Grundlagen des Datenschutzes und der IT-Sicherheit (Teil 2d)

Vorlesung im Sommersemester 2024

an der Universität Ulm

von Bernhard C. Witt

2. Grundlagen der IT-Sicherheit

Grundlagen des Datenschutzes		Grundlagen der IT-Sicherheit	
✓	Geschichte des Datenschutzes	✓	Anforderungen zur IT-Sicherheit
✓	Datenschutzrechtliche Prinzipien	✓	Mehrseitige IT-Sicherheit
✓	Technischer Datenschutz	✓	Risiko-Management
✓	Schwerpunkt: Aktuelles	➔	Konzeption von IT-Sicherheit

Konzeption von IT-Sicherheit:

- Ereignisse vs. Vorfälle
- Sicherheitsvorfallmanagement
- Schwachstellenmanagement
- Authentifizierung & Rechtevergabe
- Interessensausgleich
- Erstellung sicherer IT-Systeme
  - ° V-Modell XT
  - ° Konstruktionsprinzipien

Sicherheitsereignisse

- Verlust ausgegebener Zutrittsmittel
- Unzutreffende Klassifizierung des Schutzbedarfs von Informationen
- Detektion unbefugter Zugangs- (Einbruch) oder Zugriffsversuche
- Kenntnis über eine bestehende Schwachstelle bzw. Sicherheitslücke
- Kenntnis über das Vorliegen einer nicht entfernten Malware
- Kenntnis über abrufbare Zugangs- bzw. Zugriffscredentials (z.B. infolge eines unwirksamen Zugangsschutzes von Passwortsafes)
- Kenntnis über ungeschützt abrufbare vertrauliche Daten (z.B. infolge der Migration oder des Transfers von Daten auf einem gesharten Laufwerk)
- Kenntnis über nicht benötigte Schnittstellen zwischen Systemen und Komponenten oder auch für Unbefugte erreichbaren Ports
- Kenntnis über eingesetzte, veraltete Kryptoverfahren
- Kenntnis über eingegangene Versuche von Phishing, Social Engineering oder gezielten Angriffen (spear phishing bzw. targetted attacks)
- Kenntnis über ungewöhnliches Systemverhalten (technisches Versagen oder aufgrund unerwartet angezeigter Systemmeldungen)
- Kenntnis über unerwartet abgelegte Dateien in Speicherbereichen, auf denen Nutzer üblicherweise keine Daten ablegen

Sicherheitsvorfälle

- Ausfall eines IT-Systems, einer Anwendung oder einer IT-Komponente
  → **Verletzung der Verfügbarkeit**
- Störung der Funktionsweise eines IT-Systems, einer Anwendung oder einer IT-Komponente
  → **Verletzung der Verfügbarkeit und/oder Integrität**
- Kenntnis über den Eintritt eines unbefugten Datenabflusses (z.B. E-Mail Datentransfer aber auch Verlust von Papierunterlagen)
  → **Verletzung der Vertraulichkeit**
- Kenntnis über unerwartete Aktionen durch Nutzer oder Systemen gemäß vorliegenden Eventlogs
  → **Verletzung der Vertraulichkeit, Verfügbarkeit und/oder Integrität**
- Kenntnis über Nichteinhaltung vorgeschriebener Arbeitsanweisungen hinsichtlich des Umgangs mit Informationen
  → **Verletzung der Compliance**

Unterschied:

Ein **Sicherheitsvorfall** liegt nur dann vor, wenn ein **Sicherheitsziel verletzt** wurde

Bei **Sicherheitsereignis** nur entsprechende **Verletzung möglich**

Sicherheitsvorfallmanagement (1)

Phasen des Sicherheitsvorfallmanagement nach ISO/IEC 27035-1:

- **Plan and Prepare**
  - Richtlinie zum Umgang mit Informationssicherheitsvorfällen
  - Einsatz eines CSIRT
  - Awareness (Training und Unterweisungen; Berücksichtigung von Vorfallserfahrungen)
- **Detection and Reporting**
  - Aufzeichnung anormaler, verdächtiger oder maliziöser Aktivitäten
  - Informationssammlung über Sicherheitsereignisse (Vorfälle, bekannt gewordene Sicherheitslücken, Infos zur aktuellen Gefährdungslage)
  - Bericht relevanter Sicherheitsereignisse (incl. Erstmeldung meldepflichtiger Vorfälle)
- **Assessment and Decision**
  - Analyse der relevanten Sicherheitsereignisse
  - Entscheidung über Umgang mit Sicherheitsereignissen
- **Responses**
  - Welche relevanten Sicherheitsereignisse werden beherrscht?
  - Wiederherstellung der Arbeitsfähigkeit nach Behebung des Vorfalls
  - Ggf. Krisenkommunikation und Erfüllung vollständiger Meldepflichten
- **Lessons learnt**
  - Verbesserung Informationssicherheit, IT-Risikoanalyse & Vorfallmanagement

Sicherheitsvorfallmanagement (2)

Aufgaben Computer Security Incident Response Team (CSIRT):

- Analyse & Bewertung von **Sicherheitsvorfällen**
  - Einstufung zur Kritikalität von Sicherheitsvorfällen (je kritischer, desto rascher muss Sicherheitsvorfall wirksam behandelt werden)
  - Kategorisierung von Sicherheitsvorfällen (Angriff von außen/innen, Malwarebefall, DoS-Attacke, Rechtemissbrauch befugter User, …)
- Behandlung von Sicherheitsvorfällen (inkl. Ausführung von Notfall- bzw. Ausnahmeregeln zur Beseitigung von Sicherheitsvorfällen und Rückführung zum Normalbetrieb)
- Minimierung der Wirkung von Sicherheitsvorfällen
- Meldung über Sicherheitsvorfälle an zuständige Stellen (z.B. wg. Datenpanne oder Eskalation)
- Nachbereitung zu Erkenntnissen aus Sicherheitsvorfällen

Zudem hilfreich: **RFC 2350**, https://datatracker.ietf.org/doc/html/rfc2350

Sicherheitsvorfallmanagement (3)

Aufgaben Computer Security Incident Response Team (CSIRT) nach RFC 2350:

- Veröffentlichung von CSIRT Policies und generellen Prozeduren:
  - Welche Aufgaben führt das CSIRT aus und welche nicht?
  - Wie erreiche das CSIRT (Kommunikationswege) und kommuniziere ausreichend sicher mit diesem?
  - Wie erkenne ich, dass eine Information vom CSIRT stammt, d.h. authentisch ist?
  - Welche Befugnisse hat das CSIRT?
  - Wie habe ich das CSIRT bei Bedarf zu unterstützen bzw. mit diesem zu interagieren (inkl. Handoff-Prozeduren)?
- Interpretation eingehender Meldungen über Vorfälle
- Priorisierung bei der Behandlung von Vorfällen
- Ursachenanalyse von Vorfällen
- Lösung / Beseitigung von Vorfällen
- Unterstützung bei den Wiederherstellungsarbeiten nach Vorfällen
- Zusammenstellung von Erkenntnissen aus der Behandlung von Vorfällen
- Unterstützung bei der proaktiven Vermeidung vergleichbarer Vorfälle

Schwachstellenmanagement (1)

