

# Fondamenti di sicurezza e privacy

29 gennaio 2023



# Indice

<b>1</b>	<b>Introduzione</b>	<b>1</b>
1.1	Concetti fondamentali . . . . .	1
1.2	Categorie di attaccanti . . . . .	2
1.3	Attacchi di tipo Supply chain . . . . .	3
1.4	SolarWinds . . . . .	6
<b>2</b>	<b>Cyber Kill Chain</b>	<b>7</b>
2.1	Fasi della cyber kill chain . . . . .	7
2.2	Trickbot . . . . .	8
<b>3</b>	<b>Panorama sui cyber threat</b>	<b>9</b>
3.1	Overview 2022 . . . . .	9
3.2	Malware . . . . .	9
3.3	Trend recenti negli attacchi malware . . . . .	9
3.4	Attacchi comuni . . . . .	11
3.5	Attacchi cloud . . . . .	11
3.6	Attacchi a dispositivi IoT . . . . .	11
<b>4</b>	<b>Social engineering</b>	<b>13</b>
4.1	Ciclo di vita dell'attacco . . . . .	13
4.2	Tipi di attacchi . . . . .	14
4.3	Attacchi di phishing . . . . .	15
4.4	Attacchi di spearphishing . . . . .	16
4.5	Tattiche di influenza . . . . .	16
4.6	Come un'organizzazione può fermare gli attacchi di phishing . . . . .	16
<b>5</b>	<b>Malware</b>	<b>19</b>
5.1	Tipi di malware . . . . .	19
5.2	Prevenzione e riduzione dei danni . . . . .	20
5.3	Ransomware . . . . .	22
5.3.1	Killer switch . . . . .	23
5.3.2	Cyber kill chain di un attacco ransomware . . . . .	23
5.4	Wanna Cry . . . . .	26
5.5	Prevenzione specifica per gli attacchi ransomware . . . . .	26
<b>6</b>	<b>Cyber War e attacchi ad infrastrutture critiche</b>	<b>27</b>
6.1	Sistemi di controllo industriale . . . . .	27
6.2	Cyber kill chain per attacchi ICS . . . . .	29
6.2.1	Esempi di attacchi reali . . . . .	29

<b>7 Autenticazione dell'utente</b>	<b>33</b>
7.1 Autenticazione con password . . . . .	34
7.1.1 Attacchi alle password . . . . .	34
7.1.2 Attacchi offline . . . . .	35
7.1.3 Attacchi online . . . . .	37
7.1.4 Protezione contro gli attacchi alle password . . . . .	37
7.1.5 Attacchi passivi . . . . .	37
<b>8 Identità digitali</b>	<b>39</b>
8.1 Identità digitale . . . . .	39
8.2 Single-Sign On . . . . .	40
8.3 Identità federata . . . . .	41
8.3.1 SPID . . . . .	41
8.4 SAML . . . . .	42
8.4.1 Asserzioni SAML . . . . .	42
8.5 Shibboleth . . . . .	42
8.6 OpenID . . . . .	43
<b>9 Protocollo OAuth</b>	<b>45</b>
9.1 Authorization Code Grant Flow . . . . .	45
9.1.1 Flusso . . . . .	46
9.1.2 Flusso (pt. 2) . . . . .	46
9.2 Authorization Code Flow with PCKE . . . . .	47
9.2.1 Flusso . . . . .	47
9.3 Resource Owner Password . . . . .	47
9.3.1 Flusso . . . . .	48
9.4 Client Credential . . . . .	48
9.4.1 Flusso . . . . .	48
9.5 Device flow . . . . .	48
<b>10 Gestione degli accessi</b>	<b>49</b>
10.1 Modelli di controllo dell'accesso . . . . .	50
10.1.1 Discretionary Access Control . . . . .	50
10.1.2 Role Based Access Control (RBAC) . . . . .	50
10.1.3 Attribute Based Access Control . . . . .	51
10.2 XACML . . . . .	51
10.2.1 Specifica delle politiche . . . . .	52
10.2.2 Algoritmi di rule-combining . . . . .	53
<b>11 Risk Management</b>	<b>55</b>
11.1 Standard NIST . . . . .	56
11.1.1 Modellazione del rischio . . . . .	56
11.2 Risk assessment secondo lo standard NIST . . . . .	57
11.2.1 Preparazione al risk assessment . . . . .	57
11.2.2 Esecuzione del risk assessment . . . . .	58
11.2.3 Comunicazione dei risultati . . . . .	61
11.2.4 Mantenimento del risk management . . . . .	62

<b>12 Introduzione alla privacy</b>	<b>63</b>
12.1 Definizione di privacy . . . . .	63
12.2 Proprietà della privacy . . . . .	64
12.2.1 Anonymity . . . . .	65
12.2.2 Unlinkability . . . . .	65
12.2.3 Undetectability . . . . .	65
12.2.4 Plausible deniability . . . . .	65
12.2.5 Confidentiality . . . . .	65
12.2.6 Compliance . . . . .	66
12.2.7 Awareness . . . . .	66
12.3 Minacce alla privacy . . . . .	66
12.3.1 Information collection . . . . .	66
12.3.2 Information processing . . . . .	66
12.3.3 Information dissemination . . . . .	67
12.3.4 Invasion . . . . .	68
12.4 Privacy Enhancing Technologies (PETS) . . . . .	68
12.4.1 Data Protection technologies . . . . .	68
12.4.2 User awareness technologies . . . . .	69
12.4.3 Anonymity technologies . . . . .	69
12.4.4 Altre tecnologie . . . . .	69
<b>13 Introduzione alla data protection</b>	<b>71</b>
13.1 Dato personale . . . . .	72
13.2 Obblighi per data controller/data processor . . . . .	73
13.2.1 Lawfullnes . . . . .	73
13.2.2 Consenso . . . . .	74
13.2.3 Limitazione della finalità . . . . .	74
13.2.4 Minimizzazione dei dati . . . . .	74
13.2.5 Accuracy . . . . .	75
13.2.6 Storage Limitation . . . . .	75
13.2.7 Data Security . . . . .	75
13.2.8 Accountability . . . . .	75
13.2.9 Esempio . . . . .	76
13.3 Diritti dell'utente . . . . .	76
13.3.1 Trasparenza . . . . .	76
13.4 Riportare violazioni . . . . .	77
13.5 Sanzioni previste dal GDPR . . . . .	77
<b>14 Tecniche di anonimizzazione</b>	<b>79</b>
14.1 Tecniche di per proteggere gli identificatori esplicativi . . . . .	79
14.2 K-Anonymity . . . . .	80
14.3 L-diversity . . . . .	81
14.4 T-closeness . . . . .	83
14.5 Differential privacy . . . . .	83
14.6 Implementazione privacy differenziale . . . . .	84



# Capitolo 1

## Introduzione

Lo scopo della cybersecurity è di proteggere i servizi, i dispositivi, le applicazioni e le infrastrutture che usiamo giornalmente. Quindi qualsiasi cosa che usiamo rientra nella cybersecurity. Si differenzia dell'information security in quanto questa mira a proteggere solo le informazioni. La cybersecurity cerca di garantire:

- Confidentiality: mira a proteggere la risorsa da accessi non autorizzati. Principalmente si utilizzano, per garantirla, la crittografia e i meccanismi di controllo degli accessi;
- Integrity: può essere associata a software, dati, documenti, traffico di rete. Mira a proteggere da modifiche non autorizzate. Si utilizzano principalmente funzioni di hash per verificare possibili manomissioni;
- Availability: mira a garantire l'accesso delle risorse agli utenti legittimi. Si utilizza principalmente la ridondanza, installando il servizio su più server;
- Authenticity: mira a garantire che, nel caso dell'utente, l'utente sia che dice di essere. Si possono usare login con password o tramite dispositivi biometrici. Per i dati viene garantita tramite chiave;
- Accountability: si occupa di monitorare chi o cosa compie quali azioni all'interno del sistema. Si utilizzano principalmente i log;
- Safety: si assicura che le macchine non causino danno agli operatori che le utilizzano (es: attacco alle centrali elettriche in Ucraina).

### 1.1 Concetti fondamentali

**Asset** Un asset è qualsiasi cosa che ha valore per un'organizzazione (es: software prodotto, reputazione dell'azienda), un individuo (es: dati personali) o una nazione (es: infrastrutture per la distribuzione elettrica). L'asset può essere materiale o immateriale.

**Vulnerabilità** Una vulnerabilità è una debolezza che gli attaccanti possono usare per danneggiare un asset.

**Cyber threat** Una minaccia è una qualsiasi circostanza/evento che può potenzialmente impattare in modo negativo l'organizzazione.

**Attacco** Un attacco è la realizzazione di una specifica minaccia ad un asset.

**Threat Actor** L'attaccante è colui che esegue l'attacco.

**Rischio** Il rischio quantifica l'impatto che l'attacco avrebbe sull'organizzazione, nonché la probabilità che l'attacco venga effettuato contro l'organizzazione.

**Security controls** Le misure di sicurezza sono un insieme di meccanismi/regole messe in campo per poter proteggere l'asset.

## 1.2 Categorie di attaccanti

**Cybercriminali** sono principalmente spinti da motivazioni economiche. Si concentrano su due tipologie di informazioni: informazioni personali delle vittime (es: info sui clienti dell'azienda) oppure finanziarie (es: carte di credito, credenziali dei conti).

Usano generalmente:

- Attacchi malware, attuati principalmente con financial trojan, che sono malware che si concentrano sul recuperare le informazioni finanziarie che usano per effettuare transazioni bancarie non autorizzate;
- Attacchi ransomware, con cui con cifrano le macchine delle vittime e richiedono un riscatto in cambio della chiave di decifratura;
- Attacchi alle vulnerabilità dei sistemi;
- Attacchi DDoS con cui rendono non disponibili, ad esempio, dei servizi web e richiedono un riscatto per farli ritornare operativi.

Questi attacchi vengono generalmente attuati usando malware, email di phishing (contengono un link a un sito infetto o un allegato malevolo) o botnet (trasformano il dispositivo in uno zombi in attesa di comandi dell'attaccante. Negli attacchi DDoS vengono usati per generare elevato traffico contro il sito bersaglio). Generalmente i malware usati non vengono scritti dai cybercriminali, ma piuttosto comprati. Lo stesso vale per i siti host.

**Nation state** gruppi di hacker affiliati ad una potenza governativa mondiale. I loro obiettivi sono principalmente strutture critiche di altri paesi. Le motivazioni sono principalmente due: sabotare infrastrutture o influenzare opinioni politiche/religione nella nazione bersaglio.

Usano generalmente:

- Attacchi malware, per sabotare le strutture critiche;
- Campagne di phishing, per ottenere informazioni riservate;
- Attacchi DDoS, per non rendere disponibili le strutture critiche.

Questi attacchi vengono attuati tramite malware molto sofisticati e tecniche di evasione per non essere rilevati; utilizzano anche vulnerabilità non ancora note pubblicamente (zero-days vulnerabilities).

**Attivisti** sono motivati dalla volontà di imporre le proprie visioni religiose, politiche e sociali. Attaccano principalmente quelle organizzazioni non sono in linea con i loro ideali.

Usano generalmente:

- Web defacement, in cui modificano l'aspetto di un sito web (in cui di solito pubblicano un messaggio);
- Rilascio di info confidenziali;
- Attacchi DDoS.

Questi attacchi vengono attuati tramite botnet, email infette o vulnerabilità conosciute (exploit kit).

**Insider Threat** si tratta principalmente di persone che sono state licenziate dall'azienda, o comunque ci hanno avuto dei problemi, che, grazie ai permessi che possiedono, eseguono azioni ai danni dell'azienda stessa.

Possono:

- Rubare e vendere informazioni;
- Rilasciare informazioni pubblicamente;
- Installare bombe logiche.

Questa categoria comprende anche quei membri dell'azienda che provocano danni involontariamente, pubblicando, ad esempio, per errore file classificati oppure visitando un sito malevolo che andrà ad infettare il network dell'azienda.

### 1.3 Attacchi di tipo Supply chain

La supply chain si riferisce all'ecosistema di processi, persone, organizzazioni e distributori coinvolti nella creazione e consegna di un prodotto finale. Questo tipo di attacchi si sta diffondendo molto in quanto le aziende esternalizzano una parte (o tutto) il ciclo di vita del servizio o dell'applicazione che offre. Ad esempio un software sviluppato modularmente può fare uso di plug-in sviluppati da terzi, oppure l'infrastruttura su cui il software si appoggia appartiene a terzi (es: cloud). Gli attaccanti stanno sfruttando le vulnerabilità di questi elementi terzi per attaccare i fruitori del software/servizio fornito dall'azienda.

Gli elementi chiave della supply chain sono:

- Fornitore: fornisce il servizio/software/infrastruttura;
- Asset del fornitore: rappresenta elementi di valore che il fornitore usa per produrre il prodotto e che sono bersagli degli attacchi;
- Cliente: colui che usufruisce del prodotto del fornitore;
- Asset del cliente: elementi di valore posseduti dal cliente che sono l'obiettivo ultimo degli attacchi.

Gli attacchi di *supply chain* si compongono di due parti: una prima parte dove l'attaccante compromette uno o più asset del fornitore sfruttando una vulnerabilità e una seconda parte dove l'attaccante sfrutta gli asset infetti (e la fiducia che il cliente ha nel fornitore) per mettere in campo l'attacco vero e proprio e colpire l'asset del cliente.

Nella figura 1.1 sono riportati gli asset bersaglio del fornitore e i relativi attacchi, mentre nella figura 1.2 sono riportati gli asset del cliente e i relativi attacchi.

<b>Supplier Assets</b>	
	<b>Pre-existing Software</b> e.g. software used by the supplier, web servers, applications, databases, monitoring systems, cloud applications, firmware. It does not include software libraries.
	<b>Software Libraries</b> e.g. third party libraries, software packages installed from third parties such as npm, ruby, etc.
	<b>Code</b> e.g. source code or software produced by the supplier.
	<b>Configurations</b> e.g. passwords, API keys, firewall rules, URLs.
	<b>Data</b> e.g. information about the supplier, values from sensors, certificates, personal data of customers or suppliers themselves, personal data.
	<b>Processes</b> e.g. updates, backups or validation processes, signing certificates processes.
	<b>Hardware</b> e.g. hardware produced by the supplier, chips, valves, USBs.
	<b>People</b> e.g. targeted individuals with access to data, infrastructure, or to other people.

<b>Supplier Attack Techniques Used</b>	
	<b>Malware Infection</b> e.g. spyware used to steal credentials from employees.
	<b>Social Engineering</b> e.g. phishing, fake applications, typo-squatting, Wi-Fi impersonation, convincing the supplier to do something.
	<b>Brute-Force Attack</b> e.g. guessing an SSH password, guessing a web login.
	<b>Exploiting Software Vulnerability</b> e.g. SQL injection or buffer overflow exploit in an application.
	<b>Exploiting Configuration Vulnerability</b> e.g. taking advantage of a configuration problem.
	<b>Physical Attack or Modification</b> e.g. modify hardware, physical intrusion.
	<b>Open-Source Intelligence (OSINT)</b> e.g. search online for credentials, API keys, usernames.
	<b>Counterfeiting</b> e.g. imitation of USB with malicious purposes.

Figura 1.1: Fornitore

## Customer Assets

	<b>Data</b>	e.g. payment data, video feeds, documents, emails, flight plans, sales data and financial data, intellectual property.
	<b>Personal data</b>	e.g. customer data, employee records, credentials.
	<b>Software</b>	e.g. access to the customer product source code, modification of the software of the customer.
	<b>Processes</b>	e.g. documentation of internal processes of operation and configurations, insertion of new malicious processes, documents of schematics.
	<b>Bandwidth</b>	e.g. use the bandwidth for Distributed Denial of Service (DDoS), send SPAM or to infect others on a large scale.
	<b>Financial</b>	e.g. steal cryptocurrency, hijack bank accounts, money transfers.
	<b>People</b>	e.g. individuals targeted due their position or knowledge.

## Customer Attack Techniques

	<b>Trusted Relationship [T1199]</b>	e.g. trust a certificate, trust an automatic update, trust an automatic backup.
	<b>Drive-by Compromise [T1189]</b>	e.g. malicious scripts in a website to infect users with malware.
	<b>Phishing [T1566]</b>	e.g. messages impersonating the supplier, fake update notifications.
	<b>Malware Infection</b>	e.g. Remote Access Trojan (RAT), backdoor, ransomware.
	<b>Physical Attack or Modification</b>	e.g. modify hardware, physical intrusion.
	<b>Counterfeiting</b>	e.g. create a fake USB, modify a motherboard, impersonation of supplier's personnel.

Figura 1.2: Cliente

## 1.4 SolarWinds

L'attacco è stato scoperto nel 2021 da un'azienda da cybersecurity. Il tutto è partito con la compromissione di un software di nome Orion, prodotto da SolarWind. Viene usato per il management di rete. Ha la caratteristica di comunicare con qualsiasi dispositivo sulla rete, inviare loro comandi e raccogliere informazioni. L'attacco di attribuisce ad APT29, un gruppo di nation state associato al governo russo.

L'attacco ha compromesso il processo di aggiornamento del software, andando ad installare su tutte le macchine che lo usavano un aggiornamento malevolo. L'aggiornamento andava poi a installare un malware sulle macchine che andava a raccogliere dati sensitivi delle vittime.

L'attacco è stato portato avanti in più fasi:

1. Infiltrazione iniziale: gli attaccanti hanno raccolto info su SolarWinds e in particolare hanno fatto reconnaissance sfruttando una vulnerabilità del Microsoft Exchange Server (che gestisce le comunicazioni di posta elettronica). Grazie a questa sono riusciti ad accedere alla mail dei dipendenti e recuperare le credenziali di accesso. Da questo sono riusciti a monitorare le comunicazioni e capire come funzionava il sistema di aggiornamento di Orion;
2. Spear phishing: gli attaccanti hanno attuato una campagna di spear phishing contro i dipendenti che avevano accesso ai server usati per effettuare gli aggiornamenti e le loro distribuzioni. Le mail contenevano un allegato malevolo che andava a installare un malware sulle macchine. Questo malware andava a monitorare i processi in esecuzione sulle macchine dedicati alla compilazione degli aggiornamenti e al caricamento delle build sul server degli aggiornamenti. Una volta identificato il processo di aggiornamento, il malware è andato a copiare un secondo malware (la vera arma) in più file coinvolti nel processo di aggiornamento;
3. Weaponization: quando i clienti aggiornavano Orion, andavano a scaricare l'aggiornamento malevolo che conteneva una backdoor che dava accesso all'amministratore agli attaccanti. Il malware, una volta scaricato, restava dormiente per 2 settimane prima di tentare di comunicare con Command and Control server in attesa di comandi dagli attaccanti;
- 4.

# Capitolo 2

## Cyber Kill Chain

**Threat intelligence** Analisi dei possibili attacchi informatici che possono colpire l'organizzazione. Determinare le fasi di un attacco e le sue tecniche permette di fare una scelta mirata sulle misure di protezione da adottare per prevenirlo o ridurne l'impatto.

### 2.1 Fasi della cyber kill chain

La cyber kill chain, in generale, descrive le fasi in cui può essere suddiviso un attacco:

1. Reconnaissance: decidere e ottenere info sul target. La reconnaissance può essere:
  - Attiva: raccolgo info interagendo col target.
    - nmap per creare una mappa della rete dell'obiettivo (porte disponibili, porte aperte, servizi associati);
    - port scanning;
    - vulnerability scanner (es: nessus);
    - Linkedin per cercare ruoli specifici nell'organizzazione.
  - Passiva: raccolgo info senza interagire col soggetto.
    - whois permette di raccogliere info su un dominio o specifico IP: quando è stato creato; data rilascio e scadenza certificato; chi è l'amministratore di sistema; contatto tecnico per rinnovo dominio;
    - Shodan: porte aperte; servizi accessibili; lista vulnerabilità;
    - Social media;
    - mantego.
2. Weaponization: trovare o creare l'attacco che exploita una vulnerabilità. Posso usare:
  - Metasploit per creare un attacco che sfrutta vulnerabilità. Fornisce codice già pronto per sfruttare vulnerabilità;
  - Exploit DB;
  - Vell Framework;
  - Aircrack;
  - SQL map;
  - Cobal strike, tool legittimo a pagamento usato per penetration testing. Ha struttura modulare.

3. Deliver: decido in che modo consegnare l'exploit. Posso sfruttare: siti web, social media, email, USB, ...
4. Exploitation: exploito una vulnerabilità. Posso usare:
  - SQL injection;
  - Buffer overflow;
  - Malware;
  - Javascript hijacking;
  - User exploitation;
  - Compromissione sito legittimo (inserimento script o sfruttamento vulnerabilità framewok usati nel sito).
5. Installation: mantengo persistente l'attacco all'interno dell'ambiente. Posso usare:
  - DDL hijacking;
  - Meterpreter;
  - Remote Access Trojan;
  - Registry changes;
  - PowerShell commands.
6. Command and Control (C2): stabilisco un canale di C2 con l'attaccante per manipolare da remoto la vittima.
7. Actions and Objectives: eseguo azioni per raggiungere il goal originario. Posso:
  - Raccogliere credenziali utente;
  - Scalare privilegi;
  - Reconnaissance interna;
  - Movimento nell'ambiente;
  - Reccogliere dati;
  - Distruggere sistemi;
  - Riscrivere o corrompere dati;

## 2.2 Trickbot

Torjan avanzato che si diffonde principalmente tramite campagne di spearphishing usando mail che contengono allegati malevoli o link che eseguono sw malevoli. Gli attacchi recenti usano mail di phishing.

# Capitolo 3

## Panorama sui cyber threat

### 3.1 Overview 2022

- La maggior parte degli attacchi sono stati cybercrimini e sabotaggi/spionaggi;
- Tipologie principali attacchi: malware, phishing, attacchi web, sfruttamento vulnerabilità;
- Settori principalmente targetizzati: governo, salute, comunicazioni.

### 3.2 Malware

**Ransomware as a service** catena che si compone di più attori:

- Operatori: gang cybercriminali che dispongono delle capacità e risorse per creare e gestire ransomware e infrastrutture collegate;
- Affiliati: gang cybercriminali che affittano i ransomware dagli operatori;
- Access Broker: gang cybercriminali che ricercano info o vulnerabilità su organizzazioni target;

Se l'attacco dell'affiliato ha successo (infezione e riscatto), il guadagno viene condiviso con il relativo operatore.

