Informatieclassificatie Vlaamse overheid (Vo-ICR)

Cryptografische maatregelen

Minimale maatregelen

Team Informatieveiligheid| Digitaal Vlaanderen

Dit is een document voor publiek gebruik

AGENTSCHAP
DIGITAAL VLAANDEREN
HAVENLAAN 88 BUS 60, 1000 BRUSSEL

INHOUD VAN DIT DOCUMENT

Situering van het document

Dit document maakt deel uit van de begeleidende documentatie in context van het generieke informatieclassificatieraamwerk (ICR) van de Vlaamse overheid (Vo) op initiatief van het Stuurorgaan Vlaams Informatie- en ICT Beleid.

Doel van het document

Dit document beschrijft de kwaliteitseisen van de minimale maatregelen die moeten afgedwongen worden in het kader van de deelprocessen encryptie en sleutelbeheer. Deze kwaliteitseisen worden verbonden aan de informatieklassen zoals deze zijn beschreven in het ICR van de Vo.

Werkprincipe van het document

Het huidige document bestaat uit 3 delen. Eerst worden de minimale maatregelen besproken, alvorens in het tweede deel al de nodigde aanvullende informatie ter beschikking wordt gesteld. Vervolgens bespreken we de link met andere maatregelen.

De organisatie van de maatregelen in het kader van informatieclassificatie wordt beschreven in het document ‘Vo informatieclassificatie – Organisatie - Informatieclassificatieraamwerk’.

Verspreiding van het document

Dit document is voornamelijk voor intern gebruik en bevat informatie die mag gedeeld worden met personen verbonden aan de Vo en zijn diensten leveranciers.

Het document mag gedeeld worden met derden onder de richtlijnen openbaarheid van bestuur van de Vlaamse regering.

Vrijwaring

Dit document geeft de huidige status en daaraan gerelateerde informatie binnen het project weer. Gelieve de auteur of de bevoegde project begeleiders te contacteren om zich ervan te vergewissen dat u de laatst gevalideerde versie van dit document in handen heeft.

Eigenaar

Voorzitter werkgroep informatieveiligheid | Digitaal Vlaanderen

security@vlaanderen.be

Classificatie

Dit document valt onder de vertrouwelijkheidsklasse 1 (Publiek) en mag toegankelijk zijn voor iedereen.
Dit document valt onder integriteitsklasse 2. De organisatie heeft **geen directe hinder** indien de informatie gewijzigd wordt door onbevoegden en de **dienstverlening blijft gegarandeerd**. Maar bijsturing van de criteria en/of maatregelen is aangewezen.

Historiek

v.0.1	5 maart 2018	Chris DE VUYST	Eerste draft
v.0.2	23 mei 2018	Johan SMEKENS	〉 Criteria: VTC 02/2018 d.d. 18 april 2018. →minimale specifieke GDPR-maatregelen
v.0.3	15 juni 2018	Kristel VAN AKEN	〉 Naam document gewijzigd 〉 Scope uitbreiding 〉 Tekst aangevuld en herwerkt 〉 Technische toelichting in bijlagen verwerkt
v.1.0	18 juni 2018	Johan SMEKENS	〉 Publicatie eerste versie
v.1.1	11 september 2018	Kristel VAN AKEN	Link met andere minimale maatregelen bijgewerkt
v.1.2	16 April 2019	Kristel VAN AKEN	Versie gepubliceerd in pdf
v.1.3	1 oktober 2020	Kristel VAN AKEN	Integriteit toevoegen
v.1.4	9 april 2021	Kristel VAN AKEN	CAA toevoegen
v.1.5	11 oktober 2021	Kristel VAN AKEN	Beschikbaarheid toevoegen
v.2.0	22 juli 2022	Kristel VAN AKEN	Verbeteren leesbaarheid Geen inhoudelijke wijzigingen
V.2.1	17 oktober 2023	Nele Lowet	Update KSZ

Bronnen en verwijzingen

De inhoud van dit document werd samengesteld op basis van workshops.

