SOLCEPT
embedding your ideas!

Embedded Secure by Design

Verschlüsselung, Zertifikate...
Was sollen wir denn sonst noch tun??

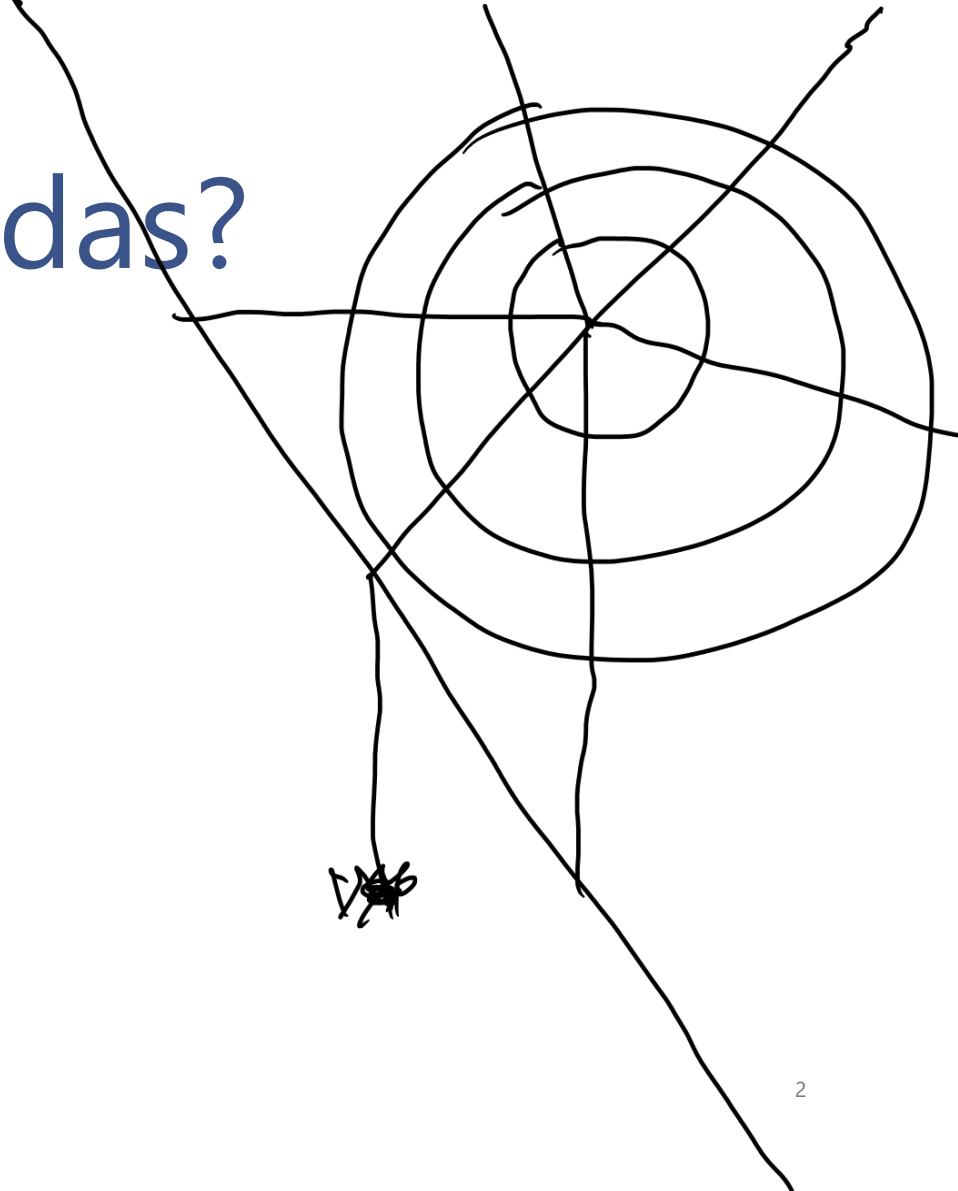
Alois Cavelti, Solcept AG (Präsentation: Andreas Stucki)

Wieso braucht es das?