#### Overview attacchi malware 2022

- Nell'ultimo anno il 66%
- Costo medio del riscatto: \$900.000;
- Il 46% delle aziende ha pagato il riscatto nonostante avesse dei backup;
- Il 61% delle paganti è riuscita a ripristinare i dati;
- Solo il 4% li ha ripristinati totalmente.

### 3.3 Trend recenti negli attacchi malware

**Tecnica Double Extortion** Oltre alla richiesta di riscatto per ottenere la chiave di de-cifratura usata per cifrare i dati, la vittima viene minacciata con la diffusione di tali dati (preventivamente copiati dall'attaccante) nel caso di mancato pagamento (tecnica double extortion).

**Tecnica Triple Extortion** Funziona come la double extortion, solo che nel caso in cui l'azienda si rifiuti di pagare, gli attaccanti si rivolgono ai suoi clienti.

Una delle vittime di questo tipo di attacco è stata un'azienda finlandese di consulenza psichiatrica. Gli attaccanti hanno minacciato i clienti di pubblicare le loro conversazioni private con gli psichiatri in caso di mancato pagamento del riscatto.

**Intermittent Encryption** Tecnica che velocizza il tempo di infezione andando a cifrare solo alcuni byte/blocchi dei file. Tende ad evadere le analisi antimalware in quanto fa un uso limitato delle operazioni di lettura/scrittura su file.

**Lockbit 3.0** Aggiornamento del malware Lockbit che introduce un programma di bug hunting con ricompensa.

**BlackCat** Scritto in Rust, è uno dei primi ad implementare la crittografia intermittente. Offre features per pubblicare i dati e ricercare info sul target.

**Attacco alla Ferrari** Attacco di supply chain o ransomware, non ancora verificato. Sono stati rubati 7GB di dati, contenenti progetti e manuali tecnici.

**Altri attacchi** Sono stati attaccati il Gestore Servizi Energetici e l'università di Pisa.

**Urnsif** Banking Trojan (malware che ruba credenziali dell'home banking o dati della carta di credito) che usa attacchi di phishing tramite mail, le quali contengono un documento office con macro che scarica Ursnif. Quando l'utente si collega all'home banking, il malware si attiva lanciando un alert di infezione che invita a scaricare una nuova app.

**Attacco alla Colonial Pipeline** Effettuato ai danni della Colonial Pipeline, ha reso inutilizzabili i suoi sistemi di distribuzione del carburante. Questo attacco può essere inteso come un attacco alle strutture critiche (blocco trasporti, blocco attività, ...).

**Nuove tecniche di evasione** Nel ransomware usato nell'attacco alla WastedLocker, creato in modo specifico per questo attacco (conosceva il nome dei file da cercare), è stata implementata una nuova tecnica per evitare la rilevazione da parte del sistema. In generale, molti sistemi anti-malware vanno a ricercare nei programmi il richiamo a specifiche api di Windows (es: apertura file, lettura, cifratura). Per evitare questo problema, il malware ha utilizzato il Windows Cache Manager. Invece di cifrare i dati direttamente su disco, e venire quindi rilevato, ha usato questo manager per caricarli in memoria e cifrarli qui, per poi risalvarli su disco.

**Botnet** Usati per attacchi DDoS e per distribuire altri malware. I principali sono Emotet e Trickbot. Emotet nasce inizialmente nel 2016 come Financial Trojan, per poi evolversi come mezzo per scaricare altri malware. L'anno scorso, grazie ad un'operazione dell'Europol, ha perso terreno. A novembre è però risorto.

**Malware cellulari** Tra i più famosi c'è Pegasus, spyware usato da molti governi per controllare personaggi di interesse. MasterFred, altro malware mobile, invece va a costruire sopra le interfacce legittime di pagamento di varie app (es: Netflix) delle interfacce fasulle, sfruttando le impostazioni di accessibilità di Android.

**FluBot** Si tratta di un malware Android/IOS, che ha come obiettivo i dati relativi alle carte di credito, diffusosi molto tramite sms contenente i link infetti. Generalmente il messaggio si spacciava come proveniente da Un'azienda di trasporti (es: DHL) che richiedeva di installare l'app allegata per facilitare/velocizzare la consegna.

### 3.4 Attacchi comuni

**Attacchi di phishing (ultimi 6 mesi)** Targetizzati INPS e BPER Group (banca).

**Smiching** Attacchi di ingegneria sociale tramite sms.

**Vshing** Voice phishing -> nella mail di phishing non metto un link ma un numero di telefono da contattare (es: contattare servizio clienti per tentato accesso all'account paypal).

**EvilProxy** Permette di superare le autenticazioni a due fattori eseguendo un attacco di *man in the middle*.

**Compromissione di siti legittimi** Hijacking di siti legittimi;

**Uso di HTTPS** Ora anche i siti malevoli potrebbero avere una connessione HTTPS, quindi la presenza di certificato non assicura la legittimità del sito.

### 3.5 Attacchi cloud

**Vulnerabilità OMIGOD** I ricercatori di sicurezza hanno rivelato i dettagli di quattro vulnerabilità, note collettivamente come OMIGOD, che interessano lo strumento Open Management Infrastructure (OMI) di Microsoft. Sostengono che un utente malintenzionato remoto non autenticato può sfruttare alcune o tutte queste vulnerabilità per ottenere l'accesso amministrativo all'ambiente virtuale Linux in esecuzione sul servizio di cloud computing Azure di Microsoft.

**ChaosDB** ChaosDB è una vulnerabilità critica nel servizio di database Azure Cosmos DB di Microsoft. Un utente malintenzionato potrebbe sfruttarlo per ottenere l'accesso in lettura/scrittura alle informazioni del database di altri utenti, nonché all'infrastruttura di hosting di Azure sottostante.

### 3.6 Attacchi a dispositivi IoT

La maggior parte dei dispositivi IoT è vulnerabile ad attacchi di media/alta gravità. Questo è dovuto a quattro fattori principali:

1. Il traffico dati non è criptato;
2. Molto spesso gli utenti non cambiano password e continuano ad usare quella di default;
3. I sistemi operativi usati (principalmente relativi al controllo industriale) sono spesso non aggiornati;
4. In generale hanno bassa memoria, capacità computazionale, spazio per salvare e potenza

**BadAlloc** Una ricerca di Microsoft ha scoperto che molti dei dispositivi che usano sistemi embedded o real time scritti in C sono facile preda di attacchi di Buffer Overflow (C è noto per le vulnerabilità legate alla gestione della memoria). Il problema è dovuto alle librerie C di allocazione dinamica della memoria (heap): se alla malloc si passa un integer che causa Integer Overflow (numero negativo o troppo grande), l'attaccante può sovrascrivere la memoria allocata dalla malloc ed eseguire codice malevolo.

**Classe di vulnerabilità Ripple20** è stato scoperto che questa libreria, implementata in molti dispositivi, sono presenti 19 vulnerabilità, di cui 4 molto gravi.

**Telecamere di videosorveglianza** Molto spesso nei sistemi di videosorveglianza non sono implementati sistemi di autenticazione o, se lo sono, utilizzano sistemi deboli (es: password).

Sul sito <https://www.shodan.io/> è possibile ricercare tutti i dispositivi IoT collegati alla rete e ricavarne informazioni. Può essere usato per studiare attacchi mirati.

**\*Malware Mirai** Da varie ricerche si è scoperta che telecamere e stampanti sono i dispositivi IoT meno sicuri. Mirai è un malware progettato per operare su dispositivi connessi a Internet, specialmente dispositivi IoT, rendendoli parte di una botnet che può essere usata per attacchi informatici su larga scala. Scansiona continuamente la rete alla ricerca di dispositivi accessibili protetti da password di default o molto comuni (es: abc123, admin123, 1234, password, ...).

**Vulnerabilità nei dispositivi healthcare** Il settore della healthcare è tra quelli più vulnerabili ad attacchi informatici. Tra i dispositivi più a rischio troviamo i sistemi di imaging (es: radiografie) e di monitoraggio dei pazienti. Ad esempio, Medtronic è stata costretta a ritirare dal mercato diverse pompe di insulina a causa di una vulnerabilità che consentiva di modificare i valori della pompa (rischio coma glicemico).

## Capitolo 4

# Social engineering

L'ingegneria sociale è la manipolazione psicologica delle persone per far si che eseguano azioni o divulgino informazioni personali. Sfrutta le vulnerabilità insite nelle persone.

Secondo Kevin Mitnick, uno degli hacker più famosi al mondo, anche se l'organizzazione investe pesantemente in sistemi di sicurezza, se l'attaccante riesce a fare quello che vuole anche a una sola persona, tutto l'investimento fatto è inutile.

Alcuni cyber criminali che usano l'ingegneria sociale sono:

- Hacker: attacchi DDoS, ransomware, attacchi finanziari;
- Identity thieves: usano info che identificano la persona per i loro obiettivi (e poi possono rivenderle/pubblicarle online);
- Scam artists: eseguono aziende fraudolente o ingannevoli per frodare altri (es: lotteria scam su facebook).

**Attacco alla banca del Bangladesh** Nel 2016 la banca del Bangladesh è stata vittima di una delle più grandi rapine informatiche. L'attacco è iniziato nel 2015, quando gli attaccanti sono riusciti ad installare un malware utilizzando un file malevolo spacciato per curriculum per una delle posizioni aperte nella banca. Lo scopo del malware era di eseguire una serie di trasferimenti verso una società no-profit estera. L'attacco è stato sventato dalla Federeal Reserve, dalla quale passano i trasferimenti internazionali, che si è accorta dell'attacco grazie ad un errore di spelling nel nome dell'associazione.

**Attacco a Macron** Caso di Identity Steal. Due francesi sono riusciti ad accedere alla mail di Macron, dalla quale hanno inviato mail che descrivevano i 10 motivi per cui non votare Macron.

**Lotteria su Facebook** La persona viene contattata tramite Facebook informandola della vincita ad una lotteria e che l'assegno le verrà recapitato a casa. Alla consegna, alla vittima viene chiesto un pagamento per ricevere l'assegno.

### 4.1 Ciclo di vita dell'attacco

1. Selezione del target;
2. Reconnaissance;

3. Raccolta delle info: posso cercare nella spazzatura della vittima ("dumpster diving" -> cerco bollette, stampe conto corrente, post-it con login e password, ...) o la osservo standone in prossimità ("shoulder surfing");
4. Orchestrione dell'attacco;
5. Esecuzione dell'attacco: sfrutto le info e le relazioni per spingere la vittima a compiere le azioni che voglio;
6. Impatto dell'attacco: raggiungo il mio obiettivo (es: installazione malware, eliminazione tracce attacco sulla macchina della vittima, ...).

Possibili modi per ottenere informazioni: osservare alla spalle il target (es: mentre la persona sta lavorando al pc), cercare nella spazzatura del target.

### Esempio di attacco

- Obiettivo: ottenere dalla vittima le info necessarie ad eseguire un trasferimento di denaro;
- Selezione del target, reconnaissance, raccolte info: ad esempio verifico online se il target è stato vittima di attacchi di data breach;
- Attack orchestration: ad esempio contatto la vittima tramite mail+telefono o sms+telefono;
- Esecuzione dell'attacco:
  1. Creo lo stage per l'attacco, creando ad esempio un sito web di phishing che simula il sito web di autenticazione della banca della vittima;
  2. Creo un app malevola da far installare alla vittima, la quale mi girerà i codici ricevuti necessaria a superare l'autenticazione a più fattori;
  3. Invio una mail alla vittima con il link alla pagina web;
  4. Contatto al vittima per telefono e la convinco a installare l'app. Per incrementare le possibilità di successo, posso camuffare il numero di telefono usando tecniche di spoofing facendo comparire il numero di telefono della banca;
  5. Se tutto ha successo, ottengo accesso all'home banking della vittima.
- Impatto dell'attacco: Elimino ogni traccia dell'attacco (es: l'app si rimuove terminato il suo lavoro).

## 4.2 Tipi di attacchi

- Phishing: condotto generalmente via mail, cerca di attaccare più persone possibili. Generalmente l'attaccante si finge una persona di alto rango o richiede che un'azione venga svolta entro un breve limite di tempo;
- Spear phishing: mira ad una persona particolare;
- Whaling: mira a persone con elevate responsabilità nell'organizzazione;
- Viral hoaxe: mira a divulgare info false e quindi influenzare l'opinione delle persone. Sfrutta la curiosità delle persone (diffuso nei social, dove il post chiede di aprire il link per avere più info sull'argomento);

- Vphishing: usa chiamate telefoniche;
- Impersonation e tailgating: fatti di persona;
- SMiShing attacco tramite SMS.

**Impersonation** Solitamente mi fingo qualcuno con autorità, qualcuno con seniority o un ente governativo.

**Vishing** Un esempio di attacco è stato quello del falso servizio clienti di Apple. L'attaccante chiamava la vittima fingendosi il servizio clienti di Apple dicendo che c'era un problema con l'account Appl. Quando la vittima richiamava per risolvere, l'attaccante chiedeva i dati di accesso dell'account.

**Tailgating** Sfrutta l'indole delle persone ad aiutare gli altri (es: mi fingo un dipendente della banca che ha dimenticato il tesserino di accesso l'accesso, mi accodo ad un dipendente e chiedo se posso passare con lui; mi fingo un dipendente dell'azienda che ha versato il caffè sopra dei documenti e chiedo di ristamparli. I documenti sono in una chiavetta che installa poi il malware).

### 4.3 Attacchi di phishing

Il phishing è il tentativo di acquisire informazioni sensitive come username, password e dettagli di carte di credito (e a volte, indirettamente, denaro), spesso con scopi malevoli, spacciandosi per un'entità di fiducia in comunicazioni elettroniche.

Dati:

- L'88% delle organizzazioni mondiali è stata vittima di un attacco di spear phishing nel 2019;
- Il 95% degli attacchi alle reti aziendali deriva da un attacco andato a buon fine di spear phishing;
- Il 22% delle violazioni dei dati nel 2019 ha riguardato il phishing;
- Il 97% degli utenti non è in grado di riconoscere mail di phishing sofisticate;
- Il 30% delle mail di phishing vengono aperte;
- Nel 12% di queste, il link viene clickato;
- Il 15% delle vittime viene targetizzato da almeno un altro attacco nello stesso anno.

**Attacchi che sfruttano la pandemia** Durante la pandemia Google ha bloccato milioni di mail che proponevano un raccolta fondi per finanziare la ricerca di un vaccino.

**Possibili indicatori che la mail è di phishing :**

- Non mi stavo aspettando quella mail con quel contenuto;
- Errori grammaticali nel testo o nell'oggetto;
- Indirizzo del mittente o del CC strano;
- Link effettivo (che vedo muovendoci sopra il mouse) diverso da quello scritto;
- Assenza della grafica classica di quel tipo di mail;

**Attenzione:** l'anteprima della mail dal client solitamente mostra solo il nome del mittente e non anche il suo indirizzo; molto spesso i link presenti nella mail di phishing usano servizi di URL shortening.

**PhishTank:** sito web che, dato il link ad una pagina web, questo rientri tra la lista dei siti conosciuti di phishing.

#### 4.4 Attacchi di spearphishing

Un attacco di spearphishing è un attacco di phishing lanciato specificamente contro un'organizzazione o un utente target, spesso adattato alla vittima rappresentando informazioni uniche per loro al fine di creare autenticità.

#### 4.5 Tattiche di influenza

Posso "usare":

- Autorità: assumo una posizione di autorità rispetto alla vittima (manager, ceo, direttore risorse umane, ente governativo);
- Scarsità: una risorsa limitatamente disponibile spinge la vittima ad agire con velocità;
- Commitment: le persona agiscono in maniera consistente. Se una persona ha svolto un'azione in passato, è probabile che la rifaccia. Quindi posso prima chiedere alla vittima informazioni poco sensitive per poi passare a quelle che i interessano (es: prima chiedo il nome del cane e poi la risposta alle domande per recuperare il codice di un account);
- Liking: le persone tendono ad eseguire un'azione se viene chiesta da persone che conoscono;
- Reciprocità: se alla vittima viene offerto qualcosa è più probabile che esegua l'azione (es: per ricevere il premio è necessario aprire il link);
- Social proof: la vittima tende a svolgere l'azione se altre persone l'hanno già svolta.

#### 4.6 Come un'organizzazione può fermare gli attacchi di phishing

In figura 4.1 è riportato come un'organizzazione può fermare gli attacchi di phishing.

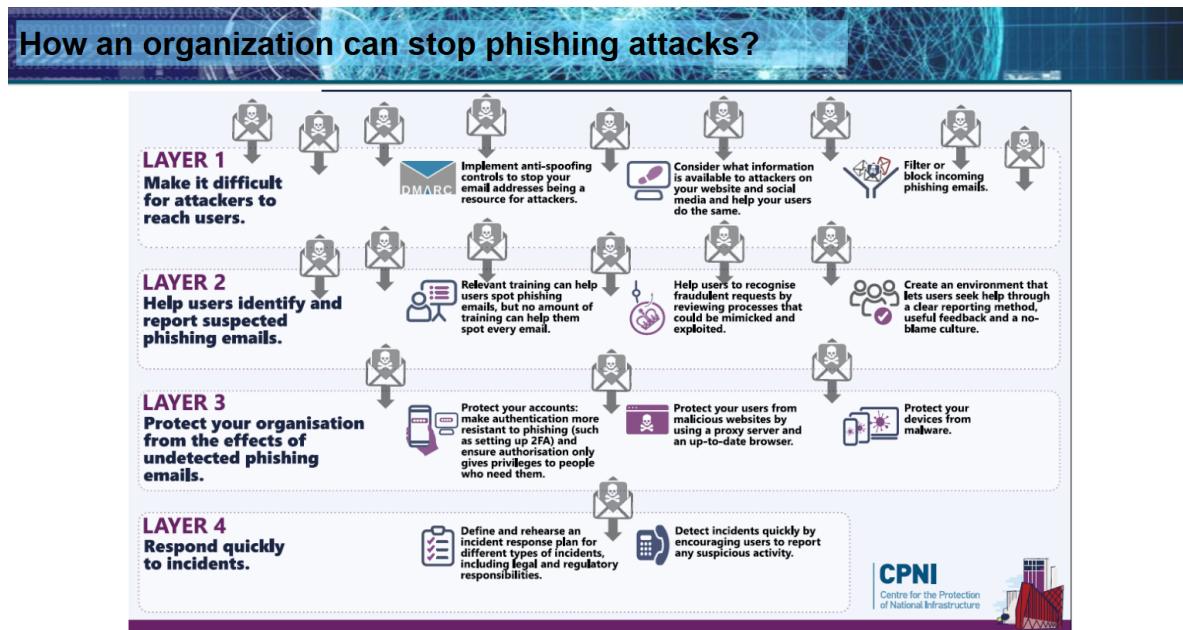


Figura 4.1: Come fermare gli attacchi di phishing



# Capitolo 5

## Malware

Un malware (contrazione di malicious + software) è un software malevolo che compie azioni che mirano a compromettere la riservatezza, disponibilità o l'integrità dei dati o dei sistemi infettati.

I sistemi possono venire infettati tramite:

- Accesso diretto al sistema con dischi/chiavette USB infetti;
- Attacchi di ingegneria sociale;
- Campagne di phishing;
- Visitando siti malevoli.

### 5.1 Tipi di malware

**Virus** Modifica/altera/compromette file o sw presenti sulla macchina della vittima. Ha la capacità di riprodursi, infettando le altre macchine presenti nella stessa rete. Richiede però l'azione umana per essere eseguito (link email, sito malevolo, apertura documenti malevoli). Questo tipo di malware è facilmente identificabile dagli antivirus, in quanto i virus hanno una specifica signature riconoscibile. I tipi di virus principali sono tre:

1. Macro virus, codificati come una serie di comandi inseriti in una macro all'interno del file malevolo (pdf, doc, ...). Non c'è quindi un eseguibile vero e proprio, ma si compone di comandi scritti in un linguaggio di scripting (es: VBA, powershell, ...);
2. Virus polimorfici, che cambiano il loro comportamento a seconda dell'OS dove vengono eseguiti o a seconda dei sw presenti nella macchina;
3. Companion virus, i più insidiosi in quanto si mascherano come sw legittimo (aggiornamento sw o sw comunemente presenti nelle macchine windows).

**Worm** Simile al virus, ha l'obiettivo di permettere all'attaccante di ottenere il controllo della macchina. Questo viene solitamente fatto installando backdoor. Ottenuto il controllo, l'attaccante può rubare i dati presenti. Si può replicare nelle altre macchine presenti nella rete. Non necessitano di azione umana in quanto sfruttano vulnerabilità presenti nella macchina (OS, protocolli, ...).

**Trojans** Tra le tipologie più diffuse. Nato come sw malevolo che accede ad info sensitive sulla macchina della vittima (credenziali, info finanziarie, ...), è diventato nel tempo uno strumento usato per mantenere il controllo della macchina della vittima (installando una backdoor). Una volta ottenuto il controllo, solitamente vengono sfruttati per installare altri malware;

**Rootkit** Solitamente installato nel kernel (che si interfaccia tra le componenti sw e hw della macchina), monitora tutte le chiamate di funzioni a libreria effettuate dal sw in esecuzione permettendo di mascherare la presenza di altri malware nella macchina della vittima andando ad intercettare le chiamate a Windows API effettuate dagli altri malware. Offre anche funzionalità di accesso come amministratore/root alla macchina.

Esistono tre categorie principali di Rootkit:

1. Quelli a livello applicativo;
2. Quelli a livello del kernel (kernel rootkit);
3. Quelli installati nel master boot record.

I primi, che vengono mascherati come sw legittimo, sono facili da rimuovere (dagli antivirus), i restanti sono più difficili in quanto vanno a compromettere l'OS. Per eliminarli solitamente è necessario reinstallare il sistema.

**Dropper/downloader** Il dropper è un malware che contiene al suo interno il vero malware da infettare. Tipicamente prende la forma di macro incluse all'interno di allegati malevoli. Quando il documento viene aperto, la macro viene eseguita e il malware estratto.

Il downloader, invece, non include il malware al suo interno, va a scaricarlo da un sito malevolo.

**Keylogger** Malware che viene solitamente installato dal altri malware. Cattura tutti i caratteri digitati su tastiera, li salva in un file nascosto e periodicamente manda una copia del file all'attaccante (mail o command & control).