- Je kritischer ein Supporting Asset ist, desto intensiver ist zu prüfen, ob es ausnutzbare Schwachstellen gibt, die möglichst zeitnah geschlossen werden sollten
- Informationsquellen hierfür bieten Hersteller bzw. Distributoren selbst (im Zuge Cyberresilienz-VO der EU explizit sogar mit Pflicht zur zeitnahen Information vorgeschrieben), als auch diverse einschlägige Plattformen wie https://www.cvedetails.com/
- Regelmäßig ist der jeweilige Patchstand zu prüfen, z.B. durch monatliche (automatisierte) Schwachstellenscans
- Für besonders wichtige Supporting Assets empfiehlt sich zudem mind. einmal pro Jahr die Durchführung eines Penetrationstests durch einen fachkundigen Tester (gezielter & intelligent modellierter Versuch, Zielsystem unter Ausnutzung unzureichend implementierter Schutzvorkehrungen zu erreichen, indem z.B. unzureichend geschützte Daten abgerufen, Authentisierungsmechanismen umgangen, relevante Datensätze verändert oder IT-Services gestört werden)
- Angriffsvektoren zudem modellierbar über sog. Red Teaming (bedrohungsorientierte Penetrationstests)

Schwachstellenmanagement (2)

- Zur Schließung oder zumindest Abmilderung bestehender Schwachstellen sind Maßnahmen zum Patch Management erforderlich
  → erfolgt i.d.R. im Rahmen des Change Managements
  → ggf. Sonderregeln für sicherheitskritische Hot Fixes (ggf. mit Rückführung in Regelbetrieb mit zwischenzeitlich verstärkter Beobachtung der Lage)
  → wichtig ist abgesicherter Fail-Safe-Mechanismus für den Fall, dass aus Patching Fehler resultieren (oder bereits durch Angreifer kompromittierte Systeme vorliegen ggf. auch aus Datensicherung heraus)
- System-Härtung führt dazu, dass weniger Angriffsvektoren bestehen (z.B. unter Ausnutzung von mit den Herstellern abgestimmter Hardening Guidelines)
- Bei Legacy Systemen, für die es keine aktuellen Sicherheitspatches mehr gibt, bieten sich als kompensatorische Maßnahmen z.B. an:
  - Separierung des Legacy-Systems in eine eigene Netzwerkzone
  - Aktivierung zusätzlicher Detektionsmechanismen zur Erkennung unbefugter Angriffsversuche
  - tägliche Kontrolle wichtiger Systemfunktionalitäten
  - häufigere Datensicherungen (vorzugsweise im migrationsfähigen Format)
  - besonders angelegte Zugangsbefugnisse

Weitere hilfreiche Quellen

- Vulnerability & Patch Management:
  https://www.heise.de/
  https://www.cisecurity.org/cis-benchmarks
  https://www.vmware.com/security/hardening-guides.html
- Secure Coding Guidelines:
  https://owasp.org/www-project-proactive-controls/
  https://wiki.sei.cmu.edu/confluence/display/seccode/Top+10+Secure+Coding+Practices
- Maßnahmenplanung:
  https://www.bsi.bund.de/DE/Themen/Unternehmen-und-Organisationen/Standards-und-Zertifizierung/IT-Grundschutz/IT-Grundschutz-Kompendium/it-grundschutz-kompendium_node.html

Zur Authentifizierung

- Sicherung der Benutzeridentifikation (gemäß Authentisierung) anhand
  - ° Wissen → z.B. Password
  - ° Besitz → z.B. Chipkarte / Token
  - ° Merkmal → z.B. Biometrie
  - ° Zwei-Faktor-Authentifizierung (anhand zweier der drei aufgeführten Mechanismen); im Zuge NIS2-Richtlinie explizit eingefordert (vor allem für Admins empfehlenswert)
- nur Feststellung, ob Benutzer berechtigt ist, nicht ob dessen (vorgegebene) Identität tatsächlich korrekt ist!

→ Zugangs-/Zugriffskontrolle mittels Rechteprüfung

Zur Rechtevergabe

- Subjekt (Benutzer & Prozesse)
- Objekt (Dateien & Datenträger)
- Zugriffsart (lesen, schreiben, ausführen, löschen) jeweils

→ **Access Control List**: wer darf auf gegebenes Objekt zugreifen?

→ **Capability List**: auf welche Objekte darf ein gegebener Benutzer zugreifen?

