Zusammenfassung

**Lernziele**

* Sie beherrschen Methoden zur Verbesserung der Informationssicherheit
* Sie kennen aktuelle Probleme und Abwehrmassnahmen in den Bereichen Web und Mobile Application Security
* Sie können die Grundprinzipien zur Verbesserung der Informationssicherheit anhand aktueller Anwendungbeispiele erklären
* Sie kennen das Vorgehen bei Sicherheitsanalysen

**Unterlagen / Bücher**

* .......

**Lerninhalte**

* Methoden und Werkzeuge zur Verbesserung der Informationssicherheit
* Standards, Best Practices
* Security Guidelines
* Gesetze (Datenschutz)
* Open (Web) Application Security Project (OWASP)
* Client Side Security
* Sandboxing
* Security Zones
* Web Application Security
  + Konzepte (Rollen und Rechte, Session Management,…)
  + Web Application Vulnerabilities (XSS, Broken Authentication, Injections, …)
* Mobile Application Security
  + Mobile Application Vulnerabilities
  + Platform Security
  + Code Distribution
* Security Reviews

Der Inhalt wird nach Bedarf angepasst, um aktuelle Sicherheitsprobleme behandeln zu können. Lern- und Unterrichtsmethoden: Vorlesung mit "Chats" und Kurztests, Übungen am Computer (Security Lab). Unterrichtsunterlagen: Folien mit Zusatztexten (Kommentar); Tutorials/White Papers; Online-Infos

Übersicht des Modules

[Information Security Management 3](#_Toc477772625)

[Information Security Requirements 4](#_Toc477772626)

[Schadenindikatoren 4](#_Toc477772627)

[Erfüllung der Anforderungen (Compliance Laws & Regulations) 4](#_Toc477772628)

[Threats 6](#_Toc477772629)

[Attacker und ihre Motivation 6](#_Toc477772630)

[National Security Agency 7](#_Toc477772631)

[Symantec Internet Security Thread Report 2016 8](#_Toc477772632)

[Software Security – Building Security 8](#_Toc477772633)

[NIST Statistiken zu diesem Thema 8](#_Toc477772634)

[CWE/SANS Top 25 der meist gefährlichsten Software Fehler 9](#_Toc477772635)

[Buffer Overflow im StrongSwan – Projekt 9](#_Toc477772636)

[The Trinity of Trouble 9](#_Toc477772637)

[Die Komplexität von Windows 10](#_Toc477772638)

[Fehler + Mängel = Defekte 10](#_Toc477772639)

[Bug oder Mangel(Flaw)? – Finden Sie die Defekte 11](#_Toc477772640)

[Bugfreie Software 11](#_Toc477772641)

[Die drei Pfeiler der Software Security 11](#_Toc477772642)

[Pfeiler 1 – Angewandtes Risiko Management 11](#_Toc477772643)

[Pfeiler 2 – Software Security Best Practices 12](#_Toc477772644)

[Pfeiler 3 – Wissen zu Software Security 13](#_Toc477772645)

[Code Review Tools 14](#_Toc477772646)

[Coding Rule Sets 14](#_Toc477772647)

[Architectural Risk Analysis 15](#_Toc477772648)

[Microsoft Security Development Lifecycle 16](#_Toc477772649)

[Die 25 schlechtesten Produkte aller Zeiten 16](#_Toc477772650)

[Trustworthy Computing Initiative in 2002 16](#_Toc477772651)

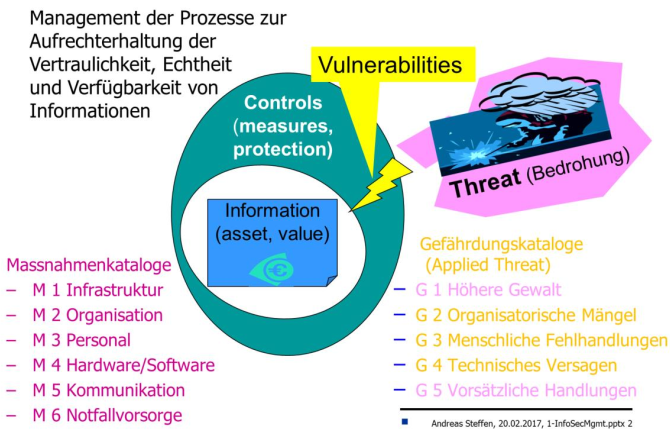
[Die drei Hauptkonzepte des Microsoft SDL 16](#_Toc477772652)

[SDL Optimization Model 17](#_Toc477772653)

[Das Modell 17](#_Toc477772654)

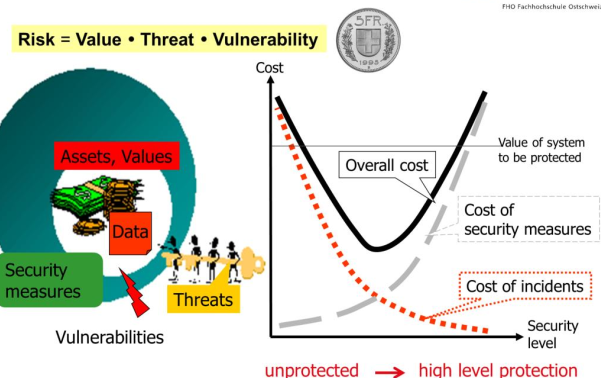
[Vorlesung Woche 4 18](#_Toc477772655)

# Information Security Management

**Information Security Management (BSI)**  
Ein Informationssicherheitsvorfall (Event) ist ein unerwünschtes Ereignis, welches die Informationssicherheit beeinträchtigt bzw. zum Verlust von einem oder von mehreren Werten der Information führt. Ob bzw. wie häufig ein solcher Informationssicherheitsvorfall auftritt hängt einerseits von der Gefährdungslage (Bedrohungspotenzial, Threat) und andererseits vom Schutz der Information (Protection, Measures, Controls) und von der Verletzlichkeit (Vulnerabilities) der Schutzmassnahmen ab.