Documentverwijzingen:

- Vo informatieclassificatie – Organisatie Informatieveiligheid (PDF)
- *Vo Informatieclassificatie – Organisatie Informatieclassificatieraamwerk* (PDF)
- Handbook of applied cryptography
- Kennisplatform NORA
- BIR- Encryptiebeleid
- Adviezen Vlaamse Toezichtcommissie
- Vo Informatieclassificatie – Minimale maatregelen – (PDF)
  - *Vo Informatieclassificatie - Minimale maatregelen – IAM*
  - *Vo Informatieclassificatie - Minimale maatregelen – netwerken*
- Vo Informatieclassificatie – Overzicht baseline maatregelen (XLS)

Inhoudsopgave

1. MINIMALE MAATREGELEN

1.1 Minimale maatregelen – Cryptografie

1.1.1 Minimale algemene maatregelen

Vertrouwelijkheid

IC klasse	Minimale maatregelen
	〉 Versleuteling vindt plaats conform *Common practices*, crypto-algoritmes en protocollen: open standaarden. 〉 DIM 〉〉 Terminatie op de perimeter van het beveiligde netwerk; en 〉〉 Technische standaard: 〉〉〉 TLS-protocol: *forward secrecy* verplicht indien technisch mogelijk. 〉 DIU 〉〉 Mitigerende maatregelen na risicoanalyse.
	Alle maatregelen van Klasse 1 + 〉 DAR 〉〉 Encryptie niet verplicht. Afscherming op niveau van organisatie d.m.v. fysieke en/of logische toegangsmaatregelen.
	Alle maatregelen van Klasse 1 + Klasse 2 + 〉 DAR 〉〉 In een beschermde omgeving (d.w.z. een omgeving waar het veiligheidsbeheer onder Vo-controle is of de genomen maatregelen gekend zijn en als afdoende geacht worden door de Vo): afscherming op niveau functionele behoefte d.m.v. fysieke afscherming, logische toegangsbeveiliging. Enkel encryptie na risicoanalyse; en 〉〉 In een onbeschermde omgeving (fysieke en logische toegangsmaatregelen niet afdoende en/of beveiligingsbeheer niet onder Vo-controle): encryptie voor de volledige verwerkingsketting: op opslagniveau, database (DB) of *middleware* …, werkposten, mobiele toestellen, back-up, ... 〉 DIM 〉〉 Veilige export buiten de toepassing (*application layer*, database, ...): fysieke beveiliging, logische toegangsbeveiliging (incl. *interapp/intralayer* transport); 〉〉 Veilige export buiten de organisatie (Veiligheidsbeheer niet onder controle van de Vo): encryptie op niveau transport indien netwerk niet beschermd is en terminatie op niveau vertrouwde infrastructuur (bv. in DMZ); en 〉〉 Technische standaard: 〉〉 Transport (TLS) protocol (*system-to-system*): wederzijdse authenticatie *(2-way TLS*); 〉〉 Transport (TLS) protocol (*client-server*): wederzijdse authenticatie; 〉〉〉 *2-way TLS* of; 〉〉〉 *1-way TLS* + eIDAS substantiële authenticatie.

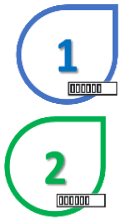
	〉 Certificaten en sleutels implementatiecriteria 〉〉 Controle op gebruik sterke versleuteling verplicht; en 〉〉 Implementatie CAA en DNSSEC op DNS CAA-records.
	Alle maatregelen van Klasse 1 + Klasse 2 + Klasse 3 + 〉 DAR 〉〉 Encryptie voor de volledige verwerkingsketting: op opslagniveau, database (DB) of *middleware* …, werkposten, mobiele toestellen, back-up, ... 〉 DIM 〉〉 Transport context-onafhankelijk (zowel binnen als buiten de organisatie); en 〉〉 Technische standaard: 〉〉〉 Gebruik van recentste versie TLS en *forward secrecy* verplicht.
	Alle maatregelen van Klasse 1 + Klasse 2 + Klasse 3 + Klasse 4 + 〉 DIM 〉〉 Encryptie op zowel berichtniveau als transportniveau (tunnel).

