Datenschutz aus Europa – was hat dies mit Technik zu tun?

Marit Hansen
Landesbeauftragte für Datenschutz
Schleswig-Holstein

Kiel, 14. November 2018

forum <privatheit> selbstbestimmtes_leben_in_der_digitalen_welt

ULD Unabhängiges Landeszentrum für Datenschutz Schleswig-Holstein

Überblick

1. **Datenschutz und die Datenschutz-Grundverordnung**
2. **(Technik-)Gestaltung – wirklich neu?**
3. **Anforderungen der DSGVO**
4. **Was macht das ULD?**

Sicherheit

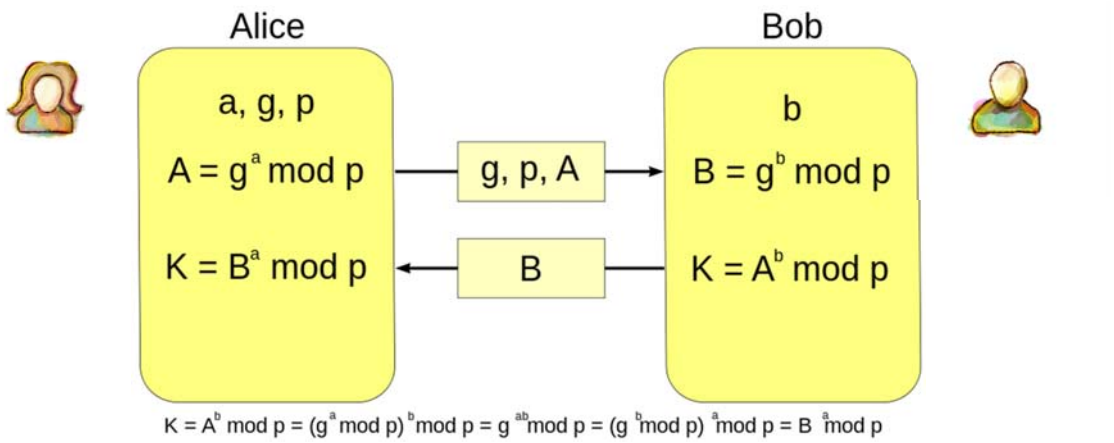
Bild: Das Wortgewand via Pixabay

Datenschutz nötig:

Machtgefälle zwischen Individuen und Organisationen

Bild: beludise via Pixabay

Datenschutz: weiter als IT-Sicherheit



IT-Sicherheit: Die Angreiferin ist Eve (oder Mallory).

Datenschutz: Der Angreifer ist Bob!
(Zumindest auch.)

Datenschutz: Schutz von Individuen

Artikel 1

Gegenstand und Ziele

(1) Diese Verordnung enthält Vorschriften zum Schutz natürlicher Personen bei der Verarbeitung personenbezogener Daten und zum freien Verkehr solcher Daten.

(2) Diese Verordnung schützt die Grundrechte und Grundfreiheiten natürlicher Personen und insbesondere deren Recht auf Schutz personenbezogener Daten.

(3) Der freie Verkehr personenbezogener Daten in der Union darf aus Gründen des Schutzes natürlicher Personen bei der Verarbeitung personenbezogener Daten weder eingeschränkt noch verboten werden.

DSGVO: Vereinheitlichung und Modernisierung

- Idee: Eine für alle
  und
  alle für eine
- Ziel:
  echte Harmonisierung
- Rechtssicherung durch
  Gleichklang der Aufsicht
- Aber: 70 Öffnungsklauseln
  für die Mitgliedstaaten

