Open Systems Security

1. Security and access control

1.1. Security and dependability features

- **Confidentiality** features, ensuring that only authorized users can read data managed by the software system
- **Integrity** features, ensuring that only authorized users can can modify the software system's resources
- **Availability** features, ensuring that only authorized users can use the software system.

1.2. Software security is about…

Software security is about:

1. protecting "plain" software systems by adding to them a number of specific security features;
2. programming techniques for developing secure software systems, designing and coding them to withstand (rather than prevent) attacks

Software security includes both adding security features (such as an **access control facility**) and achieving security-by-coding (e.g., via robust **coding techniques** that make attacks more difficult).

Example 1. Access control vs secure coding

> Let us assume that a web server has a buffer overflow fault 2, and that we want to prevent a remote attacker from overflowing the buffer by sending to the server an oversize HTTP GET request. A way to prevent this buffer overflow attack could be adding a software feature to the web server, a monitor function that observes HTTP requests as they arrive over port 80, and drops them if they are bigger than a pre-set threshold. Another way to achieve the same result consists in fixing the web server source code to eliminate the buffer overflow fault altogether. Clearly, the latter approach can only be adopted when the server's source code is available and its operation is well understood.

1.3. Access control

Access Control (AC) is the ability to allow or deny the use of resources. Decides who *subject* is authorized to perform certain operations on a given *object* and who is not

The factors to be made available by a subject for gaining access to an object are described as * something you *have*, such as a credential, * something you *know*, e.g. a secret PIN, or * something you *are*, typically a fingerprint/eye scan or another biometric input

- Discretionary Access Control (DAC)

- Mandatory Access Control (MAC)
- Role-Based Access Control (RBAC)

Authorization determines which actions a subject can do on the system; the semantics of actions depends on the nature of the objects involved. Permissions are implemented differently in systems based DAC than in MAC.

Access Control Lists (ACLs) are data structures widely used to implement both discretionary and mandatory access control models.

Discretionary Access Control

> A means of restricting access to objects based on the identity of subjects and/or groups to which they belong. DAC is discretionary in the sense that a subject with a certain access permission is capable of passing on that permission (perhaps indirectly) to any other subject (unless explicitly restrained from doing so by mandatory access control)
>
> — TCSEC, Orange Book

An individual user, or program operating on the user's behalf, is allowed to specify explicitly the types of access other users (or programs executing on their behalf) may have to information under the user's control.

Mandatory Access Control

MAC model enforces access control based on rules defined by a central authority

> a means of restricting access to objects based on the sensitivity (as represented by a label) of the information contained in the objects and the formal authorization (i.e., clearance) of subjects to access information of such sensitivity.
>
> — TCSEC, Orange Book

For a subject to be able to access an object, its sensitivity level must be at least equal or higher than the objects' sensitivity level

BLP (Bell-LaPadula) model

BLP partitions the organizational domain into security levels and assigns a security label to each level.

Each subject and object within the system need to be associated with one of those predefined levels and therefore associated with a security label.

The security labels denominations may change from one organization to another, based on the

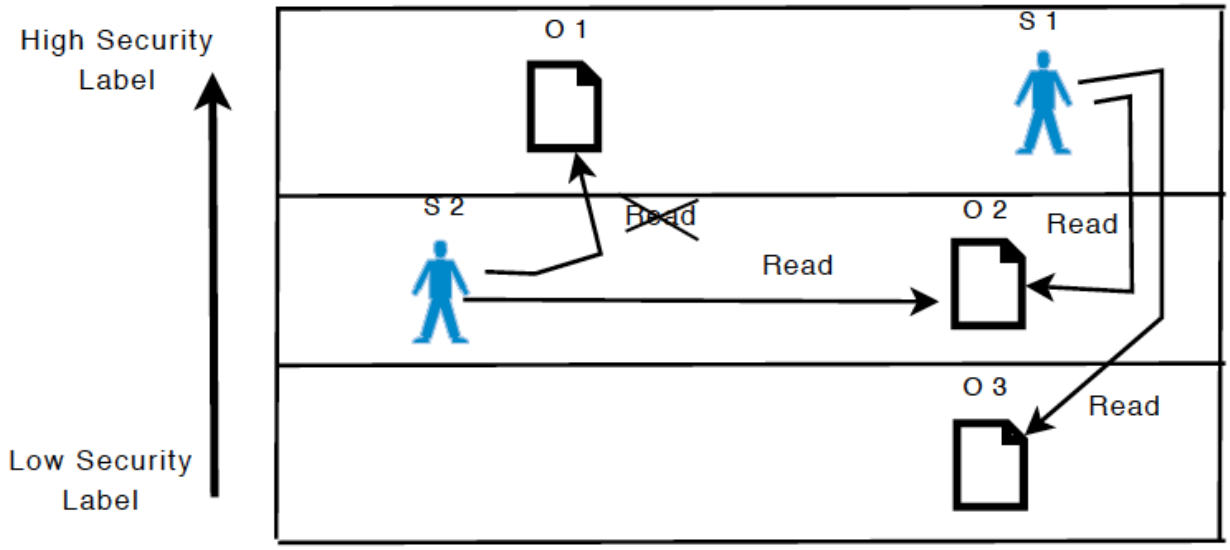
terminology used in each organization. For instance:

- Military: Top secret, Secret, Classified
- Business: Board Only, Managerial, etc.

