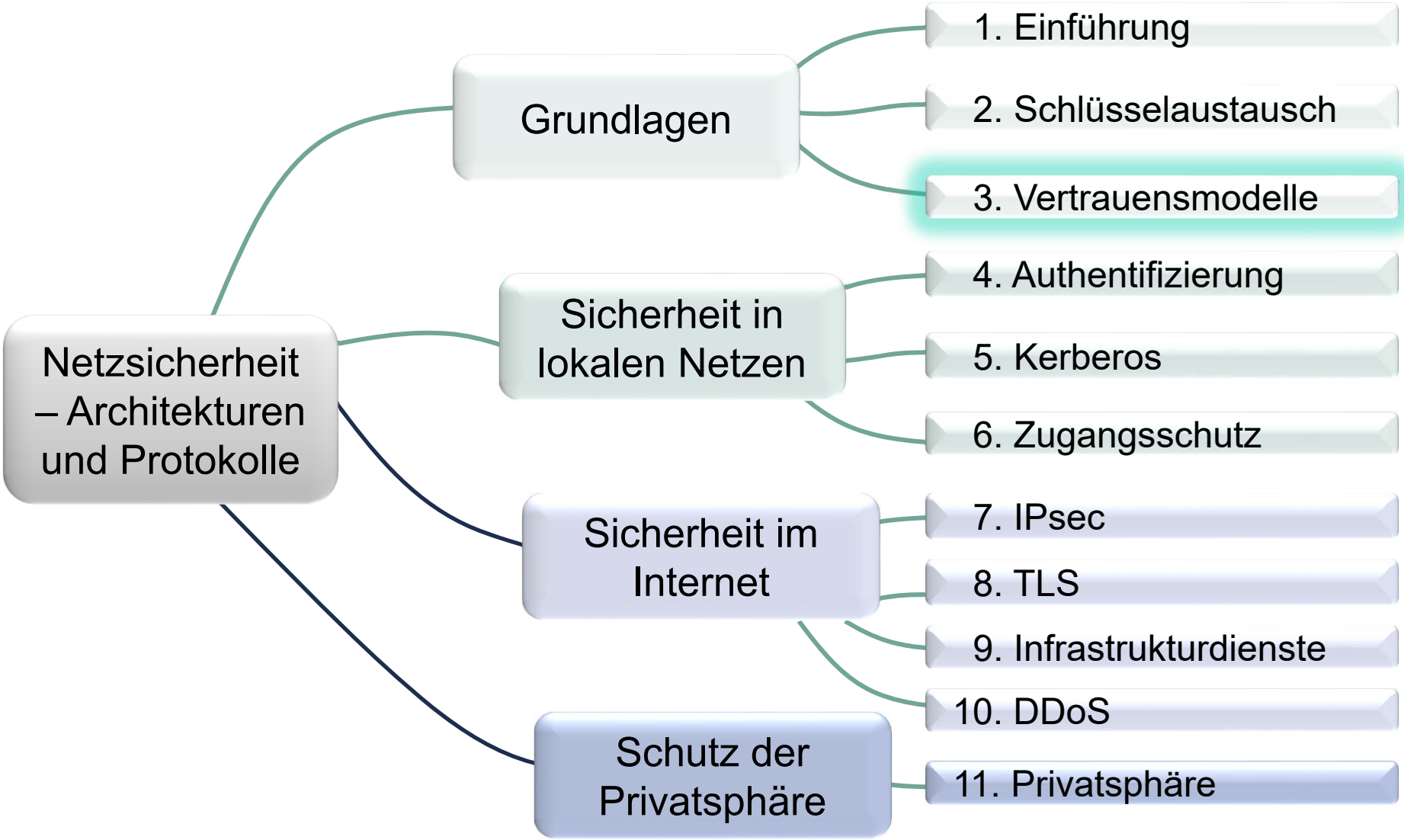
Vorlesung Netzsicherheit
Kapitel 3 – Vertrauensmodelle

PD Dr. Ingmar Baumgart, PD Dr. Roland Bless, Matthias Flittner, Prof. Dr. Martina Zitterbart
baumgart@fzi.de, [bless, flittner, zitterbart]@kit.edu

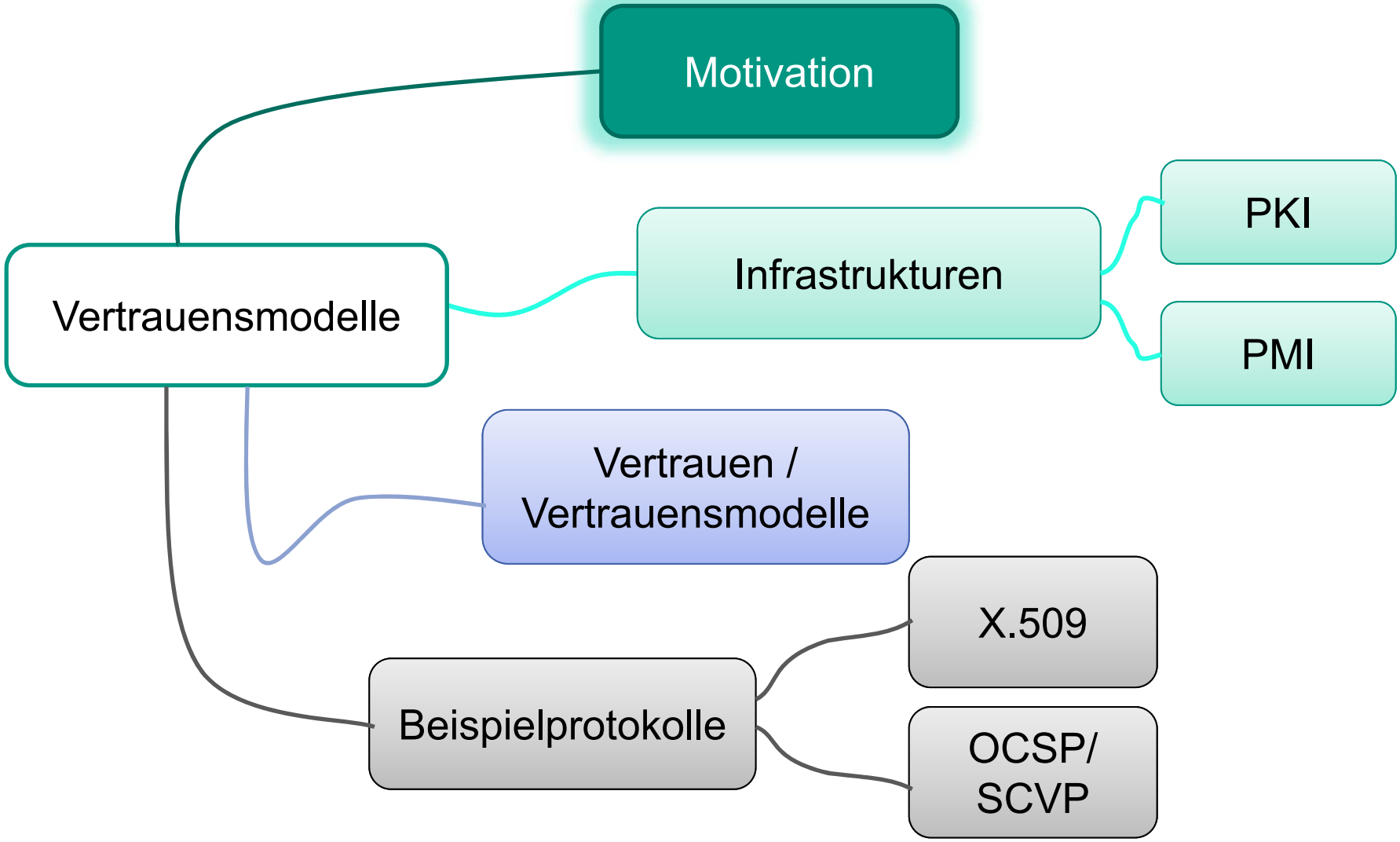
Institut für Telematik, Prof. Zitterbart

© Peter Baumung

Inhalte der Vorlesung

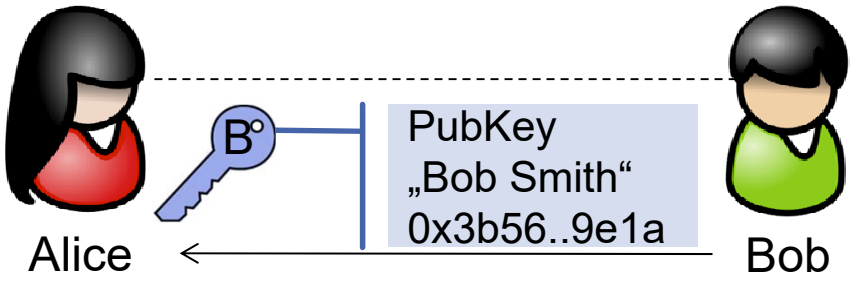


Überblick


Motivation
Vertrauensmodelle
Infrastrukturen
PKI
PMI
Vertrauen / Vertrauensmodelle
Beispielprotokolle
X.509
OCSP/ SCVP

Wiederholung: Beziehen öffentlicher Schlüssel

- Alice benötigt öffentlichen Schlüssel von Bob
- Frage: woher bezieht man den öffentlichen Schlüssel?
  - Persönlicher Austausch?
    - Alice identifiziert Bob als Person
    - Bob muss versichern, dass 0x3b56..9e1a sein Public Key ist
    - Keine elegante Lösung für das Schlüsselverteilproblem



  - Via E-Mail oder Web-Site? → Maskerade ggf. ebenfalls möglich

→ Sichere Zuordnung benötigt:
Öffentlicher Schlüssel ⇔ Identität Kommunikationspartner

Wiederholung: Digitale Zertifikate

Problemstellung

- Authentifizierung eines Sachverhaltes, den man nicht selbst überprüfen kann

→ man verlässt sich auf vertrauenswürdige Dritte, die ihn schon kontrolliert haben

Frage: was ist ein digitales Zertifikat?

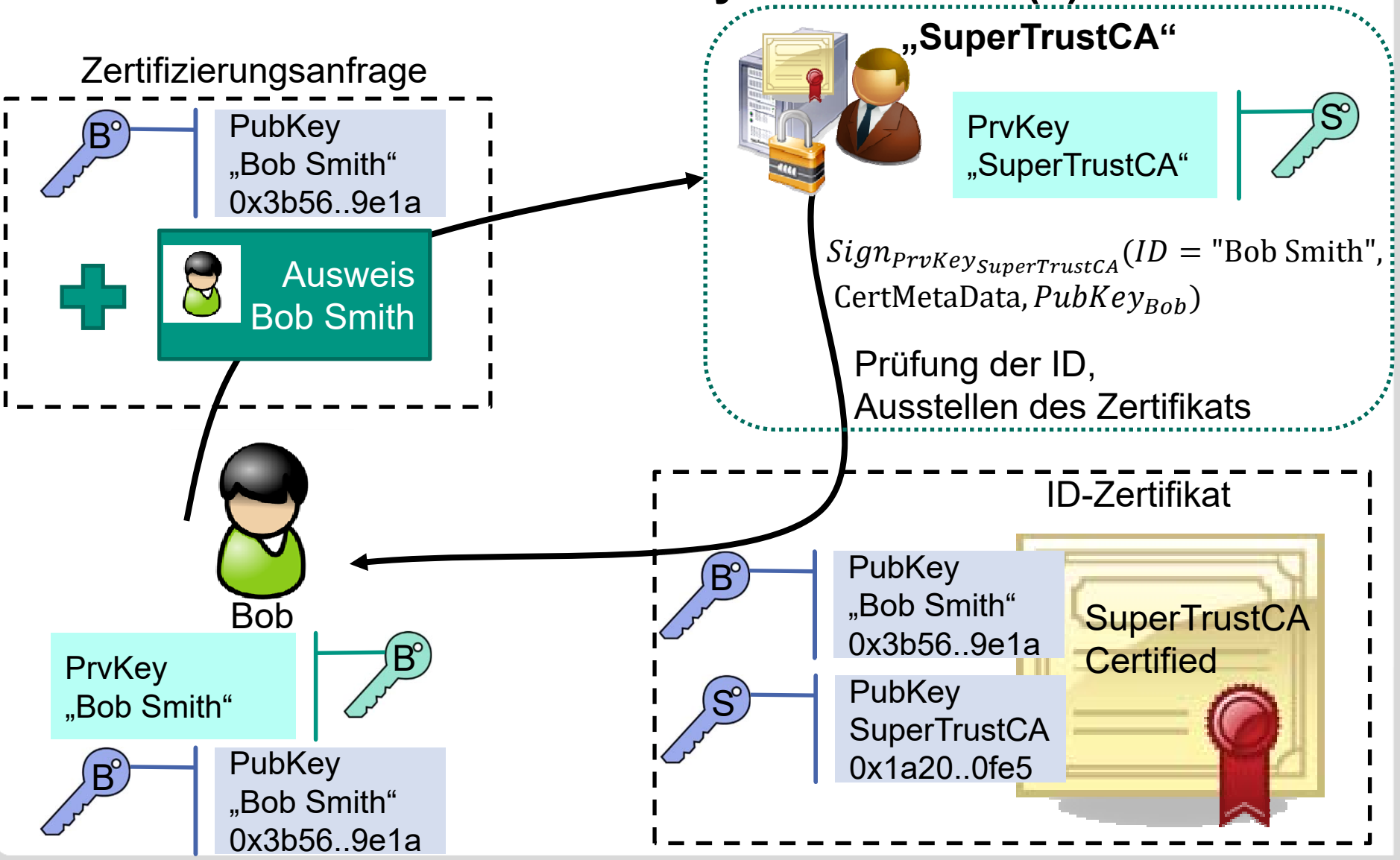
- ein digitales Dokument, in dem eine Instanz einen bestimmten Sachverhalt mittels digitaler Signatur bestätigt

→ erzeugt Vertrauen in den Sachverhalt z.B. ID-Zertifikat: E-Mailadresse alice@wonderland.org → PubKey Alice

Frage: wer erstellt die Zertifikate?