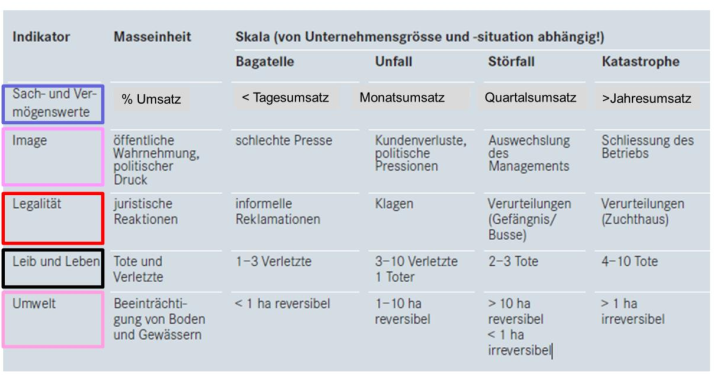
Im Glossar des Bundesamts für Sicherheit in der Informationstechnik (BSI) findet man folgende

**Definitionen**:  
Eine ***Bedrohung*** (englisch "threat") ist ganz allgemein ein Umstand, durch den ein Schaden entstehen kann. Der Schaden bezieht sich dabei auf einen konkreten Wert wie Vermögen, Wissen, Gegenstände oder Gesundheit. Übertragen in die Welt der Informationstechnik ist eine Bedrohung ein Umstand, der die Verfügbarkeit, Integrität oder Vertraulichkeit von Informationen beeinträchtigen kann. Beispiele für Bedrohungen sind höhere Gewalt oder vorsätzliche Handlungen. Trifft eine Bedrohung auf eine Schwachstelle (z.B. technische oder organisatorische Mängel), so ergibt sich eine Gefährdung.  
Eine ***Gefährdung*** (englisch "applied threat") ist eine Bedrohung, die konkret über eine Schwachstelle auf ein Objekt einwirkt. Eine Bedrohung wird somit aufgrund einer vorhandenen Schwachstelle zur Gefährdung für ein Objekt.   
Computer-Viren sind beispielsweise eine Bedrohung oder eine Gefährdung für Anwender, die im Internet surfen. Alle Anwender sind prinzipiell durch Computer-Viren im Internet bedroht. Der Anwender, der eine virenverseuchte Datei herunterlädt, wird von dem Computer-Virus gefährdet, wenn sein Computer anfällig für diesen Computer-Viren-Typ ist. Für Anwender mit einem wirksamen Schutzprogramm, einer Konfiguration, die das Funktionieren des Computer-Virus verhindert, oder einem Betriebssystem, das den Virencode nicht ausführen kann, bedeutet das geladene Schadprogramm hingegen keine Gefährdung.

Information Security – What Risk to Take?  
Es geht darum einen Mittelweg zwischen Sicherheitskosten und Schadenkosten zu finden.

## Information Security Requirements

### Schadenindikatoren

Die Gewichtungen werden in einem ersten Schritt auf einer reinen Expertenschätzung oder einem Self-Assesment beruhen. Sie sollten aber zunehmend mit Erfahrungswerten validiert werden. Dazu ist der Aufbau von Schadensfalldatenbanken nützlich. Die Bewertung der Sach- und Vermögenswerte (Gebäude, Geräte, Wertschriften, Geld) würde man wohl besser in der Jahresumsätzen rechnen oder man hätte beim Schadenpotenzial Bezeichnungen wie Konkurs, vorübergehend rote Zahlen, merklicher Verlust, kaum spürbar. Die Kategorien Image (Reputation, Vertrauen der Kunden) könnte beim Schadenpotenzial anstelle der oben angegebenen Auswirkungsschäden auch Bezeichnungen haben wie breite, anhaltende Diskussion in allen Medien bis einmaliger Bericht in der Lokalzeitung.

### Erfüllung der Anforderungen (Compliance Laws & Regulations)

Regulationen sind die Treiber für die Informationssicherheit. Die Regulationen unterschieden sich je nach Bereich.

* Bearbeiter von Personendaten
  + Datenschutzgesetz, HIPAA
* Finanzdienstleister
  + Bankengesetze, EU Directive on Payment Services (PSD), PCI
* Telecom/ICT-Anbieter
  + Fernmeldegesetz, Lawful Interception

#### Bundesgesetz über den Datenschutz

**Art 8 – Allgemeine Massnahmen**  
Wer als Privatperson Personendaten bearbeitet oder ein Datenkommunikationsnetz zur Verfügung stellt, sorgt für die Vertraulichkeit, die Verfügbarkeit und die Richtigkeit der Daten, um einen angemessenen Datenschutz zu gewährleisten. Insbesondere schützt er die Systeme gegen folgende Risiken:

* unbefugte oder zufällige Vernichtung;
* zufälligen Verlust;
* technische Fehler;
* Fälschung, Diebstahl oder widerrechtliche Verwendung;
* unbefugtes Ändern, Kopieren, Zugreifen oder andere unbefugte Bearbeitungen.

#### Strafgesetzbuch – Strafbare Handlungen gegen den Geheim- oder Privatbereich

**Art. 179septies – Missbrauch einer Fernmeldeanlage**  
Wer aus Bosheit oder Mutwillen eine Fernmeldeanlage zur Beunruhigung oder Belästigung missbraucht, wird, auf Antrag, mit Haft oder Busse bestraft.

