

Digital Forensics and Cybercrime

Gianvito Caleca

2024

Contents

I Cybercrime: threats, modus operandi, underground economy, financially-motivated malware	3
1 Cybercrime: Threat Landscape	4
1.1 Threat landscape	4
1.1.1 A gartner quadrant of threats	6
1.1.2 Examples of internal treaths	7
1.1.3 Data breaches and targeted attacks	8
1.2 Brief history of malicious software	9
1.3 Financially motivated attackers	10
1.4 Direct monetization: ransomware attacks	11
1.4.1 Brief history of ransomware	11
1.4.2 Ransomware screenshot and social meanings	11
1.4.3 How to get infected	12
1.4.4 Encryption mechanism: how do ransomware work	12
1.5 Indirect monetization: Botnets	13
1.5.1 Rise of the bots	13
1.5.2 Geolocalization of botnets command and control	14
1.5.3 Type of (botnet) malware families	15
1.6 The cybercrime ecosystem	16
1.6.1 Identity sales	17
1.6.2 Drive by download	17
1.6.3 Exploitation kits sales	17
1.6.4 Monetization on the dark web	17
1.6.5 Cybercrime and perception	18
1.6.6 Money mules and money laundering	18
2 Cryptocurrencies: abuses and forensics	19
2.1 Bitcoin and blockchain	19
2.2 Wallet and addresses	19
2.3 The bitcoin transaction life cycle	20
2.4 Mining	20
2.5 Fork events	21
2.6 Bitcoin and black market	21
2.6.1 Pseudo-anonymity	21

2.6.2	Why do people use Bitcoin for ransoms?	22
II	Fraud detection and analysis	23
III	Digital Forensics principles	24
3	Introduction to Digital Forensics	25
3.1	What does forensics mean?	25
3.2	The Daubert standard (USA)	25
3.2.1	The Daubert Standard	25
3.2.2	What is scientific? (Italy)	26
3.2.3	Daubert Test for scientific	26
3.3	Example of forensic engagements	27
3.4	Phases of an investigation (Pollitt)	27
IV	Acquisition, analysis, evaluation and presentation of evidence	28
4	Acquisition	29
4.1	Brittleness of digital evidence	29
4.2	The usage of hashes in digital forensics	30
4.3	Hardware and software for acquisition	30
4.4	Bitstream images	31
4.5	Basic procedure of acquisition	31
4.5.1	Challenges: time	31
4.5.2	Challenges: size	32
4.5.3	Challenges: encryption	32
4.6	Alternative procedures	32
4.6.1	Alternative 1: booting from live distribution	32
4.6.2	Alternative 2: Target powered on	32
4.6.3	Alternate 3: live network analysis	33
4.6.4	New Challenges (separate classes)	33
5	Identification	34
V	Ethics elements	37
VI	Optional: casi studio specifici	38

Part I

Cybercrime: threats, modus operandi, underground economy, financially-motivated malware

Chapter 1

Cybercrime: Threat Landscape

Working on security we work on **risk**, which is composed of different elements:

$$Risk = Asset \times Vulnerabilities \times Threats$$

The risk is the statistical and economical evaluation of the exposure to damage because of the presence of vulnerabilities and threats.

With no threats there are no risks, because it is the only thing that can nullify this equation. Assets and vulnerabilities are never absent.

With no *Threat model* we're completely misguided on the management of our system in terms of security.

1.1 Threat landscape

Threats can be roughly divided along three directions:

Internal vs External threats an internal threat comes from the inside of the organization, it is part of the organization, while an external one don't.

Generic vs Targeted many security threats are generic:

- Generic threats: when you walk towards the underground, you're subjected to pickpocketers: they don't pick you because it's you, you're one of the others, it is a generic threat.
Generic threats are not linked to us by definition, they are linked to us because we look alike somebody easy to be pickpocketed, not because we were the target.
- We can also consider also threats which are not so very generic: aggressions for sexual reasons are more targeted to one gender than one other, while still being generic.

- Targeted threats: specifically designed against us: criminals target **that specific company** (*e.g. they know how for racing cars, i want to steal from THAT company in particular*).

These directions also affect the kind of attacker: pickpocketers vs highly "professional" skilled stealer of information

Financially motivated vs Anything Else we divide:

- **financial attackers** (*which are the most of them*),
- **other attackers** (*governments, secret services, hacktivists..*).

Financially motivated attackers has 2 important positive characteristics:

- **easy to predict:** you look at valuable goods from companies and you can predict who the attackers can be (*e.g. ransomwares*)
- **they're relatively easy to handle:** just deny them the opportunity to take money, like making it too costly wrt what they want to earn.

Notice that what is easy to understand is how to handle them, not to actually handle.

Non financially motivated ones are more difficult to handle:

- We cannot make them too costly or too risky (*e.g. Russians vs Ukraine's Donbass power plant: they have an entire state funding them, and they cannot be arrested for it*), they have more money and they can take more risks
- If they're internal, they already know about the company and its security systems.
- The more motivated is the attack, the more determinated is the attacker

1.1.1 A gartner quadrant of threats

	Generic	Specific
Internal	disgruntled¹ employee	socially-engineered or dishonest employee <i>(financially motivated)</i>
External	Criminals, usually looking to make money <i>(financially motivated)</i>	A variety of advanced hackers <i>(mostly financially motivated)</i>

- **Generic internal threats:** a disgruntled employee who wants a raise, has something against his colleagues, was fired . . .
The threat doesn't need to be linked on what the organization does.
- **Specific internal attacks:** former employee stealing from old employer and bringing to the new one, it also exists a specific job which consists in being hired from companies just to steal from them.
Or, employee can also be *social-engineered* into behaving like attackers.
- **Generic external attackers:** attackers which want to earn quick money, they aim to things which are common for all companies, like stealing money from bank accounts and so
They are mostly motivated by financial reasons.
- **Specific external attackers:** industrial spies which aim for money, or governments or terrorists aiming for the destruction of a power plant near Donetsk.