**Bot** Pone la macchina infetta sotto il controllo dell'attaccante. Solitamente vengono posto sotto il controllo di un pc chiamato Bot Master, che invia comandi agli zombi. Tipicamente questo viene utilizzato per creare reti di bot (botnet) per effettuare attacchi DDoS. Uno scenario tipico è quello dove il bot master impedisce il comando ping verso uno specifico target, fintanto che l'obiettivo non è più in grado di rispondere.

**Cripto Miner** Utilizza le macchine infette per minare criptovalute e trasferirle nel wallet dell'attaccante. Molti di questi usano software di mining open source.

**Ransomware** Cifra i file nella macchine della vittima. La chiave di decifratura è fornita a fronte di un pagamento.

## 5.2 Prevenzione e riduzione dei danni

### Come posso prevenire la ricezione di malware

- Uso un sw di antispam e che vada ad analizzare il contenuto delle mail (link e file);
- Uso security gateway che vanno ad ispezionare il contenuto di alcuni dei protocolli di rete, come quello smtp, per cercare se sono presenti malware noti;

- Blocco l'accesso a siti potenzialmente malevoli a livello del browser con specifici plug-in, che si appoggiano ad una lista di url in cui è sicuro navigare;

### Come posso prevenire la diffusione del malware alle altre macchine della rete

- Mantengo l'OS aggiornato per impedire al malware di usare vulnerabilità ora patchate;
- Prevengo l'accesso alle credenziali dell'amministratore o di specifici utenti, utilizzando sistemi di autenticazione MFA (es: oltre a utente e password richiedo una One Time Password che arriva tramite sms);
- Limito i privilegi degli utenti che hanno accesso alle macchine (solitamente gli amministratori di sistema concedono più permessi del necessario), per limitare le azioni che il malware può effettuare;
- Faccio usare all'amministratore due account differenti: il primo per gestire la parte social (email, comunicazioni, ...), e che è quindi più soggetto a campagne di phishing, e il secondo, con utente e password diversi dal primo, per gestire la rete.

### Educazione dei dipendenti

- Educo sulle tecniche di ingegneria sociale usare per diffondere i malware;
- Educo sulle tipologie di malware esistenti;
- Educo sui rischi che comporta un'infezione per l'organizzazione e su come limitare i danni;
- L'educazione deve essere fatta in maniera continua, per stare al passo con l'evoluzione tecnologica.

### Backup regolari

- Mantengo più copie dei file e sw utilizzando diversi meccanismi (es: cloud) e/o dispositivi (es: disco esterno alla rete aziendale). I backup devono essere effettuati offline e tenuti in posizioni separate dalla rete/sistema dell'azienda. Per sicurezza effettuo più copie;
- Mi assicuro che i dispositivi contenenti il backup non devono essere lasciati collegati perennemente alla rete;
- Mi assicuro che i backup sono connessi solo a dispositivi puliti prima di iniziare il ripristino. Per sicurezza, scansione i backup alla ricerca di malware prima di iniziare il ripristino.
- Mantengo aggiornati i programmi di backup.

### Ripristino in seguito ad attacco malware

- Sconnetto tutti i dispositivi infetti dalla rete;
- Spengo il Wi-Fi e disabilito ogni connessione importante della rete (es: switch);
- Formatto i dispositivi in modo sicuro e reinstallo l'OS;
- Scansiono il backup alla ricerca di malware;

- Ricollego il dispositivo ripristinato ad una rete pulita ed eseguo le installazioni/aggiornamenti necessari;
- Installo, aggiorno e avvio un antivirus;
- Riconnetto il dispositivo alla rete dove era precedentemente collegato;
- Controllo il traffico di rete e scansiono alla ricerche di infezioni rimanenti.

### 5.3 Ransomware

Esistono molte famiglie di ransomware, che differiscono per le tecniche che implementano nelle varie fasi della cyber kill chain di un attacco ransomware. Possono colpire tutti i sistemi operativi.

#### Tipi

- Encryption malware: cifrano i dati;
- Locker: bloccano l'interfaccia utente con cui l'utente ha accesso all'OS e mostrano una schermata in cui si richiede un riscatto in cambio dello sblocco;
- Master Boot Record ransomware: cifra il master boot record o lo modifica in modo che l'os non sia più caricabile;
- Wiper: cancella i file presenti sulla macchina, senza possibilità che la vittima li recuperi.

#### Componenti principali di un ransomware

- Componente trojan: ha il compito di far arrivare il ransomware sulla macchia della vittima. Può implementare tre tipi di consegna:
  1. Tramite email di phishing con allegato malevolo;
  2. Tramite Exploit Kit che sfrutta vulnerabilità dell'OS o del sw;
  3. Tramite la compromissione di siti legittimi;
- Componente di cifratura/decifratura: generalmente vengono usati due tipi di cifratura:
  1. Simmetrica per cifrare i dati della vittima (la cifratura simmetrica è veloce);
  2. A chiave pubblica (asimmetrica) per cifrare la chiave simmetrica. Una volta cifrata viene inviata all'attaccante, che possiede la chiave privata.
- Routin di estrazione della chiave: estraе la chiave utilizzata per cifrare i dati e la invia all'attaccante (solitamente la chiave viene generata nella macchina infetta);
- Interfaccia con l'utente: presenta le istruzioni per pagare il riscatto.

### Fasi della cifratura

- Genero la chiave simmetrica -> può essere generata al momento nelle macchine, inviata dall'attaccante tramite il command and control server o codificata nel codice del ransomware (soluzione usata agli inizi);
- Cifro i dati con la chiave simmetrica;
- Cifro la chiave simmetrica con la chiave pubblica e la invio all'attaccante;
- Cancello la chiave simmetrica dalla macchina della vittima.

I primi ransomware usavano tecniche di cifratura primitive o deboli:

- Analizzando il codice del malware è possibile risalire alla chiave codificata nel codice;
- La cifratura con XOR e RC4 sono deboli ad attacchi di forza bruta.

#### 5.3.1 Killer switch

Si tratta condizioni implementate dagli sviluppatori del malware per stoppare il loro funzionamento (possono essere state lasciate all'interno per errore).

**Kill Switch per WannaCry** Il funzionamento di questo ransomware era legato alla registrazione di uno specifico domain name. Il ransomware andava inizialmente a controllare se il dominio era attivo, tramite DNS request, e, se riceveva l'indirizzo IP come risposta, andava ad effettuare una richiesta HTTP al dominio. Se riceveva anche qui risposta, bloccava la sua esecuzione. Registrando il dominio è stato possibile bloccare globalmente il malware.

**Killer Switch per Bad Rabbit** L'obiettivo di questo malware era distruggere il Master Boot Record della macchina. Si è scoperto che se il malware non riusciva a scrivere su disco il file *C:\Windows\infpub.bat*, la cifratura veniva stoppata. Questo killer switch non ne bloccava però la propagazione sulla rete.

#### 5.3.2 Cyber kill chain di un attacco ransomware

1. La vittima riceve una mail di phishing contenente un link ad un sito malevolo. La vittima visita il sito;
2. Il server web, che sta hostando un exploit kit, analizza la macchina della vittima alla ricerca di vulnerabilità;
3. Sfruttando le vulnerabilità trovate, il ransomware viene consegnato alla macchina ed si esegue;
4. L'eseguibile elimina le eventuali shadow copy presenti nel pc (introdotte da windows, sono copie di backup di file) e si propaga nel file system;
5. Il ransomware ricerca file con specifiche estensioni e li cifra;
6. Il ransomware contatta il C&C per inviare all'attaccante la chiave di cifratura e info sulla macchina;
7. Il ransomware riceve info sul pagamento dal C&C;
8. Il ransomware mostra le info sul pagamento all'utente;

- Se l'utente paga, il ransomware contatta il C&C per riottenere la chiave per decifrare la chiave usata per cifrare. Non è detta che la chiave venga fornita;
- Se l'utente non paga entro il limite di tempo la chiave viene cancellata.

### Fase di weaponization

- Ransomware basato su script-> inserisco in un allegato una macro che carica in memoria lo script malevolo. Difficile da individuare da parte degli antivirus;
- Diversificazione del payload: nascondo il payload malevolo in file con differenti formati (es: odt, PDF, SVG, dll, ...). I client di posta bloccano mail contenenti eseguibili, quindi inserisco i ransomware all'interno di altri file;
- Diversifico i pattern di accesso ai file: il classico pattern apertura file-cifratura-salvataggio file è facilmente individuabile dagli antivirus. Per evitare di essere scoperto posso modificare l'estensione del file prima di salvarlo oppure posso rimuovere i file eliminando le info associate ai file, che sono salvate nella MFT (Master File Table), presente in tutti i file system NTFS (in questo modo posso rimuovere l'accesso al file senza doverlo cifrare).

### Fase di evasione

Tecniche per rendere più difficile il riconoscimento e/o l'analisi:

- Time-based evasion techniques: per rallentare l'analisi o la scoperta del malware posso fare in modo che cifri i file solo in risposta a determinati eventi nel pc oppure inserire dei periodi di sleep nella fase di cripitura;
- Data evasion techniques: eliminano le tracce del malware dal pc della vittima (es: file di config, log, ...). Per evitare l'analisi posso usare anche le tecniche di anti-dump: generalmente per analizzare in un malware si aspetta che venga caricato tutto in memoria e poi, con sw specifici, viene scaricato. Tento di impedire questa tecnica per bloccare l'analisi.
- Code evasion techniques: prevedono tecniche di anti-debugging, anti-disassembling (es: cifro il ransomware, aggiungo offuscamento impacchetto il malware per evitare il disassemblaggio per la generazione del codice macchina) e anti-sandboxing (es: blocco o modifco il comportamento del malware se eseguito su una macchina virtuale -> controllo il mac address per capire se sono in un macchina virtuale in quanto vmware e virtualbox usano pattern specifici) per evitare l'analisi
- Network evasion techniques: cifra, anonimizzano il traffico (usando reti anonime) o cambiano l'indirizzo IP del C&C (domain shadowing);

### Fase di delivery

- Phishing;
- Spearphishing;
- Malvertisement: la vittima clicca su un link pubblicitario e viene rediretta ad un sito infetto che hosta il ransomware o verso un exploit kit che cerca vulnerabilità nel sistema;
- Sistemi di distribuzione del traffico: redirigono il traffico di un sito web legittimo verso uno malevolo che hosta un malware drive-by-download.

### Fase di exploitation

- Exploit kit;
- Vulnerabilità del target (vulnerabilità zero-days e vulnerabilità trovate durante la riconoscizione).

### Fase di installazione

- Rendere i file della vittima inaccessibili (cifro i file, li zippo in un archivio protetto da chiave, cifro il master boot record, distruggo i backup);
- Diffusione attraverso la rete.

**Fase di command and control (C2)** Durante la fase di command and control si ha la comunicazione tra il C2 e il ransomware. La comunicazione può avvenire in due momenti dell'infezione:

- Prima del processo di cifrature, se la chiave non viene generata sulla macchina della vittima, ma sul C2;
- Dopo che è terminato il processo di cifratura, per comunicare all'attaccante il chiave di cifratura e ricevere dal C2 le info di pagamento. Parte fondamentale di questa parte è la comunicazione col C2:
- Nei ransomware più *naive* (== ingenui), gli IP dei C2 sono solitamente presenti come una lista nel codice dei ransomware -> posso risalire e blacklistare questi indirizzi /domini;
- Nei ransomware più "furbi", viene implementato un Domain General Algorithm, che prevede che ransomware e C2 si mettano d'accordo sull'algoritmo da usare per generare il nome dei domini associati ai C2. In questo modo il nome viene generato dinamicamente ogni volta che il C2 viene contattato, rendendo quindi impossibile blacklistare il dominio.
- Si può anche utilizzare un bot di un botnet esistente come C2.

**Fase di actions and objectives** Il ransomware raggiunge il suo obiettivo. Nel caso di un encryption ransomware l'obiettivo è di portare le vittime a pagare il riscatto. Questo può essere fatto:

- Dando direttamente l'IBAN/paypal account alla vittima (strategia molto ingenua);
- Richiedendo pagamenti in criptovalute;
- Creando portali dedicati per supportare l'operazione di pagamento;

In alcuni ransomware sono presenti siti ospitati sulla rete Thor (garantisce l'anomimia) che fungono da servizio clienti del pagamento del riscatto.

## 5.4 Wanna Cry

Sfrutta una vulnerabilità del protocollo SMB usato da Windows per condividere file e accedere a dispositivi come le stampanti. Questa vulnerabilità consentiva all'attaccante di eseguire codice in remoto sulla macchina. Per sfruttare questa vulnerabilità è stato usato l'EternalBlue exploit kit.

Una volta arrivato su una macchina infetta che implementa il protocollo SMB, utilizza EternalBlue per installare una backdoor sulle macchine che sono collegate alla macchina infettata e che utilizzano anch'esse il protocollo SMB per condividere file con la macchina. Tramite la backdoor, l'attaccante va ad installare una copia di Wanna Cry e lo esegue. Un altro modo utilizzabile per arrivare sulle macchie è sfruttare tecniche di ingegneria sociale per far pluggare delle chiavette USB infette col ransomware nelle macchine delle vittime. L'esecuzione avviene sfruttando la funzionalità di autorun: basta modificare il file di autorun della chiavetta per fare eseguire in automatico il ransomware.

Un altro modo ancora consiste nel sfruttare la funzione di file share, usati nelle organizzazioni per effettuare il backup dei dati. Di solito è presente un server, usato per la condivisione dei file, a cui si connettono tutte le macchine dell'organizzazione. Il pc infetto carica sul server dedicato al file sharing una copia del ransomware e, quando le altre macchine si connetteranno, andranno a scaricare la copia caricata. Una variante di questa tecnica consiste nel non scaricare il file stesso, ma creare dei link al file. Quando viene creato un link (es: link su desktop ad un file per renderlo più facilmente accessibile), viene creato un file LNK. All'interno di questi file è possibile includere dei comandi, che vengono eseguiti quando si clicca sul file LNK. Così basta inserire il comando per eseguire il ransomware sulla macchina dove è stato copiato il file LNK.

## 5.5 Prevenzione specifica per gli attacchi ransomware

Molti consigli sono gli stessi visti per gli attacchi di ingegneria sociale.

## Capitolo 6

# Cyber War e attacchi ad infrastrutture critiche

La cyber war è una guerra condotta tramite reti e computer. Di solito coinvolge due paesi, dove l'attaccante impiega un gruppo di hacker per sabotare strutture critiche dell'altro paese tramite cyber weapons (solitamente malware molto specializzati).

Un gruppo di hacker nord coreano che effettua un attacco DDoS a uno dei maggiori provider telefonici americani NON è un attacco di cyber war. Un attacco di cyber war dovrebbe avere un impatto sulla vita sociale/economica/politica della nazione colpita. Un gruppo di hacker cinese che attacca una stazione elettrica americana può essere considerato un attacco di cyber war, in quanto l'obiettivo è una struttura critica e l'assenza della corrente elettrica ha un grande impatto sulla vita dei cittadini.

**Infrastrutture critiche** Infrastrutture, sistemi, siti, informazioni, persone, reti o processi necessari al funzionamento e vita di una nazione. Comprendono anche quelle organizzazioni, persone e siti che non sono critici al mantenimento dei servizi essenziali, ma la cui protezione è necessaria a causa del grande pericolo che possono provocare al pubblico (es: nucleare civile, siti chimici).

**Elenco strutture critiche** Trasporti pubblici, rete elettrica, comunicazione difesa, servizi di emergenza, finanza, salute, centrali nucleari, ...

**Principali pericoli** Poiché queste infrastrutture forniscono servizi vitali per la vita di tutti i giorni, sono interessato a garantirne la disponibilità, l'integrità e la safety (es: attacco che modifica la velocità delle turbine per farla esplodere -> può ferire gli operatori vicini). In figura 6.1 sono riportati alcuni dei principali attacchi conosciuti.

### 6.1 Sistemi di controllo industriale

La maggior parte delle strutture critiche sono controllate e monitorate da sistemi di controllo industriale (ICS), che garantiscono la safety e la disponibilità.

**Componenti degli ICS** Ho uno o più processi da monitorare (produrre una macchina, sollevare la sbarra del parcheggio, produrre energia elettrica), implementati su macchine a cui sono associati sensori, che ne catturano i dati sul funzionamento. I dati sono verificati da un controllore. Il sistema di controllo industriale implementa due tipi di controllori:

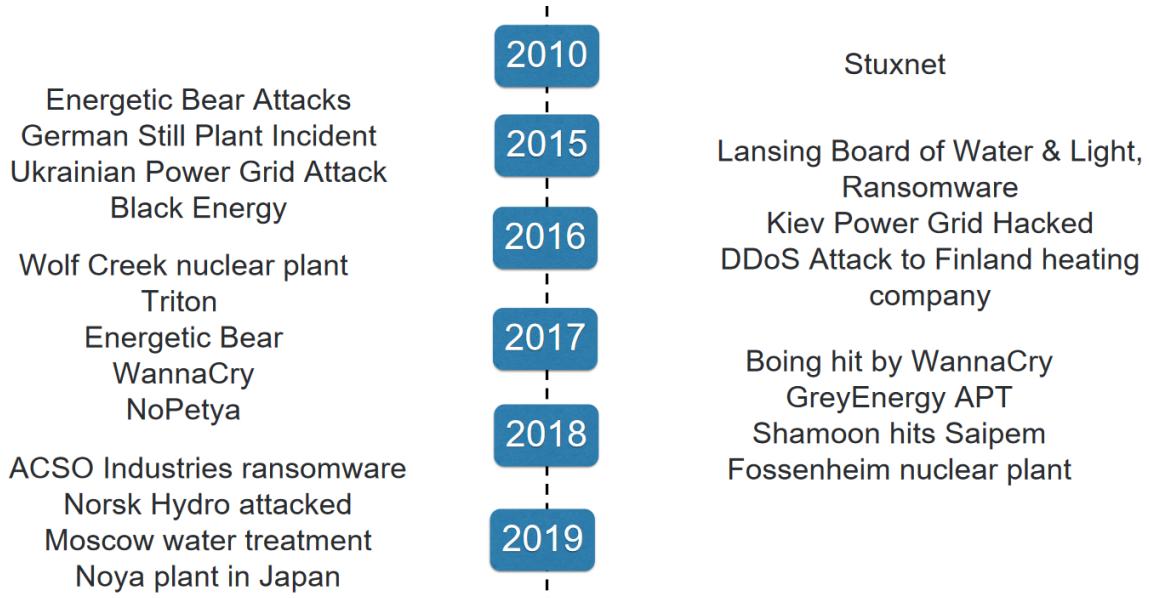


Figura 6.1: Principali attacchi alle infrastrutture critiche

- Uno più vicino alle macchine, che implementa il processo di controllo industriale. Prende il nome di Program Logic Controller (PLC) - quello che posso vedere con <https://shodan.io>
- Uno più distribuito, chiamato SCADA, che controlla tutti i dispositivi coinvolte nel processo industriale.

Il controllore comunica con la Human Machine Interface (HMI), che è l'interfaccia usata dall'operatore. L'operatore invia dei comandi al controllore, che li inoltra agli attuatori.

**Sistemi SCADA** Tra i sistemi più diffusi, è un sistema distribuito consiste di un server che comunica con più stazioni (es: quelle per la distribuzione dell'energia elettrica locali di ogni città) e ne raccoglie i dati. I dati raccolti vengono salvati in un Data Historian ed utilizzabili dagli operatori in caso di necessità. Gli operatori possono anche, tramite le Engineering Workstation, inviare comandi alle diverse stazioni. Un esempio di dispositivo che posso trovare nelle stazioni locali sono gli interruttori per trasmettere l'energia elettrica

Questi sistemi presentano diverse vulnerabilità.

La più importante è legata al fatto che inizialmente questi sistemi erano chiusi, quindi non connessi alla rete. Ora sempre di più utilizzano connessioni internet normali per funzionare. Questo li espone logicamente ad una serie di attacchi.

Molto spesso, inoltre, per garantire la disponibilità dei servizi, il ciclo di vita del macchinario viene estesa molto (circa 10-20 anni). Quindi molti dei sistemi in uso utilizzano software obsoleti non più supportati dal venditore.

Un'altra problematica è legata all'accesso a questi dispositivi. Molti dei sistemi di autenticazione utilizzano l'autenticazione con password, che viene spesso lasciata a quella di default.

Spesso non vengono mantenuti log delle attività svolte dai vari dispositivi, rendendo quindi difficile ricostruire gli attacchi subiti.

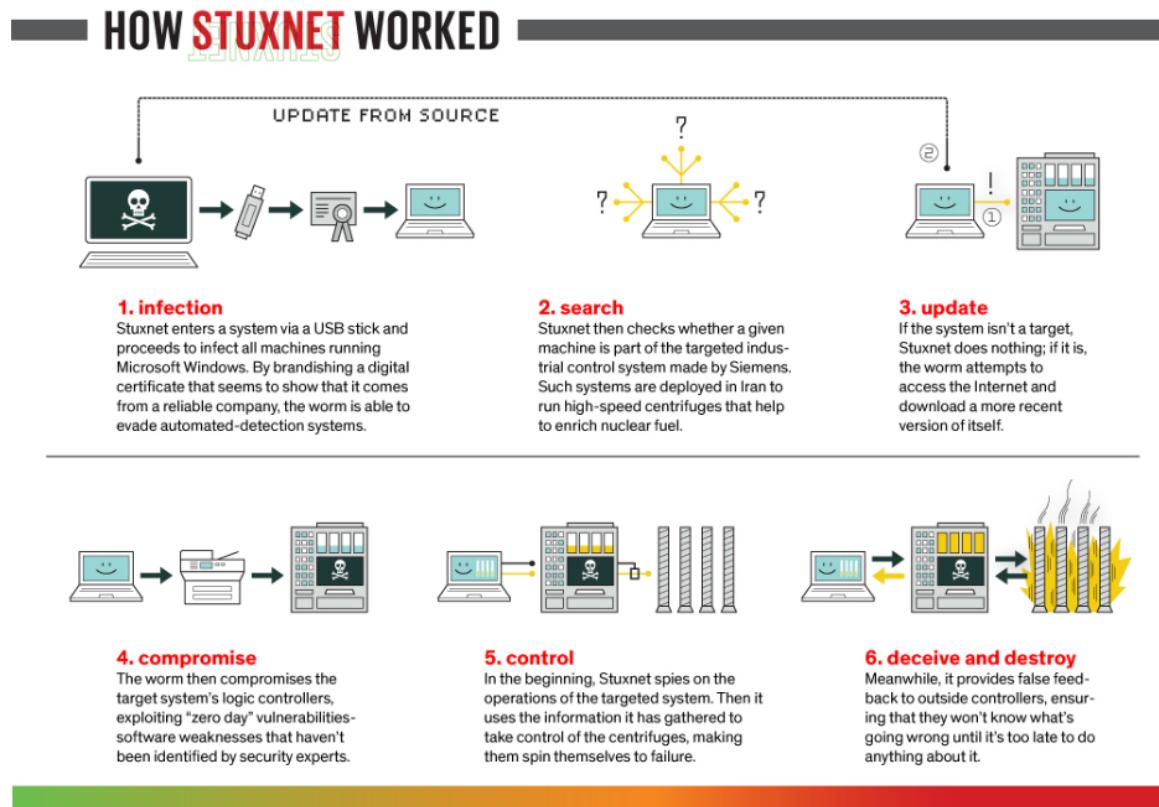


Figura 6.2: Cyber kill chain di Stuxnet

## 6.2 Cyber kill chain per attacchi ICS

Per gli attacchi agli ICS è stata sviluppata una cyber kill chain specifica. Secondo la ricerca svolta, questa si compone di due fasi:

- La prima ha l'obiettivo di ottenere accesso alla rete dell'ICS, e segue le stesse fasi viste nella cyber kill chain classica.
- La seconda fase è quella in cui avviene l'attacco vero e proprio.