**Art. 179novies – Unbefugtes Beschaffen von Personendaten**  
Wer unbefugt besonders schützenswerte Personendaten oder Persönlichkeitsprofile, die nicht frei zugänglich sind, aus einer Datensammlung beschafft, wird auf Antrag mit Gefängnis oder mit Busse bestraft.

#### Health Insureance Portability and Accountability Act (HIPAA)

HIPAA wurde 1996 durch den US Kongress eingeführt. Der erste Titel regelt den Health Care Access, Portability and Renewability. (Vorreiter der Obamacare). Im zweiten Titel wird geht es um Privacy Reules, Transactions and Code Sets Rules, Security Rules (Sicherheit für die elektronischen Patientendaten), Unique Identifier Rules und Enforcement Rules.

#### Sorgfaltspflicht der Banken

Die Schweizerische Bankiervereinigung definiert das Bankgeheimnis folgendermassen: «Der Bankkunde hat ein Recht auf Schutz seiner ökonomischen Privatsphäre, die Bank hat somit die Pflicht, über alle Tatsachen, die ihre Kunden betreffen, Verschwiegenheit zu wahren.» Das Bankgeheimnis ist ein Berufsgeheimnis, dem nicht nur die Angestellten einer Bank unterworfen sind, sondern auch Organe, Beauftragte oder Liquidatoren einer Bank, Untersuchungs- oder Sanierungsbeauftragte der Bankenkommission sowie Organe oder Angestellte einer anerkannten Revisionsstelle. Das Bankgeheimnis ist im Bundesgesetz über die Banken und Sparkassen (Bankengesetz, BankG) in Artikel 47 verankert.

**Themen**

* Identifizierung des Vertragspartners und Feststellung des wirtschaftlich Berechtigten
* Verbot der aktiven Beihilfe zur Kapitalflucht
* Verbot der aktiven Beihilfe zu Steuerhinterziehung oder ähnlichen Handlungen

#### Payment Card Industry Data Security Standard (PCI)

* Installation und Pflege einer Firewall zum Schutz der Daten
* Ändern von Kennwörtern und anderen Sicherheitseinstellungen nach der Werksauslieferung
* Schutz der gespeicherten Daten von Kreditkarteninhabern
* Verschlüsselte Übertragung sensibler Daten von Kreditkarteninhabern in öffentlichen Rechnernetzen
* Einsatz und regelmäßiges Update von Virenschutzprogrammen
* Entwicklung und Pflege sicherer Systeme und Anwendungen
* Einschränken von Datenzugriffen auf das Notwendige
* Zuteilen einer eindeutigen Benutzerkennung für jede Person mit Rechnerzugang
* Beschränkung des physikalischen Zugriffs auf Daten von Kreditkarteninhabern
* Protokollieren und Prüfen aller Zugriffe auf Daten von Kreditkarteninhabern
* Regelmäßige Prüfungen aller Sicherheitssysteme und -prozesse
* Einführen und Einhalten von Richtlinien in Bezug auf Informationssicherheit

Der Payment Card Industry Data Security Standard (PCI) st ein Regelwerk im Zahlungsverkehr, das sich auf die Abwicklung von Kreditkartentransaktionen bezieht und von allen wichtigen Kreditkartenorganisationen unterstützt wird. Handelsunternehmen und Dienstleister, die Kreditkarten-Transaktionen speichern, übermitteln, oder abwickeln, müssen die Regelungen erfüllen. Halten sie sich nicht daran, können Strafgebühren verhängt, Einschränkungen ausgesprochen, oder ihnen letztlich die Akzeptanz von Kreditkarten untersagt werden. Die Einhaltung der Regeln wird üblicherweise in Abhängigkeit vom Umsatzvolumen des Unternehmens überprüft:

* Händler oder Dienstleister, die mehr als 6 Mio. Kreditkartentransaktionen pro Jahr abwickeln, bereits einem Angriff erlagen, müssen ihr Rechnernetz vierteljährlich mittels eines externen Sicherheitsscans durch einen von Mastercard zugelassenen Scanvendor prüfen lassen und zusätzlich einmal im Jahr eine Begehung vor Ort durch ein unabhängiges, von VISA zugelassenes Unternehmen oder eines eigens dazu ernannten Sicherheitsbeauftragten durchführen lassen.
* Händler, die zwischen 20.000 und 6 Mio. Kreditkartentransaktionen pro Jahr abwickeln, müssen ihr Rechnernetz ebenfalls mittels eines externen Sicherheitsscans durch einen von Mastercard zugelassenen Approved Scanning Vendor vierteljährlich prüfen lassen und zusätzlich einmal im Jahr einen PCIFragebogen (Self-Assessment Questionnaire, SAQ)) ausfüllen.
* E-Commerce Händler, die weniger als 1 Mio. Kreditkartentransaktionen pro Jahr abwickeln, müssen seit dem 1. Oktober 2009 einen PCI DSS-zertifizierten Service Provider mit der Abwicklung der kompletten Kreditkartentransaktionen beauftragen oder ihrem Acquirer die eigene PCI DSS-Zertifzierung durch Ausfüllen des PCI Self-Assessment Questionnaire und ggf. Durchführung eines vierteljährlichen Sicherheitsscan durch einen vom PCI Security Standards Council zugelassenen Approved Scanning Vendor nachweisen