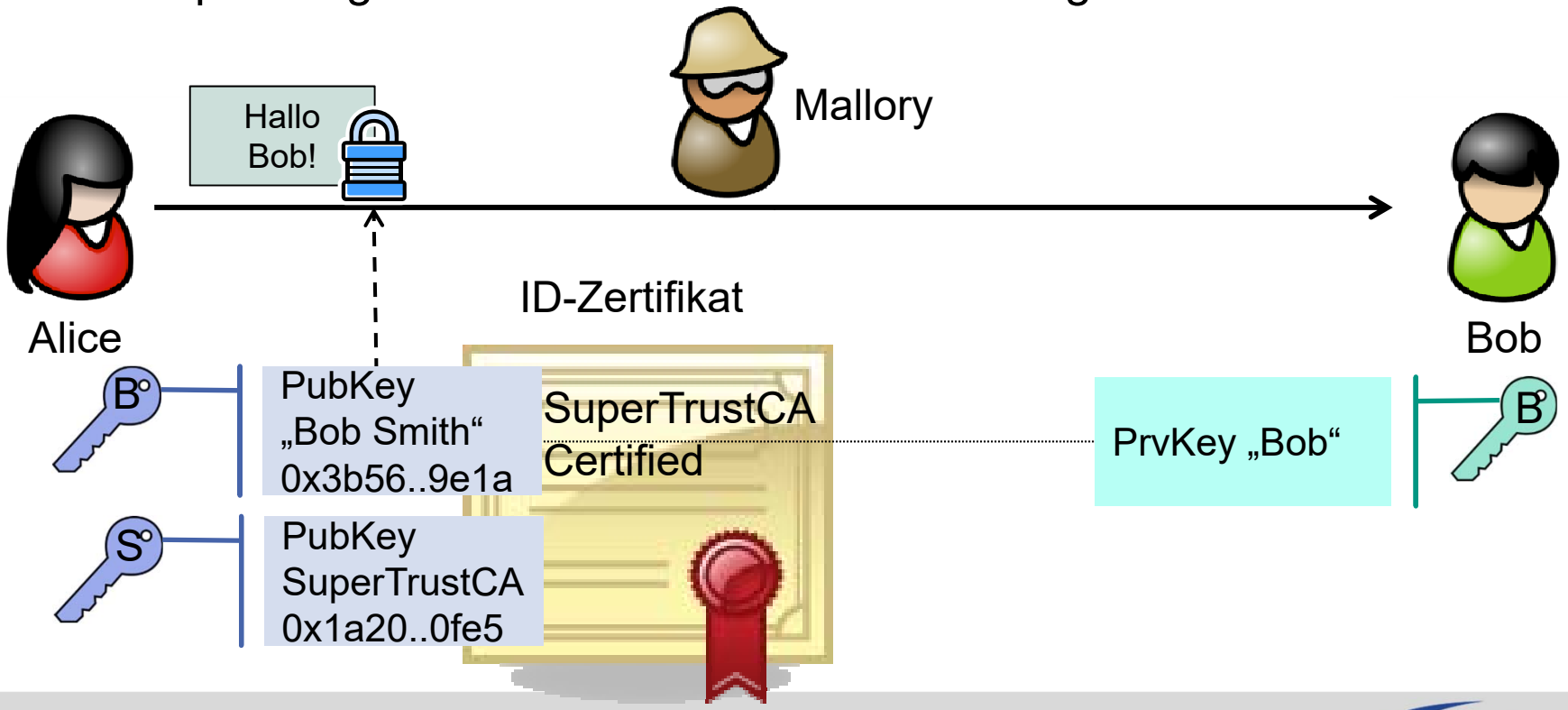
- eine vertrauenswürdige Instanz: Certification Authority (CA)
  - Beispiele für CA: Institution, Behörde, eine einzelne Person

Alice und Bob – mit Public Key Zertifikaten (1)

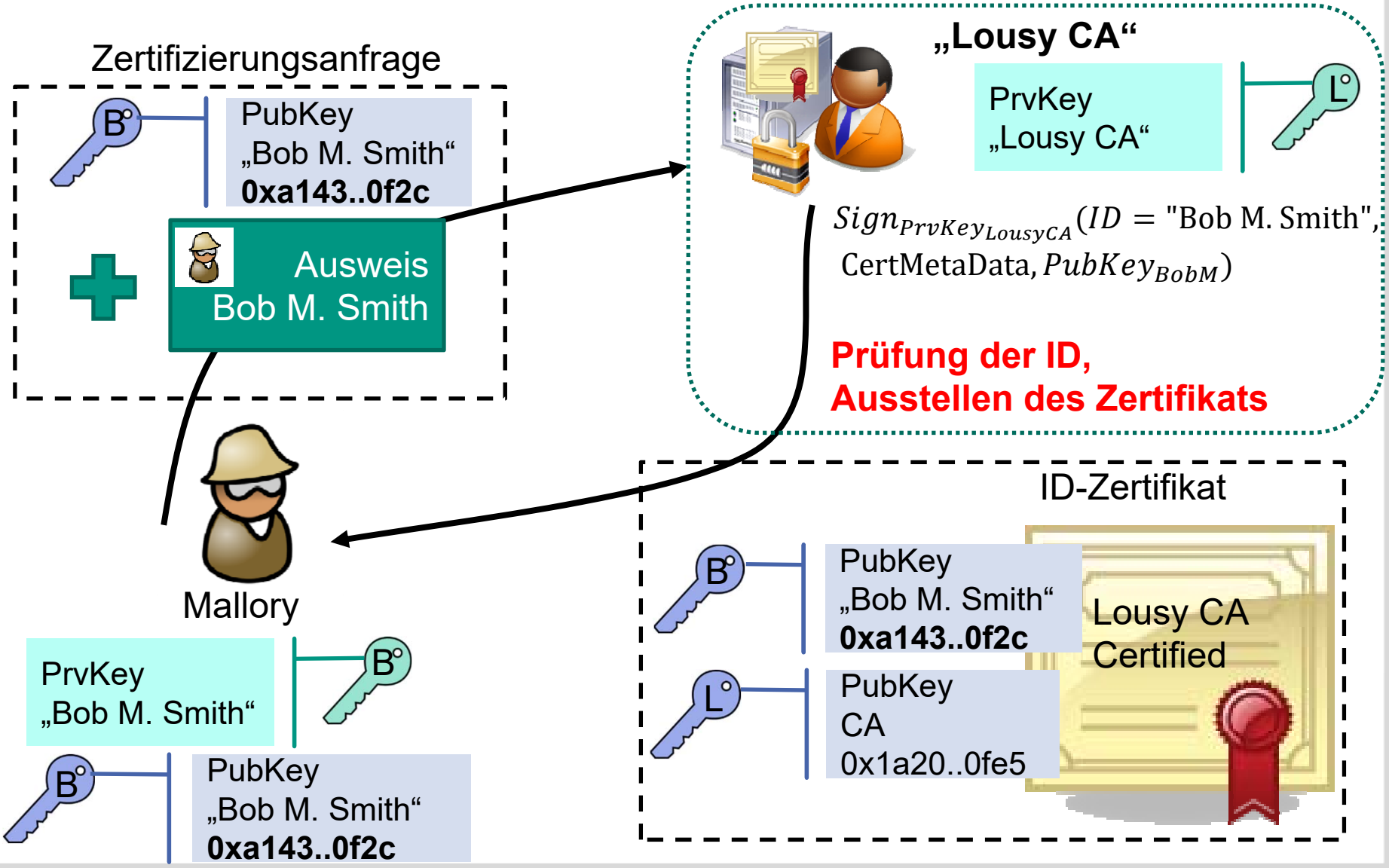


Alice und Bob – mit Public Key Zertifikaten (2)

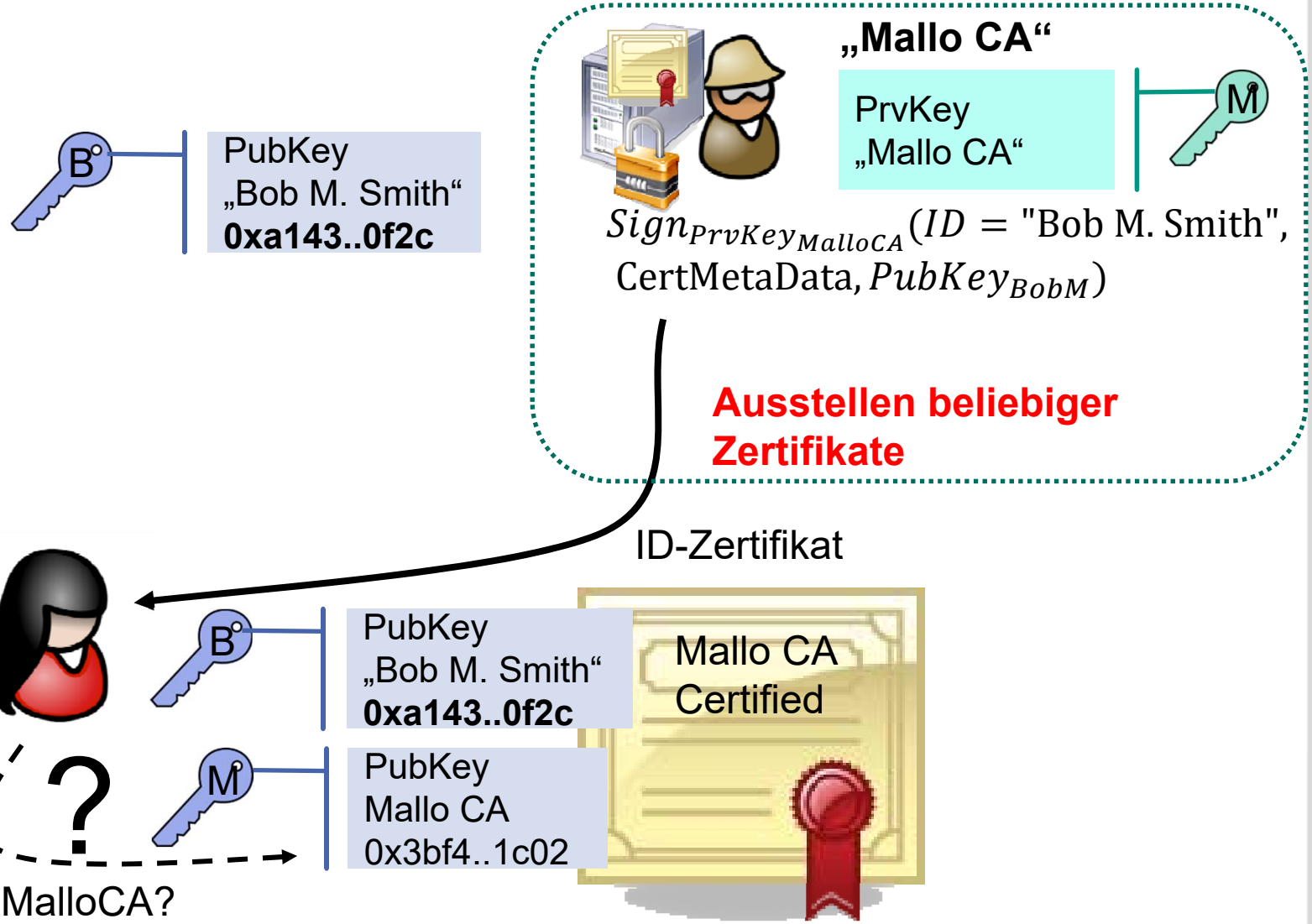
- Alice beschafft sich Bobs Zertifikat, z.B.
  - von Bob
  - aus einem öffentlichen Verzeichnis
  - von anderer Quelle…
- Alice prüft Signatur des ID-Zertifikats und Gültigkeit



Neues Problem – Vertrauen in CA



Neues Problem – Vertrauen in CA



Neue Probleme (1)

- Vertrauen in CA bzw. deren Integrität
  - Wie gewissenhaft wird die Identität der Nutzer geprüft?
  - Wie gewissenhaft wird bei der CA gearbeitet (Einhaltung der Policies)?
  - Korruption möglich?
  - Sicherheitsmaßnahmen zum Schutz der CA-Schlüssel?
- Gültigkeit
  - Vergleich aktuelle Zeit mit Ablaufdatum
  - Gab es einen Widerruf?
  - Ist die CA überhaupt berechtigt den Sachverhalt zu beglaubigen?
- Authentizität des öffentlichen Schlüssels der CA?
  → Problem nur auf CA verschoben?
- Skalierbarkeit?
- CA für CA notwendig? → CA-Hierarchie (CA + Sub-CAs)

