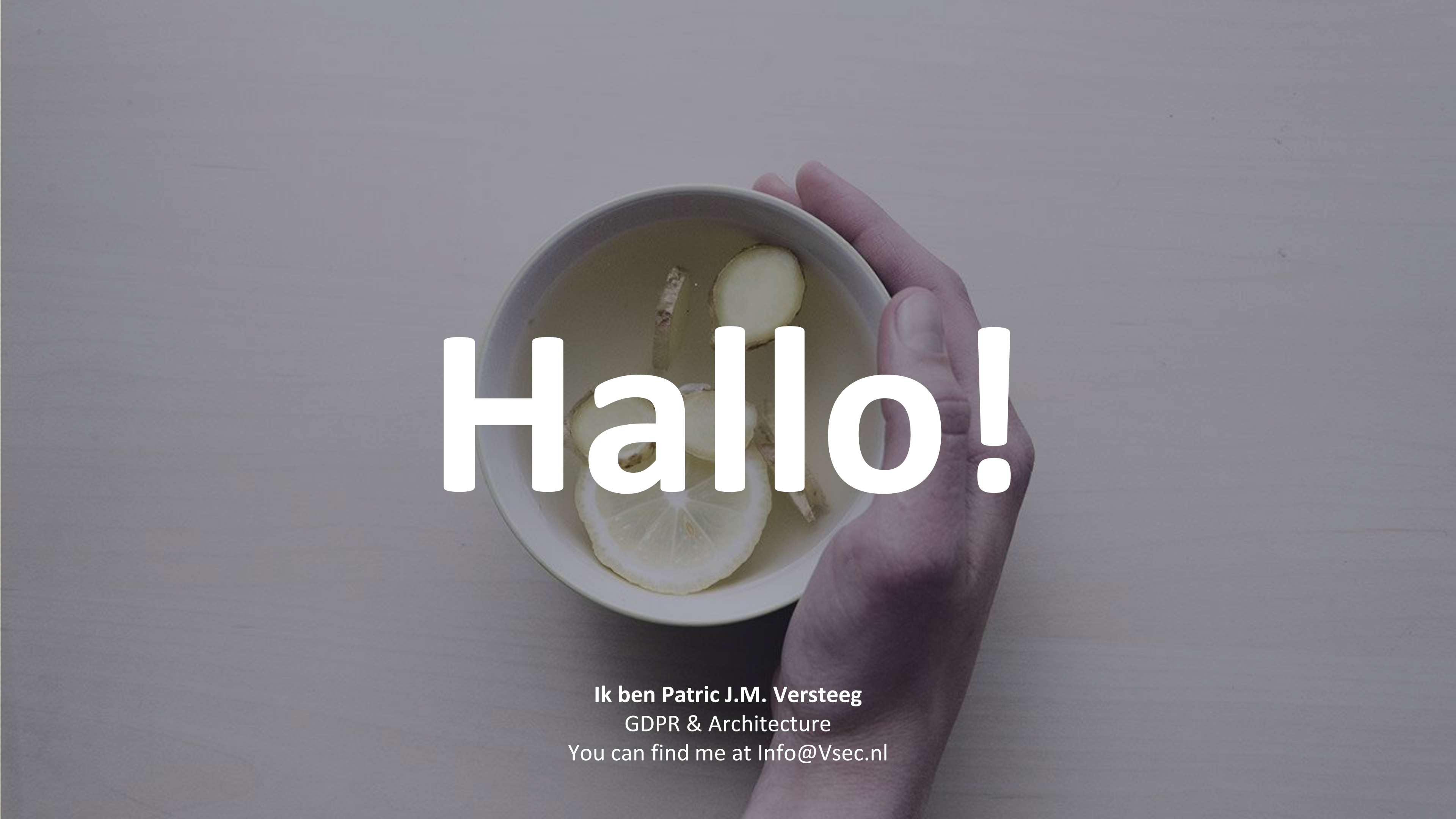