Vernetzung d.h. Angriffsfläche

-> Regulierung

-> Markt/ Ruf

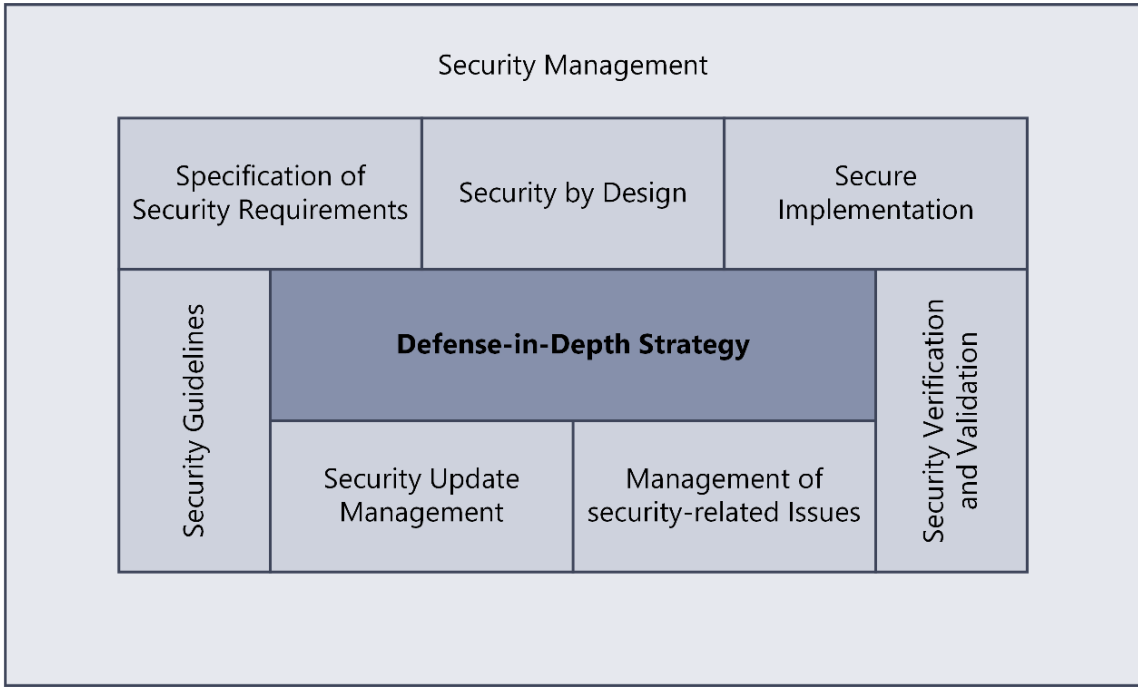


Wieso Defense in Depth?

Kernidee: alles ist kompromittiert:

jede Komponente, jedes Subsystem, jede Schnittstelle

Managing Risk with Quality

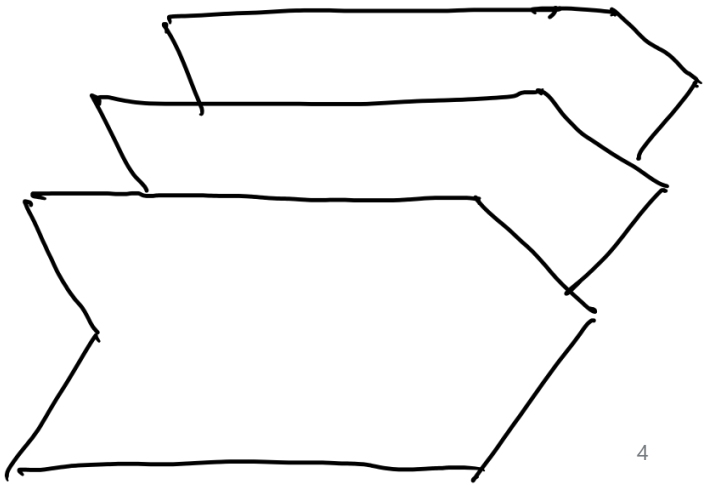


Was ist gefordert?