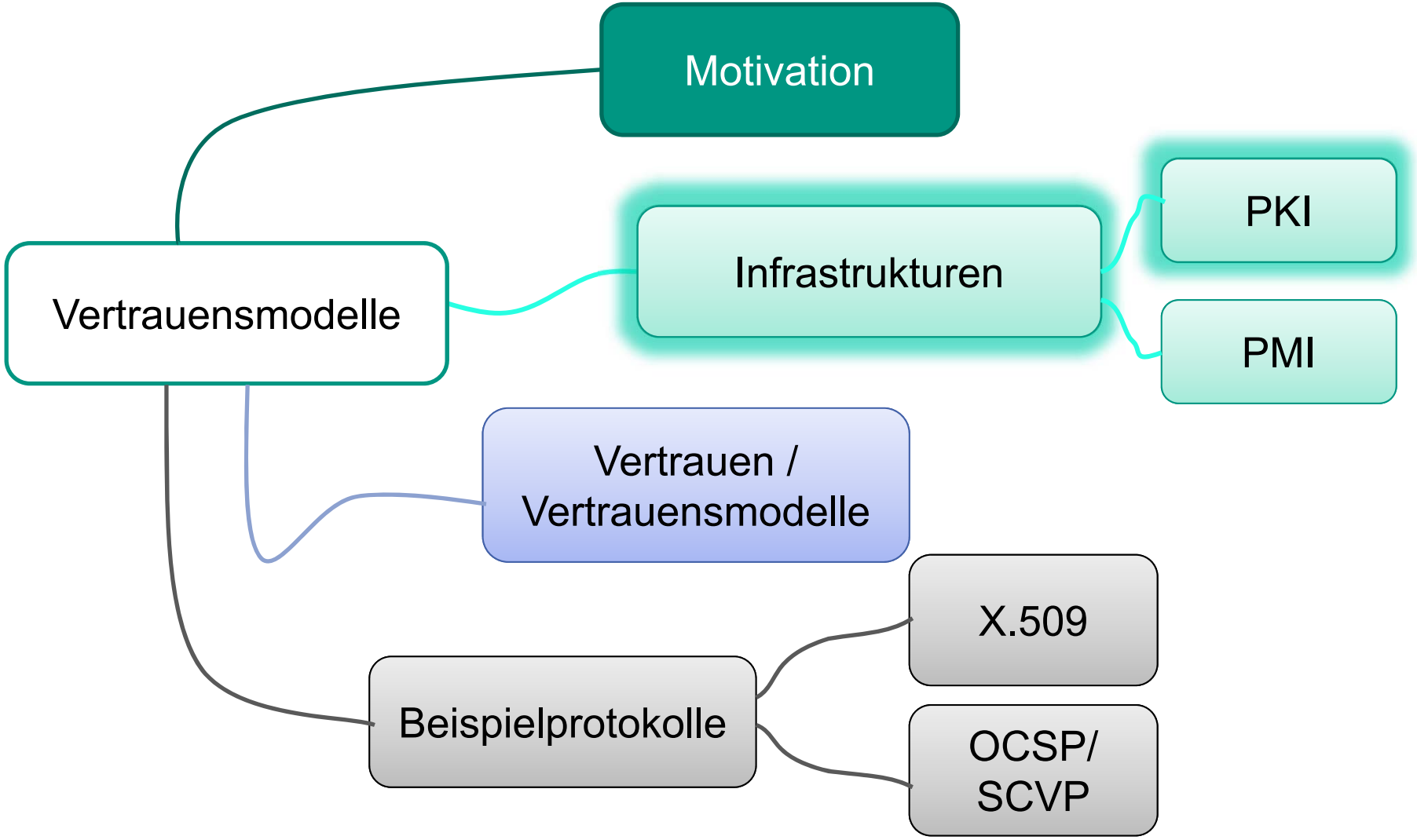
Neue Probleme (2)

- Welche CA ist zur Ausstellung eines bestimmten Nutzerzertifikats autorisiert?
  - Annahme es gibt zwei Zertifikate für alice@wonderland.org $Cert_{SuperTrustCA}$ + $Cert_{LousyCA}$, welches ist das Richtige oder sind beide gültig?
- Wer steht an der Spitze der Hierarchie? → Root-CA?
  - Abhängigkeit aller Sub-CAs von der Root-CA
  - Single Point of Failure
  - Machtposition
  - absolutes Vertrauen in Integrität der Root-CA
    - Auswirkungen von Sicherheitsvorfall bei Root-CA?

Digitale Zertifikate – Widerruf

- Widerruf digitaler Zertifikate manchmal notwendig
  - falsch/irrtümlich ausgestellte Zertifikate
  - privater Schlüssel nicht mehr nutzbar, z.B.
    - befürchtete Kompromittierung des privaten Schlüssels
      - z.B. Trojaner oder Fehler wie TLS-Heartbleed-Bug
    - Verlust des privaten Schlüssels bzw. dessen Passworts
      - Crash der Festplatte, Zerstörung der Chip-Karte
      - Vergessen des Passwortes, das den Schlüssel sichert
      - Diebstahl des Rechners
  - Änderung von Angaben (z.B. Firmenzugehörigkeit) oder Parametern (Algorithmus unsicher)
- CA verwaltet Liste mit widerrufenen Zertifikaten – Certificate Revocation List (CRL)
- Problem
  - Widerruf muss ebenfalls gesichert werden → CRL sollte von CA signiert sein
  - Aktuelle Information notwendig (Online-Verfahren sinnvoll)

Überblick


Motivation
Vertrauensmodelle
Infrastrukturen
PKI
PMI
Vertrauen / Vertrauensmodelle
Beispielprotokolle
X.509
OCSP/ SCVP

Was ist eine Public-Key-Infrastruktur?

- Public Key Infrastructure (PKI)
  - ist eine Infrastruktur zum Management von ID-Zertifikaten
  - ermöglicht somit Authentifizierung öffentlicher Schlüssel

- Bausteine einer PKI
  - Organisatorische Bausteine
    - Zertifizierungsrichtlinie (Certification Policy)
    - Dokumentation interner Abläufe (Certification Practice Statement)
      - soll Vertrauen in CA stärken (Transparenz)
    - Zugrundeliegendes Vertrauensmodell
  - Technische Bausteine
    - Zertifikatsformat (z.B. X.509), welches das Vertrauensmodell unterstützt
    - Managementprotokolle zur technischen Umsetzung der PKI-Dienste

Anforderungen an eine PKI

- Sicherheit interner Abläufe
  - Registrierung, Prüfung von Identität und Schlüsselpaar
  - Zusammenspiel der Komponenten
  - Prozesse im Fehlerfall
- Sicherheit der Signaturschlüssel der CA
  - i.d.R. Rechner ohne Netzwerkanschluss
  - Physikalische Zugangsbeschränkung (CA-Rechner in einem Safe)
  - Aufwendige Authentifizierung (z.B. via Smartcards, evtl. 4- bzw. Mehr-Augen-Prinzip)
- Effizienz: Validierung eines Zertifikates
- Skalierbarkeit
- Komfort: vertretbarer Aufwand für Benutzer
  - Entfernung zur nächsten Registrierungsstelle
  - Zeitlicher Aufwand bei Registrierung/Zertifizierung
- Vertrauenswürdigkeit

PKI-Modell (1)

Eine PKI besteht aus folgenden Elementen

- Benutzer (Subject, End Entity)
  - Mensch, Maschine oder Prozess
  - meldet sich bei der PKI an (Enrollment) und lässt sich ein Zertifikat ausstellen
  - will andere öffentliche Schlüssel authentifizieren

Zertifizierter Benutzer

- Registration Authority (RA)
  - Implementiert administrative Aspekte der PKI
  - Schnittstelle zwischen Benutzer und CA
    - Prüft z. B. Identität der Nutzer
  - Entkopplung (RA i.d.R. offline!)
  - evtl. direkt Teil der CA

PKI-Modell (2)

- Certification Authority (CA)
  - Implementiert Zertifizierung
  - folgt technischen Standards, die Formate spezifizieren
  - erzeugt durch Signatur Zertifikate
  - Schutz dieses Signaturschlüssels
  - Erstellung von Widerruflisten

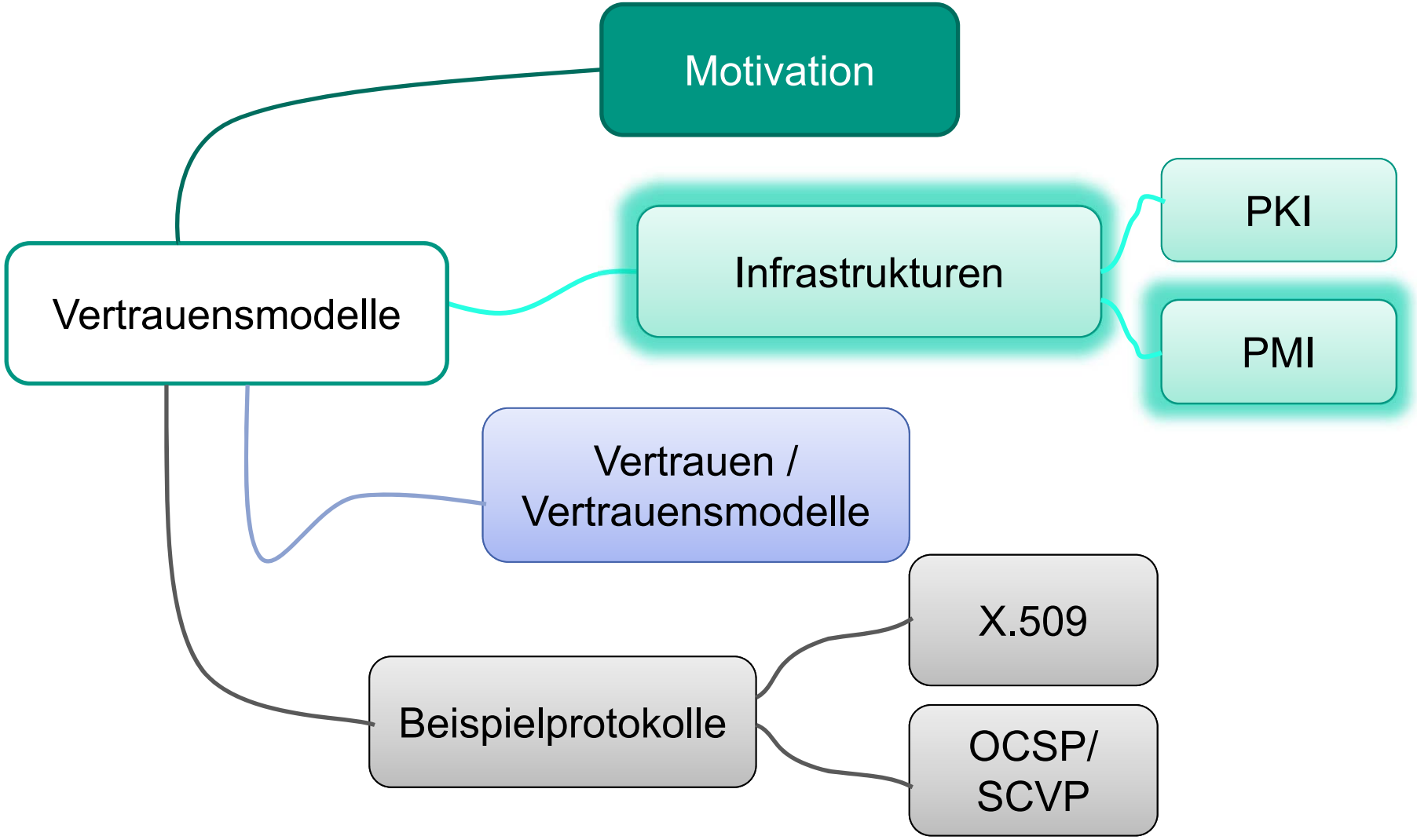
CA

- Speicher/Verzeichnis (Directory)
  - für ausgestellte Zertifikate, gültige und historische
  - Abruf von Zertifikaten über diverse Protokolle möglich (z.B. LDAP)
  - Publizierung der Gültigkeit von Zertifikaten über Widerruflisten

Widerruf von ID-Zertifikaten

- Eine CA kann ein Zertifikat (evtl. vom Benutzer ausgelöst) vor Ablauf seiner Gültigkeitsdauer widerrufen, wenn
  - das Zertifikat nicht mehr benutzt wird
  - der private Schlüssel nicht mehr nutzbar ist
  - der zu einem Zertifikat gehörige private Schlüssel sicher oder eventuell kompromittiert wurde
  - Angaben in dem Zertifikat nicht mehr stimmen
  - Parameter des Schlüsselpaares oder die eingesetzten Algorithmen (z.B. zur Signatur, z.B. sha1WithRSAEncryption) nicht mehr adäquat sind
- Abruf einer Liste mit ungültigen Zertifikatsnummern (CRL)
  - Glaubwürdigkeit der Liste?
  - Muss ebenfalls von der ausstellenden CA signiert werden

Überblick


Motivation
Vertrauensmodelle
Infrastrukturen
PKI
PMI
Vertrauen / Vertrauensmodelle
Beispielprotokolle
X.509
OCSP/ SCVP

Autorisierung

Bisher

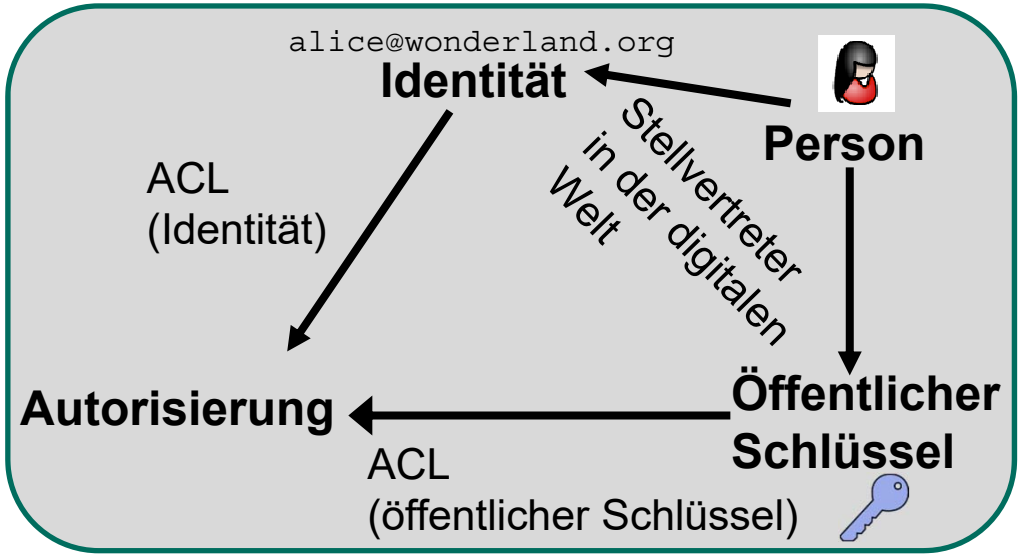
- Authentifizierung → realisiert über ID-Zertifikate

Jetzt

- Wie kann Autorisierung realisiert werden?
  - welcher Benutzer hat in Bezug auf eine Ressource welche Privilegien?
  - wer darf Privilegien vergeben?
  - wer darf Privilegien weitergeben?