#### Sarbanes-Oxley Act (SOX), 2002

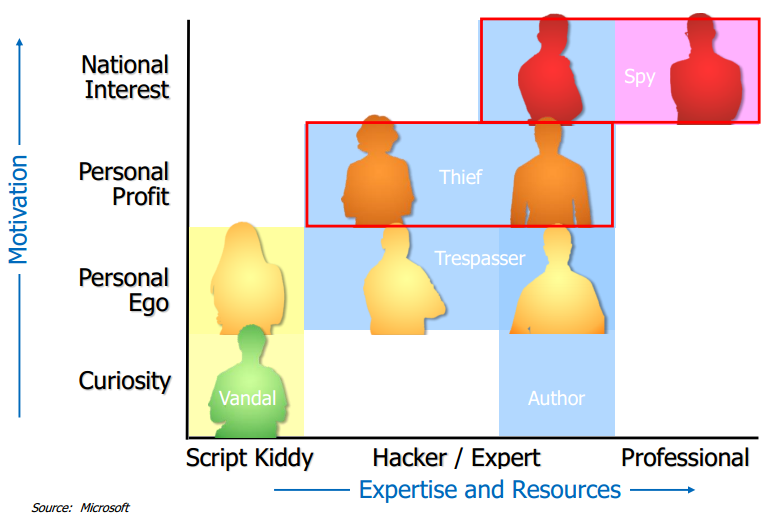
Das Sarbanes-Oxley Act of 2002 (auch SOX, SarbOx oder SOA) ist ein USBundesgesetz, das als Reaktion auf Bilanzskandale von Unternehmen wie Enron oder Worldcom die Verlässlichkeit der Berichterstattung von Unternehmen, die den öffentlichen Kapitalmarkt der USA in Anspruch nehmen, verbessern soll.

**Ziel**  
Vertrauten der Anlegen in die Richtigkeit und Verlässlichkeit der veröffentlichten Finanzdaten wiederherstellen.

Es fordert verschärfte interne Kontrollsysteme und führt zu höheren Anforderungen an die Corporate Governance. Der Jahresbericht muss die Berurteilung der Wirksamkeit des internen Kontrollsystems enthalten.

## Threats

### Attacker und ihre Motivation

Die "**Vandal**" ist die Person, die, zum Beispiel: Hacks in einem schlecht Website geschützt und verunstaltet den Inhalt.

"**Trespassers**" sind fähiger als Vandalen und sie sind motiviert durch Ego und ein Gefühl von persönlichem Ruhm. Ihre Absichten sind relativ gutartig, aber sie können erhebliche Probleme verursachen. Die Hacker, die viele der Würmer und Viren, die Nachrichten machen in der Regel fallen in diese Kategorie zu schaffen. Weil ihre Angriffe riesige Mengen an Verkehr und manchmal Denial-of-Service-Angriffe verursachen, können ihre Aktionen zu schweren Sachschäden an Computerbenutzern, Unternehmen und anderen Organisationen führen.

Der "**Autor**" ist der hoch-fähige Hacker, der die Werkzeuge und das Fachwissen hat, um einen Patch umzukehren und einen Exploit-Code zu schreiben oder Schwachstellen in Sicherheitssoftware, Hardware oder Prozessen zu finden. Autoren sind in der Regel von Ego, Ideologie und / oder persönlichem Ruhm motiviert. Autoren erstellen die Bausteine für kriminelle Hacker. Die Werkzeuge und der Code, den sie produzieren, sind in der Regel leicht verfügbar für die weniger anspruchsvolle, was bedeutet, dass die Vandalen und die Skript-Kiddies in der Lage, viel mehr Ärger mit weniger Arbeit verursachen.

Die "**Diebe**" sind Menschen, die in ihr für das Geld sind, und sie beinhalten organisierte Verbrechensyndikate aus der ganzen Welt. Diebe sind aktiv und effektiv beim Hacken in Unternehmens- und Unternehmenssysteme, manchmal um Informationen zu stehlen, die monetären Wert haben (wie Kreditkartennummern), manchmal um Bargeld in ihre Konten umzuleiten und manchmal Zahlungen zu erpressen. Die Diebe profitieren von den Bemühungen des Autors.

Die "**Spione**", die im Auftrag der Regierungen arbeiten, sind hochqualifiziert und haben praktisch unbegrenzte Ressourcen. Und die größten Aufwendungen für den Schutz - Aufbau von starken Verteidigungen - werden von den Spies gemacht.

### National Security Agency

Etwa 30’000 Mitarbeiter mit einem Budget von etwa 8 Milliarden USD Dollar.

#### Patriot Act 2001

Wurde vom US Kongress am 25.10.2001 als Reaktion auf Terroranschläge vom 11.9.2001 verabschiedet. Bei Telefon- oder Internetüberwachung gibt es keine echte Kontrollinstanz mehr. Telefongesellschaften und Internetprovider müssen ihre Daten offenlegen. Hausdurchsuchungen sind ohne Wissen der betreffenden Person durchführbar. FBI darf Einsicht in finanzielle Daten von Bankkunden nehmen, ohne dass vorliegende Beweise für ein Verbrechen vorliegen. CIA (Auslandsgeheimdienst) - unterliegt im Gegensatz zum FBI keiner öffentlichen Kontrolle - erhält das Recht, auch im Inland zu ermitteln.

#### General Keith Alexander

NSA Director von 2005 bis 2013. Sein Motto war “Collect it all, tag it, store it… and whatever it is you want, you go searching for it.”.

#### PRISM Project PRISM ist ein seit 2005 existierendes und als Top Secret eingestuftes Programm zur Überwachung und Auswertung elektronischer Medien und elektronisch gespeicherter Daten. Es wird von der US-amerikanischen National Security Agency (NSA) geführt und gehört wie die anderen Teilprogramme „Mainway“, „Marina“ und „Nucleon“ zu dem gross angelegten Überwachungsprogramm „Stellar Wind“.

