20 20 22 61 62 64 22       "7zip",        "aac",        "ab4",        "abd"
```

The config-file parameters clearly visible (highlighted in blue) in the executable using a simple hex-editor.

The config file contains the following parameters:

- Version number
- Features list
- Excluded countries

We extracted and compared all the config files from the Kraken samples that we found and built a large matrix.

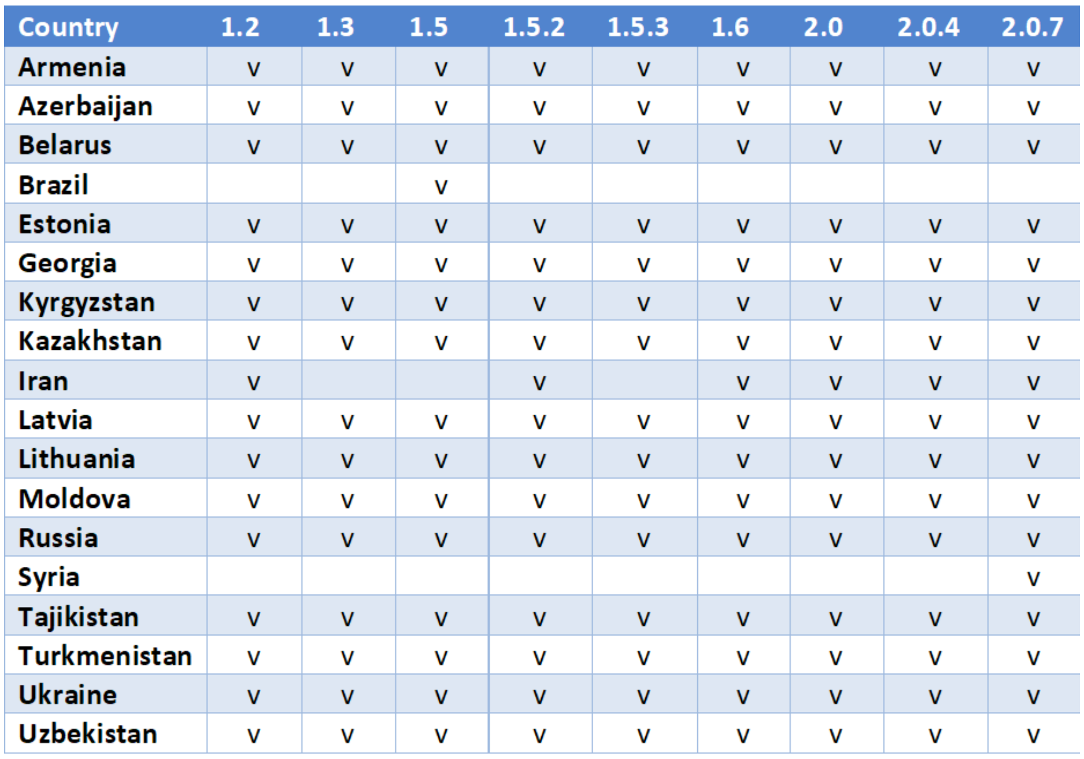


Features	1.2	1.3	1.5	1.5.2	1.5.3	1.6	2.0	2.0.4
Antiforensic		v	v	v	v		v	
Antireverse	v	v	v	v	v	v	v	v
Antivirtual	v				v			v
Anti-SMB								
Anti-RDP								
Country check	v	v	v	v	v	v	v	v
Keyboard check	v	v	v	v	v	v	v	v
Registry check	v	v	v	v	v	v	v	v
Fix device	v	v	v	v	v	v	v	v
Network device	v	v	v	v	v	v	v	v
Flash device	v	v	v	v	v	v	v	v
Extension bypass	v	v	v	v	v	v	v	v
Rapid mode	v	v	v	v	v	v	v	v

Kraken's different version numbers and features

All the versions we examined mostly contain the same options, changing only in some of them the anti-virtual protection and anti-forensic capabilities.

Other differences in Kraken's config file include the list of countries excluded from encryption. The standouts are Brazil and Syria, which were not named in the original forum advertisement. Having an exclusion list is a common method of cybercriminals to avoid prosecution. Brazil's addition to the list in Version 1.5 suggests the involvement of a Brazilian affiliate. The following table shows the exclusion list by country and version. (The √ means the country appears on the list.)



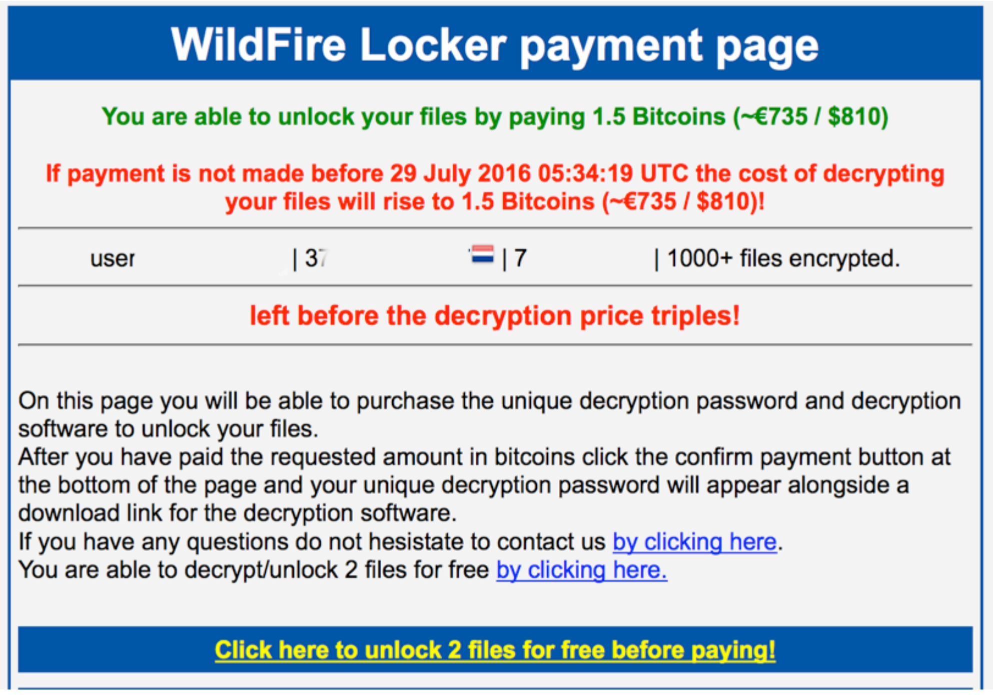
Country	1.2	1.3	1.5	1.5.2	1.5.3	1.6	2.0	2.0.4	2.0.7
Armenia	v	v	v	v	v	v	v	v	v
Azerbaijan	v	v	v	v	v	v	v	v	v
Belarus	v	v	v	v	v	v	v	v	v
Brazil			v						
Estonia	v	v	v	v	v	v	v	v	v
Georgia	v	v	v	v	v	v	v	v	v
Kyrgyzstan	v	v	v	v	v	v	v	v	v
Kazakhstan	v	v	v	v	v	v	v	v	v
Iran	v			v		v	v	v	v
Latvia	v	v	v	v	v	v	v	v	v
Lithuania	v	v	v	v	v	v	v	v	v
Moldova	v	v	v	v	v	v	v	v	v
Russia	v	v	v	v	v	v	v	v	v
Syria									v
Tajikistan	v	v	v	v	v	v	v	v	v
Turkmenistan	v	v	v	v	v	v	v	v	v
Ukraine	v	v	v	v	v	v	v	v	v
Uzbekistan	v	v	v	v	v	v	v	v	v

Kraken's list of excluded countries per version

A detailed analysis of Kraken can be found on the McAfee's research blog page.

Wildfire

Another Ransomware family that used a built-in configuration option and had a list of excluded countries was Wildfire ransomware. Wildfire was targeting small-to-medium business owners in the Netherlands and Belgium.



WildFire Locker payment page

You are able to unlock your files by paying 1.5 Bitcoins (~€735 / $810)

If payment is not made before 29 July 2016 05:34:19 UTC the cost of decrypting your files will rise to 1.5 Bitcoins (~€735 / $810)!

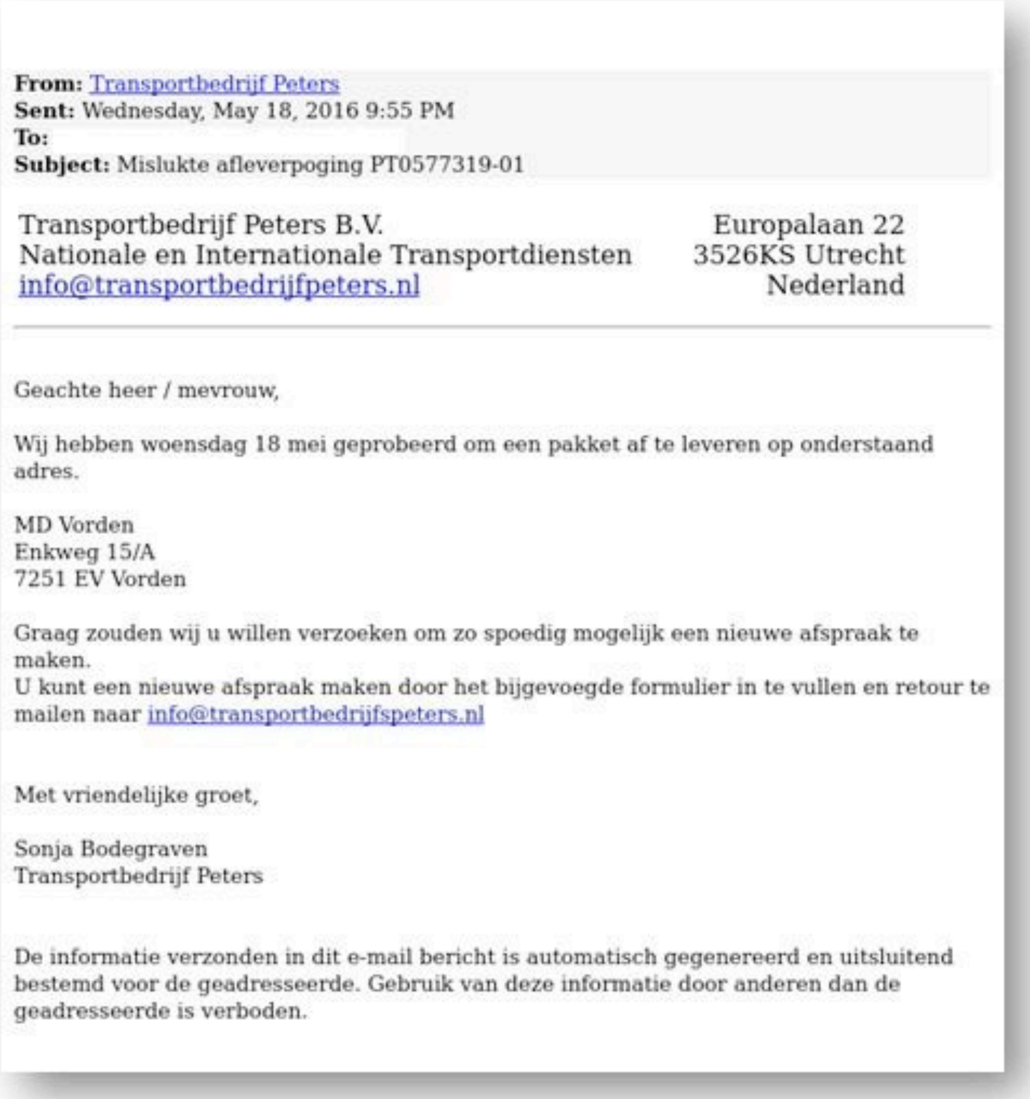
user | 37 | 7 | 1000+ files encrypted.

left before the decryption price triples!

On this page you will be able to purchase the unique decryption password and decryption software to unlock your files.
After you have paid the requested amount in bitcoins click the confirm payment button at the bottom of the page and your unique decryption password will appear alongside a download link for the decryption software.
If you have any questions do not hesitate to contact us by clicking here.
You are able to decrypt/unlock 2 files for free by clicking here.

The Wildfire lock-screen

Wildfire was spread by phishing mail in flawless Dutch and extra convincing was that the criminals included the actual victim's name and business address.



From: Transportbedrijf Peters
Sent: Wednesday, May 18, 2016 9:55 PM
To:
Subject: Mislukte afleverpoging PT0577319-01

Transportbedrijf Peters B.V.
Nationale en Internationale Transportdiensten
info@transportbedrijfpeters.nl

Europalaan 22
3526KS Utrecht
Nederland

Geachte heer / mevrouw,

Wij hebben woensdag 18 mei geprobeerd om een pakket af te leveren op onderstaand adres.

MD Vorden
Enkweg 15/A
7251 EV Vorden

Graag zouden wij u willen verzoeken om zo spoedig mogelijk een nieuwe afspraak te maken.
U kunt een nieuwe afspraak maken door het bijgevoegde formulier in te vullen en retour te mailen naar info@transportbedrijfspeters.nl

Met vriendelijke groet,

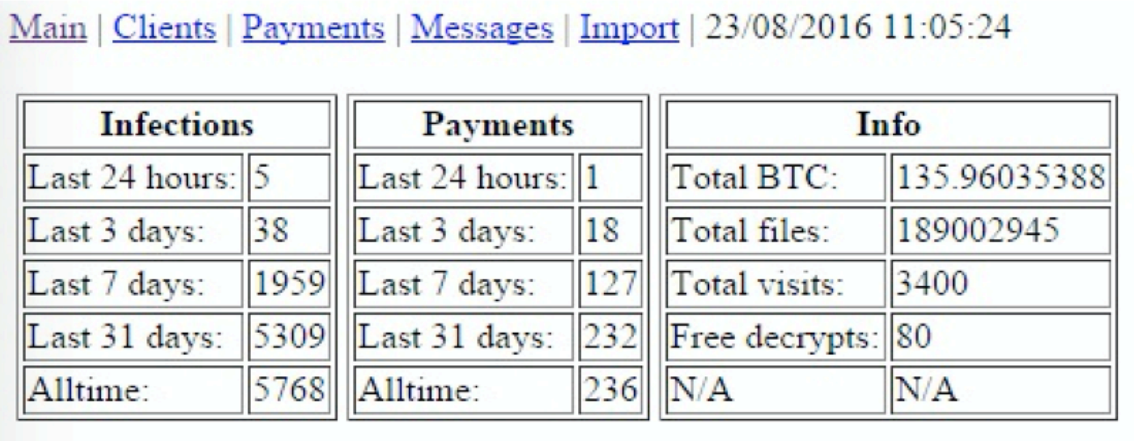
Sonja Bodegraven
Transportbedrijf Peters

De informatie verzonden in dit e-mail bericht is automatisch gegenereerd en uitsluitend bestemd voor de geadresseerde. Gebruik van deze informatie door anderen dan de geadresseerde is verboden.

One of the phishing mails used to spread Wildfire ransomware.

Wildfire was active mid 2016 when Dutch High Tech Crime unit received a report from the private sector that the ransomware payment site with the decryption keys was located in the Netherlands. We started an investigation and pretty quickly we seized a copy of the specific server. The server actually happened to be the command and control server for the ransomware and it had a web portal for the criminals to log into and see their infection statistics. Below is an overview of the web panel they used.

We see from this overview that in 31 days the campaign has infected 5,309 systems and earned total revenue of about BTC 136. Not a bad "paycheck" for a month of work.

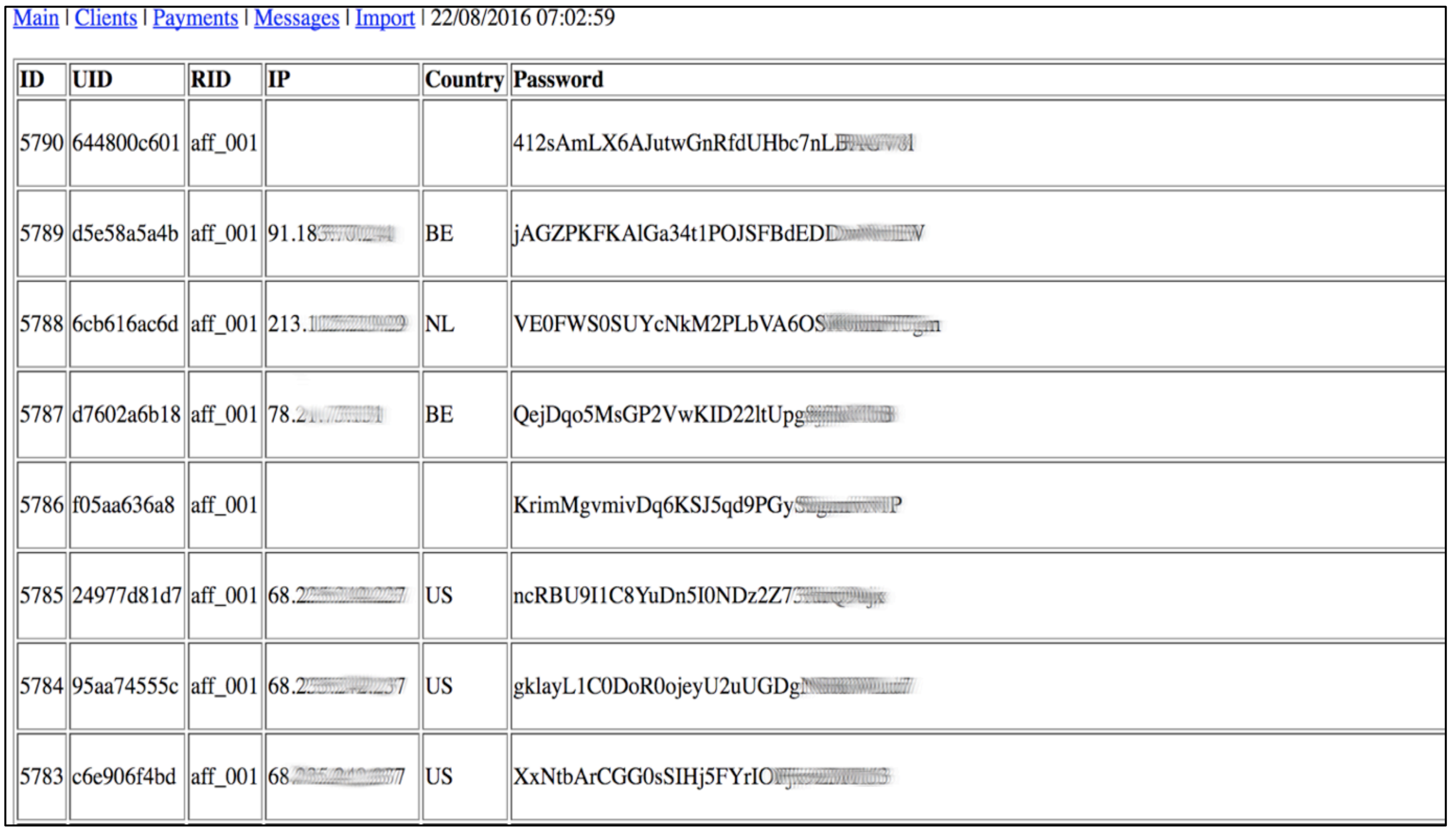


Main | Clients | Payments | Messages | Import | 23/08/2016 11:05:24

Infections		Payments		Info	
Last 24 hours:	5	Last 24 hours:	1	Total BTC:	135.96035388
Last 3 days:	38	Last 3 days:	18	Total files:	189002945
Last 7 days:	1959	Last 7 days:	127	Total visits:	3400
Last 31 days:	5309	Last 31 days:	232	Free decrypts:	80
Alltime:	5768	Alltime:	236	N/A	N/A

Wildfire's web panel that shows the overview of infections and payments

When we look at the "clients" page, we see that every victim is assigned a Unique Identifier (UID), their IP-address and country is stored as well as their private key which is used to decrypt their locked files.

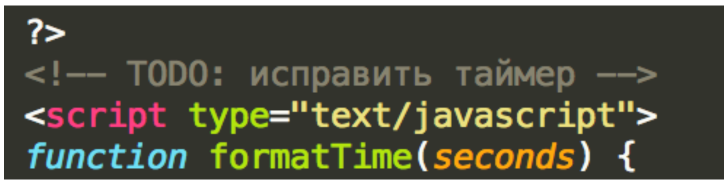


Main | Clients | Payments | Messages | Import | 22/08/2016 07:02:59

ID	UID	RID	IP	Country	Password
5790	644800c601	aff_001			412sAmLX6AJutwGnRfdUHbc7nL[redacted]
5789	d5e58a5a4b	aff_001	91.18[redacted]	BE	jAGZPKFKAlGa34t1POJSFBdEDD[redacted]
5788	6cb616ac6d	aff_001	213.1[redacted]	NL	VE0FWS0SUYcNkM2PLbVA6OS[redacted]
5787	d7602a6b18	aff_001	78.2[redacted]	BE	QejDqo5MsGP2VwKID22ltUpg[redacted]
5786	f05aa636a8	aff_001			KrimMgvmivDq6KSJ5qd9PGy[redacted]
5785	24977d81d7	aff_001	68.2[redacted]	US	ncRBU9I1C8YuDn5I0NDz2Z7[redacted]
5784	95aa74555c	aff_001	68.2[redacted]	US	gklayL1C0DoR0ojeyU2uUGDg[redacted]
5783	c6e906f4bd	aff_001	68[redacted]	US	XxNtbArCGG0sSIHj5FYrIO[redacted]

Web panel page with the different victims and decryption keys

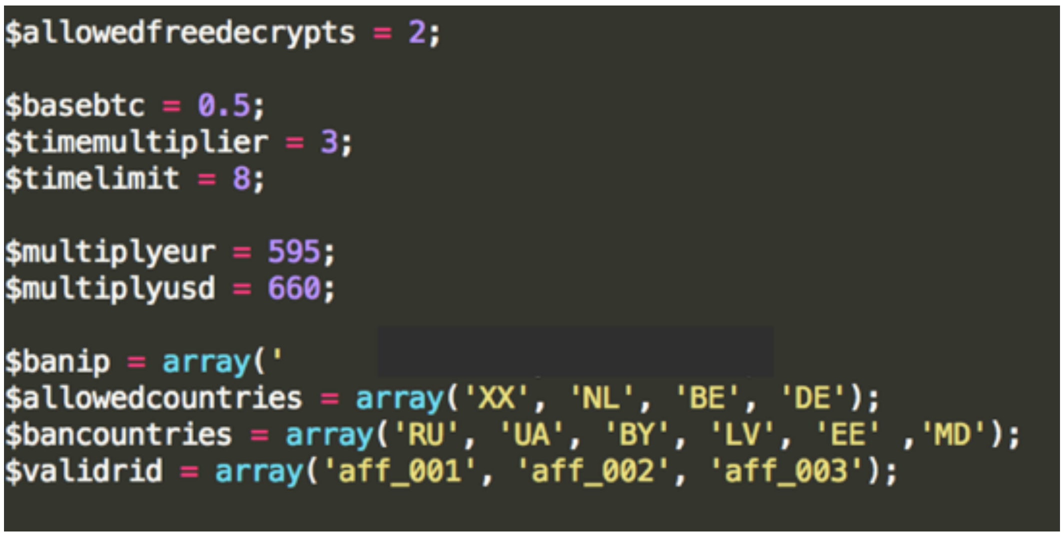
When we took a closer look at the machine itself, we found several files of interest. Something that stood out was the index.php file of the web panel, which contained comments in Russian Cyrillic in spite of the flawless Dutch used in the phishing mail. "исправить таймер" means "fix timer" and refers to the timer function of the ransomware.



```
?>
<!-- TODO: исправить таймер -->
<script type="text/javascript">
function formatTime(seconds) {
```

Cyrillic comments in the web panel source code

This is an indicator that the person who built Wildfire might not be the one spreading it, thus being an affiliate-based ransomware-as-a-service (RaaS), similar to Kraken and CTB-locker we discussed earlier. Another indicator to Wildfire being a RaaS can be found in the config file of the source code. This config-file contains a list of exempted countries that Wildfire will not encrypt when a victim is from a certain country. Wildfire based this exemption on the IP-address of the victim. Another common method used by ransomware for determining exemption is by looking at a victims keyboard layout settings in addition to the IP-address.



```
$allowedfreedecrypts = 2;

$basebtc = 0.5;
$timemultiplier = 3;
$timelimit = 8;

$multiplyeur = 595;
$multiplyusd = 660;