→ **Grundsatz Zugriff: need-to-know** (nur benötigte Rechte!)

→ Pflege erfordert z.T. hohen Aufwand (darum: Benutzerrollen → Role-Based Access Control; RBAC)

→ beachtenswert: spezifischere Regeln vor allgemeineren Regeln!

→ **Grundsatz Systemzugang**: **need-to-use** (welche Systeme zwingend benötigt?)

Interessenausgleich zwischen Betroffene & Systemnutzer

Beispiele für abweichende Interessen:

- Systemnutzer möchten möglichst detaillierte Daten angezeigt bekommen, um sicher gehen zu können, dass sie keine fehlerhaften Daten eingeben bzw. bearbeiten. Betroffene möchten, dass verantwortliche Stellen nur so viel Daten über sich haben, wie unbedingt nötig. Der **Ausgleich** erfolgt daher **durch** das **Berechtigungskonzept**, in dem festgelegt ist, welcher Nutzer welche Daten (zu welchem Zweck) einsehen und bearbeiten darf.
- Systemnutzer wünschen eine umfassende Datensicherung, damit im Falle eines ungewollten Datenverlustes oder bei einem zeitlich späteren Vorgang noch die Historie berücksichtigt werden kann. Betroffene möchten, dass ihre Daten nur für die vorgeschriebene Dauer abrufbar sind. Der **Ausgleich** erfolgt daher **über** die Regelungen zur **Sperrung** (= „Einschränkung“ nach EU-DSGVO) von Daten.