#### Quantum Project

#### Zum Arsenal der NSA gehört eine Methode, mit der sich nahezu jeder Rechner unbemerkt mit Spähsoftware bestücken lässt. Streng geheime Dokumente zeigen, wie das System genau funktioniert - das keineswegs nur gegen Terrorverdächtige zum Einsatz kommt.

#### XKeyScore Project

#### Der Zweck von XKeyscore ist es, den Analysten die Suche nach den Metadaten sowie den Inhalt von E-Mails und anderen Internetaktivitäten wie dem Browserverlauf zu ermöglichen, auch wenn es kein bekanntes E-Mail-Konto gibt (ein "Selektor" im NSA-Sprachgebrauch), das mit dem Individuum verknüpft ist.

Analysten können auch nach Namen, Telefonnummer, IP-Adresse, Schlüsselwörter, die Sprache, in der die Internet-Aktivität durchgeführt wurde, oder die Art des verwendeten Browsers suchen.

### Symantec Internet Security Thread Report 2016

Es gibt Inhalte über Mobile Devices, Internet of Things, Web Threats, Social Media (E-Mails), Targeted Attacks, Data Breaches, Cloud Infrastructures und Best Practice Guidelines frei.

# Software Security – Building Security

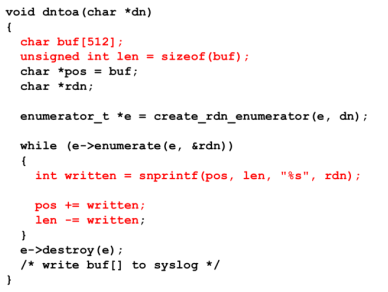
## NIST Statistiken zu diesem Thema

|  |  |
| --- | --- |
| Schwachstellen auf Basis von CVEs | Schwachstellen mit hoher Schwere |

## CWE/SANS Top 25 der meist gefährlichsten Software Fehler

|  |  |
| --- | --- |
| **Unsichere Interaktion zwischen Komponenten**  Diese Schwächen beziehen sich auf unsichere Wege, in denen Daten zwischen getrennten Komponenten, Modulen, Programmen, Prozessen, Threads oder Systemen gesendet und empfangen werden. | **Riskante Ressourcenverwaltung**Die Schwächen in dieser Kategorie beziehen sich auf die Art und Weise, in der Software die Erstellung, Nutzung, Übertragung oder Zerstörung wichtiger Systemressourcen nicht ordnungsgemäß verwaltet. |
| **Poröse Verteidigung** Die Schwächen in dieser Kategorie beziehen sich auf defensive Techniken, die oft missbraucht, missbraucht oder einfach nur ignoriert werden. | |

### Buffer Overflow im StrongSwan – Projekt

Die Anfälligkeit wurde durch Zufall erkannt, als unser Sponsor Astaro Internet Security mit einem japanischen Zertifikat einen Systemtest machte, das einen Distinguished Name (DN) enthielt, der größer als 512 Bytes war, was dazu führte, dass der strongSwan VPN-Daemon zum Absturz kam.

Problem war also die festgesetzte Grösse des Buffers.

### The Trinity of Trouble

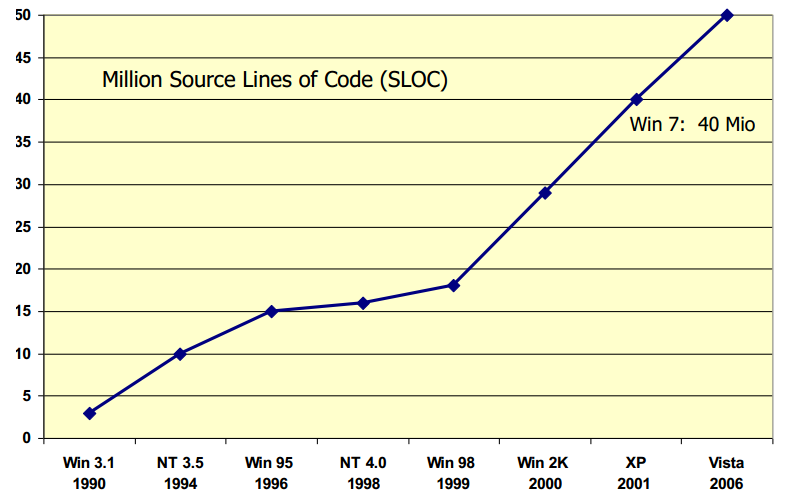
**Connectivity**Die wachsende Konnektivität von Computern über das Internet hat sowohl die Anzahl der Angriffsvektoren als auch die Leichtigkeit erhöht, mit der ein Angriff gemacht werden kann. Aufgrund von SOA werden Legacy-Anwendungen, die niemals für die Internetarbeit gedacht waren, nun als Dienste veröffentlicht.

**Extensibility**Die Plug-In-Architektur von Webbrowsern ermöglicht eine einfache Installation von Viewer-Erweiterungen für neue Dokumenttypen. Betriebssysteme unterstützen die Erweiterbarkeit durch dynamisch ladbare Gerätetreiber und Module. Anwendungen unterstützen die Erweiterbarkeit durch Scripting, Controls, Komponenten und Applets dank Java und dem .NET Framework.

**Complexity**Ungebremstes Wachstum in der Größe und Komplexität moderner Softwaresysteme. In der Praxis neigt die Defektrate dazu, als das Quadrat der Codegröße zu steigen.

### Die Komplexität von Windows

Aus der Grafik lässt sich ein eindeutiger Trend erkennen.