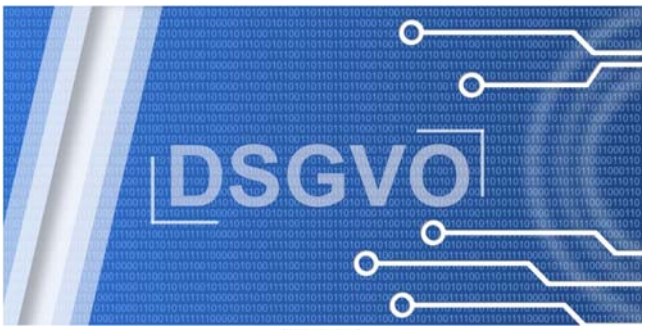


Bild: skylarvision via Pixabay

Vorbemerkung: Wichtigkeit von „by Design“

Erwägungsgrund 4

„The processing of personal data should be designed to serve mankind. […]“

http://www.simulee.com/wp-content/uploads/2015/05/3.jpg

Überblick

 Bild: athree23 via Pixabay

1. **Datenschutz und die Datenschutz-Grundverordnung**
2. **(Technik-)Gestaltung – wirklich neu?**
3. **Anforderungen der DSGVO**
4. **Stand der Technik**
5. **Was macht das ULD?**

Security by Design?


Building Security In
Editor: Gary McGraw, gem@cigital.com

Software Security

Software security is the idea of engineering software so that it continues to function correctly under malicious attack. Most technologies acknowledge this undertaking's importance, but they need some help in understanding how to tackle it. This new department aims to provide that help by exploring software security best practices.

The software security field is a relatively new one. The first books and academic classes on the topic appeared in 2001, demonstrating how recently developers, architects, and computer scientists have started systematically studying how to build secure software. The field's recent appearance is one reason why best practices are neither widely adopted nor obvious.

A central and critical aspect of the computer security problem is a software problem. Software defects with security ramifications—including implementation bugs such as buffer overflows and design flaws such as inconsistent error handling—promise to be with us for years. All too often, malicious intruders can hack into systems by exploiting software defects.[1] Internet-enabled software applications present the most common security risk encountered today, with software's ever-expanding complexity and extensibility adding further fuel to the fire. By any measure, security holes in software are common, and the problem is growing: CERT Coordination Center identified 4,129 reported vulnerabilities in 2003 (a 70 percent increase over 2002, and an almost fourfold increase since 2001).[2,3]

Software security best practices leverage good software engineering practice and involve thinking about security early in the software life cycle, knowing and understanding common threats (including language-based flaws and pitfalls), designing for security, and subjecting all software artifacts to thorough objective risk analyses and testing. Let's look at how software security fits into the overall concept of operational security and examine some best practices for building security in.

...versus application security

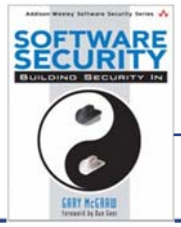
Application security means many different things to many different people. In *IEEE Security & Privacy* magazine, it has come to mean the protection of software after it's already built. Although the notion of protecting software is an important one, it's just plain easier to protect something that's a defect-free than something riddled with vulnerabilities.

Pondering the question, "What is the most effective way to protect software?" can help untangle software security and application security. On one hand, software security is about building secure software: designing software to be secure, making sure that software is secure, and educating software developers, architects, and users about how to build secure software. On the other hand, application security is about protecting software and the systems that software runs in a post-facto way, after development is complete. Issues critical to this subfield include sandboxing code (as the Java virtual machine does), protecting against malicious code, obfuscating code, locking down executables, monitoring programs as they run (especially their input), enforcing the software use policy with technology, and dealing with extensible systems.

Application security follows naturally from a network-centric approach to security, by embracing standard approaches such as prevent and patch[4] and input filtering (trying to block malicious input) and by providing value in a reactive way. Put succinctly, application security is based primarily on finding and fixing known security problems after they've been exploited in fielded systems. Software security—the process of designing, building, and testing software for security—identifies and expunges problems in the software itself. In this way, software security practitioners attempt to build software that can withstand attack proactively. Let me give you a specific example: although there's some real value in stopping buffer overflow attacks by observing HTTP traffic as it arrives over port 80, a superior approach is to fix the broken code and avoid the buffer overflow completely.