Autorisierung

- Autorisierung über Zugriffskontrollliste (Access Control List – ACL)
- definiert, wer auf eine Ressource zugreifen darf, z.B. anhand von
  - Identität (via Passwort/Ticket/etc., z.B. pop, imap, smtp-auth)
  - Besitz eines privaten Schlüssels (z.B. SSH: authorized_keys)



- Problem: Zuordnung von Privilegien zu Personen lokal oder auf Privilegien-Server zentral

Access Control Methoden

- Discretionary Access Control
  - individuelle Rechte pro Benutzer
  - feingranulare Rechtevergabe und Zugangskontrolle

- Mandatory Access Control
  - jede Ressource wird klassifiziert
  - Benutzer bekommen Zugangsrechte zu Klassen

- Role-based Access Control
  - Rechte abhängig von der Rolle des Benutzers

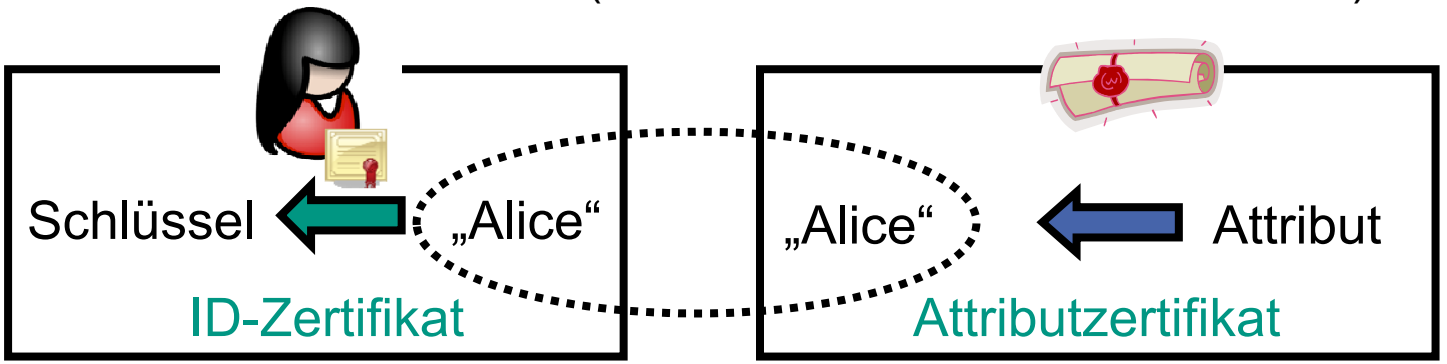
Admin → {read, execute}

  - Rollen ist eine Menge von Zugriffsrechten zugeordnet
  - Vorteil: Benutzer können wechseln, ohne dass die Rechte geändert werden müssen
- Hierarchical Role-based Access Control
  - Hierarchische Organisation der Rollen, d.h. höhere Rollen bauen auf niedrigeren Rollen auf
  - Erben und Vererben von Zugriffsrechten

Attributzertifikate

Nachweis der Autorisierung über Attributzertifikate

- attestieren Identität bestimmtes Privileg (=Attribut)
- zeitlich einschränkbar
- basieren auf PKI und deren ID-Zertifikaten
- können widerrufen werden (Attribute Certificate Revocation List)



- Analog zur PKI ist eine Infrastruktur zum Management dieser Zertifikate notwendig

Privilege Management Infrastructure (PMI)

→ PMI ist für Autorisation, was PKI für Authentifizierung ist

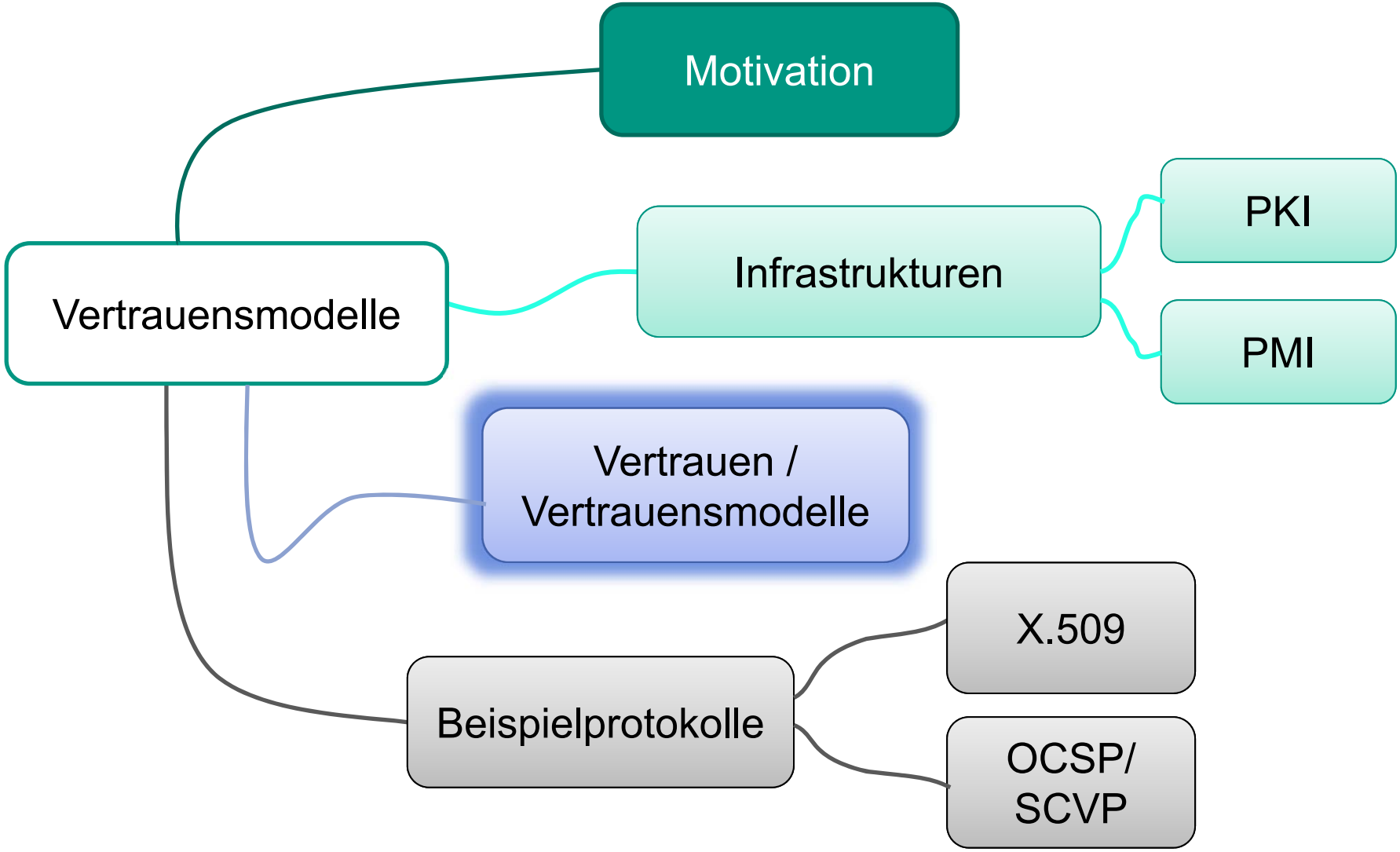
Privilege Management Infrastructure

- ist eine Infrastruktur zum Management von Zugriffsrechten basierend auf Attributzertifikaten
- Aufbau einer PMI ähnlich Aufbau einer PKI

- CA der PMI: Attribute Authority (AA)
  - vergibt Zugriffsrechte
  - zertifiziert diese Rechte in Form von Attributzertifikaten

- Root CA der PMI: Source of Authority (SOA)
  - Verifizierung eines Privilegs beginnt bei der SOA
  - oberste AA für dieses Privileg
  - Attributzertifikat der SOA
    - selbstzertifiziert
    - signiert mit dem zum ID-Zertifikat der SOA gehörenden privaten Schlüssel

Überblick


Motivation
Vertrauensmodelle
Infrastrukturen
PKI
PMI
Vertrauen / Vertrauensmodelle
Beispielprotokolle
X.509
OCSP/ SCVP

Vertrauen – Begriffsklärung

Was ist Vertrauen? Welche Eigenschaften hat es?

- Definition von Gambetta in „can we trust trust?“

  > “trust [..] is a particular level of the subjective probability with which an agent assesses that another agent [..] will perform a particular action, both before he can monitor such action [..] and in a context in which it affects his own action [..]”.

Vereinfacht

- A vertraut B, wenn A davon ausgehen kann, dass B sich erwartungsgemäß verhält
- A vertraut in einen Sachverhalt, wenn es von seiner Korrektheit überzeugt ist

Eigenschaften von Vertrauen

Vertrauen ist

- normal im Alltagsleben(!), im Zusammenhang mit Sicherheit in Netzen hingegen nicht
- subjektiv
- unscharf (Misstrauen < Ungewissheit < blindes Vertrauen)
- gerichtet (nicht zwangsläufig gegenseitig)
- bedingt transitiv, nimmt bei Transitivität ab
- kontextgebunden
- an Fragestellung gebunden
- Risiko-abhängig
- basiert auf Erfahrungen

Wie Vertrauen in Zertifikate entsteht, wird durch ein Vertrauensmodell beschrieben…

Vertrauensmodelle

- Ein Vertrauensmodell (Trust Model) beschreibt,
  - welchen Zertifikaten ein Benutzer trauen kann,
  - mit welchen Elementen des Modells Vertrauen hergestellt wird,
  - wie dieses Vertrauen eingeschränkt bzw. kontrolliert werden kann

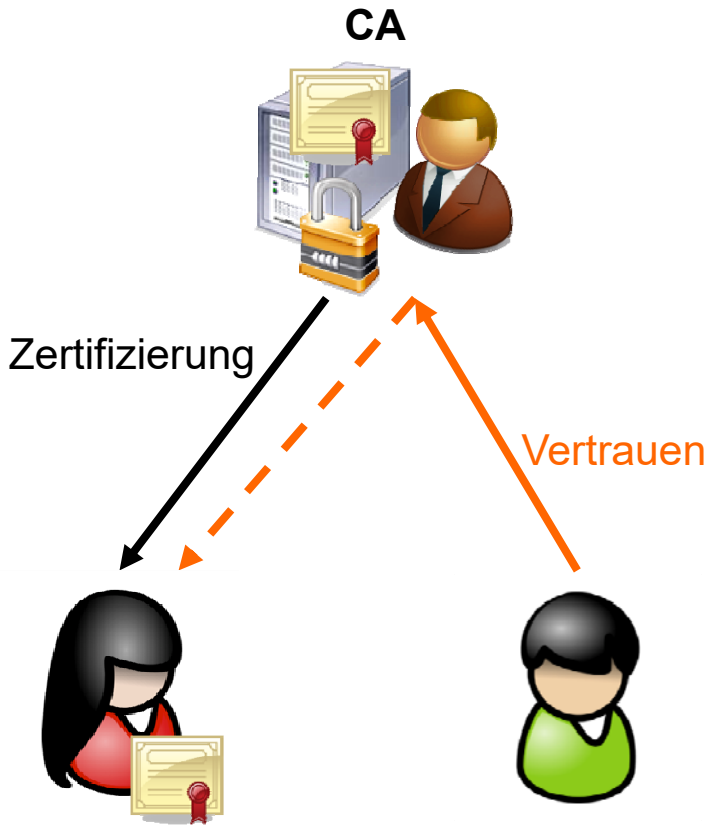
- Vertrauen basiert meist auf einem oder mehreren Vertrauensankern (Trust Anchor)
  - Ausgangspunkte für die Validierung eines Zertifikats
  - technisch gesehen: selbstsigniertes Zertifikat

- Bei erfolgreicher Validierung überträgt sich das Vertrauen in einen Vertrauensanker in den Inhalt des von ihr ausgestellten Zertifikats (in Bildern: – – →)

Modell: Single-CA

- Eine CA erstellt alle Zertifikate, d.h. ist für Registrierung und Zertifizierung zuständig

- Bewertung:
  - ☺ nur ein Vertrauensanker erleichtert Validierung
  - ☹ alle Teilnehmer müssen dieser einen CA trauen
  - ☹ Kompromittierung des CA-Schlüssels hat globale Konsequenzen
  - ☹ CA hat Monopolstellung (politischer bzw. kommerzieller Gesichtspunkt)



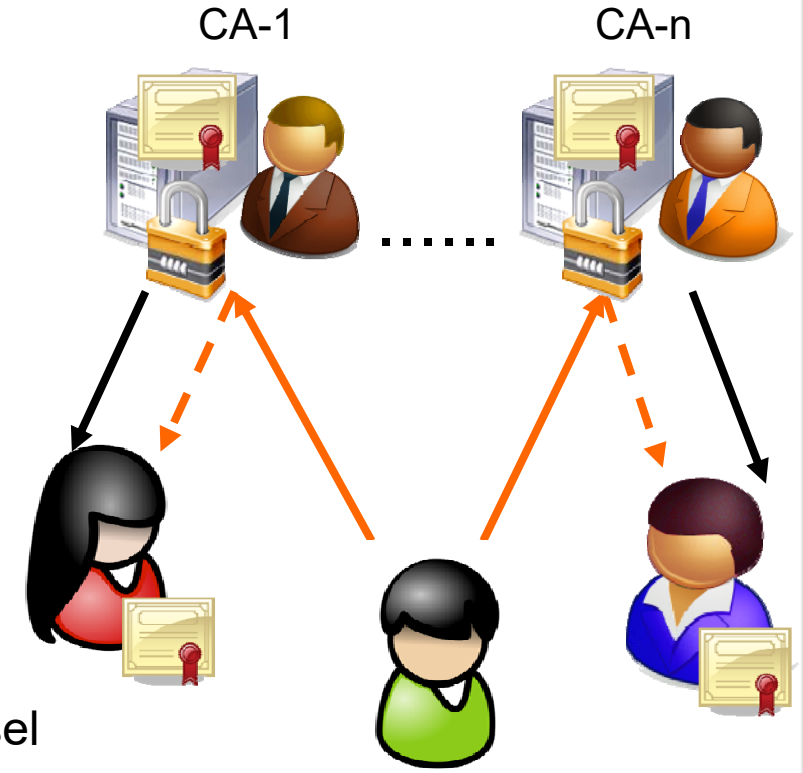
Welche Probleme gibt es bei Single-CA mit globaler Abdeckung?