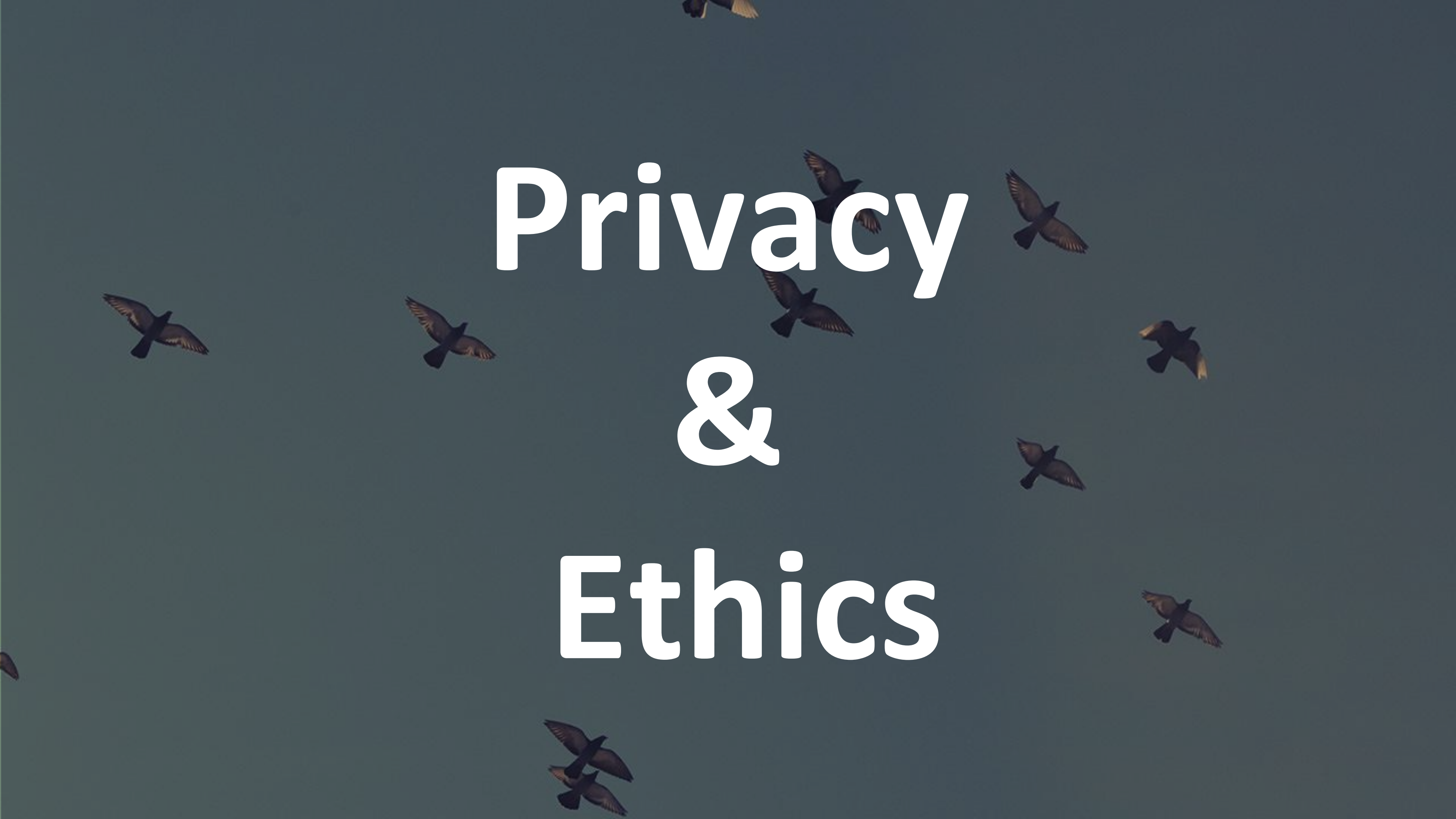
GDPR & Architecture