BLP properties:

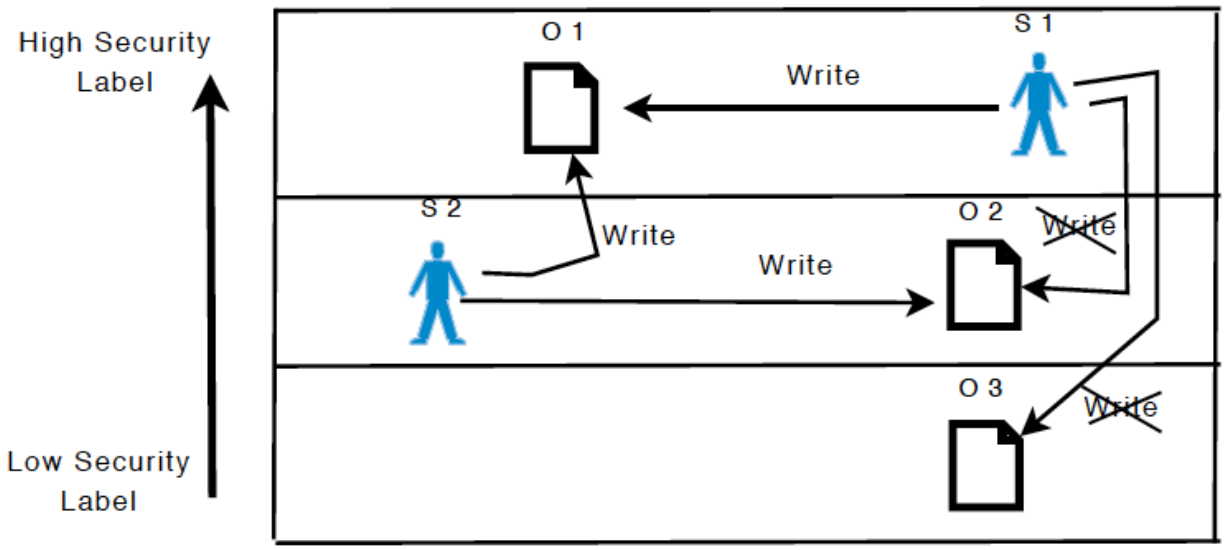
- **Simple Security Property**: A subject can only have read access to objects of lower or equal security level (no read-up).
- **star-security property**: A subject can only have write access to objects of higher or equal security level (i.e, no write-down is allowed).

Example 2. Simple security property



Subject S1 can read both objects O2 and O3, since they belong to a lower security level. Subject S2 can read object O2 since it belongs to the same security level. S2 is not authorized to read object O1, since it belongs to a higher security level

*Example 3. *-security property*

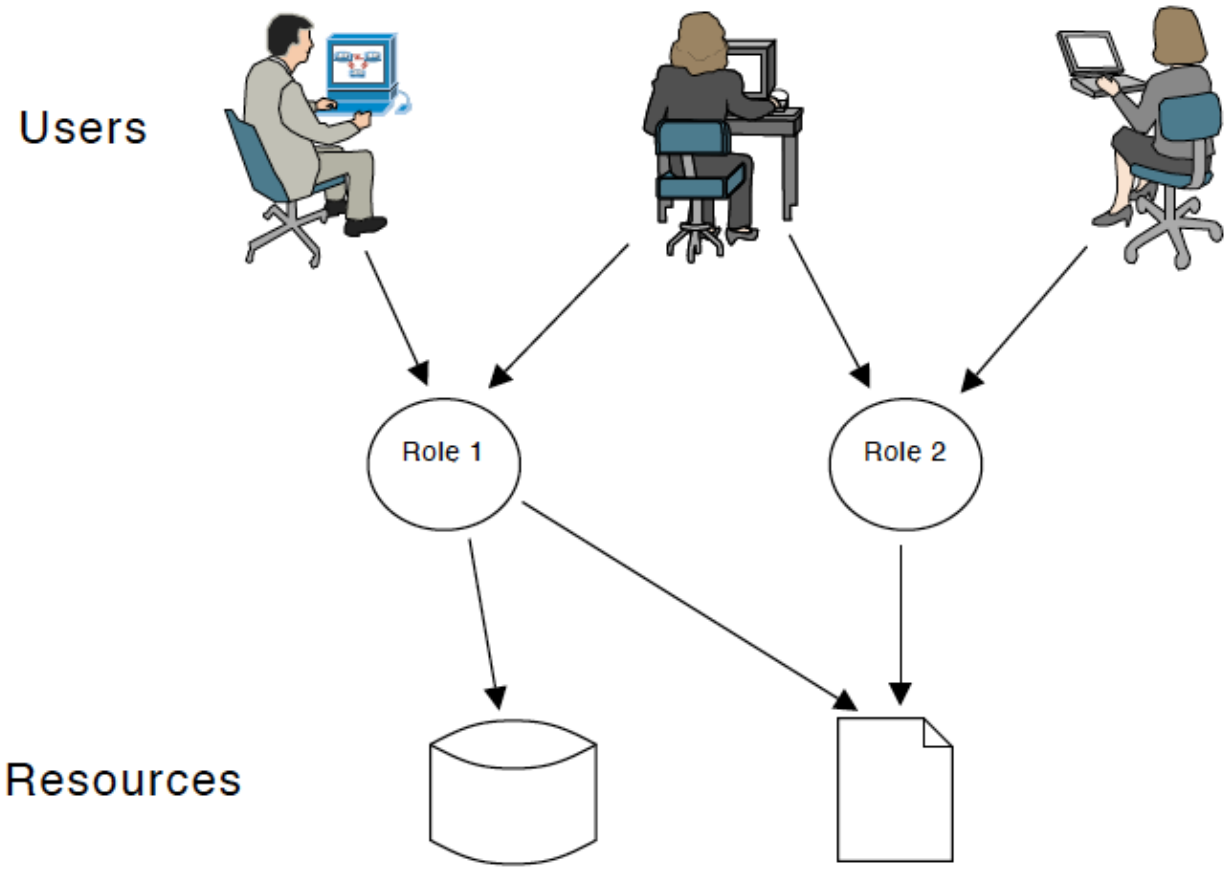


Subject S2 can write both in object O1 and O2 since they belong respectively to a higher and to the same security level. Subject S1 can write only to object O1 since it belongs to the same security level.