Probleme Single-CA

- Welche Organisation käme aus Vertrauenssicht für die CA in Frage?
  - keine Organisation genießt global absolutes Vertrauen
- Aufwand für Zertifizierung? → Skalierbarkeitsproblem
  - Lösung evtl. Lokale „Ableger“ / Lokale Registration Authorities
    → Kontrolle der RAs?
- Qualität der Prüfung der zu verifizierenden Daten?
  - Prüfung aller weltweit vorkommenden IDs…
- Monopolstellung ohne Kontrollinstanz?
  - Wirtschaftliche Auswirkungen!
- Single-Point-of-Failure (organisatorisch)
- Der erste Versuch einer sicheren Internet-Mail-Architektur (PEM) basierte auf einer globalen Single-CA Lösung und ist unter anderem daran gescheitert

Modell: Oligarchie von CAs

- Zertifizierung durch mehrere CAs:
  → Distributed Trust Architecture



- Bewertung: Wie „Single-CA“, aber
  - ☺ keine Monopolstellung einer CA mehr, Wettbewerb
  - ☺ Kompromittierung hat begrenzte Auswirkung
  - ☹ initiale Prüfung mehrerer CA-Schlüssel
  - ☹ Validierung mittels mehrerer CA-Schlüssel
  - ☹ mehrere CA-Schlüssel müssen geschützt werden
  - ☹ falsche CA einfacher in Software implantierbar

Frage

Was ist der Vertrauensanker im World-Wide-Web für https-URLs?

Beispiel: CA-Zertifikate im Firefox

- Welche der folgenden Institutionen haben kein Zertifikat im Firefox Browser eingebaut?
  - The Go Daddy Group
  - VeriSign
  - Staat der Nederlanden
  - TURKTRUST
  - Google ← einziges Zertifikat, welches nicht im Firefox ist
  - NetLock
  - StartCom
  - Buypass
  - Quo Vadis
  - Chunghwa Telecom

Transitivität von Vertrauen

- Vertrauen
  - bisher basierten alle Modelle auf direktem Vertrauen
  - für komplexere Modelle ist jedoch transitives Vertrauen notwendig

- Frage: Was ist Transitivität von Vertrauen?
  - wenn A Vertrauen in B (und seine Zertifizierungen) hat, und B Vertrauen in C (und seine Zertifizierungen) hat, so kann A auch C (und seinen Zertifizierungen) vertrauen

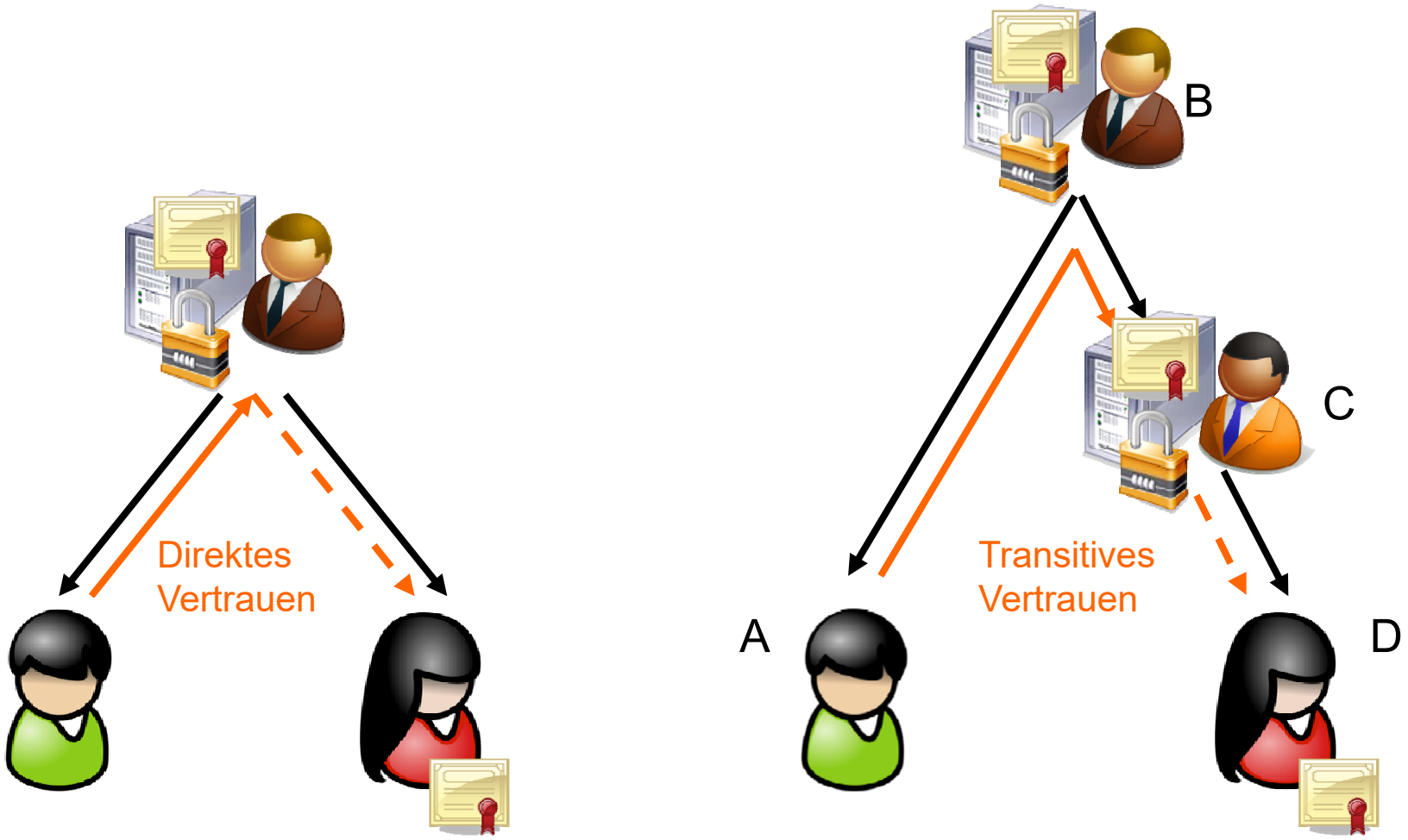
→ ist transitives Vertrauen ein sinnvolles Konzept?

→ warum bzw. warum nicht?

→ wieviele Hops würden Sie zulassen?

- Notwendig für transitives Vertrauen: Delegierung

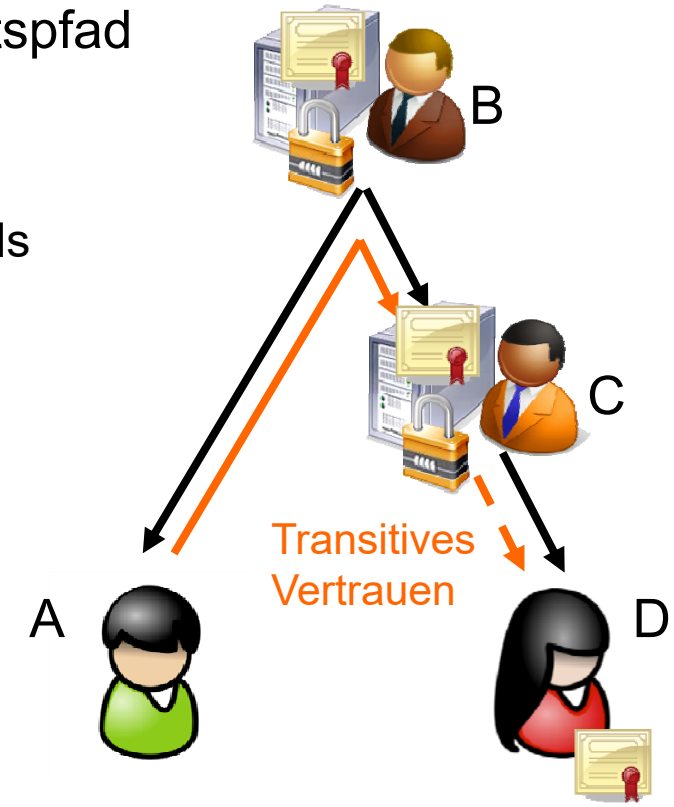
Direktes und transitives Vertrauen



Validierung bei Transitivität

Bei Transitivität spaltet sich Validierung eines Zertifikats in

- Konstruktion: Zertifikatskette bzw. Zertifikatspfad
  - Ausgangspunkt: Vertrauensanker
  - Endpunkt: zu validierendes Zertifikat
  - Aufgabe: suche nach Zertifikaten, die mittels transitivem Vertrauen einen Pfad zwischen dem Vertrauensanker und dem Endpunkt herstellen
- Validierung der Zertifikatskette
  - Prüfung der Korrektheit des Pfades (Verkettung, Delegierung)
  - Validierung jedes einzelnen Zertifikats



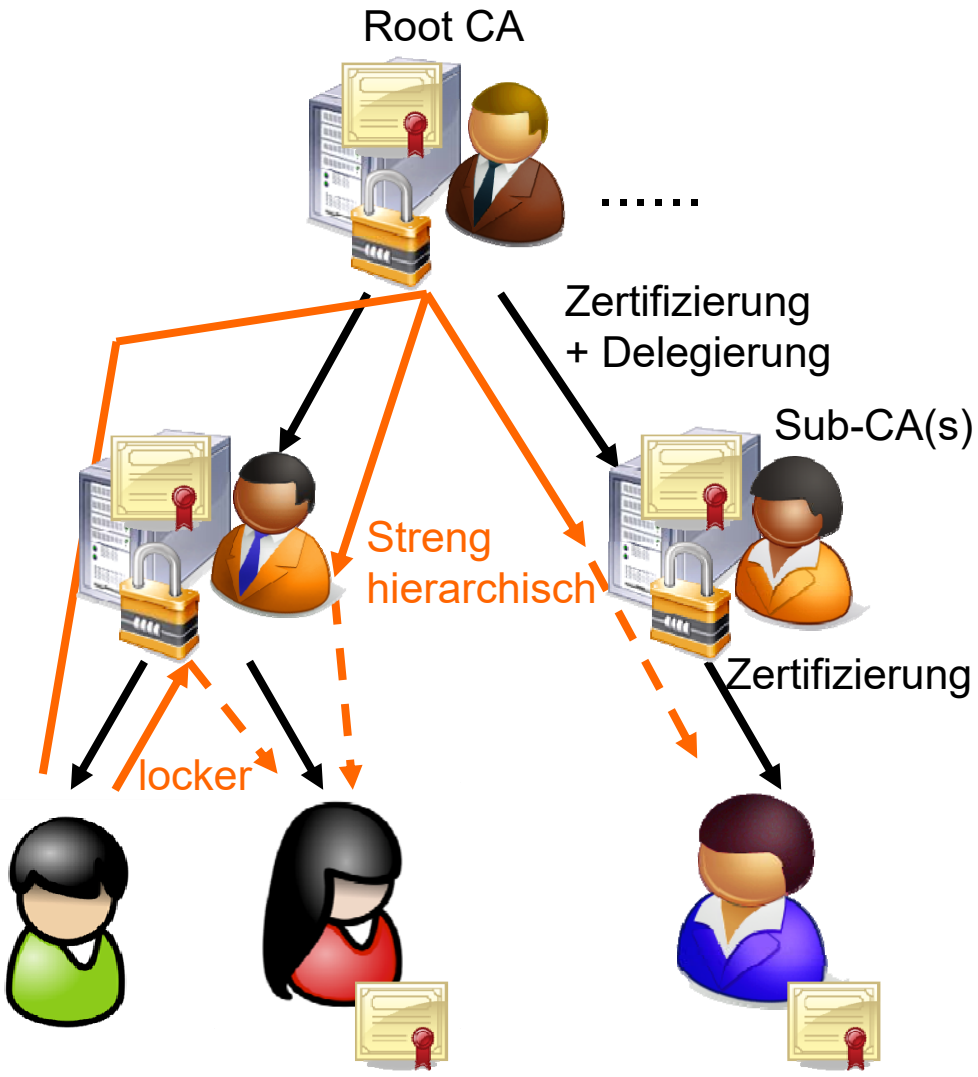
Weitere Modelle

- Bereits behandelte, nicht-transitive Modelle
  - Single-CA
  - Oligarchie von CAs
- Im Folgenden: komplexere, transitive Modelle
  - Oligarchie von CAs + Delegierung
  - Top-Down
  - Anarchie