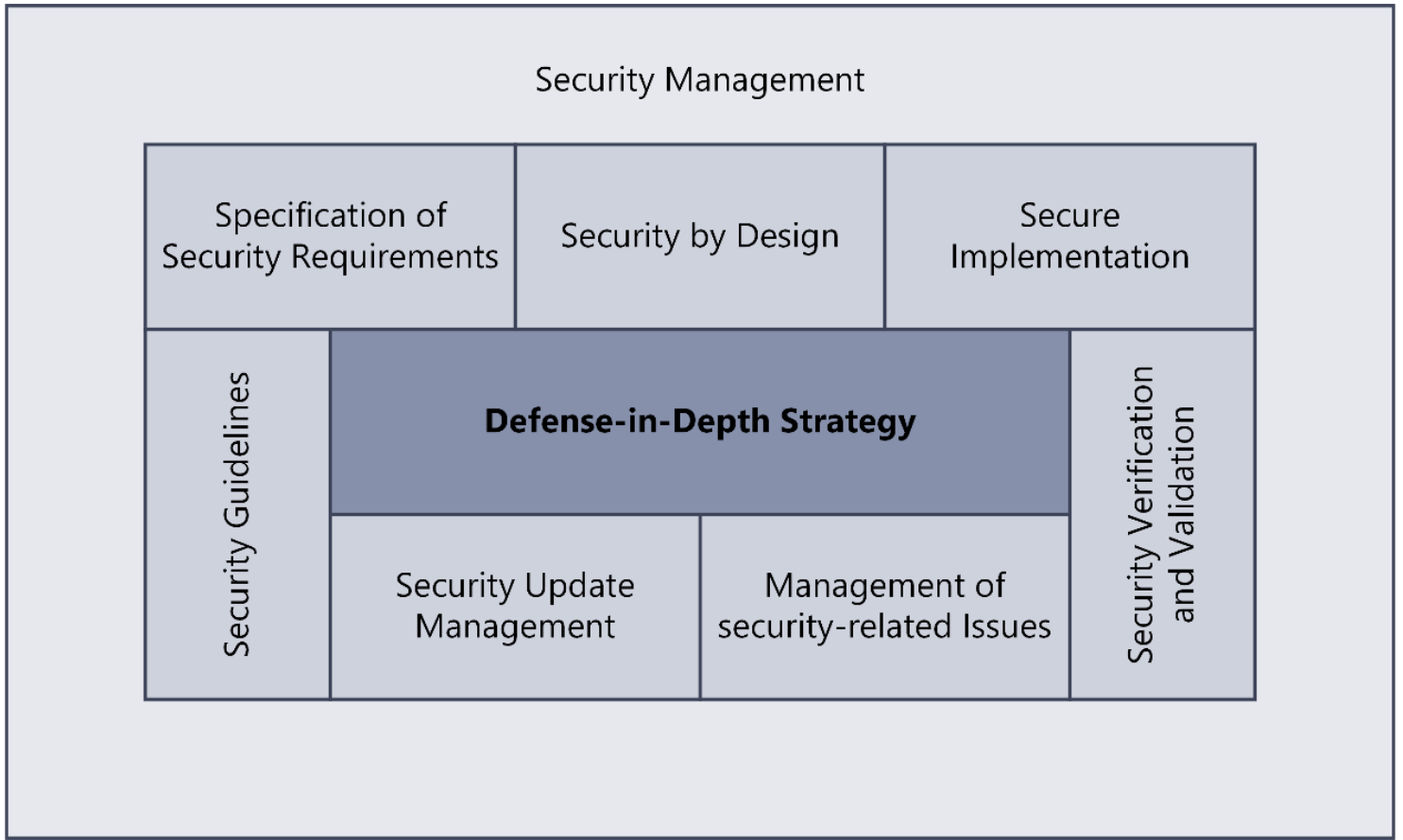
Technische Lösungen: teilweise

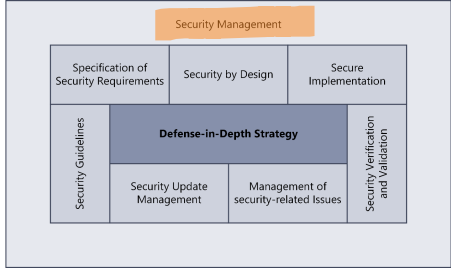
Vor allem Praktiken == Prozesse

Safety: dito



Was ist gefordert?





1 – Sicherheits-management I

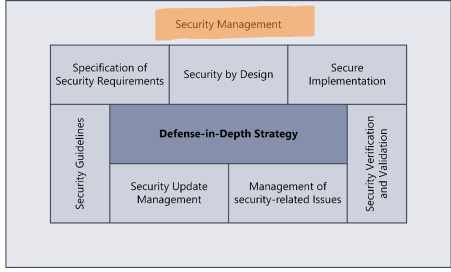
Planung

Dokumentation

Ausführung

Sicherheitsplan

1.
2.
3.
4.
5.



1 – Sicherheits-management II

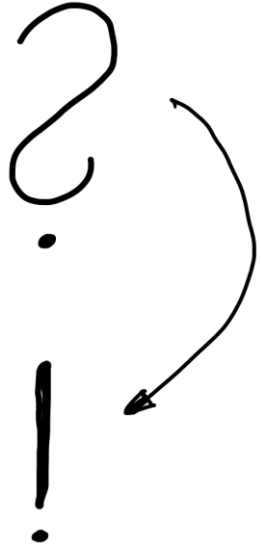
Sicherheitsplan:

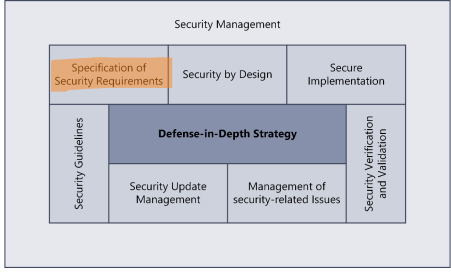
Was wollen wir wie tun

Wie erfüllen wir die Normen

Sicherheitsnachweis:

Was haben wir getan

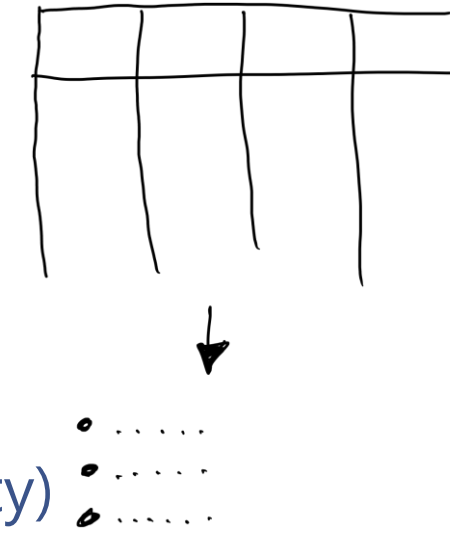




2 – Spezifikation von Sicherheitsanforderungen

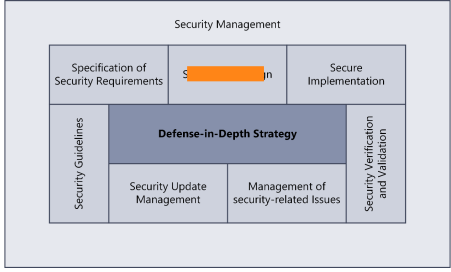
Bedrohungsmodell über den Lebenszyklus

Abgeleitet: dokumentierte Sicherheitsanforderungen



Safety-Analysen/ Safety-Anforderungen

Achtung: Modell und Anforderungen ändern (!= Safety)