The most important quadrant is than the one which comprehends dishonest or social engineered people (*human factor*): our instinct say us that we need to protect from the outside because we are clan-based animals, and we have our group of trustworthy animals (*our clan*) to which we are positively implicitly biased. **Think about the most famous security technologies:**

- Firewalls are meant to keep people out
- Antiviruses are meant for generic attacks, they use a generic list of malwares

Most of our security technologies protect against external attackers.

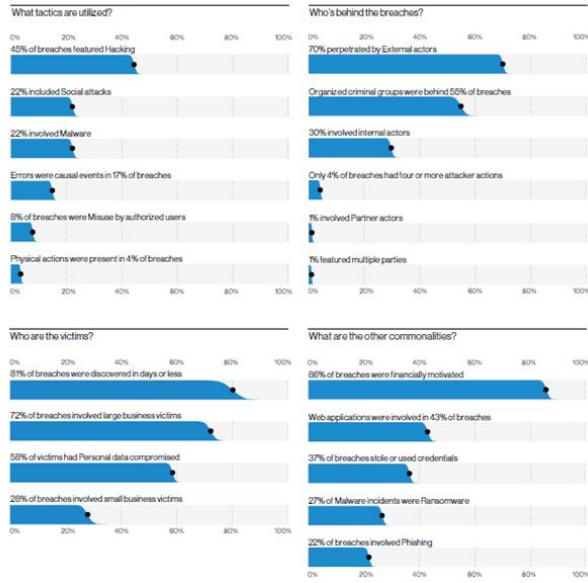
1.1.2 Examples of internal threats



A bit more on:

- **Reckless or socially engineered employees:** in security we often have a wrong perception of the *human element*.
When a company gets infected by a malware because a user clicked on a link or something similar, we blame the *stupid* user for it. But, actually it's not fault of the employee: it is fault of who had the responsibility to manage the fact that a threat was present and it had to be controlled.
Since there will always be someone to click on links, our job is to make sure that they doing it doesn't cause damages to the organization.
e.g. in aviation they study human factor to understand what to do to don't let people make mistakes: in airplanes cockpits there are different levers with different shapes, them are exactly shaped the same in different kinds of planes, it is a standard, and the reason is because people lost their lives because of similar levers.
The solution wasn't to train pilots, but to make the threat difficult to happen.
- **Thirty-party users:** they're for example consultants that work in a company for a long time, where they actually do jobs very similar to the ones of the actual employees while not being true employees of the company.

1.1.3 Data breaches and targeted attacks



Verizon has a big sample of incidents made to their customers:

- The largest part of the attacks features hacking of some sort, the second larger chunk comprehends social attacks.
- The 70% of breaches are perpetrated by external actors. This means that 30% of them starts from the inside, an incredibly high percentage. Consider that data are skewed², in the real world the internal attacks can be even more! (*remember the external threats reasoning, rationality (data) tells us that a lot of threats come from the inside, while our instinct guide us to be aware of the outside*)
- The 86% of them were financially motivated, 14% which are not, are a lot too!

These data are biased and are not representative of the true reality of threats, because simply they're not all collected. This collection can still help us to rationally think on how to manage the overall situation.

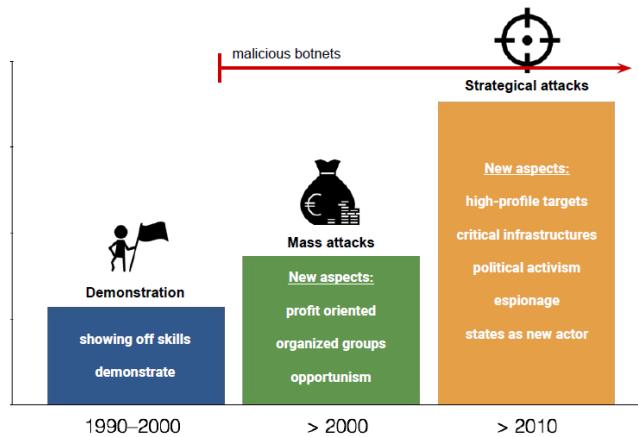
Keep in mind that some security incidents are specifically designed to be difficultly found, a lot of them has never been. This means that there is an entire set of breaches that no one ever investigated.

We measure attacks that we investigated/detected, and this is called the **observation bias**:

Other attacks by their nature be discovered, (*e.g. Dos, ransomware ...*) that's the reason why they are so prevalent in our statistics: they are observed.

²biased

1.2 Brief history of malicious software



Over the last few decades attacks and attackers changed:

- **1990–2000:** most of the attacks were meant to show skills and explore
- **2000–2010:** with the birth of the Internet, massive, profit oriented attacks born. In this period were born also groups working as profitable enterprises of cybercrime.
- **2010-today:** attacks evolved: mass attacks kept happening, but now a lot of them are high profile financial attacks (*before: ransomwares, i ask small amount of money to single persons. now: ransomware attack specific companies to get a lot of money*)

1.3 Financially motivated attackers

Today financially motivated attackers are the "mass", they're interested in monetizing their attacks in possibly two ways:

- **Direct Monetization:**

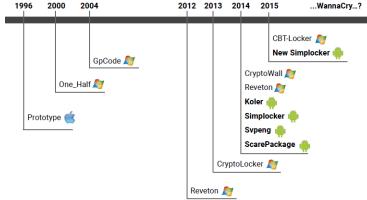
- Credit card/bank account frauds
- Ransomware attacks
- Crypto-miners
- Fake antivirus: they show you warnings that say your computer is infected, and to pay to activate the premium version to get rid of the malware. In reality they are just fake programs asking you to pay for a license.
- Premium calls: back in the days when you called to connect to the provider, someone managed to change the number to let you call a costly one.

- **Indirect Monetization:**

- Abuse of computing resources
- Information gathering (stealing of account infos to sell them)
- Making the machines part of a botnet to rent or sell the botnet

1.4 Direct monetization: ransomware attacks

1.4.1 Brief history of ransomware



In 1996, a paper describing a crypto-malware that was exactly a ransomware was published about 15 years before CryptoLocker which was the first really successful one.