Hallo!

Ik ben Patric J.M. Versteeg
GDPR & Architecture
You can find me at Info@Vsec.nl



Privacy & Ethics

Privacy by Design

Privacy by Default

“

- Customer value first

“

- Doelgroep centraal

“

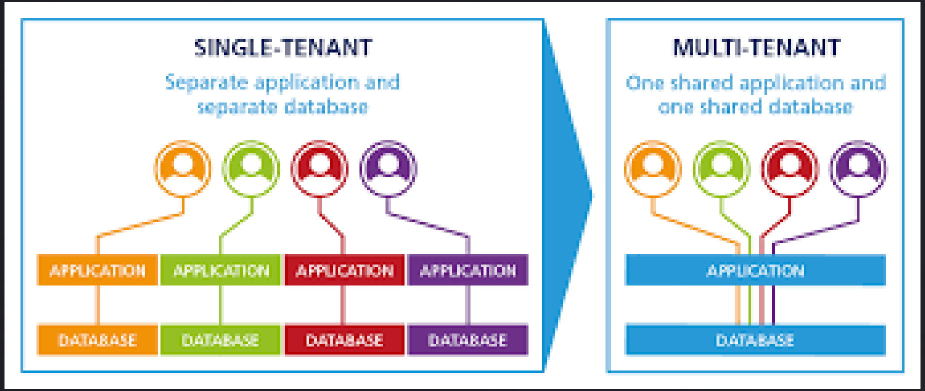
- Meereizen met de klant

- Autonome business-units

“

- Data is de belangrijkste asset

“



- Multi tenant IT

“

- Iteratief evolueren door versioneren

Thanks!

Any questions?

Info@VSec.nl

Main principle: privacy & security by design

Description:

- System design for the highest data classification must follow all privacy & security principles
- For each data implementation a data classification must be made. For any classification lower than the highest only a selection of the principles is applicable.
- The implementation of the privacy & security measures must be measurable, auditable and indisputable.

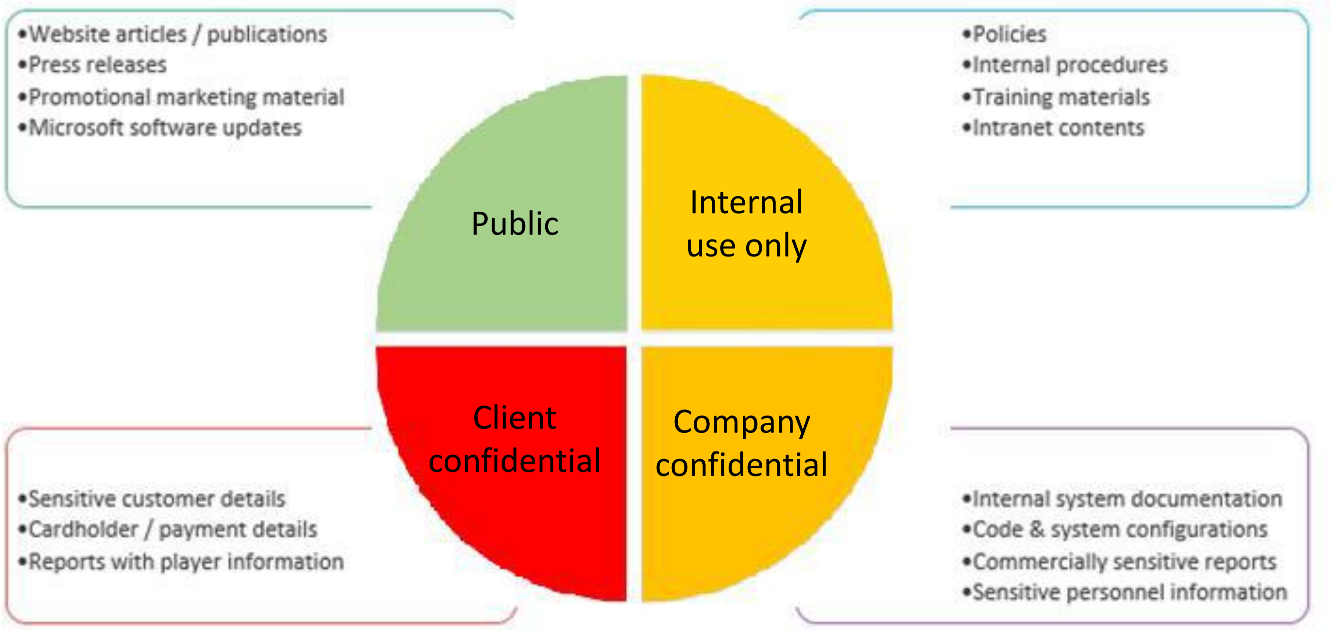
Rationale:

Security & privacy measures must be taken to prevent lawful action, e.g. as a consequence of GDPR and data protection act. Security & privacy measures are also important to prevent loss of reputation, trust and consequently loss of customers. Part of the security & privacy measures that can be taken can (and therefore must) be resolved in our IT systems. The IT system must facilitate that human error and fraud can be prevented.