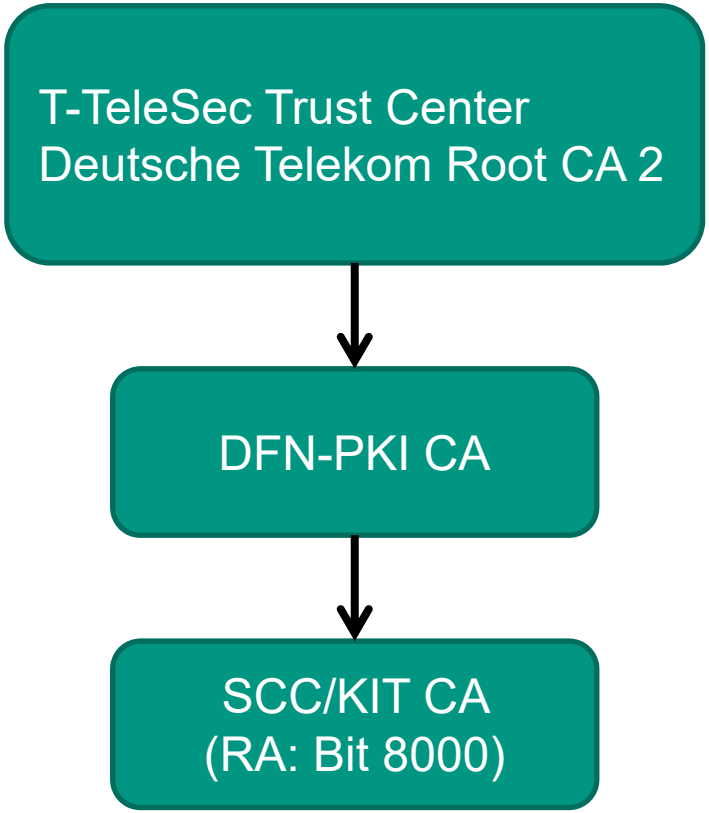
Modell: Oligarchie von CAs + Delegierung

- Delegierung: CAs können untergeordnete CAs einsetzen
- CA-Bezeichnungen
  - Root-CA: Vertrauensanker
  - Parent-CA: direkt übergeordnete CA
  - Sub-CA: untergeordnete CA

- Bewertung: wie „Oligarchie von CAs“, aber
  - ☺ Kompromittierung eines Sub-CA-Schlüssels hat beschränkteren Wirkungsbereich
  - ☺ Skalierbarkeit
  - ☹ höhere CA-Schlüsselanzahl
  - ☹ Validierung aufwändiger



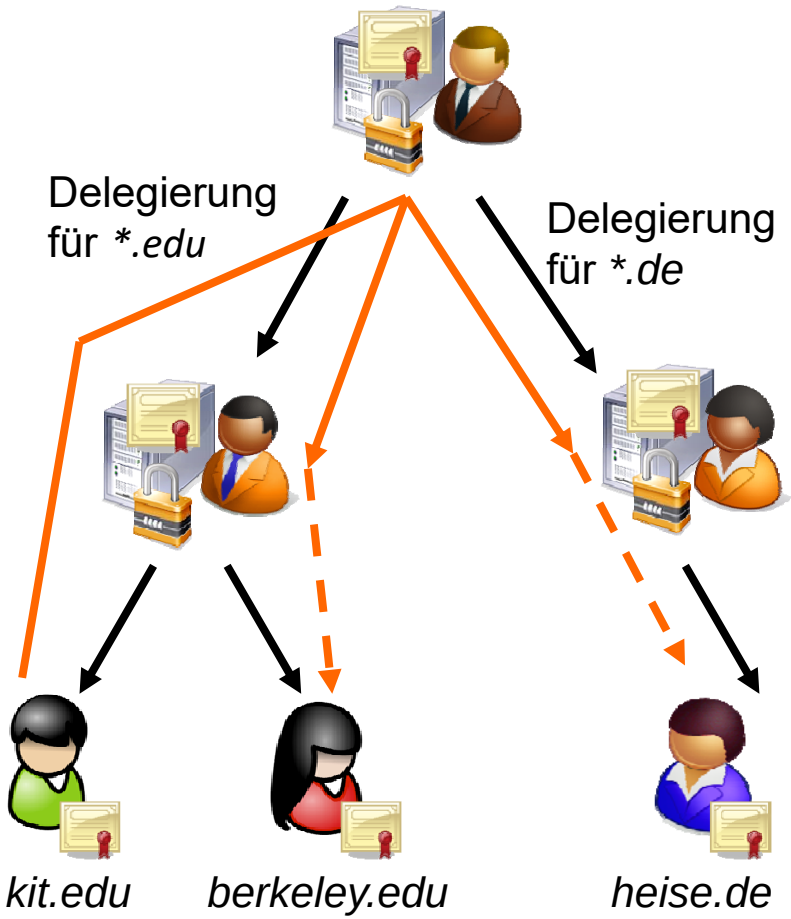
Beispiel KIT-Zertifikat Hierarchie



Modell: Top-Down

- Single-CA mit Delegation und Einschränkung der Delegierung auf Teilbereich eines hierarchischen Namensraums (Name Subordination)
  - Beispiele: DNS oder X.500

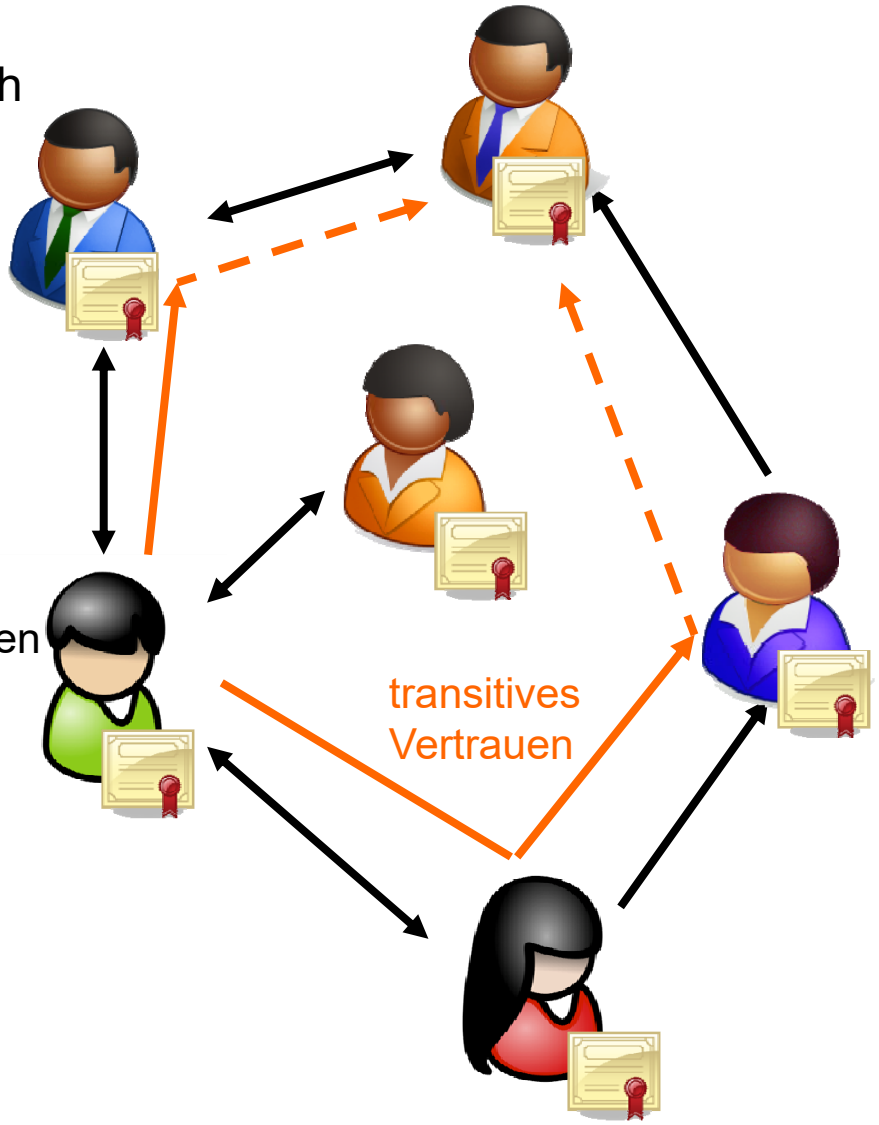
- Bewertung: wie Single-CA, aber
  - ☺/☹ Delegation (siehe letztes Modell)
  - ☺ kontrollierte Delegation
  - ☹ Validierung des ganzen Pfades (nie One-Hop)



Modell: Anarchie

Jeder Benutzer ist eine CA und kann je nach Vertrauen eingesetzt werden (auch transitiv)

- Beispiel: PGP
- Bewertung:
  - ☺ Auswirkung bei Kompromittierung beschränkt
  - ☹ alle Schlüssel sind CA-Schlüssel
  - ☹ Skalierbarkeit (hohe Anzahl von Signaturen Pfadfindung schwer, da nicht eindeutig)
  - ☹ keine einheitliche Zertifizierungspolitik, somit Transitivität von Vertrauen problematisch
  - ☹ Zertifizierungen schwer kontrollier- bzw. einschränkbar



Zusammenfassung Modelle

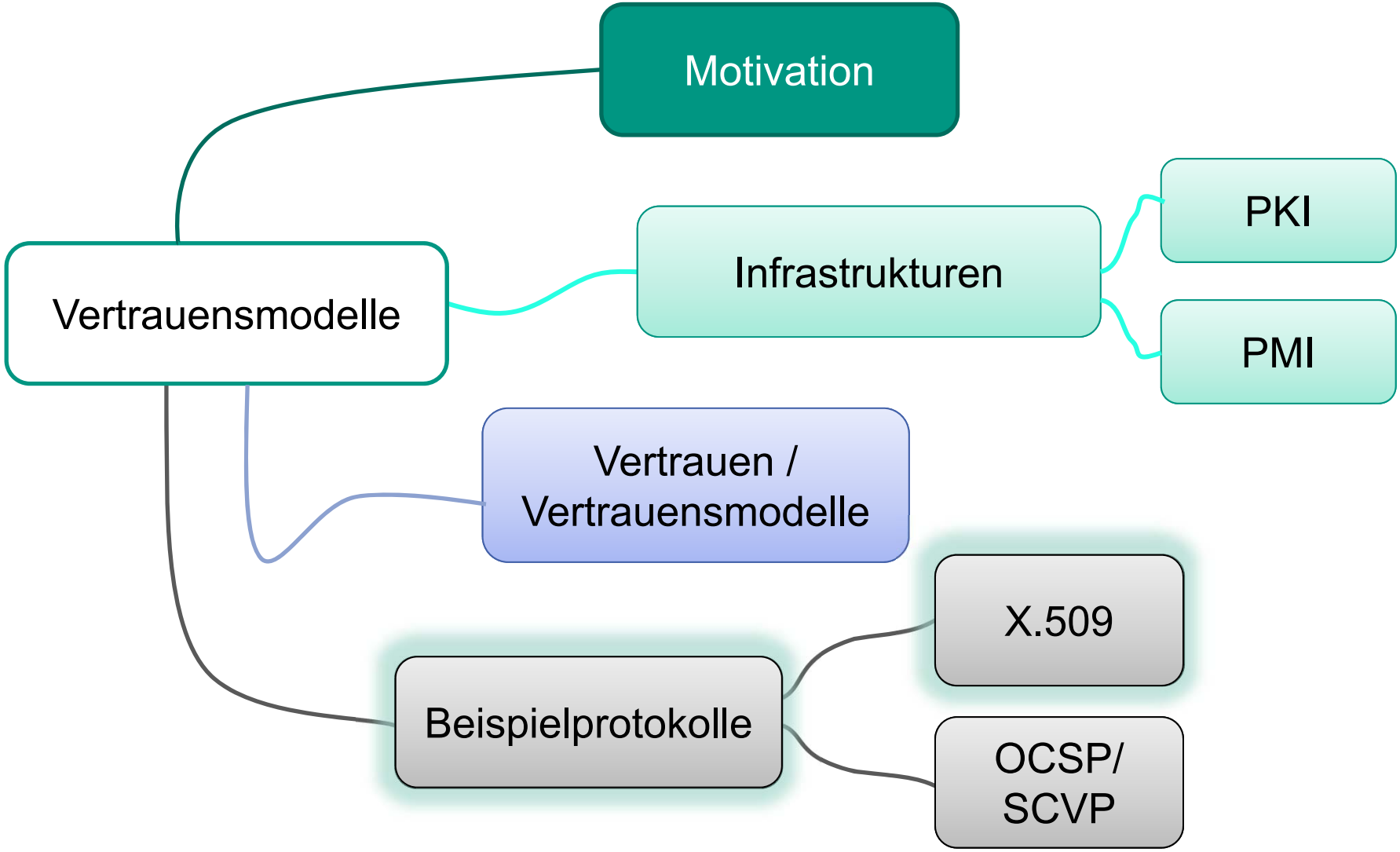
- Abschließende Bewertung
  - Single-CA nur in kleinen Umgebungen realisierbar, sonst Oligarchie
  - Skalierbarkeit durch Verteilung/Delegierung möglich
  - Bei Oligarchie kann unklar sein, welche CA die „richtige“ für ein Zertifikat ist
    → SSL: jede CA kann Zertifikat für jede Domäne ausstellen
  - Sub-CAs können auch Problem darstellen
  - Anarchie leicht einsetzbar, aber schwer kontrollierbar

- Implementierung des gewünschten Vertrauensmodells über einen technischen Zertifikats-Standard
  - **X.509**
  - **PGP**

Aus der Praxis

- Bisher präsentiert: Vertrauensmodelle
  - mit binären Vertrauensentscheidungen
    - Alice vertraut Bob voll oder gar nicht
      - ist das eine sinnvolle Modellierung?
      - lassen sich Zwischenstufen finden?
    - Abweichende Lösung: PGP-Vertrauensmodell
  - Zwischenweg: Beispiel Extended Validation SSL-Zertifikate (EV-SSL)
    - Konzept: Zwei Vertrauensstufen (normal, extended validation).
    - Aus technischer Sicht bestätigt die CA lediglich im Zertifikat, dass sie eine „erweiterte Überprüfung“ vorgenommen hat
  - die sich nur auf Identitäten beziehen
    - Wiederum: PGP-Vertrauensmodell weicht ab
  - die direkt in Zertifikaten abgebildet sind
    - Alice vertraut Bob ⇔ Alice stellt Bob Zertifikat aus