Role-Based Access Control

A role is “a set of actions and responsibilities associated with a particular working activity”

The relation between roles and individuals is many-to-many, like the one between roles and system resources. A specific role may include one or more individuals, and at the same time a specific role can have access rights to one or more resources.



2. Open-source software security

2.1. Security certifications

Standards Standards for security certification of software:

- TCSEC (1985) in USA - Orange Book
- ITSEC (1991) in Europe
- CTCPEC (1993) in Canada

2.2. COTS and OSS

Commercial Off-The-Shelf (COTS) products change every few months and the certification process requires a year Certified products may become obsolete by the time certificates are available.

Often, users have to decide whether to adopt a non-certified recent release of a system, or settle for a previous version which completed the certification process.

Many agree that security certification should follow the embedded software model: it should deal with software products and the environment in which they run at the same time.

Another challenge is the rapidly changing landscape of security threats, which makes it difficult to maintain the set of desirable properties against which products must be certified.

The validity of a security certificate over time is essentially associated with the software product's ability to repeal new attacks.

Some certification authorities do offer a certified product monitoring activity which consists of regularly recomputing the certificate including properties considered desirable in order to repeal or prevent emerging threats. As long as the evaluation body in charge of this monitoring activity repeats this procedure each time a new threat emerges, the certificate is considered to be monitored.

OSS projects are functionality-driven, collaborative, rapidly updated and rapidly released, and pushed by volunteers working with a diversity of interests, skills, and hardware sets.

The IBM Linux Technology Center has provided many contributions to the development community on the topic of Linux security

2.3. Bibliography

- K.S. Shankar and H. Kurth. Certifying open source: The linux experience. IEEE Security & Privacy, 2(6):28–33, November-December 2004.
- G. Wilson, K. Weidner, and L. Salem. Extending linux for multi-level security. In SELinux Symposium, 2007

2.4. OSS criteria

1. **Free redistribution**. An open source license must permit anyone who obtains and uses the covered software to give it away to others without having to pay a royalty or other fee to the original copyright owner(s).
2. **Access to source code**. All types of open source licenses require everyone who distributes the software to provide access to the program source code. Often, distributors provide the source code along with the executable form of the program, but the license does not bind them to do so; for instance, they could make the code available via Internet download, or on other media, free or for a reasonable fee to cover the media cost.
3. **Derivative works**. An open source license must allow users to modify the software and to create new works (called derivatives) based upon it. An open source license must permit the distribution of derivative works under the same terms as the original software. This provision, together with the requirement to provide source code, fosters the rapid evolution pace of open source software.
4. **Integrity** of the author's source code must be preserved.
5. **No discrimination on users**. An open source license does not discriminate against persons or groups. Everybody can use open source software, provided they comply with the terms of the open source license.
6. **No discrimination on purpose**. An open source license does not discriminate against application domains. In other words, the license may not restrict anyone from using the software based on the purpose of such usage. Specifically, it may not restrict the program from being used for commercial purposes. This permits business users to take advantage of open

source products for commercial purposes.

7. **License Distribution**. The wording of an open source license must be made available to all interested parties, and not to the purchaser alone.
8. **Product Neutrality**. An open source license must not be specific to a single software product.
9. **No transfer of restrictions**. An open source license on a software product must not restrict the use of other software products, both open source and proprietary. In other words, an open source license must not mandate that all other programs distributed together with the one the license is attached to are themselves open source. This clause allows software suppliers to distribute open source and proprietary software in the same package. Some widespread licenses, including the GPL (General Public License) presented below, require that all software components “constituting a single work” to fall to under the GPL if anyone of them is distributed under GPL. This requirement may seem to have been spelled out clearly, but wrapping and dynamic invocation techniques have sometimes been used as a work-around to it.
10. **Technology Neutrality**. An open source license must not prescribe or supply a specific technology.

Relevant OSS Licenses

- **GNU General Public License (GPL)**: the original programmer does not retain any right on modified versions of the software. For instance, Linux is distributed under a GPL license.
- **Mozilla Public License (MPL)**: Less liberal than GPL. It requires the inclusion or publishing of the source code within one year (or six months, depending on the specific situation) for all publicly distributed modifications.
- **Berkeley Software Distribution (BSD)** License: Permits users to distribute BSD-licensed software for free or commercially, without providing the source code; they also may modify the software and distribute the changes without providing the source code. A major difference between BSD and GPL is that organizations or individuals who create modified versions of software originally licensed under BSD can distribute them as proprietary software, provided that they credit the developers of the original version.
- **Apache Software License** and **MIT License**: very similar to the BSD License.

3. Data privacy

3.1. Data privacy

Privacy is important when one wants to make use of data that involve individuals' sensitive information, especially when data collection is becoming easier and sophisticated data mining techniques are becoming more efficient.

Data custodians

- Health service providers
- Government agencies

- Insurance companies
- Etc

They have data they would like to release to data analysts and researchers for the *public good*: evaluation of economic models, identification of social trends, etc.

Such data contain *personal information* (e.g. medical records, salaries, etc.) so that a straightforward release of data is not appropriate.

One approach to solving this problem is to require data users to sign *non-disclosure agreements* (NDA):

- Will need legal resources and enforcement mechanisms.
- Cannot protect against data theft even if taking precautions.

Thus, it is important to explore technological solutions which anonymize the data prior to its release.

3.2. Census data

Problem of releasing public-use data sets: should not reveal information about individuals in the population.