$banip = array(' [redacted]
$allowedcountries = array('XX', 'NL', 'BE', 'DE');
$bancountries = array('RU', 'UA', 'BY', 'LV', 'EE' ,'MD');
$validrid = array('aff_001', 'aff_002', 'aff_003');
```

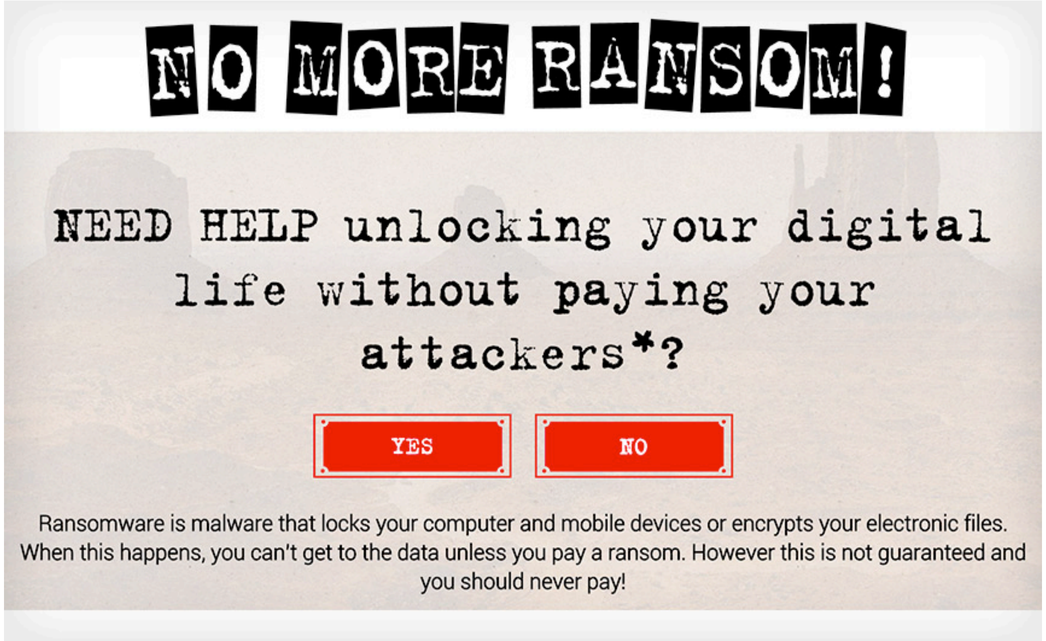
The list of allowed and banned countries in the ransomware config file.

Since we seized the complete server, we also obtained all the decryption keys. So now it was possible again to build another decryption tool with the help of the private sector. The decryption tool helped unlock about 20% of Wildfire's victims within the first month.

NoMoreRansom

The Coinvault investigation wasn't only a success because of the arrest of the criminal developers, but it was the start of something bigger.

The successful public-private partnership that developed the first decryptor could not be ignored and asked for an initiative on a much larger scale. This initiative was going to be called NoMoreRansom (www.nomoreransom.org) and was started July 2016 by the Dutch Police, Europol's European Cybercrime Centre (EC3) and two cyber security companies–Kaspersky Lab and McAfee.



NO MORE RANSOM!

NEED HELP unlocking your digital life without paying your attackers*?

Ransomware is malware that locks your computer and mobile devices or encrypts your electronic files. When this happens, you can't get to the data unless you pay a ransom. However this is not guaranteed and you should never pay!

The NoMoreRansom logo

The goal of NoMoreRansom is to help victims of ransomware retrieve their encrypted data without having to pay the criminals. This is achieved by law enforcement and the private sector working together building ransomware decryption tools based on decryption keys from seized servers, or flaws in the ransomware itself.

Today, NoMoreRansom has more than 129 partners and offers 59 free decryption tools that help decrypt more than 91 different ransomware families. Since the start, the initiative has already helped more than 72.000 infected computers, thus preventing more than 22 million US dollars from falling into criminal hands. It is the world's number 1 repository of ransomware decryption tools.

The most recent success was releasing a decryption tool for a large number of versions of the GandCrab ransomware variant. This tool alone already decrypted more than 4400 victims of GandCrab, with several hundreds of thousands of other victims who could be helped.

General Prevention Advice

Benjamin Franklin once said ***"An ounce of prevention is worth a pound of cure"*** - this saying also goes for ransomware.

Below are some useful ransomware prevention tips gathered from various sources. Please note that this is not a limitative list of tips and prevention/security is an ongoing process. As ransomware evolves, so do the prevention methods.

- **Back-up! Back-up! Back-up!** Have a recovery system in place so a ransomware infection can't destroy your personal data forever. It's best to create two back-up copies: one to be stored in the cloud (remember to use a service that makes an automatic backup of your files) and one to store physically (portable hard drive, thumb drive, extra laptop, etc.). Disconnect these from your computer when you are done. Your back-up copies will also come in handy should you accidentally delete a critical file or experience a hard drive failure.

- **Use robust antivirus software** to protect your system from ransomware. Do not switch off the 'heuristic functions' as these help the solution to catch samples of ransomware that have not yet been formally detected.

- **Network segmenting -** Create a defense in depth for your network, so ransomware can't easily propagate through your network once it has been activated.

- **Robust identity management -** Set-up policies on access control and consider multi-authentication on systems to prevent attackers from logging in with stolen passwords.

- **Keep all the software on your computer up to date.** When your operating system (OS) or applications release a new version, install it. And if the software offers the option of automatic updating, take it.

- **Trust no one. Literally.** Any account can be compromised, and malicious links can be sent from the accounts of friends on social media, colleagues or an online gaming partner. Never open attachments in emails from someone you don't know. Cybercriminals often distribute fake email messages that look very much like email notifications from an online store, a bank, the police, a court or a tax collection agency, luring recipients into clicking on a malicious link and releasing the malware into their system.

- **Enable the 'Show file extensions' option in the Windows settings on your computer.** This will make it much easier to spot potentially malicious files. Stay away from file extensions like '.exe', '.vbs' and '.scr'. Scammers can use several extensions to disguise a malicious file as a video, photo, or document (like hot-chics.avi.exe or doc.scr).

- If you discover a rogue or unknown process on your machine, **disconnect it immediately from the internet or other network connections (such as home Wi-Fi)** — this can prevent the infection from spreading.

- **Consider setting up a software restriction Policy (SRP)** Prevent executables from running in a user's %APPDATA% directory (TEMP and temporary Internet Files). A very large portion of ransomware runs from a user's TEMP directory, so preventing executables from running here is an effective method in stopping a large portion of ransomware. However, this might need some additional fine tuning to make sure that all legitimate programs are able to run normally. I would only advise to do this if you know what you are doing. More info on Software Restriction Policies can be found in this useful article.

General Ransomware Incident response tips

- Below are some useful ransomware incident response tips for the technical consumer or small businesses. A larger corporate environment often has a better disaster recovery policy and there are more incident response resources.

- Try to **create a memory dump** as soon as possible after you discover that you have been encrypted. DO NOT SHUT DOWN the computer before you have made a RAM capture. There might be valuable traces and even encryption keys left in the RAM memory. Dumpit is a great open source tool for this.
- **DO NOT delete any files** or ransom notes. Some families, like GandCrab, use the ransom note for decryption.
- **Take pictures** of the ransom note, e-mail addresses, Bitcoin addresses, etc. This will come in handy when determining the family of ransomware and when you want to file a police complaint.
- Try to **recover files with PhotoRec**. Some ransomware versions make a copy of the existing files, encrypt them and then delete the original files. A tool like PhotoRec might help you to recover some of the lost files.
- If you are planning on putting back a back-up or formatting your drive, it is best to buy a **new hard drive** and store the encrypted one. There is a good chance that a decryption option might come available in the future.
- **Check online** to determine which ransomware family has infected the machine. Knowing which ransomware you are facing is essential in finding a possible remediation. There are several useful resources online that can help for this such as:
  - ✦https://id-ransomware.malwarehunterteam.com/
  - ✦https://www.nomoreransom.org/crypto-sheriff.php?lang=en
  - ✦https://www.botfrei.de/de/ransomware/galerie.html
  - ✦https://www.bleepingcomputer.com/forums/f/239/ransomware-help-tech-support/
- Visit NoMoreransom.org to find out if there is a decryption tool available, how to file a police complaint and general tips.

About the Author

John Fokker is Head of Cyber Investigations for McAfee Advanced Threat Research. Prior to joining McAfee, he worked at the National High-Tech Crime Unit (NHTCU), the Dutch national police unit dedicated to investigating advanced forms of cybercrime. Within NHTCU he led the data science group, which focused on threat intelligence research. During his career he has supervised numerous large-scale cybercrime investigations and takedowns. Fokker is also one of the cofounders of the NoMoreRansom Project. He started his career with the Netherlands Police Agency as a digital forensics investigator within a task force against organized crime. Before joining the national police, he served in the special operations and counterterrorism group of the Royal Netherlands Marine Corps.

Twitter: @john_fokker.

Ransomware attacks in insurance claims

by Alistair Ewing & Jason Bergerson

As ransomware attacks become more common and impact global business critical equipment and systems, insurers must educate their clients. Businesses may be underwritten to some degree for business interruption and data loss, but rarely through a dedicated cyber policy. Often a firm will only to start to think about such requirements on the day it happens.

Recent headlines have highlighted significant cyber threats to global businesses. The US insurer Anthem paid a record $16 Million for a security breach when a hacker threatened to turn over patient records to the public. At the Arran Brewery in Scotland, bad actors used a call for résumés as a route to infection via a rogue PDF, causing the loss of 3 months of sales data. A water utility in North Carolina lost access to all automated systems from a virus dropped a week and a half after the initial Trojan attack. As ransomware attacks become more common and impact global business critical equipment and systems, insurers must educate their clients. Businesses may be underwritten to some degree for business interruption and data loss, but rarely through a dedicated cyber policy. Often a firm will only to start to think about such requirements on the day it happens.

Some good news is that the threat of being hit by a ransomware attack has decreased by about 30% from 2017 to 2018[1]. However, the level of sophistication for the new variants of attacks is increasing, and 75% of infected organisations had an up-to-date virus protection system in place[2]. The cost to the business affected by these attacks averaged around $133,000 in 2017[3] with the estimated global damages reaching $11.5 billion by 2019[4]. These are the real costs to the industry covering downtime, response and lost opportunities, and exclude the ransom that may, or may not, have been paid.

Essentially, ransomware is software that prevents proper access to computers and business systems by locking the data using encryption. The software then demands payment, or ransom, in a cryptocurrency such as Bitcoin to restore the system/data to operate normally. Imagine arriving into work and reading the words on the screen 'Your files have been locked'. As referred to above, getting back up and running can be costly. The firm must pay for restoring the computers or data to a pre-attack condition so that they can continue the business operation. These costs can come in the form of overtime that is needed to cover activities during the interruption, fees for the diagnosing firm, which is typically an outside consultancy firm, additional monitoring of the systems for new attacks, restoration of backups or timely recreation of lost data, and in some cases a complete overhaul of the equipment.

Historical forms of ransomware still allowed the user control of the system, and although they had to battle through annoying popups, the ransoms were typically under €100. More recently, however, the sophisticated ransomware attacks have become more targeted to publicly traded companies and critical infrastructures such as water plants and hospitals, with the amounts requested starting at around €100,000 and going into the millions. Some can gain entry through phishing, an Adobe Flash vulnerability, RDP and numerous other methods.

When it comes to adjusting a cyber-claim, it can be complicated and confusing. Having a high-level process to use can help bring some clarity, and streamline towards a successful claim investigation. While it isn't possible to create a script that will work for every claim, there are a few steps and questions that can be used to put you onto the correct path.

The initial goal is to understand the impact of the breach and determine the scope of what was affected. In order to gain a basis of understanding to what will transpire during the investigation, there are a number of critical elements that need to be identified. This stage is mostly about gathering information on how the breach was detected. Questions such as: "Who identified the event and how?", "What actions were observed within the network?", and "What alerts were presented?" should be used to create the observed chain of events before and during the attack.

Also at this stage, it is necessary to gain an understanding of the infrastructure of the organisation. This will help with the type and extent of infiltration being reviewed. Other questions include: "What sort of network technology and topology is being used?", "What type of network and physical security is in place?", "How is the security monitored?", and "How many workstations are on the network?" These are details of the environment that need to be understood that will lead to developing a practical approach on how to investigate the event properly.

Now that the background information has been initially assessed, it is essential to determine the infection and level of access reached quickly. There are thousands of different types of attacks that can happen and knowing what kind was used will be key to work through the event successfully. Here it is essential to understand what may have been compromised, the extent of the damage, what measures were used to determine the type of attack, and how the loss affects the insured and other parties.

Areas of consideration are determining if antivirus/antimalware was used, if and how much data was taken, how many of the systems were infected, what level of access was granted, and what was the method of attack. The various forms of attack such as phishing, malware, unsupported software, and password attacks all present themselves differently. Knowing the defences and attack types will lead to an understanding of the level of access gained and resulted from damage, as well as the remediation required.

It is best to proceed with caution if an investigation shows that data may have been lost or exfiltrated by a third-party, primarily financial, private, or protected information. Personally Identifiable Information (PII), Protected Health Information (PHI) and Intellectual Property (IP) can all be areas that may require regulatory compliance reporting and/or legal action. This is also the stage when the decision should be made to bring in a qualified forensic investigation firm. There may be requirements to retain information for analysis and testimony, to verify that no further breach occurred, to determine cause and effect of the event accurately, and to quantify costs to bring the insured back to a pre-loss condition.

The specific details of each claim will be different, but they should all mostly follow this rough process we have described. The best way to illustrate this process is by using a couple of case studies.

Case Example #1 – Manufacturer's distribution plant finds that they can't access their ordering system.

Ordering systems are vital to productivity, and without them, the distribution warehouse computer database is paralysed. It was found that an employee had opened a phishing email a month prior, which compromised one of their servers. As it was a large firm, the vulnerability was posted for sale to the Darknet making it available to any cybercriminal. The malware was later executed on 13 other computers in the network. During the inactivity of night, the bad actor then used remote desktop protocol (RDP) to connect through to the remaining 12 servers and ran encryption to block access to the data. The result was thousands of encrypted files, Microsoft Exchange emails encrypted and unusable, the booking system compromised and their product storage allocation data no longer accessible.

The systems were running an antivirus solution. However, this wasn't kept up to date or used to scan the computers regularly, therefore allowing the threat to escalate freely. Additionally, RDP connections were allowed on most of the machines from unsecured sources and were internet facing, enabling the risk to spread to 13 servers. Disabling this option or limiting the RDP capabilities may have mitigated the threat to only one server. Given the configuration in place, full administrator access was gained through a weak RDP password, and the danger was successful in accessing the network.

Intellectual property and PII was available to the attacker allowing further leverage for the attacker to increase the threat by intimidating the firm with the additional danger that the details would be sold to the highest bidder, or worse, published to the public on the dark web or similar. In addition, the backups were networked and encrypted too. There were no backups available on the cloud, and any remaining physical tapes that existed were all out of date, causing a loss of data.

As a result of the data loss, the company decided that the ransom of 20 Bitcoins, around €200,000 at the time of this writing, was to be paid due to the fact it would have cost them more than this amount in business interruption and lost orders.

Although this business' contingency plan was non-existent, the recovery was reasonably efficient. The most critical item was the business development and operations platform. Even after paying the ransom, due to the modular dependencies of software, it was not possible to reconstruct the full database successfully. Luckily, the firm that produced their software had a copy that was used to rebuild the interface to the database. The software interface was restored, and the newly decrypted database was reconnected, allowing them to take orders, access current instructions and so forth.

Case Example #2 – US University PII Compromise

A Microsoft Windows Workstation was infected with a backdoor, remote access Trojan (RAT) that allowed uninhibited administrative access. The infection arrived by way of malicious code attached to one of the emails. Once installed, it used the RAT to open a specific Ethernet port providing access to the bad actor(s).

Software utilities such as file compression utilities, network scanners, and name server scanners were identified as being installed. Also, other possible services, such as password harvesters, were found to have artefacts available to confirm their use. The harvesters were in place to mainly gather typical office documents and PDF type extensions.

Additional computers had keyloggers installed and were utilised to harvest credentials. Domain administrator credentials were stolen and used to take over 21 systems within the network. Another 12 systems were compromised by introducing a web-based threat to a web server, a web variant of malware that a web-browser would provide access to. This allowed the risk to spread laterally throughout the network.

The bad actor completed searches in an attempt to steal more data from the various campuses. A PowerShell utility was used to obtain Active Directory and local password data from the Active Directory database. The Active Directory is the way Microsoft Windows authenticates users and, once this is compromised, the whole network is compromised.

The bad actors didn't aim to disrupt the day to day business of the university but instead harvested details. As the bad actors were careful not to do any activity that would be detected, this allowed lateral movement through the network. Once the breach was identified, a minimum of 53 systems had been accessed, and the remediation efforts required hundreds of thousands of euros, and many hours of lost time, to rectify the situation.

By understanding some of the more common types of attacks, and how ransomware, in particular, is utilised, the process of adjusting a claim can be streamlined.

A single user clicking on an unscrupulous link is usually the route that enables malware to execute on a single system. Other common ways are through a URL link in an Office document, using a PDF embedded in an email file inside another email to evade detection or macros inside Microsoft Office documents. Macros execute commands within a document, such as sending emails or running executables, and by merely opening and running the macros this can install or create the conditions to install ransomware.

Weak RDP password credentials, allowing connections without network level authentication, and allowing RDP connections can enable the threat to spread laterally instead of isolating it to one workstation. In our experience, the antivirus programs on workstations are usually weak, non-existent, non-reactive, not kept updated or a mixture of these factors.

When it comes to the time taken to effect recovery, we have found that on many occasions offline backups fail to exist or are out of date. Having close to real-time backups stored in a way that will isolate them from a ransomware event can circumnavigate the need to pay the ransom in order to access the encrypted data.

In the cases that we have discussed, the ransom was paid as the loss financially, and hours lost to the business would be considerably more than the ransom sum. This lack of preparedness by companies creates the opportunity for system hackers, and the need for ransom payments thus encouraging more attacks in the future.

Contingency planning, vetting security firms that can aid before and after an attack, staff training and discussing options for obtaining an additional, more specific cyber policy can all soften the impact, or even stop altogether, the spectre of ransomware that looms over the global business institution.

[1] KSN Report: Ransomware and malicious cryptominers 2016-2018, www.kaspersky.com

[2] Businesses Impacted by Repeated Ransomware Attacks and Failing to Close the Gap on Exploits, According to Sophos Global Survey, https://www.sophos.com/en-us/press-office/press-releases/2018/01/businesses-impacted-by-repeated-ransomware-attacks-according-to-sophos-global-survey.aspx, Oxford, U.K., January 30, 2018

[3] Understanding ransomware and the impact of repeated attacks, https://news.sophos.com/en-us/2018/02/01/understanding-ransomware-and-the-impact-of-repeated-attacks/, Matthew Phillion, January, 2, 2018

[4] Global Ransomware Damages Costs Predicted to Exceed $8 Billion in 2018, https://cybersecurityventures.com/global-ransomware-damage-costs-predicted-to-exceed-8-billion-in-2018/, Steve Morgan, June 28, 2018

About Alistair

Alistair Ewing has over eight years of experience in Digital Forensic Analysis, Data Recovery, Mobile Phone Forensics, Litigation Support, and has served as an Expert Witness in criminal and civil cases in the UK. Mr Ewing began performing digital forensics in 2011 and has had hundreds of hours of experience in this sector. Qualified as an expert witness for some years and vetted by Sweet and Maxwell he has presented evidence in tribunals, civil and criminal courts in the UK and been involved in corporate investigations, litigation support and collections.

https://www.envistaforensics.com/experts/Alistair/Ewing/

About Jason

Jason Bergerson has over twenty years of digital forensic experience with over 20 years at Kroll Ontrack. Jason has worked on cases involving fraud, IOC, murder terrorism, data recovery and more recently cyber attacks involving ransomware.

https://www.linkedin.com/in/jason-bergerson-9513182/

Frogo ransomware memory analysis

by Paulo Henrique Pereira

This article discusses the difficulties encountered in performing the memory analysis of a Windows Server 2008 R2 machine apparently infected by ransomware. A company based in São Paulo called us in a case in which its database server had been infected by malware. The infection, unfortunately, had encrypted the files that contained the client data. There was no backup of these files.

I. The cyber crime scene

This article discusses the difficulties encountered in performing the memory analysis of a Windows Server 2008 R2 machine apparently infected by ransomware. A company based in São Paulo called us in a case in which its database server had been infected by malware. The infection, unfortunately, had encrypted the files that contained the client data. There was no backup of these files.

In terms of an analysis of malicious artifacts, it is usually sought to follow an investigative methodology for gathering evidence. In cases involving ransomware, generally, the artifacts are the encrypted files (which in this case have been found) and files that are related to the creation of public and private keys. Not in this case.

An investigative methodology proposed by SANS makes the following points:

- Identify rogue processes
- Analyze process DLLs and handles
- Review network artifacts
- Look for evidence of code injection
- Check for signs of rootkit
- Dump suspicious processes and drivers

This methodology may or may not be followed strictly by the forensic analyst, and in the case we are investigating, we chose to follow what was possible with this methodology because we were not at the crime scene at the time of the attack and we were not the ones who captured the memory of the compromised machine. So, in the moment of the investigative work, we decided to identify rogue processes and dump suspicious processes.

II. The infection

The malware actually featured a vector type that created a **.frogo** extension in each file, encrypting the data in its directories, and is believed to have uninstalled itself right after installation, since no evidence of the machine was found.

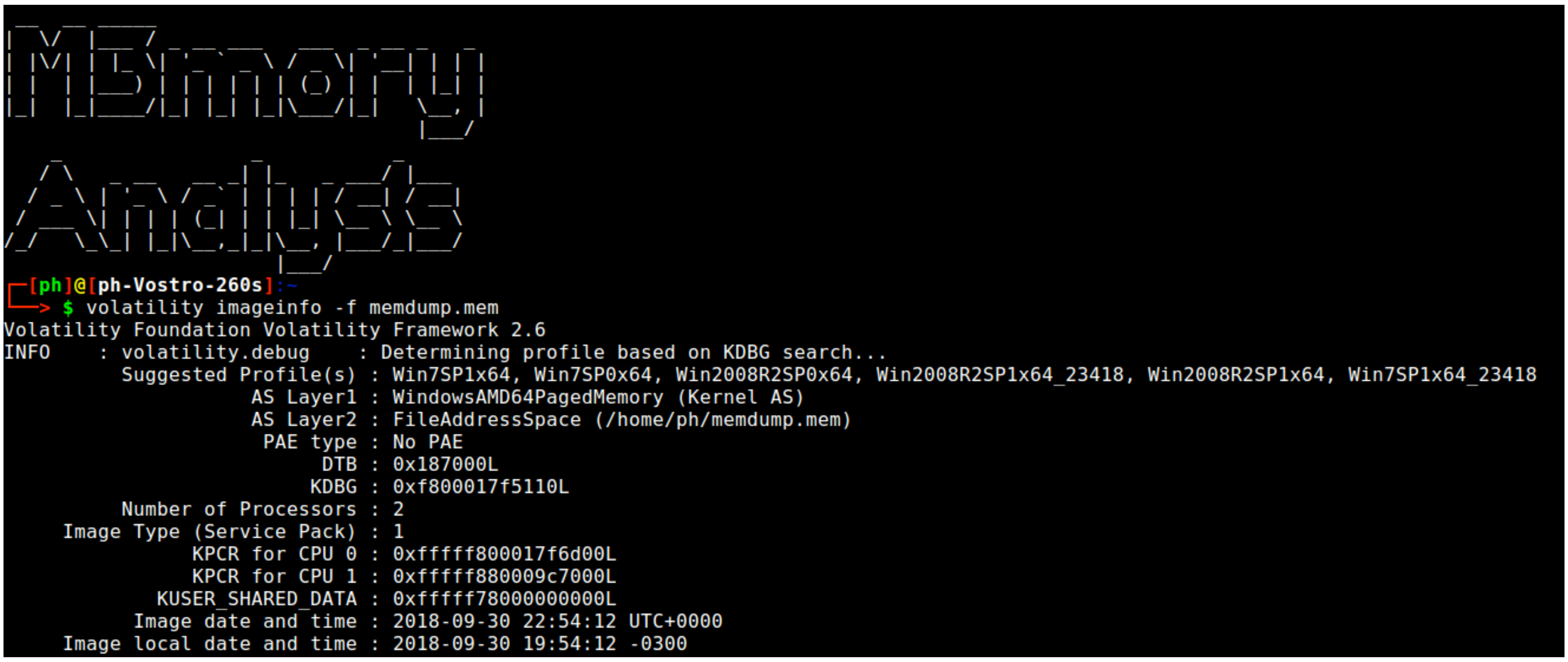
III. Approach for analyzing memory artifacts and extracting data from files

The type of extension was not known to us. Site searches have been conducted for a preliminary survey on this type of artifact. However, the scarcity of information about this ransomware did not allow us to find tools to decrypt the files. The initial approach was to perform forensic analysis of files and the compromised machine by capturing memory. An important detail is that the memory capture was done five days after the attack, and the compromised machine being rebooted in that time by the company's employees (in an attempt to recover the data).

For the analysis of the captured memory, we used **Volatility** (on a Linux machine) and for extracting data from files, we used **Autopsy** and **bulk_extractor** (both on a Windows machine). Memory capture was done using the **FTK Imager Lite**.

IV. Using Volatility for Forensic Investigation

The restricted access that we initially had to understand what was happening was only diminished when we received the captured memory of the compromised machine. So the preliminary data that could reveal some trace of the contamination began to be investigated. We started with Volatility to find out something about the machine. The first result in the figure below reveals the operating system and memory capture date.

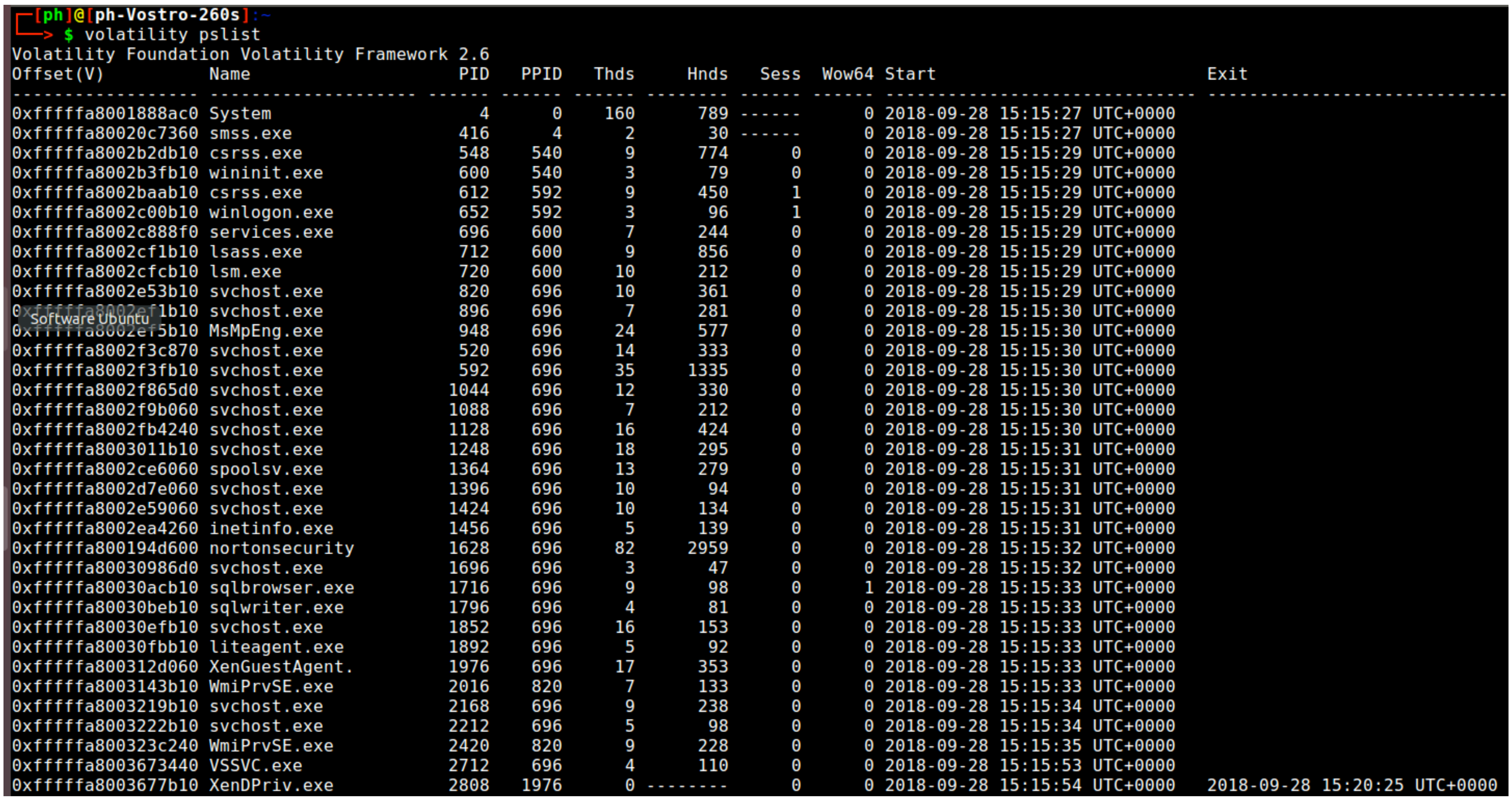


```
┌─[ph]@[ph-Vostro-260s]:~
└──> $ volatility imageinfo -f memdump.mem
Volatility Foundation Volatility Framework 2.6
INFO    : volatility.debug    : Determining profile based on KDBG search...
          Suggested Profile(s) : Win7SP1x64, Win7SP0x64, Win2008R2SP0x64, Win2008R2SP1x64_23418, Win2008R2SP1x64, Win7SP1x64_23418
                     AS Layer1 : WindowsAMD64PagedMemory (Kernel AS)
                     AS Layer2 : FileAddressSpace (/home/ph/memdump.mem)
                      PAE type : No PAE
                           DTB : 0x187000L
                          KDBG : 0xf800017f5110L
          Number of Processors : 2
     Image Type (Service Pack) : 1
                KPCR for CPU 0 : 0xfffff800017f6d00L
                KPCR for CPU 1 : 0xfffff880009c7000L
             KUSER_SHARED_DATA : 0xfffff78000000000L
           Image date and time : 2018-09-30 22:54:12 UTC+0000
     Image local date and time : 2018-09-30 19:54:12 -0300
```

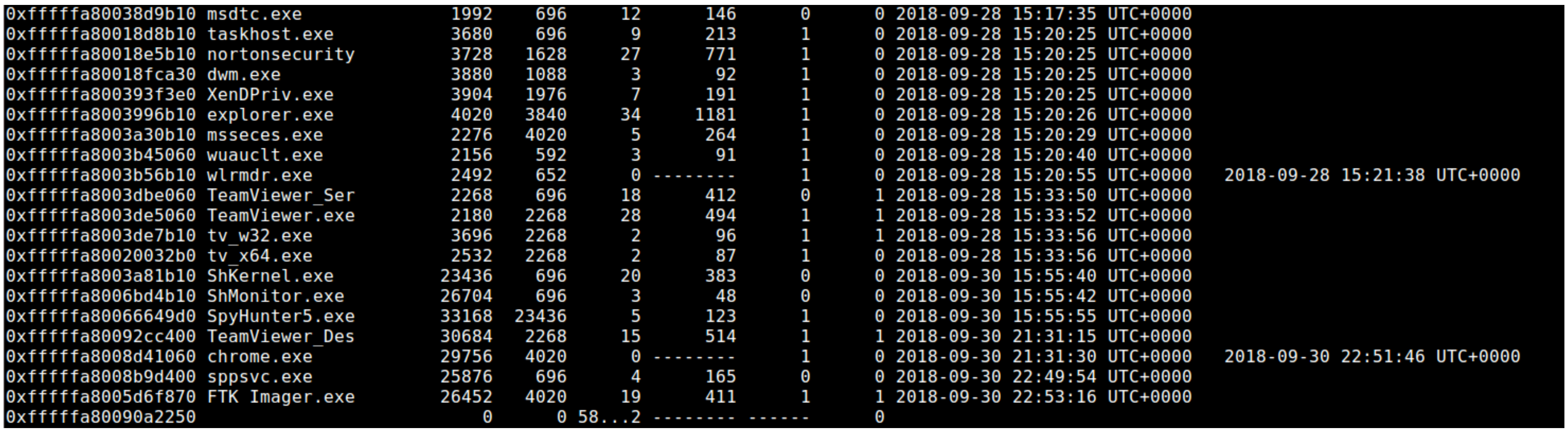
As you can see, the operating system has **Service Pack 1**, so we followed the investigation with this profile looking for rogue processes still active on the machine. Our biggest problem is that memory was captured five days after the infection started.

IV.a Searching for rogue process

As mentioned by SANS methodology approach, below is the output of the **pslist** command in an attempt to find evidence of processes that are extraneous to the operating system.

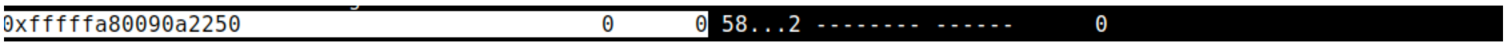


```
┌─[ph]@[ph-Vostro-260s]:~
└──> $ volatility pslist
Volatility Foundation Volatility Framework 2.6
Offset(V)          Name                    PID   PPID   Thds     Hnds   Sess  Wow64 Start                          Exit
------------------ -------------------- ------ ------ ------ -------- ------ ------ ------------------------------ ------------------------------
0xfffffa8001888ac0 System                    4      0    160      789 ------      0 2018-09-28 15:15:27 UTC+0000
0xfffffa80020c7360 smss.exe                416      4      2       30 ------      0 2018-09-28 15:15:27 UTC+0000
0xfffffa8002b2db10 csrss.exe               548    540      9      774      0      0 2018-09-28 15:15:29 UTC+0000
0xfffffa8002b3fb10 wininit.exe             600    540      3       79      0      0 2018-09-28 15:15:29 UTC+0000
0xfffffa8002baab10 csrss.exe               612    592      9      450      1      0 2018-09-28 15:15:29 UTC+0000
0xfffffa8002c00b10 winlogon.exe            652    592      3       96      1      0 2018-09-28 15:15:29 UTC+0000
0xfffffa8002c888f0 services.exe            696    600      7      244      0      0 2018-09-28 15:15:29 UTC+0000
0xfffffa8002cf1b10 lsass.exe               712    600      9      856      0      0 2018-09-28 15:15:29 UTC+0000
0xfffffa8002cfcb10 lsm.exe                 720    600     10      212      0      0 2018-09-28 15:15:29 UTC+0000
0xfffffa8002e53b10 svchost.exe             820    696     10      361      0      0 2018-09-28 15:15:29 UTC+0000
0xfffffa8002ef1b10 svchost.exe             896    696      7      281      0      0 2018-09-28 15:15:30 UTC+0000
0xfffffa8002ef5b10 MsMpEng.exe             948    696     24      577      0      0 2018-09-28 15:15:30 UTC+0000
0xfffffa8002f3c870 svchost.exe             520    696     14      333      0      0 2018-09-28 15:15:30 UTC+0000
0xfffffa8002f3fb10 svchost.exe             592    696     35     1335      0      0 2018-09-28 15:15:30 UTC+0000
0xfffffa8002f865d0 svchost.exe            1044    696     12      330      0      0 2018-09-28 15:15:30 UTC+0000
0xfffffa8002f9b060 svchost.exe            1088    696      7      212      0      0 2018-09-28 15:15:30 UTC+0000
0xfffffa8002fb4240 svchost.exe            1128    696     16      424      0      0 2018-09-28 15:15:30 UTC+0000
0xfffffa8003011b10 svchost.exe            1248    696     18      295      0      0 2018-09-28 15:15:31 UTC+0000
0xfffffa8002ce6060 spoolsv.exe            1364    696     13      279      0      0 2018-09-28 15:15:31 UTC+0000
0xfffffa8002d7e060 svchost.exe            1396    696     10       94      0      0 2018-09-28 15:15:31 UTC+0000
0xfffffa8002e59060 svchost.exe            1424    696     10      134      0      0 2018-09-28 15:15:31 UTC+0000
0xfffffa8002ea4260 inetinfo.exe           1456    696      5      139      0      0 2018-09-28 15:15:31 UTC+0000
0xfffffa800194d600 nortonsecurity         1628    696     82     2959      0      0 2018-09-28 15:15:32 UTC+0000
0xfffffa80030986d0 svchost.exe            1696    696      3       47      0      0 2018-09-28 15:15:32 UTC+0000
0xfffffa80030acb10 sqlbrowser.exe         1716    696      9       98      0      1 2018-09-28 15:15:33 UTC+0000
0xfffffa80030beb10 sqlwriter.exe          1796    696      4       81      0      0 2018-09-28 15:15:33 UTC+0000
0xfffffa80030efb10 svchost.exe            1852    696     16      153      0      0 2018-09-28 15:15:33 UTC+0000
0xfffffa80030fbb10 liteagent.exe          1892    696      5       92      0      0 2018-09-28 15:15:33 UTC+0000
0xfffffa800312d060 XenGuestAgent.         1976    696     17      353      0      0 2018-09-28 15:15:33 UTC+0000
0xfffffa8003143b10 WmiPrvSE.exe           2016    820      7      133      0      0 2018-09-28 15:15:33 UTC+0000
0xfffffa8003219b10 svchost.exe            2168    696      9      238      0      0 2018-09-28 15:15:34 UTC+0000
0xfffffa8003222b10 svchost.exe            2212    696      5       98      0      0 2018-09-28 15:15:34 UTC+0000
0xfffffa800323c240 WmiPrvSE.exe           2420    820      9      228      0      0 2018-09-28 15:15:35 UTC+0000
0xfffffa8003673440 VSSVC.exe              2712    696      4      110      0      0 2018-09-28 15:15:53 UTC+0000
0xfffffa8003677b10 XenDPriv.exe           2808   1976      0 --------      0      0 2018-09-28 15:15:54 UTC+0000   2018-09-28 15:20:25 UTC+0000
```



```
0xfffffa80038d9b10 msdtc.exe              1992    696     12      146      0      0 2018-09-28 15:17:35 UTC+0000
0xfffffa80018d8b10 taskhost.exe           3680    696      9      213      1      0 2018-09-28 15:20:25 UTC+0000
0xfffffa80018e5b10 nortonsecurity         3728   1628     27      771      1      0 2018-09-28 15:20:25 UTC+0000
0xfffffa80018fca30 dwm.exe                3880   1088      3       92      1      0 2018-09-28 15:20:25 UTC+0000
0xfffffa800393f3e0 XenDPriv.exe           3904   1976      7      191      1      0 2018-09-28 15:20:25 UTC+0000
0xfffffa8003996b10 explorer.exe           4020   3840     34     1181      1      0 2018-09-28 15:20:26 UTC+0000
0xfffffa8003a30b10 msseces.exe            2276   4020      5      264      1      0 2018-09-28 15:20:29 UTC+0000
0xfffffa8003b45060 wuauclt.exe            2156    592      3       91      1      0 2018-09-28 15:20:40 UTC+0000
0xfffffa8003b56b10 wlrmdr.exe             2492    652      0 --------      1      0 2018-09-28 15:20:55 UTC+0000   2018-09-28 15:21:38 UTC+0000
0xfffffa8003dbe060 TeamViewer_Ser         2268    696     18      412      0      1 2018-09-28 15:33:50 UTC+0000
0xfffffa8003de5060 TeamViewer.exe         2180   2268     28      494      1      1 2018-09-28 15:33:52 UTC+0000
0xfffffa8003de7b10 tv_w32.exe             3696   2268      2       96      1      1 2018-09-28 15:33:56 UTC+0000
0xfffffa80020032b0 tv_x64.exe             2532   2268      2       87      1      0 2018-09-28 15:33:56 UTC+0000
0xfffffa8003a81b10 ShKernel.exe          23436    696     20      383      0      0 2018-09-30 15:55:40 UTC+0000
0xfffffa8006bd4b10 ShMonitor.exe         26704    696      3       48      0      0 2018-09-30 15:55:42 UTC+0000
0xfffffa80066649d0 SpyHunter5.exe        33168  23436      5      123      1      0 2018-09-30 15:55:55 UTC+0000
0xfffffa80092cc400 TeamViewer_Des        30684   2268     15      514      1      1 2018-09-30 21:31:15 UTC+0000
0xfffffa8008d41060 chrome.exe            29756   4020      0 --------      1      0 2018-09-30 21:31:30 UTC+0000   2018-09-30 22:51:46 UTC+0000
0xfffffa8008b9d400 sppsvc.exe            25876    696      4      165      0      0 2018-09-30 22:49:54 UTC+0000
0xfffffa8005d6f870 FTK Imager.exe        26452   4020     19      411      1      1 2018-09-30 22:53:16 UTC+0000
0xfffffa80090a2250                           0      0 58...2 -------- ------      0
```

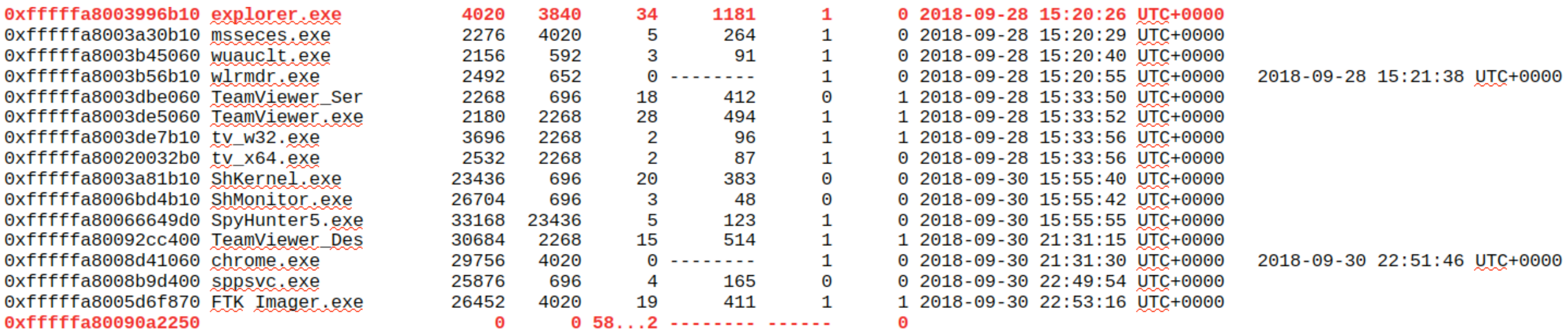
At first glance, what strikes us is a last process that was not finalized in the system, that is, it is still active and whose offset is:



```
0xfffffa80090a2250                           0      0 58...2 -------- ------      0
```

In summary, two pieces of evidence came to our attention: the first is the explorer.exe process whose PPID is not listed and the other is the offset **0xfffffa80090a2250** of a process with

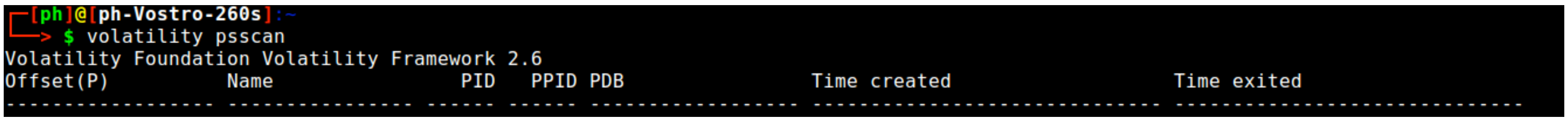
PID = 0 and PPID = 0.



```
0xfffffa8003996b10 explorer.exe          4020   3840     34     1181      1      0 2018-09-28 15:20:26 UTC+0000
0xfffffa8003a30b10 msseces.exe           2276   4020      5      264      1      0 2018-09-28 15:20:29 UTC+0000
0xfffffa8003b45060 wuauclt.exe           2156    592      3       91      1      0 2018-09-28 15:20:40 UTC+0000
0xfffffa8003b56b10 wlrmdr.exe            2492    652      0 --------      1      0 2018-09-28 15:20:55 UTC+0000   2018-09-28 15:21:38 UTC+0000
0xfffffa8003dbe060 TeamViewer_Ser        2268    696     18      412      0      1 2018-09-28 15:33:50 UTC+0000
0xfffffa8003de5060 TeamViewer.exe        2180   2268     28      494      1      1 2018-09-28 15:33:52 UTC+0000
0xfffffa8003de7b10 tv_w32.exe            3696   2268      2       96      1      1 2018-09-28 15:33:56 UTC+0000
0xfffffa80020032b0 tv_x64.exe            2532   2268      2       87      1      0 2018-09-28 15:33:56 UTC+0000
0xfffffa8003a81b10 ShKernel.exe         23436    696     20      383      0      0 2018-09-30 15:55:40 UTC+0000
0xfffffa8006bd4b10 ShMonitor.exe        26704    696      3       48      0      0 2018-09-30 15:55:42 UTC+0000
0xfffffa80066649d0 SpyHunter5.exe       33168  23436      5      123      1      0 2018-09-30 15:55:55 UTC+0000
0xfffffa80092cc400 TeamViewer_Des       30684   2268     15      514      1      1 2018-09-30 21:31:15 UTC+0000
0xfffffa8008d41060 chrome.exe           29756   4020      0 --------      1      0 2018-09-30 21:31:30 UTC+0000   2018-09-30 22:51:46 UTC+0000
0xfffffa8008b9d400 sppsvc.exe           25876    696      4      165      0      0 2018-09-30 22:49:54 UTC+0000
0xfffffa8005d6f870 FTK Imager.exe       26452   4020     19      411      1      1 2018-09-30 22:53:16 UTC+0000
0xfffffa80090a2250                          0      0 58...2 -------- ------      0
```

No PID and no PPID. What is this process supposed to be? Is the ransomware signed on the machine? Is any process running by company employees? Of course, this process is a candidate to be examined more closely. Why does the explorer.exe appear without a PPID?

Trying to find answers to these questions, maybe a possible cause of hidden process, the next step is to launch the **psscan** plugin that can be used to check for rootkits and hidden artifacts. According to what is exemplified on the Volatility Command reference page, the plugin is used against a Windows 7 memory image, which suggests that it can also be used in a Windows Server 2008 R2 memory image. According to what is exemplified on the Volatility Command reference page, the plugin is used against a Windows 7 memory image, which suggests that it can also be used in a Windows Server 2008 R2 memory image. The plugin scanning the _POOL_HEADER tag. But, we did not find anything with **psscan**.

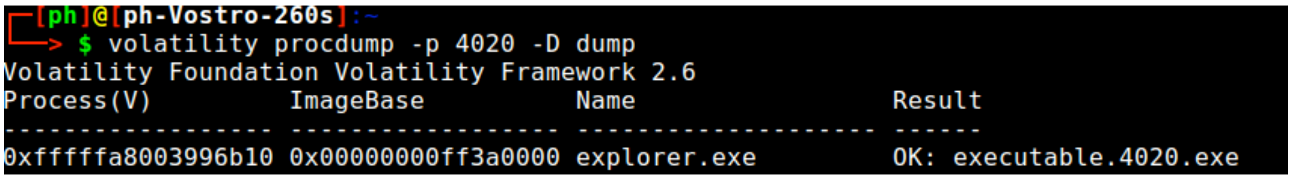