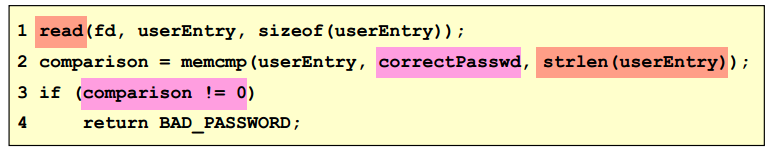
## Fehler + Mängel = Defekte

**Security Bug**  
Ein Sicherheitsfehler ist eine Implementierungsebene-Sicherheitsanfälligkeit, die mit Hilfe von modernen Code-Review-Tools leicht entdeckt und behoben werden kann. Dazu zählen Buffer Overflow, Race Conditions, nicht sichere Systemaufrufe.

**Security Flaw**  
Ein Sicherheitsfehler ist eine Sicherheitsanfälligkeit auf Design-Ebene, die von automatisierten Tools noch nicht erkannt werden kann, aber in der Regel eine manuelle Risikoanalyse der Software-Architektur erfordert, die von Experten durchgeführt wird. Dazu zählen Method Overriding, Error Handling und Type Safety Confusion.

**Security Defect**  
Beides, Bugs und Mängel sind Defekte, die jahrelang in der Software liegen können, um nur in Feldsystemen mit großen Konsequenzen zu begegnen. In der Praxis werden Software-Sicherheitsprobleme 50/50 zwischen Bugs und Fehlern geteilt.

### Bug oder Mangel(Flaw)? – Finden Sie die Defekte



**Bug in der Zeile 1**  
Der Rückgabewert von read() wird ignoriert. Dies ist immer eine schlechtes Zeichen, aber resultiert nicht direkt in einer Attacke.

**Bug in der Zeile 2**  
Die Funktion strlen() basiert auf read(), indem es einen Null Terminator am Ende des Strings setzt. Dies ist aber nicht garantiert.

**Mangel in der Zeile 2**  
Das System speichert die Passwörter im Klartext. So etwas kann am besten während einer Architektur Risiko Analyse gefunden werden.

**Mangel in der Zeile 3**  
Der Vergleich geht auch, wenn das Passwort eine Länge von 0 hat. Dieses Problem könnte durch gutes Testing verhindert werden.

### Bugfreie Software

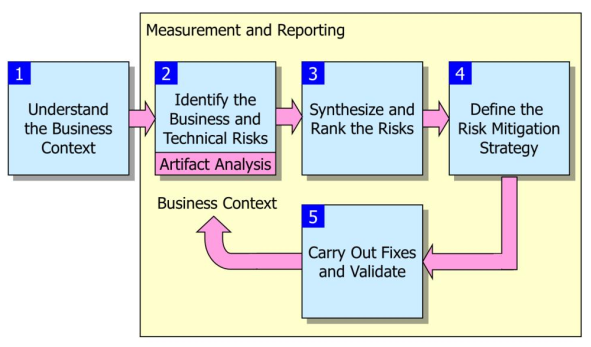


## Die drei Pfeiler der Software Security

### Pfeiler 1 – Angewandtes Risiko Management

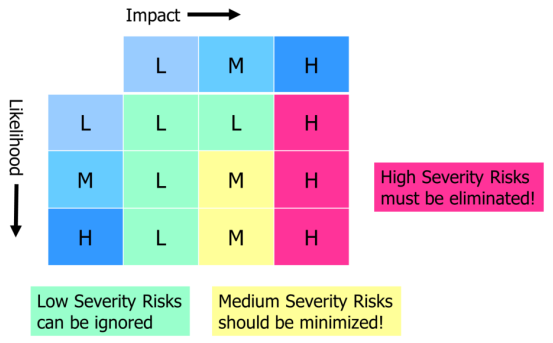
1. Verstehe den Business Context (Who cares?)
2. Identifiziere die technischen sowie die geschäftlichen Risiken
3. Synthetisieren und priorisieren Sie die Risiken und produzieren einen Rang
4. Definieren Sie die Strategie zur Risikobegrenzung
5. Führen Sie die erforderlichen Korrekturen aus und bestätigen Sie, dass sie korrekt sind

#### Risk Management Framework

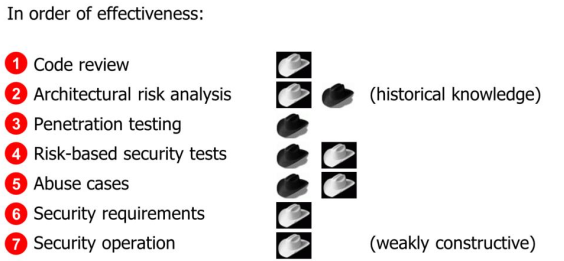


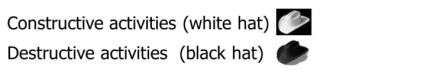
#### Beispiel KillerAppCo – Tabelle der Risiken

#### Matrix zur Risikobewertung



### Pfeiler 2 – Software Security Best Practices

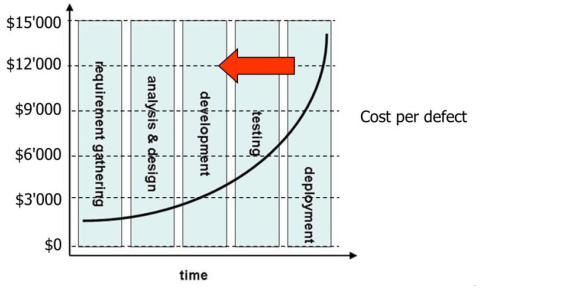




Alle Best Practices der Software-Sicherheit werden am besten von Personen angewendet, die nicht an der ursprünglichen Gestaltung und Implementierung des Systems beteiligt sind.

Viel Arbeit für externe Berater ;-)

#### Barry Boehm’s Cost of Change Law



### Pfeiler 3 – Wissen zu Software Security

**Prescriptive Wissen**

*Prinzipien*  
Hochrangige architektonische Prinzipien, z.B. Das Prinzip der geringsten Privilegien.