Integriteit

IC klasse	Minimale maatregelen
	〉 Versleuteling vindt plaats conform *Common practices*, crypto-algoritmes en protocollen: open standaarden. 〉 DIM 〉〉 Encryptie op transportniveau omwille van integriteitsdoeleinden (d.w.z. https-ontsluiting van publieke websites); en 〉〉 Terminatie op de perimeter van het beveiligde netwerk 〉 DIU 〉〉 Mitigerende maatregelen na risicoanalyse.
	Alle maatregelen van Klasse 1 + 〉 DAR 〉〉 Encryptie is niet verplicht. Afscherming op niveau van de organisatie d.m.v. fysieke en/of logische toegangsmaatregelen.
	Alle maatregelen van Klasse 1 + Klasse 2 + 〉 DAR 〉〉 In een beschermde omgeving (d.w.z. omgeving waar het veiligheidsbeheer onder Vo-controle is of de genomen maatregelen gekend zijn en als afdoende geacht worden door de Vo): afscherming op niveau functionele behoefte d.m.v. fysieke afscherming, logische toegangsbeveiliging. Enkel encryptie na risicoanalyse; en

	〉 In een onbeschermde omgeving (fysieke en logische toegangsmaatregelen niet afdoende en/of beveiligingsbeheer niet onder Vo-controle): encryptie voor de volledige verwerkingsketting: op opslagniveau, database (DB) of *middleware* …, werkposten, mobiele toestellen, back-up, ... 〉 DIM 〉 Veilige export buiten de toepassing (*application layer*, database, ...): fysieke beveiliging, logische toegangsbeveiliging (incl. *interapp/intralayer* transport); 〉 Veilige export buiten de organisatie (Veiligheidsbeheer niet onder controle van de Vo): encryptie op niveau transport indien netwerk niet beschermd is en terminatie op niveau vertrouwde infrastructuur (bv. in DMZ); en 〉 Technische standaard: 〉 Transport (TLS) protocol (*system-to-system*): wederzijdse authenticatie *(2-way TLS*); 〉 Transport (TLS) protocol (*client-server*): wederzijdse authenticatie; 〉 *2-way TLS* of; 〉 *1-way TLS* + eIDAS substantiële authenticatie. 〉 Certificaten en sleutels implementatiecriteria 〉 Controle op gebruik sterke versleuteling verplicht; en 〉 Implementatie CAA en DNSSEC op DNS CAA-records.
	Alle maatregelen van Klasse 1 + Klasse 2 + Klasse 3 + 〉 DAR 〉 Encryptie voor de volledige verwerkingsketting: op opslagniveau, database (DB) of *middleware* …, werkposten, mobiele toestellen, back-up, ... 〉 DIM 〉 Transport context-onafhankelijk (zowel binnen als buiten de organisatie).
	Alle maatregelen van Klasse 1 + Klasse 2 + Klasse 3 + Klasse 4 + 〉 DIM 〉 Encryptie op zowel berichtniveau als transportniveau (tunnel).

Beschikbaarheid

IC klasse	Minimale maatregelen
	Klasse 1 en Klasse 2 kennen dezelfde maatregelen: 〉 Reserveonderdelen en -componenten voorzien.

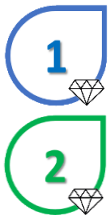
	Klasse 3 en Klasse 4 en Klasse 5 kennen dezelfde maatregelen: Alle maatregelen van Klasse 1 / Klasse 2 + 〉 High-availability-infrastructuur implementeren (zie 'Vo informatieclassificatie – minimale maatregelen – netwerken').

1.1.2 Minimale specifieke (GDPR) maatregelen

Vertrouwelijkheid

IC klasse	Minimale maatregelen
	Er zijn geen GDPR maatregelen voor klasse 1 en klasse 2.
	Klasse 3 en Klasse 4 kennen dezelfde maatregelen: 〉 DAR 〉 Advies VTC 02/2018 d.d. 18 april 2018 〉 Encryptie toepassen bij centrale opslag (*datacenter*) in een beschermde omgeving (d.w.z. omgeving waar het veiligheidsbeheer onder Vo-controle is of de genomen maatregelen gekend zijn en als afdoende geacht worden door de Vo): afscherming op niveau functionele behoefte d.m.v. fysieke afscherming, logische toegangsbeveiliging; en 〉 In een onbeschermde omgeving (fysieke en logische toegangsmaatregelen niet afdoende en/of beveiligingsbeheer niet onder Vo-controle): Encryptie voor de volledige verwerkingsketting: op opslagniveau, database (DB) of *middleware* …, werkposten, mobiele toestellen, back-up, ... 〉 DIM 〉 Tunnel- of berichtencryptie
	Er zijn geen GDPR maatregelen voor klasse 5.