```
┌─[ph]@[ph-Vostro-260s]:~
└──> $ volatility psscan
Volatility Foundation Volatility Framework 2.6
Offset(P)          Name                PID   PPID PDB                Time created                   Time exited
------------------ ---------------- ------ ------ ------------------ ------------------------------ ------------------------------
```

Why did this occur? Is it a normal fact that in the Windows Server 2008 r2 memory image the **psscan** couldn't scan the _POOL_HEADER tag and show us a blank output? There are many unanswered questions.

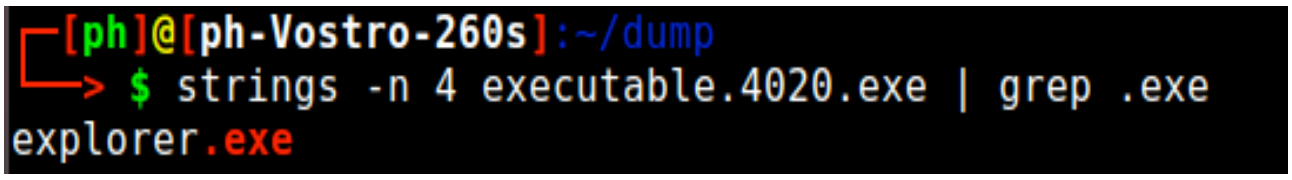
IV.b. Dumping the suspicious process

An attempt to find some answers was to extract the process through the procdump plugin. We created a folder called ***dump*** and the extraction was done. For the listed offset, we were unable to use **procdump**.



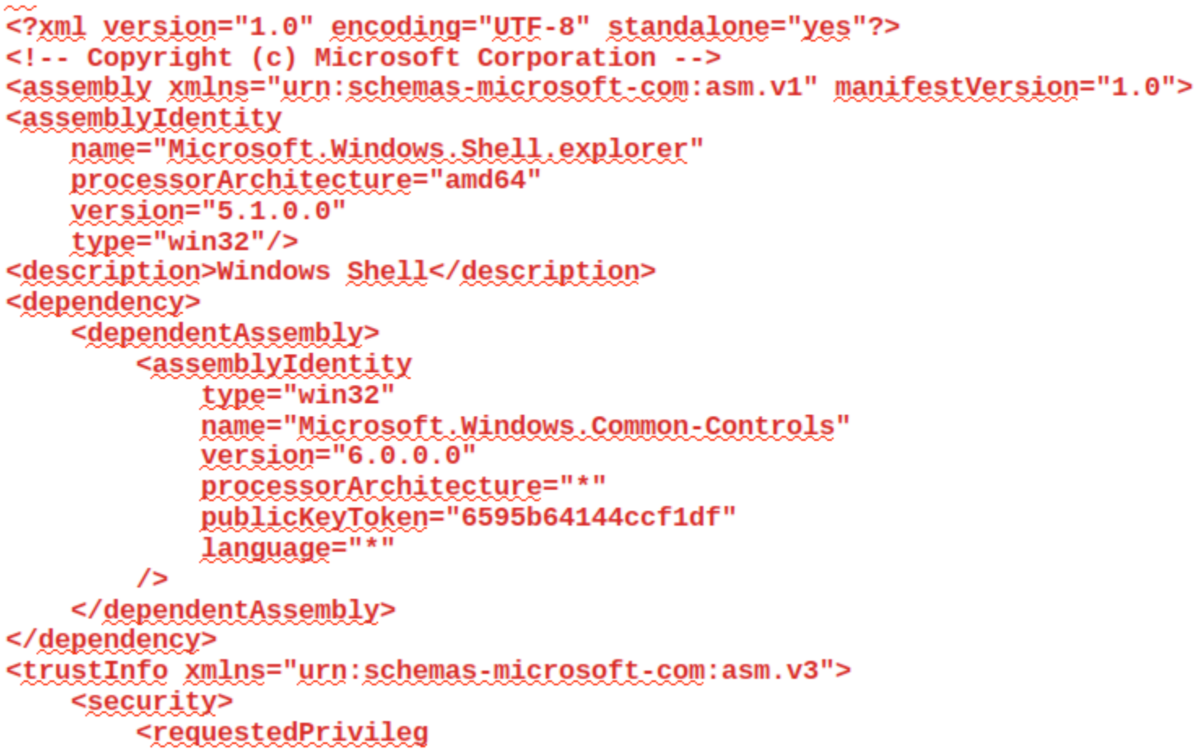
```
┌─[ph]@[ph-Vostro-260s]:~
└──> $ volatility procdump -p 4020 -D dump
Volatility Foundation Volatility Framework 2.6
Process(V)         ImageBase          Name                 Result
------------------ ------------------ -------------------- ------
0xffffa8003996b10 0x00000000ff3a0000 explorer.exe         OK: executable.4020.exe
```

After this step, the use of the **strings** tool with the **-n 4 | grep.exe** (see the complete command in the figure below) was made and the result was a single output: explorer.exe.



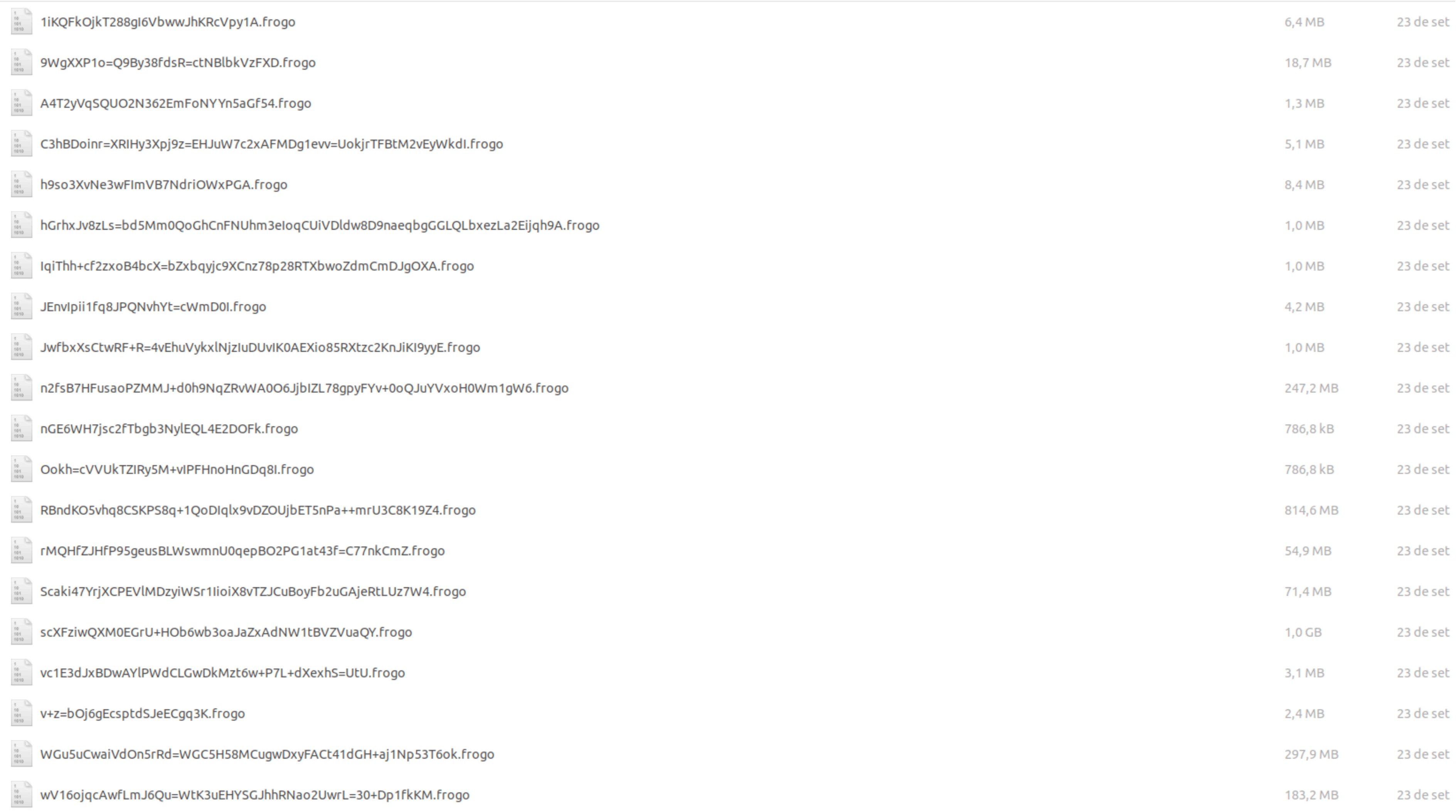
```
┌─[ph]@[ph-Vostro-260s]:~/dump
└──> $ strings -n 4 executable.4020.exe | grep .exe
explorer.exe
```

The **| grep** option just reveals the file explorer.exe. Using strings against the executable in the dump folder, without | grep, no significant textual records were found, such as passwords or keys (as in the case of Wannacry). We can see in the final output the C++ signature.



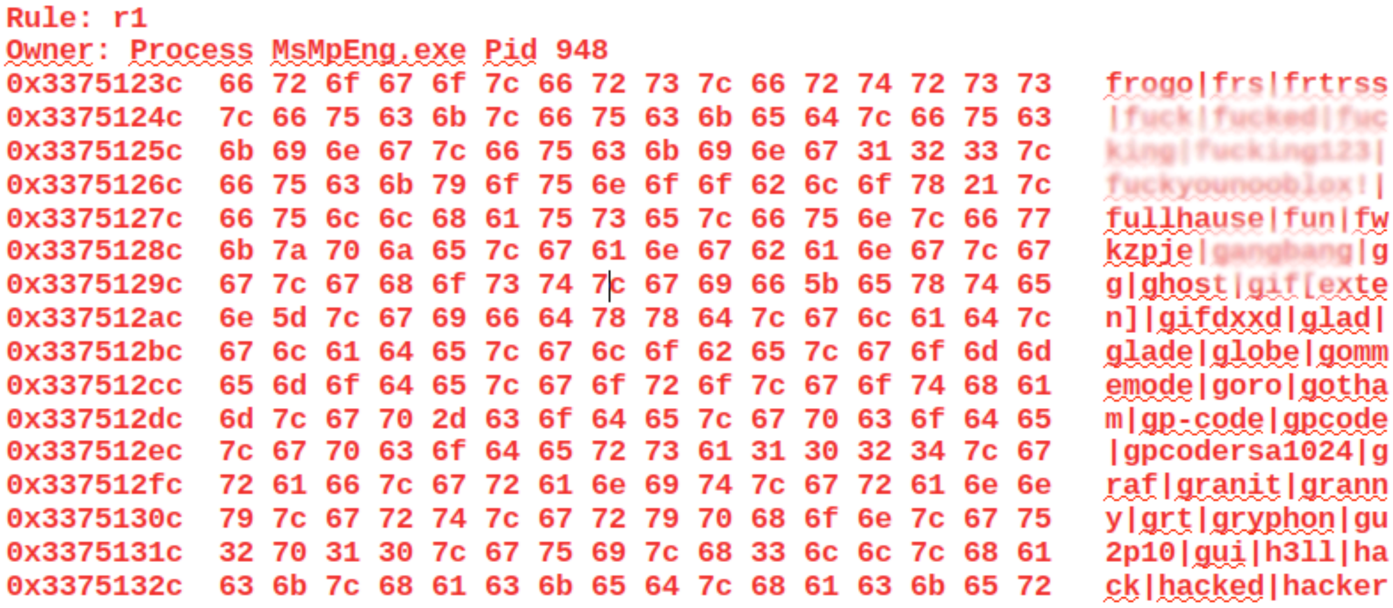
```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<!-- Copyright (c) Microsoft Corporation -->
<assembly xmlns="urn:schemas-microsoft-com:asm.v1" manifestVersion="1.0">
<assemblyIdentity
    name="Microsoft.Windows.Shell.explorer"
    processorArchitecture="amd64"
    version="5.1.0.0"
    type="win32"/>
<description>Windows Shell</description>
<dependency>
    <dependentAssembly>
        <assemblyIdentity
            type="win32"
            name="Microsoft.Windows.Common-Controls"
            version="6.0.0.0"
            processorArchitecture="*"
            publicKeyToken="6595b64144ccf1df"
            language="*"
        />
    </dependentAssembly>
</dependency>
<trustInfo xmlns="urn:schemas-microsoft-com:asm.v3">
    <security>
        <requestedPrivileg
```

Another way to look for traces is to use the **yarascan** plugin. The first attempt was made to use a **-Y "frogo"** rule *as a defining rule on the fly* to look for evidence of infection or, to some extent, some kind of record that might present the presence of this malware in the machine because the only demonstration of the presence of the infection is the extension of the encrypted files (figure below).



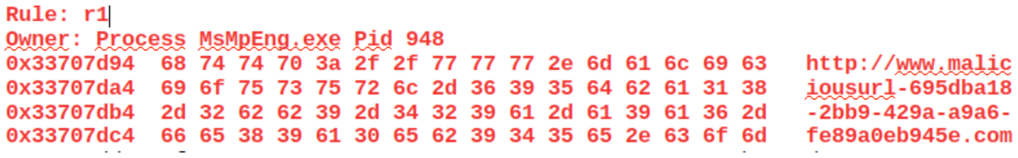
```
1iKQFkOjkT288gI6VbwwJhKRcVpy1A.frogo                                        6,4 MB     23 de set
9WgXXP1o=Q9By38fdsR=ctNBlbkVzFXD.frogo                                      18,7 MB    23 de set
A4T2yVqSQUO2N362EmFoNYYn5aGf54.frogo                                        1,3 MB     23 de set
C3hBDoinr=XRIHy3Xpj9z=EHJuW7c2xAFMDg1evv=UokjrTFBtM2vEyWkdI.frogo           5,1 MB     23 de set
h9so3XvNe3wFImVB7NdriOWxPGA.frogo                                           8,4 MB     23 de set
hGrhxJv8zLs=bd5Mm0QoGhCnFNUhm3eIoqCUiVDldw8D9naeqbgGGLQLbxezLa2Eijqh9A.frogo 1,0 MB     23 de set
IqiThh+cf2zxoB4bcX=bZxbqyjc9XCnz78p28RTXbwoZdmCmDJgOXA.frogo                1,0 MB     23 de set
JEnvIpii1fq8JPQNvhYt=cWmD0I.frogo                                           4,2 MB     23 de set
JwfbxXsCtwRF+R=4vEhuVykxlNjzIuDUvIK0AEXio85RXtzc2KnJiKI9yyE.frogo           1,0 MB     23 de set
n2fsB7HFusaoPZMMJ+d0h9NqZRvWA0O6JjbIZL78gpyFYv+0oQJuYVxoH0Wm1gW6.frogo      247,2 MB   23 de set
nGE6WH7jsc2fTbgb3NylEQL4E2DOFk.frogo                                        786,8 kB   23 de set
Ookh=cVVUkTZIRy5M+vIPFHnoHnGDq8I.frogo                                      786,8 kB   23 de set
RBndKO5vhq8CSKPS8q+1QoDIqlx9vDZOUjbET5nPa++mrU3C8K19Z4.frogo                814,6 MB   23 de set
rMQHfZJHfP95geusBLWswmnU0qepBO2PG1at43f=C77nkCmZ.frogo                      54,9 MB    23 de set
Scaki47YrjXCPEVlMDzyiWSr1IioiX8vTZJCuBoyFb2uGAjeRtLUz7W4.frogo              71,4 MB    23 de set
scXFziwQXM0EGrU+HOb6wb3oaJaZxAdNW1tBVZVuaQY.frogo                           1,0 GB     23 de set
vc1E3dJxBDwAYlPWdCLGwDkMzt6w+P7L+dXexhS=UtU.frogo                           3,1 MB     23 de set
v+z=bOj6gEcsptdSJeECgq3K.frogo                                              2,4 MB     23 de set
WGu5uCwaiVdOn5rRd=WGC5H58MCugwDxyFACt41dGH+aj1Np53T6ok.frogo                297,9 MB   23 de set
wV16ojqcAwfLmJ6Qu=WtK3uEHYSGJhhRNao2UwrL=30+Dp1fkKM.frogo                   183,2 MB   23 de set
```

So, **yarascan** can reveal something like (some offender words are founded and blanked in the figure):



```
Rule: r1
Owner: Process MsMpEng.exe Pid 948
0x3375123c  66 72 6f 67 6f 7c 66 72 73 7c 66 72 74 72 73 73   frogo|frs|frtrss
0x3375124c  7c 66 75 63 6b 7c 66 75 63 6b 65 64 7c 66 75 63   [redacted]
0x3375125c  6b 69 6e 67 7c 66 75 63 6b 69 6e 67 31 32 33 7c   [redacted]|
0x3375126c  66 75 63 6b 79 6f 75 6e 6f 6f 62 6c 6f 78 21 7c   [redacted]|
0x3375127c  66 75 6c 6c 68 61 75 73 65 7c 66 75 6e 7c 66 77   fullhause|fun|fw
0x3375128c  6b 7a 70 6a 65 7c 67 61 6e 67 62 61 6e 67 7c 67   kzpje|[redacted]|g
0x3375129c  67 7c 67 68 6f 73 74 7c 67 69 66 5b 65 78 74 65   g|ghost|gif[exte
0x337512ac  6e 5d 7c 67 69 66 64 78 78 64 7c 67 6c 61 64 7c   n]|gifdxxd|glad|
0x337512bc  67 6c 61 64 65 7c 67 6c 6f 62 65 7c 67 6f 6d 6d   glade|globe|gomm
0x337512cc  65 6d 6f 64 65 7c 67 6f 72 6f 7c 67 6f 74 68 61   emode|goro|gotha
0x337512dc  6d 7c 67 70 2d 63 6f 64 65 7c 67 70 63 6f 64 65   m|gp-code|gpcode
0x337512ec  7c 67 70 63 6f 64 65 72 73 61 31 30 32 34 7c 67   |gpcodersa1024|g
0x337512fc  72 61 66 7c 67 72 61 6e 69 74 7c 67 72 61 6e 6e   raf|granit|grann
0x3375130c  79 7c 67 72 74 7c 67 72 79 70 68 6f 6e 7c 67 75   y|grt|gryphon|gu
0x3375131c  32 70 31 30 7c 67 75 69 7c 68 33 6c 6c 7c 68 61   2p10|gui|h3ll|ha
0x3375132c  63 6b 7c 68 61 63 6b 65 64 7c 68 61 63 6b 65 72   ck|hacked|hacker
```

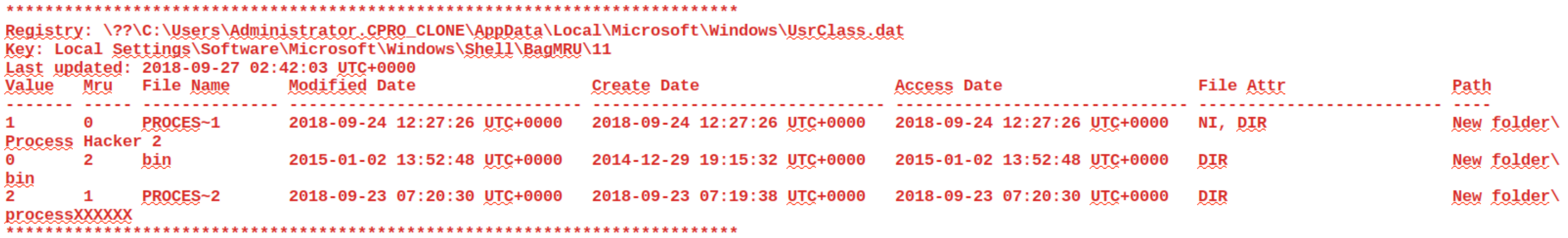
The "http" scanner option is applying the on the fly rule to the PID process = 948 (MsMpEng.exe) to find url:



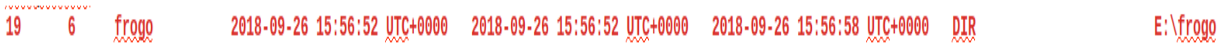
```
Rule: r1
Owner: Process MsMpEng.exe Pid 948
0x33707d94  68 74 74 70 3a 2f 2f 77 77 77 2e 6d 61 6c 69 63   http://www.malic
0x33707da4  69 6f 75 73 75 72 6c 2d 36 39 35 64 62 61 31 38   iousurl-695dba18
0x33707db4  2d 32 62 62 39 2d 34 32 39 61 2d 61 39 61 36 2d   -2bb9-429a-a9a6-
0x33707dc4  66 65 38 39 61 30 65 62 39 34 35 65 2e 63 6f 6d   fe89a0eb945e.com
```

According to Volatility Lab site, the following information can be found in Shellbags:

- Windows sizes and preferences
- Icon and folder view settings
- Metadata such as MAC timestamps
- Most recently used files and file type (zip, directory, installer)
- Files, folders, zip files, installers that existed at one point on the system (even if deleted).
- Network Shares and folders within the shares
- Metadata associated with any of the above types which may include timestamps and absolute paths
- True crypt volumes

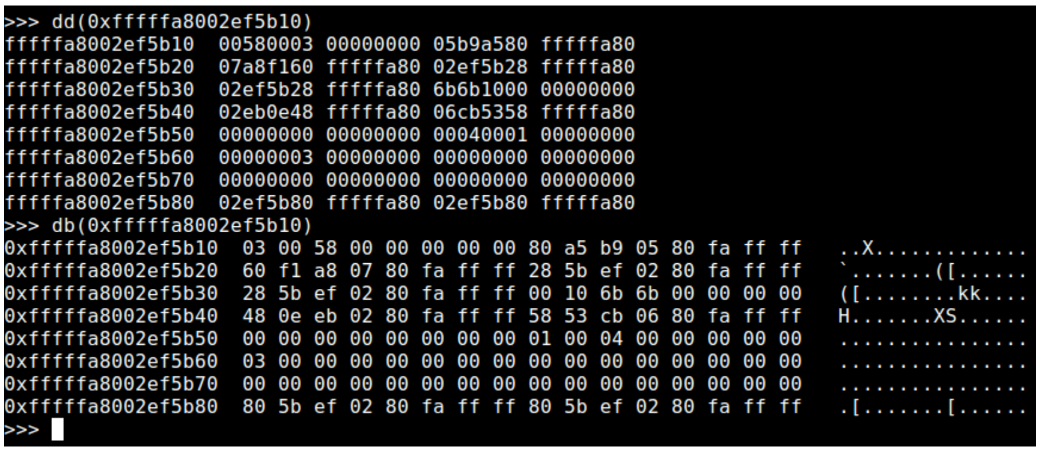


```
***************************************************************************
Registry: \??\C:\Users\Administrator.CPRO_CLONE\AppData\Local\Microsoft\Windows\UsrClass.dat
Key: Local Settings\Software\Microsoft\Windows\Shell\BagMRU\11
Last updated: 2018-09-27 02:42:03 UTC+0000
Value   Mru   File Name      Modified Date                  Create Date                    Access Date                    File Attr                 Path
------- ----- -------------- ------------------------------ ------------------------------ ------------------------------ ------------------------- ----
1       0     PROCES~1       2018-09-24 12:27:26 UTC+0000   2018-09-24 12:27:26 UTC+0000   2018-09-24 12:27:26 UTC+0000   NI, DIR                   New folder\Process Hacker 2
0       2     bin            2015-01-02 13:52:48 UTC+0000   2014-12-29 19:15:32 UTC+0000   2015-01-02 13:52:48 UTC+0000   DIR                       New folder\bin
2       1     PROCES~2       2018-09-23 07:20:30 UTC+0000   2018-09-23 07:19:38 UTC+0000   2018-09-23 07:20:30 UTC+0000   DIR                       New folder\processXXXXXX
***************************************************************************
```



```
19      6     frogo          2018-09-26 15:56:52 UTC+0000   2018-09-26 15:56:52 UTC+0000   2018-09-26 15:56:58 UTC+0000   DIR                       E:\frogo
```

The **volshell** plugin to disassemble 0xfffffa8002ef5b10:



```
>>> dd(0xfffffa8002ef5b10)
fffffa8002ef5b10  00580003 00000000 05b9a580 fffffa80
fffffa8002ef5b20  07a8f160 fffffa80 02ef5b28 fffffa80
fffffa8002ef5b30  02ef5b28 fffffa80 6b6b1000 00000000
fffffa8002ef5b40  02eb0e48 fffffa80 06cb5358 fffffa80
fffffa8002ef5b50  00000000 00000000 00040001 00000000
fffffa8002ef5b60  00000003 00000000 00000000 00000000
fffffa8002ef5b70  00000000 00000000 00000000 00000000
fffffa8002ef5b80  02ef5b80 fffffa80 02ef5b80 fffffa80
>>> db(0xfffffa8002ef5b10)
0xfffffa8002ef5b10  03 00 58 00 00 00 00 00 80 a5 b9 05 80 fa ff ff   ..X.............
0xfffffa8002ef5b20  60 f1 a8 07 80 fa ff ff 28 5b ef 02 80 fa ff ff   `.......([......
0xfffffa8002ef5b30  28 5b ef 02 80 fa ff ff 00 10 6b 6b 00 00 00 00   ([........kk....
0xfffffa8002ef5b40  48 0e eb 02 80 fa ff ff 58 53 cb 06 80 fa ff ff   H.......XS......
0xfffffa8002ef5b50  00 00 00 00 00 00 00 00 01 00 04 00 00 00 00 00   ................
0xfffffa8002ef5b60  03 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00   ................
0xfffffa8002ef5b70  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00   ................
0xfffffa8002ef5b80  80 5b ef 02 80 fa ff ff 80 5b ef 02 80 fa ff ff   .[.......[......
>>>
```

V. Attempt to extract evidence from an image

The traces found with the use of Volatility reveal what could be evidence of infection caused by malware, but no executable file was found, no file containing the public key. What you have is a generic notice with payment instructions. Faced with this situation, the research strategy we set up (which may not be the most profitable) was to analyze the image of the memory of the compromised machine. After the investigation step with **Volatility,** it was decided to use **bulk_extractor** and **Autopsy** (whose records will not be shown here).

As mentioned before, the cyber attack that affected the machine encrypted the files and prevented the company's business continuity. We did not have access to the malware executable, which usually deletes itself from the system after infection.

In that sense, the company had its files infected by malware of the type that encrypts the data, called ransomware. The ransomware is called FROGO. Immediately, efforts were begun to break the encryption imposed on SQLITE files, unsuccessfully so far. We were in person at the company on 10/10/2018 from 2 pm until around 7 pm. Immediately, we were informed that a type of attack was in progress trying to capture the password of the database system. Promptly, we began a job of replacing a cyber defense line for the company, finding out what was open on the company's external IP, because the database was exposed to the internet traffic and the server did not have a proper defense to the outgoing traffic and the database could be directly accessed from the Internet.

Once these loopholes had been discovered, we executed the closing of the open ports of the database and other services. Even after the establishment of a cyber defense line, the attacks continued. Two software programs installed in the company reported these attacks: Norton Security and Pfense. With this data, server protection, the main target of the attacks, was initiated with the above mentioned protective measures. It turned out that the company was exposed and we corrected that flaw. The attack denounced by these software clearly had the purpose of accessing the server where the database was.

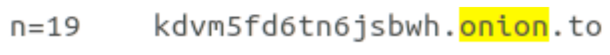
There are two clear possibilities for the attack:

- The attack was random, and the company was chosen at random, which usually occurs.
- The attack was deliberate because the encrypted files directly affected the business continuity of the company.

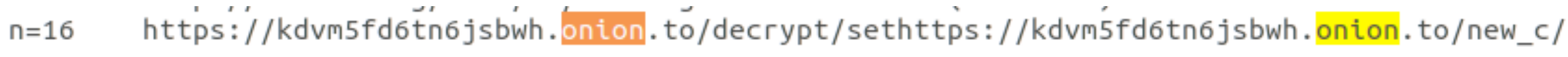
Our analysis only comes to this point, because to show if the attack is random or deliberate requires further investigation.

VI. Forensics analysis using bulk_extractor

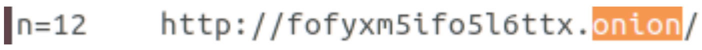
The forensic analysis carried out by me has been restricted to the moment, to find vestiges and evidences of the attack itself, based on a forensic technique called "RAM analysis". The procedure stems from the memory capture of the compromised machine, performed under my guidance, using **FTK IMAGER Lite** software, version 3.11. Below is what was discovered analyzing **bulk_extractor** URLs histogram for **onion domain**. In descending order (the number before the onion's domain name is the requisitions or visits/accesses). This is strong evidence that this access has been made from the compromised machine:



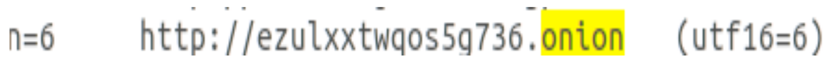
```
n=19     kdvm5fd6tn6jsbwh.onion.to
```



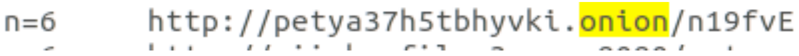
```
n=16     https://kdvm5fd6tn6jsbwh.onion.to/decrypt/sethttps://kdvm5fd6tn6jsbwh.onion.to/new_c/
```



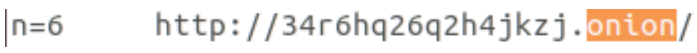
```
n=12     http://fofyxm5ifo5l6ttx.onion/
```



```
n=6     http://ezulxxtwqos5q736.onion    (utf16=6)
```

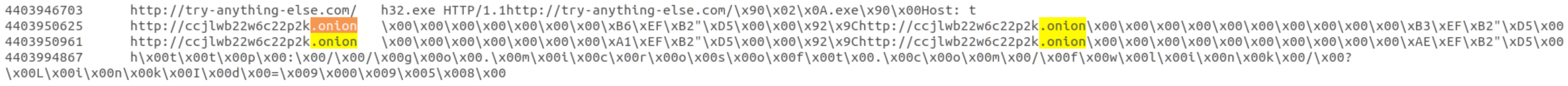


```
n=6     http://petya37h5tbhyvki.onion/n19fvE
```



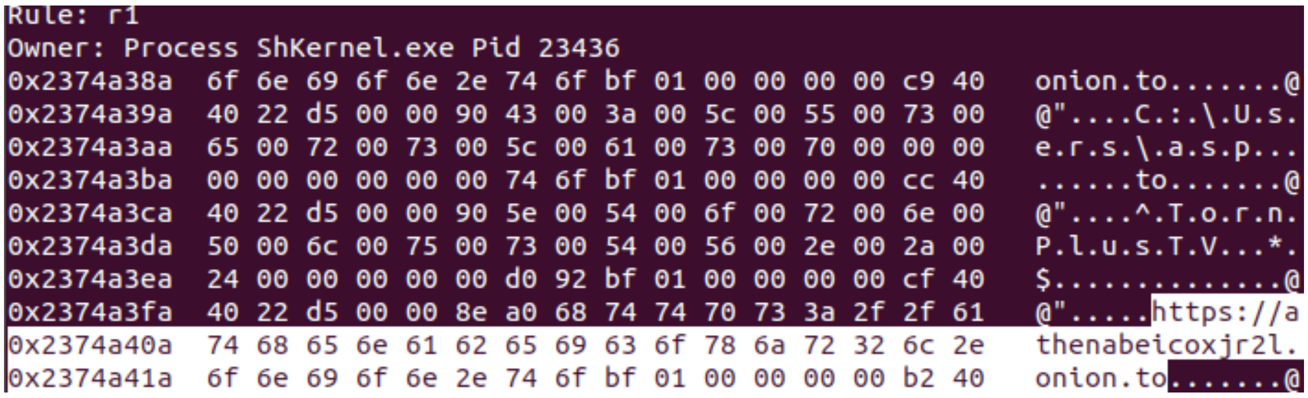
```
n=6     http://34r6hq26q2h4jkzj.onion/
```

More evidence found in the bulk_extractor URLs indexes are the requisitions:



```
4403946703      http://try-anything-else.com/   h32.exe HTTP/1.1http://try-anything-else.com/\x90\x02\x0A.exe\x90\x00Host: t
4403950625      http://ccjlwb22w6c22p2k.onion   \x00\x00\x00\x00\x00\x00\x00\xB6\xEF\xB2"\xD5\x00\x00\x92\x9Chttp://ccjlwb22w6c22p2k.onion\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\xB3\xEF\xB2"\xD5\x00
4403950961      http://ccjlwb22w6c22p2k.onion   \x00\x00\x00\x00\x00\x00\x00\xA1\xEF\xB2"\xD5\x00\x00\x92\x9Chttp://ccjlwb22w6c22p2k.onion\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\xAE\xEF\xB2"\xD5\x00
4403994867      h\x00t\x00t\x00p\x00:\x00/\x00/\x00g\x00o\x00.\x00m\x00i\x00c\x00r\x00o\x00s\x00o\x00f\x00t\x00.\x00c\x00o\x00m\x00/\x00f\x00w\x00l\x00i\x00n\x00k\x00/\x00?
\x00L\x00i\x00n\x00k\x00I\x00d\x00=\x009\x000\x009\x005\x008\x00
```

Volatility's **yarascan** plugin running against the "onion" rule reveals that the process ShKernel.exe has been a registry of onion domain, as we can see in figure below.