*Guidelines*  
Mid-Level-Richtlinien, z.B. Machen alle Java-Objekte und Klassen endgültig, es sei denn, es gibt einen Grund nicht zu. Sicherheitsmuster anwenden.

*Regeln*  
Taktische Code-Level-Regeln, z.B. Vermeiden Sie die Verwendung der Bibliotheksfunktion gets () in C

**Diagnostisches Wissen**

*Schwachstellen*  
Beschreibungen von Software-Schwachstellen, die in realen Systemen erlebt und gemeldet wurden (oft mit einer Vorspannung).

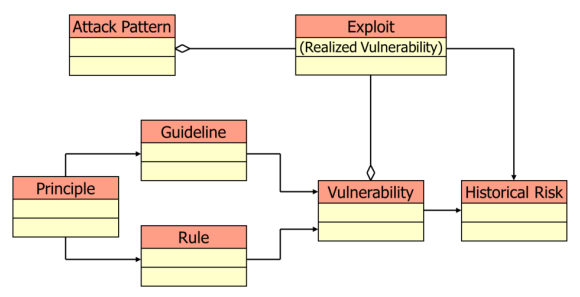
*Ausnutzung*  
Beschreibungen, wie Fälle von Schwachstellen verwendet werden, um die Sicherheit bestimmter Systeme zu beeinträchtigen.

*Angriffsmuster*  
Beschreibungen von gemeinsamen Sätzen von Exploits in einer abstrakteren Form, die über mehrere Systeme angewendet werden können.

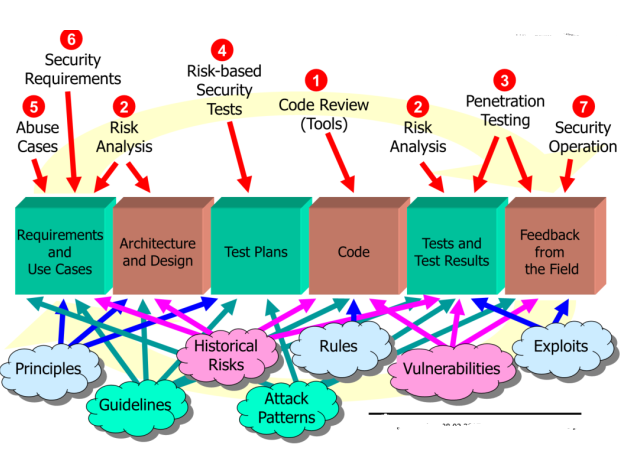
**Historisches Wissen**

*Historische Risiken*  
Detaillierte Beschreibungen von spezifischen Fragen, die in real-wolrd Software-Entwicklungsbemühungen aufgedeckt wurden. Muss eine Aussage über die Auswirkungen auf die Geschäfts- oder Missionspräparate enthalten. Als Ressource bietet dieses Wissen einen enormen Wert bei der Identifizierung ähnlicher Probleme bei neuen Software-Bemühungen, ohne von vorne anzufangen.

#### Die zusammenhängende Architektur des Wissens



#### Das Wissen in den Best Practices



## Code Review Tools

|  |  |
| --- | --- |
| First Generation Code Scanners | Advanced Source Code Analysis Tools |

## Coding Rule Sets

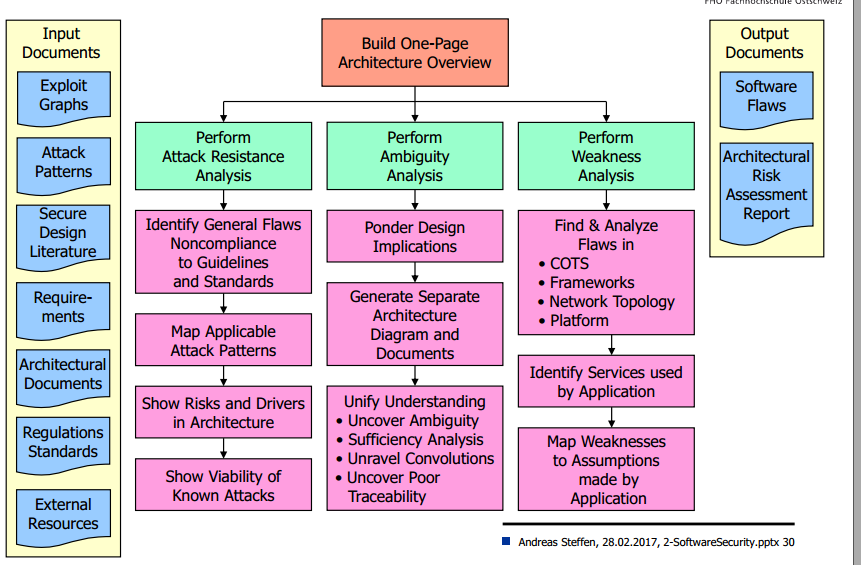
In den Codierungsregelsätzen, die von kommerziellen und freien Quellcodeanalyse-Tools verwendet werden, ist ein umfangreiches Fachwissen enthalten. Das US-Departement Von Homeland Security "Build Security In" -Portal zählt derzeit 173 grundlegende C / C ++ - Codierungsregeln, die hauptsächlich von ITS4 abgeleitet werden.

## Architectural Risk Analysis

**Angriffsresistenzanalyse**  
Checkliste-Ansatz mit Katalogen von Angriffsmustern und Schwachstellen. Gut bei der Suche nach bekannten Problemen, aber nicht mit neuen oder kreativen Angriffen.