Implications:

- Always perform a data classification.
- Follow the privacy & security principles based on that data classification.
- Apply the principles to each process and system that deals with that data.

Data classification



Public

- Website articles / publications
- Press releases
- Promotional marketing material
- Microsoft software updates

Internal use only

- Policies
- Internal procedures
- Training materials
- Intranet contents

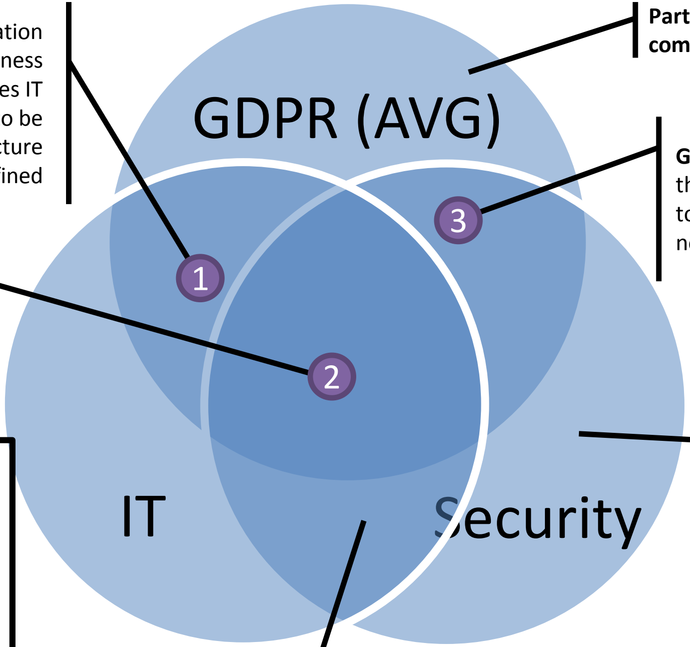
Client confidential

- Sensitive customer details
- Cardholder / payment details
- Reports with player information

Company confidential

- Internal system documentation
- Code & system configurations
- Commercially sensitive reports
- Sensitive personnel information

Overlap GDPR, security and IT



GDPR and IT: next to information security GDPR (business requirements) enforces IT measures e.g. “right to be forgotten”. Privacy architecture principles are defined

Part of legislation compliancy

GDPR and Security: About 80% of the security guidelines contributes to the GDPR. This implies both IT as not IT related guidelines.

GDPR and IT and Security: security architecture principles are defined.

Part of Risk Management Beschikbaarheid, Integriteit en Vertrouwelijkheid (BIV) Based on best practices within information security: **ISO 27002**

1. GDPR requirements based on articles with IT implications. E.g. “right to be forgotten” or “Cookie wall”. Security requirements supporting GDPR implementation, part is formulated as security architecture principle with IT implications. E.g. data encryption.
2. Security requirements supporting GDPR for business and IT processes (e.g. processes, procedures)