Supponiamo che il mio obiettivo sia modificare il sw che gira sul PLC che controlla gli interruttori che attivano/bloccano il trasferimento della corrente elettrica. Per fare ciò sviluppo un malware specifico, che posso testare sul medesimo PLC presente nella stazione che ho acquistato a priori sul mercato (il mio attacco è molto specifico, quindi devo effettuare dei test prima di metterlo in pratica). La delivery del malware la posso fare tramite campagne di phishing oppure sfruttando una chiavetta USB portata all'interno da un insider.

### 6.2.1 Esempi di attacchi reali

**Stuxnet** Prima cyber weapon. Può essere considerato ad oggi il malware più sofisticato mai realizzato. Sviluppato dai servizi di sicurezza americani e dell'intelligence israeliana per sabotare lo sviluppo nucleare iraniano. la cyber kill chain di Stuxnet è riportata in figura 6.2. Stuxnet utilizza due tecniche di propagazione:

1. Propagazione tramite network:

- Infettando le macchie WinCC sfruttando hardcoded passwed (psw salvate in chiaro);
  - Propagandosi sfruttando la vulnerabilità zero-day MS10-061 Print;
  - Propagandosi sfruttando la vulnerabilità MS08-067 del Windows Server Service;
2. Propagazione tramite dispositivi rimovibili, tramite:
- Vulnerabilità LNK (CVE-2010-2568);
  - Autoorun.inf.

Quando veniva infettata una chiavetta USB, venivano creati nella chiavetta tre file: un collevalimento a Shortcul.lnk e due file temporanei. Quando la chiavetta veniva inserita, veniva eseguito lo shortcut, che andava a caricare il primo file tmp, che a sua volta caricava il secondo, che conteneva effettivamente il malware.

Nella prima parte del file Autorun.inf c'era la copia di stuxnet. Questo è visibile dal fatto che ogni eseguibile Windows inizia con i caratteri *MZ*. Quando la chiavetta veniva pluggata, l'autorun rimandava all'esecuzione del programma che conteneva.

La parte di Commando e Control permetteva di scaricare versioni aggiornate del malware, ma sembra che questa feature non sia mai stata usata.

Stuxnet andava a infettare pc che eseguivano il programma Step7, che consentiva di leggere e scrivere i programmi eseguiti dal PLC, e sostituiva una libreria legittima che permetteva di modificare il codice eseguito dal PLC, e quindi lanciare comandi, con una versione malevola. La libreria permetteva di modificare la velocità dei rotori delle turbine (per farle surriscaldare) o modificare la pressione del gas inserito (causando un aumento critico).

**Attacchi del gruppo Sandworm** Gruppo affiliato al governo russo. I membri sono autori di diversi attacchi ad infrastrutture critiche di diversi paesi a partire dal 2015, come:

- Attacco alla rete elettrica ucraina ();
- Campagna di Spearphishing contro i membri del partito del presidente Macron (2017);
- Infezioni col ransomware NotPetya (2017);
- Attacchi contro le olimpiadi invernali (2017);
- Rallentamento delle investigazioni sulla morte della spia Novick (2018);
- Campagne contro infrastrutture critiche e siti governativi della Georgia.

Il primo attacco è stato effettuato il 23 dicembre 2015 in una stazione elettrica in Ucraina, che ha causato il blackout in una regione dei dintorni della capitale Kiev per qualche ora. Questo è stato il primo attacco ad una struttura critica. L'obiettivo dell'attacco erano le stazioni di distribuzione dell'energia elettrica.

Generalmente, un sistema di distribuzione si compone di una centrale che produce l'energia e la trasferisce alle stazioni di trasmissione, che ne abbassano il voltaggio e la trasferiscono agli utenti finali. Sandworm aveva compromesso gli interruttori che consentivano al trasmissione agli utenti finali di interromperla, impedendola.

Questo è stato possibile in quanto nell'estate precedente Sandworm ha compiuto una campagna di spearphishing contro i dipendenti dell'azienda che gestiva le stazioni, per rubare le credenziali per accedere alla rete e collegarsi alle unità remote che controllavano gli interruttori.

Le mail di spearphishing contenevano una macro che installava un keylogger sulle macchine delle vittime per rubare le credenziali. Una volta ottenuto accesso alla VPN hanno anche reso inutilizzabili, da parte degli operatori, le work station (usate per inviare comandi agli interruttori). Per impedire agli operatori di riottenere il controllo delle work station, è stato utilizzato un altro malware, KillDisk, che ne cancellava il master boot record (rendendole inoperabili). Inoltre, per impedire che venisse notificato da parte della popolazione il blackout, Sandworm ha effettuato un attacco DDoS verso il servizio di assistenza ai clienti della compagnia elettrica.

Questo attacco ha coinvolto due malware: KillDisk e BlackEnergy 3. Quest'ultimo era quello che veniva installato dalle mail di phishing. Aveva una struttura modulare, dove due moduli si occupavano di ricostruire la mappatura del sistema di controllo delle stazioni, mentre altri due del furto delle password. Aveva inoltre la capacità di collegarsi ad un command and control server da cui scaricava l'altro malware.

Il 17 dicembre 2016, Sandworm ha attaccato la trasmissione dell'energia elettrica nella capitale Kiev. L'attacco è stato simile a quello precedente: mesi prima è stata effettuata nuovamente una campagna di spearphishing per ottenere accesso alla rete. Questa volta però il malware usato era Industroyer. Si tratta di un malware di complessità simile a stuxnet, ed è il secondo malware mai creato per sabotare un sistema industriale. Il suo obiettivo era quello di rendere inutilizzabili i remote terminal unit che controllavano gli interruttori per la trasmissione dell'energia elettrica. Si componeva di moduli che implementavano i quattro possibili protocolli di comunicazione che la work station degli operatori utilizzava per inviare i comandi ai remote terminal unit delle stazioni di trasmissione. Implementava anche un attacco DDoS contro i protection relay, i meccanismi che permettevano di richiudere gli interruttori in caso si verificasse una situazione anomala.

Industroyer aveva struttura modulare, con una main backdoor che permetteva l'accesso alla rete dell'azienda. Questa installava inoltre un'altra backdoor, utilizzata nel caso la prima fosse eliminata e scaricava due tool, uno per rubare le credenziali delle work station e l'altro per effettuare l'attacco DDoS. Il cuore del malware è rappresentato dal launcher, che implementava ed eseguiva i quattro payload che implementavano i quattro possibili protocolli di comunicazione.

NotPetya era un ransomware che colpì inizialmente l'Ucraina, per poi diffondersi in tutto il mondo. In Ucraina venne diffuso come update del software di contabilità fiscale M.E.Doc (molto usato nel paese): Sandworm era stata in grado di intercettare il traffico degli aggiornamenti di questo sw e ridirigerlo in un server situato in Francia, da cui installavano il malware. Una volta installato iniziava a cifrare i dati delle macchine infette. La stranezza di questo ransomware era che utilizzava lo stesso indirizzo bitcoin per il riscatto in ogni infezione, consentendo quindi di rintracciare i creatori. Inoltre richiedevano che il pagamento fosse avvenuto via email. Soprattutto, la chiave del ransomware, usata dagli attaccante per identificare le vittime, era uguale in ogni infezione e, nel caso vi fossero memorie esterne collegate, queste venivano cifrate con chiavi differenti. Quindi, anche se le vittime pagavano il riscatto, gli attaccanti non avevano modo di identificare la macchina e quindi fornire la chiave corretta. Si pensa quindi che l'obiettivo dell'attacco era ottenere accesso alle macchine delle vittime per poi effettuare altri attacchi (la richiesta fungeva da diversivo).



# Capitolo 7

## Autenticazione dell'utente

Il processo di autenticazione è uno dei primi metodi utilizzabili per proteggersi da attacchi informatici. Consiste nel verificare l'identità dell'utente, ovvero che sia chi dice di essere. Questo è importante per:

- Garantire la proprietà di authentication -> se ho l'identità di un utente verificata, gli posso associare i permessi per lui previsti all'interno del sistema;
- Garantire la proprietà di accountability -> posso determinare chi ha effettuato determinate azioni all'interno del sistema (mantengo log sull'utente).

Ci sono tre modi principali per verificare l'identità di un utente, che possono essere usati singolarmente o in combinazione:

- Tramite un segreto che l'utente possiede;
- Tramite qualcosa che l'utente possiede;
- Tramite qualcosa che l'utente è.

**Sistemi token-based** Si utilizza un token generato da un'app, un dispositivo (OTP), una smart card o un barcode per autenticarsi. I dispositivi OTP solitamente implementano un algoritmo di hashing per generare la OTP, che prende in input un segreto memorizzato sul device/app. Il codice generato ha solitamente una validità limitata (30-60 secondi) e può essere usato in un'unica sessione di autenticazione. L'algoritmo più usato è il Time Based One Time Password Generation Algorithm, che prende in input un clock interno sincronizzato con quello del server di autenticazione per generare la OTP (utilizzo quindi il tempo).

L'altra tecnica token-based usata è quella che prevede l'uso delle smart card (carta con un piccolo processore), dove vengono memorizzati i certificati della chiave pubblica e della chiave privata rilasciati agli utenti. Per poter utilizzare la smartcard per autenticarsi è necessario utilizzare un lettore per collegarla la pc. Il processo di autenticazione segue un protocollo di challenge and response:

- Prima di tutto l'utente sblocca la carta con il pin associato;
- Il lettore genera un valore casuale che invia alla smart card;
- La card genera a sua volta un valore casuale e lo combina con quello del lettore. Firma il risultato con la chiave privata;
- Il lettore verifica la validità della firma con la corrispondente chiave pubblica.

Problemi: posso perdere/mi viene rubato il sistema che mi genera il token, rendendomi impossibile accedere al sistema.

**Autenticazione con sistemi biometrici** Come per le smart card, è necessario uno scanner della caratteristica biometrica che l'utente usa per autenticarsi. Le caratteristiche utilizzabili per autenticarsi devono essere:

- Universali -> comuni a tutti gli individui;
- Distintive -> ogni persona dovrebbe avere differenze notabili in quella caratteristica;
- Permanenza -> la caratteristica non dovrebbe cambiare notevolmente nel tempo;
- Collezionabili -> la caratteristica dovrebbe essere effettivamente determinabile e quantificabile.

La caratteristica viene quantificata tramite lo scanner e salvata. Al momento dell'autenticazione, il sistema confronterà il dato precedentemente salvato con quello quantificato al momento: se la differenza è limitata, l'utente viene autenticato.

Esempi: firma, impronta digitale, retina, voce, volto, ...

Problemi: gli algoritmi usati possono generare falsi positivi (autorizzo un utente che non sarebbe autorizzato) e falsi negativi (non autorizzo un utente legittimo); a seconda della caratteristica scelta, un attaccante potrebbe facilmente rubarla (es: impronta digitale); l'utente potrebbe non voler farsi scannerizzare una determinata caratteristica.

## 7.1 Autenticazione con password

Si tratta di sistemi semplici e poco costosi da implementare, ma molto vulnerabili ad attacchi, principalmente in quanto richiedono agli utenti uno sforzo cognitivo superiore a quello che sono in grado di effettuare.

Uno dei problemi è che gli utenti utilizzano le stesse password sia in ambito domestico e lavorativo.

### 7.1.1 Attacchi alle password

Esistono diversi attacchi che permettono agli attaccanti di ottenere le password delle vittime:

- Offline attacks: hanno l'obiettivo di accedere al \*\*password file\*\* salvato sul server di autenticazione in cui sono contenute tutte le password degli utenti;
- Online attacks: possono essere attivi o passivi. Richiedono che gli attaccanti interagiscano con l'obiettivo dell'attacco (es: servizio di cui voglio ottenere la password per accedere, account sul pc dell'organizzazione). Negli attacchi attivi, l'attaccante prova diverse password per trovare quella corretta, mentre in quelli passivi utilizza altri metodi, come l'intercettazione del traffico dell'utente;
- Non technical attacks: utilizzano tecniche di ingegneria sociale per ottenere le credenziali.

Tra le tecniche di attacco ci sono:

- Attacchi di brute force;
- Attacchi con key logger;
- Attacchi con mail di phishing;



Figura 7.1: Esempio di password file.

### 7.1.2 Attacchi offline

Dove sono memorizzate le password?

- In Windows il password file è accessibile in due modi:
  - In C:\Windows\System32\config nel file SAM;
  - Nei registri di Windows sotto HKEY\_LOCAL\_MACHINE -> SAM -> SAM.

Solitamente l'accesso al file è garantito solo ad account amministratore o system.

- In Linux ci sono due file usati per salvare la password:
  - \etc\passwd contiene gli user di tutti gli utenti collegati alla macchina;
  - \etc\shadow contiene gli hash delle password memorizzate nel password file.

Generalmente un password file memorizza quattro parametri per le password (figura 7.1):

- Utente;
- SID;
- Due Hash.

Salva quindi il nome dell'utente, un SID associato a tale nome e due hash della password, create tramite due algoritmi differenti (figura 7.2):

- LM: usato da sistemi Windows fino alla versione XP. Crea hash a 128 bit ma utilizza un character set di 142 caratteri e accetta password si max 14 caratteri. Divide la password in due sottostringhe di 7 caratteri e non è case sensitive (genera lo stesso hash se le due password sono uguali ma una con caratteri maiuscoli e una minuscoli);
- NTLM: introdotto successivamente e più sicuro. Crea hash a 128 bit ma utilizza tutti i caratteri ASCII e accetta password di max 256 caratteri. Non divide la password ed è case sensitive.

Per motivi di compatibilità, nelle versioni più recenti di Windows, sono presenti entrambi gli hash.

LM	NTLM
128 bits	128 bits
Character set: 142	Character set: 65000
Max length: 14 characters	Max length: 256 characters
Password split in two 7 char strings	Based on the entire password
Case insensitive	Case Sensitive

Figura 7.2: Algoritmo LM e nTLM.

### Algoritmo LM

- Memorizza password di max 14 caratteri;
- Converte tutti i caratteri a MAIUSCOLI;
- Filla gli spazi vuoti della passwword fino a raggiungere 14 caratteri;
- Divide la password in due sottostringhe di 7 caratteri;
- Cifra separatamente le due sottostringhe;
- Combina i due risultati.

**Principali attacchi offline** I principali attacchi offline sono tre:

1. Attacchi di forza bruta (meglio se conosco le policy usate per generare password in quel sistema -> caratteri usabili, lunghezza massima);
2. Attacchi a dizionario;
3. Attacchi ibridi (uso un dizionario e includo variazioni delle parole con numeri e caratteri speciali).

Posso velocizzare questo processo usando le rainbow tables, che sono delle tabelle che, per tutte le parole di uso comune, hanno precomputato il corrispondente hash per i principali algoritmi di hashing. Per memorizzarle è però necessario un grande spazio di memoria.

Uno dei tool usati per condurre i tre attacchi sopra è **John the Ripper**.

La forza di una password misura la sua resistenza contro gli attacchi di brute force (quindi quanti tentativi l'attaccante deve fare prima di poterla indovinare). Normalmente dipende dalla lunghezza della password e dal numero di caratteri utilizzabili. Per questo le organizzazioni impongono una password policy che vincolano la creazione della password a delle condizioni.

Questa misura non è però molto buona in quanto non tiene conto dei pattern comuni che possono usare gli utenti nella creazione delle password (es: lettera maiuscola all'inizio, numeri alla fine, sequenze sulla tastiera, ...).

**\*Misura della robustezza di Dorpbox (zxcvbn)** Verifica i possibili pattern che si possono applicare alla password dell'utente, stima i tentativi che l'attaccante deve fare per determinare le sottostringhe individuate dai pattern e selezione quelli che ricostituiscono la password e che richiedono meno tentativi da parte dell'attaccante. Il numero di tentativi rimanenti determina la stima della robustezza della password.

### Contromisure agli attacchi offline

- Proteggo i file delle password con il meccanismo di controllo degli accessi;
- Mantengo gli hash delle password separati dagli username degli utenti, in modo che sia più difficile determinare a quale utente appartiene l'hash/la password (Linux).
- Aumento i tentativi che gli attaccanti devono fare per indovinare la password, aggiungendo alla password un numero casuale (salt). In questo modo l'attaccante deve ricostruire l'hash e questo numero causale, aumentando le combinazioni che deve tentare.
- Resetto il password file se viene compromesso.

#### 7.1.3 Attacchi online

Gli attacchi offline visti possono essere effettuati anche online. Ad esempio, se l'attaccante vuole accedere all'online banking della vittima può provare le password di un dizionario.

### Contromisure agli attacchi online

- Setto una politica delle password per gli utenti;
- Cambio frequentemente le password;
- Aiuto l'utente a generare la password in maniera automatica (per evitare i pattern visti).

#### 7.1.4 Protezione contro gli attacchi alle password

Diverse ricerche hanno dimostrato che le contromisure viste sopra non sono più sicure (es: cambiare continuamente password porta l'utente a seguire un pattern).

Devo quindi usare misure più effettive:

- Introduco meccanismi di lockout -> dopo  $n$  tentativi falliti blocco temporaneamente l'account dell'utente. Implementare questi meccanismi non è facile in quanto si rischia di bloccare l'account di un utente legittimo che si è dimenticato al password. Generalmente si impone il limite a 8-10 tentativi falliti;
- Rallento i tentativi fatti dall'attaccante introducendo un intervallo di tempo in cui aspettare per effettuare il successivo tentativo, in caso di login fallito (\*\*throttling\*\*);
- Implemento meccanismi di protective monitoring che notifichino in casi di attività inusuale dell'account (es: accesso a google su nuovi dispositivi);
- Impedisco l'utilizzo di password listate come compromesse (password blacklisting);
- Implemento l'autenticazione a più fattori;

#### 7.1.5 Attacchi passivi

Un altro modo per ottenere la password usata per accedere ad un servizio online è effettuare il cosiddetto *man in the middle attack*: l'attaccante snifferà il traffico di rete e se è fortunato il traffico tra il pc e il server non è cifrato (niente HTTPS) o il protocollo è facilmente crackabile, vedendo così la password in chiaro.

Il key logger cattura tutte le comunicazioni fatte dalla macchina della vittima.

Gli attacchi di ingegneria sociale sono molteplici:

- Mail di phishing;
- Tecniche che richiedono la vicinanza della vittima (shoulder surfing);
- Ricerca di pezzi di carta/documenti nella spazzatura della vittima (dumpster diving);

### **Contromisure agli attacchi passivi**

- Educo gli utenti.

# Capitolo 8

## Identità digitali

L'autenticazione con password rappresenta spesso anche un rischio per la privacy dell'utente: molti fornitori di servizi mantengono altre info personali richieste all'utente per il rilascio di user e password (es: nome, cognome, mail, indirizzo fisico, ...). Se il fornitore del servizio è soggetto ad un databreach, l'utente rischia la diffusione dei suoi dati. Inoltre il fornitore, se soggetto a databreach, è costretto a pagare una multa salata (GDPR) in quanto non ha protetto adeguatamente i dati personali degli utenti.

Quindi i sistemi di autenticazione con password rappresentano problemi sia per l'utente che per il fornitore.

Tutti questi problemi possono essere risolti dai sistemi di gestione dell'identità digitale. In questi sistemi il processo di autenticazione viene delegato dai fornitori dei servizi al sistema di identità digitale stesso, che si occuperà di verificare l'identità dell'utente nel momento in cui vuole accedere al servizio online. Le credenziali e le eventuali info associate vengono mantenute dal sistema di identità digitale. Questo solleva il service provider dall'implementare il sistema di autenticazione e da eventuali sanzioni derivanti da attacchi che rivelano le info personali degli utenti. Per gli utenti, previene che questi debbano mantenere credenziali potenzialmente diverse per ciascuno dei servizi online a cui vogliono accedere. L'utente deve infatti mantenere un solo set di credenziali, rilasciato da servizio di identità digitale, che utilizzerà per accedere ai vari servizi.

### 8.1 Identità digitale

Un'identità digitale è un insieme di attributi che identificano l'utente, come:

- Nome e cognome;
- User e password;
- Numero della carta di identità/passaporto;
- Indirizzo;
- Ruolo all'interno dell'organizzazione;
- Email;
- ...;

Il sistema di identità digitale, quindi:

- Mantiene l'identità dell'utente e ci associa vari attributi;



Figura 8.1: Il service provider è anche l'identity provider.

- Verifica l'identità dell'utente basandosi sui suoi attributi di identità.

Ogni volta che l'utente vuole accedere ad un servizio online, verifica la sua identità e certifica al fornitore del servizio che l'utente è stato autenticato con successo.

Il sistema di identità digitale comprende tre attori principali:

- L'utente;
- L'identity provider, che è il soggetto legale che verifica l'identità dell'utente e rilascia ai fornitori del servizio una prova che l'utente è stato autenticato con successo;
- Service provider, che fornisce il servizio online e ne concede l'accesso in base alle dichiarazioni rilasciate dall'identity provider. Si fidano quindi del processo di autenticazione effettuato dall'identity provider.