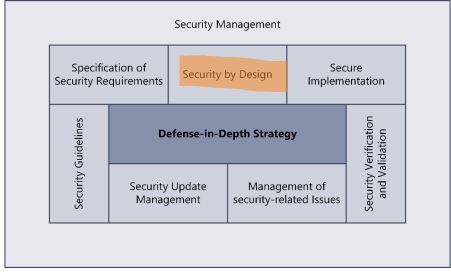
3 – Sicherheit durch Design/ im Design I

Im engeren Sinn

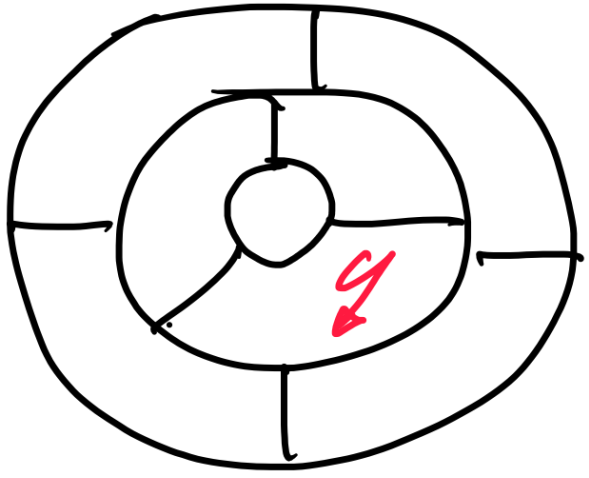
Berücksichtigung der Sicherheit vom Systemdesign bis Detaildesign

Schnittstellen

- äussere UND innere



3 – Sicherheit durch Design/ im Design II

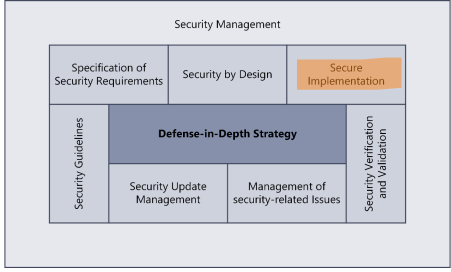


Technische Defense in Depth:

Verteidigunsschichten: jede Schicht, jedes Subsystem wird als komprimittiert angesehen

Sichere Technolgien (Secure Boot, Verschlüsselung, mechanischer Schutz…)

Safety Design, Sicherheitsmechanismen



4 – Sichere Implementation

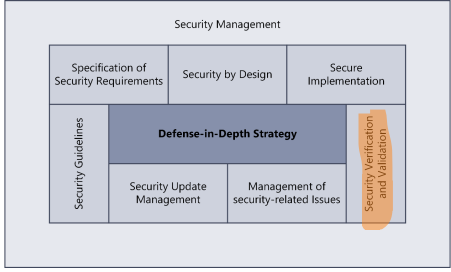
Vorgaben:

Design Rules, Coding Rules, Hardware-Rules…

Reviews & statische Codeanalyse

Dynamische Codeanalyse (Unit Test & Code Coverage)

Implementation Safety



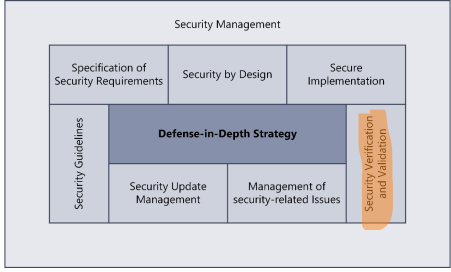
5 – Verifikation & Validation I

Test der Sicherheitsanforderungen

- Traceability!
- Risiko-basiert

Automatisiert oder manuell

Safety Testing



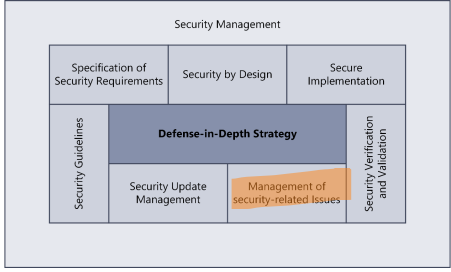
5 – Verifikation & Validation II

Test der Verteidigunsschichten

Test auf bekannte Schwachstellen (MITRE, OWASP…)