Mechanisms intended to protect individual privacy in public-use data: *statistical disclosure control* (SDC)

Aggregation of data

- **Contingency tables**: frequency count tabulated data (e.g. zip code, age range, smoking status)
- **Microdata**: non-aggregate data, where each row refers to a person in the population

Confidentiality mechanisms

- Cell supression and noise addition (contingency tables)
- Data swapping (contingency tables and microdata)
- Sampling, geographic coarsening and top/bottom coding (microdata): e.g. 1% sample, population > 100K, age ≥ 90
- Synthetic data (microdata): produce data with similar distributional characteristics

3.3. Attacks

Linking Attacks

External data are combined with an anonymized data set.

Example 4. Group Insurance Commission case

An insurance dataset contained medical records of Massachusetts state employees. Since the data did not contain identifiers such as names, social security numbers, addresses, or phone numbers, it was considered safe to give the data to researchers.

he data did contain demographic information such as birth date, gender, and zip code. It is not common for two individuals to have the same birth date, less common for them to also live in the same zip code, and less common still for them to also have the same gender.

According to the Massachusetts voter registration list (available for $20), no one else had the same combination of birth date, gender, and zip code as *William Weld*, who was then the governor. Thus, his medical records were easy to identify in the data provided.

Example 5. AOL case

On Sunday, August 6, 2006, AOL released a 2GB file containing approximately 20 million search queries from 650,000 of its users, which were collected over a period of three months. In addition to the queries themselves, the data set contained information such as which URL from the search results was clicked and what was its ranking.

Although the data set was withdrawn within a few hours, it had already been widely downloaded. The anonymization scheme used to protect the data consisted of assigning a random number (pseudonym) to each AOL user and replacing the user id with this number.

Three days later, two New York Times reporters found and interviewed user number 4417749 from the data set. They tracked down this user based on the semantic information contained in her search queries: the name of a town, several searches with a particular last name, age-related information, etc.

In the case of AOL, there was no single authoritative table (such as a voter list) to link against; instead, there were many scattered sources of information that were used. The privacy breach occurred since AOL failed to reason about these sources and about the semantic content of search queries.

Record linkage Attacks

Record linkage techniques are frequently used to estimate *re-identification* probabilities: the probabilities that users in a data set can be re-identified through auxiliary data. These techniques can often handle varying amounts of noise in the auxiliary data, and are also commonly used for the purpose of data cleaning.

Example 6. Netflix case

Netflix announced a prize for the development of an accurate movie recommendation algorithm. To aid participants in their research efforts, Netflix also released a data set of 100 million ratings for 18,000 movie titles collected from 480,000 randomly chosen users. Personal information had been removed, and user ids were replaced with pseudonyms, as in the AOL

data.

This data set contained movie ratings and the dates when the ratings were created. The high dimensionality of the data set proved to be a tempting target and an attack on such a data set was anticipated to show that *movie ratings can be linked to posts in an online forum*. The Netflix data were attacked shortly after it came out to show that external information (such as IMDB reviews) can indeed be linked to the Netflix data set using techniques that are commonly known as **record linkage**.

3.4. An example

A centralized trusted data collection agency (e.g. City Hospital) collects information from a set of patients. The information collected from each patient consists of identifying information (name), demographic information (age, gender, zip code, and nationality) and the patient's medical condition. The data are put into Table 1. Researchers from another institution (e.g. University), who study how diseases correlate with patients' demographic attributes, can benefit from analyzing these data and have made a request to the hospital for releasing the table.

Table 1. Medical record table

	Name	Age	Gender	Zip Code	Nationality	Condition
1	Ann	28	F	13053	Russian	Heart disease
2	Bruce	29	M	13068	Chinese	Heart disease
3	Cary	21	F	13068	Japanese	Viral infection
4	Dick	23	M	13053	American	Viral infection
5	Eshwar	50	M	14853	Indian	Cancer
6	Fox	55	M	14750	Japanese	Flu
7	Gary	47	M	14562	Chinese	Heart disease
8	Helen	49	F	14821	Korean	Flu
9	Igor	31	M	13222	American	Cancer
10	Jean	37	F	13227	American	Cancer
11	Ken	36	M	13228	American	Cancer
12	Lewis	35	M	13221	American	Cancer

The question is whether releasing Table 1 is safe. In fact, the hospital has a privacy policy that prevents it from releasing patients' identifying information. Obviously, releasing Table 1, which

contains names, would violate this policy.

Does removal of names from Table 1 make the table safe for release?

Consider a researcher who knows that *Eshwar* is a 50-year-old Indian male having zip code 14853. He also knows that Eshwar visited the City Hospital several times. If the researcher saw this table with names removed, he would be almost sure that his friend Eshwar got cancer, because the 5th record is the only record that matches his knowledge about Eshwar. Age, gender, zip code, and nationality are called **quasi-identifier** attributes, because by looking at these attributes an adversary may potentially identify an individual in the data set.

One way to prevent the researcher from being able to infer Eshwar's medical condition is to make sure that, in the released data, *no patient can be distinguished from a group of k patients by using age, gender, zip code, and nationality*. We call a table that satisfies this criterion a **k-anonymous** table.

Table 2 is a modified version of the medical record table that is 4-anonymous, where names have been removed, age values have been generalized to age groups, gender values have been generalized to Any, zip codes have been generalized to the first few digits and nationality values have been generalized to different geographical granularities.

Now, when the researcher sees this generalized table, he only knows that Eshwar's record is in the second group and is not sure whether Eshwar had flu or cancer.

Table 2 fulfills 4-anonimity, i.e. no record can be distinguished from a group of 4 based on Age, Gender, Zip Code, and Nationality. Does the Table 2 is now safe for release?