## 8.2 Single-Sign On

I sistemi di identità digitale permettono di implementare il concetto di SSO: un utente può riutilizzare lo stesso set di credenziali per accedere a più risorse/servizi online. Tipicamente l'utente, quando richiede l'accesso ad un servizio online, viene ridiretto sulla pagina dell'identity provider che gli ha rilasciato l'identità digitale, dove questo gli verifica l'identità. Se l'identità è verificata con successo, l'identity provider rilascia un'asserzione che l'autenticazione è avvenuta con successo e la inoltra al fornitore. Il fornitore verifica la validità dell'asserzione e decide se garantire l'accesso all'utente.

Nelle figure 8.1 e 8.2 sono riportati degli esempi di SSO.

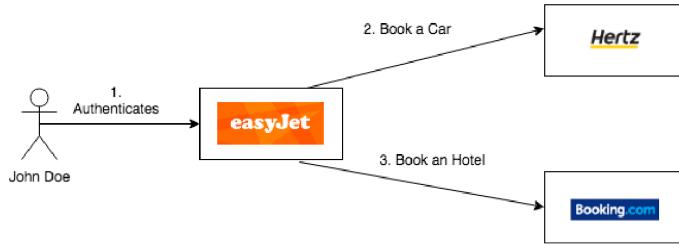


Figura 8.2: Il service provider è diverso dall'identity provider.

### 8.3 Identità federata

L'identità federata sta alla base degli SSO cross domain. In questo caso diversi service provider decidono di fidarsi del processo di autenticazione effettuato da un altro service provider che fa parte della stessa federazione. Qualsiasi entità della federazione può svolgere il ruolo di identity provider.

#### 8.3.1 SPID

Lo SPID è un esempio di sistema di identità federate in Italia. Con questo è possibile accedere ai servizi della PA e a quelli dei privati che hanno aderito alla federazione.

I principali attori coinvolti nello SPID sono:

- L'utente;
- L'Agenzia per l'Italia Digitale (AgID), che è l'entità che si occupa di accreditare tutti i fornitori di servizi e le entità che possono rilasciare lo SPID;
- L'Identity Provider, che è l'entità che rilascia l'identità digitale (es: Poste Italiane);
- Service provider;
- Attribute provider, che è responsabile di rilasciare gli attributi su cui viene poi rilasciata l'identità digitale.

Lo SPID ha tre livelli di identità digitale, che corrisponde a tre diversi livelli di autenticazione sicura:

- Il livello 1 lo SPID viene rappresentato da user e password;
- Il livello 2 richiede, oltre a quanto previsto per il livello 1, anche un OTP che viene inviata tramite app o sms;
- Il livello 3 richiede, oltre a quanto previsto per i due livelli sopra, anche un dispositivo rilasciato dall'identity provider.

Con l'aumentare del livello di sicurezza aumentano anche i rischi se viene compromessa l'identità digitale.

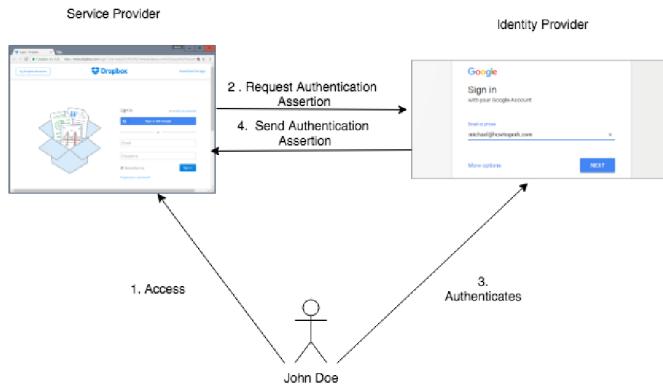


Figura 8.3: La SAML Authentication Assertion di Google (prodotta in seguito alla request di Dropbox) specificherà quando è avvenuto il processo di autenticazione dell’utente, con quale modalità e per quanto tempo è valida l’autenticazione.

## 8.4 SAML

Per implementare i sistemi di identità federate, è necessario un meccanismo standard per trasmettere info sul processo di autenticazione effettuato dall’identity provider e che deve essere validato dai service provider.

Il protocollo usato è SAML (Security Assertion Markup Language). Si tratta di un protocollo xml che consente agli identity provider e ai servide provider di scambiarsi attributi identificativi degli utenti e/o asserzioni per indicare che il processo di autenticazione ha avuto successo.

Un tipico caso in cui il protocollo viene usato è quello della cross authentication (8.3).

### 8.4.1 Asserzioni SAML

SAML permette di fare dichiarazioni su attributi posseduti da un soggetto (attribute assertion), sul processo di autenticazione effettuato dall’utente (authentication assertion) e sui permessi che sono garantiti all’utente sulle risorse fornite dal servizio online (authorization assertion).

Tipicamente ogni asserzione contiene:

- L’identity provider che ha rilasciato l’asserzione;
- Quando l’asserzione è stata generata;
- Un ID che consente all’identity provider di distinguere l’asserzione dalle altre;
- Informazione relativa al soggetto a cui è legata l’asserzione (nome e dominio a cui eventualmente appartiene);
- Condizione di validità (es: periodo di tempo, accesso solo a specifici servizi, ...).

## 8.5 Shibboleth

Altro protocollo che permette di implementare le identità federate. Si basa sullo standard di SAML. Viene usato principalmente per autenticare utenti che voglio accedere a risorse fornite da università o istituti di ricerca. Permette quindi di implementare il concetto di identità federate anche in ambito accademico.

Un esempio di federazione accademica è la UK Access Management Federation, che raccolge tutte le università inglesi.

Prevede tipicamente tre identità: l'utente, il fornitore di servizi (università X) e l'identity provider (università Y). Quando l'utente prova ad accedere all'università Y, viene ridiretto al servizio WAYF (Where Are You Form), che gli consente di selezionare l'università X all'interno della federazione che fornisce le sue credenziali. A questo punto il servizio lo ridirige al servizio di autenticazione dell'università X selezionata che si occuperà di verificare la sua identità. Se la sessione di autenticazione ha acuto successo, l'università X genera un Handle che identifica la sessione di autenticazione (condiviso con Y). L'università Y, prima di concedere l'accesso ai suoi servizi online, rimanda l'Handle appena ricevuto e può richiedere attributi dell'utente per garantirgli l'accesso (potrebbe non accettare solo la prova dell'Handle). L'università X fornisce gli attributi aggiuntivi e l'università Y concede l'accesso all'utente.

## 8.6 OpenID

OpenID Connect (OIDC) è un protocollo di autenticazione aperto che profila ed estende OAuth 2.0 per aggiungere un livello di identità. OIDC consente ai client di confermare l'identità di un utente finale utilizzando l'autenticazione da parte di un Server di autorizzazione. L'implementazione di OIDC su OAuth 2.0 crea un unico framework che promette di proteggere API, applicazioni native mobili e applicazioni browser in un'unica architettura coesa.



# Capitolo 9

## Protocollo OAuth

Vedremo ora degli scenari in cui è l'utente che decide se garantire o meno l'accesso alle sue risorse (servizi cloud come Drive, dispositivi IoT, ...).

Per gestire queste situazioni è necessario un protocollo che garantisca che solo le applicazioni decise dall'utente possano accedere a queste risorse (risorse ospitate sul web, dispositivi smart da controllare, ...).

Il protocollo usato è OAuth (Open Authorization), che è un protocollo standard che consente ad applicazioni di terze parti di accedere a risorse protette hostate su un server HTTP (es: schermata di richiesta di accesso ad info dell'account Google nei giochi per cellulare). L'accesso viene garantito se approvato dall'utente.

Il protocollo prevede la presenza di un Authorization Server che rilascia all'applicazione un access token quando l'utente garantisce l'accesso alle proprie risorse usando quell'app.

Gli attori principali sono quanto:

- Owner delle risorse, che è l'entità che può garantire l'accesso alla risorsa;
- Resource Server, che è il server che ospita le risorse dell'utente;
- Authorization Server, che è il server che rilascia l'access token al client dopo aver autenticato il proprietario della risorsa e ottenuto l'autorizzazione;
- Client, che è l'app di terze parti che l'utente usa per accedere alle risorse.

Scenario tipico: voglio accedere a Spotify tramite le credenziali Facebook. Le risorse protette sono le mie credenziali Facebook.

Il protocollo supporta 5 casi differenti:

1. **Authorization Code Grant Flow;**
2. **Authorization Code Grant Flow with PCKE;**
3. **Resource Owner Password;**
4. **Client Credential;**
5. **Device Flow.**

### 9.1 Authorization Code Grant Flow

Usata quando si accede a un'applicazione usando un account Google o Facebook.

### 9.1.1 Flusso

Il client reindirizza l'utente al server di autorizzazione con i seguenti parametri nella stringa di query:

- **response\_type** con il valore **code**;
- **client\_id** con l'identificatore del client;
- **redirect\_uri** con l'URI di reindirizzamento del client. Questo parametro è facoltativo, ma se non viene inviato l'utente verrà reindirizzato a un URI di reindirizzamento preregistrato.
- **scope** con un elenco di scope;
- **state** con una stringa casuale usata per prevenire attacchi (token CSRF).

Tutti questi parametri verranno convalidati dal server di autorizzazione. All'utente verrà quindi chiesto di accedere al server di autorizzazione e approvare il client.

Se l'utente approva il client, verrà reindirizzato dal server di autorizzazione al client (in particolare all'URI di reindirizzamento) con i seguenti parametri nella stringa di query:

- **code** con il codice di autorizzazione;
- **state** con il parametro **state** inviato nella richiesta originale. È consigliabile confrontare questo valore con il valore archiviato nella sessione dell'utente per assicurarsi che il codice di autorizzazione ottenuto risponda alle richieste effettuate da questo client anziché da un'altra applicazione client.

### 9.1.2 Flusso (pt. 2)

Il client invierà ora una richiesta POST al server di autorizzazione con i seguenti parametri:

- **grant\_type** con il valore di **authorization\_code**;
- **client\_id** con l'identificatore del client;
- **client\_secret** con il segreto client;
- **redirect\_uri** con lo stesso URI di reindirizzamento a cui l'utente è stato reindirizzato;
- **code** con il codice di autorizzazione dalla stringa di query.

Il server di autorizzazione risponderà con un oggetto JSON contenente le seguenti proprietà:

- **token\_type** questa sarà di solito la parola "Bearer" (per indicare un token al portatore);
- **expires\_in** con un numero intero che rappresenta il TTL del token di accesso (ovvero quando il token scadrà);
- **access\_token** con token di accesso stesso
- **refresh\_token** con un refresh token che può essere utilizzato per acquisire un nuovo token di accesso alla scadenza dell'originale.

A questo punto il client presenta l'access token all'Authorization Server, che fornisce la risorsa all'app.

Questo processo è applicato a tutte le app native (es: su cellulare) o web app in cui il fornitore del client è diverso dal fornitore della risorsa.

## 9.2 Authorization Code Flow with PCKE

Simile all'Authorization Code Flow, ma presenta due differenze fondamentali:

- È usato in client basati su user-agent (ad esempio app Web a pagina singola) che non possono mantenere segreto un client perché tutto il codice dell'applicazione e l'archiviazione sono facilmente accessibili;
- L'authorization server ritorna direttamente l'access token piuttosto che in codice di autorizzazione da scambiare per l'access token.

### 9.2.1 Flusso

Il client reindirizza l'utente al server di autorizzazione con i seguenti parametri nella stringa di query:

- **response\_type** con il valore "token";
- **client\_id** con l'identificatore del client;
- **redirect\_uri** con l'URI di reindirizzamento del client. Questo parametro è facoltativo, ma se non viene inviato l'utente verrà reindirizzato a un URI di reindirizzamento preregistrato;
- **scope** con un elenco di scope delimitato da spazi;
- **state** con un token CSRF.

Tutti questi parametri verranno convalidati dal server di autorizzazione. All'utente verrà quindi chiesto di accedere al server di autorizzazione e approvare il client.

Se l'utente approva il client, verrà reindirizzato al server di autorizzazione con i seguenti parametri nella stringa di query:

- **token\_type** con il valore "Bearer";
- **expires\_in** con un numero intero che rappresenta il TTL del token di accesso;
- **access\_token** con l'access token stesso;
- **state** con il parametro state inviato nella richiesta originale.

Questa soluzione non restituisce un refresh token perché il browser non ha mezzi per mantenerlo privato.

## 9.3 Resource Owner Password

Questa soluzione è usata client first party affidabili sia sul Web che nelle applicazioni per dispositivi nativi.

### 9.3.1 Flusso

Il client chiede all'utente le credenziali di autorizzazione (di solito un nome utente e una password). Il client invia quindi una richiesta POST con i seguenti parametri del corpo al server di autorizzazione:

- **grant\_type** con il valore "password";
- **client\_id** con l'ID del client;
- **client\_secret** con il segreto del client;
- **scope** con un elenco delimitato delle autorizzazioni richieste;
- **username** con il nome utente dell'utente;
- **password** con la password dell'utente.

Il server di autorizzazione risponderà con un oggetto JSON contenente le seguenti proprietà:

- **token\_type** con il valore "Bearer";
- **expires\_in** con un numero intero che rappresenta il TTL del token di accesso;
- **access\_token** con il token di accesso stesso;
- **refresh\_token** con un refresh token che può essere utilizzato per acquisire un nuovo token di accesso alla scadenza dell'originale.

## 9.4 Client Credential

La più semplice di tutte le concessioni OAuth 2.0, questa soluzione è adatta per l'autenticazione da computer a computer in cui non è richiesta l'autorizzazione di un utente specifico per accedere ai dati.

### 9.4.1 Flusso

Il client invia una richiesta POST con i seguenti parametri al server di autorizzazione:

- **grant\_type** con il valore "client\_credentials";
- **client\_id** con l'ID del cliente;
- **client\_secret** con il segreto del cliente;
- **scope**.

Il server di autorizzazione risponderà con un oggetto JSON contenente le seguenti proprietà:

- **token\_type** con il valore "Bearer";
- **expires\_in** con un numero intero che rappresenta il TTL del token di accesso;
- **access\_token** con il token di accesso stesso.

## 9.5 Device flow

Vedere PDF.

# Capitolo 10

## Gestione degli accessi

I sistemi di autenticazione sono la prima linea di difesa contro i principali attacchi informatici. La seconda misura di protezione fondamentale è quella dei sistemi di controllo dell'accesso. Questi sistemi associano ad un utente autenticato con successo i permessi di cui dispone all'interno di quel servizio/organizzazione.

Un sistema di controllo degli accessi si compone di:

- Una funzione di autenticazione, che verifica l'identità dell'utente;
- Una componente di access control, che, seguendo le politiche di controllo dell'accesso settate, decide se l'utente può accedere o meno alle risorse;
- Una componente di auditing, che tiene traccia delle risorse a cui accede l'utente e quali permessi gli sono stati assegnati.

Queste tre componenti permettono di garantire le proprietà di:

- Autenticazione;
- Autorizzazione;
- Accountability.

Le politiche per il controllo dell'accesso specificano le condizioni che l'utente deve rispettare per accedere al sistema. Queste vengono formalizzate secondo un modello per il controllo dell'accesso e vengono applicate tramite un meccanismo di controllo dell'accesso, che è rappresentato da un'architettura con più componenti logici che sono responsabili di decidere se l'accesso è garantito o negato.

Un sistema di controllo degli accessi ha tre elementi fondamentali:

- L'utente autenticato che richiede l'accesso a determinate risorse o servizi;
- Le risorse a cui viene ristretto l'accesso;
- il diritto di accesso che l'utente possiede per le risorse (es: leggere, scrivere, eseguire, creare, cercare, ...).

## 10.1 Modelli di controllo dell'accesso

Esistono quattro principali modelli per il controllo dell'accesso:

- Discretionary Access Control (DAC), dove i permessi vengono direttamente associati all'identità verificata dell'utente;
- Mandatory Access Control (MAC), dove vengono assegnate delle etichette di sicurezza (*security label*) sia agli utenti che alle risorse;
- Role based Access Control (RBAC), dove i permessi sono associati al ruolo che l'utente ricopre;
- Attribute-based Access Control (ABAC), dove le politiche di controllo dell'accesso sono espresse come condizioni sugli attributi posseduti dal soggetto che richiede l'accesso, dalla risorsa a cui vuole accedere o dal contesto in cui l'utente richiede l'accesso.

### 10.1.1 Discretionary Access Control

L'amministratore della risorsa decide a chi garantire i permessi per accedere a quella risorsa ed, eventualmente, revoca i permessi assegnati.

Un modo per rappresentare le politiche di controllo dell'accesso secondo questo modello è tramite la Matrice di Controllo degli Accessi. Le colonne rappresentano le risorse, le righe gli utenti e le celle i permessi assegnati. Questa soluzione richiede però un elevato spazio di memoria per salvare la tabella.

Per limitare il problema della memoria è possibile utilizzare le Liste per il Controllo degli Accessi, dove per ogni risorsa viene specificata, tramite lista, quali utenti possono accedervi e con quali permessi. Anche questa soluzione presenta però problemi: se voglio rimuovere tutti gli accessi ad un utente, devo scansionare tutte le liste alla ricerca della sua entry (stessa cosa devo fare se voglio vedere che permessi sono associati ad un utente). Si possono anche utilizzare le Capability List, che memorizzano per ogni utente le risorse a cui possono accedere con i relativi permessi.

### 10.1.2 Role Based Access Control (RBAC)

Introdotto per superare le limitazioni del DAC. In questo caso i permessi vengono associati direttamente ad un ruolo, che generalmente rappresenta una qualifica nell'organizzazione, che viene associato poi all'utente.

Esistono tre famiglie di RBCA:

- La versione base comprende gli utenti, i ruoli, i permessi e le sessioni (specifica qual è il ruolo che l'utente ricopre in un dato momento);
- La versione RBCA1 introduce il concetto di gerarchia dei ruoli, che riduce l'assegnamento dei permessi in quanto un ruolo superiore eredita i permessi dei ruoli sottostanti a cui è collegato;
- La versione RBCA2 introduce i *separation of duty constraints*, che permettono di implementare il principio del *least privilege* -> per completare un'azione su una risorsa è necessaria l'azione di due utenti diversi (l'approvazione di un mutuo richiede l'approvazione di un impiegato e del direttore della banca). Questo principio è implementato con i vincoli di separation of duty. Ne esistono di due tipi:

- Vincoli statici, dove un utente non può essere assegnato a più di  $n$  ruoli definiti in un insieme di ruoli;
- Vincoli dinamici, dove un utente non attivare più di  $n$  ruoli appartenente all'insieme in quella sessione.

Questo modello di controllo dell'accesso permette di assegnare facilmente permessi ad un nuovo utente (basta che trovo il ruolo adatto) e permette di evitare di ricercare tutti i permessi di cui dispone un utente.

RBCA è usato in molti servizi/organizzazioni. Può comunque soffrire di problemi di scalabilità e molto spesso si rischi di assegnare un ruolo troppo elevato per l'utente (o creargli un ruolo ad-hoc).

Un altro problema dei modelli in generale è che gli utenti possono abusare dei loro privilegi.

### 10.1.3 Attribute Based Access Control

Supponiamo di avere uno scenario in dipendente può accedere ai dati dei clienti solo se si trova fisicamente nella città dove ha sede l'organizzazione. Con un modello RBAC non sono in grado di specificare questo tipo di politica. Nel caso migliore dovrei specificare un ruolo per l'impiegato che gli dà accesso solo e soltanto alle info personali dei clienti, ma non posso porre come condizione la posizione fisica dell'utente.

ABAC permette di specificare politiche molto specifiche e di garantire autorizzazioni dinamiche, in quanto posso semplicemente modificare il valore degli attributi specificati nelle condizioni delle politiche di controllo dell'accesso per cambiare l'effetto della politica.

Se ad esempio ho un impiegato che lavora in un determinato progetto e poi viene trasferito ad altro progetto all'interno dell'organizzazione, posso semplicemente cambiare la politica imponendo una condizione diversa sull'attributo "progetto". Posso così evitare situazioni in cui ho dipendenti che dispongono di permessi di cui non hanno bisogno.

ABAC viene utilizzato in molte organizzazioni dove è necessario specificare condizioni precise e dinamiche.

## 10.2 XACML

Lo standard che implementa l'Attribute Based Access Control è l'eXtensible Access Control Markup Language (XACML):

- Fornisce un linguaggio per specificare le politiche di controllo dell'accesso come condizioni sugli attributi del soggetto richiedente, della risorsa o del contesto in cui si richiede l'accesso;
- Fornisce un linguaggio per specificare la richiesta di accesso da parte dell'utente e la risposta sull'accesso;
- Fornisce un'architettura che identifica le componenti software che bisogna implementare per decidere, sulla base delle politiche sul controllo dell'accesso, se una richiesta deve essere accettata o negata;
- Fornisce algoritmi per valutare la richiesta.

L'architettura è formata da quattro elementi fondamentali:

- Policy Enforcement Point (PEP), che intercetta le richieste di accesso da parte dell'applicazione dell'utente e le redirige al POP;

- Policy Decision Point (POP), che valuta la richiesta secondo le policy di accesso decise e gli attributi ricevuti;
- Policy Amministration Point (PAP), che è usata per configurare le politiche di controllo dell'accesso da parte dell'amministratore di sistema;
- Policy Information Point (PIP), che fornisce gli attributi sul soggetto, sulla risorsa o sul contesto al PDP.

Come avviene l'interazione tra le componenti:

- L'amministratore inserisce le politiche sul controllo dell'accesso tramite il PAP;
- L'utente richiede l'accesso ad una determinata risorsa;
- La richiesta viene intercettata dal PEP, che la redirige al Context Handler (altro componente sw). Non è sempre presente, ma viene usato quando le richieste non vengono formulate direttamente nel linguaggio supportato dallo standard. Il CH traduce la richiesta e la invia al PDP;
- Il PDP può richiedere al CH di fornirgli gli attributi necessari per decidere se garantire o meno la richiesta. Il CH contatta il PIP che li recupera (ad esempio dal sistema di gestione dell'identità digitale dell'azienda);
- Il PIP decide se concedere l'accesso oppure no basandosi sugli attributi e le condizioni della policy e restituisce la risposta al PEP. Nella risposta è possibile includere le Obbligazioni, che sono azioni che devono essere obbligatoriamente eseguite dal PEP nel momento in cui garantisce l'accesso al soggetto (es: inviare una mail all'owner della risorsa per notificare l'accesso, creare un log per tenere traccia dell'accesso, ...).