Integriteit

IC klasse	Minimale maatregelen
	Er zijn geen GDPR maatregelen voor klasse 1 en klasse 2.
	Klasse 3 en Klasse 4 kennen dezelfde maatregelen: 〉 DAR 〉〉 Advies VTC 02/2018 d.d. 18 april 2018 〉〉〉 Encryptie toepassen bij centrale opslag (*datacenter*) in een beschermde omgeving (d.w.z. omgeving waar het veiligheidsbeheer onder Vo-controle is of de genomen maatregelen gekend zijn en als afdoende geacht worden door de Vo): afscherming op niveau functionele behoefte d.m.v. fysieke afscherming, logische toegangsbeveiliging; en 〉〉〉 In een onbeschermde omgeving (fysieke en logische toegangsmaatregelen niet afdoende en/of beveiligingsbeheer niet onder Vo-controle): Encryptie voor de volledige verwerkingsketting: op opslagniveau, database (DB) of *middleware* …, werkposten, mobiele toestellen, back-up, ... 〉 DIM 〉〉 Tunnel of bericht encryptie.
	Er zijn geen GDPR maatregelen voor klasse 5.

Beschikbaarheid

Er zijn geen specifieke GDPR-maatregelen voor beschikbaarheid.

1.1.3 Minimale specifieke (NISII) maatregelen

In afwachting van de goedkeuringen omtrent NISII is er in dit document alvast de nodige ruimte voorzien voor toekomstige minimale specifieke NISII maatregelen.

1.1.4 Minimale specifieke (KSZ) maatregelen

De minimale algemene maatregelen voor sleutelbeheer moeten toegepast worden: per klasse zijn de overeenkomende maatregelen van toepassing (zie hoofdstuk '*minimale algemene maatregelen*').

Volgens de Minimale Normen van de Kruispuntbank Sociale Zekerheid moeten volgende maatregelen in het kader van Cryptografische maatregelen toegepast worden:

Beschikbaarheid, Integriteit en vertrouwelijkheid

IC klasse	Minimale maatregelen
	Klasse 1 t/m Klasse 5 kennen dezelfde maatregelen: 〉 De organisatie dient een formeel beleid voor het gebruik van cryptografische controles op te zetten, te valideren, te communiceren en te onderhouden. Hierbij moet ze gebruik maken van de 'Richtlijnen rond het gebruik van cryptografische controles' zoals opgesomd in de bijlage C van de beleidslijn 'vercijferen' (Ref. KSZ 5.7.1). 〉 KSZ Bijlage C "Richtlijnen rond het gebruik van cryptografische controles": ○ Maatregelen moeten bepaald worden op basis van een duidelijke formele risico-analyse waarbij antwoord wordt gegeven op de volgende vragen: ▪ Hoe wordt omgegaan met data die opgeslagen wordt op verwijderbare media? ▪ Waar wordt data opgeslagen of verwerkt? ▪ Hoe wordt de Cryptografische vertrouwelijkheid, integriteit of authenticiteit van de data gewaarborgd? ▪ Hoe wordt de onweerlegbaarheid van een activiteit gewaarborgd? ○ Wanneer cryptografie vereist is, moet steeds zo sterk mogelijke cryptografische maatregel gebruikt worden. ○ De organisatie moet een overzicht bijgehouden waarin terug te vinden is waar cryptografische maatregelen worden toegepast, welke cryptografische maatregelen worden toegepast en wie hiervoor verantwoordelijk is. ○ De gebruikte cryptografische maatregelen moeten door onafhankelijke betrouwbare deskundige getoetst worden. ○ De ICT veiligheids-verantwoordelijke moet bepalen welke cryptografische maatregelen in welke gevallen toegepast moeten worden, gelet op de huidige goede praktijken. ○ De toepassing en gepastheid van cryptografische oplossingen en maatregelen moet periodiek beoordeeld worden. ○ Versleutelde data van derden die binnenkomen op het netwerk van de organisatie moeten eerst gedecrypteerd worden om gescand te worden op virussen en andere malware.