```
Rule: r1
Owner: Process ShKernel.exe Pid 23436
0x2374a38a  6f 6e 69 6f 6e 2e 74 6f bf 01 00 00 00 00 c9 40   onion.to.......@
0x2374a39a  40 22 d5 00 00 90 43 00 3a 00 5c 00 55 00 73 00   @"....C.:.\.U.s.
0x2374a3aa  65 00 72 00 73 00 5c 00 61 00 73 00 70 00 00 00   e.r.s.\.a.s.p...
0x2374a3ba  00 00 00 00 00 00 74 6f bf 01 00 00 00 00 cc 40   ......to.......@
0x2374a3ca  40 22 d5 00 00 90 5e 00 54 00 6f 00 72 00 6e 00   @"....^.T.o.r.n.
0x2374a3da  50 00 6c 00 75 00 73 00 54 00 56 00 2e 00 2a 00   P.l.u.s.T.V...*.
0x2374a3ea  24 00 00 00 00 00 d0 92 bf 01 00 00 00 00 cf 40   $..............@
0x2374a3fa  40 22 d5 00 00 8e a0 68 74 74 70 73 3a 2f 2f 61   @".....https://a
0x2374a40a  74 68 65 6e 61 62 65 69 63 6f 78 6a 72 32 6c 2e   thenabeicoxjr2l.
0x2374a41a  6f 6e 69 6f 6e 2e 74 6f bf 01 00 00 00 00 b2 40   onion.to.......@
```

VII. Conclusion

The article presented a stage of our work in the company. We're still working on breaking the file encryption key, but the customer was happy that we were able to recover the entire credit card share through Autopsy. But this step needs to be described separately because it is very extensive. However, with respect to the challenge presented by the type of ransomware, some lessons could be drawn by us, such as the need to develop plugins to analyze ransomware artifacts.

These were our efforts to find traces of the attack. There are other procedures being performed at this time, **like the packet analysis**.

VIII. Reference

https://volatility-labs.blogspot.com/2012/09/movp-32-shellbags-in-memory-setregtime.html

About the Author

I was born in São Paulo, the big, boring and bestial industrial city of my country, Brazil. I obtained my PhD at São Paulo University (USP) in analytical induction, a math, logic, statistical and philosophic area. I never work in this area…but, one day, walking on a Sunday morning, I discovered that I could use my statistical skills to analyze malware behavior, like a math model. So, I invested my time in this area since 1989. Nowadays, I teach forensics at the University Nove de Julho (UNINOVE) and I work with forensic analysis and malware analysis (reverse engineering of malware) as a free consultant. To escape from reality, in my spare time, I go to some place to practice fly fishing in the rivers that cut through the mountains and I keep going to programming in C and Python my own pieces of software.

A forensic analysis of the Electronic Point Record System

by André Ruschel

The electronic point registration system is a set of computerized equipment and programs with the purpose of electronically recording the entry and exit of employees in companies or institutions. Although the electronic record brings several benefits, some companies can use only one computer program to carry out the journey record and this work comes to show that this record may not always reflect the actual journey of the worker, which has been causing numerous technical and legal discussions.

ABSTRACT

The electronic point registration system is a set of computerized equipment and programs with the purpose of electronically recording the entry and exit of employees in companies or institutions. Although the electronic record brings several benefits, some companies can use only one computer program to carry out the journey record and this work comes to show that this record may not always reflect the actual journey of the worker, which has been causing numerous technical and legal discussions. With the use of computer forensics, this article details studies of cases in which this scenario has been identified that results in some learning and suggestions.

Keywords – Electronic point registration system, Employees.

I. INTRODUCTION

It is not uncommon for technology, in some cases, to replace human labor, and the current point clock was already a manual annotation work carried out by an official, until in the year 1888, the American jeweler named Willard Le Grand Bundy (New York, USA) invented the Clock [1] that allowed employees to drill a card to record their workday. As soon as Bundy created this watch, he already predicted that there could be manipulations and adopted individual keys for each employee, thus making it impossible for a person to "knock" multiple cards.

In Brazil, the point clock was adopted by the Brazilian business and working classes only in the government of the then president, Getúlio Vargas, in which a set of laws was created – the CLT (consolidation of labor laws) [2]. Thus, the first equipment emerged, precisely, to meet the needs arising from the control of the working day in the country.

In these laws, three forms of journey control were established: by manual point (manuscript), mechanical control or even electronic point. Initially, there were only the first two forms, performed by means of annotations, in the book point, or by marking in mechanical clock.

However, it was in Law n°. 7.855, of October 24, 1989 [3], which the "Electronic System" appeared. Only after the Ministry of Labor and Employment (MTE) created and approved Ordinance 1.510 [4] were guidelines established for the use of the electronic system in the control of the journey and the technical requirements necessary for the equipment of the point that was the Electronic Point Recorder – REP.

The electronic point recorder-REP is a piece of automation equipment used exclusively for the registration of work hours and with the ability to issue fiscal documents and carry out fiscal controls, referring to the entry and exit of employees at work sites. Although this electronic peer recorder understands equipment (hardware), he also needs a program (software) with the function of performing the treatment of the markings, that is, it allows to connect (seek/import) the markings of the electronic register and to treat this information, generating reports of frequency and mirror of point, being still possible to administer overtime, clearances, days paid and bank of hours.

In addition, many Brazilian companies are based on the ordinance of the Ministry of Labor and Employment (MTE) n° 373, of February 25, 2011 [5], in which they are authorized by convention or collective labor agreement, use only the program (software) to perform your employees' journey records. However, it occurs that not all companies have the electronic point recorder (hardware) and program (software) set. However, they are in agreement with the Ordinance of the Ministry of Labor and Employment (MTE) n° 1.510, of August 21, 2009.

With this work, it seeks to cover scenarios of companies that follow the ordinance of the Ministry of Labor and Employment (MTE) n° 1.510, but it deepens exactly in cases of companies that are only based on the ordinance of the Ministry of Labor and Employment (MTE) n° 373, those that use only the program (software) to perform the registration, in which this point control system has several pre-established operating policies (rules). In some situations, the program (software) was developed (created/elaborated) by the company itself, which holds the source code and which, in the vast majority of cases, works only within the intranet (similar to the Internet, however, of exclusive use of an organization), in which access to the system is performed through login.

Based on this, the present work aims to present not only legal aspects involved, but especially the technical aspects, considering several situations found in loco about the electronic point registration system.

This work is organized as follows: in Section II, the legal aspects and technical standards are presented; Section III will discuss alerts and situations in the work environment; Section IV presents a case study; Section V shows the proof of concept and finally, in section VI, the conclusion and future works.

II. LEGAL ASPECTS AND TECHNICAL STANDARDS

Let us now take an approach on some legal aspects and technical norms more relevant to this work.

The annotation of timesheet appeared in the Consolidation of Labor Laws n°. 5.452 of May 1, 1943, we cite Art. 74 - § 1 - The working schedule will be recorded in the register of employees with the indication of collective agreements or contracts concluded.

Paragraph 2 - For establishments with more than ten employees, a record of the time of entry and exit must be recorded in a manual, mechanical or electronic register, according to instructions to be issued by the Ministry of Labor, and there must be pre-signaling of the rest period.

The so-called "Electronic Point Law" is represented by Ministry of Labor and Employment (MTE) n°. 1.510, dated August 21, 2009, whose purpose is to faithfully record the markings made by the employee, in which no action is allowed as time restrictions on point marking, automatic point marking, predetermined times or contractual hours, requirement by the prior authorization system for overtime marking, or the existence of any device that allows the change of recorded data.

In addition to the Ordinance of the Ministry of Labor and Employment (MTE) n° 1.510, there is the Ministerial Order of the Ministry of Labor and Employment (MTE) n°. 373, dated February 25, 2011, which came to update some information of Ordinance 1.510, the use of methods and provides for the possibility of adoption by employers of alternative systems of working day control since authorized by Convention or Collective Bargaining Agreement.

In the technical aspect, there is Ordinance N° 480, dated December 15, 2011 - National Institute of Metrology, quality and technology - INMETRO [6] and later the Ministry of Labor and Employment (MTE) n° 101, of January 13, 2012 [7] in which there was a Cooperation Agreement signed between INMETRO and the Ministry of Labor and Employment (MTE), as well as the Ministry's initiative to formally delegate to INMETRO the activities of planning, developing and implement the Compliance Assessment Program of the Electronic Point Recorders -REP, through the advice of the MTE.

Subsequent to the aforementioned Ordinances, others have appeared that have brought complements, although all of them are fundamental for the elaboration of Electronic Point Recorder (hardware), as well as standardization standards for firmware development.

III. ALERTS AND SITUATIONS IN THE WORK ENVIRONMENT

In business, many computers contain evidence that is useful in many human resource circumstances. Thus, allegations of discrimination, sexual harassment and unfair offenses are serious threats that are best understood when one knows what an employee did.

Because computers and programs are such a pervasive part of the professional life of most employees, analyzing the data stored on those computers and databases helps solve numerous problems. In investigations involving human resources, we analyze some technical procedures that are classified as "non-compliance", when faced with norms, ordinances and labor laws.

With regard to data collection and analysis found in forensic computer skills, there was a wealth of information on the activities of human resources and information technology workers who are useful, if not determinant, in investigations of the system (software) of the Electronic Point Record.

In addition, it is in this system (software) that information relevant to workdays is recorded and, using forensic computation, one can trace the steps used by a dishonest employee or employer to provide the evidence necessary for fair and resolute decisions.

Usually, problems always involve one or more employees who have privileged access to the system. However, there are different types of fraud or theft by employees ranging from misappropriation of assets and supplier fraud, payroll fraud, accounting fraud, data theft, and also bribery and corruption. Here are a few that they involve:

- *Data theft*

Data theft is greater, thanks to the way you currently work using computerized systems instead of records and paper files. There are thefts of trade secrets, customer data and contact lists, and the theft of "personally identifiable information," such as people's credit card numbers and bank details.

- *Misappropriation of assets*

Misappropriation of assets has a variety of faces, all of which are quite common in business. There is theft of physical inventory, as well as theft of money and services. This way, employees can defraud your company by lying about their expenses and falsifying values from those accounts.

Payment fraud includes items like vendor fraud schemes and fake customer accounts created specifically to make fraudulent payments. This can mean making self-authorized payments as well as working with others to claim money under false pretenses. In that sense, an employer may fraudulently claim an employee compensation policy, conduct a health insurance fraud, or falsify sales data to obtain higher commission payments.

- *Account Fraud*

When unscrupulous people have access to their accounting systems, it can cause havoc, embezzlement and fraud of accounts payable, fake vendors, unauthorized personal purchases, and accounts receivable fraud. Therefore, accounts receivable fraud covers things like deviance, where an employee gets the money that his or her company effectively voided, which is rarely tracked. Some people even create fake accounts and sales in an effort to make the company more successfully achieve its goals.

- *Bribery and corruption*

Bribes, kickbacks, ghost-company schemes and surrogate products are examples of bribery and corruption. And like all other types of fraud being covered, the systems that companies use daily are exploited by fraudsters for their own purposes.

Also, people usually end up being greedy and are discovered, but since it's too late to take preventive measures, it makes sense to conduct sensible background checks on any employee who has access to their IT systems, even more, for people who have total freedom or really control those systems.

Many of these evidences are difficult to eliminate, even a computer user who wants to cover their tracks can wipe some of this information, but usually some data ends up being discovered in an investigation. However, even savvy and sophisticated people will have trouble eliminating everything, as even deleting information will leave traces.

- *Payroll Fraud*

So-called ghost employee schemes involve wages paid to a fake employee or a former employee who is still on the payroll. And time-table fraud is when someone falsifies their time sheets, adds overtime, or causes another person to log in and out of the company, or even modifies timekeeping information in the company's system.

In addition to the mentioned situations, there is still a new situation, it is understood that the Electronic Point Record system is reliable, especially when using the Electronic Recorder Point (hardware) and program (software). It turns out that when only the software program is used, the scenario already suffers from some "nonconformities":

A. When using the Electronic Point Recorder (hardware) and the software program (Software)

Following the Ordinance of the Ministry of Labor and Employment (MTE) n° 1.510, of August 21, 2009, in which adopting Electronic Point Recorder (hardware) and software (software) for registration, data reliability is the employee's record is made directly at the recorder.

Although not mentioned in the above Ordinance, the use of biometric features, in this case, fingerprint with an optical biometric reader, which compares to an image database with the recorded digital, would bring numerous safety benefits.

Some Electronic Point Record systems offer other forms of access control: Use of biometrics, magnetic card, typing of a password, using the Electronic Point Recorder's keyboard directly or an approach card.

According to INMETRO Ordinance n° 595, of December 5, 2013 [8], all records and occurrences are stored in the Point Register Memory (MRP) and can also generate the Data Source File (AFD) that is the file where all the data is stored in the MRP.

In this case, MRP is constituted as a means of data storage, with the capacity to retain recorded data for at least 10 (ten) years, which can not be erased, overwritten or altered, directly or indirectly.

Even though it is very efficient to use the electronic point recorder, there are some warning situations when using the "Point Record Processing Program":

✦Due to lack of paper in the watch's print coil, employees will not record their work in the Electronic Point Recorder;

✦Although INMETRO Ordinance n° 595, of December 5, 2013, cite in item 5.2.4.2 "In the case of paper jammed, of paper lacking that does not allow to complete the printing or of other usual events of inhibition of the impression of the Voucher, the REP cannot allow the next point marking."

It occurs that, often, the paper is not replaced on the same day, preceding precedent to manual releases.

✦Although mentioned in the Ministry of Labor and Employment (MTE) Ordinance No. 1,510 - Art. 12 Sole Paragraph. "The data processing function shall be limited to adding information to complement any omissions in the point record or to indicate undue markings", an aggravating circumstance is visualized in cases in which the employee stops recording directly in the Electronic Point Recorder (be it the day of entry, interval or exit), because, in this way, the registration is done later manually by your hierarchical superior or responsible for the specific sector, which may not reflect with your actual working day, as in the same way , being informed of contradictory times.

B. When using only software for registration

The use of technology usually solves numerous problems, but on the other hand, it also brings others, since, when the Electronic Point Record System is implemented, especially when only the program for registration is used, some situations occur; the main ones are:

- It is not verified, in the Ordinances, what would be the technical standards for the development of the point registration program;
- Although the Ministerial Order of Labor and Employment (MTE) n° 1.510 cites in Art. 7 the receipt of the employee's proof, this resource is not investigated when only the program is used and not every employee has a printer of his own disposition;

The INMETRO Ordinance n° 595, dated December 5, 2013, cites in item 5.1.14 "The REP must have a printer mechanism in paper reel, integrated and exclusively for the equipment, that allows contrasting color printing with paper, in legible characters with the following characteristics:

a) Maximum horizontal density of 8 (eight) characters per centimeter;

b) The character cannot have height less than 3 (three) millimeters;

c) The durability of the printing cannot be less than five years, using the type of printing paper indicated by the supplier in the Operational Manual."

- At the time the program allows printing, the common type of paper is usually used, which has no durability over the years;
- Time zone issues when the company is based in several regions;
- Inexistence of the adoption of Electronic Point Recorder (hardware) equipment;
- In every computer system, more specifically, a program has the "root" or "administrator" super-user who has superior privileges, which can or could make undue access to the point registration system;
- Fragility of security by using only the username and personal password, not using biometrics feature or double authentication factor;
- Implementations in the source code of routines containing predefined rules or automated routines;
- Locks in the point registration system (be it for start, interval or end of journey);
- A system that logs off and "knocks" the user off, so that the exact working time is not recorded correctly;
- It does not offer the function of generating the Data Source File - AFD, from the data stored in the MRP, much less the option of using an external memory device through the Fiscal Port;
- When tax files are requested, only one .txt file (text file extension) is exported from the system without at least bringing the hash of the file;
- It does not offer the possibility of issuing the RIM as a quotation INMETRO Ordinance n° 595, of December 5, 2013, in item 5.1.2 "REP must have a unique button with" RIM ", in the red color, for the emission of the Instantaneous Link of Markings; and another unique button, "i" ID, in italic text, in blue color, for the printing of your public key and the software identifiers."

IV. REAL CASE

This paper seeks to show a real case about the situation where the employee registration - Point Electronic Registration System uses only the program developed by the company's own IT team.

The scenario brings some relevant points:

A. The system was hosted on company servers, in the case of private cloud;

B. Using the company's computers, employees made their work records;

C. They logged in on computers that had Windows 7 and Windows 10 installed;

D. Only after logging on to Windows, they accessed a web system, re-entered their credentials (user and password), and finally performed the registration;

E. It is assumed that in this process, depending on the configuration of the workstation, such as the intranet speed, you will lose a good amount of time to register the journey.

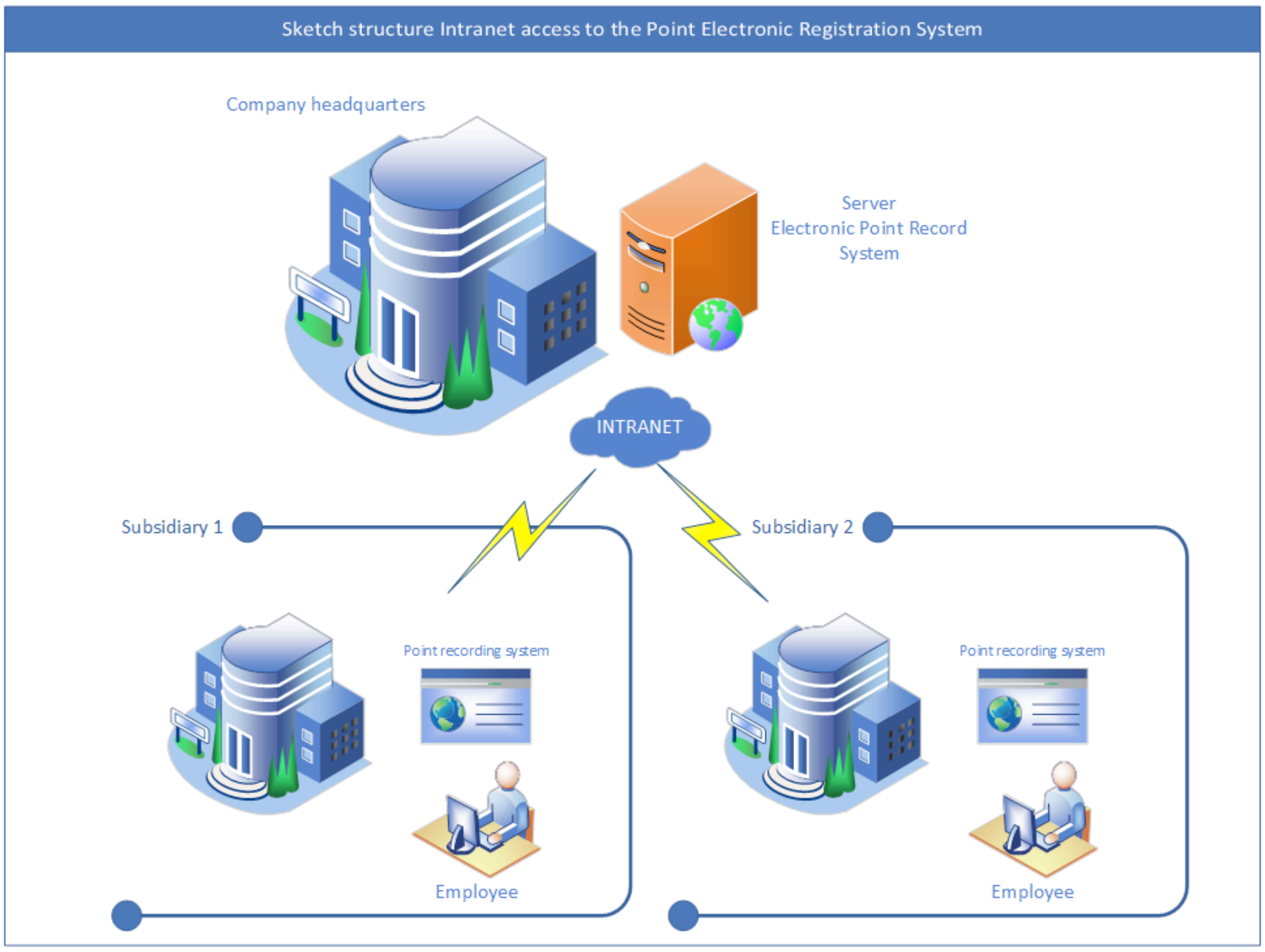


Figure 1. Client / server infrastructure for access to electronic registration system.

Software:

The Electronic Point Record program is a web system, developed in Java [9] and that uses the PostgreSQL database [10].

When developing an Electronic Point Record system, it is necessary to have a clear definition of rules by means of computational functions and algorithms, to define, therefore, the access to this system of point registers. Also, it is essential to have an authentication, which is the act of providing identity to an application or network resource.

Typically, the identity is approved by a cryptographic operation that uses a key known only to the user, such as public key cryptography, or a shared key. Thus, the server side of the authentication exchange compares the signed data with a known encryption key to validate the authentication attempt [11].

In addition, in order for the employee to log his/her work day, it is advisable to first access the computer with a username and password (authenticating in the Active Directory - Microsoft Windows Server Directory Service) [12] and then perform a new authentication in the Electronic Point Record System using a new user and password again. However, this time, authenticating in the Electronic Point Record system itself.

In authentication, the user must present something that only he knows or possesses, and may even involve verifying personal physical characteristics. Most current systems require a password (something that only the user knows), however, there are more modern systems using smart cards (something that the user has) or physical characteristics (something intrinsic to the user), such as hand, retina or face format, fingerprint and speech recognition.

Biometric systems are automatic identity analysis systems based on physical characteristics of the user, which are intended to address security deficiencies of passwords that may be revealed or discovered.

In this case, there are some vulnerabilities because of the fragility of the password and also the significance of these, as well as a possibility of fraud in the employee's work records.

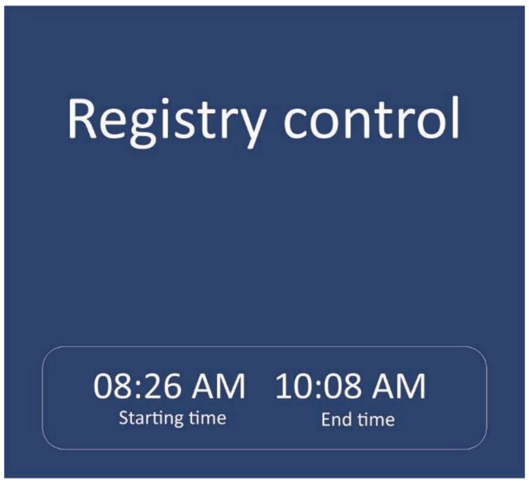
Thus, with no biometric authentication, the expertise pointed to evidence that a user (employee) who worked in branch A and who had never worked in branch B, performed registrations through the Electronic Point Registration System, using a branch office workstation B.

It was also verified that the Electronic Point Record based on a web system allowed the use of a bot, that is, allows autonomous applications running through the Internet to perform the predetermined task of recording the day.

We observed that there was not even the impression of proof of daily work, which would be the principal mechanism of proof.

It is analyzed that when the employee registers his/her morning entry day before or after 8:30 am, the system itself performs the standard day appointment. If his schedule is to start at 8:30 am and finish at 5:30 pm, it is at those exact times that the system made the note, without making any previous notice to him.

Example of an original markup:



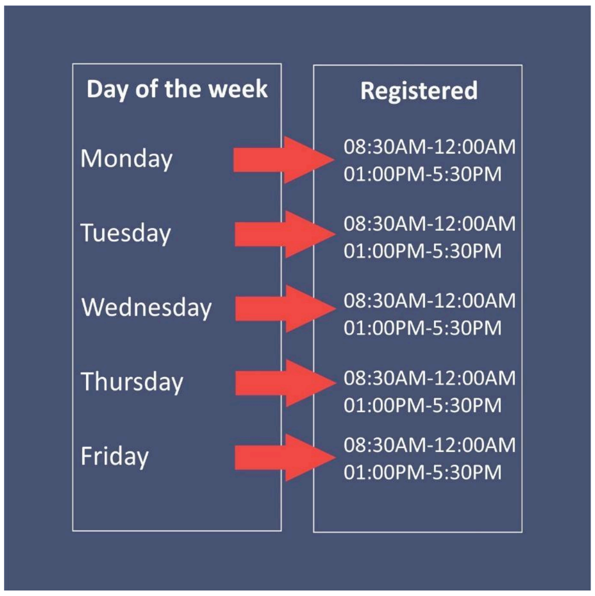
Registry control

08:26 AM
Starting time

10:08 AM
End time

Figure 2. Timetable made by the employee.

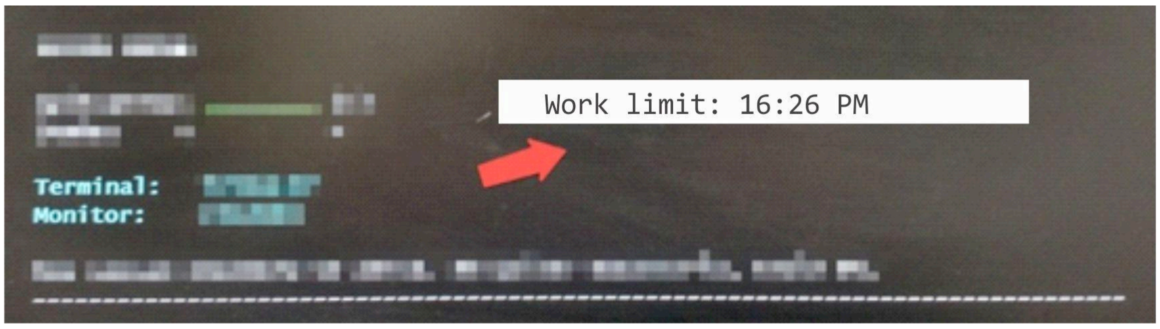
Example of times that have been logged by the system:



Day of the week	Registered
Monday	08:30AM-12:00AM 01:00PM-5:30PM
Tuesday	08:30AM-12:00AM 01:00PM-5:30PM
Wednesday	08:30AM-12:00AM 01:00PM-5:30PM
Thursday	08:30AM-12:00AM 01:00PM-5:30PM
Friday	08:30AM-12:00AM 01:00PM-5:30PM

Figure 3. Standard marking performed by the system.

There are locks on the system that logs the user down with predefined time in the Electronic Point Record system.

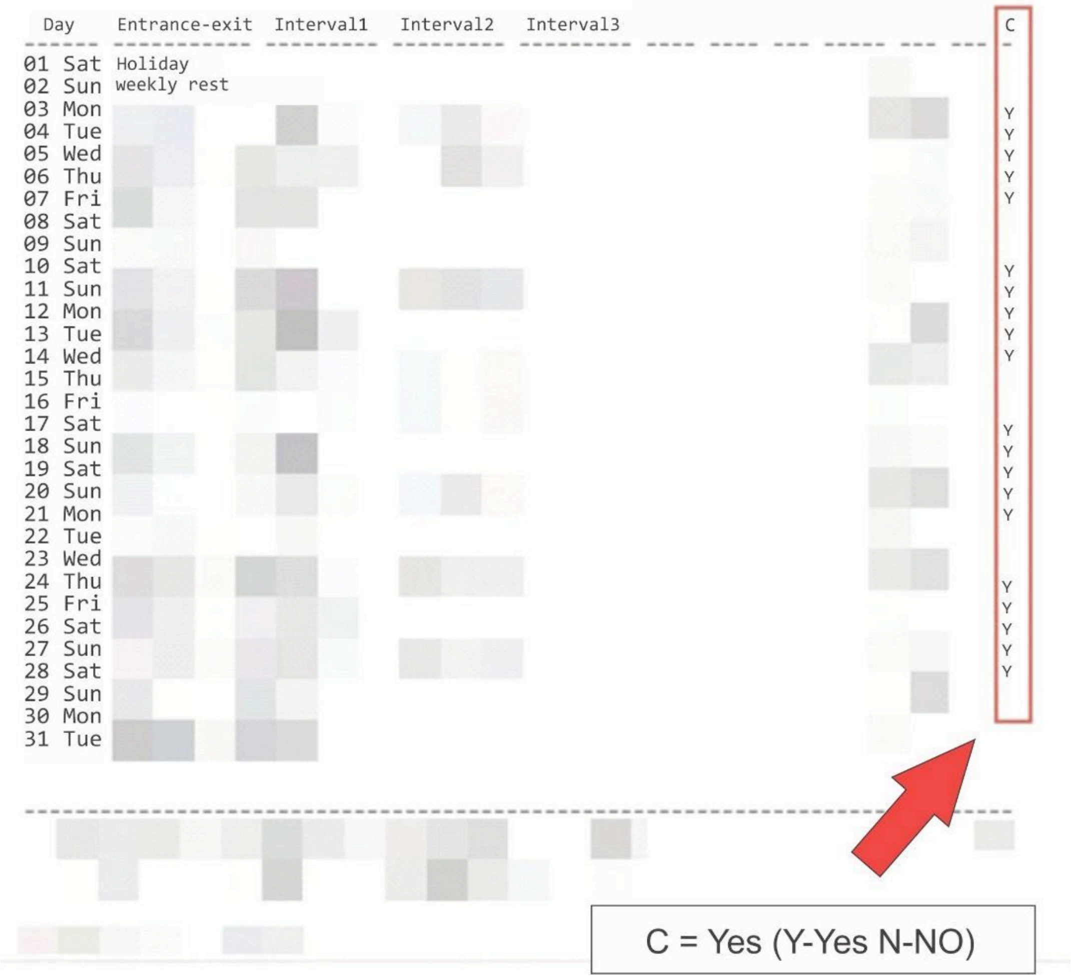


```
[redacted]

[redacted]                              Work limit: 16:26 PM

Terminal: [redacted]
Monitor:  [redacted]

[redacted]
--------------------------------------------------------------------------
```

Figure 4. Information of the time that the system will log off (the electronic point record fails).

It was observed that at the moment in which there are automated registrations by the system (logoff), the employee has the option to agree or disagree. It is only registered when the SIM option is selected.

Only record of concordances not reflecting reality.

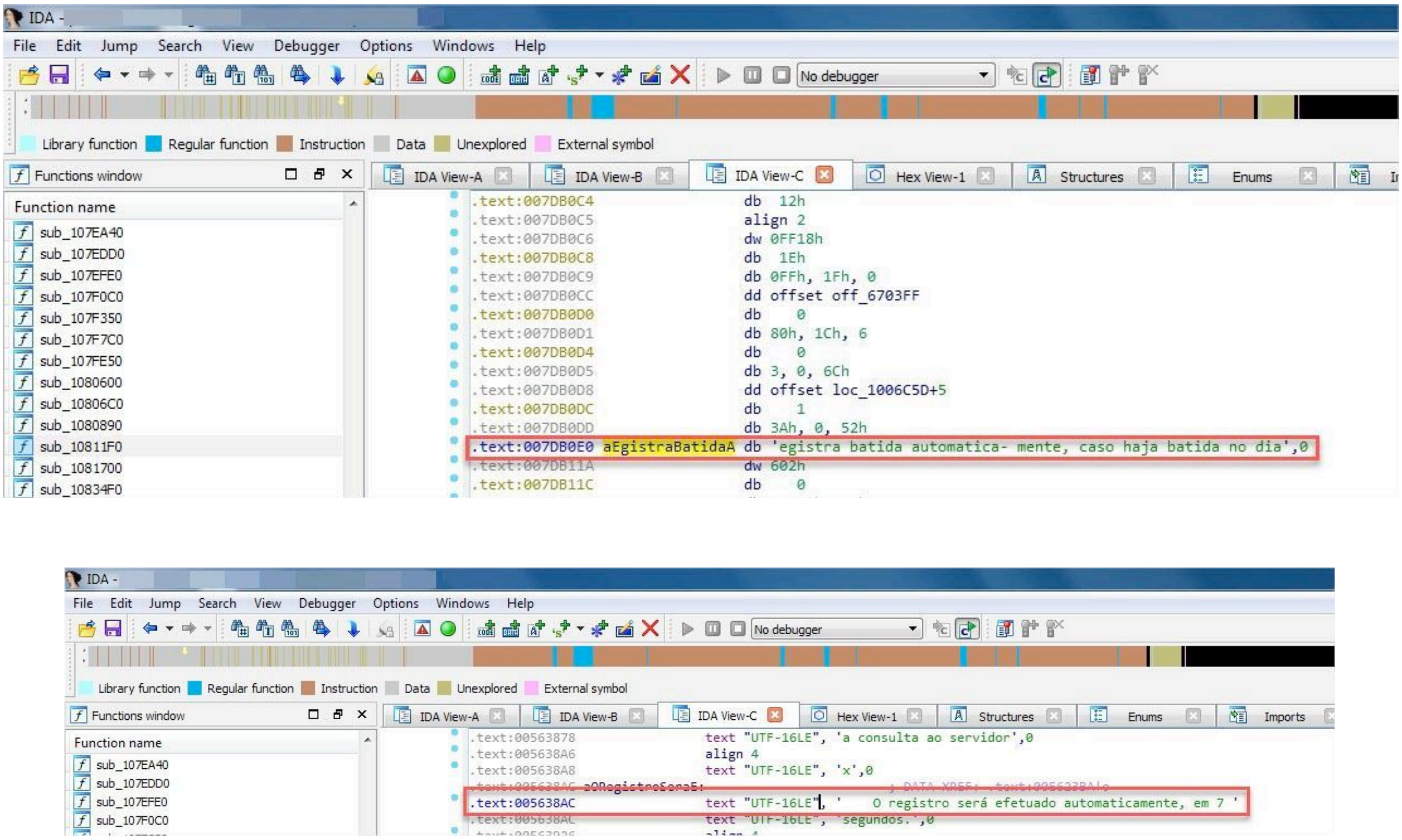


```
 Day     Entrance-exit  Interval1  Interval2  Interval3                          C
------   -------------  ---------  ---------  ---------  ---- ---- --- ----- --- --- -
01 Sat  Holiday
02 Sun  weekly rest
03 Mon                                                                           Y
04 Tue                                                                           Y
05 Wed                                                                           Y
06 Thu                                                                           Y
07 Fri                                                                           Y
08 Sat
09 Sun
10 Sat                                                                           Y
11 Sun                                                                           Y
12 Mon                                                                           Y
13 Tue                                                                           Y
14 Wed                                                                           Y
15 Thu
16 Fri
17 Sat                                                                           Y
18 Sun                                                                           Y
19 Sat                                                                           Y
20 Sun                                                                           Y
21 Mon                                                                           Y
22 Tue
23 Wed
24 Thu                                                                           Y
25 Fri                                                                           Y
26 Sat                                                                           Y
27 Sun                                                                           Y
28 Sat                                                                           Y
29 Sun
30 Mon
31 Tue
-------------------------------------------------------------------------------------
[redacted]
```

C = Yes (Y-Yes N-NO)

Figure 5. Marking of concordances.

IDA Pro [13] was used, which is a multiplatform and multiprocessor debugger that converts machine executable code into assembly language source code for debugging and reverse engineering purposes. After debugging the software (program) used for Electronic Point Record, there are some non-conformities about automation of records such as "log automatically beat ...." and "the registration will be done automatically .."



```
.text:007DB0C4                  db  12h
.text:007DB0C5                  align 2
.text:007DB0C6                  dw 0FF18h
.text:007DB0C8                  db  1Eh
.text:007DB0C9                  db 0FFh, 1Fh, 0
.text:007DB0CC                  dd offset off_6703FF
.text:007DB0D0                  db    0
.text:007DB0D1                  db 80h, 1Ch, 6
.text:007DB0D4                  db    0
.text:007DB0D5                  db 3, 0, 6Ch
.text:007DB0D8                  dd offset loc_1006C5D+5
.text:007DB0DC                  db    1
.text:007DB0DD                  db 3Ah, 0, 52h
.text:007DB0E0 aEgistraBatidaA db 'egistra batida automatica- mente, caso haja batida no dia',0
.text:007DB11A                  dw 602h
.text:007DB11C                  db    0
```

```
.text:00563878                  text "UTF-16LE", 'a consulta ao servidor',0
.text:005638A6                  align 4
.text:005638A8                  text "UTF-16LE", 'x',0
.text:005638AC aORegistroSeraE:
.text:005638AC                  text "UTF-16LE", '    O registro será efetuado automaticamente, em 7 '
.text:005638AC                  text "UTF-16LE", 'segundos.',0
```

Figure 6. Evidence of automation in the Electronic Point Record system.

V. PROOF OF CONCEPT

To exemplify how some Electronic Point Record systems have automated procedures, which are largely obscure to the user, an example that must be considered a proof of concept will be presented in this subsection. A very small fragment was created to show how Electronic Point Record systems record the scenario in which:

1. Only one program (software) is used for Electronic Point Record of employees;

2. The system contains routines in the code that restrict the registration at the point and now automates "beats" without the employee's consent;

3. No proof of receipt (receipts);

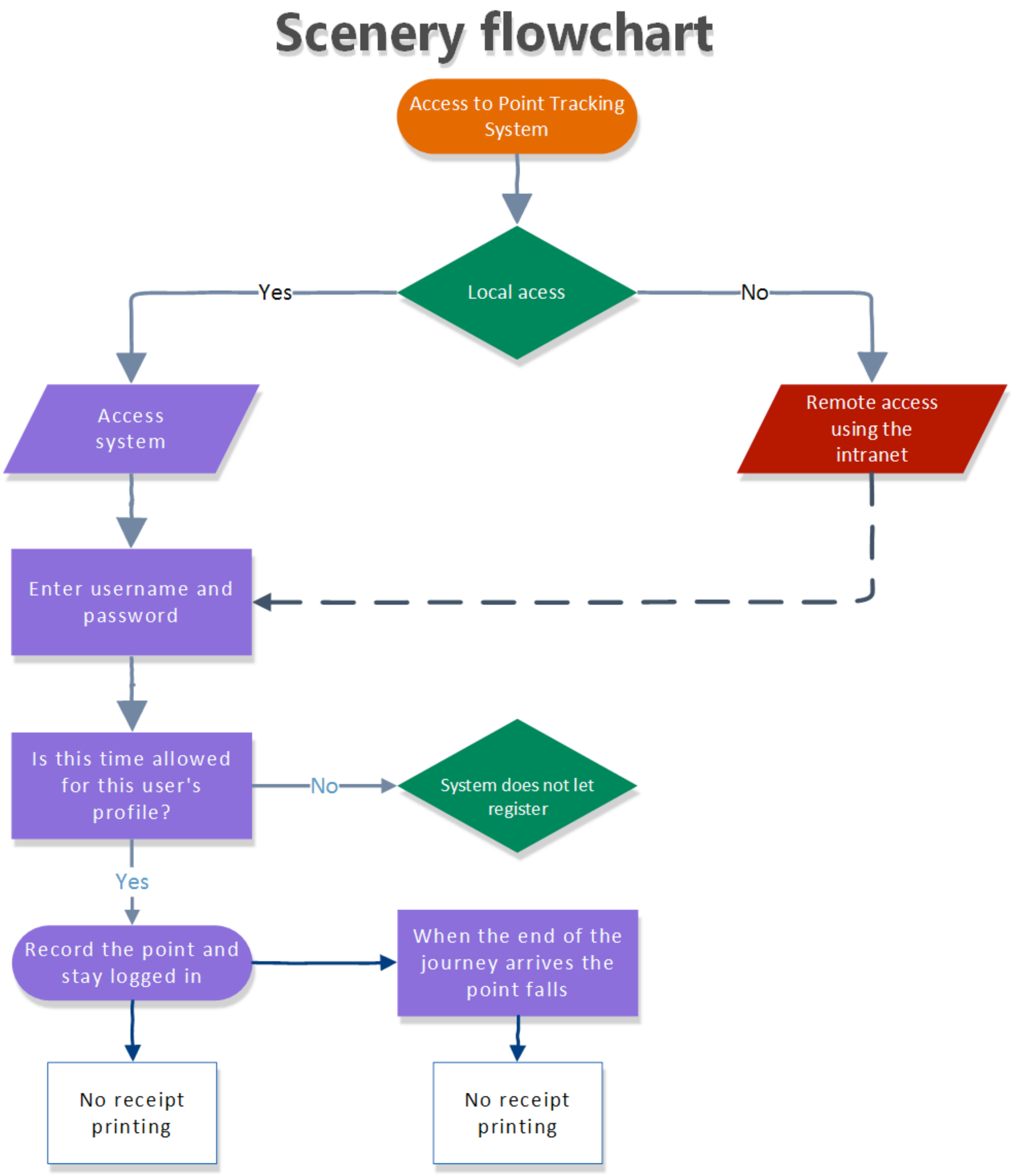

Scenery flowchart
Access to Point Tracking System
Local acess
Yes
No
Access system
Remote access using the intranet
Enter username and password
Is this time allowed for this user's profile?
No
System does not let register
Yes
Record the point and stay logged in
When the end of the journey arrives the point falls
No receipt printing
No receipt printing

Figure 7. Flowchart of the Electronic Point Record system.

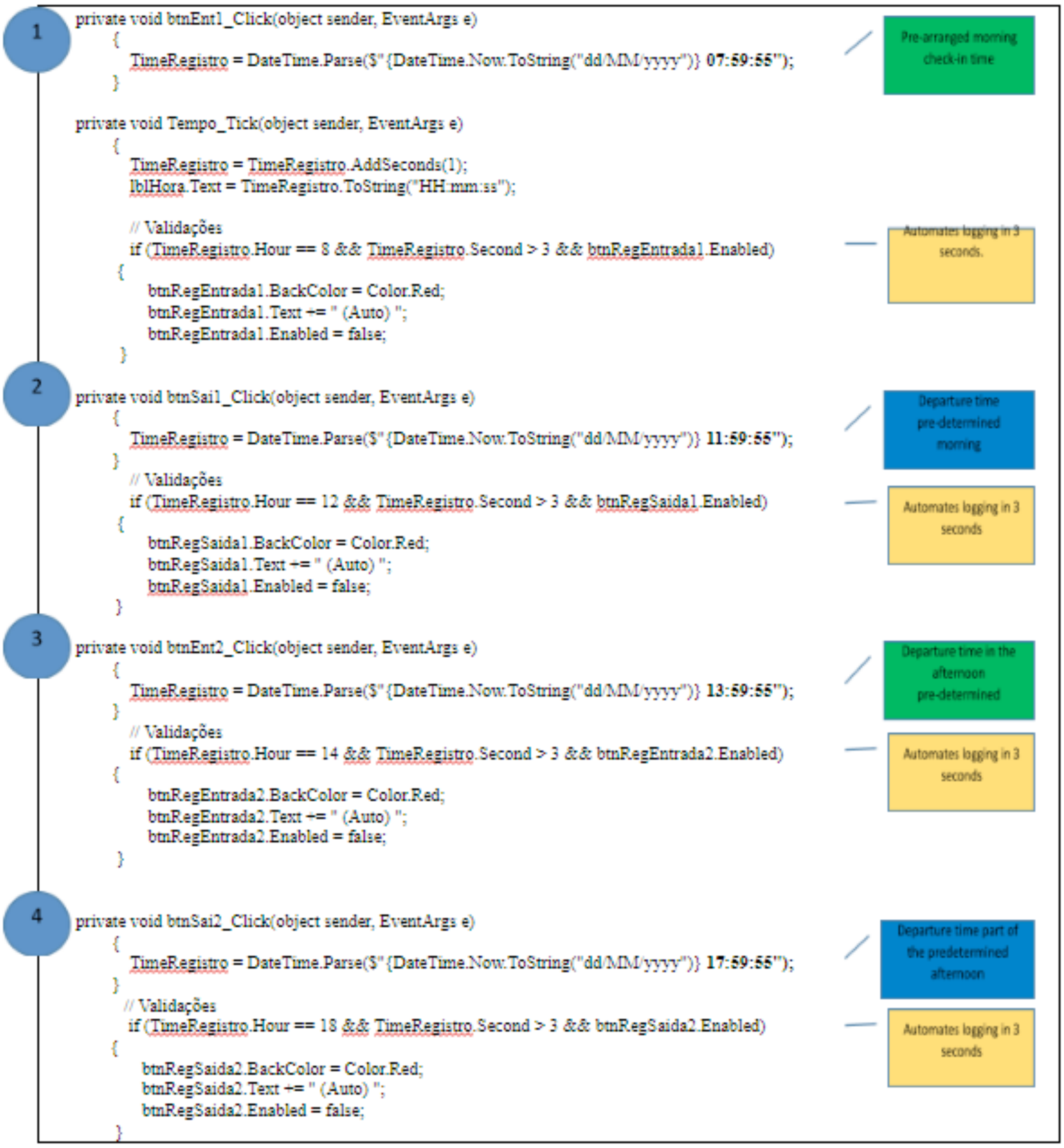
For example, some lines of code were formed:

1- Registers entry time1: Registers appointment automatically after 3 seconds - 08h00min03s;

2- Registers exit time1: Automatically registers appointment after 3 seconds - 12h00min03s;

3- Registers entry time2: Automatically registers entry after 3 seconds - 14h00min03s;

4 - Records exit time2: Automatically registers after 3 seconds - 18h00min03s.