They just *"looked into the crystal ball"* of what in the future would be valuable: taking as hostage digital valuables could become a good way to steal money.

Then something happened:

the internet in 1996 was really different than the one of 2013, ransomwares needed a way for the user to pay a ransom in a way that was easy to perform. The preferred way to perform payments on internet is with credit cards, but this kind of payments have two important defects:

- they are trackable
- they are refundable when fraudulent

Then a new suitable way for payments was born: **cryptocurrencies**.

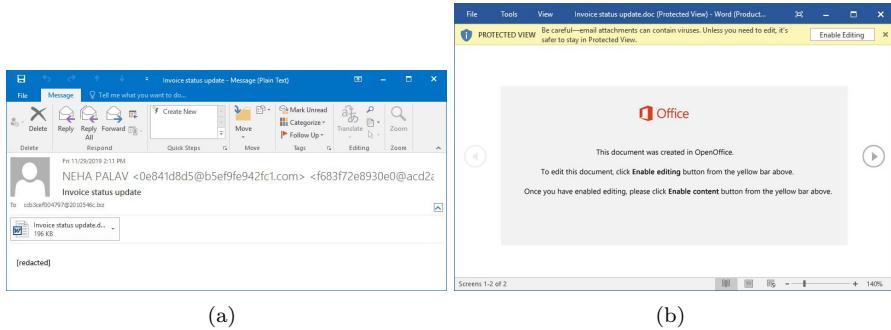
They perfectly fit this need: they are easily accessible, and payments are not reversible.

1.4.2 Ransomware screenshot and social meanings



Look at the structure of the timer: it is made to let you to perceive urgency. The stress of the urgency makes human take bad decisions, with the timer running you're more prone to pay because you feel that you have limited time. In social engineering attacks, urgency is always an important part. (*think about ultimatums in diplomacy, job offers with time limits, poltrone e sofà...*)

1.4.3 How to get infected



Looking at the images we can see an interesting example:

- User receives an e-mail with a short text that contains some sensitive information and an attachment which is social engineered in a way that many people would click
- Microsoft Office shows a message to try to prevent code execution, but the user is prone to click "Enable" because the text contained in the document makes him think that's the right thing to do, even if a lot of grammar errors are present and the whole situation looks suspicious

One could have said that the user must had to be trained to not click, but the reality is that he is not understanding what's happening on his computer.

We can't train a generic user to not click: if your job is to open invoices received by e-mail, there is no way you can be trained.

Even if we train him to click only on what is coming from attendable e-mail addresses, if the "*attendable*" part was hijacked?

What we really should do is to figure out a way that when someone clicks enable, this won't cause any damage. Of course, to train the user to let this happen the less frequent possible is good, but not the main thing to do.

1.4.4 Encryption mechanism: how do ransomware work

Once the malware starts running on target computer, it generates a random symmetric key³, usually one per file, and it encrypts each file with it.

The symmetric key itself it's encrypted with a public key, of an asymmetric key pair generated on the server of the group that runs the malware. The private key is stored on the server, and it will released only when the ransom is payed. Most of this process is automated: when cryptocurrencies are transferred to the criminals' address, the server releases the key associated to it. Different keys to allow the attackers to decrypt only a part of the samples to demonstrate that they can do it.

³used both to encrypt and decrypt information

1.5 Indirect monetization: Botnets

The word botnet comes from the words *robot* and *network*. A botnet is composed by few hundreds to millions of infected devices which run some sort of malware they got by opening an infected email attachment, by plugging an infected USB drive, visiting an infected website (*these are examples of ways to get infected*) Each botnet has a **botmaster** who controls and rents it out to perform tasks (*denial of service, spamming, phishing campaigns, crypto mining ...*), and his only objective is to earn money by renting them.

The significant challenge with this type of crime is that each bot per-se is not dangerous for the machine itself, but for other machines, and since the cost of cleaning up a machine falls on its owner, some can decide to not do anything about. This cost can be seen as a small fee to be payed by *the community* to get everybody safe. Like a vaccination, some cases of infected machines are not a big trouble for the community, while a lot of them can be really dangerous for everybody.

1.5.1 Rise of the bots

Back in the days botnets were used to control IRC chats (*see IRC wars*). Then instead of using the compromised machines to control the chat, botmasters started to use the chat to control the bots.

In 1999, there was one of the first DDoS attacks, which was against University of Minnesota and used at least 227 bots.

In the 2000s DDoS attacks against high profile websites (*Amazon, CNN, eBay ...*) got huge media coverage.

1.5.2 Geolocalization of botnets command and control

Rank	Country	Q2 2020	% Change Q on Q
#1	United States	896	7%
#2	Russia	812	32%
#3	Netherlands	337	61%
#4	Germany	185	7%
#5	Singapore	131	157%
#6	France	108	35%
#7	Great Britain	89	37%
#8	China	74	-15%
#9	Bulgaria	72	38%
#10	Hungary	70	New Entry

This chart shows the number of new botnet C&Cs detected by *Spamhaus* in the second quarter of 2020, and the increase wrt the first one.

By keeping in mind the *observation bias*, we can also introduce another kind of bias which is called **heatmap effect**: we're biased on the density of the population and on how much data we collect from a certain country.

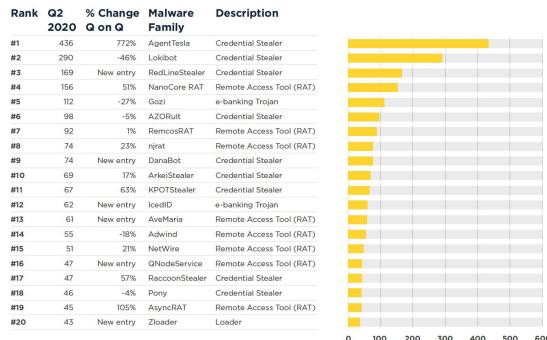
In this kind of charts we'll always have U.S.A. on top, because there's more penetration in the internet, and also this kind of things is tracked. While we'll have less data from less technological advanced states or *perfectly democratic countries* like Russia or the P.R.C. which can result in lower positions while in reality being the first ones.