1.2 Minimale maatregelen – Sleutelbeheer

Merk op: Onderstaande implementatie criteria zijn minimale maatregelen.

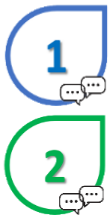
〉 Iedere entiteit dient een beleid en proces voor *lifecycle-management* van certificaten en cryptografische sleutels op te stellen en toe te passen:
- Generatie;
- Distributie;
- Operationeel beheer (rotatie, back-up/herstel);
- Verwijderen /vernietigen;
- Archivering; en
- Compromitteren sleutels en revocatie certificaten.

〉 De vertrouwelijkheid, integriteit en authenticiteit van cryptografische sleutels dient gewaarborgd te zijn tijdens generatie, gebruik, transport en opslag van de sleutels.

- Naarmate de klasse hoger is, zullen procedures strikter zijn en de mate van functiescheiding toenemen.

1.2.1 Minimale algemene maatregelen

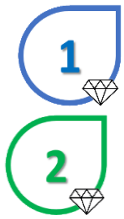
Vertrouwelijkheid

IC klasse	Minimale maatregelen
	Klasse 1 en Klasse 2 kennen dezelfde maatregelen: 〉 Generatie: in een veilige omgeving: fysieke en logische toegangsbeveiliging + gebruik van standaard cryptomodules (softwarebibliotheken, API's, hardware modules, ...); 〉 Distributie sleutels: via een geschikt protocol en/of via een geschikte datamedia of via communicatieverbindingen in een vorm die de vertrouwelijkheid, integriteit, en authenticiteit ervan waarborgt; 〉 Registratie van alle in omloop zijnde sleutels en certificaten; 〉 Bescherming opslag sleutelmateriaal: geëncrypteerde opslag of hardware token; 〉 Verschillende sleutels per omgeving (test- vs. productieomgeving)/ toepassing/ organisatie/ finaliteit (encryptie vs. digitale handtekening); 〉 Sleutelsterkte afgestemd op informatieklasse; 〉 Geldigheidsduur van cryptografische sleutels wordt afgestemd op het beoogde gebruik en is vastgelegd in het cryptografische beleid; 〉 Gebruik en beheer sleutels: stringente toegangscontrole (*Identity and Access Management* of IAM en principe van *Least privilege*) met sterke garanties op vlak van traceerbaarheid (zie hoofdstuk: link met IAM als maatregel en hoofdstuk: link met logging als maatregel); 〉 De authenticiteit van publieke sleutels wordt gegarandeerd d.m.v. van een PKI (naargelang de use case: intern, Vo-PKI of commerciële erkende CA); 〉 Het beheer van certificaten binnen de entiteit wordt vastgelegd (wie-wat-hoe); 〉 Geen certificaten delen tussen verschillende omgevingen (bv. test versus productie); 〉 OV- of EV-certificaten worden aangeraden indien de site ontsloten wordt naar een onbeveiligd netwerk (internet); en 〉 Sensibilisering medewerkers (zorgvuldig omgaan met encryptie en sleutelbeheer).

	Klasse 3 en Klasse 4 kennen dezelfde maatregelen:
	
	Alle maatregelen van Klasse 1 / Klasse 2 +
	
	〉 Functiescheiding:
	  〉 Toegangsbeheer;
	  〉 Applicatiebeheer;
	  〉 Sleutelbeheer; en
	  〉 Sleutelgebruik.

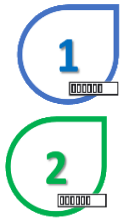
	Alle maatregelen van Klasse 1 / Klasse 2 + Klasse 3 / Klasse 4 +
	
	〉 Generatie: crypto module = FIPS 140-2 level 2 hardware;
	〉 Sleutelceremonie voor nieuwe mastersleutels;
	〉 FIPS 140-2 level 2 hardware voor bewaren geheime sleutels, minimaal:
	  〉 *Root CA private*-sleutel; en
	  〉 *Master keys* of mastersleutels;
	〉 Controle functiescheiding: sleutelbeheer *(4-ogenprincipe*); en
	〉 Jaarlijkse periodieke herziening van de toegangen m.b.t. sleutelbeheer.