```
private void btnEnt1_Click(object sender, EventArgs e)
    {
      TimeRegistro = DateTime.Parse($"{DateTime.Now.ToString("dd/MM/yyyy")} 07:59:55");
    }

private void Tempo_Tick(object sender, EventArgs e)
    {
      TimeRegistro = TimeRegistro.AddSeconds(1);
      lblHora.Text = TimeRegistro.ToString("HH:mm:ss");

      // Validações
      if (TimeRegistro.Hour == 8 && TimeRegistro.Second > 3 && btnRegEntrada1.Enabled)
     {
        btnRegEntrada1.BackColor = Color.Red;
        btnRegEntrada1.Text += " (Auto) ";
        btnRegEntrada1.Enabled = false;
     }
private void btnSai1_Click(object sender, EventArgs e)
    {
      TimeRegistro = DateTime.Parse($"{DateTime.Now.ToString("dd/MM/yyyy")} 11:59:55");
    }
      // Validações
      if (TimeRegistro.Hour == 12 && TimeRegistro.Second > 3 && btnRegSaida1.Enabled)
     {
        btnRegSaida1.BackColor = Color.Red;
        btnRegSaida1.Text += " (Auto) ";
        btnRegSaida1.Enabled = false;
    }
private void btnEnt2_Click(object sender, EventArgs e)
    {
      TimeRegistro = DateTime.Parse($"{DateTime.Now.ToString("dd/MM/yyyy")} 13:59:55");
    }
      // Validações
      if (TimeRegistro.Hour == 14 && TimeRegistro.Second > 3 && btnRegEntrada2.Enabled)
    {
        btnRegEntrada2.BackColor = Color.Red;
        btnRegEntrada2.Text += " (Auto) ";
        btnRegEntrada2.Enabled = false;
     }

private void btnSai2_Click(object sender, EventArgs e)
    {
      TimeRegistro = DateTime.Parse($"{DateTime.Now.ToString("dd/MM/yyyy")} 17:59:55");
    }
     // Validações
     if (TimeRegistro.Hour == 18 && TimeRegistro.Second > 3 && btnRegSaida2.Enabled)
    {
       btnRegSaida2.BackColor = Color.Red;
       btnRegSaida2.Text += " (Auto) ";
       btnRegSaida2.Enabled = false;
    }
```

1
Pre-arranged morning check-in time
Automates logging in 3 seconds.
2
Departure time pre-determined morning
Automates logging in 3 seconds
3
Departure time in the afternoon pre-determined
Automates logging in 3 seconds
4
Departure time part of the predetermined afternoon
Automates logging in 3 seconds

Figure 8. Code lines with automations (code listing for copying at the end of the article).

VI. CONCLUSION

At a time when the future of work is being discussed, where the personification of robots is already discussed, with the use of artificial intelligence and machine learning, one encounters existing Ordinances and norms that are or were essential and of great value, but as in the cases presented in this paper, they do not contemplate globally and do not go deeper when one chooses to use only a computer program to perform the registration related to the entry and exit of employees in the workplace, where even with the whole the advent of technology, allows for possible manipulation.

Although the expertise in Electronic Point Record system is complex and scarce, involving many factors, it has brought here a new perspective on the "modus operandi" whether worker or employer.

Based on the traces left in an electronic point record system (program), the information found can help in the correct interpretation of the data obtained by the expertise, which are of fundamental importance for the determination of artifacts of non-compliance with the Law.

However, in the near future, given the current reality of the increasing use of cloud computing, the increase of SaaS-based electronic record control systems (Software as a Service) and, consequently, will be widely used to provide employees with the means to carry out their workday records.

This work also brings a new and current challenge for the expert to acquire, analyze and examine evidence focused on forensic cloud computing, which now involves, in addition to IaaS (Infrastructure as a Service), SaaS, more specifically in this work, the programs that will be in the cloud performing electronic point record control.

The results presented in this paper can be used to create or improve new tools, adaptations and even new norms or guidelines aimed at the development and validation of programs (software) for companies that only use the program for electronic point record of their employees.

REFERENCES

[1] Peretti, Jacques, “The Deals that Made the World.” London: Hodder & Stoughton, 2017. ISBN-13: 978-1473646421

[2] Brazil. Decree-Law No. 5,452, of May 1, 1943.

[3] Brazil. Law No. 7,855, of October 24, 1989.

[4] Brazil. Ordinance of the Ministry of Labor and Employment - MTE nº 1,510, of August 21, 2009.

[5] Brazil. Ordinance of the Ministry of Labor and Employment - MTE nº 373, of February 25, 2011.

[6] Brazil. Ordinance of the National Institute of Metrology, quality and technology - INMETRO nº 480, of December 15, 2011.

[7] Brazil. Ordinance of the Ministry of Labor and Employment - MTE nº 101, of January 13, 2012.

[8] Brazil. Ordinance of the National Institute of Metrology, quality and technology - INMETRO no. 595, of December 5, 2013.

[9] JAVA - Official site - Available at https://java.com/en/ Accessed on July 8, 2018.

[10] PostgreSQL - Official website - Available at https: //www.pos¬tgresql.org/ Access on July 4, 2018.

[11] Computer and Information Security: User Authentication Methods. Available at https: // www.revis¬tabw.com.br/revistabw/seguranca-autenticaca-de-usuario. Accessed May 16, 2018.

[12] Overview of Windows Authentication. Available at https://msdn.microsoft.com/en-us/library/hh831472(v=ws.11).aspx. Accessed May 17, 2018.

[13] IDA - Official website - Available at: https://www.hex-rays.com/pro¬ducts/ida/ Accessed on March 25, 2018.

Code listing:

```
private void btnEnt1_Click(object sender, EventArgs e)
        {
                    TimeRegistro = DateTime.Parse($"{DateTime.Now.ToString("dd/MM/yyyy")}
07:59:55");
        }


private void Tempo_Tick(object sender, EventArgs e)
        {
            TimeRegistro = TimeRegistro.AddSeconds(1);
            lblHora.Text = TimeRegistro.ToString("HH:mm:ss");


            // Validações
                            if (TimeRegistro.Hour == 8 && TimeRegistro.Second > 3 &&
btnRegEntrada1.Enabled)
         {
                btnRegEntrada1.BackColor = Color.Red;
                btnRegEntrada1.Text += " (Auto) ";
                btnRegEntrada1.Enabled = false;
                }


private void btnSai1_Click(object sender, EventArgs e)
        {
                    TimeRegistro = DateTime.Parse($"{DateTime.Now.ToString("dd/MM/yyyy")}
11:59:55");
        }
            // Validações
```

```
                            if  (TimeRegistro.Hour  ==  12  &&  TimeRegistro.Second  >  3  &&
btnRegSaida1.Enabled)

         {

                btnRegSaida1.BackColor = Color.Red;

                btnRegSaida1.Text += " (Auto) ";

                btnRegSaida1.Enabled = false;

               }


private void btnEnt2_Click(object sender, EventArgs e)

        {

                    TimeRegistro  =  DateTime.Parse($"{DateTime.Now.ToString("dd/MM/yyyy")}
13:59:55");

        }

            // Validações

                            if  (TimeRegistro.Hour  ==  14  &&  TimeRegistro.Second  >  3  &&
btnRegEntrada2.Enabled)

        {

                btnRegEntrada2.BackColor = Color.Red;

                btnRegEntrada2.Text += " (Auto) ";

                btnRegEntrada2.Enabled = false;

         }




private void btnSai2_Click(object sender, EventArgs e)

        {

                    TimeRegistro  =  DateTime.Parse($"{DateTime.Now.ToString("dd/MM/yyyy")}
17:59:55");

        }
```

```
                  // Validações

                            if  (TimeRegistro.Hour == 18  &&  TimeRegistro.Second  >  3  &&
btnRegSaida2.Enabled)

                {

                    btnRegSaida2.BackColor = Color.Red;

                    btnRegSaida2.Text += "  (Auto)  ";

                    btnRegSaida2.Enabled = false;

                 }
```

About the Author

André Ruschel is a Researcher, an IT speaker, a Forensic Computer Scientist, a member of HTCIA (High Technology Crime Investigation Association), the author of the several books (MVP Microsoft Cloud and Datacenter Management), a specialist in interoperability and an inventor of ThinEco Cardboard Computer.

Malicious Mail Attachements

by HADBI Moussa Benameur

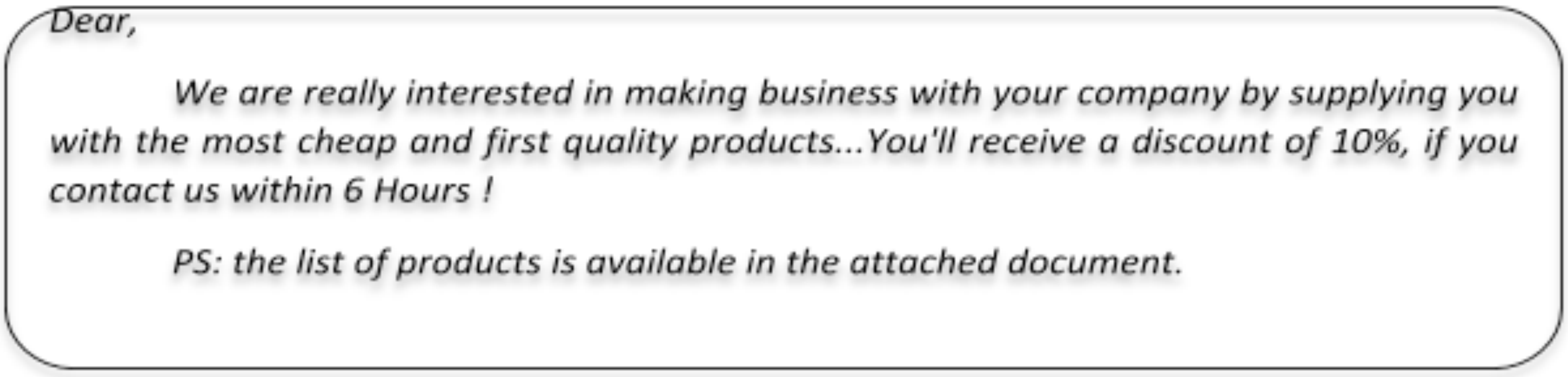
A large proportion of users across cyberspace are not properly trained (or not trained at all) to defend themselves against cyber-threats like phishing/spear-phishing, that's why cybercriminals mostly deliver their malware through malicious attachments. Also with social engineering techniques, mail messages received seem genuine and their mail attachments may seem clean (documents, spreadsheets, PDFs...), all this may deceive the user and lead him to download the attachment and sadly run the malware! In this article, we will demonstrate different techniques to let you download and safely analyze mail attachments.

Scanning & Inspecting the attachments

Mail attachments like documents, spreadsheets, and PDFs (for example) may include malicious content and you should train your employees to not open mail attachments sent by unknown persons; in fact, they should ask IT administrators/security to help them scan the attachments with an updated endpoint-security product. If nothing malicious has been detected, your IT administrators/security still may have to inspect the content with other tools: after all, spending 15 minutes, on average, inspecting a document is better than losing your data (encrypted with advanced variant ransomware).

We will give some practical examples about mail-attachment…

Let's consider this scenario: One of the employees of your company (the marketing department) receives an email from un unknown supplier:



Dear,

We are really interested in making business with your company by supplying you with the most cheap and first quality products...You'll receive a discount of 10%, if you contact us within 6 Hours !

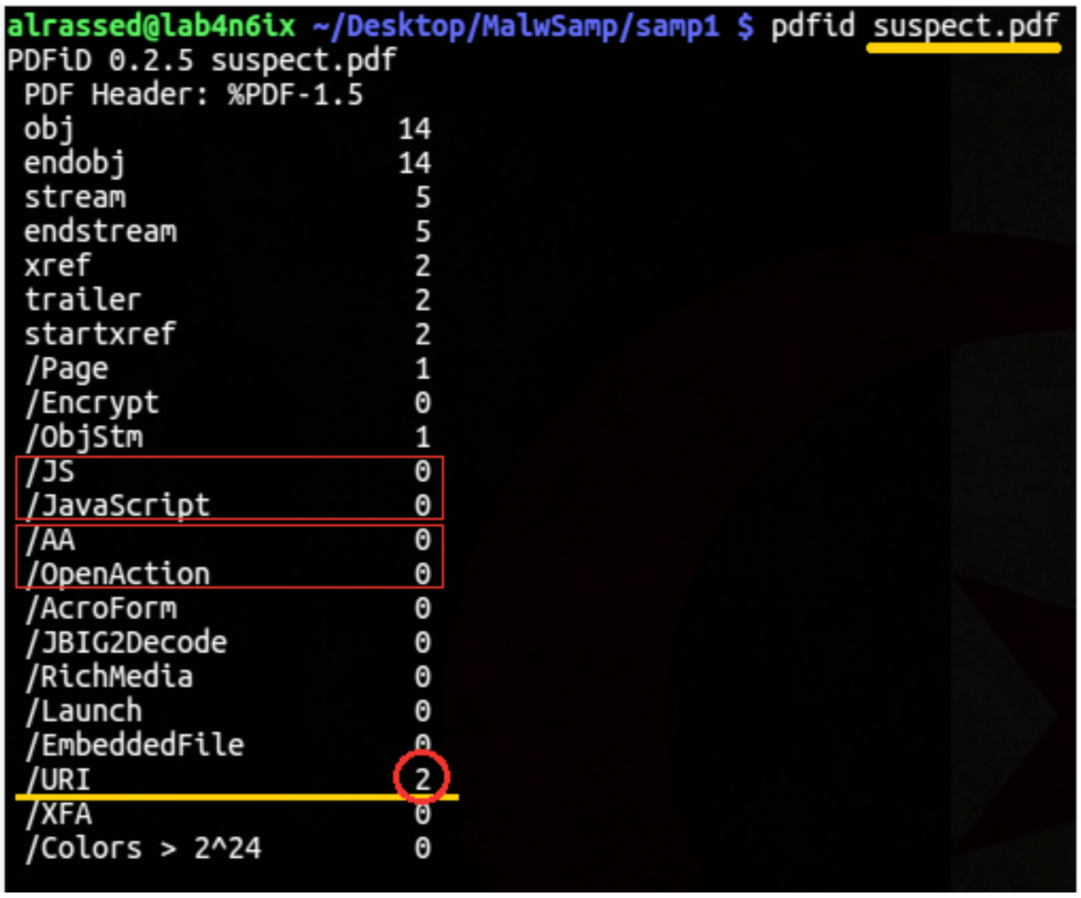
PS: the list of products is available in the attached document.

If your employee has been well trained, he'll ask for help instead of opening the document directly. If not, he may open the document and launch or download a malware. We assume that he contacted the IT/Security department for security advice:

Inspecting the attached document (pdf inspection)

The IT/security department should download the document but if their endpoint security doesn't recognize any threat within the received PDF attachment, they should inspect themselves the PDF document. Fortunately, there are plenty of tools that may help gather more information and one of them is: **pdfid.py**

pdfid*: It is a Python based script that scans the pdf, looking for specific keywords. You can download it from (https://blog.didierstevens.com/programs/pdf-tools/)



```
alrassed@lab4n6ix ~/Desktop/MalwSamp/samp1 $ pdfid suspect.pdf
PDFiD 0.2.5 suspect.pdf
 PDF Header: %PDF-1.5
 obj                   14
 endobj                14
 stream                 5
 endstream              5
 xref                   2
 trailer                2
 startxref              2
 /Page                  1
 /Encrypt               0
 /ObjStm                1
 /JS                    0
 /JavaScript            0
 /AA                    0
 /OpenAction            0
 /AcroForm              0
 /JBIG2Decode           0
 /RichMedia             0
 /Launch                0
 /EmbeddedFile          0
 /URI                   2
 /XFA                   0
 /Colors > 2^24         0
```

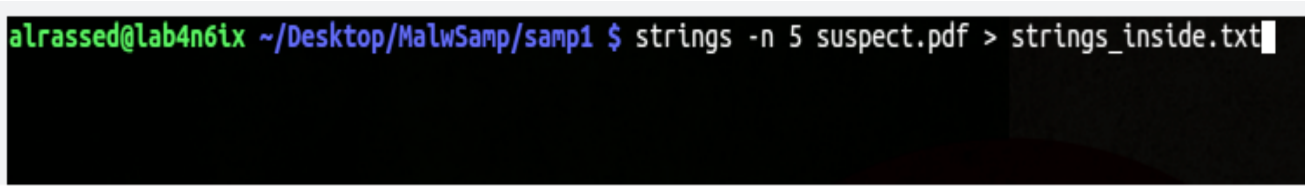
/JS /Javascript: indicates if the pdf document includes a JScript, this may indicate that the document is malicious.

/AA , and /OpenAction, indicates if an automatic action will be executed when the document is opened (launch the JScript without user interaction), this also may indicate that you're dealing with a malicious document.

To be sure about the action, you have to extract the JS script from the document and see exactly what the script is doing.

/URI Indicates URLs included in the pdf document: this may indicate that you'll be invited to open some URLs.

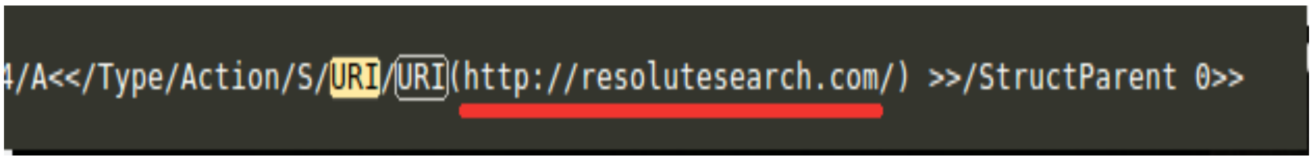
Let's investigate with the target URL; since there are no /JS and /Open Action, you can open the PDF file directly without clicking on any URLs, of course. But, if you're still afraid, you can use: **strings** (command under Linux)



```
alrassed@lab4n6ix ~/Desktop/MalwSamp/samp1 $ strings -n 5 suspect.pdf > strings_inside.txt
```

The command strings will search for any strings within the pdf (we aim to find the two URLs listed by pdfid.py before) and redirect all the matches to a text file: strings_inside.txt

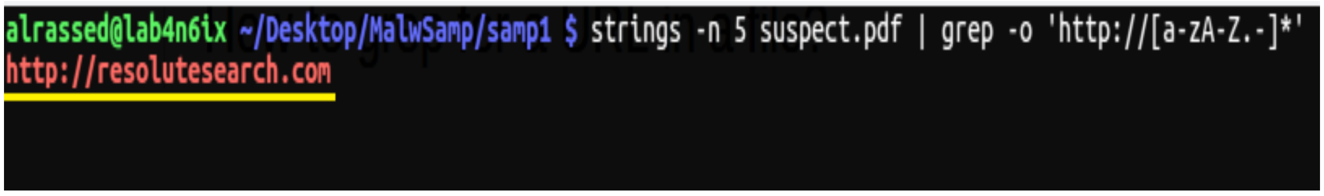
We open the resulted file with any text editor, and search for the keyword "URI".



```
4/A<</Type/Action/S/URI/URI(http://resolutesearch.com/) >>/StructParent 0>>
```

And bingo! It is the URL.

Another alternative, you could directly pipe the result from the strings command to **grep** command:

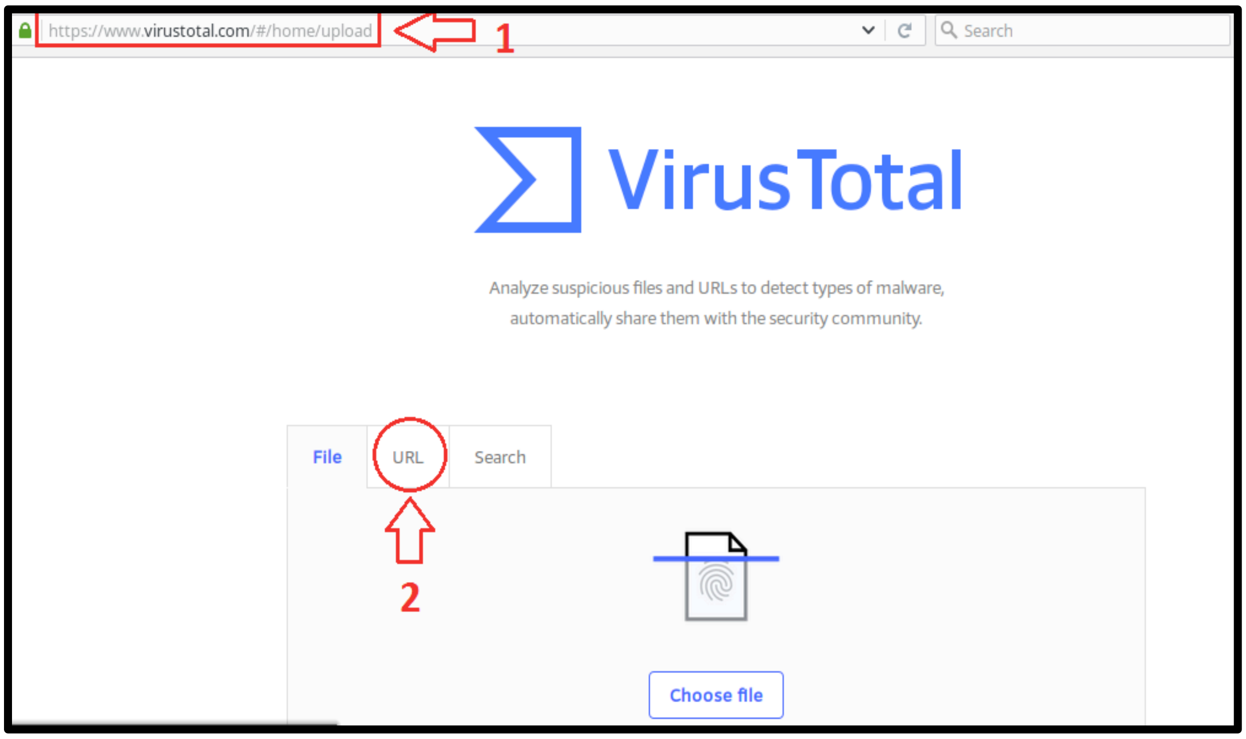


```
alrassed@lab4n6ix ~/Desktop/MalwSamp/samp1 $ strings -n 5 suspect.pdf | grep -o 'http://[a-zA-Z.-]*'
http://resolutesearch.com
```

Note: If you're under WinOs, you could use the Strings utility from Sysinternals suite (https://docs.microsoft.com/en-us/sysinternals/downloads/strings**).**

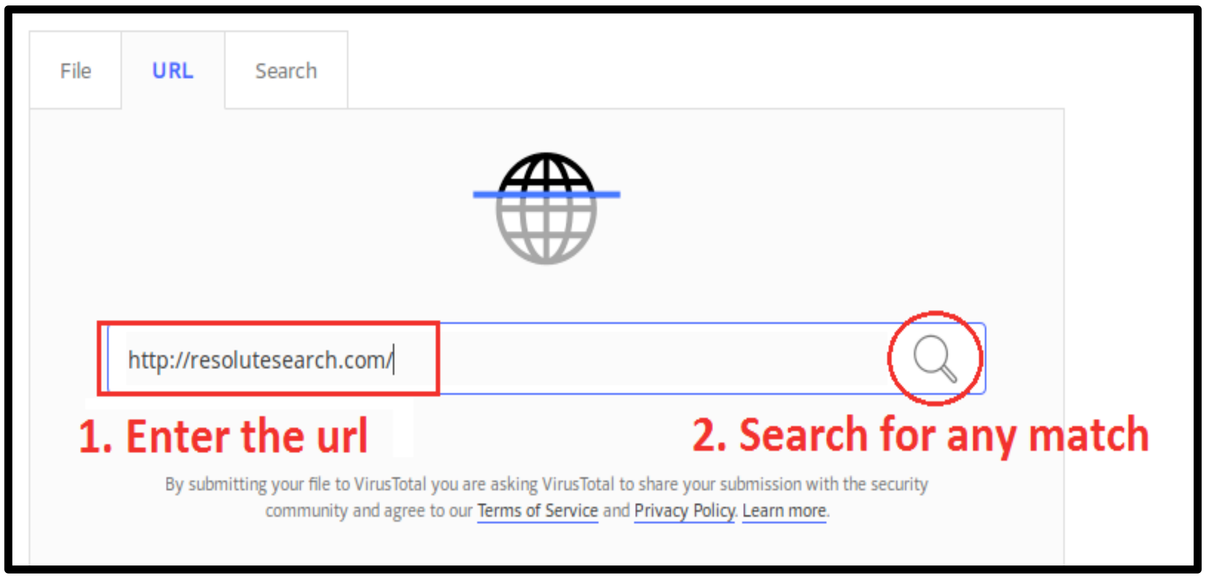
Now that we've found the URL, we search for it in VirusTotal: this gives you the ability to search for suspicious files and also "URLs".

Open in your browser: virus-total.com



Analyze suspicious files and URLs to detect types of malware, automatically share them with the security community.

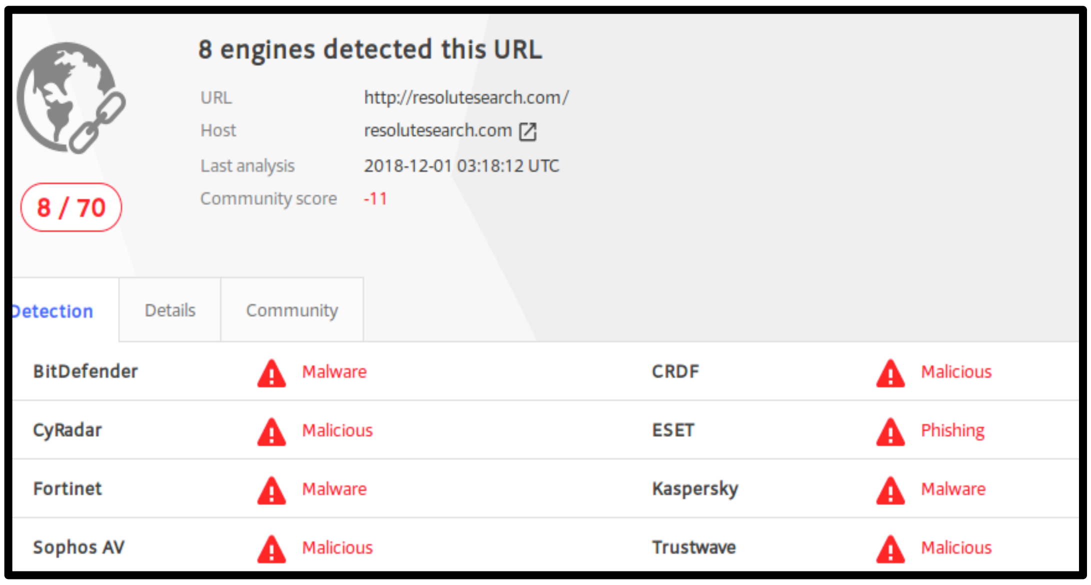
Enter the suspicious URL, and search for any match:



http://resolutesearch.com/

By submitting your file to VirusTotal you are asking VirusTotal to share your submission with the security community and agree to our Terms of Service and Privacy Policy. Learn more.

And voila!



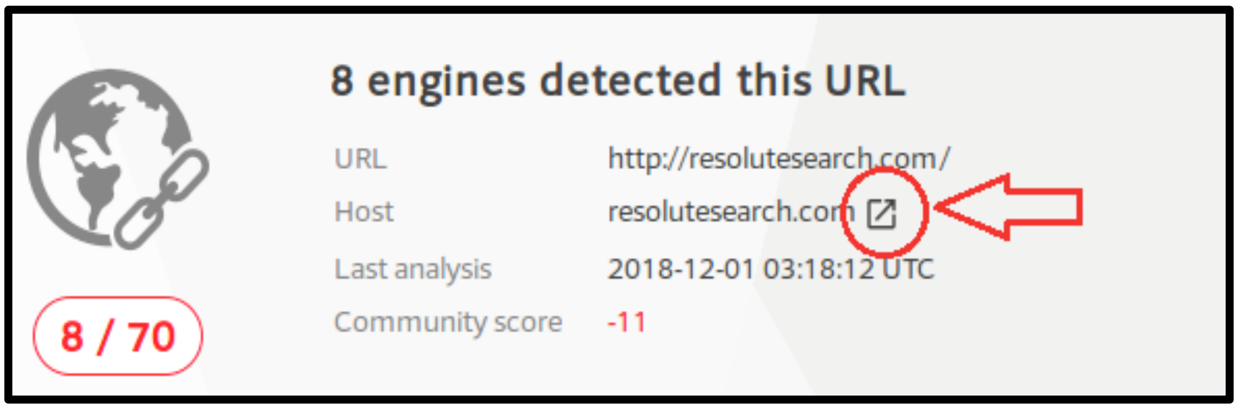
8 engines detected this URL

URL	http://resolutesearch.com/
Host	resolutesearch.com
Last analysis	2018-12-01 03:18:12 UTC
Community score	-11

8 / 70

BitDefender	Malware	CRDF	Malicious
CyRadar	Malicious	ESET	Phishing
Fortinet	Malware	Kaspersky	Malware
Sophos AV	Malicious	Trustwave	Malicious

For more information, click on:

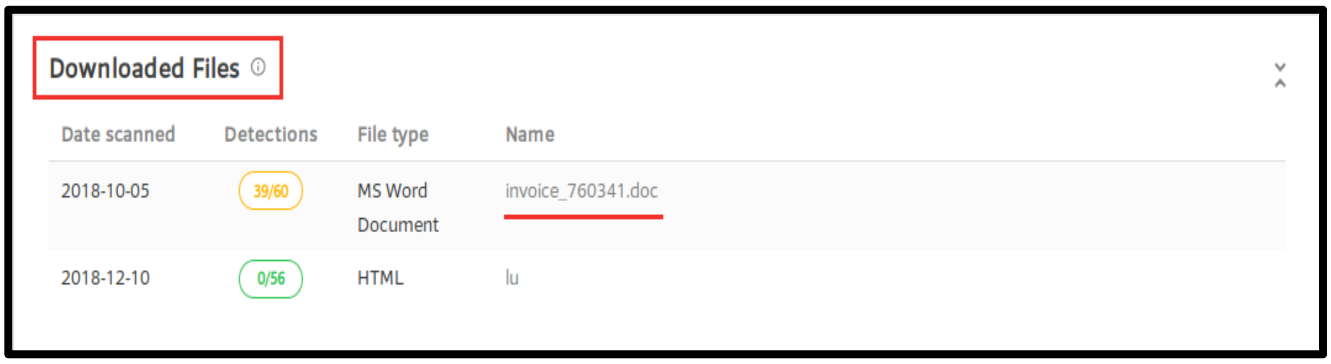


8 engines detected this URL

URL	http://resolutesearch.com/
Host	resolutesearch.com
Last analysis	2018-12-01 03:18:12 UTC
Community score	-11

8 / 70

And look for the downloaded file section:



Downloaded Files

Date scanned	Detections	File type	Name
2018-10-05	39/60	MS Word Document	invoice_760341.doc
2018-12-10	0/56	HTML	lu

We understand that when you open this URL you'll be invited to download another MS WORD document: It started with a phishing mail and now it is a Drive-by-Download attack! We know for sure that someone is trying to infect your company, and we shouldn't open any attachment.

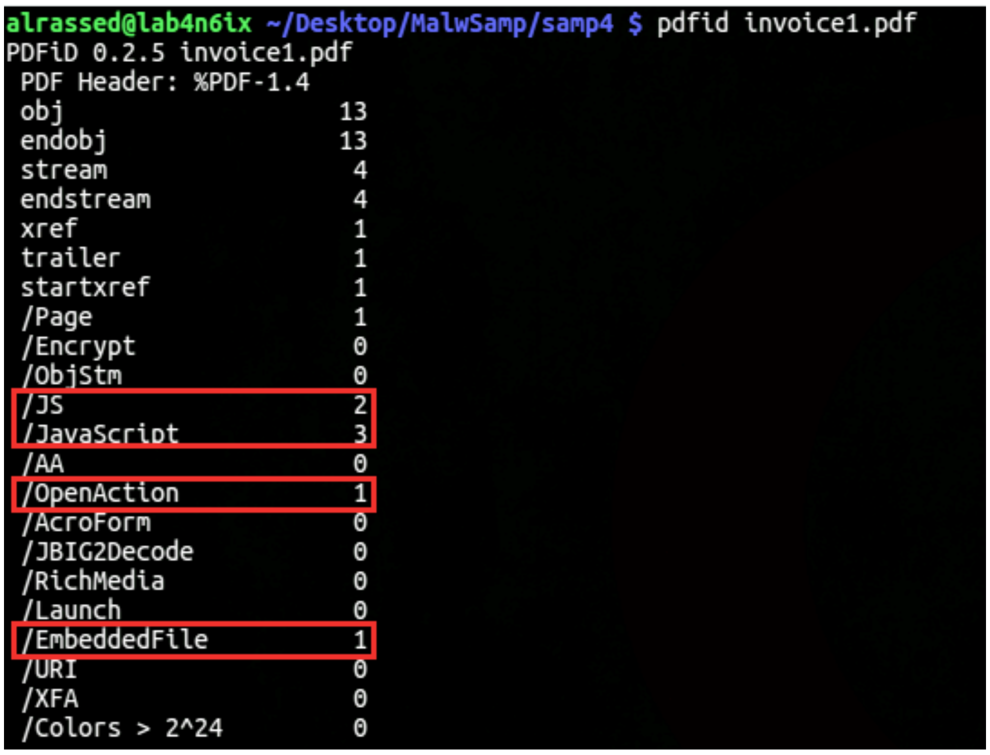
You should be aware that VirusTotal may not include any information about your URLs/File.

Let's do another scenario: We assume the employee didn't call the security department and opened that URL, then the MS Word document is saved on his host machine: What would happen if the user will open the Word document?

Inspecting the MS Document:

You should proceed carefully with MS Word documents; they may contain macros that execute some malicious script, or may download some other payload from a remote server.

Before analyzing the downloaded Word document, we'll analyze another PDF attachment (**invoice1.pdf**) just to show you how to proceed when you deal with PDF that contains /JS & /AA, and again we use the Python script: **pdfid**

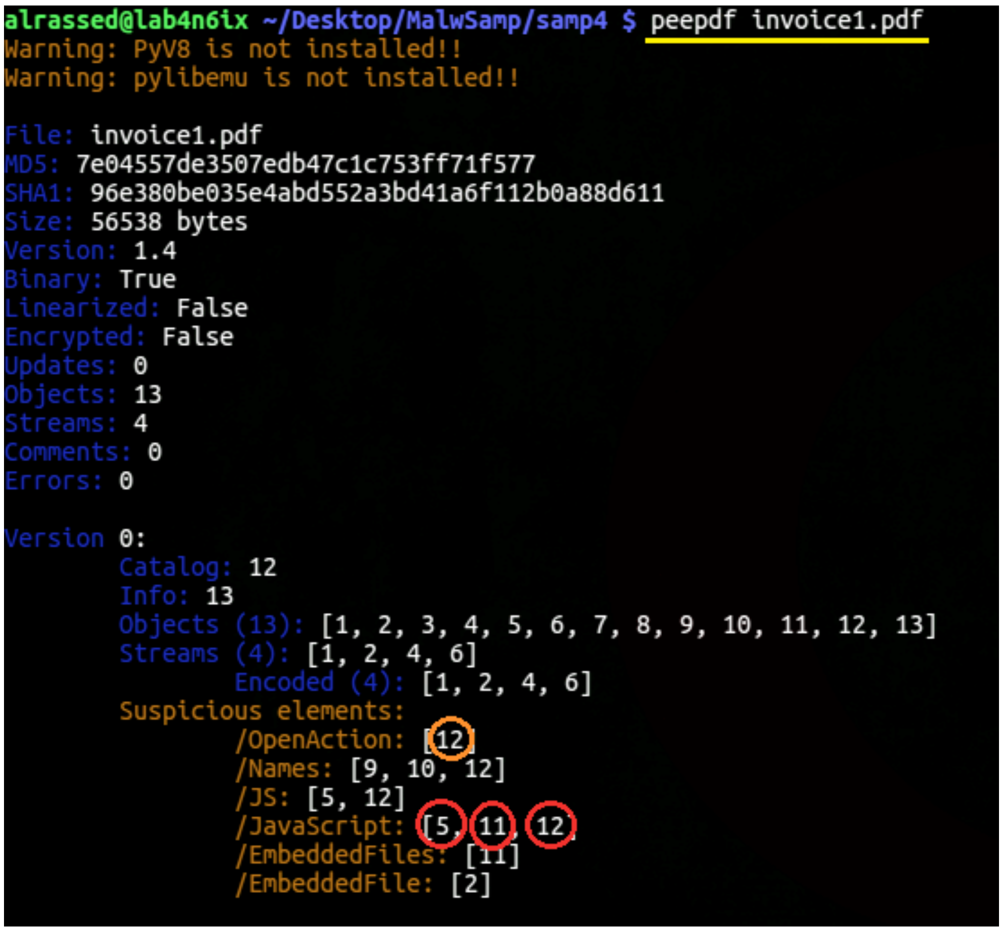


```
alrassed@lab4n6ix ~/Desktop/MalwSamp/samp4 $ pdfid invoice1.pdf
PDFiD 0.2.5 invoice1.pdf
 PDF Header: %PDF-1.4
 obj                   13
 endobj                13
 stream                 4
 endstream              4
 xref                   1
 trailer                1
 startxref              1
 /Page                  1
 /Encrypt               0
 /ObjStm                0
 /JS                    2
 /JavaScript            3
 /AA                    0
 /OpenAction            1
 /AcroForm              0
 /JBIG2Decode           0
 /RichMedia             0
 /Launch                0
 /EmbeddedFile          1
 /URI                   0
 /XFA                   0
 /Colors > 2^24         0
```

This time, we have something else:

When you open the PDF document (**/OpenAction 1**) there'll be an action performed by the script (**/JavaScript 3**), which may extract an (**/EmbeddedFile 1**)

To inspect the PDF more closely, we use another Python script: **peepdf.py** (http://eternal-todo.com/tools/peepdf-pdf-analysis-tool)

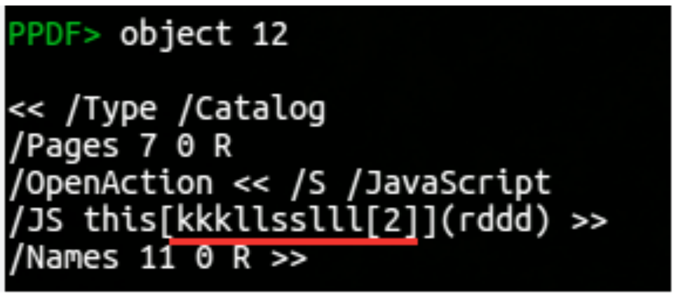


```
alrassed@lab4n6ix ~/Desktop/MalwSamp/samp4 $ peepdf invoice1.pdf
Warning: PyV8 is not installed!!
Warning: pylibemu is not installed!!