Überblick


Motivation
Vertrauensmodelle
Infrastrukturen
PKI
PMI
Vertrauen / Vertrauensmodelle
Beispielprotokolle
X.509
OCSP/ SCVP

Zertifikatsstandard X.509

- X.509: bekanntester und verbreitetster Zertifikatsstandard
  - ursprünglich Authentifizierungsmechanismus für Verzeichnis auf X.500-Basis
  - 7 Versionen, aktuell: X.509-2012
  - spezifiziert technisches Framework für eine PKI
    - Syntax eines Zertifikat in ASN.1 (aktuell: Version X.509v3)
    - Syntax einer CRL in ASN.1
    - Validierung einer Zertifikatskette
  - Vielzahl optionaler Parameter und Erweiterungen
    - auch als Attributzertifikate einsetzbar
  - eine PKI selbst
    - entscheidet welche Teile von X.509 sie nutzt → Profile
    - kann Standard durch eigene Erweiterungen ergänzen

ITU-T X.509

Aktiv genutzt von

- SSL/TLS (Schutz von TCP-Verbindungen)
  (→ Vorlesungsthema)
- S/MIME (Schutz von E-Mails)
- IPsec/IKE (Schutz von IP-Paketen)
  (→ Vorlesungsthema)
- SET (Schutz von Kreditkarten-Transaktionen)
- (W)LAN-Sicherheit: 802.1x bzw. 802.11i
  (→ Vorlesungsthema)
- S-BGP/soBGP/psBGP (Routing-Sicherheit)
  (→ Vorlesungsthema)

Globales Namensschema: Distinguished Names

- Identität des Schlüsselbesitzers ist in X.509 zentral
  - Zertifikat enthält ID der CA
  - Zertifikat enthält ID des Schlüsselbesitzers
  - ID ist Grundlage für Aufbau der Zertifikatskette

- Distinguished Name ist hierarchisches Namenschema, das sich aus mehreren Attributen zusammensetzt
  - Land (country – c)
  - Bundesland (state – s)
  - Stadt (locality – l)
  - Name der Firma/Organisation (organisation – o)
  - Abteilung (organisational unit – ou)
  - Name (common name – cn), einziges nicht optionales Attribut
  - weitere...

Aufbau des ID-Zertifikates

Ein ID-Zertifikat hat folgende Struktur

Feldname der ASN.1-Struktur	Beschreibung
`version`	Versionsnummer des Zertifikatformates
`serialNumber`	Seriennummer, zusammen mit `issuer` eindeutig
`issuer`	ID des Erzeugers des Zertifikates
`signatureAlgorithm`	Für Signatur genutzter Algorithmus
`validity`	Gültigkeitsdauer des Zertifikates
`subject`	X.500-ID des Zertifikatsbesitzers
`subjectPublicKeyinfo`	Öffentlicher Schlüssel
`issuerUniqueIdentifier`	Erweiterte ID des Zertifizierenden (v2)
`subjectUniqueIdentifier`	Erweiterte ID des Besitzers (v2)
`extensions`	Erweiterungen (v3)

Gesamte Struktur ist signiert (wird angehängt)!

Beispiel-Zertifikat: codiert ;-)

```
-----BEGIN CERTIFICATE-----
MIIFwTCCBKmgAwIBAgIHGhDPCWSUkTANBgkqhkiG9w0BAQsFADCBvzELMAkGA1UE
BhMCREUxGzAZBgNVBAgTEkJhZGVuLVd1ZXJ0dGVtYmVyZzESMBAGA1UEBxMJS2Fy
bHNydWhlMSowKAYDVQQKEyFLYXJsc3J1aGUgSW5zdGl0dXRlIG9mIFRlY2hub2xv
Z3kxJzAlBgNVBAsTHlN0ZWluYnVjaCBDZW50cmUgZm9yIENvbXB1dGluZzEPMA0G
A1UEAxMGS0lULUNBMRkwFwYJKoZIhvcNAQkBFgpjYUBraXQuZWR1MB4XDTE1MDkx
MDA5NDYzM1oXDTE4MDkwOTA5NDYzM1owXTELMAkGA1UEBhMCREUxKjAoBgNVBAoM
IUthcmxzcnVoZSBJbnN0aXR1dGUgb2YgVGVjaG5vbG9neTELMAkGA1UECwwCVE0x
FTATBgNVBAMMDFJvbGFuZCBCbGVzczCCASIwDQYJKoZIhvcNAQEBBQADggEPADCC
AQoCggEBAMqiaMlKhfamffqBMKD3qfJ5L/FKsBLVCvBnNB/M7yUDztgQtHyKp2Ux
78hQAD1PO8cf4iLyJgpFDtaIailbqAYgOrE8o/6yfqDrGg6zfkt2pJQPrlvySLSO
lKxlOpiW0JKCqpAYYI8BWl3c/TXBJi86BwiHIm7K7LkKc/+7LMuUq/OGoRO7iYPb
P/D5KVA/NXLbbiIUsnRdo3IPkjsMHi0f6EVrgxsDYObFM7ftIRDb0oObiEJTI7UD
P57om4kNesbm0KaJrBdjbMf40KpO+UQO8nHIBPM5v9FsBMNbAbU8WCK3TP1ld6uH
Gnhc+8w7AicaTiUUMG1LZMeOVu+vSxECAwEAAaOCAiEwggIdMEAGA1UdIAQ5MDcw
EQYPKwYBBAGBrSGCLAEBBAMDMBEGDysGAQQBga0hgiwCAQQDATAPBg0rBgEEAYGt
IYIsAQEEMAkGA1UdEwQCMAAwCwYDVR0PBAQDAgXgMB0GA1UdJQQWMBQGCCsGAQUF
BwMCBggrBgEFBQcDBDAdBgNVHQ4EFgQUwV2Ro6IsFWuuH1mPsdQJhbkeRuQwHwYD
VR0jBBgwFoAUH3Rl9JodevYx6d9hG3MrDW3QM0kwHwYDVR0RBBgwFoEUcm9sYW5k
LmJsZXNzQGtpdC5lZHUwdwYDVR0fBHAwbjA1oDOgMYYvaHR0cDovL2NkcDEucGNh
LmRmbi5kZS9raXQtY2EvcHViL2NybC9jYWNybC5jcmwwNaAzoDGGL2h0dHA6Ly9j
ZHAyLnBjYS5kZm4uZGUva2l0LWNhL3B1Yi9jcmwvY2FjcmwuY3JsMIHHBggrBgEF
BQcBAQSBujCBtzAzBggrBgEFBQcwAYYnaHR0cDovL29jc3AucGNhLmRmbi5kZS9P
Q1NQLVNlcnZlci9PQ1NQMD8GCCsGAQUFBzAChjNodHRwOi8vY2RwMS5wY2EuZGZu
LmRlL2tpdC1jYS9wdWIvY2FjZXJ0L2NhY2VydC5jcnQwPwYIKwYBBQUHMAKGM2h0
dHA6Ly9jZHAyLnBjYS5kZm4uZGUva2l0LWNhL3B1Yi9jYWNlcnQvY2FjZXJ0LmNy
dDANBgkqhkiG9w0BAQsFAAOCAQEAliwmjmCiRTvyYmAfAHba7RHyRuWtkvoz1P8X
JHaJqhimzmXL+u+VcyUiO5kRjYYTJPL9LJ2EkKYJQzAwOdGxAle4IzDzsqPBNHVz
qb88Q14G1Y7gNvPKwvV+3IWkdtiz9WUuvoEcg1RTYNzQ12F3d3HQ/1h0aTsh9IXH
DLzdDmTdYDUUFqVZIKDSEtfOnASgbW0MoaNbbUqSmC+8v1SRubIGA/OFGSteudBb
AvIRupaP3p6AfZ3Wpjnt08RJLRHkIaQ2m9EDStD/jilYmKeoYzD54Kd0CeRcMFuT
tU+jq1GZwWcx/PU/4MFjXI2yjD+mt4hRtCy4i1P29lL3fj3BXw==
-----END CERTIFICATE-----
```

X.509 Beispielzertifikat – dekodiert (1/2)

```
Version: 3 (0x2)
Serial Number: 733683079633832l (0x1a10cf09649491)
Signature Algorithm: sha256WithRSAEncryption
Issuer: C=DE, ST=Baden-Wuerttemberg, L=Karlsruhe, O=Karlsruhe
Institute of Technology, OU=Steinbuch Centre for Computing, CN=KIT-
CA/emailAddress=ca@kit.edu
Validity
            Not Before: Sep 10 09:46:33 2015 GMT
            Not After : Sep  9 09:46:33 2018 GMT
Subject: C=DE, O=Karlsruhe Institute of Technology, CN=Roland Bless
Subject Public Key Info: Public Key Algorithm: rsaEncryption
                    Public-Key: (2048 bit)
                    Modulus:
                    00:ca:a2:68:c9:4a:85:f6:a6:7d:fa:81:30:a0:f7
                    …4b:11
                Exponent: 65537 (0x10001)
X509v3 extensions:
X509v3 Certificate Policies:
       Policy: 1.3.6.1.4.1.22177.300.1.1.4.3.3
       Policy: 1.3.6.1.4.1.22177.300.2.1.4.3.1
```

… (Fortsetzung nächste Folie)

X.509 Beispielzertifikat – dekodiert (2/2)

```
X509v3 Basic Constraints: CA:FALSE
X509v3 Key Usage: Digital Signature, Non Repudiation, Key
Encipherment
X509v3 Extended Key Usage: TLS Web Client Authentication, E-mail
Protection
X509v3 Subject Key Identifier:
 C1:5D:91:A3:A2:2C:15:6B:AE:1F:59:8F:B1:D4:09:85:B9:1E:46:E4
X509v3 Authority Key Identifier:
 keyid:1F:74:65:F4:9A:1D:7A:F6:31:E9:DF:61:1B:73:2B:0D:6D:D0:33:49
X509v3 Subject Alternative Name: email:roland.bless@kit.edu
X509v3 CRL Distribution Points:
Full Name: URI:http://cdp1.pca.dfn.de/kit-ca/pub/crl/cacrl.crl
Full Name: URI:http://cdp2.pca.dfn.de/kit-ca/pub/crl/cacrl.crl
 Authority Information Access:
OCSP - URI:http://ocsp.pca.dfn.de/OCSP-Server/OCSP
CA Issuers - URI:http://cdp1.pca.dfn.de/kit-ca/pub/cacert/cacert.crt
CA Issuers - URI:http://cdp2.pca.dfn.de/kit-ca/pub/cacert/cacert.crt
Signature Algorithm: sha256WithRSAEncryption
         96:2c:26:8e:60:a2:45:3b:f2:62:60:1f:00:76:da:ed:11:f2:
         …
         b8:8b:53:f6:f6:52:f7:7e:3d:c1:5f
```

Wichtige Erweiterungen

- Verwendung mehrerer Schlüssel
  - Erweiterung: Authority key identifier, Subject key identifier
    - Standardfelder enthalten nur Aussteller des Zertifikats
    - Identifizierung des Aussteller- bzw. Inhaber-Schlüssels
    - erlaubt Verwendung mehrerer Schlüssel und Zertifikate
    - vereinfacht Pfad-Konstruktion
  - Erweiterung: Key usage, Extended key usage
  - verschiedene Verwendungszwecke der Schlüssel
    - z.B. Zertifizierung nicht mit jedem Schlüssel erlaubt
    - unterschiedliche Policies der Zertifizierung
    - Key usage: z.B. digitalSignature, keyCertSign, cRLSign, encipher-only
    - Extended Key usage: z.B. clientAuth, EmailProtection
- Problem: X.500-Namen wenig verbreitet
  - Erweiterung: Subject-Issuer Alternative Name
  - alternative Namensformen mit dem Zertifikat assoziierbar
  - Beispiele: eMail-Adresse, IP-Adresse, Domain-Name, URI, AS-Nummer
  - jedes strukturiertes Namensschema möglich

Aus der Praxis: Extended Validation-Zertifikate

- X.509-Standard kennt nur zwei Vertrauensstufen
  - Vertrauen besteht / besteht nicht
- Wunsch aus der Praxis: höheres Maß an Vertrauen für bestimmte Anwendungen → Erweiterte Überprüfung
  - Online-Banking u.ä.
- Definition einer Richtlinie mit technischen und organisatorischen Anforderungen
  - insbesondere an Durchführung der Identitätsprüfung durch die CA
  - Richtlinie wurden durch das CA/Browser-Forum definiert
    - https://cabforum.org/ev-code-signing-certificate-guidelines/
- Zertifikate enthalten certificatePolicies-Erweiterung
  - im Wesentlichen enthalten
    - OID der Policy der CA für EV-Zertifikate
    - URL des Certification Practice Statement
- Prüfende Instanz kennt OIDs der EV-Policies vertrauter CAs
- Ergebnis: weitere Vertrauensstufe
  - sollte dem Nutzer deutlich angezeigt werden

PKI-Unfälle (1): Diginotar, Comodo, TürkTrust

- Diginotar  [Heis11a, Heis11b]
  - 2011 Einbruch bei CA durch Hacker
  - Eindringen auf 8 CA-Server, obwohl diese vom Internet „getrennt“ sind
  - Ausstellen von Zertifikaten für google.com, microsoft.com, skype.com
  - Ziel: vermutlich Man-in-the-Middle-Angriffe durch iranische Regierung, um Bürger auszuspionieren
  - Diginotar wurde liquidiert
- Comodo 2011  [Heis11c]
  - Einbruch bei InstantSSL.it, einer RA der Comodo CA
  - Ausstellen von Zertifikaten für Microsoft, Google, Yahoo, Mozilla, Skype
- TürkTrust 2011/2012  [Heis13]
  - hat Sub-CA-Zertifikate ausgestellt
  - Ausstellen eines Wildcard-Zertifikats für *.google.com
  - Einsatz auf einer Firewall-Appliance

→ generelles Problem: jede CA kann Zertifikate für beliebige Domänen ausstellen! Welche CA ist die „richtige“?