Esistono più modi in cui è possibile implementare le quattro componenti. Per esempio è possibile avere uno scenario centralizzato, in cui le componenti sono posizionate all'interno della stessa organizzazione. Qui potrei avere quattro server in cui girano ciascuna delle componenti.

Posso avere anche il caso in cui le componenti sono tutte in outsourcing, come negli scenari cloud. Qui invece di implementare all'interno della rete dell'organizzazione, il PDP e il PEP sono forniti dal cloud provider. L'unica possibilità che viene data all'organizzazione è quella di configurare le politiche che regolano l'accesso alle risorse.

### 10.2.1 Specifica delle politiche

Esistono due componenti principali per specificare le politiche:

- L'elemento policy;
- Le regole.

Una policy che si applica ad una determinata risorsa è costituita da una o più regole. Ciascuna regola non può però essere valutata da sola, ma deve essere valutata come elemento policy (non ci sarà mai una decisione che si basa su una regola).

```
<Policy>
  <Target>
    <Resources>
    <Subjects>
    <Actions>
  </Target>
```

```

<RuleSet ruleCombiningAlgId = DenyOverrides >
  <Rule ruleId= R1 >
  <Rule ruleId= R2 >
  ...
  <Obligations>
    <RuleSet>
  </Policy>

<Rule RuleId= R1    Effect= Permit >
  <Target>
    <Resources>
    <Subjects>
    <Actions>
    <Condition>
  </Rule>

```

Ciascuna regola dispone di un elemento ‘<target>’ che associa una risorsa richiesta ad una policy applicabile. Può contenere uno o più di questi elementi:

- <match> che specifica le condizioni rispetto al soggetto, alla risorsa e alle azioni;
- <AnyOf> che funziona come OR logico per le condizioni, quindi mi basta soddisfare una sola condizione specificata in ‘<match>’;
- <AllOf> che indica che tutte le condizioni specificate in <match> devono essere soddisfatte.

```

<Target>
  <AnyOf>
    <AllOf>
      <Match>conditionA</Match>
    </AllOf>
    <AllOf>
      <Match>conditionB</Match>
    </AllOf>
  </AnyOf>
</Target>

```

### 10.2.2 Algoritmi di rule-combining

Questi algoritmi specificano come i risultati derivanti dalla valutazione delle regole devono essere combinati per valutare la policy. Esistono quattro tipi di algoritmi standard:

- **Deny-overrides**: se una regola ritorna *deny* allora la valutazione della politica sarà anch'essa *deny* e quindi l'accesso sarà negato;
- **Permit-overrides**: se una regola ritorna *permit* allora la valutazione della politica sarà anch'essa *permit* e quindi l'accesso sarà consentito;
- **First-applicable**: ogni regola viene valutata nell'ordine in cui è listata nella policy. Per una particolare regola, se il <target> mettta la <condition> valutata a Permit, Deny o Indeterminate, allora tale risultato sarà anche il risultato della policy;
- **First-applicable**: applicabile solo a PolicySet:

1. Se nessuna Policy è applicabile, allora il risultato è NotApplicable;
2. Se una o più policy sono applicabili allora il risultato è Indeterminate;
3. Se solo una policy è applicabile allora il risultato è il risultato della valutazione di quella policy.

# Capitolo 11

## Risk Management

Il risk management è il processo che consiste nell'identificare, analizzare e valutare il rischio. Si tratta dell'unico modo per assicurare che i controlli di cybersecurity scelti siano appropriati ai rischi che l'organizzazione affronta. Senza risk assessment ad informare le scelte di cybersecurity, c'è il rischio di sprecare tempo, sforzo e risorse. Ha poco senso infatti implementare misure di sicurezza contro minacce improbabili che accadano o che abbiano impatto materiale limitato sull'organizzazione, senza contare che vi è il rischio di sottostimare o ignorare rischi che potrebbero effettivamente colpire l'organizzazione.

Le organizzazioni devono decidere quanto tempo e denaro spendere proteggere la loro tecnologia e i loro servizi. Uno degli obiettivi principali della gestione del rischio è quello di informare e migliorare queste decisioni. In base al settore in cui l'azienda opera, il risk management può essere un obbligo. Ad esempio, un'organizzazione che vuole la certificazione ISO 27001, avere una strategia di risk management è uno dei requisiti chiave (è richiesto anche da GDPR).

Un processo di risk management si compone di più fasi:

- Nella fase iniziale, l'azienda che lo mette in atto deve definire la strategia per il risk management, dove vengono definiti la metodologia usata dall'azienda, le componenti essenziali dell'organizzazione che ne devono essere soggette (es: informazione, processi che elaborano i servizi), chi sono gli stakeholders del sistema;
- Nella fase cardine, quella di risk assessment, si identificano, analizzano e valutano i rischi;
- Nella fase di Risk Treatment l'organizzazione, per ciascun dei rischi individuati, vengono identificate le misure di protezione. In questa fase non bisogna tenere presente solo il rischio però, ma anche le risorse disponibili (economiche, personale, ...). Alla fine del processo esisterà sempre un rischio minimo che non è stato mitigato. L'azienda dovrà quindi decidere se accettare il rischio, per continuare a fornire i suoi servizi, monitorare il rischio in attesa di cambiamenti oppure trasferire il rischio a terzi (assicurazione);
- I rischi identificati vanno infine comunicati agli stakeholder del sistema;
- Nella fase di monitor and review le misure di protezione selezionate e implementate vengono monitorate per garantire che forniscano sempre un livello di protezione adeguato rispetto agli attacchi identificati nella fase di risk assessment. Questo può essere fatto con penetration testing o valutando le performance delle misure adottate intervistando gli stakeholders. In base al risultato di questa fase, potrebbe essere necessario ripartire da capo. L'attività di revisione del rischio può anche essere fatta qualora l'azienda decide di introdurre nuove funzionalità/sistemi o raccogliere nuove informazioni. Ad esempio,

se l'azienda decide di spostare i dati dei clienti dai server locale a un servizio cloud, è necessario rifare il processo da capo

Il processo di risk management è un processo ciclico.

## 11.1 Standard NIST

Il NIST fornisce uno standard per le fasi di risk management viste. Il vantaggio di usare gli standard del NIST è che sono open-source.

Tra gli standard che fornisce troviamo:

- Lo standard 800-39 che ci dà un'introduzione generale ai concetti alla base del processo di information management;
- Lo standard 800-37 che ci descrive come implementare un programma di risk management;
- Lo standard 800-30 che ci descrive il processo di risk assessment;
- Lo standard 800-53 che riguarda la fase di risk treatment;
- Lo standard 800-53a che descrive come valutare l'efficacia dei security controls identificati nella fase di risk treatment;
- Lo standard 800-39 che descrive come classificare le informazioni per il risk assessment, se il focus dell'azienda consiste in quelle;

Oltre gli standard del NIST ne esistono altri:

- ISO/IEC IS 27005 che fornisce un processo generale per il risk management, non legato specificatamente alla sicurezza;
- ISO 31000, un altro standard generale;
- Altre metodologie nazionali.

Esistono anche metodologie legate solo al risk assessment: OCTAVE, CORAS (basata solo su grafici), EBIOS, CRAMM, ...

### 11.1.1 Modellazione del rischio

Quando parliamo di rischio dobbiamo definire i fattori che andiamo ad usare per quantificare il rischio. Quasi tutte le metodologie parlano di rischio rispetto all'asset. Gli asset contengono vulnerabilità. I threat actor iniziano uno o più threat event che vanno a creare un threat scenario, un attacco che consiste di più fasi.

Il rischio è dato da due componenti: la probabilità che avvenga e l'impatto che questo potrebbe avere su uno o più asset dell'organizzazione.

Vediamo uno scenario generico: il rischio si materializza quando abbiamo un threat agent che inizia un attacco sfruttando delle vulnerabilità che risulta in un impatto negativo per l'organizzazione. Il rischio viene dato dall'impatto negativo e dalla likelihood. Questa è composta dalla probabilità che l'attacco avvenga e dalla probabilità che questo ha di impattare (11.1).

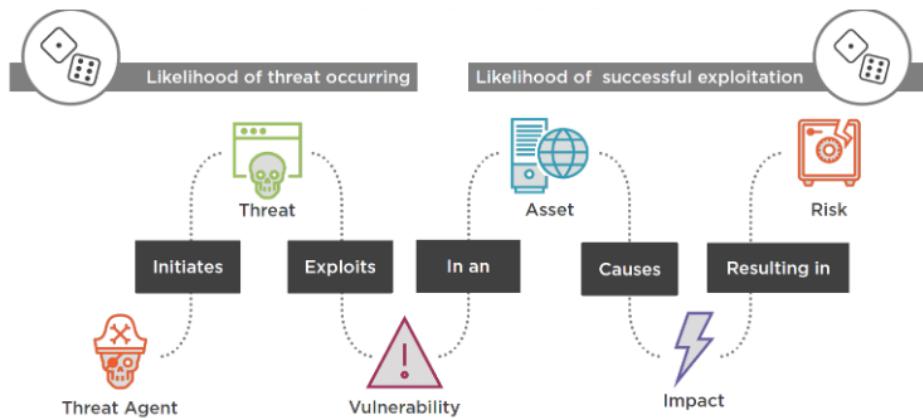


Figura 11.1: Modello di rischio.

**OWASP Risk Rating Methodology** Fornisce una serie di fattori per stimare la probabilità e l'impatto. La probabilità è stimata sulla base di due serie di fattori:

- Fattori dell'agente di minaccia;
- Fattori di vulnerabilità.

L'impatto è stimato sulla base di:

- Fattori di impatto tecnico;
- Fattori di impatto sul business.

## 11.2 Risk assessment secondo lo standard NIST

Supponiamo di avere Poste Italiane che permette di eseguire operazioni di online banking, sia tramite app online che app da cellulare. Le informazioni vengono memorizzate da un online banking service, a cui le due app accedono. Per accedere ai servizi, i clienti devono utilizzare un username e una password.

Per identificare i rischi di questo scenario usiamo il risk assessment definito nello standard NIST 800-30. Lo standard si compone di 4 fasi:

1. Raccolta di tutti documenti necessari per fare l'analisi del rischio;
2. Identificazione dei rischi e valutazione della loro severità in base alla likelihood;
3. Comunicazione dei risultati del risk assessment al CEO, CESO e manager, che decidono quali rischi mitigare;
4. Revisione periodica del risk assessment, per adeguarlo ai cambiamenti.

### 11.2.1 Preparazione al risk assessment

L'obiettivo di questa fase è definire il contesto per il risk assessment. Si compone di quattro task:

1. Se determina il perché si sta facendo l'assessment. Esistono due possibilità:

- Si tratta del primo assessment. Vogliamo stabilire una baseline dei possibili attacchi e conseguenti rischi (attaccanti, vulnerabilità, come possono essere sfruttate le vulnerabilità, ...);
  - Stiamo revisionando un assessment precedente per revisionare i rischi a cui è soggetto l'organizzazione. Viene fatto per due motivi:
    - Lo scenario è cambiato, sono emersi nuovi attacchi a cui è soggetta l'organizzazione (es: è stata scoperta la vulnerabilità log4j);
    - è stata rilasciata una nuova applicazione, o una precedente è stata ad esempio rilasciata su cloud, e vogliamo valutare quali sono i rischi associati.
2. Si definisce lo scope dell'assessment, ovvero quali parti dei sistemi gestiti dall'organizzazione sono soggetti al risk assessment (target dell'analisi - app sul cloud, determinate informazione e servizi associati);
  3. Si documentano quali sono le assunzioni fatte relative all'analisi. Bisogna recuperare qual è la strategia aziendale rispetto al risk assessment. Questa comprende la risk tolerance (rischio che l'organizzazione è disposta a tollerare sopra che le misure di protezione identificate dal risk assessment sono state implementate) e risk acceptance. Si definiscono anche la metodologia per effettuare il risk assessment e quali sono le categorie di attaccanti, attacchi e vulnerabilità rilevanti per l'organizzazione. Si definiscono infine i valori da usare per quantificare la likelihood di un attacco e dei possibili impatti negativi che possono verificarsi;
  4. Si definisce il risk model, ovvero i fattori che usiamo per quantificare il rischio (generalmente i modelli usano threat, vulnerability, ...). Si valuta anche l'approccio per valutare il rischio. Esistono tre approcci principali:
    - Approccio qualitativo: si esprime la probabilità dell'impatto usando delle etichette come probabilità alta, media, bassa o impatto alto, medio, moderato;
    - Approccio quantitativo: likelihood e impatto vengono espressi numericamente;
    - Approccio ibrido: usa i numeri per quantificare il valore associato a likelihood e impatto e presenta tali risultati poi con delle etichette. SI definisce infine l'approccio di analisi. Questo dipende dalla metodologia che si utilizza (alcune metodologie di risk assessment partono dall'asset e risalgono a tutti i possibili attacchi o che partano dall'individuare i possibili attaccanti e legano le possibili categorie trovate agli asset del sistema).

### 11.2.2 Esecuzione del risk assessment

L'obiettivo di questa fase è produrre una lista di possibili attacchi col proprio livello di rischio, ordinati per tale livello (quelli con priorità più alta saranno in cima alla lista). Per identificare questi rischi questa fase si articola in 6 parti:

- **Identificazione delle categorie di attaccanti:** il NIT identifica quattro categorie di attaccanti:
  - Adversarial: comprende cybercriminal, nation state, insider threat e in generale tutti gli attaccanti esterni all'organizzazione;
  - Accidental: comprende tutti gli utenti che compiono un'azione che ha un effetto negativo sul sistema ma non con scopo malevolo (es: amministratore che setta male la policy di accesso ad una risorsa);

**TABLE G-2: ASSESSMENT SCALE – LIKELIHOOD OF THREAT EVENT INITIATION (ADVERSARIAL)**

Qualitative Values	Semi-Quantitative Values	Description	
Very High	96-100	10	Adversary is <b>almost certain</b> to initiate the threat event.
High	80-95	8	Adversary is <b>highly likely</b> to initiate the threat event.
Moderate	21-79	5	Adversary is <b>somewhat likely</b> to initiate the threat event.
Low	5-20	2	Adversary is <b>unlikely</b> to initiate the threat event.
Very Low	0-4	0	Adversary is <b>highly unlikely</b> to initiate the threat event.

Qualitative Values	Semi-Quantitative Values	Description	
Very High	96-100	10	Error, accident, or act of nature is <b>almost certain</b> to occur; or occurs <b>more than 100 times a year</b> .
High	80-95	8	Error, accident, or act of nature is <b>highly likely</b> to occur; or occurs <b>between 10-100 times a year</b> .
Moderate	21-79	5	Error, accident, or act of nature is <b>somewhat likely</b> to occur; or occurs <b>between 1-10 times a year</b> .
Low	5-20	2	Error, accident, or act of nature is <b>unlikely</b> to occur; or occurs <b>less than once a year, but more than once every 10 years</b> .
Very Low	0-4	0	Error, accident, or act of nature is <b>highly unlikely</b> to occur; or occurs <b>less than once every 10 years</b> .

Figura 11.2: Tabelle NIST likelihood singolo.

**TABLE G-4: ASSESSMENT SCALE – LIKELIHOOD OF THREAT EVENT RESULTING IN ADVERSE IMPACTS**

Qualitative Values	Semi-Quantitative Values	Description	
Very High	96-100	10	If the threat event is initiated or occurs, it is <b>almost certain</b> to have adverse impacts.
High	80-95	8	If the threat event is initiated or occurs, it is <b>highly likely</b> to have adverse impacts.
Moderate	21-79	5	If the threat event is initiated or occurs, it is <b>somewhat likely</b> to have adverse impacts.
Low	5-20	2	If the threat event is initiated or occurs, it is <b>unlikely</b> to have adverse impacts.
Very Low	0-4	0	If the threat event is initiated or occurs, it is <b>highly unlikely</b> to have adverse impacts.

Likelihood of Threat Event Initiation or Occurrence	Likelihood Threat Events Result in Adverse Impacts				
	Very Low	Low	Moderate	High	Very High
Very High	Low	Moderate	High	Very High	Very High
High	Low	Moderate	Moderate	High	Very High
Moderate	Low	Low	Moderate	Moderate	High
Low	Very Low	Low	Low	Moderate	Moderate
Very Low	Very Low	Very Low	Low	Low	Low

Figura 11.3: Tabelle NIST likelihood congiunto.

Type of Impact	Impact
Harm to Operations	<ul style="list-style-type: none"> <li>▪ Inability to perform current business functions</li> <li>▪ Non compliance</li> <li>▪ Direct Financial Costs</li> <li>▪ Damage to image of reputation</li> </ul>
Harm to Assets	<ul style="list-style-type: none"> <li>▪ Damage to or loss of physical facilities</li> <li>▪ Damage to or loss of information systems or networks</li> <li>▪ Damage to or loss of equipment</li> <li>▪ Damage to or loss of information assets</li> <li>▪ Loss of intellectual properties</li> </ul>
Harm to Individuals	<ul style="list-style-type: none"> <li>▪ Loss of life</li> <li>▪ Identity Theft</li> <li>▪ Loss of PII</li> <li>▪ Damage to the reputation</li> </ul>
Harm to Other Organizations	<ul style="list-style-type: none"> <li>▪ Non compliance</li> <li>▪ Direct Financial Costs</li> <li>▪ Damage to image of reputation</li> </ul>
Harm to the Nation	<ul style="list-style-type: none"> <li>▪ Damage to a critical infrastructure</li> </ul>

Impact	Description
<b>Very High</b>	Threat event could have <b>multiple severe or catastrophic adverse effects</b> on organizational operations, assets, individuals, other organizations or the Nation
<b>High</b>	Threat event could have <b>severe or catastrophic adverse effects</b> on organizational operations, assets, individuals, other organizations or the Nation
<b>Moderate</b>	Threat event could have <b>serious effects</b> on organizational operations, assets, individuals, other organizations or the Nation
<b>Low</b>	Threat event could have <b>limited effects</b> on organizational operations, assets, individuals, other organizations or the Nation
<b>Very Low</b>	Threat event could have <b>negligible</b> on organizational operations, assets, individuals, other organizations or the Nation

Figura 11.4: Tabelle NIST impatto.

Adverse Impact	Likelihood of Threat Event				
	Very Low	Low	Moderate	High	Very High
Very High	Low	Moderate	High	Very High	Very High
High	Low	Moderate	Moderate	High	Very High
Moderate	Low	Low	Moderate	Moderate	High
Low	Very Low	Low	Low	Moderate	Moderate
Very Low	Very Low	Very Low	Low	Low	Low

Figura 11.5: Tabella rischio.

- Strutturale: comprende i casi in cui si verificano guasti al sistema o software;
- Ambientale: comprende tutti gli attacchi relativi disastri naturali (es: incendi, terremoti, ...).

Una volta identificate le categorie, dobbiamo definire le loro "capacità":

- Per gli adversarial, quali sono le loro capacità, quanto sono motivate a trovare un exploit, quali risorse stanno targettando;
  - Per i non adversarial, quali sono i loro possibili effetti.
- **Identificazione dei threat event:** determina quali eventi possono essere compiuti dagli attaccanti identificati;
  - **Identificazione delle vulnerabilità:** determina le vulnerabilità che un threat event può exploitare e la loro severità;
  - **Determinazione della likelihood:** per stimare la likelihood del threat event si considerano capacità, intento e difficoltà dell'exploit. Il NIST fornisce una serie di tabelle per quantificare il likelyhood (11.2, 11.3);
  - **Determinazione dell'impatto:** identifica il potenziale danno causato agli asset dell'organizzazione. Il NIST fornisce una serie di tabelle per quantificare l'impatto (11.4);
  - **Determinazione del rischio:** determina il livello di rischio come combinazione di likelihood e impatto (11.5). In base al livello, l'azienda può decidere di non fare nulla (low, very low), monitorare il rischio senza selezionare ancora nessun security control o politica aziendale (moderate), mitigare il rischio selezionando una soluzione da implementare (high, very high).

### 11.2.3 Comunicazione dei risultati

Questa fase consiste nella produzione di un report al fine di comunicare i risultati. Lo scopo principale è comunicare i rischi più rilevanti al management e gli altri stakeholder del sistema, nonché al personale appropriato nell'organizzazione (es: sezione IT per vulnerabilità log4j).

#### **11.2.4 Mantenimento del risk management**

L'ultima fase ha lo scopo di valutare quando rieseguire il risk assessment. Questo permette alle organizzazioni di:

- Determinare l'efficacia delle risposte al rischio implementate;
- Identificare cambiamenti ai rischi che minacciano l'organizzazione;

# Capitolo 12

## Introduzione alla privacy

Tutto ciò che facciamo online lascia una traccia che è memorizzabile e analizzabile da chi tiene traccia di chi raccoglie i dati personali.

**Surveillance Capitalism** Generalmente, tutte le informazioni raccolte su di noi vengono date in pasto ad algoritmi di machine learning che le usano per predire il nostro comportamento. In questo modo, i venditori di servizi possono proporci prodotti in linea con i nostri interessi.

**Sorveglianza dei governi** Esistono anche altri tipi di surveillance, come quello inglese/americano che raccoglieva dagli IPS i dati sulle comunicazioni internet nei due paesi e le usava per ottenere informazioni sui rispettivi cittadini.

**Data breach** Gli enti che raccolgono queste informazioni sono soggetti ad attacchi di data breach. Gli attaccanti, ad esempio, una volta ottenuti i dati di una certa persona potrebbero sfruttarli per effettuare attacchi di phishing spacciandosi per quella persona.

**Iniziative a favore della privacy** Negli ultimi anni sono nate diverse iniziative volte a tutelare/garantire la privacy, come il GDPR, Tor, Privacy Badger (previene l'online tracking), Dp3t (protocollo sviluppato da centri di ricerca europei con lo scopo di consentire il tracciamento dei contagi senza rivelare identità degli utenti coinvolti).

