E-Mail

Verschlüsselung / Signatur von E-Mail

Bei der E-Mail Verschlüsselung und Signatur unterscheidet man Client-basierende und Server-basierende Verschlüsselung.
Klassische Lösungen arbeiten Client-basiert. Das ist für Unternehmen zu unhandlich.

Zu den Unterschieden zwischen Transportverschlüsselung (TLS/SSL) und Inhaltsverschlüsselung (S/MIME, PGP..) siehe diesen Beitrag.

Die Inhalts-Verschlüsselung kann durch Key (Zertifikate) oder Passwort erfolgen.
Passwort-Verschlüsselung ist eine Notlösung, die keine Zertifikatsinfrastruktur erfordert und mit jedem Empfänger funktioniert.
Das ist lösbar mit automatischen Portallösungen oder ganz einfach mit verschlüsselten Anlagen im .ZIP oder .PDF-Format.

S/MIME

E-Mails werden mit dem öffentlichen Zertifikat des Empfängers verschlüsselt oder signiert.
Nur er hat den privaten Key und kann die E-Mails entschlüsseln.
Bei S/MIME werden die X.509-Zertifikate (SSL-Zertifikate) durch eine zentrale Zertifikatsstelle geprüft, signiert und verwaltet.
Zertifikate haben eine begrenzte Laufzeit. Alte Zertifikate sollten unbedingt aufbewahrt bzw. im System installiert bleiben, weil sonst alte verschlüsselte E-Mails nicht mehr geöffnet werden können.

Bei E-Mail-Signaturen benötigt jeder User ein **User-Zertifikat** nach dem Schema: user@domain.tld.
Gateway-Zertifikate bzw. **Domain-Zertifikate** sind gültig für alle E-Mail-Adressen unterhalb einer E-Mail-Domäne (@unternehmen.de).
Team-Zertifikate stellt man für Mailadressen aus, die nicht von einzelnen Personen verwaltet werden, wie z. B. info@unternehmen.de oder bewerbung@unternehmen.de.

Dateinamen und Content-Typen:
smime.p7m (signierte oder verschlüsselte Daten), smime.p7c (Zertifikat), smime.p7s (Signatur)

Formate:

PFX/P12/PKCS#12 (Personal Information Exchange Standard)

- PKCS#12 oder PFX/P12 stellt ein binäres Format für die Aufbewahrung des Zertifikats (samt seinem Intermediate) mit dem privaten Schlüssel dar. Die Zertifikate und der private Schlüssel sind in der PFX-Datei mit einem Passwort geschützt. PFX war der Vorgänger von PKCS#12.
- Die meistgenutzte Endung des Formates ist **.pfx** und **.p12**.
- PKCS#12 wird oft auf den Windows-Geräten für das Import und Export der Zertifikate zusammen mit dem privaten Schlüssel genutzt.

- Die in PFX gespeicherten Zertifikate dienen auch zum Signieren in Microsoft Authenticode.

  Format P7B/PKCS#7
- Das Format PKKS#7 oder P7B repräsentiert ein oder mehrere Zertifikate im Base64 ASCII-Format, das in einer Datei mit der Endung **.p7b** oder **.p7c** gespeichert ist.
- Die Datei P7B enthält das Zertifikat und seine Kette (die Intermediate-Zertifikate), aber der private Schlüssel ist in ihr nicht vorhanden.
- Am häufigsten werden die P7B Dateien unter Windows und auf der Plattform Java Tomcat eingesetzt.

  .DER-Binärformat (KEIN Textformat) zum Import in Anwendungen (Mail, Broser..)
- In DER können alle Zertifikatstypen und der private Schlüssel gespeichert sein.
- Die Endung der DER-Zertifikate ist meistens **.cer** oder **.der**.
- Private Schlüssel oder der Zertifizierungspfad können mit diesem Format nicht gespeichert werden.
- Das DER-Format wird auf den Java Plattformen genutzt.

  .PEM - ASCII-Format
- ein in Base64 codiertes Format mit ASCII-Zeichen.
- Es kann neben dem reinen Zertifikat auch Intermediate-Zertifikate, Root-CAs und private Schlüssel beinhalten.
- Die Dateierweiterung **.pem** kommt meist zum Einsatz, wenn sowohl Zertifikate und der Privatschlüssel in einer Datei gespeichert werden.
- Für PEM- Zertifikate werden auch die Endungen **.cer**, **.cert**, **.crt** oder **.key** (für den privaten Schlüssel) genutzt.
- Von diesem Format gehen Apache und alle Server auf Unix/Linux OS aus.

  CSR
- Ein Certificate Signing Request (deutsch: Zertifikatsignierungsanforderung) ist ein standardisiertes Format (PKCS#10) zum Anfordern eines digitalen Zertifikats.
- CSR enthält den öffentlichen Schlüssel und weiteren Angaben über den Antragsteller des Zertifikats.
- Die Zertifizierungsanfrage kann anschließend von einer Zertifizierungsstelle (CA) signiert werden und man erhält ein digitales Zertifikat zurück.

Zertifikatstypen:
https://serverfault.com/questions/9708/what-is-a-pem-file-and-how-does-it-differ-from-other-openssl-generated-key-file

Zertifikate konvertieren:

Mit Hilfe von OpenSSL lassen sich viele Formate schnell und einfach in ein anderes Format konvertieren.

.p7b-Dateien können einfach in .spc umbenannt werden.

LetsEncrypt Wildcard-Zertifikate müssen zwingend per DNS validiert werden.

Tobit David kann S/MIME Zertifikate zentral im Server oder lokal im Client verwalten.

PGP

Auch PGP arbeitet mit unsymmetrischer Verschlüsselung, also mit einem öffentlichen und einem privaten Schlüssel.
Im Gegensatz zu S/MIME gibt es keine zentrale Zertifikatsinstanz,- jeder Nutzer sammelt also die öffentichen Zertifikate der Gegenseite selbst ein.
Damit ist das Verfahren nicht über die Zertifikatsstelle zu kompromittieren.
Das Einsammeln und Verwalten der öffentlichen Schlüssel ist aber entsprechend mühsam.

Mail-Gateway OpenSource: Ciphermail.
Hardware Appliance made in germany: Zertificon
Secure E-Mail Gateway: SeppMail
Nutzung von OpenSSL, Zertifikats-Formate:
https://www.msxfaq.de/signcrypt/openssl.htm

Eindeutige ID: #1255
Verfasser: Uwe Kernchen
Letzte Änderung: 2022-06-23 15:32