Here we see an important increase of C&C from The Netherlands:

- it is possible that **something is on**
- or simply that some Netherlands organization decided to participate in this data collection feeding a lot of *new* data

Funny thing about Russia and P.R.C: Russian people hack russian computers too, while P.R.C citizens don't hack into their fellows machines.

1.5.3 Type of (botnet) malware families



Every botnet in general can be used to do any sort of things, but they're used to do only one of them:

- **Credential Stealers:** used to steal sensitive credentials.
- **Banking Trojans:** credential stealers specifically designed to perform stealing of bank accounts information. (*Gozi is the most common one, Zeus the most important one*)
- **Remote Access Tools:** basic botnet oriented malware, which allows to control a computer in order to perform whatever.
- **Loaders:** specifically designed to allow a botmaster to load a program of whatever sort on a computer. Maybe a client ask you to install a certain malware on a lot of computers, and you can do it with a loader.

We talk about families because there are criminal groups that only develop their source code, and who performs the attacks buys the code and personalizes it for the specific purpose they need.

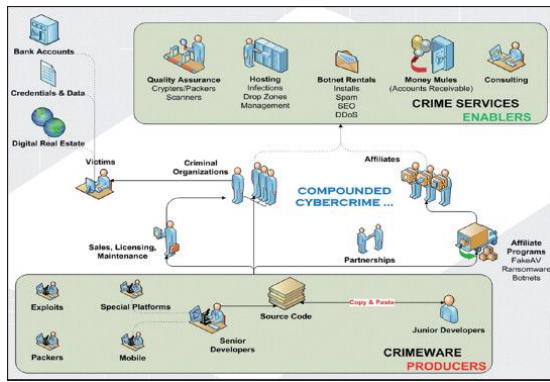
So we can consider three businesses: how to develop them, how to configure them, how to use them.

There is a market for services related to malwares and cyber attacks (*cybercrime as a service, underground market...*) structured around the needs of cyber criminals. This market is fueled by the money that these schemes make. Some of these money pays for the tools used to make that money.

1.6 The cybercrime ecosystem

The status quo consists in **organized groups** performing **various activities**:

- Development and procurement of exploits
- Site infection
- Victim monitoring
- Selling *exploit kits*



These ecosystems can exist because some of the activities done are not illegal: developing exploits per-se is not illegal, selling one is not illegal, the usage of them may be or may be not illegal. Even the services related to the configuration of malware are not illegal, while execution and operation may be. If you're sufficiently shielded and you run these *businesses* in countries that cannot persecute you, we can even know your name and surname but you cannot be arrested.

- Developers write the source code for malwares with the help of packers and exploits developers
- Crime service enablers:
 - Quality assurance makes sure that antiviruses don't detect their software
 - Bulletproof hosting is a kind of *close-an-eye* hosting, (*e.g. russina business networks*) with permit certain suspect activities over their infrastructure, they're sort of borderline organization with lot of regular customers, and some bad ones

1.6.1 Identity sales

People's identities are sold on the black market. They are worth because they can be used for fraud, to open bank accounts used for money laundering for example. The more they're useful, the more they are worth. Worth 50 dollars for you that sell it, the one who pays is going to use them for fraud which is worth more. The black market is fueled from an enormous amount of money that people stole.

1.6.2 Drive by download

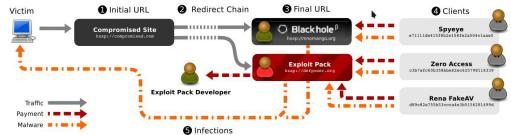


Figure 1: The drive-by-download infection chain. Within the exploit-as-a-service ecosystem, two roles have appeared: exploit kits that aid miscreants in compromising browsers (1), and Traffic PPI markets that sell installs to clients (5) while managing all aspects of a successful exploit (6, 8, 9).

This is another way in which malwares are installed: an exploit breaks into your browser and executes code on your machine. This requires your browser to be vulnerable and for you to visit a compromised website with the exploit running (*streaming, cracks of programs websites, ...*). The most compromised are *factually illegal websites*: you're not scared of the strange url if you are in need to see the last movie in streaming, download the expensive program's crack, this is in contrast with a normal situation like going to amazon to buy shoes.

But, *traps* can be also present in legitimate websites.

What actually happens is that the users end up in a series of redirects, called **redirect chain**, which will end up in a visit to a website running what is called **exploitation pack** for your browser.

1.6.3 Exploitation kits sales

This kind of kits is sold in the dark web by well known organizations, take *Blackhole* as example:

they were buying exploits from developers to earn 10s of millions per month renting them.

Their boss, surprisingly called Dmitry “Paunch” Fedotov, was arrested in 2013 after years of running the website.

1.6.4 Monetization on the dark web

Monetization takes a lot of forms, one of them is stealing credit card numbers to sell them for a relatively small price.

The hackers get *easy and fast* money, while the buyers need to use the real

money on the cards to buy expensive objects or to start frauds because they can be easily refunded. (*apple computers, ethnic travel example..*).

1.6.5 Cybercrime and perception

Online, you detach people from the picture. It's actually easier to commit crimes, it's easier to crack a program instead of stealing a car, lot of cyber criminals would never be criminals in the real world. But when you do cybercrime, your perspective is different: you have another perception of what you're doing, that's the reason because crime happens more online.

Ethical people understand the consequences of their actions, and may decide to not do them, but someone can not care about what can happen as a consequence of their actions and perform them anyways:

in the Jamal Khashoggi case, the exploiter which wrote the code for the spyware that ended up to make him killed, would never kill someone. But the consequences of his actions did.

1.6.6 Money mules and money laundering

Most cybercrimes end up with a digital form of money, which criminals want to bring in a physical form, completely disconnected from what they did in principle.