Table 2. Generalized medical record table

		Age	Gender	Zip Code	Nationality	Condition
(An n)	1	20-29	Any	130*	Any	Heart disease
(Bru ce)	2					Heart disease
(Car y)	3					Viral infection
(Dic k)	4					Viral infection

		Age	Gender	Zip Code	Nationality	Condition
(Eshwar)	5	40-59	Any	14□**	Asian	Cancer
(Fox)	6					Flu
(Gary)	7					Heart disease
(Helen)	8					Flu
(Igor)	9	30-39	Any	1322□	American	Cancer
(Jean)	10					Cancer
(Ken)	11					Cancer
(Lewis)	12					Cancer

We will see later that Table 2 is still not safe for release.

For now, let us assume that the City Hospital somehow decides to consider 4-anonymous tables to be safe for release.

But in addition to Table 2, there are many 4-anonymous tables which can be derived from the medical record table. For instance, Table 3 is another 4-anonymous table derived from the original Table 1. The 2nd record has been swapped with the 8th record, and the 4th record has been swapped with the 10th record.

Which table should the City Hospital choose to release? Table 2 or Table 3?

Table 3. Another generalized medical record table

		Age	Gender	Zip Code	Nationality	Condition
(Ann)	1	20-59	F	1□***	Any	Heart disease
(Helen)	8					Flu
(Cary)	3					Viral infection
(Jean)	10					Cancer

		Age	Gender	Zip Code	Nationality	Condition
(Eshwar)	5	20-59	M	1□***	Asian	Cancer
(Fox)	6					Flu
(Gary)	7					Heart disease
(Bruce)	2					Heart disease
(Igor)	9	20-39	M	13□**	American	Cancer
(Dick)	4					Viral infection
(Ken)	11					Cancer
(Lewis)	12					Cancer

Intuitively, the hospital should choose the one that is the most useful for the researchers who request for the data. Assume that the primary objective of the researchers is *to understand how diseases correlated with genders*. Since researchers want as little replacement of a gender value by Any as possible, Table 3 is a better choice than Table 2 in terms of the number of replacements of gender values by Any.

3.5. Privacy-preserving data publishing

Swapping

Table 4. Swapped records in medical record table

	Age	Gender	Zip Code	Nationality	Condition
(Ann)	50	F	13053	Russian	Heart disease
(Bruce)	29	M	13068	Chinese	Heart disease
(Cary)	21	F	13068	Japanese	Viral infection
(Dick)	23	M	13053	American	Viral infection
(Eshwar)	28	M	14853	Indian	Cancer
(Fox)	55	M	14750	Japanese	Flu
(Gary)	47	M	14562	Chinese	Heart disease
(Helen)	49	F	14821	Korean	Flu
(Igor)	31	M	13222	American	Cancer

	Age	Gender	Zip Code	Nationality	Condition
(Jean)	37	F	13227	American	Cancer
(Ken)	36	M	13228	American	Cancer
(Lewis)	35	M	13221	American	Cancer

Bucketization

Table 5. Bucketized medical record table

	Age	Gender	Zip Code	Nationality	BID		BID	Condition
(Ann)	28	F	13053	Russian	1		1	Heart disease
(Bruce)	29	M	13068	Chinese	1		1	Heart disease
(Cary)	21	F	13068	Japanese	1		1	Viral infection
(Dick)	23	M	13053	American	1		1	Viral infection
(Eshwar)	50	M	14853	Indian	2		2	Cancer
(Fox)	55	M	14750	Japanese	2		2	Flu
(Gary)	47	M	14562	Chinese	2		2	Heart disease
(Helen)	49	F	14821	Korean	2		2	Flu
(Igor)	31	M	13222	American	3		3	Cancer
(Jean)	37	F	13227	American	3		3	Cancer
(Ken)	36	M	13228	American	3		3	Cancer
(Lewis)	35	M	13221	American	3		3	Cancer

Table 6. Randomized medical record table

		Age	Gender	Zip Code	Nationality	Condition
(Ann)	1	30	F	13073	Russian	Heart disease
(Bruce)	2	28	M	13121	American	Heart disease
(Cary)	3	22	M	13024	Japanese	Cancer
(Dick)	4	20	M	13030	American	Viral infection
…	…	…	…	…	…	…

4. Open APIs

Digital transformation is not only about individual enterprises' digital transformations, but the entire industry's cooperation to reap more rewards for enterprises as well as customers.