**Ambiguitätsanalyse**  
Am besten durch ein Team von erfahrenen Analysten, die zuerst die verfügbaren Software-Artefakte unabhängig studieren und dann ihre Ergebnisse diskutieren. Wo gute Architekten nicht einverstanden sind, liegen in der Regel interessante Dinge (und manchmal auch neue Mängel). Gut bei der Suche nach Mehrdeutigkeit und Inkonsequenz in einem Design.

**Schwächenanalyse**  
Gut zum Verständnis der Auswirkungen von externen Software-Abhängigkeiten, z.B. Software, die auf .NET- oder J2EE-Frameworks aufgebaut ist oder Sicherheitsbibliotheken verwendet. Abhängigkeiten von Netzwerktopologie oder Plattformumgebung.



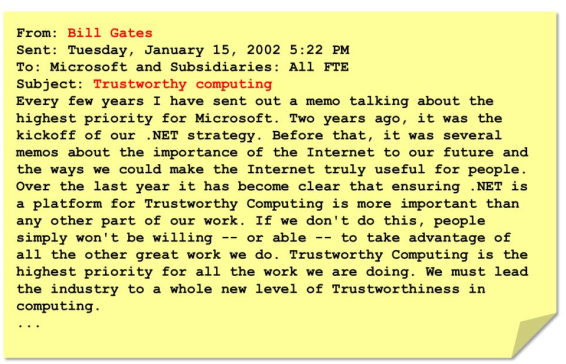
# Microsoft Security Development Lifecycle

## Die 25 schlechtesten Produkte aller Zeiten

Auf Platz 8 stand in 2006 der Microsoft Internet Explorer 6. Sie gilt als die am schlechtesten gesicherte Software auf diesem Planeten.

## Trustworthy Computing Initiative in 2002

Durch Bill Gates, um die Microsoft Produkte wieder sicher zu machen.



Das Dokument «Microsoft Security Development Lifecycle» hat in der Version 5.0 rund 134 Seiten. Der aktuellste Stand stammt aus dem Jahre 2010. Es gibt dazu eine gekürzte Version von rund 17 Seiten. Zu finden sind beide Dokumente unter [www.microsoft.com/sdl](http://www.microsoft.com/sdl) .

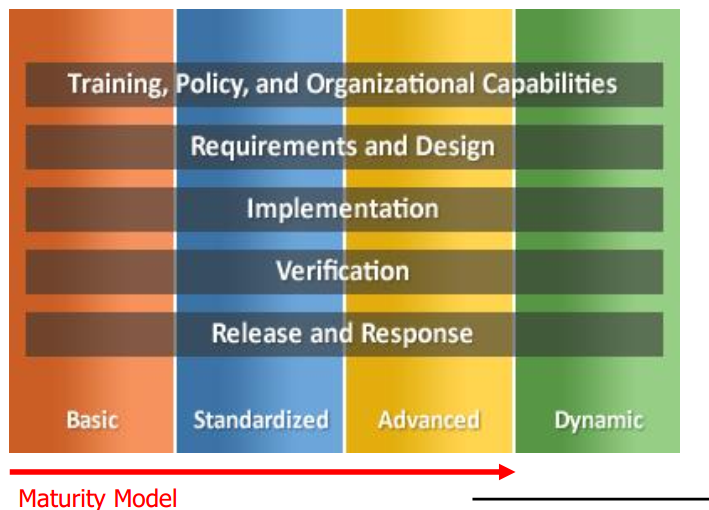
## Die drei Hauptkonzepte des Microsoft SDL

**Schulung (Education)**  
Die laufende Ausbildung und Ausbildung der technischen Berufsfelder ist entscheidend. Investitionen in den Wissenstransfer helfen, auf Veränderungen in Technologie und Bedrohungslandschaft zu reagieren.

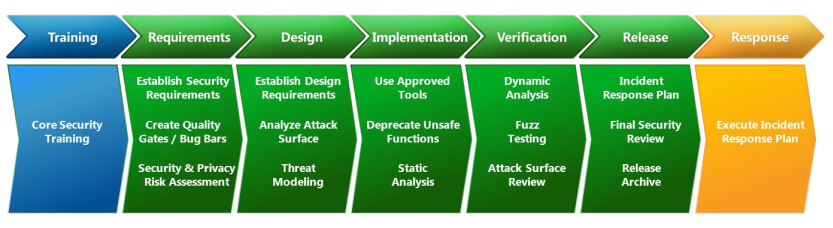
**Kontinuierliche Prozessverbesserung**Regelmäßige Auswertung von SDL-Prozessen und Änderungen in Reaktion auf neue Technologien oder neue Bedrohungen. Erhebung von Daten zur Bewertung der Trainingswirksamkeit, der In-Process-Metriken zur Bestätigung der Prozess-Compliance und der Metriken nach der Veröffentlichung, um zukünftige Änderungen zu begleiten.

**Rechenschaftspflicht**  
Archivierung aller Daten, die notwendig sind, um eine Anwendung in einer Krise zu bedienen. Detaillierte Sicherheitsreaktionen und Kommunikationspläne.

### SDL Optimization Model



### Das Modell



Der Microsoft Security Development Lifecycle (SDL) definiert 16 Best Practices, die dazu bestimmt sind, Software sicherer zu machen. Praktische Erfahrungen deuten darauf hin, dass Anwendungen, die eine oder mehrere der folgenden Merkmale aufweisen, dem SDL unterliegen sollten:

* In einer Geschäfts- oder Unternehmensumgebung eingesetzt
* Verarbeitet persönlich identifizierbare Informationen oder andere sensible Informationen
* Kommuniziert regelmäßig über das Internet oder andere Netzwerke

# Vorlesung Woche 4