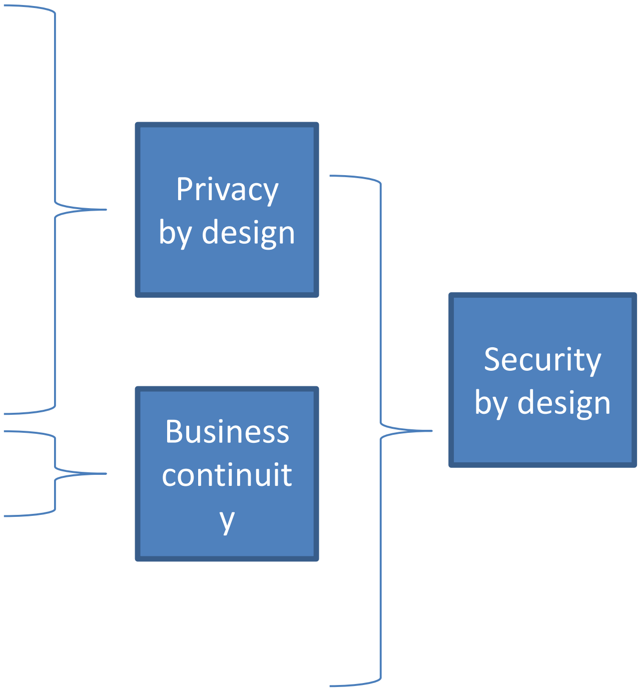
Security by design: implement measures according to data classification and a risk assessment

Risk Management system (ISO 31000 Risk Management)

Information security based on **ISO 27001**

Security & privacy principles

- Data minimalisation
- Pseudonimity
- Data subject rights
- Data deletion
- Standardized encryption
- Access control:
  - Authentication by AD
  - Need to know
  - Separation of duties
  - Differentiate access means
  - Zoning the AD
- All business means registered in a CMDB
- Ownership of all IT artefacts
- Single point of truth
- Audit trail
- Hardening / stripping
- Cloud is conditionally allowed



Privacy by design

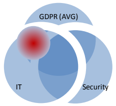
Business continuit y

Security by design

Applicability of privacy principles

	Client confidential	Company confidential	Internal use only	Public
Access by a person *(no data storage, in front of the API)* API - monitoring	Data minimalisation Pseudonimity Standardized encryption Access control: • Authentication by AD • Need to know • Separation of duties • Differentiate access means • Zoning the AD Data protection by default Data deletion (interface)	Data minimalisation Pseudonimity (HR) Standardized encryption Access control: • Authentication by AD • Need to know • Separation of duties • Differentiate access means • Zoning the AD Data protection by default Data deletion (interface)	Access control: • Authentication by AD • Differentiate access means • Zoning the AD Data deletion (interface)	None
API - servicing Automated processing *(no direct data view, behind the API)*	Data minimalisation Pseudonimity Standardized encryption Access control: • Separation of duties Data protection by default Data deletion	Data minimalisation Pseudonimity (HR) Standardized encryption Access control: • Separation of duties Data protection by default Data deletion	Data deletion	None

Data minimalisation



Description:

Only data that is required for the business function at hand is allowed to be processed in that function. So no data that is not necessary for that business function may be processed. This does not extend to subfunctions in individual software components within that business function.

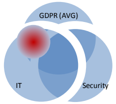
Rationale:

By law. This reduces the risk of spreading personal (privacy sensitive) data. The purpose is to minimize the access of persons to data they do not need to perform a business function. The purpose is not to restrict each individual software component, since this is practically impossible.

Implications:

- Each screen design / data entry must be restricted to require only information from the customer / prospect that is needed in the process. The following information can be asked:
  - Information needed to execute the service (e.g. name, adress)
  - Information needed to inform the customer or for marketing within the current brand
  - Information needed for tax purposes or required by law
- The following information cannot be asked:
  - Sensitive data such as believes, political preference, skin colour, personal photo's, etc.
  - Selling other brands (cross selling), unless the customer has given permission to this end
- When asking to fill in information that is not required in the process, at the very least it cannot be an obligatory field.
- Customer information is only made available to authorized personnel on a need to know basis (see principle "need to know"), and then only the data that is needed to perform the function required.
- When data is shared with a person or third party, only the data can be shared that is necessary for that party to perform the function required.
- Data can flow through software components even when that component does not need that particular data, as long as this does not become visible to unauthorized personnel and the component is secured in line with the classification of that data.
- The OTA (test) and P (production) are segregated environments. Production data may be used in test environment only when it is anonymised. Otherwise fake production data must be used in test.

Pseudonimity



Description:

Keep data that can identify a person separate from all other data as much as possible.

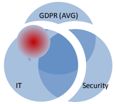
Rationale:

When the identification of a person is limited to the situations in which you must (e.g. when dealing with that person), the chance that a person is identified in undesirable context is made as low as possible.