...as practiced by operations people

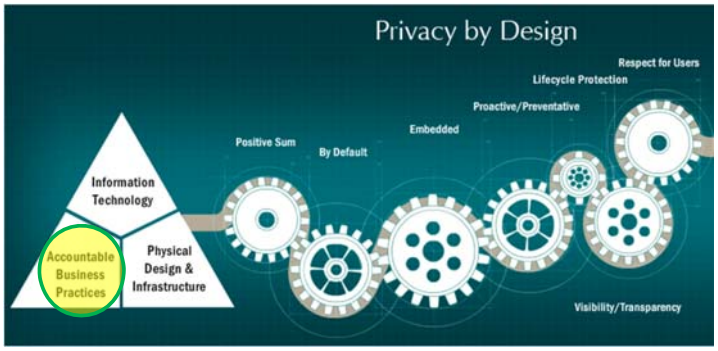
One reason that application security technologies such as firewalls have evolved the way they have is because



„Building Security In“ – Gary McGraw, 2004

> The software security field is a relatively new one. The first books and academic classes on the topic appeared in 2001, demonstrating how recently developers, architects, and computer scientists have started systematically studying how to build secure software. The field's recent appearance is one reason why best practices are neither widely adopted nor obvious.

Privacy by Design + PETs schon seit 1995++



http://privacybydesign.ca/

Überblick

Bild: athree23 via Pixabay

1. Datenschutz und die Datenschutz-Grundverordnung
2. (Technik-)Gestaltung – wirklich neu?
3. Anforderungen der DSGVO
4. Was macht das ULD?

Anforderungen der DSGVO

- Art. 32 DSGVO „Sicherheit der Verarbeitung“
- Art. 25 DSGVO „Datenschutz durch Technikgestaltung und durch datenschutzfreundliche Voreinstellungen“

- Instrumente und Pflichten
  - Datenschutz-Folgenabschätzung (Art. 35 DSGVO)
  - Meldung von Datenschutz-Vorfällen (Art. 33+Art. 34 DSGVO)

- Begriffe
  - „Risiko für die Rechte und Freiheiten natürlicher Personen“
  - „Stand der Technik“
  - Vorhandensein geeigneter Garantien wie „Verschlüsselung oder Pseudonymisierung“

Datenschutz „by Design“ & „by Default“

- Art. 25 DSGVO – mehr als „eingebaute Sicherheit“ (Art. 32 DSGVO)
- Richtet sich an:
  - Datenverarbeiter (primär: Verantwortlicher)
  - Indirekt: Dienstleister und Hersteller von IT-Systemen
- Ziel: Gestaltung von Systemen + Diensten
  von Anfang an über den gesamten Lebenszyklus
  a) datenminimierend
  b) mit möglichst datenschutzfreundlichen Voreinstellungen
- Wichtig für jede Beschaffung + Nachweispflicht

Datenschutz durch Technikgestaltung

Artikel 25 Datenschutz durch Technikgestaltung [...]

(1) Unter Berücksichtigung
des Stands der Technik,
der Implementierungskosten und
der Art, des Umfangs, der Umstände und der Zwecke der Verarbeitung
sowie der unterschiedlichen Eintrittswahrscheinlichkeit und Schwere
der mit der Verarbeitung verbundenen Risiken für die Rechte und Freiheiten natürlicher Personen

Viele möglicherweise begrenzende Bedingungen! ↑↓

trifft der Verantwortliche sowohl zum Zeitpunkt der Festlegung der Mittel für die Verarbeitung als auch zum Zeitpunkt der eigentlichen Verarbeitung geeignete technische und organisatorische Maßnahmen – wie z. B. Pseudonymisierung – trifft, die dafür ausgelegt sind, die Datenschutzgrundsätze wie etwa Datenminimierung wirksam umzusetzen und die notwendigen Garantien in die Verarbeitung aufzunehmen, um den Anforderungen dieser Verordnung zu genügen und die Rechte der betroffenen Personen zu schützen.