### 12.1 Definizione di privacy

Non è semplice definire la privacy in quanto è un concetto che può avere un significato diverso per ciascun individuo. Nel tempo sono state date delle definizioni più "adottate" di altre:

- "Il diritto ad essere lasciato solo";
- "Il diritto dell'individuo di decidere quali e quando le proprie informazioni devono essere pubbliche";
- "La libertà da vincoli irragionevoli sulla costruzione della propria identità" (se un utente sa che le sue conversazioni sono monitorate, probabilmente limiterà le informazioni scambiate/scrivereà in modo diverso);
- Tassonomia dei pericoli alla privacy: piuttosto che definire la privacy, definisce tutte quelle attività che portano alla compromissione della privacy;

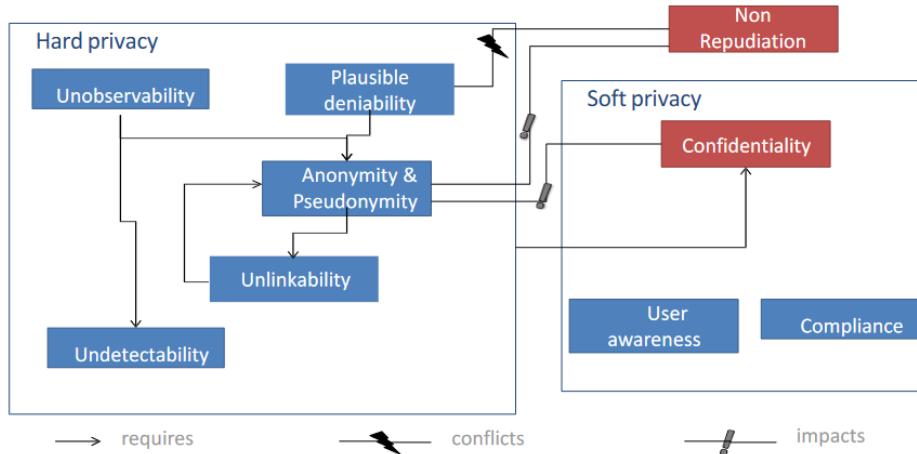


Figura 12.1: Proprietà delle due definizioni.

- "Privacy come integrità contestuale": lega la definizione di privacy alla nozione di contesto (in un contesto potrebbe essere appropriato condividere info personale, in altre no. L'appropriatezza è data da norme);
- Trasparenza, finalità, proporzionalità, responsabilità (dal GDPR):
  - Trasparenza: quando un data controller raccoglie i dati, deve specificare per quale motivo li raccoglie e come li tratterà o condividerà.
  - Purpose: deve specificare il motivo per cui li raccoglie
  - Proporzionalità: devono essere proporzionali e finalizzati al purpose; non devo raccogliere più dati del necessario
  - Accountability: il data controller deve mantenere traccia di chi ha accesso ai dati.

## 12.2 Proprietà della privacy

Le proprietà della privacy vengono suddivise in base a due macro definizioni di privacy:

- Hard privacy: parte dall'assunzione che il data controller non è un'entità di cui ci si può fidare e quindi l'individuo deve condividere con lui il minor numero di dati sensibili. Sarà l'utente a dover adottare tutte le tecniche per proteggere la propria privacy (es: cifratura);
- Soft privacy: l'utente si fida del data controller e quindi è lui a dover adottare tutte le tecniche possibili per minimizzare i rischi per l'utente. Ovviamente in questo caso è più difficile per l'utente avere controllo su come i suoi dati vengono processati o suddivisi dal data controller.

In figura sono riportate le proprietà definite per le due definizioni.

Le proprietà sono state definite rispetto ad un preciso modello di attaccante: abbiamo un sistema dove diversi attori possono compiere azioni e un attaccante che vuole monitorare le azioni eseguite dagli utenti. In particolare, l'obiettivo dell'attaccante è inferire gli **item of interest**, informazioni di interesse per l'attaccante (es: contenuto del messaggio, chi è il mittente, chi è il ricevente, ...).

### 12.2.1 Anonymity

Abbiamo anonimità quando un attaccante non può identificare l'attore all'interno di un gruppo di attori (**anonymity set**). Praticamente abbiamo anonimità quando, all'interno del sistema, non deve essere possibile legare l'identità di uno degli attori ad un determinato item of interest.

Ad esempio, un attaccante non riesce a determinare quale attore ha spedito o ricevuto un messaggio.

All'interno di una rete, possiamo avere due anonymity set: l'insieme dei possibili mittenti e l'insieme dei possibili riceventi del messaggio.

Spesso il concetto di anonimità è legato a quello di **pseudoanonimità**: invece di usare il vero nome per identificare un soggetto si usa un ID casuale. Questo comporta che, se un utente usa sempre lo stesso ID, siamo in grado di tracciare tutte le azioni che ha effettuato.

### 12.2.2 Unlinkability

Nasconde la presenza di un collegamento tra due item of interest. Se la proprietà è soddisfatta, un attaccante non deve essere, ad esempio, in grado di determinare se due messaggi appartengono alla stessa sessione, oppure se sono stati mandati dallo stesso utente.

L'unlinkability può essere usata anche per definire l'anonymity. Quest'ultima non altro che un'unlinkability tra due item of interest: l'identità dell'utente e le azioni che questo ha compiuto all'interno del sistema

### 12.2.3 Undetectability

Sotto questa proprietà cadono due sotto-proprietà:

- Undetectability: se un attaccante osserva la rete non è in grado di determinare se un dato item of interest esiste oppure no. Supponiamo che il nostro item of interest sia un messaggio, l'attaccante non deve essere in grado di distinguere tra messaggio inviato e rumore casuale sulla rete;
- Unobservability: assume che l'undetectability valga per tutti gli utenti della rete non coinvolti nello scambio del messaggio. Inoltre vuole l'anonymità di ciascun soggetto coinvolto nell'item of interest rispetto agli altri (il mettente non sa chi sarà il ricevente e il ricevente non sa chi è il mittente del messaggio -> in un database medico non possiamo dire se un record esiste; non possiamo dire se una persona ha visitato un dato sito web);

### 12.2.4 Plausible deniability

Va in contrasto con la non repudiation. Garantisce che un utente possa negare di aver visitato una data pagina web, aver spedito una data email o aver detto qualcosa.

Questa proprietà può essere desiderabile, ad esempio, nei sistemi di online voting. Al contrario, in siti di e-commerce, è desiderabile il contrario, quindi la non-repudiation.

### 12.2.5 Confidentiality

Prevede che sia responsabilità del data controller proteggere i dati da un accesso non autorizzato. In pratica si traduce nella cifratura dei dati o nell'utilizzo di meccanismi di controllo dell'accesso.

### 12.2.6 Compliance

È legata al rispettare o principi dettati dalle leggi sulla protezione dei dati personale, come i principi dettati dal GDPR.

### 12.2.7 Awareness

Più legata all'utente che al data controller, prevede che l'utente sia messo a conoscenza della conseguenza del condividere informazioni (es: l'utente posta la foto della carta di credito online senza nascondere i codici). Si focalizza sul garantire che, quando l'utente condivide i propri dati personali, lo faccia in maniera coscienziosa.

## 12.3 Minacce alla privacy

Solove si è focalizzato su come la privacy possa essere compromessa. Dalla sua analisi, è riuscito a definire quattro categorie di azioni che possono compromettere la privacy. Tre di queste sono mappate sulle azioni che un data controller esegue quando raccoglie dati su un individuo:

1. Information collection, che consiste nella raccolta di dati dell'utente;
2. Information processing, che consiste nell'elaborazione dei dati raccolti;
3. Information dissemination, che consiste nella condivisione dei dati elaborati con parti terze;
4. Invasion: non legata generalmente al data controller, racchiude tutte quelle situazioni in cui la sfera privata dell'individuo viene compromessa.

### 12.3.1 Information collection

Comprende due tipi di attacchi:

- Surveillance: le attività che un individuo compie vengono osservate e registrate. Alcuni esempi possono essere:
  - Contatori smart: danno informazioni più precise, riducendo quindi i costi, ma allo stesso tempo permettono di inferire info sull'utente, come quante docce si fa.
  - Angry birds era usata dalla NSA per raccogliere info.
- Interrogation: le informazioni sono estorte dall'individuo (lo si convince a fornire informazioni che normalmente non fornirebbe). Esempi possono essere:
  - Probing: attacchi di phishing che richiedono agli utenti di resettare login e password.

### 12.3.2 Information processing

Comprende cinque tipi di attacchi:

1. **Aggregation:** il data controller raggruppa le informazioni su un individuo da diverse sorgenti, e impara nuove informazioni che l'individuo non pensava sarebbero state inferite. Esempi:

- Target ha predetto che una cliente era incinta (prima che lei lo sapesse) tramite le transazioni catturate dalla "carta fedeltà" e ha iniziato a inviare coupon su prodotti per la maternità.
2. **Identification:** il data controller raggruppa una serie di informazioni e riesce a trovare l'identità del soggetto.
- Un'azienda americana è riuscita a raccogliere i dati sui click di un individuo su vari siti e dall'analisi di questi dati è riuscita a determinare nome e cognome dell'utente.
3. **Insecurity:** il data controller non adotta le misure di sicurezza necessarie per prevenire un accesso non autorizzato.
4. **Secondary use:** il DC raccoglie i dati per uno scopo e poi li riutilizza per qualcos'altro senza informare il soggetto. Esempi;
- Un medico raccoglie dati sul paziente li fornisce ad una casa farmaceutica;
  - Cambridge analytica: genera profili psicologici sugli utenti di facebook poi usati per influenzare la campagna elettorale in America.
5. Esclusione: l'utente non ha visione di quali sono i dati raccolti dal data controller (non c'è trasparenza).

### 12.3.3 Information dissemination

Comprende sette possibili attacchi:

1. **Breach of confidentiality:** si verifica un accesso non autorizzato ai dati mantenuti dal data controller. Esempi:
  - Equifax: società di recupero crediti che analizzava il credit store (valuta la capacità di un individuo a pagare un debito) dei cittadini americani e inglesi. Questi dati sono state rese pubbliche.
2. **Disclosure:** info su una data persona vengono divulgate. Queste info possono influenzare le opinioni delle persone su quell'individuo. Esempi:
  - Icloud nel 2014 leakka immagini di attrici americane in situazioni intime. Le vittime avevano ricevuto una mail di phishing in cui si chiedeva di resettare la password di iCloud.
3. **Exposure:** vengono rese pubbliche nudità o dolori di una persona;
4. **Increased accessibility:** si aumenta l'audience con la quale si condividono le informazioni;
5. **Blackmail:** si minaccia di diffondere dati personali di un individuo se non viene pagato un riscatto. Esempi:
  - Ransomware;
6. **Appropriation:** i dati personali di una persona vengono usati per impersonarla. Esempi:
  - Reti sociali (es: facebook): molte info personali vengono condivise e possono essere usate per impersonarci verso servizi online.

7. **Distortion:** si diffondono info false per cambiarne l'opinione degli altri su quella persona. Esempi:

- Troll;

### 12.3.4 Invasion

Comprende tre attacchi:

1. **Intrusion:** situazioni dove altri individui o governi interferiscono con la vita privata di una persona. Esempi:
  - Reti sociali e stalking: femminicidi nonostante gli ordini restrittivi. Il colpevole stalkera la vittima tramite i social;
  - Cyberbullismo.
2. **Decisional interference:** comporta l'incursione del governo nelle decisioni dell'interessato riguardanti il suo privato.

## 12.4 Privacy Enhancing Technologies (PETS)

Con Privacy Enhancing Technologies intendiamo tutti quei tool, tecnologie e software che ci permettono di proteggere la privacy di un individuo, minimizzandone i rischi. Queste tecnologie possono essere applicate sia dal data subject (utente) oppure dalle organizzazioni che raccolgono/analizzano i dati. Possono essere applicate a diversi livelli: rete, browser, database.

Esistono diverse categorie di tecnologie.

### 12.4.1 Data Protection technologies

Hanno lo scopo di garantire la proprietà di compliance con i principi di protezione dei dati stabiliti dal GDPR. Tipicamente sono adottate dal data controller. Alcuni esempi possono essere:

- Cifratura dei dati, sia durante la trasmissione che durante la memorizzazione;
- Controllo dell'accesso in base allo scopo per cui i dati sono stati raccolti (si può accedere al dato solo se lo scopo è in linea al motivo per cui sono stati raccolti);
- Sistemi di log, per garantire l'accountability;
- Fornire un'interfaccia all'utente che permette di cancellare i dati;
- Autenticazione e autorizzazione dei lavoratori che accedono ai dati.

Si assume che il data controller sia fidato, mentre non ci si fida di tutti le entità esterne che potrebbero accedere ai dati. Non c'è garanzia che il data controller usi i dati per scopi diversi da quelli per cui sono stati raccolti.

#### 12.4.2 User awareness technologies

Insieme di tecnologie il cui scopo è dare all'utente il controllo su quali dati personali vengono raccolti e come vengono usati dal data controller oppure aiutare l'utente a configurare la propria privacy in modo da minimizzare attacchi che possano portare alla divulgazione dei dati personali. Alcuni esempi possono essere:

- Privacy settings by defaults: quanto creiamo un account, se chi ha sviluppato il sito ha rispettato i principi di privacy by design e privacy by default, dovrebbe adottare delle impostazioni che proteggano i dati in automatico;
- Tool per capire con chi stiamo condividendo le informazioni: facebook permetteva di vedere quello che gli altri vedevano del nostro profilo;
- Privacy policy comprensibili: sistemi a layer che permettono di andare ad approfondire ogni aspetto della politica;
- Privacy nudges: permettono di vedere quante volte una info personale è stata condivisa e con chi.

#### 12.4.3 Anonymity technologies

Hanno l'obiettivo di soddisfare l'anonymità. Alcuni esempi possono essere:

- K-Anonymity, per i database;
- Sistemi di comunicazione anonima, come Tor;
- Sistemi che garantiscono l'anonymità durante l'autenticazione: per garantire la privacy anche nei confronti dell'Identity Provider, che ha visioni su tutti gli Identity Attributes, è stato ideato il sistema di autenticazione Idemix, basato sulla zero-knowledge.

#### 12.4.4 Altre tecnologie

- Memorizzazione sicura sul cloud: PrivateStorage fornisce una cifratura lato client. I dati sono cifrati prima di uploadare e la chiave simmetrica è conosciuta solo dal client;
- Searchable encryption: consente di ricercare se un documento cifrato contiene determinate parole chiave e di recuperare questi documenti che poi vengono decifrati dal client. Si usa quando l'utente cifra i dati prima di uploadarli. Si evita di dover scaricare tutti i dati, decifrarli e ricercare il documento che serve;
- Computazioni su dati cifrati: Homomorphic encryption (poco usata in quanto poco efficiente) o Secure multiparty computation (un insieme di utenti che vogliono calcolare una determinata funzione sulla base di input di ciascun utente, ma vogliono anche mantenere privati gli input).



## Capitolo 13

# Introduzione alla data protection

Dal 25 maggio 2018 è entrato in vigore il regolamento europeo per la protezione dei dati personali, che ha cambiato gli obblighi che chi raccoglie dati personali di cittadini UE deve rispettare. Lo scopo del regolamento è quello di armonizzare le leggi dei vari paesi UE in merito alla protezione dei dati.

Il GDPR è andato a sostituire la direttiva europea 95/46, che dettava quali erano i principi relativi alla protezione dei dati personali, ma non era direttamente applicabile. Ogni stato europeo aveva una sua legge nazionale per la protezione dei dati, e quindi i dati potevano essere trattati diversamente in base al paese dell'individuo.

L'altro cambiamento che è stato introdotto è come viene gestito il controllo che i principi vengano rispettati. Inizialmente era presente il Working Party 29, ora sostituito dall'European Data Protection Board (ESPB), composto da tutte le data protection authority di ogni paese europeo. In Italia abbiamo il Garante della privacy.

Il GDPR ha introdotto delle modifiche sostanziali rispetto alla direttiva europea:

- Riconosce ai cittadini dei diritti rispetto a come devono essere trattati i loro dati personali;
- Introduce il principio di trasparenza, che obbliga il data controller a informare l'utente su come i dati vengono trattati;
- Introduce la responsabilizzazione per chi raccogli i dati personali, imponendo loro a mantenere traccia di come rispettano i principi imposti al regolamento;
- Cambia il concetto di dato personale;
- Impone delle multe qualora i principi per la protezione dei dati non vengono rispettati.

Il GDPR identifica tre tipi di entità dal punto di vista legale:

- Data target: soggetto di cui vengono prelevati i dati personali;
- Data controller: soggetto (o insieme di soggetti) che decide lo scopo e i mezzi con cui i dati sono trattati;
- Data processor: soggetto responsabile del trattamento dei dati per conto del data controller;

Esempio 1

Supponiamo di avere un'agenzia di viaggi che deve mandare le informazioni personali dei suoi clienti alla compagnia aerea e all'hotel per effettuare la prenotazione. Confermata la prenotazione, l'agenzia le inoltra ai clienti. Vediamo ora i ruoli:

- Data controller/titolare del trattamento: agenzia di viaggio, agenzia aerea, hotel.

Tutte e tre le aziende vengono identificate come data controller dal Working Party 29 in quanto tutte e tre hanno un contratto in essere con i clienti, e quindi tutte devono definire finalità e come vengono processati i dati. Se una delle tre soffre di data breach, tutte e tre potrebbero essere ritenute responsabili.

Esempio 2

Le reti sociali forniscono un modo per condividere informazioni. Un utente può sia condividere proprie informazioni che informazioni di altre persone. Vediamo ora i ruoli:

- Data controller/titolare del trattamento: rete sociale e utenti (es: cyberbullismo).

**Articolo 82 GDPR** Dal punto di vista legale al differenza tra data processor e data controller è poco rilevante. Sia il data controller che il data processor sono soggetti agli stessi obblighi dal punto di vista della legge (diversamente dalla direttiva 95).

### 13.1 Dato personale

Per il GDPR qualsiasi informazione che rende un individuo identificato (associamo un'identità a quella persona) o identificabile (anche se non siamo in grado di associare un'identità, ma siamo in grado di distinguere all'interno di un certo gruppo di persone) la rende un dato personale. Con il GDPR diventano dati personale anche tutte quelle informazioni che tracciano un individuo, i dati genetici e i dati biometrici. Altri dati personali possono essere informazioni sensitive come opinioni politiche, orientamento sessuale, religione che si segue.

Anche l'indirizzo IP, in base al contesto, può essere considerato un dato personale. Prima del GDPR non era considerato come dato personale. Alcuni siti governativi tedeschi tracciavano una serie di informazioni sugli utenti che le visitavano, come indirizzo IP, siti visitati, quantità di dati trasferiti dalle pagine, parametri delle ricerche effettuati. Questo insieme di informazioni, anche se non contiene l'identità della persona, permette di creare un profilo molto accurato delle attività dell'utente, i suoi interessi e alcune informazioni personali.

La Corte Europea ha deliberato che se il service provider che tracciava queste informazioni dispone di altre informazioni, oltre all'IP, che permettevano di identificare la persona, l'IP address deve essere considerato dato personale.

Alcuni esempi di dati personali e non sono riportati nella tabella 13.1.

Il fatto che un dato sia considerato personale dipende dal contesto: lo stesso dato, in mano a due fornitori diversi, può essere considerato per uno dato personale e per l'altro no. In generale se l'informazione è associata ad un individuo e permette di inferire qualcosa su di lui, allora è dato personale.

Il regolamento europeo per la protezione dei dati si applica solo nel momento in cui i dati vengono raccolti/processati. Se i dati raccolti sono anonimizzati, in teoria il GDPR non si applica, in quanto l'identità dell'individuo non dovrebbe essere presente.

John Smith.	Non è sempre dato personale perché è un nome comune.
L'uomo alto e anziano con un bassotto che vive al numero 15 e guida una Porsche Cayenne.	Queste info potrebbero permettere di distinguere da altre persone che vivono nella stessa area.
Indirizzo di una persona.	La ricerca su un registro pubblico potrebbe permettere di identificare chi ci vive.
Una società di servizi pubblici non registra il nome dell'occupante della casa a cui fornisce acqua, ma semplicemente annota l'indirizzo della proprietà e indirizza tutte le fatture all'occupante.	Anche se c'è solo l'indirizzo, è possibile distinguere il consumo di energia elettrica di chi vive in quell'indirizzo rispetto agli altri indirizzi.
Le informazioni sul valore di mercato di una particolare casa vengono utilizzate a fini statistici per identificare le tendenze dei valori delle case in un'area geografica. La casa non è stata selezionata perché il raccoglitore di dati desidera sapere qualcosa sugli occupanti, ma perché è una casa indipendente con quattro camere da letto in una città di medie dimensioni.	Non è un dato personale.

Tabella 13.1: Esempi di dati personali e non.

## 13.2 Obblighi per data controller/data processor

Il data controller e il data processor sono tenuti a rispettare una serie di obblighi:

- Lawfulness;
- Consenso;
- Limitazione della finalità;
- Minimizzazione dei dati;
- Accuracy;
- Storage Limitation;
- Data Security;
- Accountability.

### 13.2.1 Lawfullnes

I data controller dei dati devono avere una base legale. Le basi legali possono essere sei:

- Consenso: l'interessato ha prestato il consenso al trattamento dei propri dati personali per una o più specifiche finalità. Il data subject può revocare il consenso;
- Contratto: il trattamento è necessario all'esecuzione di un contratto di cui l'interessato è parte o all'esecuzione di misure precontrattuali adottate su richiesta dello stesso;
- Obbligo legale: il trattamento è necessario per adempiere un obbligo legale al quale è soggetto il titolare del trattamento;

- Interesse vitale: il trattamento è necessario per proteggere gli interessi vitali dell'interessato o di un'altra persona fisica;
- Interesse pubblico: il trattamento è necessario per l'esecuzione di un compito di interesse pubblico o connesso all'esercizio di pubblici poteri di cui è investito il responsabile del trattamento;
- Interesse legittimo: il trattamento è necessario ai fini degli interessi legittimi perseguiti dal responsabile del trattamento o da terzi, a meno che tali interessi non siano superati dagli interessi o dai diritti dell'interessato (es: un'azienda con più filiali può inviare i dati tra le filiali).

### 13.2.2 Consenso

Il consenso dell'utente deve essere libero, informato, specifico e inequivocabile. Nello specifico:

- Libero: non dovrebbe essere generalmente una condizione preliminare per l'iscrizione a un servizio;
- Specifico: bisogna chiedere il consenso per ciascuna finalità e attività di trattamento;
- Informato: prima di chiedere il consenso si deve fornire una privacy policy (nome del data controller, nome di controller terzi che necessitano del consenso, scopo dell'elaborazione, attività per l'elaborazione, informare l'utente che può recedere il consenso in qualsiasi momento);
- Indicazione inequivocabile: silenzio, caselle preselezionate o inattività non devono costituire consenso, è necessaria un'azione dell'utente.