- They can pay invoices to companies somewhere in the world, which in a certain way makes sure that the money come back clean. (*traditional way*)
- They can make use of **money mules**

Money mules are intermediary people which by purpose or not make earnings coming from cybercrime clean money.

Accomplished ones may be people that have nothing to lose (*prejudiced, poor people ...*), they just open bank accounts by their names and get the money transferred and then withdrawn and handed to criminals.

Some of them may also open accounts by somebody else's name using identities that may be stolen with an attack and bought online. They're most difficult to catch because the Police needs to be actually there while they are withdrawing money.

Unaccomplished ones may be people which are fooled by things like the Nigerian prince scam: they get the money on their bank account, send 70% of them to criminals in packages or by moneytransfer, and keep 30%. Most of the cases they get arrested and have to pay back all the money, also the ones they don't own no more.

Buying and selling of goods (*ricettazione*), videogames currencies, ..., are other ways to perform money laundering.

Chapter 2

Cryptocurrencies: abuses and forensics

We talk about cryptocurrencies in the way they are used in cybercrime.

2.1 Bitcoin and blockchain

Bitcoin is an attempt to create **electronic cash**: a digital currency that has lots of properties that physical money have, one of them is **no central authority**. What Satoshi Nakamoto wanted to have is an electronic distributed ledger¹. Having no central authority is a really hard problem to solve (*byzantine consensus*), he did it with **the blockchain**.

The blockchain is a **shared, append-only, trustable** ledger of all coins transactions. The limits of distributed consensus defined in the Byzantine Problem and CAP Theorem are solved using the technique of **proof-of-work**.

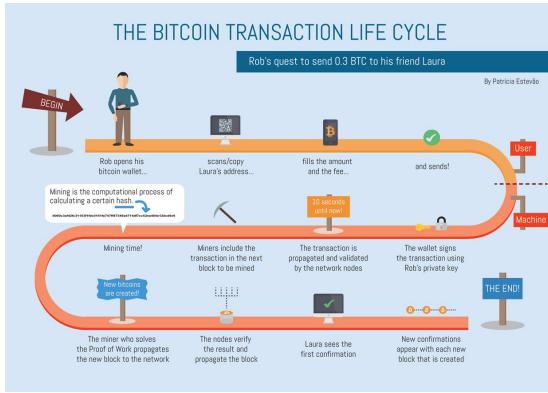
2.2 Wallet and addresses

A wallet is the software that allows to manage and store the **public** and **private** keys for each of the user's bitcoin addresses. To create and sign transactions, track the balance.

A bitcoin address is an alphanumeric string which identifies a *point* where you can send bitcoin to, and where they can be sent from. In order to know how many Bitcoin are *stored* in that key you need to go to the origin of all the transactions and track the flow of them to understand how much of them are there now. This computations is not really efficient but can run without central authority, so the only reason for using a blockchain is if it is really so important to get a way out of a central authority.

¹ledger=registro

2.3 The bitcoin transaction life cycle



As soon as the transaction is on the immutable ledger, the money are on the other address. How do we ensure that this is the only transaction done with those Bitcoins?

Our immutable ledger must be consolidated, with a central authority that would have been easy, how do we agree with no central authority? We need a history that cannot be modified, and to do this we talk about mining.

Mining is the process to generate a block containing all the transactions, and to append it to the blockchain.

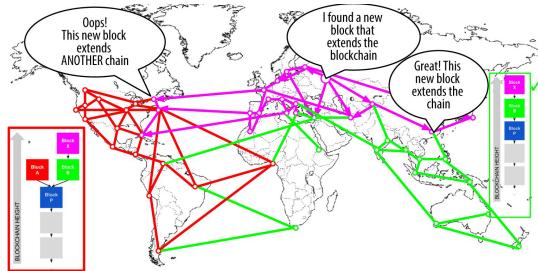
2.4 Mining

Miners compete to generate a new valid block that solves a complex mathematical problem by brute force: the solver is then rewarded a fixed number of BTC. It is a computational demanding activity: miners put a set of transactions in the new block, put there a destination for the reward and then start brute forcing the block to find a sha256 hash of it which starts **with a certain number of zeros**. The more zeros, the more difficult is the process.

When a block is added to the blockchain, the other miners stop brute forcing their blocks and start to compute the hash of another one, to link it to the last found block. The difficulty increases more and more, the reward got smaller and smaller but BTC value gets larger and larger.

Besides of the reward you get, you can also put a fee you want to receive from the transactions you included in the block. Since reward is becoming smaller, fees are becoming larger. Now the fees make small transactions unfeasible.

2.5 Fork events



Sooner or later two miners will get a solution at more or less the same time. This causes a fork where the situation is that for some miners the blockchain ends with block A and for some others with block B.

Now the blockchains are two, and sooner or later some miner will find a solution and append a new block in one of them. The rule says that **the longest blockchain is the true one**. As soon as one blockchain becomes longer, all the miners move to that leaving the other one sadly alone.

This is done because in order to revert a payment you'd need to go back on your block, go ahead mining until your blockchain is the longest one: this means that you would compete with all the other miners. To be sure that a certain transaction is there and cannot be canceled, you must wait that the block in which it finds itself is a couple blocks behind the last one.

2.6 Bitcoin and black market

Bitcoin immediately started to be used for black markets. Because you had a way to pay without an infrastructure.

2.6.1 Pseudo-anonymity

The first reason to use Bitcoin for black markets is **anonymity**.

In reality Bitcoin is actually **pseudonymous**: the identifier for the transactions is not directly connected to an identity, but to an address which can be seen as a username.

Every entity generates multiple keys for themselves, so more pseudonyms. This pseudonymity is robust because addresses don't refer in any way to their owners. Transactions are public by the way, and this is a terrible property if you're interested in anonymity; while addresses are not connected to people, there is something that connects addresses related to the same person:

- transaction can have multiple keys as input, if a transaction has multiple inputs, very likely that the inputs are all owned by the same entity

- Bitcoin works in a way in which you cannot spend less than the whole amount of bitcoin you have in your address, to send a smaller amount to somebody, you'd generate a transaction with two outputs:
 - The destination address
 - A new address called Shadow Address which is owned by the same person who started the transaction, where the rest of the currency is sent.