Integriteit

IC klasse	Minimale maatregelen
	Klasse 1 en Klasse 2 kennen dezelfde maatregelen: 〉 Generatie: in een veilige omgeving: fysieke en logische toegangsbeveiliging + gebruik van standaard cryptomodules (softwarebibliotheken, API's, hardware modules, ...); 〉 Distributie sleutels: via een geschikt protocol en/of via een geschikte datamedia of via communicatieverbindingen in een vorm die de vertrouwelijkheid, integriteit, en authenticiteit ervan waarborgt; 〉 Registratie van alle in omloop zijnde sleutels en certificaten; 〉 Bescherming opslag sleutelmateriaal: geëncrypteerde opslag of hardware token; 〉 Verschillende sleutels per omgeving (test- vs. productieomgeving) /toepassing /organisatie /finaliteit (encryptie vs. digitale handtekening); 〉 Sleutelsterkte afgestemd op informatieklasse; 〉 Geldigheidsduur van cryptografische sleutels wordt afgestemd op het beoogde gebruik en is vastgelegd in het cryptografische beleid; 〉 Gebruik en beheer sleutels: stringente toegangscontrole (*Identity and Access Management* of IAM en principe van *Least privilege*) met sterke garanties op vlak van traceerbaarheid (zie hoofdstuk: link met IAM als maatregel en hoofdstuk: link met logging als maatregel); 〉 De authenticiteit van publieke sleutels wordt gegarandeerd d.m.v. van een PKI (naargelang de use case: intern, Vo-PKI of commerciële erkende CA); 〉 Het beheer van certificaten binnen de entiteit wordt vastgelegd (wie-wat-hoe); 〉 Geen certificaten delen tussen verschillende omgevingen (bv. test versus productie); 〉 OV- of EV-certificaten worden aangeraden indien de site ontsloten wordt naar een onbeveiligd netwerk (internet); en 〉 Sensibilisering medewerkers (zorgvuldig omgaan met encryptie en sleutelbeheer).

	Klasse 3 en Klasse 4 kennen dezelfde maatregelen: Alle maatregelen van Klasse 1 / Klasse 2 + 〉 Functiescheiding: 〉 Toegangsbeheer; 〉 Applicatiebeheer; 〉 Sleutelbeheer; en 〉 Sleutelgebruik.
	Alle maatregelen van Klasse 1 / Klasse 2 + Klasse 3 / Klasse 4 + 〉 Generatie: crypto module = FIPS 140-2 level 2 hardware; 〉 Sleutelceremonie voor nieuwe mastersleutels; 〉 FIPS 140-2 level 2 hardware voor bewaren geheime sleutels, minimaal: 〉 *Root CA private*-sleutel; en 〉 *Master keys* of mastersleutels; 〉 Controle functiescheiding: sleutelbeheer *(4-ogenprincipe)*; en 〉 Jaarlijkse periodieke herziening van de toegangen m.b.t. sleutelbeheer.

Beschikbaarheid

IC klasse	Minimale maatregelen
	Klasse 1 en Klasse 2 kennen dezelfde maatregelen: 〉 Indien herstel van versleutelde informatie nodig is, overweeg dan een beveiligde back-up van sleutels en sleutelparen. 〉 Noot: gebruik van een back-up-omgeving voor sleutelparen voor digitale handtekening kan de legitimiteit en dus de bruikbaarheid van de handtekening ondermijnen.
	Klasse 3 en Klasse 4 en Klasse 5 kennen dezelfde maatregelen: Alle maatregelen van Klasse 1 / Klasse 2 + 〉 Maak gebruik van een KMS (*Key Management System*) met aandacht voor de beschikbaarheid van sleutels en sleutelparen.

1.2.2 Minimale specifieke (GDPR-)maatregelen

Er zijn op heden geen minimale specifieke maatregelen voor sleutelbeheer geïdentificeerd op basis van de criteria beschreven in de GDPR (en aanverwante) regelgeving.

1.2.3 Minimale specifieke (NISII) maatregelen

In afwachting van de goedkeuringen omtrent NISII is er in dit document alvast de nodige ruimte voorzien voor toekomstige minimale specifieke NISII maatregelen.