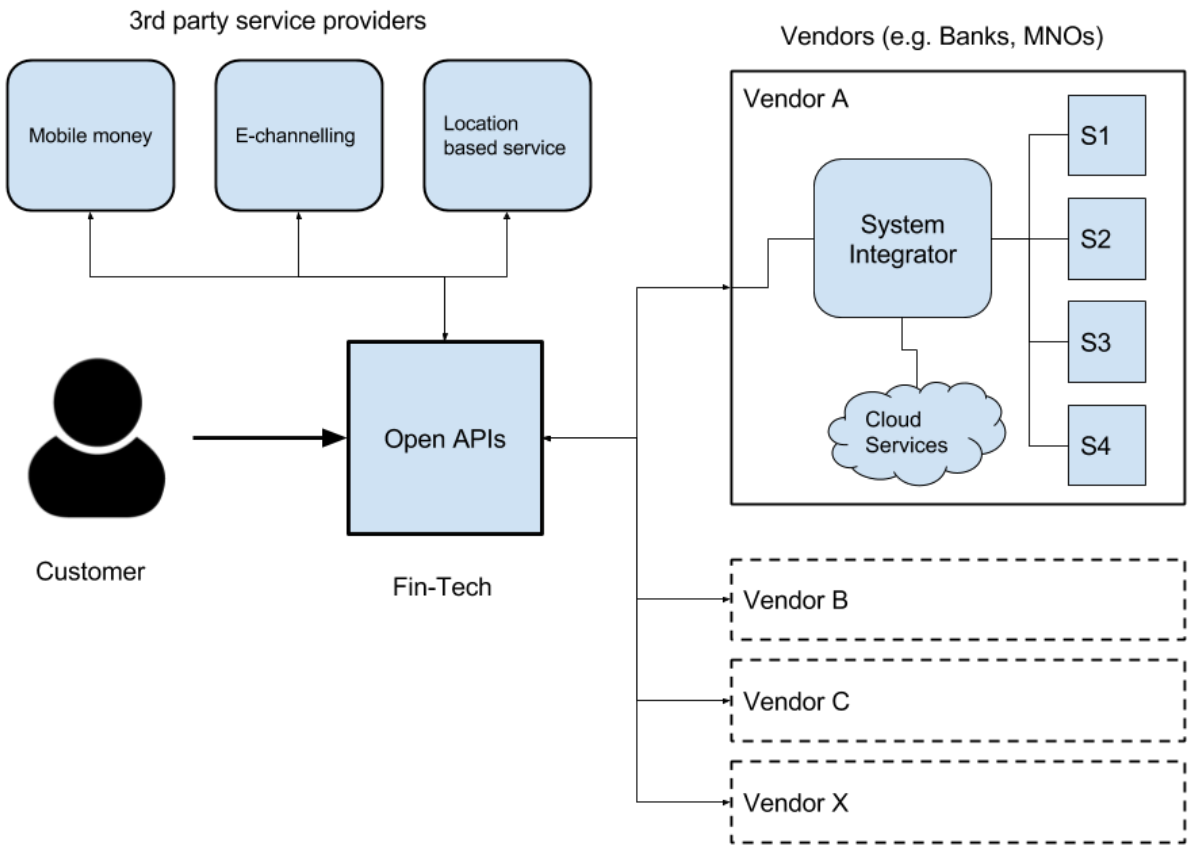
That is where the concept of “Open APIs” or “open standards” came into the grandstand. One of the hot topics in the European region is the **Payment Services Directive 2 (PSD2)** compliance regulation, which required all the banks operating in the EU to be compatible with PSD2 by January 2018.

Even though this came as a regulation, this is a revelation in the way people deal with their banks.

GSMA API Exchange is another set of Open APIs which allows multiple Mobile Network Operators (MNO) to interconnect with each other and reap the benefits of a much larger customer base than doing business with their own customer bases.

Open API standards provides a mechanism to interconnect enterprises which are offering a similar type of services to their customers (PSD2 for financial services and GSMA API Exchange for mobile network services) and make them share the customer bases they have so that they can benefit from a somewhat larger, aggregated customer base.

From the customers' perspective, they will also be able to use multiple accounts/profiles when they purchase services from 3rd parties.



Different vendors can expose information about their customers with the customers' consent in a unified manner. Their internal implementations for providing these APIs can be different, but the APIs are unified. Using Open APIs, third party service providers (e.g. online shopping, merchants, location-based services, etc) can engage with the customers when customers are purchasing their products or services.

Example 7. Banking uses case

You want to buy a laptop from Amazon.com and you need to make the payment using your existing bank account rather than a credit card.

When you check out your item from Amazon.com, the website will provide you with the option to select from which bank account you are going to make the payment. This is achieved through the Open API which has been used by all your banks to expose your account information. You select Bank A, and you will be redirected to Bank A's web site. Now you confirm with Bank A that you allow Amazon.com to debit the relevant amount for the product which you are purchasing. That's all. No credit card. No third-party credit card providers.

This example showcases the power of Open APIs in a banking use case, but this is true for all sorts of different industries. GSMA is being used in different places across the globe for various use cases like mobile connect, mobile ID, etc. It has allowed people who didn't have any facilities to connect with entities like banks using their mobile phone.

4.1. PSD2 regulation

https://eur-lex.europa.eu/legal-content/EN/TXT/?uri=intcom:C(2017)7782

Payment Services Directive (PSD2): Regulatory Technical Standards (RTS) enabling consumers to benefit from safer and more innovative electronic payments

Rationale, objectives and process

What are the objectives of PSD2?

The revised Payment Services Directive (PSD2), which enters into application on 13 January 2018, will facilitate innovation, competition and efficiency. It will give consumers more and better choice in the EU retail payment market. At the same time, it will introduce higher security standards for online payments. This will make consumers more confident when buying online. PSD2 scope extends to innovative payment services and new providers in the market, such as FinTechs. These players are also called third party payment services providers (TPPs). TPPs include:

- payment initiation services providers (PISPs): these initiate payments on behalf of customers. They give assurance to retailers that the money is on its way.
- aggregators and account information service providers (AISPs): these give an overview of available accounts and balances to their customers.

What are the objectives of the Regulatory Technical Standard?

Market players need specific requirements to comply with the new obligations in PSD2. To this end, PSD2 empowers the Commission to adopt regulatory technical standards (RTS) on the basis of the draft submitted by the European Banking Authority (EBA).

The security measures outlined in the RTS stem from two key objectives of PSD2: ensuring consumer protection and enhancing competition and level playing field in a rapidly changing

market environment.

Consumer protection is achieved through increasing the level of security of electronic payments. This is why the RTS introduces security requirements that payment service providers must observe when they process payments or providing payment-related services. Payment services providers include banks and other payment institutions. These standards define the requirements for strong customer authentication and the instances when payment service providers can be exempted from such authentication.

Another key objective is bringing more competition and innovation in the retail payment market. In this context, the RTS includes two new types of payment services, the so-called payment initiation services and the account information services.

Has the Commission amended the RTS submitted by the EBA?

The Commission made some limited substantive amendments to the draft RTS submitted by the EBA. This was done to better reflect the mandate of PSD2 and to provide further clarity and certainty to all interested parties.

When will the new rules become applicable?