Until 2013, a bug made very easy to know which of the two outputs in a transaction were the shadow address, hence making easier to track their correlations. It was then possible to track addresses in a way in which keys belonging to the same entity were collapsed in large sets. When you have one of those addresses connected to somebody's name you can actually connect all the others in the set to him.

Silk Road Example

Silk Road was a black market which used BTC as currency, it was ran by someone under the name of "*Dead Pirate Roberts*".

It was found out that someone using the nickname "*altoid*" were advertising Silk Road on different websites, and with the same nickname earlier in the past was also hiring developers for a PHP project. (*which then became Silk Road*) In that specific post, altoid was asking to contact him at "*rossulbricht at gmail dot com*" if interested.

He also posted a request for help which included PHP code with his address in it: 1LDNLreKJ6GawBHPgB5yfVLBERi8g3SbQS², which then was found to be associated with Silk Road

Ross Ulbricht was arrested in 2013.

2.6.2 Why do people use Bitcoin for ransoms?

For these two characteristics:

- BTC transactions, in contrast with common electronic payment systems, are irreversible.
- To use other kinds of digital payments you need an infrastructure (*bank account*...), with cryptocurrency you just need an address.

²today there are 20 dollars of Bitcoin

Part II

Fraud detection and analysis

Part III

Digital Forensics principles

Chapter 3

Introduction to Digital Forensics

3.1 What does forensics mean?

- **Forensics** is the application of scientific analysis to reconstruct evidence.
- **Digital Forensics** is one of the disciplines of forensics, is the application of scientific analysis methods to digital data, computer systems, and network data to reconstruct evidence.

Forensics is strongly related with laws: different jurisdiction means different procedures.

3.2 The Daubert standard (USA)

Generally speaking, we can have two sources of evidence:

- physical evidence
- eyewitness testimony

To provide physical evidence, an expert is allowed to give an opinion in court because of the fact that he is an expert, and the use of scientific methods is the reason why he is listened to.

How do we define what an expert is?

3.2.1 The Daubert Standard

We define as expert witness someone who has witness because of its own experience.

The Daubert standard define an expert witness as follows:

A witness who is qualified as an expert by knowledge, skill, experience, training, or education may testify in the form of an opinion or otherwise if:

- The expert's scientific, technical, or other specialized knowledge will help the trier of fact¹ to understand the evidence or to determine a fact in issue;
- The testimony is based on sufficient facts or data;
- The testimony is the product of reliable principles and methods; and
- The expert has reliably applied the principles and methods to the facts of the case.

Reliable principles and methods means that they are scientifically found. You need an expert to use scientific method to establish those facts:

3.2.2 What is scientific? (Italy)

- Galileo: scientific means repeatable, you actually make an experiment to demonstrate that something can happen
- Popper: scientific means falsifiable, if you're able to create, at least in your mind, an example which proves the opposite of a statement, then the statement is scientific.
 - Example: *Stefano is a kind person* is not a scientific statement, because it is not falsifiable.

Some of the questions are not falsifiable, hence not scientific. An example is a criminal which keeps images of kids on his own computer. Did he do it willingly or not? This is not falsifiable.

What is possible to do is to look at the folder to see if it was opened multiple times, look at the file to see if it was in the browser's cache or in a specific folder on the criminal's computer.

3.2.3 Daubert Test for scientific

Factors to consider (USA) to establish if something is scientific or not:

- Whether the theory or technique employed by the expert is generally accepted in the scientific community
- Whether it has been subjected to peer review and publication
- Whether it can be and has been tested
- Whether the known or potential rate of error is acceptable; and
- Whether the research was conducted independent of the particular litigation or dependent on an intention to provide the proposed testimony.

¹the judge

3.3 Example of forensic engagements

We do forensics for different situations in different contexts

Example of forensic engagements

Situations and constraints	Crimes and events (examples)
<ul style="list-style-type: none">Internal investigations (inside an organization)Criminal investigations (defense or prosecution)Post-mortem of a system to assess damage / define recovery strategyResearch (honeypot, etc)	<ul style="list-style-type: none">Child pornographyFraudCyber extortion / threatsEspionageCopyright infringementsPolicy violations

Depending on which part of the lawsuit you're working for: (*"prosecutor"*, *"judge"*, *"lawyer"*), the things that you can do are different.

The procedures has always to be contextualized in what are the purposes and which constraints the purpose has.

Crimes:

- The first investigated crime is the one that has to do with children. The reason why is that it is one of the easiest to prosecute.
- Fraud is the second one. Prosecuting fraud is difficult because there is a large amount of them which are perpetrated by people who live in countries where the laws kind of permit it. It is still denounced because people can get their money back.
- Cyber extortion is the crime which consists into stealing personal data to get money from them. Extortion happens with family, work related, revenge cases.

There exist also a lot of non-cyber crimes which involve digital components:

- Search for traces in digital devices in murder cases
- Tracking or geo-localization of mobile devices

Digital components can be fundamental.

3.4 Phases of an investigation (Pollitt)

- Source acquisition: how we preserve digital evidence
- Evidence identification: how we analyze digital evidence
- Evaluation: how we take evidence and pack with the specific case
- Presentation: how we put together all of this in a court

Part IV

Acquisition, analysis, evaluation and presentation of evidence

Chapter 4

Acquisition

Forensics was born in the USA, it borders with law so it was developed with US laws in mind.

The legal system of USA and the EUs ones are extremely different, and so is the court approach:

- In the U.S. most of the cases are tried by juries of peers, the judge works to make the court work. In fact, the judge is the one to decide if evidence is admissible or not. If not admissible, that could not be talked about from jury and lawyers or taken into account. Admissibility of scientific evidence is completely based on the concept of **chain of custody**: *tracking where the evidence was taken, who was in custody, where was it stored, who analyzed it, what was done to it*. If that is broken, evidence becomes inadmissible.
- In Italy is the judge the one who takes the decision. The jury exists only in Corte d'Assise and it is made by people extracted from a certain set. Is the judge to evaluate the admissibility of the evidence, it is inadmissible only if it was taken in violation of laws.

