Sensitive Data

Sensitive Data

> Sensitive data such as **passwords, credit card numbers, health records, personal information and business secrets** require extra protection, particularly if that data falls under privacy laws (EU's General Data Protection Regulation GDPR), financial data protection rules such as PCI Data Security Standard (PCI DSS) or other regulations. [^1]

GDPR

> Regulation (EU) 2016/679 of the European Parliament and of the Council of 27 April 2016 on the protection of natural persons with regard to the processing of personal data and on the free movement of such data, and repealing Directive 95/46/EC (General Data Protection Regulation)

Personal Data as defined in GDPR

- Name and surname

- Home address

- Email address

- Identification card number

- Location data (for example on a mobile phone)

- Internet Protocol (IP) address

- ...

§ Articles 2, 4(1) and(5) and Recitals (14), (15), (26), (27), (29) and (30)

Sensitive Personal Data as defined in GDPR

- Personal data revealing racial or ethnic origin, political opinions, religious or philosophical beliefs

- Trade-union membership

- Genetic data, biometric data processed solely to identify a human being

- Health-related data

- Data concerning a person's sex life or sexual orientation

§ Article 4(13), (14) and (15) and Article 9 and Recitals (51) to (56)

GDPR Enforcement Tracker

PCI DSS

PCI DSS is the global data security standard adopted by the payment card brands for all entities that process, store or transmit cardholder data and/or sensitive authentication data.

PCI DSS Requirements

Goals	Requirements
Secure Network and Systems	Firewall; No default credentials
Protect Cardholder Data	Protect stored data; encrypt transmissions
Vulnerability Management	Anti-Malware/-Virus; Secure Development
Strong Access Controls	Need-to-know access; Authentication; Restrict physical access
Monitoring & Testing	Monitor network and data access; Test systems/processes
Security Policy	Maintain Information Security policy for all personnel

Sensitive Data Exposure

- **Failure to determine the protection needs of data**

- Transmitting data in clear text (e.g. HTTP, SMTP, FTP)

- Employing old or weak cryptographic algorithms

- Using default or weak generated crypto keys

- Lack of proper key management/rotation

- Not enforcing encryption through browser directives/HTTP headers

- Lack of certificate verification

⚠️ *External Internet traffic is especially dangerous!*

Risk Rating

Sensitive Data Exposure

Exploitability	Prevalence	Detecability	Impact	Risk
🔶 Average	🔴 Widespread	🔶 Average	🔴 Severe	A3
(2	+ 3	+ 2) / 3	* 3	= 7.0

Prevention

- **Classify data in system** and determine sensitivity level

- **Don't store sensitive data unnecessarily**

- Encrypt data at rest

- Ensure up-to-date and strong
  - Standard algorithms
  - Protocols
  - Keys

- Encrypt data in transit (e.g. TLS) and enforce encryption (e.g. HSTS)

Information Classification

Class	Description	Examples
Public	Information without any confidentiality requirements.	User documentation, news, press releases, lunch menus
Internal	Common information inside an organization.	Memos, system documentation or meeting minutes
Confidential	Information or compartmental data with restricted access. Disclosure might induce damage.	Customer, HR, financial or PII data; source code, credentials, logfiles
Secret	Highest confidentiality and integrity requirements. Damaging to organization if disclosed.	Business secrets, secret formulae, planned mergers/acquisitions

Practices ✓/✗ for Information Classes

Practice	Public	Internal	Confidential	Secret
Publish on Internet	✓	✗	✗	✗
Publish on Intranet	✓	✓	✗	✗
Print on 🖨️	✓	✓	✓ if picked up immediately	✓ on personal or otherwise secured printer

Practice	Public	Internal	Confidential	Secret
Share with third parties	✔	✔ with NDA	✔ with NDA + permission	✔ with NDA + permission
Copy to USB key	✔	✔	✔ with encryption + permission	✔ with encryption + permission

⚠️ *Many organizations do not allow the use of USB keys **in general**. This kind of restriction would obviously **overrule** any of the above "Copy to USB" assessments with* ❌.

Data Lifecycle restrictions (●) and recommendations (○)

Phase	Internal	Confidential	Secret
Permanent storage	● Access Control (against external access)	● Access Control ○ Access logs, Encryption	● Access Control, Access logs, Encryption
Transfer (internal network)	No restrictions	○ Encryption (e.g. TLS)	● Encryption (e.g. TLS) ○/● End-to-end encryption (e.g. PGP, Signal)

Phase	Internal	Confidential	Secret
Transfer (public network)	⭕ Encryption (e.g. VPN)	⭕ Encryption (e.g. VPN, TLS)	🔴 Encryption (e.g. VPN, TLS) ⭕/🔴 End-to-end encryption (e.g. PGP, Signal)
Disposal	No restrictions	🔴 Shredding, secure deletion, data wipe	🔴 Shredding, secure deletion, data wipe ⭕/🔴 Destroy medium physically (🔨, 🔥)

ℹ️ *For "Public" data no restrictions for any lifecycle phases apply.*

HTTP Strict Transport Security (HSTS)

HTTP Strict Transport Security (HSTS) is an opt-in security enhancement that is specified by a web application through the use of a special response header. Once a supported browser receives this header that browser will prevent any communications from being sent over HTTP to the specified domain and will instead send all communications over HTTPS. It also prevents HTTPS click through prompts on browsers.

Example

```
Strict-Transport-Security: max-age=16070400; includeSubDomains
```

Secure Cryptographic Storage Design

- Only store sensitive data that you need

- Use strong approved Authenticated Encryption

- Store a one-way and salted value of passwords

- Ensure that the cryptographic protection remains secure even if access controls fail

- Ensure that any secret key is protected from unauthorized access

- Follow applicable regulations on use of cryptography

Perfect Forward Secrecy (PFS)

> Perfect forward secrecy means that a piece of an encryption system automatically and frequently changes the keys it uses to encrypt and decrypt information, such that if the latest key is compromised, it exposes only a small portion of the user's sensitive data. Encryption tools with perfect forward secrecy switch their keys as frequently as every message in text-based conversation, every phone call in the case of encrypted calling apps, or every time a user loads or reloads an encrypted web page in his or her browser.

ℹ️ Examples of crypto protocols (used for instant messaging conversations) providing PFS are OTR (Off-the-record) Messaging and Double Ratchet (used within Signal).

Best Practices

Scenario	Practice	🔑 Length
Key exchange	Diffie-Hellman	2048+ bits
Message Integrity	HMAC-SHA2	-
Message Hash	SHA2	256 bits
Asymetric encryption	**ECC** (Curve25519), RSA	2048 bits (RSA)
Symmetric-key algorithm	AES	128-**256** bits
Password Hashing	**Bcrypt**, Argon2, PBKDF2	-

Exercise 6.3 (🏠)

1. Access a confidential document (⭐)

2. Retrieve as many clear text user passwords as you can (⭐⭐⭐⭐)

3. Visit the Token Sale page before it officially goes live (⭐⭐⭐⭐⭐)

Bonus exercises on cryptography (*optional*)

4. Retrieve both the 🐰 easter eggs (⭐⭐⭐⭐)

5. Solve the steganography challenge (⭐⭐⭐⭐)

6. Solve the non-existent challenge #999 (⭐⭐⭐⭐⭐⭐)