PSD2 will become applicable as of 13 January 2018, except for the security measures outlined in the RTS. These will become applicable 18 months after the date of entry into force of the RTS. Subject to the agreement of the Council and the European Parliament the RTS is due to become applicable around September 2019. To what type of accounts will this RTS apply to? The RTS only covers payment accounts in the scope of PSD2, i.e. accounts held by one or more payment service users which can be used for the execution of payment transactions. While this definition has not changed with the adoption of PSD2, the list of payment services has evolved. It includes payment initiation services and account information services.

Strong Customer Authentication (SCA)

How will the new RTS enhance security for electronic payments?

Thanks to PSD2 consumers will be better protected when they make electronic payments or transactions (such as using their online banking or buying online). The RTS makes strong customer authentication (SCA) the basis for accessing one's payment account, as well as for making payments online.

This means that to prove their identity users will have to provide at least two separate elements out of these three: * something they know (a password or PIN code); * something they own (a card, a mobile phone); and * something they are (biometrics, e.g. fingerprint or iris scan).

Strong customer authentication is already commonly used throughout the EU. For example, when customers pay with a card at brick-and-mortar shops they are required to validate a transaction by typing their PIN codes on card readers. However, this is not the case for electronic remote payment transactions, be it a card payment or a credit transfer from an online bank. For these transactions, SCA already is applied in some EU countries only (including Belgium, the Netherlands and Sweden). In other EU countries some payment service providers apply SCA on a voluntary basis.

The RTS sets out that strong customer authentication must be used to access one's payment account and to make online payments. Banks and other payment service providers will have to put in place the necessary infrastructure for SCA. They will also have to improve fraud management. Consumers and merchants will have to be equipped and trained to be able to operate in a SCA environment.

The RTS also allows for exemptions from strong customer authentication. This is to avoid disrupting the ways consumers, merchants and payment service providers operate today. It is also because there may be alternative authentication mechanisms that are equally safe and secure. However, payment service providers that wish to be exempted from SCA must first apply mechanisms for monitoring transactions to assess if the risk of fraud is low.

All payment service providers will need to prove the implementation, testing and auditing of the security measures. In case of a fraudulent payment, consumers will be entitled to a full reimbursement. For online payments, security will be further enhanced by linking, via a one-time password, the online transaction to its amount and to the beneficiary of the payment. This practice ensures that in case of hacking, the information obtained by a potential fraudster cannot be re-used by for initiating another transaction. This procedure is already in application in countries such as Belgium and has led to significant fraud reduction for online payments.

When will strong customer authentication become mandatory?

The use of SCA will become mandatory 18 months after the entry into force of the RTS, i.e. once the RTS is published in the Official Journal of the EU, scheduled for September, 2019.

This will allow payment service providers, including banks, sufficient time to adapt their security systems to the increased security requirements defined in PSD2.

What about security of corporate payments?

The RTS also caters for the security of payments that are carried out in batches. This is the way most corporates make payments, rather than one by one. The new rules also take into account host-to-host machine communication, where for example the IT system of a company communicates with the IT system of a bank to send messages for paying invoices. Security mechanisms for this type of communication systems can be as effective as strong customer authentication. Therefore, they can benefit from an exemption from the SCA, if this is approved by national supervisors.

Could SCA have a negative impact on e-commerce?

The Commission wants to foster the development of e-commerce by building consumer trust. At the same time, the Commission wants to reduce fraud affecting online payments, which are particularly at risk. This entails a higher level of security and may require e-commerce market players to adapt their IT systems or their business models so that they are more secure.

Merchants will still be able to apply risk analysis to transactions with their customers. This method is often applied to card payments. The RTS does not prevent merchants from continuing to do so. Both PSD2 and today's RTS are addressed only to payment service providers, including the banks of the consumers and those of the merchants. Merchants are not in the scope of the RTS. It will be for merchants and their payment service providers to agree on how to meet the objective of reducing fraud.

Common and secure communication

How will common and secure communication work?

PSD2 establishes a framework for new services linked to consumer payment accounts, such as the so-called payment initiation services and account information services. In this context, the RTS specify the requirements for common and secure standards of communication between banks and FinTech companies.

Consumers and companies will be able to grant access to their payment data to third parties providing payments-related services (TPPs). These are, for example, payment initiation services providers (PISPs) and account information service providers (AISPs). TPPs are sometimes FinTech companies, but could also be other banks.

Customers will have to give their consent to the access, use and processing of their data. TPP will not be able to access any other data from the payment account beyond those explicitly authorised by the customer.

Banks will have to put in place a communication channel that allows TPPs to access the data that they need. This communication channel will also enable banks and TPPs to identify each other when accessing customer data and communicate through secure messaging at all times.

Banks may establish this communication channel by adapting their customer online banking interface. They can also create a new dedicated interface that will include all necessary information for the payment service providers.

The rules also specify the contingency safeguards that banks have to put in place when they decide to develop a dedicated interface (the so-called "fall back mechanisms"). The objective of such contingency measures is to ensure continuity of service as well as fair competition in this market.

What makes a good dedicated communication interface?

According to the RTS, all communication interfaces, whether dedicated or not, will be subject to a 3-month 'prototype' test and a 3-month 'live' test in market conditions. The test will allow market players to assess the quality of the interfaces put in place by account servicing payment service providers, including banks.

A quality dedicated communication interface should offer at all times the same level of availability and performance the interfaces made available to a consumer or a company for directly accessing their payment account online. In addition, a quality dedicated interface should not create obstacles to the provision of payment initiation or account information services.

Payment service providers, including banks, will have to define transparent key performance indicators and service level targets for the dedicated communication interfaces, if they decided to set them up. These performance indicators should be at least as stringent as those set for the online payment and banking platforms used by the customers.