By the way, the Council of Europe has an international law agreement called **Convention of Budapest on cybercrime**, it was written in 1999, and was incorporated in the Italian law in 2008. Also international standards from ISO/IEC can be applied.

4.1 Brittleness of digital evidence

Digital evidence is **brittle**¹, it is because if it is modified there is no way to tell. In other words, it is not **tamper evident**². This means that there is no way to say if the chain of custody was violated.

Digital evidence can also be fake created:

¹fragile

²non mostra segni di manomissione anche se manomessa

- By changing for example the clock of a computer, it is possible to create a fake file which was modified in a different time, and there is no way to figure out if that happened.

We need an entire process of acquisition to create ways to make digital evidence as far as possible tamper evident, there is a need to ensure:

- Legal compliance: evidence must comply with the laws.
- Ethical behavior from all parties: even the police can act unethical
- Detection of errors in good faith
- Detection of natural decay

4.2 The usage of hashes in digital forensics

Hash functions are used to record the state of an object at a given step of acquisition. It is constantly checked to ensure authenticity and non-tampered state of that at any further phase of acquisition. They are used to "freeze" the crime scene.

- We don't know what happened before that first step
- We can only tell if something has been tampered, but not what has been tampered. We also cannot restore data if tampered.
- Hashes are used to prove that something has been modified, but cannot tell what.

Since evidence is going to be stored on a certain media, hashes must be put in another place (*e.g. writing them on a physical register is the most frequent, or use digital signatures*).

4.3 Hardware and software for acquisition

- Hardware:
 - Removable HD enclosures or connectors with different plugs
 - Write blocker
 - External disks
 - USB, firewire, SATA, e-SATA controllers if possible
- Linux: extensive native file system support, ease of accessing drives and partitions without mounting them

4.4 Bitstream images

What we want to acquire, if possible, is a **bitstream image**, a bit-by-bit clone of the original evidence media. The main reason is because if we only copy the allocated content we potentially lose information stored in the unallocated part of the disk. This is called a **forensic clone** or clone copy or image of the disk.

4.5 Basic procedure of acquisition

- Disconnect the media from the original system, if possible
- Connect the source media to the analysis station, if possible with a write blocker
- Compute the hash of the source:
 - Linux: `#dd if=/dev/sda conv=noerror,sync — sha256sum`
 - `conv=noerror` means "*keep going even if the system generated some errors*"
- Copy the source:
 - Linux:
 - `#dd if=/dev/sda of=/tmp/acquisition.img conv=noerror,sync`
- Compute the hashes of the source and the clone
 - Linux: `#dd if=/dev/sda conv=noerror,sync — sha256sum`
 - `#sha256sum /tmp/acquisition.img`
- Compare the three hashes

If the hashes are different, it means that the copy didn't happen correctly or that you tampered with the drive, or that some damaged block read broke it, or that some damaged block reads everytime something different ...

It could be also good to compute MD5 and SHA-1 hashes, for redundancy and backward compatibility.

4.5.1 Challenges: time

A typical hard drive capacity today is 1TB. Transfer speeds are in the order of 600MB/s (*SATA 3*), but mechanical drives reach an average of 80MB/s (*SSDs are actually fast as SATA3*), USB connectors are up to 100MB/s.

This means that for a 1TB drive you can expect to wait several hours to complete a copy or to run a hash. Tools like **embedded duplicators** or software like dcfldd can make this run in parallel to save time.

4.5.2 Challenges: size

File systems limit to 4TB mostly for the size of a file. If a drive is larger, it has to be spliced in more files.

Space is also needed to keep data for all the devices used in an investigation, which can be in the order of the hundreds of terabytes, that's why NAS or SAN systems are common in forensic shops.

Sometimes it can be also useful to move images across a network, for example by using netcat to listen for a stream generated by the computer which is taking the acquisition.

4.5.3 Challenges: encryption

Many machines use encryption. In some cases the key for decryption is stored on the motherboard of the computer, and so the image became useless without the computer.

4.6 Alternative procedures

4.6.1 Alternative 1: booting from live distribution

Sometimes it is mandatory to work directly on the actual machine, because of systems with weird connectors, RAID devices or because of specific investigation constraints.

In these cases it is possible to live-boot the system using a Linux distribution targeted for forensic analysis, like *Tsurugi* or *BackBox*³.

This process has to be performed with great care, different systems have different ways to access the boot menu, and if by chance the original operating system is booted, it may cause a loss of useful data.

4.6.2 Alternative 2: Target powered on

For many reasons can happen that the target machine is turned on:

- Maybe it is running a critical service for the company we're working for
- Maybe it was seized while turned on

Working with a powered on machine is called **live forensics**. Can we turn it off? Probably not if the machine is a critical one. Should we turn it off? Maybe not if we're trying to do live analysis of an intruder. We want to exclude an intruder? We can disconnect the network.

Since the machine is turned on, our actions are going to modify its state, our commands, even turning the machine off will change it, because of the operating system's operations.

³The reason why we use specific distribution, is because we do not want to overwrite linux swap partitions on the media

Now the prospective changed, before we thought of evidence as immutable as possible, now we don't.

Since our actions are going to tamper with the state of the system, we need to **document what we did** (*to preserve chain of custody (USA)*), here we don't have the safeguard of the hash. Work in volatility order:

- Dumping the memory may be useful, for example most of the break-in attacks are file-less.
- So is saving runtime information: network, process information, etc...
- Consider encryption before turning the machine off (*maybe the device can become unusable after*)
- Finally, disk acquisition

If possible, perform the acquisition without a shutdown, if not, if the machine has to be reboot after, shut it down properly. If it is not crytical it is possible to pull the plug to not tamper the disk.