PKI-Unfälle (2): Microsoft und Verisign

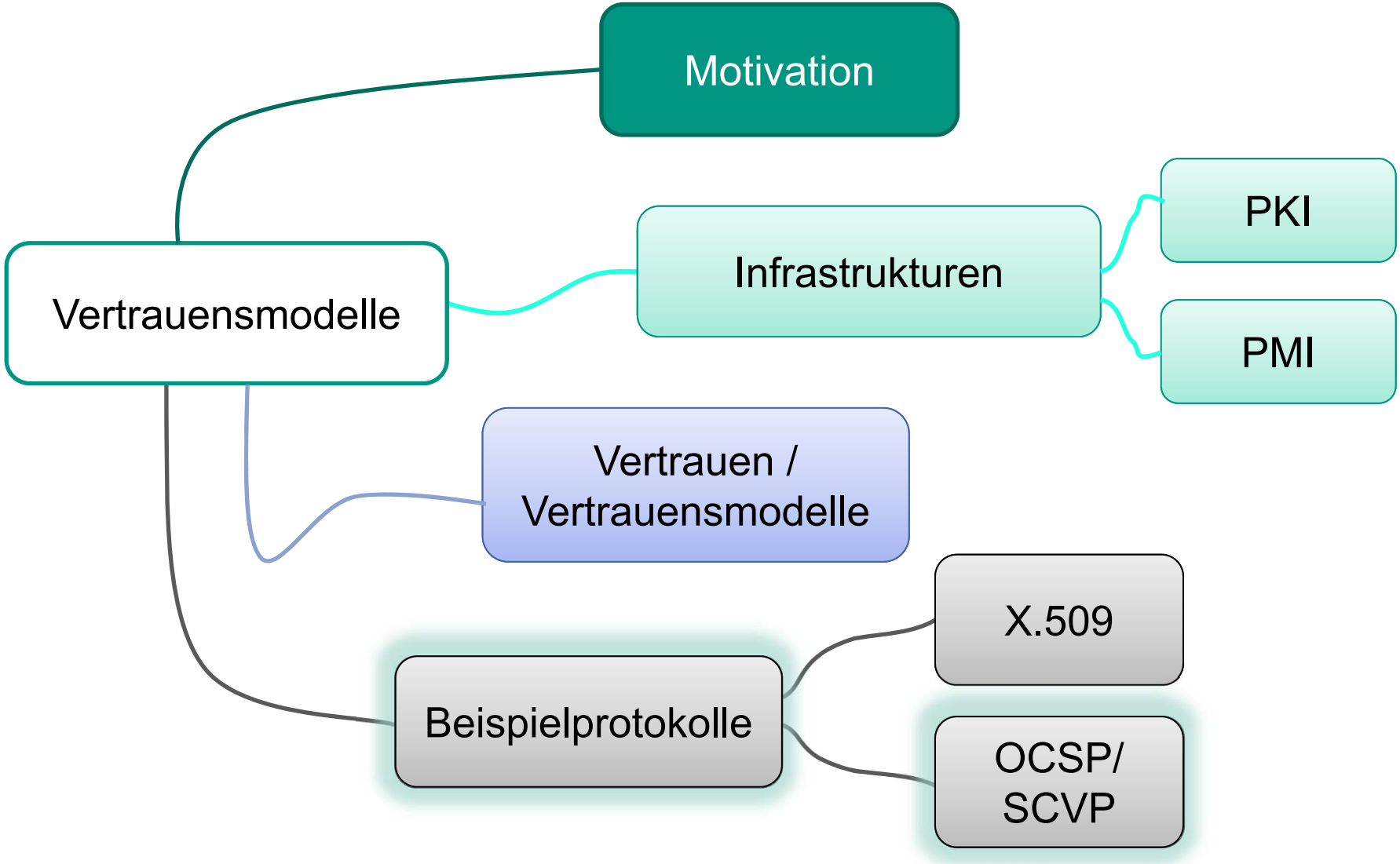
→ Schwächstes Glied der Kette bestimmt Gesamtsicherheit

- Verisign
  - stellte Code-Signing-Zertifikate für eine Firma Microsoft aus
  - Routine-Check stellte fest, dass die Zertifikate fälschlicherweise ausgestellt wurden
  - Zertifikate wurden zurückgezogen, via CRL bekannt gegeben
- Microsoft-OSe erfuhren von dem Widerruf nichts
  - Windows enthält die CA-Zertifikate von Verisign
  - Verisign-Zertifikate enthalten keine Erweiterung CrlDistributionPoint (weil Verisign-PKI schon älter als X.509v3)
  - CRL ist unter bekannter und dokumentierter URL zu finden, die jedoch durch Windows nicht genutzt wurde

→ Ergebnis: Windows kann einige Verisign-Zertifikate nicht auf Widerruf prüfen!

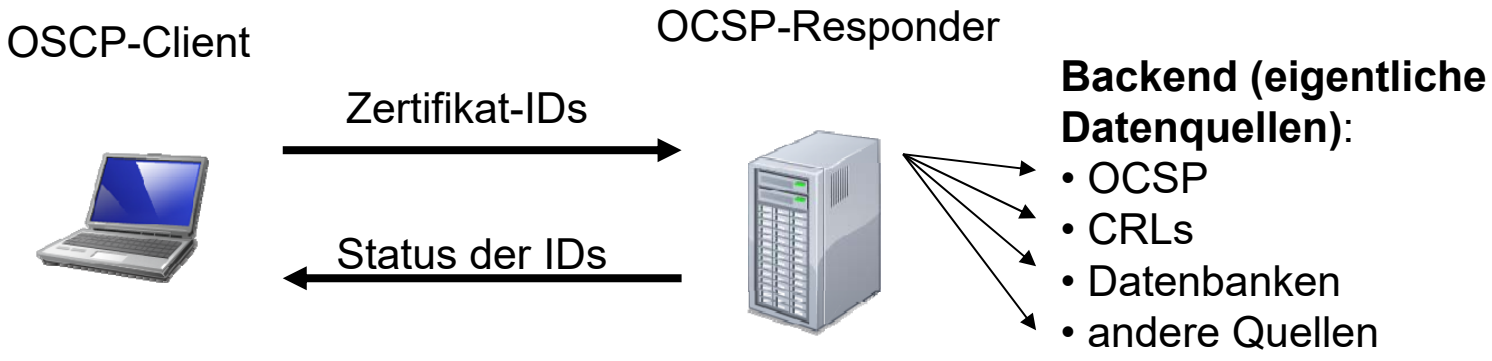

[Heis01]

Überblick


Motivation
Vertrauensmodelle
Infrastrukturen
PKI
PMI
Vertrauen / Vertrauensmodelle
Beispielprotokolle
X.509
OCSP/ SCVP

OCSP: Online Certificate Status Protocol

- OCSP war der erste Ansatz eines Protokolls zur Online-Prüfung von Zertifikaten auf Widerruf
  - einfaches Frage-Antwort-Schema
  - Erweiterung im Zertifikat: `AuthorityInfoAccess`
    - u.a. Lokalisierung von OSCP-Respondern und das verwendete Protokoll (z.B. http, ldap, ...)
  - durch die `extendedKeyUsage`-Erweiterung (Wert: OCSPSigning) autorisiert die CA den Responder zur Signatur von Antworten



Aufgaben und Einschränkungen von OCSP

- OCSP antwortet nur in Bezug auf Widerruf, prüft nicht
  - zeitliche Gültigkeit des Zertifikates
  - korrekter Verwendungszweck der Zertifikate
- On-line vs. Up-to-date
  - Unterschied?
- Signieren der Daten gefährdet Skalierbarkeit
  - CPU-Bedarf fällt synchron und bei jeder Anfrage an, nicht wie bei einer CRL einmalig
  - evtl. Vorbereiten von Antworten, wenn Aktualität ausreichend
- Nur geringe Verringerung der Komplexität der Validierung auf Client-Seite
  - Konstruktion der Zertifikatskette bleibt
  - Validierung der Zertifikatskette bis auf Widerrufprüfung bleibt

  } **siehe SCVP**
- Angreifer kann OCSP-Abfragen ggf. blockieren!

SCVP: Server-based Certificate Validation Protocol

- SCVP soll Client ein partielles bis vollständiges Auslagern der Zertifikat-Validierung ermöglichen
- Konstruktion einer Zertifikatskette (Delegated Path Discovery)
- Konstruktion und Validierung einer Zertifikatskette (Delegated Path Validation)

- Fokus auf zwei Klassen von Benutzern
  - Auslagerung der Konstruktion der Zertifikatskette, Validierung wird selbst durchgeführt
  - → kein vertrauenswürdiger Server erforderlich
  - Server kann auch CRLs bzw. OCSP-Antworten bereitstellen

  - Vollständige Auslagerung, nur Ergebnis ist interessant
    → vertrauenswürdiger Server erforderlich

[RFC5055]

Zusammenfassung

- Zertifikate für unterschiedliche Zwecke befinden sich im breiten Einsatz
  - X.509 als weit verbreitetes Format
  - CRLs sind oft ein Knackpunkt bei der Zertifikatsicherheit
- PKIs sind komplex, aber auch verwundbar → PKI-Vorfälle
  - Oligarchie von CAs macht es nicht einfacher zu vertrauen: jede CA kann Zertifikate für jede Domäne ausstellen
  - Welche die „richtige CA“ für ein Zertifikat ist, kann man mittels DANE (DNS-based Authentication of Named Entities) versuchen herauszufinden
    - Nutzt DNSsec (Top-Down), um X.509-Zertifikatsinfo zu hinterlegen
  - Liste von Root-CAs in Browser/OS dienen als „Vertrauensanker“
- Es gibt die Möglichkeit SSL-Verbindungen durch Zwischenschalten mit „passenden“ Zertifikaten abzuhören/zu manipulieren
  - wird gerne von Firewalls für Deep Packet Inspection verwendet
  - CAs stellen dann „Wildcard-Zertifikate“ aus „*.google.com“

Literatur

Historisch

[Kohn78] Loren Kohnfelder: Towards a practical public-key cryptosystem. Bachelor Thesis, MIT, Cambridge, 1978.

Buch

[AdLo03] Carlisle Adams, Steve Lloyd: Understanding PKI, Addison Wesley, 2003

Artikel

[Perl99] R. Perlman: *An Overview of PKI Trust Models*, IEEE Network 13(6):38-43, 1999.

- Wie der Titel sagt: Überblick über (theoretische und praktisch eingesetzte) Vertrauensmodelle

Online Quellen:

- PKI-Forum: CA-CA Interoperability; 2001
  - guter Überblick über das Thema, dabei recht kurz gefasst
  - http://www.oasis-pki.org/pdfs/ca-ca_interop.pdf
- The Open Source PKI Book http://ospkibook.sourceforge.net/

Literatur

[KaPe03] C. Kaufmann, R. Perlman, M. Speciner; Network Security – Private Communication in a public world; Prentice Hall; 2003

[X.509] ITU-T Recommendation X.509, 2000.URL: http://www.itu.int/ITU-T/asn1/database/itu-t/x/x509/2005/index.html

[FeSc03] N. Ferguson, B. Schneier: Practical Cryptography, Wiley, 2003.

[ScEl00] B. Schneier, C. Ellison: Ten Risks of PKI: What You're not Being Told about Public Key Infrastructure, Computer Security Journal 16 (1), S. 1-7, 2000.

[Elli02] C. Ellison: Improvements on Conventional PKI Wisdom, Proceedings of the 1st Annual PKI Research Workshop, online verfügbar: http://www.cs.dartmouth.edu/~pki02/

[Gutm02] P. Gutmann: PKI: it's not dead, just resting, IEEE Computer 35 (8), S. 41-49, August 2002.

[Gutm00] P. Gutmann: X.509 Style Guide, 2000. URL http://www.cs.auckland.ac.nz/~pgut001/pubs/x509guide.txt

[Heis01] Microsoft warnt vor Cracker-Zertifikat, Meldung im Heise-Newsticker vom 24.3.2001, URL http://www.heise.de/newsticker/meldung/16482

[Heis11a] Neuer SSL-Gau: Falsches Google-Zertifikat blieb fünf Wochen unentdeckt, Meldung im Heise-Newsticker vom 30.8.2011, http://heise.de/-1333070

[Heis11b] SSL-GAU zwingt Browser-Hersteller zu Updates, Meldung im Heise-Newsticker vom 23.3.2011, http://heise.de/-1212986

[Heis11c] Zwei weitere Comodo-SSL-Registrare gehackt, Meldung im Heise-Newsticker vom 31.03.2011, http://heise.de/-1219420

[Heis13] Fatale Panne bei Zertifikatsherausgeber Türktrust, Meldung im Heise-Newsticker vom 04.01.2013, http://heise.de/-1776879

Literatur

[RFC 2560] M. Myers, R. Ankney, A. Malpani, S. Galperin und C. Adams. X.509 Internet Public Key Infrastructure Online Certificate Status Protocol - OCSP. RFC 2560 (Proposed Standard), Juni 1999. URL: http://www.ietf.org/rfc/rfc2560.txt

[RFC 3161] C. Adams, P. Cain, D. Pinkas und R. Zuccherato. Internet X.509 Public Key Infrastructure Time-Stamp Protocol (TSP). RFC 3161 (Proposed Standard), August 2001. Updated by RFC 5816. URL: http://www.ietf.org/rfc/rfc3161.txt

[RFC 5055] T. Freeman, R. Housley, A. Malpani, D. Cooper und W. Polk. Server-Based Certificate Validation Protocol (SCVP). RFC 5055 (Proposed Standard), Dezember 2007. URL: http://www.ietf.org/rfc/rfc5055.txt