The Commission is promoting the set-up of a market group, composed of representatives from banks, payment initiation and account information service providers and payment service users. This group will review the quality of dedicated communication interfaces. This follows up on the

work carried out by the Euro Retail Payments Board on payment initiation services.

Can banks be exempted from setting up a fall-back mechanism?

Yes. They can be exempted if they put in place a fully functional dedicated communication interface responding to the quality criteria defined by the regulatory technical standards. National authorities will grant the exemption to individual banks by national authorities, after having consulted the EBA. The role of the EBA is to ensure that national authorities have similar interpretations when they assess of the quality of dedicated interfaces. Divergences of interpretation would be detrimental to the good functioning of the Single Market for retail payments.

A national authority can revoke the exemption where a dedicated communication interface no longer meets the quality criteria defined under the RTS, for more than two consecutive calendar weeks. In this case, the national authority also informs EBA. The national authority also ensures that the bank establishes an automated fall-back mechanism. This must happen in the shortest time possible, and within 2 months at the latest.

Protection of personal data

How is personal data protected?

Account holders can exercise control over the transmission of their personal data under both PSD2 and the Data Protection Directive (under the General Data Protection Regulation or GDPR as from May 25 of 2018). No data processing can take place without the express agreement of the consumer. In addition, payment service providers can only access and process the personal data necessary for the provision of the services the consumer has agreed to.

PSD2 regulates the provision of new payment services which require access to the payment service user´s data. For instance, this could mean initiating a payment from the customer's account or aggregating the information on one or multiple payment accounts held with one or more payment service providers for personal finance management. When a consumer seeks to benefit from these new payment services, she or he will have to request such service explicitly from the relevant provider.

Payment service providers must inform their customers about how their data will be processed. They will also have to comply with other customers' rights under data protection rules, such as the right of access or the right to be forgotten. All payment service providers (banks, payment institutions or new providers) must comply with the data protection rules when they process personal data for payment services.

What data can TPPs access and use via "screen scraping"?

PSD2 prohibits TPPs from accessing any other data from the customer payment account beyond those explicitly authorised by the customer. Customers will have to agree on the access, use and processing of these data.

With these new rules, ***it will no longer be allowed to access the customer's data through the use of the techniques of "screen scraping"***.

Screen scraping

Screen scraping means accessing the data through the customer interface with the use of the customer's security credentials. Through screen scraping, TPPs can access customer data without any further identification vis-à-vis the banks.

Banks will have to put in place a communication channel that allows TPPs to access the data that they need in accordance with PSD2. The channel will also be used to enable banks and TPPs to identify each other when accessing these data. It will also allow them to communicate through secure messaging at all times.

Banks may establish this communication channel by adapting their customer online banking interface. They may also create a new dedicated interface that will include all necessary information for the relevant payment service providers.

The RTS specifies the contingency safeguards that banks shall put in place if they decide to develop a dedicated interface. This will ensure fair competition and business continuity for TPPs.

Transition period

Can TPPs continue to use screen scraping during the transition period?

There will be transition period between the application date of PSD2 (13 January 2018) and the application date of the RTS (18 months after publication of the delegated act in the Official Journal of the EU). Payment market players need this transition period to upgrade their payments security systems so that they meet the RTS requirements.

This means that the PSD2 provisions on strong customer authentication and on secure communication, which are directly specified in the RTS, will not apply immediately. In other words, the application of security measures in Articles 65, 67 and 97 of PSD2 is postponed until the RTS becomes applicable. However, those parts of Articles 65, 67 and 97 that are not dependent on the RTS will apply as of 13 January 2018.

The delayed application of the RTS should not create any difficulties for the provision of existing payment-related services by market players that have been operating in Member States before 13 January 2016. Article 115(5) of PSD2 ensures the continuity of these services. These payment services providers should still apply for the relevant authorisation under PSD2 to their national authority as soon as possible.

New payment initiation service providers and account information service providers willing to provide these services must obtain the relevant authorisation to enter the market during the transition period.

Implications for screen scraping

One of the hottest topics that has been the subject of intensive lobbying on both the banking side and the fintech side has been that of the so called "screen scraping". In general terms, the banking side favored the banning of screen scraping on the basis of security and costs. The

fintech side favored the use of screen scraping as a safety net for situations where APIs don't function properly.

This is the first time an industry is forced to give free access to their customer's data to a potential competitor. In some cases it might be a win-win situation, but not necessarily, so there are plenty of reasons to justify the positions of both sides. However, it is undeniable that there will be many situations where there will not be an adequate incentive to provide an industrial strength API service.

The RTS still gives the bank a choice of two interfaces; Implementing a dedicated interface/API, which is the best possible interface if it is well implemented and cared for, or implementing electronic banking with a TPP identification layer in front of it.

In case a bank decides to go for the API route, it will also have to set an identification layer in front of the electronic banking (or share it with the identification layer of the API) which will then serve as a fallback mechanism. What this means is that third parties will have to use the API if it exists, and if it works properly. In case it doesn't work properly, then the third parties will be able to access via electronic banking after identifying themselves with a qualified certificate, using what was previously known as screen scraping.

There is however a possibility for banks to have an exemption from providing a fallback mechanism. National authorities will be able to provide such exemptions for those banks that during a 6 month period prove that their API's perform according to a set of KPIs that are still to be defined by a mix of banks and fintechs. Of course this exemption can be retired if the API ceases to function properly for a period of time (2 weeks, according to the RTS).

Lastly, screen scraping will still be used for non PSD2 regulated information, such as deposits, loans, pension plans, shares, investment funds, etc. unless banks choose to offer an API that works better than screen scraping of electronic banking, and, is either free or cheaper than screen scraping.

4.2. References

https://dzone.com/articles/top-5-rest-api-security-guidelines

https://medium.freecodecamp.org/how-to-securely-store-api-keys-4ff3ea19ebda