Repetition

- Wie viel Inhalt können mit einem 1024-Bit Schlüssel übermittelt werden? 1024-Bit.
- Ist das Padding normiert? Ja, in einem RFC
- Grösse Integer? 32- / 64-Bit begrenzt durch Architektur, BigInteger :arrow_right: Spezialbehandlung
- Was ist ein qualifiziertes Zertifikat? Personenzertifikat

Padding

Ist der Text kürzer als der Schlüssel braucht es immer zwingend ein Padding. Ansonsten ist die Sicherheit der Nachricht gefährdet (Füllzeichen).

$x = a + bx + cy$
$a^x * a^y = a^{(x+y)}$
$a^{a_0} * a^{bx} * a^{cy} = a^{a_0+bx+cy} = a^x$
$a_0 = 0 \implies a^{a_0} = 0$
$(xy)^d \equiv x^d * y^d modn$
$C_1....C_2$
$C = \prod C(M_i)^{e^i} modn \rightarrow M = \prod M_i^{e_i} modn$

ASN.1 [Abstract Syntax Notation No. 1]

Wichtig: - Basic Encoding Rules (BER) - Distinguished Encoding Rules (DER)

Beschreibungsstruktur

Module

Modulname DEFINITIONS::=BEGIN EXPORTS export liste IMPORTS imports ....

www.oid-info.com

Sequence: 30h –> Tag: 0011 0000 Universal: 00 Zusammengesetzt: 1 Sequence: 16

Tag-Value: 30-0B-03-03-00-0F-C1-....

Sequence of Seuquence: 30 - L - 30 - ....

Objekt-Identifier

x x 40 + y 1x40 + 2 = 42

Zertifikate

Erzeugung

- Werden oft bei CA erstellt.

Ablauf:
1. Person geht zur RA für die initiale Registrierung (Intermidär).
2. RA bestätigt Identität und leitet den Request der CA weiter.
3. ....
x. PKCS#12-Container geht zurück an die Person (geschützt mit PIN)

Aufbau

ID + Signatur $sS_{p_{CA}}(ID)$ ObjectIdentifier :arrow_right: Signaturtyp + Algorithmus

Subject: Enduser oder Sub-CA

X.509 Zertifikate nach RFC 5280

Bestandteil X.500-Standards (Verzeichnisdienste), nicht immer optimale Lösungen, da bereits alt, Authentifizierungsstandard für Kommunikationsnetze, 3. Teil des Standards: Formate für digitale Zertifikate

Repetition

Hauptbestandteile Zertifikat: - tbs - AlgID - Sig ($ sS_a(tbs) $)

Erweiterungen / Extensions: Standard / Privat

Auflösung Zertifikatspfad: Via AuthorityKeyIdentifier

CA Zertifikat: Basic constraints, critical, Key Usage: critical (True, sonst immer false)

Dokumentunterschreibung: Key Usage: Content commitment

Attribut CRL: CRLDistributionPoint

Qualifiziertes zertifikat: Nur für natürliche Personen

Attributzertifikat

Erweiterung Zertifikat um weitere Attribute (z.B. auch zeitbegrenzt). Keine Identifikation, sondern Autorisierung

Analog zu SAML im Web-Bereich

Attribute müssen als OID hinterlegt sein

ASN1

objectDigestInfo:objectDigest: Hashwert Identifikationszertifikat

Sperrlisten (CRL)

crlDistributionPoint: URL für Sperrliste
Key-Attribut für Revokation: Seriennummer
crlEntryExtension: Bezogen auf jedes Zertifikat
crlExtensions: Extensions für CRL selbst CRL: Nicht notwendig, wenn normal abgelaufen CRL: Nur noch für kleinere CAs oder Sub-CAs, OCSP stattdessen

Verzeichnisdienst (OCSP)

RFC2560, via HTTP oder LDAP, Request-Signatur: optional

TBSRequest (n Requests) * Request * CertID * hashAlgorithm * issuernamehash * issuerkeyhash ("AuthorityKeyIdentifier", Hash Public key von Issuer) * serialnumber * singleRequestExtension*

OCSPResponse Immer signiert durch OCSP-Dienst

- SingleResponse
  - CertID
  - hashAlgorithm
  - issuernamehash
  - issuerkeyHash
  - serialnumber
  - certStatus
  - thisUpdate
  - singleExtensions*
- OCSPResponse (nicht signiert)
  - responseStatus (nicht optional, immer)

  - responseBytes (optional, Anfrage falsch, falsches Cert, Inhalt nur wenn Anfrage beantwortbar) :arrow_right: BasicOCSPResponse
- BasicOCSPResponse
  - tbsResponseData (signiert)

XCA

ZHAW-Root-CA - ZHAW-Sub-CA

Root-CA

1. XCA starte
    1. DB alege. -> Kes Passwort setze
    2. Certificates
    3. New Certificate
        1. Create a self signed certificate with the serial (self signed bedeutet es ist ein Root Zertifikat
        2. Signature algorithm: SHA 1
        3. Subject
        4. Internal name: ZHAW-Root-CA
        5. countryName: CH
        6. stateOrProvinceName: Zuerich
        7. localityName: Zuerich
        8. organizationName: ZHAW
        9. organizationUnitName: Certificate Services
        10. commonName: zhaw-root-ca
        11. Generate a new key
        12. Create
        13. Key usage
        14. Certificate Sign
        15. CRL Sign
        16. Key usage -> Critical
        17. Extensions
        18. Type -> Certification Authority
        19. Path length: 2
            - Leer = Unendlich lang
        20. Critical
        21. Subject Key Identifier
        22. Authority Key Identifier

23. Not after: 10 Jahre gültig
24. CRL distribution point -> Edit -> Add -> Content setzen auf: http://crl.zhaw-CA.ch (gibt es aber nicht) -> Apply
25. Finales OK

Sub-CA

- use this Certificate for signing: ZHAW-Root-CA
- Subject
  - Generate a new key
  - Ausfüllen
- Key usage
  - Critical
  - Certificate sign
  - CRL sign
- Extended key usage
  - Not critical
  - OCSP signing
- Extensions
  - Type Authority: Certification Authority
  - path length: 1, critical
  - subject key Identifier, authority key Identifier
  - Time range: kleiner als Root (5)
  - CRL distribution point (URI:http://crl.zhaw.ch/subca.crl)
  - OCSP (URI:http://ocsp.zhaw.ch/)
- Export DER

Enduser Zertifikat (EE-Cert)

- New Certificate
  - Use this Certificate for signing: Sub-CA
  - SHA1
  - HTTP_client
  - Generate a new key
  - Subject (ZHAW-Mail)
  - Key usage:
  - Digital Signature
  - Key Encipherment

- Data Encipherment
- Extended key usage:
- E-mail Protection
- Extensions
- End Entity
- Path length (leer)
- Subject Key Identifier
- Authority key identifier
- Time Range: 1 Year
- Subject alternative name: email:brundan1@students.zhaw.ch
- CRL distribution point (URI:http://crl.zhaw.ch/subca.crl)
- OCSP (URI:http://ocsp.zhaw.ch/)
- Export DER, PKCS12 (Private, Public, PW:DaniBrun), PKCS7 (with Chain, Public)