### 13.2.3 Limitazione della finalità

I dati personali devono essere raccolti per finalità determinate, esplicite e legittime e non ulteriormente trattati in modo incompatibile con tali finalità. I data controller devono:

- Specificare gli scopi nell'informativa sulla privacy per le persone fisiche;
- Specificare lo scopo o gli scopi del trattamento dei dati personali all'interno dei registri del trattamento;
- Non trattare i dati per finalità incompatibili con le finalità iniziali.

Gli scopi compatibili sono scopi di archiviazione nell'interesse pubblico, scopi di ricerca scientifica o storica o scopi statistici.

### 13.2.4 Minimizzazione dei dati

I data controller devono garantire che i dati personali che stanno trattando siano:

- Adeguati: sufficienti per adempiere correttamente allo scopo dichiarato;
- Pertinenti: abbiano un legame razionale con tale scopo;
- Limitati a ciò che è necessario: non vengono mantenuti per più del necessario a quello scopo.

### 13.2.5 Accuracy

I dati personali devono essere esatti e aggiornati. Devono essere adottate tutte le misure ragionevoli per garantire che i dati personali inesatti, tenuto conto delle finalità per le quali sono trattati, siano cancellati o rettificati senza indugio.

### 13.2.6 Storage Limitation

I dati personali:

- Devono essere conservati in una forma che consenta l'identificazione degli interessati per un arco di tempo non superiore a quello necessario agli scopi per i quali i dati personali sono trattati (es: invio curriculum all'azienda, se non mi assume non deve mantenere salvato il CV);
- Possono essere conservati per periodi più lunghi nella misura in cui i dati personali saranno trattati esclusivamente a fini di archiviazione nel pubblico interesse, di ricerca scientifica o storica o a fini statistici.

### 13.2.7 Data Security

I dati personali dovrebbero essere trattati in modo da garantire un'adeguata sicurezza dei dati personali, compresa la protezione contro il trattamento non autorizzato o illecito e contro la perdita, la distruzione o il danno accidentali, utilizzando misure tecniche o organizzative adeguate.

### 13.2.8 Accountability

I data controller dei dati devono essere in grado di dimostrare la loro conformità agli obblighi GDPR. Devono mettere in atto misure tecniche e organizzative adeguate per soddisfare le esigenze di Requisiti di responsabilità. Queste misure includono:

- Adottare e attuare politiche di protezione dei dati;
- Adottare un approccio basato sulla "protezione dei dati fin dalla progettazione e per impostazione predefinita";
- Stipulare contratti scritti con organizzazioni che elaborano dati personali per conto del data controller;
- Mantenere la documentazione delle attività di trattamento;
- Attuare adeguate misure di sicurezza;
- Registrare e, se necessario, segnalare violazioni dei dati personali;
- Effettuare valutazioni d'impatto sulla protezione dei dati per gli usi dei dati personali che possono comportare un rischio elevato per gli interessi delle persone;
- Nominare un data protection officer.

### 13.2.9 Esempio

L’azienda X deve sviluppare un sito web per gestire le candidature. Questo deve permettere di cercare richieste di lavoro, creare account, aggiornare le informazioni personali, caricare CV e candidarsi per un’offerta di lavoro. Analizziamo il caso:

- Data subject: candidati e recruiter;
- Dati raccolti: username e password, CV, informazioni di contatto per le candidature;
- Lawfulness: la base legale è il contratto;
- Limitazione dello scopo: è ammessa solo la gestione delle candidature e la comunicazione agli utenti di offerte di lavoro;
- Data minimization: il form online dovrebbe richiedere solo i dati necessari per gestire la candidatura;
- Accuracy: si fornisce all’utente un’interfaccia per modificare i suoi dati;
- Storage Limitation: i dati dei candidati respinti devono essere cancellati, a meno che non vi sia un obbligo di legge, i dati dei candidati accettati dovrebbero essere trasferiti all’HR;
- Data Security: autenticazione a più fattori, role-based access control, cifratura dei dati inattivi.

## 13.3 Diritti dell’utente

Il data subject dispone di una serie di diritti, quali:

- Diritto a ricevere un’informativa sull’utilizzo dei dati dettagliata (trasparenza);
- Diritto di accesso ai dati raccolti;
- Diritto alla modifica dei dati, affinché sia aggiornati;
- Diritto a richiedere l’eliminazione dei dati raccolti;
- Diritto a richiedere che i dati non siano più processati, quando la base giuridica è diversa dal consenso;
- Diritto a non essere soggetti a decisioni automatiche (uso di algoritmi di ML usati per profilare);
- Diritto a trasferire i dati da un service provider ad un altro.

### 13.3.1 Trasparenza

I data controller devono informare le persone sul trattamento dei loro dati in modo facilmente accessibile e comprensibile. Quando i dati vengono raccolti, i data controller devono fornire un’informativa sulla privacy contenente:

- Il nome e i dettagli di contatto dell’organizzazione;
- Lo scopo del trattamento;

- La base giuridica del trattamento;
- Le categorie di dati personali ottenuti;
- I destinatari o le categorie di destinatari dei dati personali;
- I dettagli dei trasferimenti dei dati personali verso eventuali paesi terzi o organizzazioni internazionali;
- I periodi di conservazione dei dati personali;
- I diritti delle persone fisiche in relazione al trattamento.

Una delle tecniche usate per specificare le politiche è quella di usare più livelli (link che porta alla pagina che dettaglia maggiormente la politica).

## 13.4 Riportare violazioni

Il GDPR impone che venga notificato al garante della privacy il data breach entro 72 ore dal momento in cui lo si scopre. È richiesto inoltre che venga quantificato se vi è un rischio per gli individui vittime del data breach. Se il rischio è elevato è richiesto che vengano notificate anche le persone i cui dati personali sono stati resi pubblici. Questo implica che è necessario avere un sistema sviluppato di incident-response e incident-management, altrimenti l'organizzazione non è in grado di rispondere entro i tempi imposti dal GDPR.

## 13.5 Sanzioni previste dal GDPR

Il GDPR definisce due tipi di multe:

- Multe per violazione dei principi sulla protezione dei dati personali che sono considerati di severità minore (es: non è stato nominato il data protection officer, non è stato notificato il garante entro le 72 ore del data breach);
- Multe per violazione dei principi sulla protezione dei dati personali che sono considerati di severità maggiore (es: al soggetto non è consentito di visionare o cancellare i dati).



# Capitolo 14

## Tecniche di anonimizzazione

Dato un dataset con informazioni personali sensibili vogliamo calcolare e rilasciare funzioni del dataset proteggendo la privacy individuale.

Gli attributi di un record di un dataset possono essere classificati in tre categorie:

- **Identifieri esplicativi:** permettono l'identificazione univoca dell'utente (es: nome, cognome, codice fiscale, ...)
- **Quasi-identifieri:** pezzi di informazioni che non sono di per sé identifieri univoci, ma sono sufficientemente ben correlati con un'entità da poter essere combinati con altri quasi-identifieri per creare un identificatore univoco (es: data di nascita, età, numero di telefono, ...);
- **Attributi sensitivi:** attributi che non dovrebbero essere collegabili all'utente e dipendono dal contesto (es: salario, malattie, ... ).

Un esempio di attributi è riportato in figura 14.1.

### 14.1 Tecniche di per proteggere gli identifieri esplicativi

Esistono due tecniche per proteggere gli identifieri esplicativi:

- **Tokenization:** genera un token univoco per il dato;

Key Attributes		Quasi-identifiers			Sensitive attributes
ID	Name	DOB	Gender	Zipcode	Disease
12345	Andre	1/21/76	Male	53715	Heart Disease
56789	Beth	4/13/86	Female	53715	Hepatitis
52131	Carol	2/28/76	Male	53703	Brochitis
85438	Dan	1/21/76	Male	53703	Broken Arm
91281	Ellen	4/13/86	Female	53706	Flu
11253	Eric	2/28/76	Female	53706	Hang Nail

Figura 14.1: Esempio di attributi

Original Database				Released Database		
Name	Zipcode	Age	Disease	Zipcode	Age	Disease
Hilary	47677	29	Heart Disease	476***	2*	Heart Disease
Jenny	47602	22	Heart Disease	476***	2*	Heart Disease
Bob	47678	27	Heart Disease	476***	2*	Heart Disease
Izzy	47905	43	Flu	4790*	$\geq 40$	Flu
John	47909	52	Heart Disease	4790*	$\geq 40$	Heart Disease
Fred	47906	47	Cancer	4790*	$\geq 40$	Cancer
Sam	47605	30	Heart Disease	476**	3*	Heart Disease
Carl	47673	36	Cancer	476**	3*	Cancer
Sarah	47607	32	Cancer	476**	3*	Cancer

Figura 14.2: Nel dataset rilasciato è stato rimosso il nome (identificatore); le prime 3 righe sono una classe di equivalenza.

- **Sostituzione:** sostituisce il valore di un attributo con un valore alternativo (scelto casualmente).

Queste tipo di tecniche non sono sufficienti a garantire la privacy degli utenti.

### Esempio

Nel 2006 i ricercatori di AOL avevano preso tutte le query fatte dagli utenti in tre mesi e le avevano pubblicate in un dataset dopo aver applicato la tecnica di tokenization. Due giornalisti, analizzando il dataset, sono riusciti a identificare risalire all'identità di uno degli utenti le cui query erano state inserite nel dataset.

Un altro attacco a cui queste tecniche sono vulnerabili è il **record linkage**: dato un dataset anonimizzato (dataset medico), a cui sono stati rimossi gli attributi identificatori, e uno pubblico (dataset cittadini votanti), incrociando gli attributi quasi-identificatori è possibile risalire all'identità della persona.

## 14.2 K-Anonymity

Introdotto per proteggersi dal record linkage. Prevede che:

- Un record sia indistinguibile da almeno  $k-1$  altri record per quanto riguarda i quasi-identificatori;
- Ogni classe di equivalenza contenga almeno  $k$  record che hanno gli stessi valori per i quasi identificatori.

Un esempio è riportato in figura 14.2. uno dei problemi di questa soluzione è che se l'attaccante sa che Bob si trova nella prima classe di equivalenza (prime tre righe), allora ha  $1/3$  di probabilità di indovinare.

Per raggiungere la K-Anonymity si usa la **generalizzazione**: si sostituiscono i quasi-identificatori con valori meno specifici finché non si ottengono  $K$  valori identici (partiziona i domini con valori ordinati in intervalli). Quando un attributo è troppo specifico o la generalizzazione non lo generalizza abbastanza, lo rimuovo (**soppressione** - comune con valori anomali).

Esistono molti algoritmi nella letteratura che mirano a realizzare un'anomimizzazione "utile", ma spesso di solito senza alcuna chiara nozione di utilità.

Name	Age	Gender	State of domicile	Religion	Disease
*	20 < Age ≤ 30	Female	Tamil Nadu	*	Cancer
*	20 < Age ≤ 30	Female	Kerala	*	Viral infection
*	20 < Age ≤ 30	Female	Tamil Nadu	*	TB
*	20 < Age ≤ 30	Male	Karnataka	*	No illness
*	20 < Age ≤ 30	Female	Kerala	*	Heart-related
*	20 < Age ≤ 30	Male	Karnataka	*	TB
*	Age ≤ 20	Male	Kerala	*	Cancer
*	20 < Age ≤ 30	Male	Karnataka	*	Heart-related
*	Age ≤ 20	Male	Kerala	*	Heart-related
*	Age ≤ 20	Male	Kerala	*	Viral infection

Figura 14.3: L'attributo religione non serve nell'analisi e quindi lo rimuoviamo.

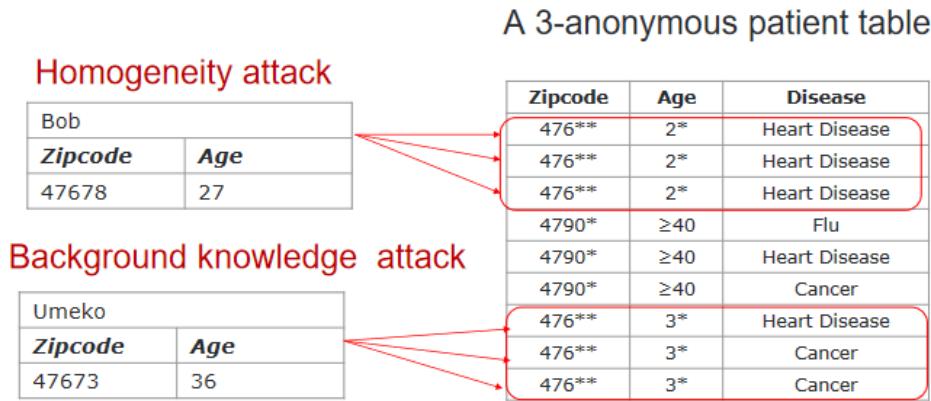


Figura 14.4: Se l'attaccante è vicino di casa di Bob e conosce il suo zip code ed età, può scoprire che soffre di heart disease, in quanto nella classe di equivalenza di Bob, tutti soffrono di quella malattia. Sapendo che Umineko è giapponese e i giapponesi difficilmente soffrono di problemi al cuore, è molto probabile che abbia in cancro.

Esempi di K-anonymity:

- Applichiamo una formula che ritorna tre possibili valori di età:  $> 20$ ,  $20 < x \leq 30$ ,  $30 < x \leq 40$ ,  $\geq 40$  (??);
- Raccogliamo sotto Grad School i valori Bachelor e Master.

La K-Anonymity non fornisce privacy nel caso in cui valori sensibili nella stessa classe di equivalenza mancano di diversità oppure l'attaccante ha conoscenze di background (14.4)

### 14.3 L-diversiry

Introdotto per risolvere i problemi della K-Anonymity. Prevede che attributi sensibili nella stessa classe di equivalenza di quasi-identificatori abbiano valori diversi (14.5). Nello specifico:

- Ogni classe di equivalenza deve avere almeno I valori sensibili ben rappresentati;
- Non previene attacchi di inferenza probabilistica.

Caucas	787XX	Flu
Caucas	787XX	Shingles
Caucas	787XX	Acne
Caucas	787XX	Flu
Caucas	787XX	Acne
Caucas	787XX	Flu
Asian/AfrAm	78XXX	Flu
Asian/AfrAm	78XXX	Flu
Asian/AfrAm	78XXX	Acne
Asian/AfrAm	78XXX	Shingles
Asian/AfrAm	78XXX	Acne
Asian/AfrAm	78XXX	Flu

Figura 14.5: Esempio.

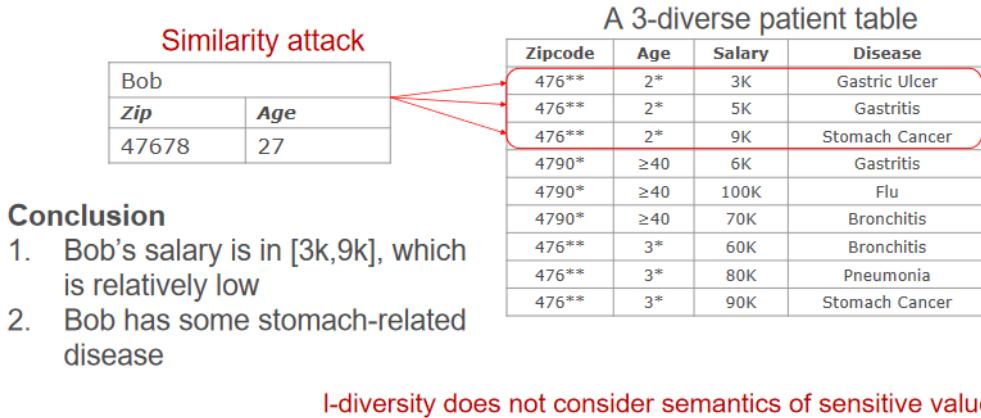


Figura 14.6: Esempio di sensitive attribute disclosure.

**Definizione 14.1** (Entropia l-diversità). Ogni classe di equivalenza non solo deve avere valori sensibili abbastanza diversi, ma anche i diversi valori sensibili devono essere distribuiti in maniera sufficientemente omogenea. L'entropia della distribuzione dei valori sensibili in ogni classe di equivalenza deve essere almeno  $\log(l)$ .

L'entropia di una classe di equivalenza  $E$  è definita come:

$$\text{Entropy}(E) = - \sum_{s \in S} p(E, s) \log p(E, s)$$

dove  $S$  è il dominio dell'attributo sensibile e  $p(E, s)$  è la frazione di record in  $E$  che ha valore sensibile  $s$ .

Questa tecnica è debole al **sensitive attribute disclosure** (14.6).

La L-diversity soffre di altri vulnerabilità:

- Se l'attaccante sa che 50% del dataset è HIV+ e il restante 50% è HIV-, può dire che Bob è al 50% HIV+ (viola privacy);
- Se 49 record sono HIV+ e 1 è HIV-, allora molto probabilmente Bob è HIV+ (problema distribuzione valori).

In generale, L-diversity non considera la semantica e la distribuzione dei valori sentitivi.

Caucas	787XX	Flu
Caucas	787XX	Shingles
Caucas	787XX	Acne
Caucas	787XX	Flu
Caucas	787XX	Acne
Caucas	787XX	Flu
Asian/AfrAm	78XXX	Flu
Asian/AfrAm	78XXX	Flu
Asian/AfrAm	78XXX	Acne
Asian/AfrAm	78XXX	Shingles
Asian/AfrAm	78XXX	Acne
Asian/AfrAm	78XXX	Flu

Figura 14.7: Esempio di T-closeness.

The diagram illustrates a reconstruction attack. It shows two versions of a dataset and how statistical queries can be answered from both to deduce information about a new record.

**Dataset 1 (Original):**

Name/Id	age	weight	sex	disease	...
Mario Rossi	65	82	M	yes	...
Daniele Bianchi	35	120	M	yes	...
Lucia Verdi	40	45	F	no	...
...	...	...	...	...	...

**Dataset 2 (After Insertion):**

Name/Id	age	weight	sex	disease	...
Mario Rossi	65	82	M	yes	...
Daniele Bianchi	35	120	M	yes	...
Lucia Verdi	40	45	F	no	...
Sergio Neri	20	140	M	yes	...

**Queries and Deductions:**

- How many men have the disease ? 2**
- What is the average age / weight of men who have the disease ? 50 / 101**
- How many men have the disease ? 3**
- What is the average age / weight of men who have the disease ? 40 / 114**
- We can deduce the exact age / weight of the new record**

Figura 14.8: Esempio di Reconstruction Attack.

## 14.4 T-closeness

Prevede che la distribuzione degli attributi sensibili all'interno di ciascun gruppo di quasi-identificatori sia "vicina" alla loro distribuzione nell'intero database originale () .

In generale, semplicemente anonimizzare i dati non è sicuro: i dati presumibilmente anonimizzati spesso contengono modi alternativi di identificazione (quasi identificatori). L'accesso alle informazioni ausiliarie appropriate può quindi comportare la reidentificazione.

## 14.5 Differential privacy

Supponiamo di avere un dataset nel quale consentiamo solo analisi statistiche, quindi solo query che ritornano valori aggregati. Questo tipo di analisi è sicura? In teoria sì, ma nella pratica no.

Supponiamo di sapere che nel dataset originale 2 uomini abbiano il diabete e che età e peso medi degli uomini con diabete siano 50 anni e 101 kg. Rieseguendo le due query dopo l'aggiunta del nuovo record, scopriamo che i malati sono saliti a 3 e che età e peso medi sono cambiati.

Possiamo quindi ricostruire età e peso esatti per il nuovo record, oltre a sapere che soffre di diabete.

Quindi la restrizione alle query aggregate non è sufficiente: anche queste query potrebbero trapelare informazioni sugli individui (**Reconstruction Attack**).

Da questo è stato introdotto il concetto di **differential privacy**: qualsiasi rischio relativo alle informazioni per una persona non dovrebbe cambiare in modo significativo a seguito dell'inclusione o meno delle informazioni di quella persona nell'analisi. In generale vogliamo che la differenza tra i risultati delle due analisi sia un valore trascurabile  $\epsilon$ .

La differential privacy garantisce due proprietà:

1. **Post processing invariance**: il rischio non aumenta se i dati non vengono toccati ancora;
2. **Robustness under Composition**: se eseguiamo una serie di analisi sul dataset che soddisfano la privacy differenziale, ciascuna col proprio  $\epsilon$ , l'applicazione di queste analisi in sequenza genera un  $\epsilon$  totale pari alla somma delle singole  $\epsilon$ .

## 14.6 Implementazione privacy differenziale

Uno dei metodi più usati è il Global Sensitivity Method, che usa la distribuzione di Laplace. Noi vogliamo effettuare un'analisi sul dataset. Il metodo dice di prendere il valore della funzione e di aggiungerci del rumore, preso da una variabile che ha una distribuzione secondo quella di Laplace, e ritorniamo il risultato.

Supponiamo che la nostra funzione calcoli la media dei valori sensitivi. Per prima cosa calcoliamo la global sensitivity della funzione, ottenuta dalla differenza tra la funzione applicata al dataset D che stiamo considerando e la funzione applicata al dataset D' che differisce dal primo per al più un record (calcoliamo la differenza record per record, e ritorniamo quella con il valore assoluto maggiore). Ottenuta la global sensitivity aggiungiamo il rumore Z, data da:

$$Z = \frac{\text{global\_sensitivity}}{\epsilon} \cdot \text{distribuzione\_Laplace}(0, 1)$$

In generale, la privacy differenziale può essere usata in:

- Analisi statistiche (conteggi, media, mediana, ...);
- Machine learning supervisionato e non supervisionato (classificazione, regressione, ...);
- Generazione di dati sintetici.

**US census bureau 2020** In America c'è un censimento periodico, e si possono fare query su questi dati. L'analisi soddisfa la privacy.

**Google** Ha un'implementazione open source che si chiama RAPPOR, ed è usata per fare analisi sulle ricerche degli utenti.

**Apple** Non ha reso pubblico l'algoritmo. Alcuni ricercatori vi hanno avuto l'accesso e hanno fatto reverse engineering del codice, e non è così private perché usa un epsilon intorno a 11.

**Privacy differenziale locale** Sia google che apple usano la privacy differenziale locale per avere info su dati telemetrici; anziché catturare il dato reale, aggiungono al dato singolo un rumore, e questo dato perturbato può comunque essere analizzato. Apple lo usa per mantenere traccia di quali sono le emoji più usate.

**Implementazioni della privacy differenziale** Ci sono due implementazioni della privacy differenziale: Tensor flow (google) e Opacus (facebook).