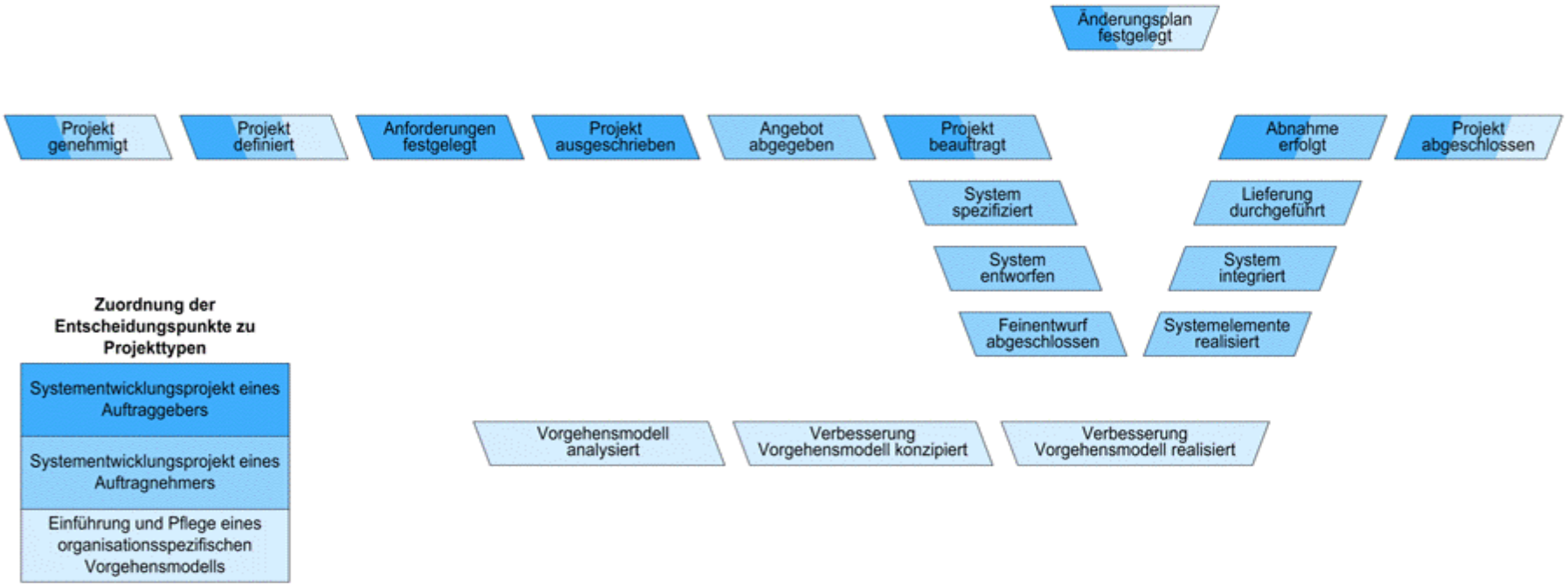
Erstellung sicherer IT-Systeme

- **Software-Erstellung**
  → V-Modell XT
  → Sichere Softwareentwicklung *(→ Übung)*

- **Konstruktionsprinzipien**
  → allgemeine Prinzipien
  → Prinzipien für Sicherheitsprozesse

- **Systemsicherheit**
  → Serversicherheit & Clientsicherheit *(→ Übung)*

Überblick zum V-Modell XT



Hinweise zum V-Modell XT (1)

- für jedes systemsicherheitskritisch eingestuftes Systemelement ist eine **Sicherheitsanalyse** durchzuführen
- Verfahrens- bzw. Betriebssicherheit sowie Zuverlässigkeit, Fehlertoleranz und Korrektheit als Maßstäbe für **Safety**
- Gewährleistung von Verfügbarkeit, Integrität, Vertraulichkeit und Verbindlichkeit (= beweisbare zugesicherte Eigenschaften) beim Einsatz der IT als Maßstäbe für **Security**

Hinweise zum V-Modell XT (2)

- Systemsicherheitsanalyse mittels
  - **Blackbox-Test** durch Auftraggeber
    → Stellen sich erwartete Ergebnisse ein?
  - **Whitebox-Test** durch Auftragnehmer
    → Werden alle Konstruktionselemente durchlaufen?
- **jeder Konstruktionsphase** (Anforderungsfestlegung, Spezifikation, Entwurf, Implementation) **ist eine Kontrollphase zugeordnet**, unter Beachtung von:
- **Verifikation**: System wurde zu jedem Zeitpunkt nach den „Regeln der Kunst“ erstellt & weist vordefinierte Eigenschaften auf
  → Vollständigkeit, Widerspruchsfreiheit, Durchführbarkeit, Testbarkeit
- **Validierung**: System entspricht den vom Nutzer gewünschten Kriterien & den geltenden Anforderungen
  → Adäquatheit, Benutzbarkeit, Funktionsverhalten im Fehlerfalle

Konstruktion sicherer IT-Systeme (1)