Datenschutz durch datenschutzfreundliche Voreinstellungen

Artikel 25 […] durch datenschutzfreundliche Voreinstellungen

(2) Der Verantwortliche trifft geeignete technische und organisatorische Maßnahmen, die sicherstellen, dass durch Voreinstellung ~~grundsätzlich~~ nur personenbezogene Daten, deren Verarbeitung für den jeweiligen bestimmten Verarbeitungszweck erforderlich ist, verarbeitet werden. Diese Verpflichtung gilt für die Menge der erhobenen personenbezogenen Daten, den Umfang ihrer Verarbeitung, ihre Speicherfrist und ihre Zugänglichkeit.

Solche Maßnahmen müssen insbesondere sicherstellen, dass personenbezogene Daten durch Voreinstellungen nicht ohne Eingreifen der Person einer unbestimmten Zahl von natürlichen Personen zugänglich gemacht werden.

Datenschutz „by Design“ & „by Default“ gemäß Erwägungsgrund 78 DS-GVO

- Nachweis durch interne Strategien & t+o Maßnahmen, u.a.
  - Datenminimierung
  - Schnellstmögliche Pseudonymisierung
  - Transparenz in Bezug auf Funktionen+Verarbeitung
  - Ermöglichung der Überwachung der Verarbeitung durch die betroffenen Personen
  - Ermöglichung für Sicherheitsfunktionen „on top“ durch Verantwortlichen
- Ermutigung für Hersteller
- Berücksichtigung in öffentlichen Ausschreibungen

Aggregation
Anonymisierung
Attributbasierte Berechtigungszertifikate
Dashboard
Auskunftsportal
Elektronischer Datenbrief
Machine-readable Policies
Icons
Zweck-Kennzeichnung
Dezentralisierung
Kein Freitext
Automatisches Löschen
Sticky Policies
Schnittstellen zu Selbstdatenschutz-Tools

Überblick

Bild: athree23 via Pixabay

1. **Datenschutz und die Datenschutz-Grundverordnung**
2. **(Technik-)Gestaltung – wirklich neu?**
3. **Anforderungen der DSGVO**
4. **Was macht das ULD?**

Beispiele für Projekte (BMBF-gefördert)

a) AN.ON Next Generation
Anonymität online (www.anon-next.de/)

b) AppPETS
Datenschutzfreundliche Smartphone-Anwendungen ohne Kompromisse (www.app-pets.org/)

c) Forum Privatheit
Interdisziplinäre Erforschung der Weiterentwicklung des Datenschutzes
(www.forum-privatheit.de/)

Beteiligung an ENISA-Reports

- Privacy and Data Protection by Design – from policy to engineering (2015), https://www.enisa.europa.eu/activities/identity-and-trust/library/deliverables/privacy-and-data-protection-by-design
- Readiness Analysis for the Adoption and Evolution of Privacy Enhancing Technologies (2016), https://www.enisa.europa.eu/publications/pets
- PETs controls matrix - A systematic approach for assessing online and mobile privacy tools (2016), https://www.enisa.europa.eu/publications/pets-controls-matrix/pets-controls-matrix-a-systematic-approach-for-assessing-online-and-mobile-privacy-tools
- Privacy Enhancing Technologies: Evolution and State of the Art (2017), https://www.enisa.europa.eu/publications/pets-evolution-and-state-of-the-art
- A tool on Privacy Enhancing Technologies (PETs) knowledge management and maturity assessment (2018), https://www.enisa.europa.eu/publications/pets-maturity-tool
- To come:
  - Data pseudonymisation techniques (2019)
  - Exploring the notion of data protection by default (2019)





https://www.datenschutzzentrum.de/sdm/

Gewährleistungsziele


Ich freue mich
auf die Diskussion.