Implications:

- Keep identification data separate from other data. We do this by having a customer administration separate from all other data about the customer.
- Data that can identify a person can be hashed, such as name, adress, place information in a customer administration.
- In situations where the real customer does not need to be known, a pseudo-ID must be used. E.g. in data analytics, IT tests, etc.
- Data that identifies a person will be connected to other data only when this is explicitly necessary.

Data subject rights



Description:

Prospect data may be processed only after registered permission by the Prospect. Henceforth this data must be protected by offering insight in registered data, withdrawal of permission, and deletion of data on customer request. Also, the data must be kept up to date.

Rationale:

By law (ref. article 15-22 of the GDPR https://www.eugdpr.org/the-regulation.html). Only when a prospect gives permission, his data may be processed. A customer must be in control of the data we register about him and he can expect us to keep his data actual for as far as possible.

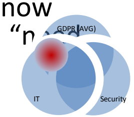
Implications:

- A prospect can give permission by ticking a box. This box cannot be ticked in advance. It must be active permission by the prospect, not passive. E.g. Permission is granted when the prospect clicks “agreed” on a webform while the privacy statement is clearly noted on the site.
- Register opt-ins and opt-outs.
- When building a system, the following services must be provided to the customer:
  - To gain insight in all data we have registered about the customer
  - To withdraw a particular permission.
  - To delete all data (right to be forgotten), which concerns all permissions and all gathered data.
  - To give permission to register an “opt-out” not to contact him ever again. This resolves the paradox of a customer that wants to be forgotten (and implies that he does not want to receive mail ever again). Basically, the customer gives permission for one service only: to make sure he does not get any information from us.
- When building a system, create a process to clean and update data on a regular basis, so that data is kept as actual as possible. A change in data must be registered throughout so that data is kept actual.
- For each automated rejection of a prospect/customer request (e.g. a blacklisted customer), also the human processing and (re)evaluation of this request & rejection must be implemented.
- For all information requested from the prospect / customer, we must inform him/her for what that information is used in our process. E.g. this could be implemented by a clickable question mark next to each field in the form.
- Cookie wall not allowed: a customer has the right to visit the site and perform actions on it even when he does not agree with cookies.

Data deletion

Description:

For all data a term of deletion must be specified, based on reasonable arguments related to the need to know When processing is no longer required, data must be deleted. Terms by law, such as tax regulations, are a “need to know” argument.



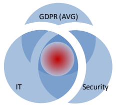
Rationale:

By law. It is only allowed to process data for which permission is granted and as long as you need to know the data for processing.

Implications:

- When building a system, gain insight in all data, where it is stored and determine what the term is after which it must be removed.
- Implement removal of data by creating (automated) data deletion processes on all components that contain personal data. In order to reduce complexity, follow this solution direction:
  - Select components that hold lead (customer) data. This is data that is key to determine for a whole set of data what the term of deletion must be. E.g. when a customer exits the organisation, after a term his data must be deleted. The key to all data that must be deleted is the customer identifying data in the customer database. All data such as participation, vip card, etc can then be deleted at the same time.
  - Each component that holds lead customer data has its own rules for deletion. When data must be deleted, it sends a trigger to a process that facilitates deletion of data that depend on it. This process holds the rules for what the dependent data is. It also holds the rules for what data to delete when a customer wants to be forgotten.
  - Each component must offer a service to delete data. The deletion process can use that service.
- When insert only is practised, the data cannot be deleted. Then it must be made inaccessible, e.g. by inserting a deletion or obscuring the key to the data.
- Removal includes data in a datawarehouse. Only when data is anonymized, it may be kept longer for statistical reasons.
- All devices (including e.g. printers) must be cleaned from all data when being taken out of production or when a device is handed in by a user (also when the device will be used by a next user)
- Data that needs to be kept for legal / tax purposes only can be deleted and saved in a back-up with very limited access. When applicable, the information must be stripped from photo’s of customers.

Standardized plugable encryption



Description:

Encryption must be implemented conform a known standard in a pluggable manner. Minimal level depending on situation.