File: invoice1.pdf
MD5: 7e04557de3507edb47c1c753ff71f577
SHA1: 96e380be035e4abd552a3bd41a6f112b0a88d611
Size: 56538 bytes
Version: 1.4
Binary: True
Linearized: False
Encrypted: False
Updates: 0
Objects: 13
Streams: 4
Comments: 0
Errors: 0

Version 0:
        Catalog: 12
        Info: 13
        Objects (13): [1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13]
        Streams (4): [1, 2, 4, 6]
                Encoded (4): [1, 2, 4, 6]
        Suspicious elements:
                /OpenAction: [12]
                /Names: [9, 10, 12]
                /JS: [5, 12]
                /JavaScript: [5, 11, 12]
                /EmbeddedFiles: [11]
                /EmbeddedFile: [2]
```

You should type: peepdf –i your_pd_file.pdf , to enable the interactive mode!

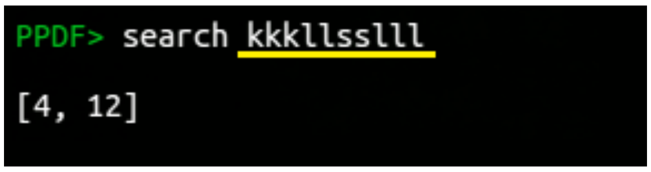
Let's dissect the behavior of this pdf file!

As you can see, when you open the document (/OpenAction) will be triggered and the object 12 is "called", let's check what it's about, type in the console: **object 12**



```
PPDF> object 12

<< /Type /Catalog
/Pages 7 0 R
/OpenAction << /S /JavaScript
/JS this[kkkllsslll[2]](rddd) >>
/Names 11 0 R >>
```

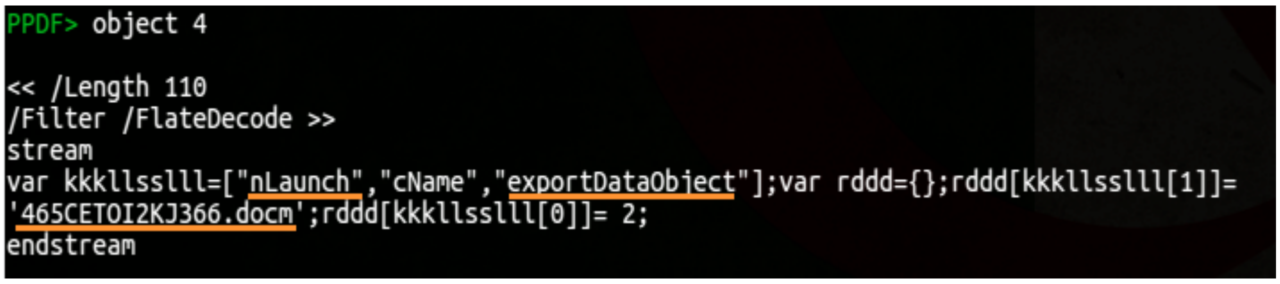
OpenAction has a reference to another object; to find where it resides use the command: search kkkllsslll



```
PPDF> search kkkllsslll

[4, 12]
```

We inspected before the object 12, let's check this time the object 4.

Again use the command: **object xx** {xx is the number of the object}



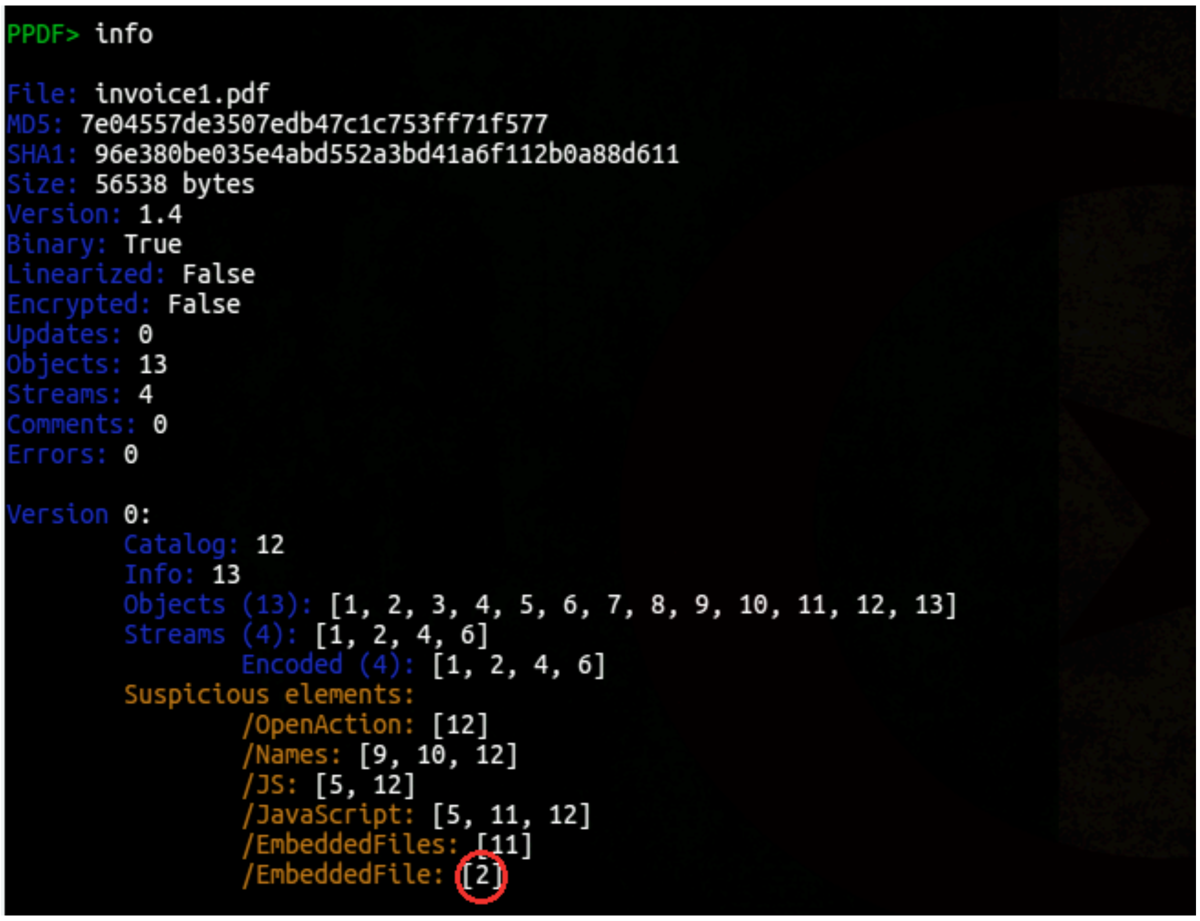
```
PPDF> object 4

<< /Length 110
/Filter /FlateDecode >>
stream
var kkkllsslll=["nLaunch","cName","exportDataObject"];var rddd={};rddd[kkkllsslll[1]]=
'465CETOI2KJ366.docm';rddd[kkkllsslll[0]]= 2;
endstream
```

By using alternatively the two commands **object** & **search**, you can learn more about the functionality of the scripts contained within your PDF, and sometimes you've to "guess" and not read all the code.

In our case, the script within the PDF file will launch another file "**465CETOI2KJ366.docm**"

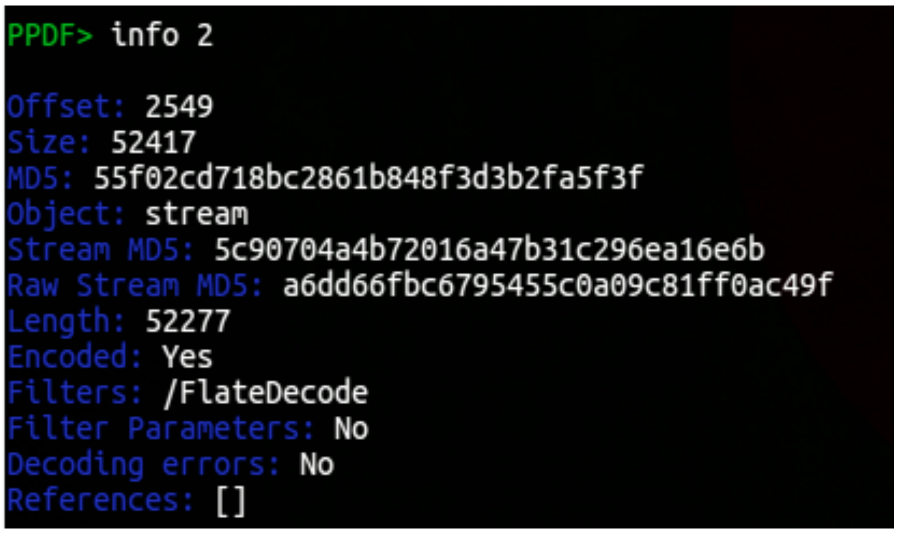
But where is this file stored?

Always within the peepdf console type: **info**, and look for embedded files.



```
PPDF> info

File: invoice1.pdf
MD5: 7e04557de3507edb47c1c753ff71f577
SHA1: 96e380be035e4abd552a3bd41a6f112b0a88d611
Size: 56538 bytes
Version: 1.4
Binary: True
Linearized: False
Encrypted: False
Updates: 0
Objects: 13
Streams: 4
Comments: 0
Errors: 0

Version 0:
        Catalog: 12
        Info: 13
        Objects (13): [1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13]
        Streams (4): [1, 2, 4, 6]
                Encoded (4): [1, 2, 4, 6]
        Suspicious elements:
                /OpenAction: [12]
                /Names: [9, 10, 12]
                /JS: [5, 12]
                /JavaScript: [5, 11, 12]
                /EmbeddedFiles: [11]
                /EmbeddedFile: [2]
```

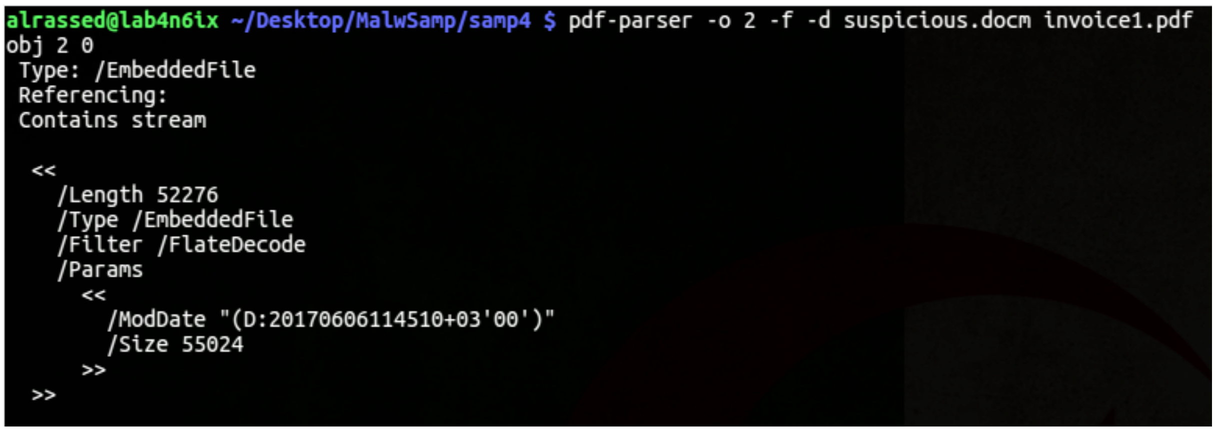
You can then type on the console: **info xx** {**xx the number of object**}



```
PPDF> info 2

Offset: 2549
Size: 52417
MD5: 55f02cd718bc2861b848f3d3b2fa5f3f
Object: stream
Stream MD5: 5c90704a4b72016a47b31c296ea16e6b
Raw Stream MD5: a6dd66fbc6795455c0a09c81ff0ac49f
Length: 52277
Encoded: Yes
Filters: /FlateDecode
Filter Parameters: No
Decoding errors: No
References: []
```

This embedded object has a size of 51Kb so we should be very suspicious now: this may contain a malicious content.

Time to extract this object from the PDF attachment. We'll use another Python utility: **pdfparser**



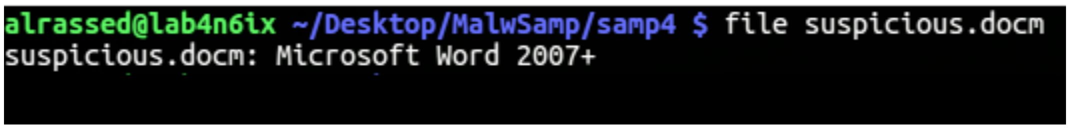
```
alrassed@lab4n6ix ~/Desktop/MalwSamp/samp4 $ pdf-parser -o 2 -f -d suspicious.docm invoice1.pdf
obj 2 0
 Type: /EmbeddedFile
 Referencing:
 Contains stream

  <<
    /Length 52276
    /Type /EmbeddedFile
    /Filter /FlateDecode
    /Params
      <<
        /ModDate "(D:20170606114510+03'00')"
        /Size 55024
      >>
  >>
```

-o 2 (The number of objects to extract, in our case 2)

-f Applies a filter to the object (/FlateDecode for our embedded file)

-d xxxxx.doc (dump the stream of object 2 to a target file: suspicious.docm)

This will produce the file suspicious.docm, which we will check with the Linux command **file**:

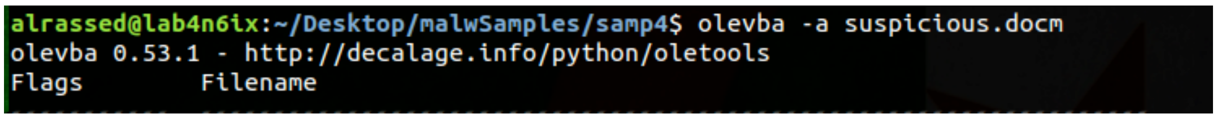


```
alrassed@lab4n6ix ~/Desktop/MalwSamp/samp4 $ file suspicious.docm
suspicious.docm: Microsoft Word 2007+
```

You should ask yourself:

Why is a Word document embedded within a pdf document?

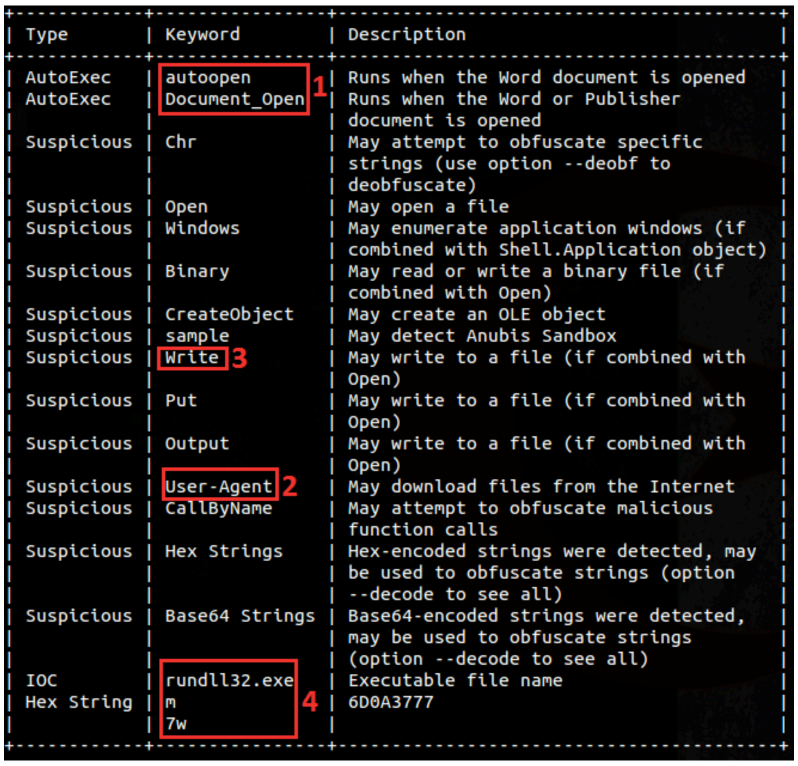
Let's inspect our "potential" malicious document with another utility: **olevba** (https://github.com/decalage2/oletools/wiki/olevba)



```
alrassed@lab4n6ix:~/Desktop/malwSamples/samp4$ olevba -a suspicious.docm
olevba 0.53.1 - http://decalage.info/python/oletools
Flags        Filename
```

Run the command: olevba –a suspicious.docm

{-a : To display only the analysis without displaying the VBA-Macro}



```
+------------+-----------------+-----------------------------------------+
| Type       | Keyword         | Description                             |
+------------+-----------------+-----------------------------------------+
| AutoExec   | autoopen        | Runs when the Word document is opened   |
| AutoExec   | Document_Open   | Runs when the Word or Publisher         |
|            |                 | document is opened                      |
| Suspicious | Chr             | May attempt to obfuscate specific       |
|            |                 | strings (use option --deobf to          |
|            |                 | deobfuscate)                            |
| Suspicious | Open            | May open a file                         |
| Suspicious | Windows         | May enumerate application windows (if   |
|            |                 | combined with Shell.Application object) |
| Suspicious | Binary          | May read or write a binary file (if     |
|            |                 | combined with Open)                     |
| Suspicious | CreateObject    | May create an OLE object                |
| Suspicious | sample          | May detect Anubis Sandbox               |
| Suspicious | Write           | May write to a file (if combined with   |
|            |                 | Open)                                   |
| Suspicious | Put             | May write to a file (if combined with   |
|            |                 | Open)                                   |
| Suspicious | Output          | May write to a file (if combined with   |
|            |                 | Open)                                   |
| Suspicious | User-Agent      | May download files from the Internet    |
| Suspicious | CallByName      | May attempt to obfuscate malicious      |
|            |                 | function calls                          |
| Suspicious | Hex Strings     | Hex-encoded strings were detected, may  |
|            |                 | be used to obfuscate strings (option    |
|            |                 | --decode to see all)                    |
| Suspicious | Base64 Strings  | Base64-encoded strings were detected,   |
|            |                 | may be used to obfuscate strings        |
|            |                 | (option --decode to see all)            |
| IOC        | rundll32.exe    | Executable file name                    |
| Hex String | m               | 6D0A3777                                |
|            | 7w              |                                         |
+------------+-----------------+-----------------------------------------+
```

The summary gives us a "big picture" of the behavior of the malicious word document:

1. When the user opens the document, a script will be run

2. This script will download a file from the internet (the payload)

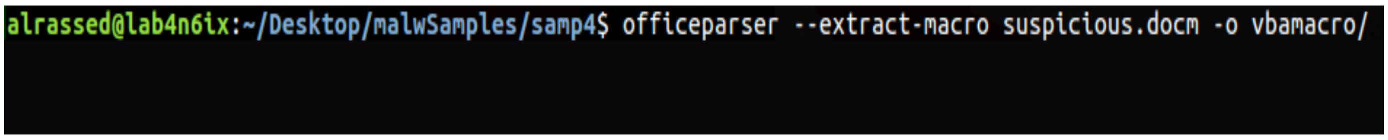
3. The payload will be saved somewhere in the victim's machines

4. The script will run the payload through rundll32.exe

You can also extract all the macro code, by using the option –c and dump the result to a text file: olevba –c suspicious.docm > macro.txt

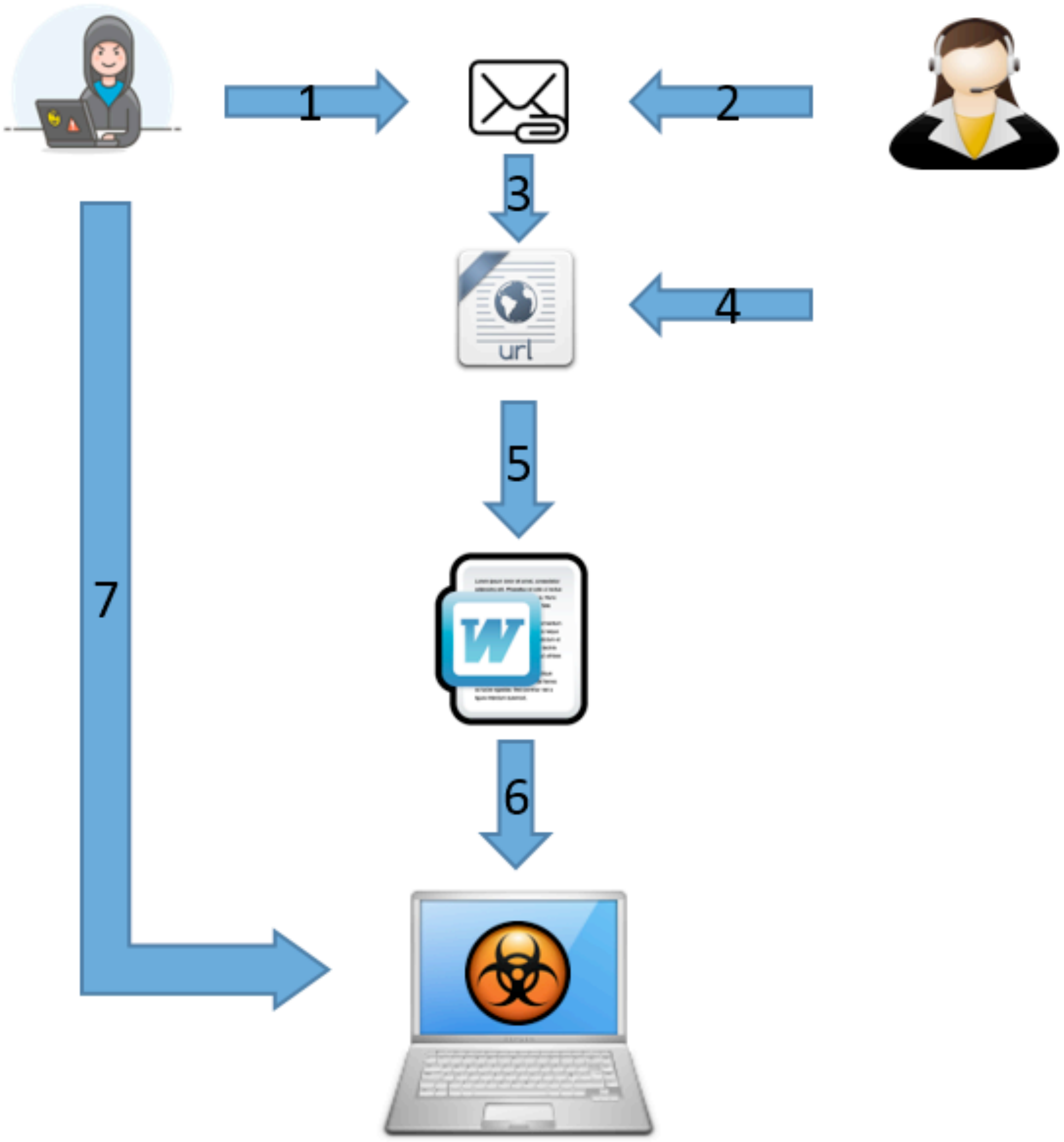
Or,

You can extract each part of the macro separately by using: officeparser.py (https://github.com/unixfreak0037/officeparser)



```
alrassed@lab4n6ix:~/Desktop/malwSamples/samp4$ officeparser --extract-macro suspicious.docm -o vbamacro/
```

This will extract (--extract-macro) all the macros existing on the "suspicious.docm" to the directory vbamacro (-o)



1. The cybercriminal sends the malicious attachment to the victim,
2. The victim opens the malicious attachment,
3. The attachments invite the victim to click on a link,
4. The victim clicks on the malicious URLs,
5. A Word Office document is downloaded automatically on the victim machines,
6. When the victim opens the Word Office document, a macro is run automatically and malicious software is installed on the victim's machine,
7. The attacker takes control of the victim machines.

The Shellcode case!

Until now we've seen how a cybercriminal through phishing (harpooning) could download malicious software and install this malware on the victim's machine only with JavaScript or VBA macro, or through drive-by-download attack, but cybercriminals have other tricks and one of them is sending an attachment that contains a shellcode that exploits a vulnerability within the target system and it could be:

- The web-browser or a web-browser plugin …
- PDF Reader…
- Microsoft Office suite…
- Etc…

Dealing with shellcode isn't an easy task even if there are tools that may help you identify malicious content; for example, you could use:

- OfficeMalScanner to scan Office documents (RTF, or DOC…) and find any shellcode signatures/pattern matches
- Peepdf can also help inspect your PDF attachment, after that you could extract the malicious content with pdf-parser for further analysis

You could also perform a basic dynamic analysis by opening your document in a safe environment and monitor your system for any malicious behavior (network traffic, file system, registry, etc.).

Basic dynamic analysis

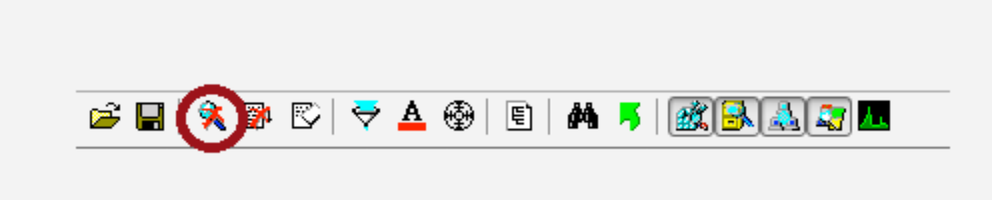
If you're unsure about a mail attachment after examination with the techniques above, you could open it in an isolated environment, such as a virtual machine, and perform a basic dynamic analysis, which consists of opening the mail attachment and releasing "any evil code" hiding inside.

Tools like "procmon" from sysinternals (https://docs.microsoft.com/en-us/sysinternals/downloads/procmon) could help you monitor:

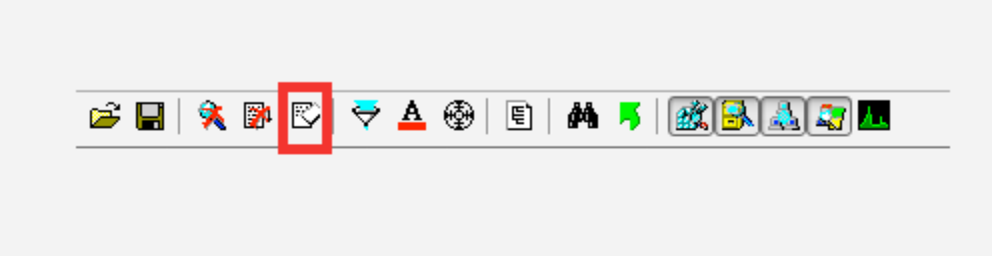
- *File activity*: Files that could be opened, created or deleted by the "evil code" within your mail attachment
- *Process activity:* Processes that could be created by the "evil code"
- *Registry activity:* Register entries that are opened, created, or deleted by the "evil code"
- *Network activity:* The malware within the attachment, could open a port –backdoor-, send some DNS queries, or download other malwares on the infected machine…

Let's give it a try…

Run "procmon" and stop monitoring by clicking on (or **CTRL+E**):

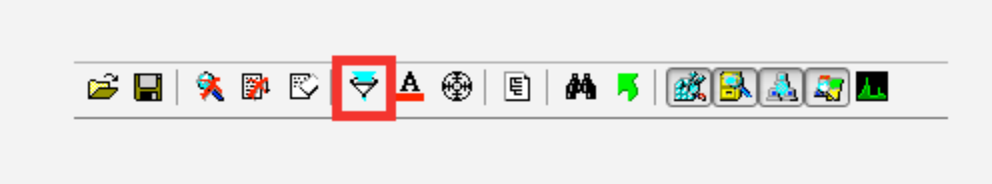


Then clear all monitored events by clicking on (or **CTRL+A**):

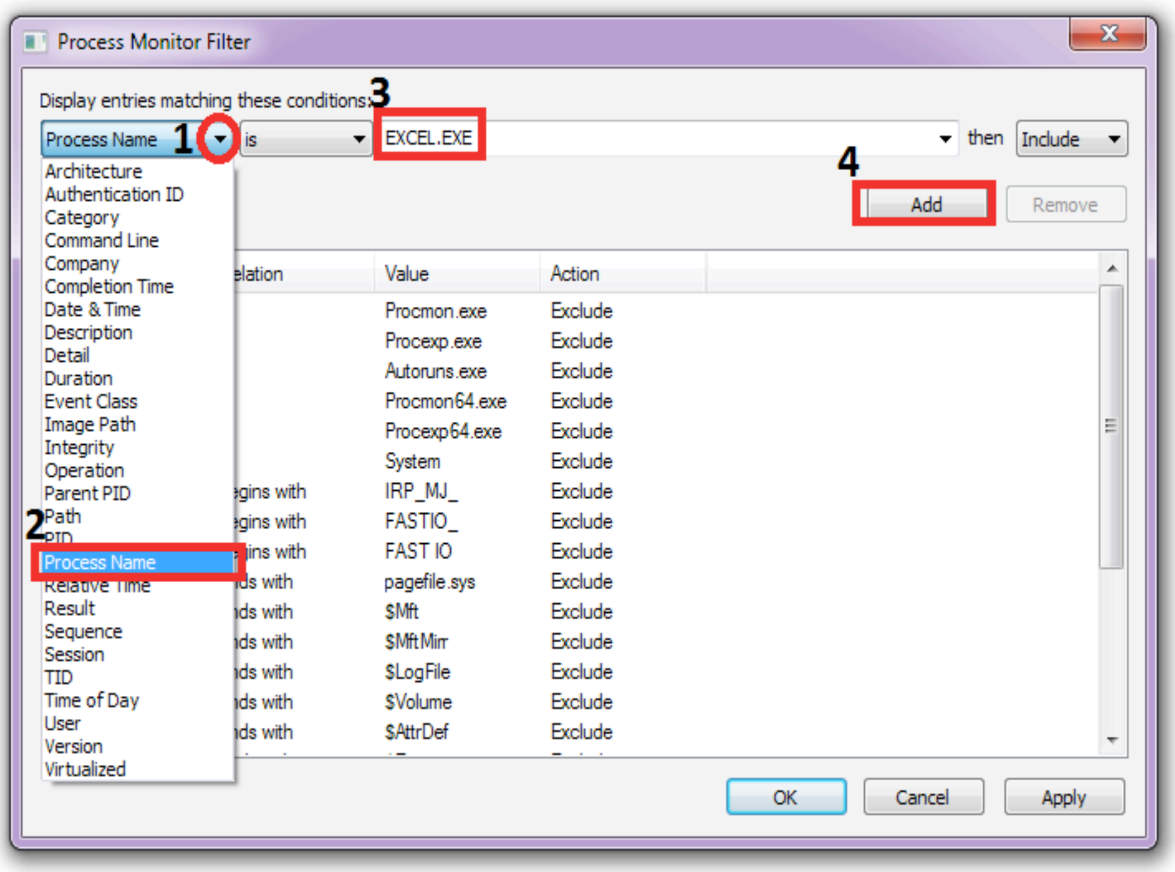


Before we can start, we should create a filter because procmon by default will monitor all system calls (thousands of events per minute), but in our case we focus our attention on EXCEL process (you could try any other process that launch when you open a mail attachment: pdf reader, Word, etc.):

To create the filter, click on (or **CTRL+L**):



Next,

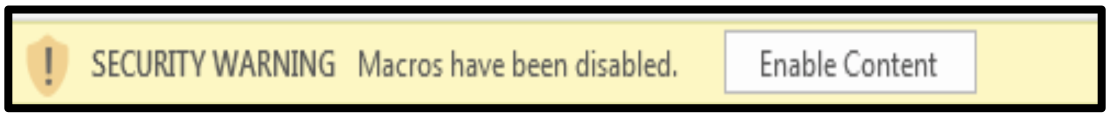


elation	Value	Action
	Procmon.exe	Exclude
	Procexp.exe	Exclude
	Autoruns.exe	Exclude
	Procmon64.exe	Exclude
	Procexp64.exe	Exclude
	System	Exclude
gins with	IRP_MJ_	Exclude
gins with	FASTIO_	Exclude
ins with	FAST IO	Exclude
ds with	pagefile.sys	Exclude
ds with	$Mft	Exclude
ds with	$MftMirr	Exclude
ds with	$LogFile	Exclude
ds with	$Volume	Exclude
ds with	$AttrDef	Exclude

1. Click on the list
2. Choose from options Process Name
3. Type the process name to monitor (EXCEL.EXE in our example)
4. Click ADD

Next time when you click on capture (**CTRL+E**), procmon will monitor only the activity of the process EXCEL.EXE…

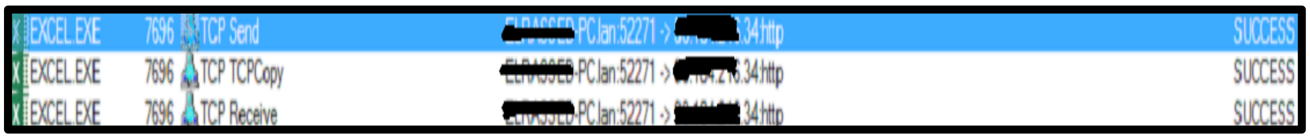
Now, let's release "the evil code" and see what is happening behind the scene:



SECURITY WARNING Macros have been disabled.

By default, Excel disables macros for security reasons (and you should never enable the content if you're unsure about the mail attachment you've received) but since we're running in a safe environment, we enable the content!

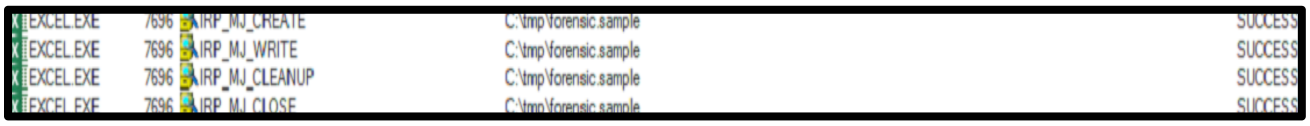
If it's a malicious attachment, you should see some weird activity, for example, in my case:



```
EXCEL.EXE   7696 TCP Send       [redacted]-PC.lan:52271 -> [redacted].34:http   SUCCESS
EXCEL.EXE   7696 TCP TCPCopy    [redacted]-PC.lan:52271 -> [redacted].34:http   SUCCESS
EXCEL.EXE   7696 TCP Receive    [redacted]-PC.lan:52271 -> [redacted].34:http   SUCCESS
```

There is some network activity: The process EXCEL.EXE (in fact, it's the malicious macro) is connecting to a distant server, and receiving something.

Next, there is some file activity:



```
EXCEL.EXE   7696 IRP_MJ_CREATE    C:\tmp\forensic.sample   SUCCESS
EXCEL.EXE   7696 IRP_MJ_WRITE     C:\tmp\forensic.sample   SUCCESS
EXCEL.EXE   7696 IRP_MJ_CLEANUP   C:\tmp\forensic.sample   SUCCESS
EXCEL.EXE   7696 IRP_MJ_CLOSE     C:\tmp\forensic.sample   SUCCESS
```

As you can see, the received content is saved in the disk with the name "forensic.sample".

So, this is a simple example of basic malware analysis, but with real malware, you could monitor other activities like process creation, and registry interaction, etc., and with this "tactic", you could rapidly unveil other malicious behavior.

About the Author

HADBI Moussa Benameur, based in Algeria graduated from Algerian university and CISCO Cybersecurity certified. Working on Healthcare, as Network & IT security manager with an experience of 8 years in the security domain, Im also doing research on computer forensics and malware analysis.

Obtaining your Certified Forensic Computer Examiner Certification- Tips and Tricks

by Matt Beers, CFCE

The purpose of this article is not to scare you away from trying to get your CFCE, but rather educate the reader of the CFCE program how best to prepare for its process and be successful. I'll bold little tips and tricks that will help you along the way.

The International Association of Computer Investigative Specialists, otherwise known as IACIS, is a world leader in digital forensics training and offers one of the most sought-after certifications in the digital forensic world, the Certified Forensic Computer Examiner (CFCE).

Achieving the CFCE certificate is a process that can take close to six months and it is not one for the faint of heart. One friend of mine who received her CFCE before I did, told me to expect that on at least one occasion, you'll be sitting on your kitchen floor, crying and drinking from a bottle of wine. This was true for her. Me; I enjoyed a lot of whiskey during my days and nights working through the process.

The purpose of this article is not to scare you away from trying to get your CFCE, but rather educate the reader of the CFCE program how best to prepare for its process and be successful. I'll bold little tips and tricks that will help you along the way.

To give you a little background on me, I come from a small police department in Colorado who didn't have digital forensic capabilities in house. Because of this, we weren't seizing digital evidence to help strengthen our cases. In my previous career, I worked in the Information Technologies field where I was a jack of all trades, master of none.

Naturally, after becoming a police officer, my interest started to point me back towards computers and technology. I started by joining the Internet Crimes Against Children Task force also referred to as ICAC. This allowed me to take various classes and it got me into a couple of forensic focused classes. I was hooked. I managed to build small capabilities on a budget (which may be another article someday) and started doing basic forensic cases. I reached out to a mentor at a neighboring jurisdiction and asked him what I needed to do next and he told me to get my CFCE.

When I looked into attending the BCFE (Basic Computer Forensic Examiner) course in Florida for two weeks, I was really put off by the cost. This is a huge struggle for a lot of people and a lot of agencies, but this is one of the most sought after and prestigious certifications in our industry. **If you are on any task forces that have money to give your agency for training or equipment, that might be a solution to the money problem for you**. At the time of the BCFE event, I had no in depth digital forensic training (if I couldn't find the file by a known hash set, I didn't know what to do), I felt like I needed to start with the BCFE course. **For those readers who have a strong grasp of the different file systems, your tools and looking at hexadecimal, you may opt to test for your CFCE through the external certification process where it isn't necessary to attend the BCFE event. Check the IACIS website for pre-requisites for the External CFCE process.**

The BCFE course is a two week course hosted by IACIS in Florida, while also hosting several other advanced training courses. The BCFE training is intense. During this training you'll be drinking from the fire hose, but they start at the basics of how to use the start menu and progress from there - I saw one person after the second day, politely close up his laptop, pack up his bag and never return. Just breath, you'll make it through. You aren't expected to have it all memorized and there are labs after class that you can attend if you wish.

When you attend the BCFE event as a student, you receive a backpack of goodies, you get a laptop computer (it isn't amazing but it is more than sufficient to complete the CFCE testing process), a write-blocker, a USB 3.0 hard drive, a thumb drive and the training manuals. My class also received a one year license to Forensic Explorer and that was the main tool we used during the training.

I was bright eyed and bushy tailed when I got to Florida for the BCFE event. The weather was warm, the hotel was beautiful, and I was so excited to start! My BCFE class was the largest to date and I signed up after they were sold out. They then allowed additional registrations. This was supposed to push my testing back to December, but by chance, I was moved to the June testing group. After two weeks of staring at hexadecimal code, byte sweeping and head spinning, I was really intimidated to start the testing process.

The CFCE Testing Process

Obtaining your CFCE requires the successful completion of two phases; The Peer Review Phase and the Certification Phase.

During the Peer Review phase you are given four practical problems sets to complete in order. You are given thirty days to complete each problem set with 100 percent accuracy. During these thirty days, you will submit your answers to a coach to review. You coach will check the answers for accuracy and send you back your results. You then review and make corrections and repeat.

The goal of the coach is to help guide you through the problem set to make sure you understand the learning points, but they aren't going to answer the question for you. In fact, you may have a coach who will just tell you that it is wrong with no explanation. They want you to try to figure this out. **During the Peer Review Phase you are able to use any tool and resource of your choice with the exception of talking to another individual who is not your coach.** (Reddit, message boards, colleagues, etc., are completely forbidden to be used as a resource for figuring out the problems.)

The biggest piece of advice I can offer here is that **within the first 10 days of a problem set, you need to have already submitted your answers to your coach at least once.**

I heard a rumor from one of the coaches that after the first thirty days, 20% CFCE candidates in one of the regions had failed to complete the first problem set and therefore were unable to continue. I find this number to be pretty accurate because out of the five people I made friends with while at the BCFE class, three of them were unable to complete the entire process for various reasons. Each of these individuals were already working in the Digital Forensic world.

Your success will completely dependent on your time management and your dedication- Are you only going to work on this during paid work time or do you want this bad enough that you're going to give up your summer plans to ensure your success? Your coach during this process likely has a full time job, a family and is also busy, but he has volunteered to coach. So get the answers to the problem set sent in early and fast. When **your coach sends you back corrections, move heaven and earth to look at them and start making those corrections to get the problem set back to the coach as soon as possible. The more dedication you show to the process, the more dedication your coach is going to show to you.** You can't expect to submit your problem set for the first time on day twenty-eight of thirty and expect that within a few hours your coach will have corrections back to you. You also shouldn't expect that any corrections you are going to make will take you less than three hours.

I wasn't paid work time to complete these problem sets. I have a family and I had to work patrol full time, every day off was spent in front of the computer looking at my manuals, looking at the screen and answering questions. **Plan to dedicate about 30-40 hours of time to each problem set. This number may vary on your experience, but I feel like as someone who had close to zero experience and speaking with experience examiners, this is pretty fair.**

Another hint **- If you complete a problem set early, you can start on the next problem set before the allotted time for the previous problem has ended.** So while you have thirty days for each problem set, you don't have to wait four months to complete all the problem sets- because of this I was able to complete all four peer review problems in two months and move into the Certification Phase.

Once you reach the certification phase, you are on your own. You can no longer reach out to your coach asking for help or advice. The Certification Phase is made up of two "tests". The first is a hard drive practical where you are given 30 days to answer various questions about a hard drive image. This is another reason to get through the Peer Review Phase quickly, so you're able to retain and use the knowledge you gained and apply it to the Certification Phase. There is not much to be said about the hard drive problem other than it is similar to peer review but without the assistance of a coach and the questions are all completed through an online portal.

The hard drive problem was difficult and, at times, really stretched my digital forensic knowledge. Once you feel confident in the answers you have, you can submit the answers to your hard drive practical where it is immediately graded. You are required to get an 80% on the practical. This is pass/ fail- you don't get to know your score- you just know you got at least an 80% if it allows you to proceed with the Knowledge Based Test.

The Knowledge Based Test is 100 questions, also online, of general forensic knowledge. You will be given 14 days to complete this test. After I completed the hard drive practical, I was pretty mentally exhausted. I had planned on taking at least a three day break before starting the Knowledge Based Test- but I decided to just take a look at the test to see what I was getting into. I guess when your mind is in the zone, it's in the zone, and I had completed 69 of the questions immediately after submitting my practical test. **Use your motivation and successes to keep you going- Don't allow failure to be an option for you.**

The written test, once submitted, is physically reviewed by a real person to give you any benefit of the doubt on answers; this can take several painfully long and stressful days before you hear if you passed and completed the CFCE process. But once you do hear back that you passed, you complete some paperwork and you have officially joined the ranks of us Certified Forensic Computer Examiner.

While the whole process seems extremely stressful (which it is), I have yet to complete something that has been so rewarding for me. After completing my process and having dedicated so much time and effort into it, I found myself somewhat lost and bored with all my newly found free time. So, I decided that I wanted to help others through this process. I quickly signed up to be a coach for the next testing cycle, which I am extremely excited about! I also have applied to go on staff with IACIS and hopefully teach some of these practices and principals at the next BCFE event in 2019. If I make it, I would love to meet any of you that read this article and are attending the event.

About the Author

Matt Beers is a law enforcement professional and digital forensic examiner. His career started in the corporate software world and for a brief time he transitioned into the intelligence sector of government contracting where he focused on social media exploitation before making the jump to full time law enforcement. Matt resides in the Rocky Mountains of Colorado with his wife and daughter.

Digital Forensics and the Law

by Doug Carner

The public's blind faith has been reinforced by television and movies that depict forensic science as being both unlimited and infallible. The reality is that classic forensic practices originated in the field, far from labs and scholars, and was based on unproven assumptions. This stemmed from the need for investigative tools, outpacing the deployment of the peer reviewed science to validate those tools. The end result has been wrongful convictions based upon junk science that has been codified into case law.

In 2006, the U.S. Department of Justice sought to quantify the public's confidence with forensic science. One such study found that jurors trusted forensic evidence far more than the testimony of the police, eyewitnesses, or victims.

The public's blind faith has been reinforced by television and movies that depict forensic science as being both unlimited and infallible. The reality is that classic forensic practices originated in the field, far from labs and scholars, and was based on unproven assumptions. This stemmed from the need for investigative tools, outpacing the deployment of the peer reviewed science to validate those tools. The end result has been wrongful convictions based upon junk science that has been codified into case law.

For example, polygraphs have existed for nearly a century, have been at the center of high profile convictions, and are generally perceived as trustworthy by the public. This is inconsistent with the well-documented cases of subjects cheating the polygraph to produce their desired results, and the National Academy of Sciences determining Polygraphs to be unreliable, unscientific and biased. Despite these facts, public confidence in this pseudo-science remains strong.

Fingerprint analysis had a similar origin. However, after a century of review and refinement, fingerprint analysis has become an accepted science. At issue is its high rate of false matches, primarily due to assumptions that must be made by the reviewing analyst.

In my specialty of audio-video enhancement and authentication, practitioners routinely apply judgment based assumptions and filters without an adequate level of training or peer review. My colleagues boast about how they just figure it out as they work. Unfortunately, the courts have accepted prior work from these "experts", so their unsubstantiated results continue to be unchallenged due to those precedents. This has resulted in a proliferation of flawed evidence being accepted as fact into the court.

In 2013, I testified in a case that centered around altered video evidence. This opinion was easy to prove because the manipulations were crude and obvious. Later that same year, I testified in a federal case where the hiring attorney was only given a subtly altered video during discovery, and then the opposing counsel sought a Motion for Summary Judgment due to our opinion's reliance upon that flawed video. While that decision was overturned on appeal, that manipulated evidence should never have passed through the court as legitimate evidence.

The unfortunate reality is that, among all legal cases in the United States, faulty forensic science is second only to faulty eyewitness testimony as the leading causes of wrongful convictions. There are serious and harmful consequences when faulty forensic science is able to reach the courtroom, and it is unreasonable to expect judges to possess the required expertise in each area of science to make that determination.

To avoid forming flawed opinions, analysts must test everything, and assume nothing. Unfortunately, video authentication testing was still in its infancy in 2013, and analysts generally had to accept the integrity of their discovery as fact. In response, scientists and practitioners of the Forensic Working Group developed a scientific approach to evaluate the authenticity of recorded multimedia files. This led to a suite of relevant peer-reviewed tests being compiled into a logical workflow called the Multimedia Authentication Testing form, or MAT for short.

The MAT provides a roadmap to assist the analyst in assessing key file metrics and characteristics, including hidden metadata. When the test results are compared to known case facts and established manipulation models, the analyst is able to form a fact based opinion with regards to the tested file's evidentiary trust-worthiness.

For example, let's say that a case relies upon an audio recording with an incriminating admission, but the accused claims that the recording was edited after the fact. Only the interviewer and accused were present at the time of the recording so, rather than hoping for a favorable courtroom battle between testimony credibility, science can search for provable signs of content manipulation.

The Locard's exchange principle states that, "It is impossible for a criminal to act, especially considering the intensity of a crime, without leaving traces of this presence." Since the behavioral characteristics of microphones are well established, any tampering events such as stop-start, editing, resaving, speed changes, and muting should leave behind some measurable trace evidence.

For example, an audio recording may have captured a rhythmic sub-audible sound (e.g. a seemingly silent 60Hz electrical harmonic) which can be isolated and amplified. Any disruption in this otherwise predictable sound pattern would serve as compelling evidence of editing, and such a finding would negatively impact the credibility of all evidence originating through the same chain of custody. Favorable dispositions are almost always ensured when damaging evidence can be excluded and/or the opposing side discredited.

In one of my early cases, I examined a telephone recording that sounded natural. Upon a spectral view (frequency over time), I observed a shift in the background noise during the incriminating portion of the recording. Upon deeper examination, it became obvious that this portion of the recording had originated from a different audio file. At that moment, I felt like a super-detective and my career choice was forever solidified.

Metadata within, or about, a tested file can aid in determining a file's origin and chain-of-custody. I remember a case where a plaintiff and eyewitness accused the defendant of initiating an altercation. The eyewitness even produced a corroborating video the following day. The mobile phone video appeared authentic, and the case was presumed obvious and self-evident.

Upon forensic inspection, I found that the recording's GPS metadata deviated from the incident address defined in the police report.

Opposing counsel responded with an article explaining how a GPS discrepancy can result from a lag in the phone's ability to receive updated GPS satellite signals. However, suspicion grew once I proved that the video's GPS coordinates matched the home address of the plaintiff, who happened to be a professional video editor.

Authentication testing has the capacity to identify the originating recorder's make and model, and can sometimes identify the specific unit that made the recording. This is accomplished through the analysis of compression tables, file metadata, comparison to recordings found in social media posts, and characteristics unique to the specific recording device. Such tests can be performed years after the fact, even when the original recording device has been lost or destroyed.

For example, photo response non-uniformity (PRNU) refers to the natural microscopic defects in the silicon wafer used to create a camera's imager. The relative position of those defects creates a complex and unique watermark that gets imprinted onto every recording made by that device. Because the defects within the imager can be represented by a relative pattern, PRNU testing is able to survive repeated image processing. This capability makes PRNU testing comparable to the identity analysis applied to ballistic markings and DNA profiling.

In ruling on the 1993 case of Daubert v. Merrell Dow Pharmaceuticals, Inc., 509 U.S. 579, the U.S. Supreme Court set the Daubert standard for evaluating the scientific credibility of an expert's methodology, in that it must: be generally accepted in the scientific community, have undergone peer review or publication, have been (or be able to be) tested, and have a measurable error rate.

PRNU was first accepted into the scientific community during the 2005 SPIE conference. It was subsequently published and peer-reviewed, which validated its error rate parameters. PRNU became codified in a 2011 Alabama case (United States of America v. Nathan Allen Railey) where it passed its first Daubert challenge, and was used to prove that the images in this case originated from the suspect's camera.

Some common examples of authentication methods that have been codified include analysis of browser history and cookies to retrace an individual's internet activity, prior cell tower pings to track a user's movement, vehicular infotainment systems to reconstruct how and where someone drives, and temporal Electric

Network Frequency (ENF) discrepancies in our nation's power plants to determine where and when a given audio recording was produced.

Having access to all this information raises an interesting legal question: Is the analyst committing a crime under the 1984 Computer Fraud & Abuse Act 18 USC 1030(a) simply by extracting this data? It is unlikely that the end user explicitly authorized the analyst to access this identifying data, as they may not even have been aware that such data existed. At best, the governing laws are vague and, using the Fifth and Fourteenth Amendments to the United States Constitution as reference, the courts generally find that vague laws violate a citizen's right to due process. While this challenge has yet to be tested in Federal court, it serves as a reminder that science evolves faster than the legal landscape it serves.

Courts change slowly, and judges are reluctant to rule against legal precedent or to hear a challenge, even when those rulings are found to have been based upon faulty science. In response, attorneys are hiring experts to elevate their evidence, even when the integrity of that evidence was not in question. This allows their opening statement to include, "This case relies upon recorded evidence. Opposing counsel would prefer that you trivialize it during your deliberations, so they never had it tested. We had this evidence analyzed by an independent forensic expert, who concluded that the recordings in this case are authentic and trustworthy." This is an effective tactic that has become the fastest growth segment of my business. When authentication testing is neglected, it leaves the evidence in a vulnerable status.

An example of this was a restaurant slip-and-fall case where, in the absence of eyewitness testimony, the case hinged upon video clips exported from different surveillance camera views. Each video spanned a brief time period, and none of the clips captured the actual incident. Although the originating DVR model was known, the site's DVR unit no longer existed, nor did the original recordings that it had contained. At question was why the video clips missed the incident.

Plaintiff's expert dismissed motion detection recording as a possible source, because on-going motion was observed as the video clips ended. The expert then determined that the event was not part of the exported video clips because of either selective video exporting or post-production editing.

As a result, plaintiff asserted that the defendant had intentionally omitted incriminating evidence, engaged in evidentiary spoliation, and perpetrated a fraud upon the court. Plaintiff's expert had decades of experience, and an extensive CV detailing relevant training and prior testimony. On paper, this expert had no equal, and it came as no surprise that the court accepted their qualifications and opinions as fact. The problem is that prior training and testimony on any given subject does not guarantee a correct understanding of that subject matter.

As the rebuttal expert, I used the MAT method to prove that the video clips were trustworthy and had not been manipulated. I then use the visual contents of those videos to work backwards and define both the motion detection zones, along with their relevant pre-record/post-record settings, all of which exactly matched the available setting options defined in the user manual for the originating DVR model. Next I proved that the calculated settings of each video clip validated the calculations from the remaining video clips. I then used the raw data within the opposing expert's report to fully validate my findings, and thus prove that the other expert's opinions were based upon unsubstantiated assumptions. The case settled favorably after my deposition.

Within the field of multimedia, remaining current across ever evolving technology is critical to an analyst's effectiveness. Until a test becomes Daubert compliant, the analyst must either apply some unverified assumption or apply the unverified test, the latter of which is generally favored since it is easier to defend in the courtroom.

The future is unlikely to resolve these challenges as audio and video recording equipment becomes more complex. Although numerous public and private organizations see the need for industry oversight, this is unlikely to occur due to their competing interests and those of the software vendors who want to protect their proprietary solutions.

Within my profession, the real gatekeepers are the forensic groups and the accredited certifications, as they require continued education and testing. If an expert has not been certified, or re-certified, within the last few years, their knowledge may be seriously outdated. The same warning applies to experts where their only recent qualifications are public speaking and lecture/workshop attendance, where they may have been texting, confused, or otherwise disconnected from the educational content claimed on their CV. Right or wrong, it is often up to the attorneys to evaluate the case experts, and the opinions that they present.

About the Author

Doug Carner is a certified Audio Video Forensic Analyst, Certified Computer Forensics Examiner, Certified Protection Professional, Certified Forensic Hitech Investigator, chairs the Forensic Working Group, and serves on the Los Angeles Superior Court's Panel of Experts. He is an active member of the American Academy of Forensic Sciences, the Audio Engineering Society, the International Association for Identification, the International Association of Law Enforcement Intelligence Analysts, and the Scientific Association of Forensic Examiners. Mr. Carner is an expert witness, forensic educator, widely published, praised by both plaintiff and defense, leads a forensic collaboration forum, has processed evidence in thousands of cases, invented industry methods and filters, created law enforcement's most relied upon enhancement and authentication software, exposed junk science, and routinely donates his time to cold cases and innocence projects. To learn more, ask your smart phone, "Who is Doug Carner?"

Internet of Things: Amazon Echo Forensics

by Rachael Medhurst

Internet of Things (IoT), sometimes referred to as the Internet of Everything, is a network of IP (Internet Protocol) enabled devices; these devices connect to one another and the internet with the aim to bring the users convenience in their everyday lives. Examples of internet enabled devices, otherwise known as 'things', range from self-driving cars, smart watches, heart monitors, smart microwaves, smart locks, smart light bulbs, Amazon Echo and many more.

Internet of Things (IoT), sometimes referred to as the Internet of Everything, is a network of IP (Internet Protocol) enabled devices; these devices connect to one another and the internet with the aim to bring the users convenience in their everyday lives. Examples of internet enabled devices, otherwise known as 'things', range from self-driving cars, smart watches, heart monitors, smart microwaves, smart locks, smart light bulbs, Amazon Echo and many more.

In order for these devices to communicate effectively between each other and for better convenience, each device has a sensor embedded.

There are many examples of how IoT devices can bring convenience to people's lives; one example would be the use of an IoT enabled alarm clock, this device is constantly updating information about the weather and traffic all while you're sleeping. This can provide the user with convenient information.

For example, when there has been a car accident on your normal route to work, the IoT enabled device will update this information and reset your alarm time from 7am to 6:30am to account for the extra travel time required to arrive at work on time.

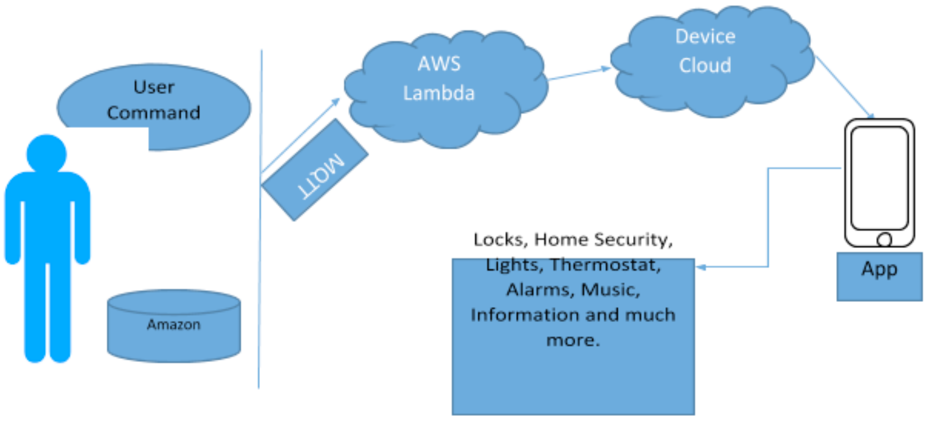
Another example would be the smart watch monitoring your heart rate; this could provide you with information about your health that you may not have known otherwise. An example of this is when an Apple Watch notified James Green that his heart rate was higher than normal. James then sought medical attention to discover he had a blood clot in his lungs; this information potentially saved his life.

The aim of the internet-enabled devices is to help people; these are just two examples of the convenience and usefulness of IoT devices from everyday convenience to life saving information. There is an estimation that there will be 200 billion connected devices by 2020. However, with so many devices available to users and with the change in data storage, are Digital Forensic Investigators equipped to deal with these devices?

In this article, the focus will be on the virtual assistant Amazon Echo. Amazon is currently selling a range of virtual assistants. These include:

- Echo Dot
- Echo
- Echo Plus
- Echo Spot
- Echo Show

The focus will be on the Echo device with the built in Echo voice recognition; this device is a virtual assistant that is a ‘cloud’ based service with many capabilities to help the user. These capabilities include voice interaction, music playback, setting alarms, purchasing items from Amazon, general enquiries and streaming information: this can include news, sports, weather, traffic updates and many more. The diagram below shows how Amazon Echo works:


User Command
Amazon
MQTT
AWS Lambda
Device Cloud
App
Locks, Home Security, Lights, Thermostat, Alarms, Music, Information and much more.

There are a number of steps taken for the Amazon Echo to work effectively for the user, as shown in the diagram above. Users of an Echo device usually have some knowledge of how it works, i.e. the user knows to give a command to the device, using the application on a phone or tablet and that the device will follow that command. However, detailed aspects of this diagram may be unfamiliar to users such as MQTT and AWS Lambda.

MQTT (Message Queuing Telemetry Transport) is a machine to machine or 'Internet of Things' connectivity protocol on top of TCP/IP. This is a publish/subscribe messaging transport.

AWS Lambda is a platform created by Amazon as part of the Amazon Web Services; this enables users to run code without provisioning for applications. AWS provides a platform that scales automatically and bills users according to resource demand.

As Echo is a cloud-based service, the requests and responses are stored onto the Amazon servers. The confirmation of this has been located via the Amazon Echo policy guide; this is where Amazon has stated:

'When you speak to Echo, a recording of what you asked is sent to Amazon's servers so our systems for speech recognition and natural language understand what the user is asking and so can respond to your request'.

Another question often raised to Amazon from their customers is whether Amazon is recording all of their conversations. Amazon has responded to this with the following statement:

'No, devices are designed to detect only your wake word EG: Echo. No audio is stored or sent to the Cloud unless the device detects the wake word'.

The user activates the devices using a wake word; this can be 'Echo', 'Amazon' and 'Alexa'. As the device needs to listen for these wake words, the device records 60 seconds of audio and stores it in the memory. However, this is only the most recent 60-second audio clip. Each 60 seconds, a new audio clip is created and the previous audio clip is deleted. This information is stored locally and not sent to the cloud.

Data on an Echo device ranges from material held locally, the use of an Echo application on the user's mobile/tablet and data that is stored externally on cloud servers. With the knowledge of how data is stored when using the Echo device, the Digital Forensic Investigator will have to complete different forensic techniques to uncover any potential evidence.

With data held locally, if you looked inside the Echo, one of the components is a 4GB eMMC flash storage memory chip. eMMC (embedded Multi-Media Controller) is a storage chip that is built onto the motherboard of each device. A Digital Forensic Investigator would need to retrieve the data stored on this chip ready for analysis.

The investigator would first physically remove the chip from the Echo device and then use specialized equipment/software to extract the data for analysis.

As mentioned previously, Echo is a cloud-based service; the 'cloud' refers to services where the files and data are stored on servers connected to the internet. This ultimately means the user does not store this data locally on their device but on an Amazon server.

There are many benefits of cloud computing, but this can bring a range of challenges for Digital Forensic Investigators. With cloud computing becoming common practice, what does this mean for Digital Forensic Investigators if the majority of data is stored online and no longer locally?

Due to the popularity of Cloud Computing, this has led to an increase and development of what is now known as 'Cloud Forensics'. This means that a Digital Forensic Investigator can no longer retrieve all of the information simply by gaining access to the device. In order to retrieve all of the information, further steps are required, which present their own challenges.

The Investigator also needs to complete Network Forensics. Network Forensics works by analysing the computer network traffic for the purposes of monitoring data, information gathering, and evidence collection and intrusion detection. Due to the Cloud being a network, this method is very important in the evidence collection stage, especially as this is volatile data and it can be lost after transmission.

With an Echo, the owner has to register an Amazon account or log in to an existing one, in order for the device to work effectively. If a Digital Forensic Investigator is unable to access the user's account, the appropriate authorities can enforce the use of RIPA (Regulations Investigatory Powers Act) 2000 to ascertain the user's Amazon account credentials for further detailed investigation.

RIPA (Regulatory Investigatory Powers Act) 2000 is an act of parliament of the UK, which regulates how public bodies conduct investigations, surveillance and the interception of co-mmunications to gather evidence. However, section 49 of RIPA covers protected information, which states:

'A disclosure requirement in respect of any protected information is necessary on grounds falling within this subsection if it is necessary

- *In the interests of national security*
- *For the purpose of preventing or detecting crime; or*
- *In the interest of the economic well-being of the United Kingdom'.*

However, if the suspect does not present the information such as account credentials/ passwords voluntarily, then this can result in up to 2 years imprisonment or 5 years imprisonment for government or indecent image crimes.

With the credentials of the user's account, a Digital Forensic Investigator can gain a large amount of information that can assist them during their investigation from the application.

The Digital Forensic Investigator will do this by having an active internet connection and accessing the user's account. Accessing a number of URLs will provide the investigator with a large amount of information. The URLs used to gather information are shown in the table below:

URL	Information that can be accessed
https://pitangui.amazon.com/api/devices/devices	The investigator could gain information relating to all the devices connected to the Amazon account, which can also provide further information about the device, such as: - Model type - Serial Number - Customer ID - Software Version - Device Account ID
https://pitangui.amazon.com/api/cards?	The investigator could gain information relating to all the credit/ debit cards associated with the account.
https://pitangui.amazon.com/api/wifi/configs?	This URL lists all the wireless network information that the user chooses to save to the Amazon cloud (This is set by default). This information can include the SSID (Service Set Identifier) and a plain text password.
https://pitangui.amazon.com/api/phoenix?	This URL lists all the smart home devices connected to that Amazon account within the user's home. This could include bulbs, smart locks, smart TVs, etc.
https://pitangui.amazon.com/api/activities?size=100&offset=-1	This URL will show the investigator the last 50 activities performed by the Echo device.

However, if the Digital Forensic Investigator cannot gain access to the credentials, then a different option would be to use a 'Subpoena'. This can be issued if a party wants a person or organisation to provide him/her with certain documents or physical evidence. With Echo's information being stored in the Cloud, there are times when the investigator, court or the defendant can request the recordings from Amazon's servers for investigation, shown in the case study explained below:

James Bates gave Amazon permission to hand over recordings from his Amazon Echo for evidence after he was charged with the murder of a man, Victor Collins, found dead in his hot tub in 2015. During his trial, prosecutors tried to use Bates' smart home against him with information from the smart water meter to argue that a lot of water used after the death showed he was potentially hiding evidence, such as blood.

Another example of the use of Amazon Echo devices in law enforcement was documented in November 2018. A judge in New Hampshire requested an investigation of an Amazon Echo device, believing it could hold the key to solving a double murder. Christine Sullivan and Jenna Pellegrini were both killed in January 2017. The housemate (Sullivan's boyfriend) Timothy Verrill, was charged with first-degree murder. After Verrill pleaded not guilty, the judge ruled that Amazon had to hand over recordings taken by the device at the property as evidence because the device records snippets of information before the wake word activates the device.

In some instances of the recordings required from Amazon, they have said:

'*They won't release the information without a valid and binding legal demand properly served on us'*.

IoT enabled devices continue to increase, this will drastically change how Digital Forensic Investigators complete investigations. With IoT devices collecting information, this could help investigators collecting evidence in a criminal investigation; this will result in Digital Forensic Investigators requiring a wide range of skills to deal with such a vast amount of equipment such as smart watches, drones, water meters, fridges, smart locks, healthcare equipment and many more. With the challenges of investigating all these devices, the Digital Forensic industry will continue to grow and provide challenges to a Digital Forensic Investigator.

This article has focused on the IoT device Amazon Echo, with the forensic methods mentioned and how much data is gathered from this device/application. From the two examples shown where the recordings are being used in courtrooms, there is currently high interest in what these devices can bring to a criminal case and how they can possibly convict the suspects if the evidence is located. There is definitely no doubt that Digital Forensics is about to change drastically with such a development in internet related equipment. If the industry and law enforcement are ready for this, is another question.

References:

- https://lcdiblog.champlain.edu/2016/02/08/amazon-echo-forensics-project-blog-1/
- https://www.dfrws.org/conferences/dfrws-usa-2017/sessions/digital-forensic-approaches-amazon-alexa-ecosystem

About the Author

Rachael Medhurst is a graduate of the University of South Wales where she gained her Digital Forensic qualifications at both Bachelor's and Master's level. After graduating, Rachael became a Digital Forensic Investigator for a private firm that offered their assistance to a variety of forces throughout the country, while here she completed hundreds of cases and attended court as an Expert Witness. In the summer of 2018, Rachael decided to fulfill a role as a Digital Forensics and Cyber Security lecturer within the University of South Wales for their initiative BSc Applied Cyber Security program at the 'National Cyber Security Academy.

Windows Live Forensic

by Nikhil Singhvi S

This article will put some insight on basic commands and tools that can be used while performing Live Forensic. Data on a system has an order of volatility. Data from the Memory, Swap Space, Network Process and Running System Processes are the most volatile data and will be lost if a system reboots or powers down. The Internet Engineering Task Force (IETF) released a document titled, ***Guidelines for Evidence Collection and Archiving***. It is also known as **RFC 3227**. This document explains that the collection of evidence should start with the most volatile item and end with the least volatile item.

Introduction:

What is Live Forensic/Live Response?

A methodology that advocates extracting "live" system data before pulling the cord to preserve memory, process, and network information that would be lost with a traditional forensic approach.

Live forensic recognizes the value of the volatile data that may be lost by a power down and seeks to collect it from a running system. While performing a forensic investigation, it is important that we preserve the integrity of evidence and try collecting as much evidence as possible. Volatile data or RAM memory are very crucial evidence and can help us in many ways during our investigation.

Further, this article will put some insight on basic commands and tools that can be used while performing Live Forensic.

Data on a system has an order of volatility. Data from the Memory, Swap Space, Network Process and Running System Processes are the most volatile data and will be lost if a system reboots or powers down. The Internet Engineering Task Force (IETF) released a document titled, ***Guidelines for Evidence Collection and Archiving***. It is also known as **RFC 3227**. This document explains that the collection of evidence should start with the most volatile item and end with the least volatile item. So, according to the IETF, the Order of Volatility is as follows:

Order of Volatility

1. Registers, Cache:

Data in CPU and registers are extremely volatile, since they are changing all of the time. Nanoseconds make the difference here. It can be flushed easily by performing a simple action by the computer. An examiner needs to get to the cache and register immediately and extract that evidence before it is lost.

2. Routing Table, ARP Cache, Process Table, Kernel Statistics, Memory:

The routing table and the process table have data located on network devices. In other words, that data can change quickly while the system is in operation, so evidence must be gathered quickly. Kernel statistics are highly volatile because it moves back and forth between cache and main memory. The RAM stays longer than other memory. However, it can be lost if there is a power spike or if power goes out. So, the information should be obtained quickly.

3. Temporary File Systems:

Even though the contents of temporary file systems have the potential to become an important part of legal proceedings, the volatility concern is not as high here. Temporary file systems usually stick around for a while.

4. Disk:

We think that the data we place on disk will be around forever, but that is not always the case. However, likelihood that data on a disk cannot be extracted is very low.

5. Remote Logging and Monitoring Data

The potential for remote logging and monitoring data to change is much higher than data on a hard drive, but the information is not as vital. So, even though the volatility of the data is higher, we still want that hard drive data first. Further, to know the user logged in on to the system both physically and remotely we can use **"PsLogged On"** command. We will discuss this in the later part of this article.

6. Physical Configuration, Network Topology, and Archival Media:

These are the items that are either not that vital in terms of the data or are not at all volatile. The physical configuration and network topology is information that could help an investigation, but is likely not going to have a tremendous impact. Finally, archived data is usually going to be located on a DVD or tape, so it isn't going anywhere anytime soon. It is great digital evidence to gather, but it is not volatile.

Basic Commands:

There are two main ways that we can transmit the data to the forensic workstation. The first way is to use "important command" of network administrators called netcat. netcat simply creates TCP channels. netcat can be executed in a listening mode, like a telnet server or in a connection mode, like the telnet client. You can start a netcat server on your forensic workstation with the following command:

```
nc -v -l -p 2222 > Name.txt
```

The -v switch places netcat in verbose mode. The -l switch places netcat in listening mode (like a telnet server). The -p switch tells netcat on which TCP port to listen for data. By using this command, any data sent to TCP port 2,222 on your forensic workstation will be saved to Name.txt. On the victim computer, you will want to run a command to collect live response data. The output of the command is sent over our TCP channel on port 2,222 and saved on the forensic workstation instead of the victim's hard drive.

The data can be sent from the victim computer with the following command:

```
Name | nc
forensic_workstation_ip_address 2222
```

You will have to rename the italicized keywords, such as Name, with the command you run to collect the live response data. Moreover, you will want to substitute the IP address of your forensic workstation where it says *forensic_ workstation_ip_address.* After these commands have completed, you will press CTRL-C (^C) to break the netcat session, and the resulting file Name.txt will contain all of the data from the command we executed. A simple MD5 checksum of Name.txt should be calculated so that you may prove its authenticity at a later date.

Analysing Volatile Data:

The volatile data of a victim computer usually contains significant information that helps us determine the “who”, “how” and possibly “why” of the incident. To answer these questions, you will have to collect data from the following areas on the victim machine:

System Date and Time:

This is the easiest information to collect and understand, but it is one of the most important pieces of information to the investigator and is easily missed. Without the current time and date, it would be difficult to correlate the information between victim machines if multiple machines were affected.

The time and date are simply collected by issuing the “time” and “date” commands at the prompt.

The current time is: 10:24:26.78

The current date is: 04-11-2018

Current network connection:

It is possible that while we are performing our live response process, the attacker is still connected to the server. It could also be possible that the attacker is running a brute force mechanism against other machines on the Internet from this server.

You can view a machine’s network connection by using the netstat command. You need to specify the –an flags with netstat to retrieve all of the network connections and see the raw IP Address.

When you run the netstat command on the victim’s machine, you will receive the following type of information:

Active Connections:

```
Proto   Local Address                  Foreign
Address        State

TCP     0.0.0.0:80                   0.0.0.0:0
LISTENING

TCP     0.0.0.0:135                  0.0.0.0:0
LISTENING

TCP     0.0.0.0:445                  0.0.0.0:0
LISTENING

TCP     0.0.0.0:623                  0.0.0.0:0
LISTENING

TCP     0.0.0.0:5985                 0.0.0.0:0
LISTENING

TCP     0.0.0.0:12373                0.0.0.0:0
LISTENING

TCP     0.0.0.0:16992                0.0.0.0:0
LISTENING

TCP     0.0.0.0:47001                0.0.0.0:0
LISTENING

TCP     0.0.0.0:49664                0.0.0.0:0
LISTENING
```

```
TCP     0.0.0.0:49665               0.0.0.0:0
LISTENING

TCP     0.0.0.0:49666               0.0.0.0:0
LISTENING

TCP     0.0.0.0:49667               0.0.0.0:0
LISTENING

TCP     0.0.0.0:49668               0.0.0.0:0
LISTENING

TCP     0.0.0.0:49669               0.0.0.0:0
LISTENING

TCP     0.0.0.0:49670               0.0.0.0:0
LISTENING

TCP     0.0.0.0:49671               0.0.0.0:0
LISTENING

TCP     0.0.0.0:49675               0.0.0.0:0
LISTENING

TCP     0.0.0.0:49676               0.0.0.0:0
LISTENING

TCP     0.0.0.0:58001               0.0.0.0:0
LISTENING

TCP     0.0.0.0:58002               0.0.0.0:0
LISTENING

TCP     0.0.0.0:65200               0.0.0.0:0
LISTENING

TCP     127.0.0.1:53                0.0.0.0:0
LISTENING

TCP     127.0.0.1:8898              0.0.0.0:0
LISTENING

TCP     127.0.0.1:8899              0.0.0.0:0
LISTENING

TCP     127.0.0.1:8998              0.0.0.0:0
LISTENING

TCP     127.0.0.1:8999              0.0.0.0:0
LISTENING

TCP     127.0.0.1:10001             0.0.0.0:0
LISTENING

TCP             127.0.0.1:49794
127.0.0.1:49795        ESTABLISHED

TCP             127.0.0.1:49795
127.0.0.1:49794        ESTABLISHED

TCP             127.0.0.1:49796
127.0.0.1:10001        ESTABLISHED

TCP     127.0.0.1:49942             0.0.0.0:0
LISTENING

TCP     127.0.0.1:50039             0.0.0.0:0
LISTENING

TCP             127.0.0.1:50397
127.0.0.1:50398        ESTABLISHED

TCP             127.0.0.1:50398
127.0.0.1:50397        ESTABLISHED

TCP             127.0.0.1:50399
127.0.0.1:50400        ESTABLISHED

TCP             127.0.0.1:50400
127.0.0.1:50399        ESTABLISHED

TCP      ip_address:139             0.0.0.0:0
LISTENING

TCP             ip_address:50251
52.138.169.124:443     ESTABLISHED

TCP             ip_address:50713
151.101.0.175:443      ESTABLISHED

TCP             ip_address:50251
52.138.169.124:443     ESTABLISHED

TCP             ip_address:50715
151.101.1.181:443      ESTABLISHED
```

```
TCP    ip_address:50719    23.0.141.68:443
ESTABLISHED

TCP    ip_address:50721
151.101.0.175:443      ESTABLISHED

TCP    ip_address:50913
103.229.206.39:443     ESTABLISHED

TCP    ip_address:50915
104.121.240.198:21    ESTABLISHED

TCP     [::]:80                      [::]:0
LISTENING

TCP     [::]:135                     [::]:0
LISTENING

TCP     [::]:445                     [::]:0
LISTENING

TCP     [::]:623                     [::]:0
LISTENING

TCP     [::1]:49940                  [::]:0
LISTENING

TCP    [::1]:55364             [::1]:55366
ESTABLISHED

TCP    [::1]:55366             [::1]:55364
ESTABLISHED

UDP    0.0.0.0:123           *:*

UDP    0.0.0.0:500           *:*

UDP    0.0.0.0:3544          *:*

UDP    0.0.0.0:3702          *:*

UDP    0.0.0.0:3702          *:*

UDP    0.0.0.0:3702          *:*

UDP    0.0.0.0:3702          *:*

UDP    0.0.0.0:4500          *:*
```

```
UDP    0.0.0.0:5000           *:*

UDP    0.0.0.0:5050           *:*

UDP    0.0.0.0:5353           *:*

UDP    0.0.0.0:5353           *:*

UDP    0.0.0.0:5353           *:*

UDP    0.0.0.0:5355           *:*

UDP    0.0.0.0:11546          *:*

UDP    0.0.0.0:12373          *:*

UDP    0.0.0.0:39999          *:*

UDP    0.0.0.0:58468          *:*

UDP    0.0.0.0:58469          *:*

UDP    0.0.0.0:59944          *:*

UDP    0.0.0.0:63703          *:*

UDP    127.0.0.1:53           *:*

UDP    [::]:123               *:*

UDP    [::]:500               *:*

UDP    [::1]:54949            *:*

UDP       [fe80::243b:edc:3f57:ff97%17]:546
*:*

UDP      [fe80::6d82:d701:3f19:54d3%9]:1900
*:*

UDP     [fe80::6d82:d701:3f19:54d3%9]:54948
*:*
```

The bolded lines represent the active network connections. The additional lines are open ports. Because you know that your forensic workstation is at the IP address **ip_address**, you can ignore corresponding connections. After removing all of the other extraneous data, we are left with the following interesting lines:

```
TCP          ip_address:50251          52.138.169.124:443
ESTABLISHED

TCP          ip_address:50251          52.138.169.124:443
ESTABLISHED

TCP           ip_address:50713           151.101.0.175:443
ESTABLISHED

TCP           ip_address:50715           151.101.1.181:443
ESTABLISHED

TCP            ip_address:50719            23.0.141.68:443
ESTABLISHED

TCP           ip_address:50721           151.101.0.175:443
ESTABLISHED

TCP          ip_address:50913          103.229.206.39:443
ESTABLISHED

TCP          ip_address:50915          104.121.240.198:21
ESTABLISHED
```

The first line is of victim's workstation connecting to port 443, the HTTPS, on system **ip_address.** The last line is connecting to port 21, the FTP port, on system **ip_address.** So using this method you can identify the connection established from the victim's workstation.

Open TCP or UDP ports:

If we look at the lengthy netcat listing shown earlier, all of the lines that are not bolded are open ports. One reason on how they can be helpful to us is that an open rogue port usually denotes a backdoor running on the victim machine. We realize that Windows opens a lot of legitimate ports during the course of doing its business, but you can weed many of them out quickly.

You can further examine the strange ports that are open on the machine using freely distributed tools like the following:

1. FPort
2. IP finger prints and so on

Now, I will be demonstrating how PsTools suite can be helpful during your forensic investigation. This tool can easily be executed using command prompt.

1. User Currently Logged:

During your live response, you could run PsLoggedOn, which is a tool distributed within the PsTools suite. This tool will return the users that are currently logged onto the system or accessing the resource shares. When you execute this tool on a victim's workstation without command-line parameters, you receive the following information:

```
PsLoggedon v1.35 - See who's logged on

Copyright (C) 2000-2016 Mark Russinovich

Sysinternals - www.sysinternals.com

Users logged on locally:

03-11-2018 16:42:02 Victim\Administrator

No one is logged on via resource shares.
```

2. List of files opened remotely:

You could run PsFile. This command shows a list of files on a system that are opened remotely, and it also allows you to close opened files either by name or by a file identifier.

3. GetSid:

You could run PsGetSid. This command allows you to translate SIDs to their display name. It works on built-in accounts, domain accounts, and local accounts.

```
PsGetSid v1.45 - Translates SIDs to names and
vice versa

Copyright (C) 1999-2016 Mark Russinovich

Sysinternals - www.sysinternals.com

SID for \\<account name>:

<SID name>
```

4. PsInfo:

PsInfo shows information for the local system. Specify a remote computer name to obtain information from the remote system. *PsInfo* relies on remote Registry access to obtain its data, the remote system must be running the Remote Registry service and the account from which you run *PsInfo* must have access to the HKLM\System portion of the remote Registry. You will get the following output.

```
PsInfo v1.78 - Local and remote system
information viewer

Copyright (C) 2001-2016 Mark Russinovich

Sysinternals - www.sysinternals.com

System information for \\<account name>:

Uptime: 0 days 19 hours 20 minutes 40 seconds

Kernel version: Windows 10 Enterprise,
Multiprocessor Free

Product type: Professional

Product version: 6.3

Service pack: 0

Kernel build number: 15063

Registered organization:

Registered owner:
```

IE version: 9.0000

System root: C:\Windows

Processors: 4

Processor speed: 2.7 GHz

Processor type: Intel(R) Core(TM) i5-7300U CPU @

Physical memory: 2 MB

Video driver: Intel(R) HD Graphics 620

5. PsPing:

This command implements Ping functionality, TCP ping, latency and bandwidth measurement.

6. PsKill:

Running PsKill with a process ID directs it to kill the process of that ID on the local computer. If you specify a process name, *PsKill* will kill all processes that have that name.

7. PsList:

This command lists the detailed information about processes. You will see the following output.

```
PsList v1.4 - Process information lister

Copyright (C) 2000-2016 Mark Russinovich

Sysinternals - www.sysinternals.com

Process information for <Account name>:

Name     Pid Pri Thd  Hnd   Priv   CPU Time     Elapsed Time

Idle       0   0   4    0     52  11:44:10.125  19:28:48.418

System     4   8 191 4428    168   0:08:00.890  19:28:48.418

smss     464  11   2   52    444   0:00:00.156  19:28:48.374

csrss    672  13  10  935   1940   0:00:06.484  19:28:40.405

wininit  772  13   1  140   1308   0:00:00.046  19:28:40.060

csrss    784  13  13  791   2484   0:00:32.015  19:28:40.055

services 860   9   9  719   5912   0:02:33.968  19:28:40.001

lsass    868   9  11 1723   9064   0:00:39.281  19:28:39.978

svchost  984   8   2   89   1588   0:00:00.046  19:28:39.767
```

8. PsLoglist:

This command shows the contents of the System Event Log on the local computer. Command line options let you view logs on different computers, use a different account to view a log, or to have the output formatted in a string-search friendly way.

9. PsPassword:

You can use *PsPasswd* to change the password of a local or domain account on the local or a remote computer.

```
pspasswd <Domain\Account> [NewPassword]
```

10. PsServices:

This command is used to display the configured services both running and stopped on the local system. You will get the following output.

```
SERVICE_NAME: RetailDemo

DISPLAY_NAME: Retail Demo Service

The Retail Demo service controls device activity while the
device is in retail demo mode.

TYPE: 20 WIN32_SHARE_PROCESS

STATE: 1  STOPPED

(NOT_STOPPABLE,NOT_PAUSABLE,IGNORES_SHUTDO
WN)

WIN32_EXIT_CODE: 1077 (0x435)

SERVICE_EXIT_CODE: 0  (0x0)

CHECKPOINT: 0x0

WAIT_HINT: 0 ms
```

Conclusion:

There are many more ways/tools to extract the data when performing live forensic. But while performing a forensic acquisition on a Live system, it is important that you don't lose the integrity of data. Similarly, also try acquiring as much information as possible. Internet-based application servers will be harder for forensic examiners to physically collect. Additionally, Internet-based applications may generate diskless workstations, leaving the only evidence in physical memory. Finally, software vendors are starting to deploy a larger amount of software that securely deletes data because of identity-theft concerns. In the near future, traditional forensics will become more impractical, and live investigations will become a necessity. Traditional methodologies are becoming somewhat obsolete. We will need to adopt a new way of conducting these types of investigations. While we have to be careful while touching the computer in order to prevent any changes, it is now obvious that there are times when an examiner must interact with a live computer in order to retrieve vital data. We should also be able to provide a reasonable reason in a court of law to the judge/jury on why we performed live forensic and should also present the chain of custody.

Reference:

1. http://scitechconnect.elsevier.com/wp-content/uploads/2013/09/Incident-Response-Live-Forensics-and-Investigations.pdf

2. https://www.digitalforensics.com/blog/2267-2/

3. https://www.ietf.org/rfc/rfc3227.txt

4. https://blogs.getcertifiedgetahead.com/cfr-and-order-of-volatility/

5. https://pdfs.semanticscholar.org/f140/f3bdb657f4dcfbfd4bf0183524bfa925a872.pdf

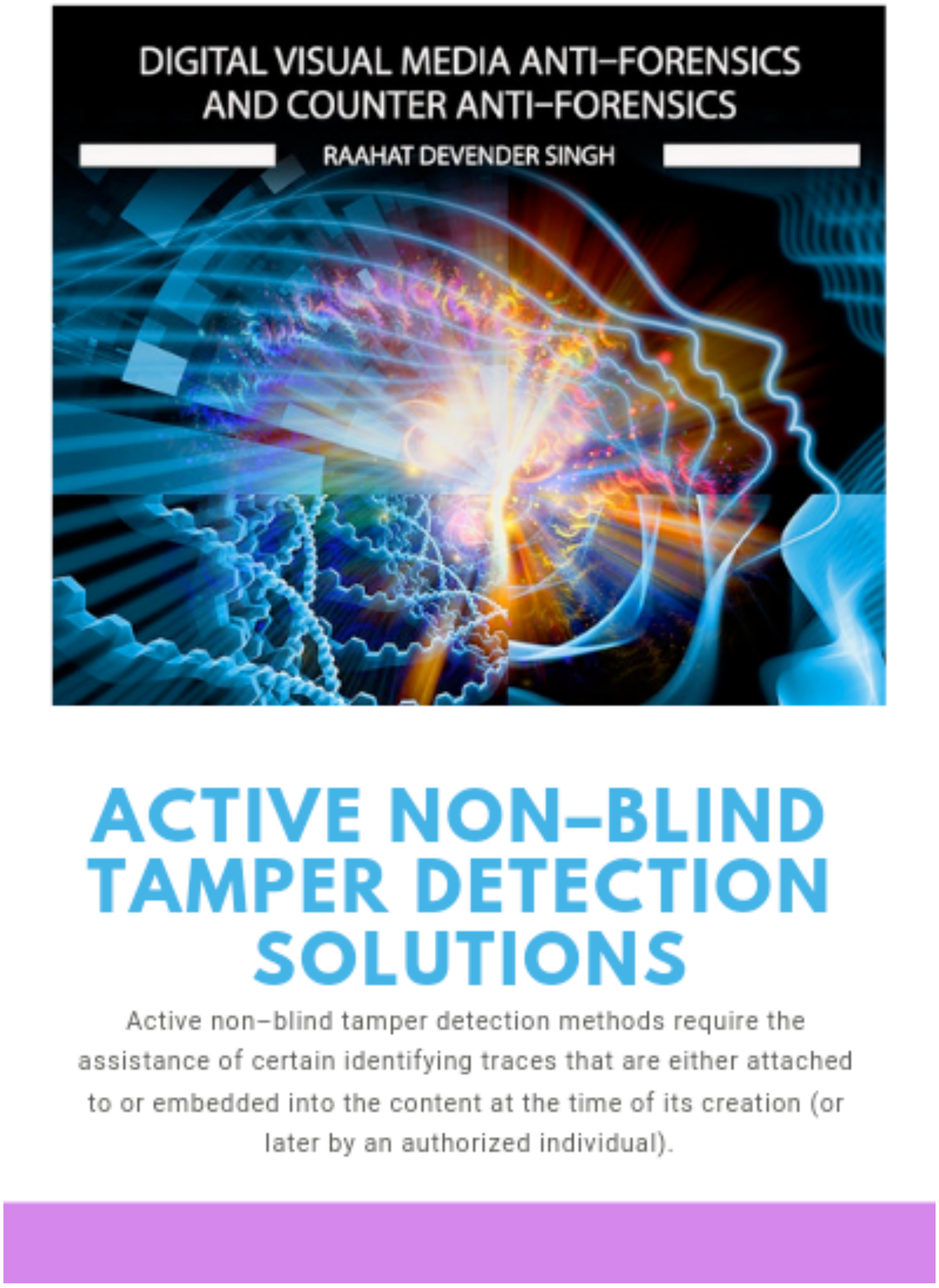
6. https://www.computer-forensics-recruiter.com/order-of-volatility/

7. https://www.ietf.org/rfc/rfc3227.txt

About the Author

Nikhil Singhvi S is currently working as an Information Security Analyst with Risk Advisory Services Practice of Ernst & Young (India). He is a BCA graduate from Loyola College, which is one of the reputed colleges in India and also currently pursuing his MSc. Cyber Forensic and Information Security from Madras University and Diploma in Cyber Law from Government Law college, Mumbai. He holds several reputed certifications like CCNA, CEH, CND, CIPR and he is also a certified Web Security Analyst. He has 1.5 years of experience in implementation relating to Information Security Management System (ISO 27001:2013). He is also an active associate member of National Cyber Safety and Security Standards and Cyber Management Alliance.

From eForensics course platform...



DIGITAL VISUAL MEDIA ANTI-FORENSICS AND COUNTER ANTI-FORENSICS

RAAHAT DEVENDER SINGH

ACTIVE NON-BLIND TAMPER DETECTION SOLUTIONS

Active non-blind tamper detection methods require the assistance of certain identifying traces that are either attached to or embedded into the content at the time of its creation (or later by an authorized individual).

ActiveNon–Blind Tamper Detection Solutions

Active non–blind tamper detection methods (Figure D1) require the assistance of certain identifying traces that are either attached to or embedded into the content at the time of its creation (or later by an authorized individual).

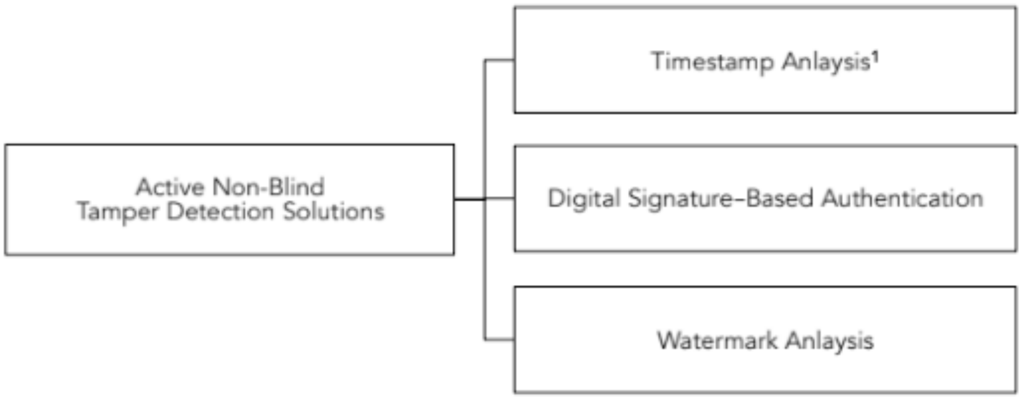

Active Non-Blind
Tamper Detection Solutions
Timestamp Anlaysis¹
Digital Signature–Based Authentication
Watermark Anlaysis

Figure D1 Active non-blind tamper detection solutions.

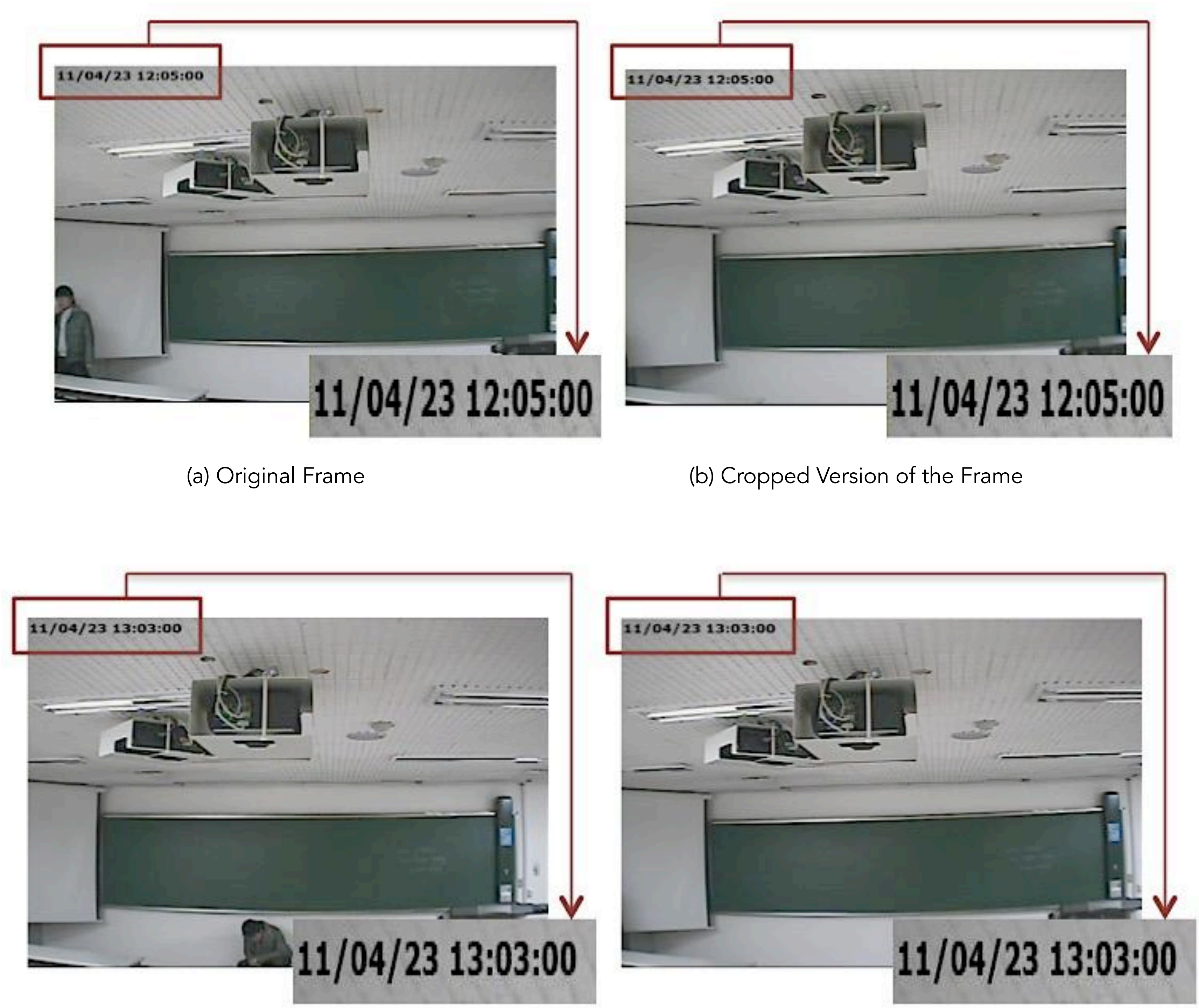
Please note that the methods illustrated in Figure D1 are sometimes also referred to as *active non–blind integrity verification measures* or *active non–blind content authentication methods*.

1. Overlaid Timestamp Analysis

Overlaid timestamps are amongst the most primary defenses against content manipulation. An overlaid timestamp is a sequence of characters embedded onto the video frames that identify when a certain event occurred (and was subsequently captured by the camera), usually giving date and time of day, sometimes accurate to a small fraction of a second.

1.1 Limitations of Timestamp–Based Tamper Detection

Timestamp–based tamper detection is a very primitive solution to a challenge as elaborate and consequential as establishing the authenticity of digital videos, and while it may be considered to be a precursory method of authentication, its results must not be accepted as an absolute proof of content reliability, primarily because not only are timestamps prone to inaccuracies (which can occur if the internal clock of the camera is inaccurate, or because of failure to pay attention to daylight saving in countries where it is applicable), but they are susceptible to tampering as well (timestamps can be superimposed from one frame onto another). Consider the illustrative examples of timestamp manipulation presented in Figure D2.



(a) Original Frame

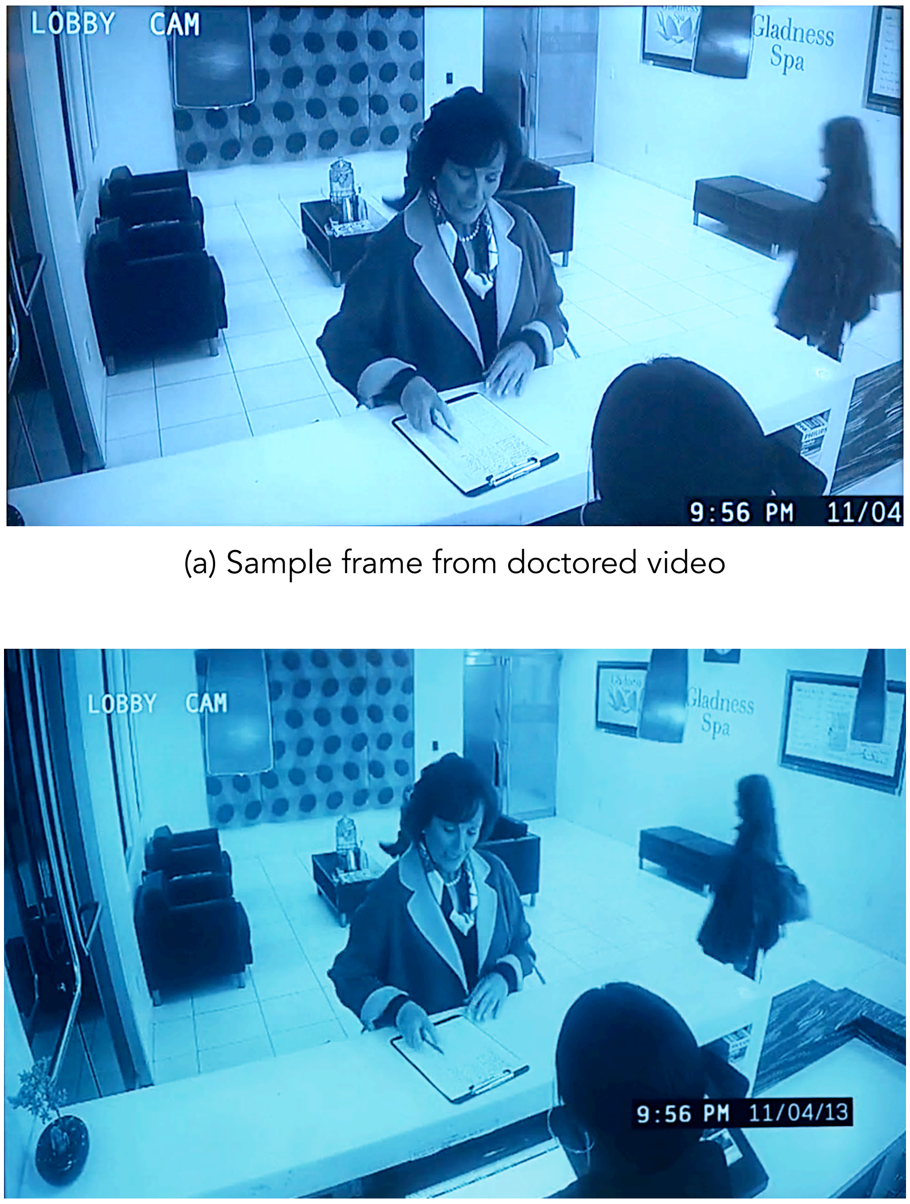
(b) Cropped Version of the Frame

(c) Frame with Original Timestamp

(d) Frame with False Timestamp

Figure D2 Examples illustrating susceptibility of timestamps to conscious manipulation. (a) depicts an original frame with an accurate timestamp whereas (b) depicts a cropped version of this frame with the original timestamp pasted onto the cropped frame. (c) and (d) depict two different frames with identical timestamps. In this case, the frame in (c) is the one with the accurate timestamp whereas the timestamp on the frame in (d) is false. [Images courtesy of Hyun et al. 2013].

An example of timestamp manipulation can be found in Episode 1 of Season 2 of the TV series ‘How To Get Away With Murder’, where the characters manipulated the timestamp on a surveillance video to falsify the timeline of the events depicted in said video.



(a) Sample frame from doctored video

(b) Sample frame from original video

Figure D3 An example of timestamp manipulation. (a) depicts a sample frame from a doctored video and (b) depicts the corresponding frame from the original version of the video. In the doctored video, some portion of the timestamp was eliminated by simply cropping the video’s frames. [Original video courtesy of ABC]

As we can observe from these examples, overlaid timestamps are not impervious to manipulation. Therefore, relying solely on timestamps analysis for the task of content authentication is not prudent.

2. Digital Signature–Based Tamper Detection

A digital signature is an encrypted translation of a succinct representation of the data under consideration. It primarily consists of key information regarding the data and some kind of unique identification of the data producer in the form of a condensed bit–stream. Digital signature based tamper detection schemes operate by examining the pre–embedded signature in the content under consideration; any discrepancy in this signature is considered to be an evidence of tampering.

2.1 Limitations of Digital Signature–Based Tamper Detection

Although the digital signature technology has proved to be quite proficient for the purposes of content confidentiality and integrity verification, it is rendered ineffective in the content authentication domain, primarily because of the following reasons:

Digital signature schemes are computationally intensive and therefore suffer from scalability issues. While within a controlled environment, digital signature–based schemes can be effective for tamper detection, they do not adjust well to the necessities of a large–scale media domain.

The external security token that digital signatures rely on is quite easy to remove, which implies that such a technique will work only if preservation of this token is in the interest of all the parties involved (which is not the case when the possibility of content tampering exists).

Digital signatures are extremely fragile and even innocuous operations such as lossy compression, noise removal, contrast or brightness enhancement, color adjustment, cropping, geometric operations like rotation, and local changes to data (such as blackening out a logo) are capable of destroying or damaging the signatures.

Generation, verification, and deployment of digital signature–based content authentication schemes are not only time–intensive processes but they incur great costs as well. Both these factors impose severe restrictions on the practical feasibility of such schemes.

Digital signatures rely heavily on the technology they are based on; in today's age of transient technology, such techniques end up with a very short shelf–life.

Based on these constraints, we can state that digital signature–based tamper detection methods do not possess the capability to meet the needs of a realistic forensic environment.

3. Watermark Analysis–Based Tamper Detection

The process of watermarking entails embedding content integrity information as well as unique identification of the content producer into the data content itself. Watermark analysis–based tamper detection schemes operate in a manner similar to digital signature–based schemes; any deviation from the norm in the appearance or characteristics of the pre–embedded watermark is considered to be an evidence of tampering.

3.1 Limitations of Watermark Analysis–Based Tamper Detection

Though digital watermarking technology has some utility in the copyright protection domain, its content authentication properties are very limited; it suffers from the following shortcomings:

Watermark analysis–based content authentication works for only those videos that have already been watermarked. This restriction, along with the fact that most digital cameras do not contain a watermark–embedding module, leads to a very limited scope of applicability for such schemes.

Some watermarks have been shown to degrade the visual quality of the digital content.

A damaged watermark may not always be indicative of tampering; attacks such as collusion attack, uncorrelated noise attack, stir mark attack, mosaic attack, jitter attack, inversion attack, and echo–hiding attack, are known to affect the integrity of watermarks. Moreover, completely harmless content modification operations, such as lossy compression, noise removal, contrast or brightness enhancement, color adjustment, cropping, and local changes to data (such as blackening out a logo), also have a negative impact on the integrity of watermarks.

Watermarking techniques do not possess the ability to resist willful attempts to remove the watermark. Generation of a counterfeit watermark is also not a difficult endeavor.

Watermarking–based schemes do not allow for the possibility of independent third–party verification, since the only people capable of detecting watermarks are the content producers or owners themselves. Such lack of independent verification may be unacceptable in certain sensitive situations, for instance, during a criminal trial.

All these shortcomings render watermark analysis–based methods ill–suited for the task of real–world tamper detection.

If you want to find out more about Digital Visual Media Anti-Forensics and Counter Anti-Forensics click here: https://eforensicsmag.com/course/dvm-anti-forensics-counter-anti-forensics-w34/

References and Further Readings:

- P.A. Bernstein and E. Newcomer (2009). *Principles of Transaction Processing.* 2nd Ed., Morgan Kaufmann, pp. 263.
- C.M.B. Medeiros, Ed. (2009). *Advanced Geographic Information Systems*, Vol. 1. Oxford, UK, pp. 59.
- P. Yin P, X.-S. Hua, and H.-J. Zhang (2002). *Automatic time stamp extraction system for home videos*, IEEE International Symposium on Circuits and Systems, Phoenix-Scottsdale, AZ.
- X. Yu, J. Cheng, W. Song, and B. He (2013). *An algorithm for timestamp removal for panorama video surveillance*, 5th International Conference on Internet Multimedia Computing and Service, Huangshan, China, pp. 364–367.
- D.-K. Hyun, S.-J. Ryu, H.-Y. Lee, and H.-K. Lee (2013). *Detection of upscale-crop and partial manipulation in surveillance video based on sensor pattern noise*, *Sensors* Vol. 13, No. 9, pp. 12605–12631.
- X. Yu, W. Song, J. Cheng, B. Qiu, and B. He (2013). *An automatic timestamp replanting algorithm for panorama video surveillance*, Pacific-Rim Symposium on Image and Video Technology, Guanajuato, Mexico, pp. 162–171.
- J. Cheng, X. Yu, S. Liao, and G. Zhao (2015). *Timestamp removal for panorama video surveillance*, 7th International Conference on Internet Multimedia Computing and Service, Hunan, China, Article No. 52.
- X. Yu, J. Cheng, S. Wu, and W. Song (2016). *A framework of timestamp replantation for panorama video surveillance*, Multimedia Tools and Applications, Vol. 75, Issue 17, pp. 10357–10381.
- W. Diffie and M.E. Hellman (1976). *New directions in cryptography*, IEEE Transactions on Information Theory, Vol. 22, No. 6, pp. 644–654.
- S. Shaw (1999). *Overview of watermarks, fingerprints, and digital signatures*, JISC Technology Applications Programme Report, pp. 1–20.

- C.-Y. Lin (2000). *Watermarking and digital signature techniques for multimedia authentication and copyright protection*, PhD Thesis, Graduate School of Arts and Science, Columbia University.

- T. Sencar and S. Memon (2008). *Overview of State-of-the-art in Digital Image Forensics*, in Digital Image Forensics. Algorithms, Architectures and Information Systems Security, B.B Bhattacharya, S. Sur-Kolay, S.C. Nandy, and A. Bagchi, Eds., World Scientific Press, pp. 325–347.

- H. Farid (2009). *Image forgery detection: A survey*, IEEE Signal Processing Magazine, Vol. 16.

- T.-T. Ng, S.-F. Chang, C.-Y., Lin, and Q. Sun (2011). *Passive blind image forensics*, in Multimedia Security Technologies for Digital Rights Management, W. Zeng, H. Yu, and C.-Y. Lin, Eds., Elsevier.

- A. Rocha, W. Scheirer, T. Boult, and S. Goldenstein (2011). *Vision of the unseen: Current trends and challenges in digital image and video forensics*, ACM Computing Surveys, Vol. 43 Issue 4, pp. 26:1–26:42.

- M. Kirchner (2012). *Notes on digital image forensics and counter-forensics*, Part of author's dissertation: Forensic Analysis of Resampled Digital Signals

- O.M. Al-Qershi and B.E. Khoo (2013). P*assive detection of copy-move forgery in digital images: state-of-the-art*, Forensic Science International, Vol. 231, Issues 1–3, pp. 284–295.

- C.-T. Li, Ed. (2013). *Emerging Digital Forensics Applications for Crime Detection*, *Prevention, and Security*, IGI Global.

- K.N. Sowmya and H.R. Chennamma (2016). *Video Authentication using digital signature and watermark: A study*, in Advances in Intelligent Systems and Computing, Vol. 507. S.C. Satapathy, V.K. Prasad, B.P. Rani, S.K. Udgata, and K. S. Raju, Eds., Springer Science and Business Media.

- I.J. Cox, M. Miller, J. Bloom, J. Fridrich, and T. Kalker (2007). Digital Watermarking and Steganography. 2nd Ed., Morgan Kaufmann.

- F.Y. Shih (2017). *Digital Watermarking and Steganography: Fundamentals and Techniques*, CRC Press.

- S. Wolthusen (2001). *On the limitations of digital watermarks: A cautionary note*, World Multiconference on Systems, Cybernetics, and Information, Vol. 4, pp. 489–495.

- S.R.M. Oliveira, M.A. Nascimento, and O.R. Zaïane (2002). *Digital watermarking: status, limitations and prospects*, Technical Report TR02-01, Department of Computer Science, University of Alberta.

- (2004). *Digital Watermarking*, in Lecture Notes in Computer Science, T. Kalker, I. J. Cox, and Y.M. Ro, Eds., Vol. 2939, Springer Science and Business Media.

- C.-T. Li. Ed. (2008). *Multimedia Forensics and Security*, IGI Global.

- X. Zhao, P. Bateman, and A.T.S. Ho (2011). *Image authentication using active watermarking and passive forensics techniques*, Multimedia Analysis, Processing and Communications, pp. 139–183.
- M.A. Nematollahi, C. Vorakulpipat, and H.G. Rosales (2016). *Digital Watermarking: Techniques and Trends,* Springer Topics in Signal Processing, Vol. 11, Springer Science and Business Media.

About the Instructor

Raahat Devender Singh is a PhD research scholar and a guest lecturer working in the Department of Computer Science and Engineering in University Institute of Engineering and Technology, and the Forensics Department in Panjab University, Chandigarh, India. She has been actively working in the Digital Video Forensics domain for over three years, and her fields of specialization include digital signal processing, digital image and video content authentication and forgery detection, and forensic analysis and interpretation of digital visual media evidence. She has participated in a number of national and international conferences, and has written several articles and research papers for magazines and scientific journals of various publishing houses including Springer, World Scientific, and Elsevier.