Penetrationstests

!= Safety



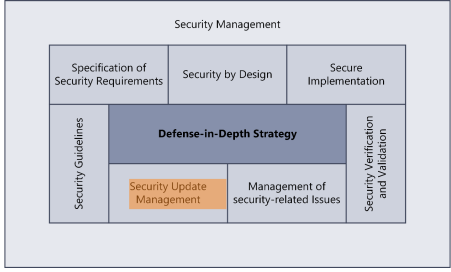
6 – Management von sicherheitsrel. Sachverhalten

Sicherheitsprobleme aufnehmen & behandeln

Quellen:

- intern (Tester…)
- extern (Lieferanten, Anwender, Mitre (CVE)…)

CAPA's…



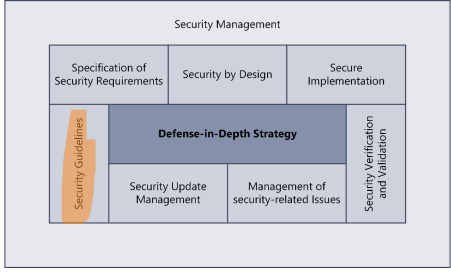
7 – Verwaltung von Sicherheitsupdates

Sicherheitsupdates zur Verfügung stellen

Und vorher testen

mindestens Regression

!= Safety

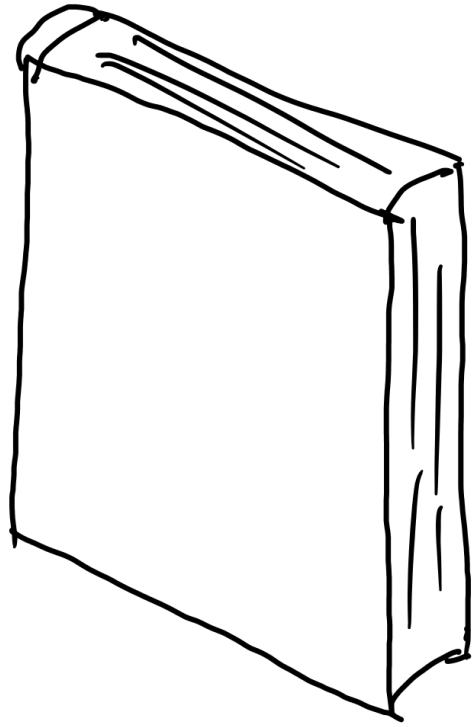


8 – Sicherheitsrichtlinien für Benutzer

Benutzerdokumentation für sichere:

Konfiguration, Betrieb, Wartung, Entsorgung

d.h. voller Lebenszyklus



Safety Manual

Was müssen Sie nun tun?

Prozesse, Richtlinien und Checklisten schon vorhanden?

z.B. für Safety (funktionale Sicherheit): «nur» anpassen

Sonst: damit starten, denn um Security werden Sie weniger herumkommen als um Safety

Fragen? Anregungen?

Links zu externen Quellen unter

https://www.solcept.ch/de/blog/kritische-systeme/secure-by-design/

oder a.stucki@solcept.ch

Änderungsverzeichnis

Version	Entwurf/ für Review/ Freigegeben	Datum	Verant-wortlich	Kommentare/ Änderungsverlauf
00.01	Entwurf	2023-11-20	a2s	first draft
00.02	Entwurf	2023-12-05	a2s	as presented
01.00	Freigegeben	2023-12-08	a2s	no changes