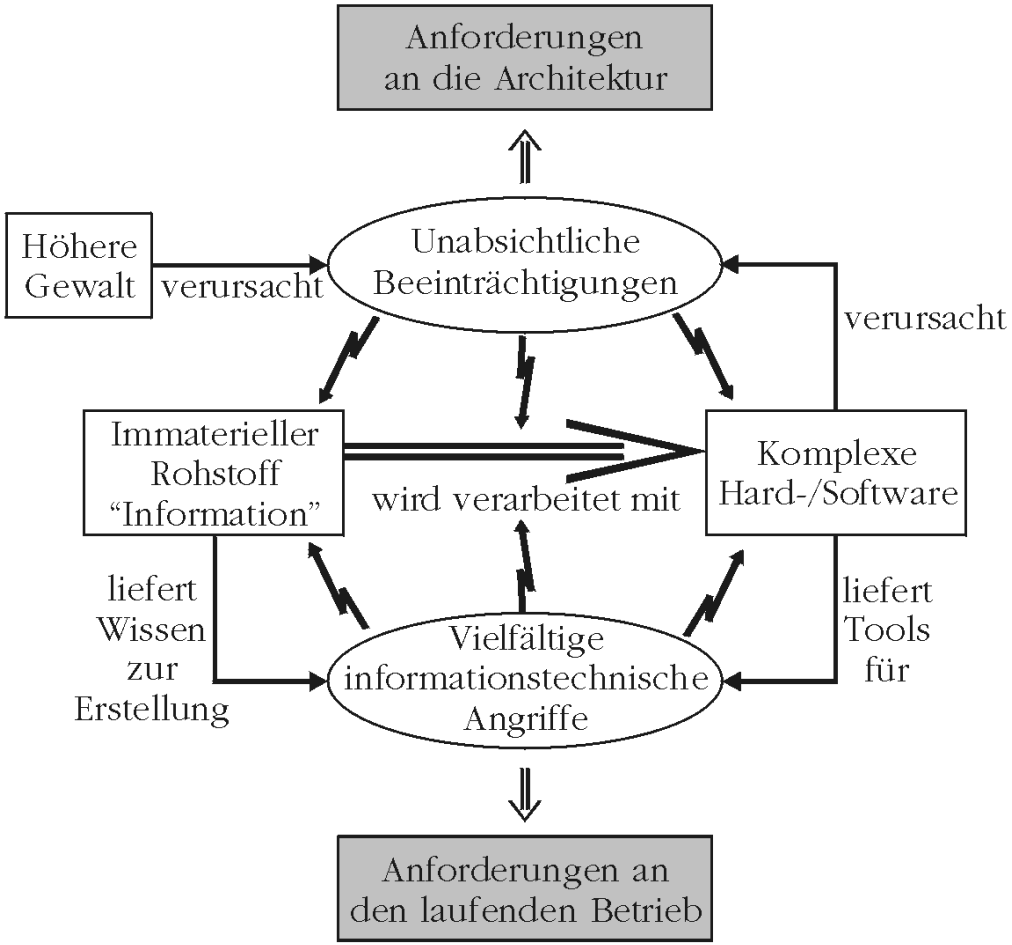
Allgemeine Prinzipien (nach Saltzer und Schroeder, 1975):

- **Prinzip einfacher Sicherheitsmechanismen**: wirksame, aber möglichst einfache Konstruktion
- **Erlaubnisprinzip**: Zugriff muss ausdrücklich erlaubt werden
- **Prinzip vollständiger Rechteprüfung**: Rechteprüfung bei allen Aktionen
- **Prinzip des offenen Entwurfs**: angewandte Verfahren und Mechanismen sind offenzulegen → Kerckhoffs‘ Prinzip
- **Prinzip der differenzierten Rechtevergabe**: keine Rechte aufgrund nur einer einzigen Bedingung
- **Prinzip minimaler Rechte**: Vergabe nur der Rechte, die zur Aufgabenstellung unbedingt benötigt werden
- **Prinzip durchgreifender Zugriffskontrollen**: Vermeidung verdeckter Kanäle
- **Prinzip der Benuterzakzeptanz**: einfache Anwendbarkeit

Konstruktion sicherer IT-Systeme (2)

Prinzipien für Sicherheitsprozesse (nach Schneier, 2000):

- **Risiko durch Aufteilung verringern**: nur benötigtes Privileg vergeben
- **das schwächste Glied sichern**: Angriffsbaum betrachten
- **Choke-Points verwenden**: Benutzer durch engen Kanal zwingen
- **gestaffelte Abwehr**: hintereinander geschaltete Barrieren aufbauen
- **Folgeschäden begrenzen**: Rückkehr zum sicheren Normalzustand bei Systemausfällen
- **Überraschungseffekt nutzen**: innere Einstellungen des IT-Systems verdeckt halten
- **Einfachheit**: lieber wenige, dafür effektive Schutzmechanismen
- **Einbeziehung der Benutzer**: Insider so weit & oft wie möglich beteiligen
- **Gewährleistung**: Produktverhalten gemäß Zusicherung
- **Alles in Frage stellen**: Nicht mal sich selbst vertrauen



Umsetzung der Konstruk-tionsprin-zipien