Useful commands

- Network data: `"ifconfig -a ; netstat -anp ; route -n ; arp"`
- Process data (to store process information): `"ps aux ; lsof file"`
- Users data: `"who; last; lastlog"`
- Memory acquisition: `"Mantech mdd, win32dd, Mandiant Memoryze"`

4.6.3 Alternate 3: live network analysis

In enterprise environments if we want to observe an attacker *live* whithout him being scared of our actions, we can use two observation points that are outside of the machine: logs and network traffic (*we'll have a separate class*)

4.6.4 New Challenges (separate classes)

- Forensics in cloud environments
- Mobile forensics
- SSD drives work different than magnetic drives

Chapter 5

Identification

understanding DB structure, understanding source code, reverse engineering
Identification means to perform actions that fall anywhere in computer science.
We look at either the methodology and specific forensics cases.

Typically we use Linux:

- extensive native file system support
- we can just dump a disk image as a file with linux, and we can mount the copy as if it were a drive, also read only to prevent writing and keep the copies not touched.
- common to use virtualized windows machines to execute some utilities, software available only on winzozz

There is an exception, several forensics tools run under windows and replicate a lot of these functionalities e.g. xways

Nobody uses windows by itself. It must be confined, it stinks. It automatically mounts drives and writes on them. No hot swap devices, no support for every file system Best is to use linux with windows virtual machine. Typical way to share file use vmware tools or network share. Samba-SMB Keep in mind that if you share something in these ways, this sharing is file level, any utility that analize drives cannot work in this way. To use a windows utility to analyze drive, you need to mount the copy as a drive for windows, usually with virtualization software. In these cases, you can use non-persistent mode of vmware/virtualbox: map a device so that any write get recorded on the side, not actually changing the drive.

CONSIDERATIONS THAT WE NEED TO TAKE IN OUR APPROACHES TO ANALYSIS Basic consideration goes back to our definition of scientific = repeatable

Experiment: without explanation of how the experiment works, it doesn't carry any meaning examples: machine does dna matching (two samples, outputs a probability). The only way for this to be acceptable as scientific is by showing

that the machine under reasonable circumstances performs the operation, we can assess how well, steps are always the same, once we've opened the machine, closed and sealed it, we could at almost admit the results. We should be able to tell if the machine is still working correctly

Use open source sw or you need to be able to show me that the tool outputs can be reproduced with another experiment Law enforcement only tools like kid's images thing You have to show me that the same can be extracted with a non proprietary tool. proprietary software have open source access for judges or people related

DURING ANALYSIS we WORK ON EVERYTHING We may need to apply many techniques from computer science. The reason why people goes to a forensic expert being able of doing analysis competence: things he knows about cs in general.

good forensics expert maybe cannot be competent in processor design call someone else to work with (La Sciuto)

RECOVERY OF DELETED DATA Data/evidence voluntarily or unwillingly deleted because time passed, by the OS, by formatting drive, by a fault damaged drive. In many of these cases, we may be able to reconstruct all or part of the information ora vediamo il file system di unix

Is a way to organize space on a drive to store information. Archive like thing The basic idea is to have units of information= files, to put on the storage drive, so to have an index of what is stored and where. If the index is there I can go to the place where things are stored, with no index i may not reach them but they can still be stored. inodes keep pointers to blocks (metadata) in unix you have a series of ten direct block entries that tell there is this content on the drive and it belongs to the drive ease pointers* that point to data blocks i have metadata with pointers

when we delete a file the operating system just marks the file as deleted and not show it no more these things will go away at random and independently from each others sometimes it happens that the os reconstructs the tree of the filesystem, it doesn't care about deleted files, at this point the metadata flagged as deleted disappears. Independently from this, sometimes when the os need the space of a deleted datablock it takes it. those things happen at random, and in the case of blocks they won't necessarily happen all at the same time. statistically on a large hard drive there is a good possibility that metadata goes away before data blocks.

DISK GEOMETRY tracks, cylinders when you read and write, the minimum portion of the track is called sector we still use this terminology to refer at the "geometry" of the drive. What we care about is the sector The os can not really work on a sector, because of different sector sizes filesystems are made to work on clusters of sectors mimo NTFS created clusters of 4kb independently from the sector dimension on the drive.

OS store files cluster by cluster. You store a 5kb file on a filesystem of size 4k you use 4k + 1k/5k of another sector

Interesting effect is that the file is blocked by clusters, written by sectors. This file spans 7 sectors, (image). But the drive will not be overwritten by

data because the data we wrote is less. Slack space In the slack space we know there are remainances of old files. Btw some files get stored once and stay there forever. Example dll Shared libraries What happens if i save a dll on top of a file that was there before? The slack space is there with the dll forever. Years later you can find files deleted years before. Disadvantage is that they are a small portion. Less than 4k of an email, webpage could be useful to retrieve of an image, video, big files less useful.

Some filetypes need the complete file to be interpreted. Small fragment of encrypted file is useless.

disconnected image

Sectors are one after the other (slide 12), if we place a file on a drive there is a good possibility that all the sectors of a file are one after the other. Put ourselves in the situation in which metadata are lost but data still there. I can take the entire drive, scan it as a string of bytes and seek issues: fragmentation or other things he said 13:57

typical though by seeking we can seek for specific things that have headers and footers and collect them. Some of them will be broken, some of them will have been overwritten in some places Some of them can be weird Some of them will be perfect.

The issues are what happens if the file is fragmented, it was a big problem on small drives, now in most cases drives are so large to not fragment and most of os try to not break files for performance reasons. If files are broken usually they're broken in two There are some techniques to find fragmented portions of a file

Another issues is with encrypted, compressed files. In those cases we really need the file to be complete Headerless files because they were taken away header or footer compressed/encrypted headerless are not approachable.

slide 13 tools photorec is a carver but it works better with pictures and videos gpart to recover damaged partition tables carver that looks for partitions testdisk to recover

Part V

Ethics elements

Part VI

Optional: casi studio specifici