Rationale:

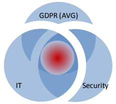
Using a standardized type of encryption of a certain quality is a proven method as opposed to a proprietary form of encryption. A proven method is to be preferred over non proven when it comes to encryption.

pluggable: we can easily change to a different standard, when it is compromised.

Implications:

- Use this encryption any time encryption is required.
- See other principles when encryption is required.

Authentication by AD



Description:

Single sign on should be used when possible. Internal persons should be authenticated towards an active directory (AD). Access to each application is provided by the AD such that a person gains access to all applications applicable to that persons role, using one means of access.

Rationale:

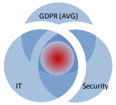
Reducing the number of access points for a person reduces the number of passwords to be memorized and the number of possible entry points for unauthorized personnel.

Using an AD increases access control and access management, since authorization can be controlled on a limited number of applications (all applicable AD's).

Implications:

- No application has its own access control.
- There must be an appropriate AD installed to serve each user role.
- Single sign on is a good way to apply Authentication by AD.

Need to know



Description:

Access to data is limited to persons and systems that need access to that data to perform their role. Access to other data must be made as inaccessible as possible.

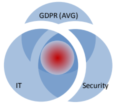
Rationale:

The need to know principle reduces the number of people and systems having access to data and reduces the possibilities to manipulate that data. In turn this reduces the number of possibilities to commit fraud. It also provides the possibility to manage a separation of duties (see next principle). The need to know principle helps to reduce the number of mistakes made in data that does not need to be processed.

Implications:

- Role based access is a good way to implement the need to know principle.
- The AD must be configured to provide access to data to those roles that need that access.
- Access control administration must be encrypted, so access to manipulating the roles in the AD is as difficult as possible.

Separation of duties



Description:

Monitoring and fiat must be separated from data input and/or update in role, access and system.

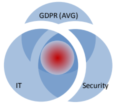
Rationale:

To prevent fraud, a person may never be able to manipulate data as well as give a fiat on it. This also increases data integrity and prevents mistakes.

Implications:

- Each fraud sensitive operation on data must have a control function that is separated in role, access and system configuration.
- System configuration implies that the system must be build and configured to treat the data entry role completely separate from the data fiat / monitor role. This implies separate screens and processflow for these roles and a connection to the AD so these roles can be managed by the AD.

Differentiate access means



Description:

Access to a system within the (secured) company walls can be based on one-factor authentication. Access to a system outside these walls must be based on two-factor authentication. Any person information that is brought outside these walls must be encrypted.

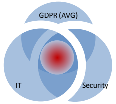
Rationale:

Within the company walls people had to gain physical access to the building, which reduces the risk of unauthorized access to data and dataleaks. Outside the company walls, extra measures must be taken to create the same level of security.

Implications:

- Access via systems inside the company walls can be password based
- Access via systems outside the company walls must be based on password and a second factor
- Password rules to be implemented:
  - Minimum length 12+
  - Check against passwords on a list of most breached passwords, on variety in the password and on trivials such as e-mail adress, login, URL, etc.
- When sharing person information with parties outside the company walls, the interface must be build using encryption and if possible the option to delete data from a distance.

Zoning the AD



Description:

Users, assets and organization domains can be divided in zones. Each zone is managed by a separate AD.

Rationale:

Using an AD to manage and control access improves security, but also increases the risk (single point of failure). By zoning users one can reduce the risk. Also, one can reduce the complexity and number of roles in the AD.

Implications:

- Each zone is managed by a separate AD.
- E.G. some employees have overarching (international) roles. They could have a separate zone to which access to local applications can be promoted. E.g. access to Github required internationally in which case such users can get a role in an overarching AD that gets access to such applications promoted.
- Typical zones can be “customers”, “call center agents”, etc. In order to reduce the risk of failure for the call center, this may be split up further into zones such that at least part of the agents can work in case of failure.
- Assets can be zoned according to their place on the infrastructure (e.g. in a DMZ).
- One could solve the problem of difference in access between international architects and local architects by defining two different architecture roles. However, one can also define two zones (international and local) and have just one architecture role. So creating a matrix of zones and roles reduces the number of roles we define.