1.2.4 Minimale specifieke (KSZ) maatregelen

De minimale algemene maatregelen voor sleutelbeheer moeten toegepast worden: per klasse zijn de overeenkomende maatregelen van toepassing (zie hoofdstuk '*minimale algemene maatregelen*').

Volgens de Minimale Normen van de Kruispuntbank Sociale Zekerheid moeten volgende maatregelen in het kader van Sleutelbeheer toegepast worden:

IC klasse	Minimale maatregelen
	Klasse 1 t/m Klasse 5 kennen dezelfde maatregelen: 〉 Een formeel beleid voor het gebruik, bescherming en levensduur van cryptografische sleutels voor de ganse levenscyclus opzetten, valideren, communiceren en onderhouden. Hierbij moet ze gebruik maken van de 'Richtlijnen rond het sleutel beheer' zoals opgesomd in de bijlage D van de beleidslijn 'vercijferen'(Ref. KSZ 5.7.1). 〉 KSZ Bijlage D "richtlijnen rond het sleutelbeheer": ○ De organisatie is verantwoordelijk voor effectief sleutelbeheer. Specifieke processen en procedures gerelateerd aan sleutelbeheer moeten opgesteld, gevalideerd, gecommuniceerd worden aan alle betrokken actoren en ook regelmatig onderhouden worden. Het sleutelbeheer moet minimaal de volgende thema's omvatten: ▪ Aanvragen/genereren van sleutels ▪ Opslag van (privé)sleutels ▪ Transport van (privé)sleutels ▪ Gebruik van sleutels ▪ Vervangen en vernietigen van sleutels ▪ Archiveren van sleutels ▪ Omgaan met gecompromitteerde sleutels ○ De volgende minimale richtlijnen moeten gelden voor het aanvragen/genereren van sleutels: ▪ Er moet gekozen worden voor de sterkste cryptografische maatregel die in de praktijk werkbaar is. ▪ Sleutels moeten een activatie- en verloopdatum hebben. ▪ De geldigheidsduur moet afhankelijk zijn van het beoogde doel en de tijd die het zou kosten om de sleutel te kraken. ▪ Elke sleutel moet uniek zijn. ▪ Een sleutel moet alleen voor een specifiek doel en omgeving gegenereerd worden. ▪ Sleutels moeten door een erkende partij geleverd worden die werkt volgens een goede praktijk. De volgende minimale richtlijnen moeten gelden voor de opslag van (privé)sleutels: ▪ Sleutels moeten op zo weinig mogelijk locaties opgeslagen worden. ▪ Systemen moeten de door het systeem gebruikte sleutels afschermen voor gebruikers.

	▪ Sleutels moeten beschermd worden tegen verlies of wijzigingen (bv. door een kopie bij te houden). Toegang tot sleutels moet tot een minimum beperkt zijn (tot de verantwoordelijke van de sleutel). ▪ Sleutels zijn alleen toegankelijk voor de technische experten ▪ Bij gevoelige of kritieke data zijn er minimaal twee beheerders. ▪ Sleutels moeten minimaal even goed beschermd worden als de betrokken data. o De volgende minimale richtlijnen moeten gelden voor het transport van (privé)sleutels: ▪ Wanneer sleutels leesbaar overgedragen worden, moet dit in persoon gebeuren of via een alternatief betrouwbaar kanaal gebeuren. ▪ Deze middelen en methodes om sleutels te communiceren moeten eerst goedgekeurd worden door de informatieveiligheidsconsulent (CISO) / functionaris van gegevensbescherming (DPO). ▪ Minimaal de volgende richtlijnen moeten gelden voor het gebruik van sleutels: • Elke sleutel moet alleen voor het toegewezen doel en omgeving ingezet worden. • Een sleutel die gebruikt wordt in productiesystemen mag niet gebruikt worden in niet productie systemen. • Binnen de organisatie is het belangrijkste gebruik van cryptografie van toepassing op: o Beveiliging van data op mobiele apparatuur • Opslag van wachtwoorden • Beveiliging van toepassingen • Beveiliging van communicatie van niet-publieke data over publieke netwerken (zoals VPN verbindingen). o Opslag en beveiliging van communicatie van kritieke data op het interne netwerk. o De volgende minimale richtlijnen moeten gelden voor het vervangen en vernietigen van sleutels: ▪ Alle sleutels moeten na de verloopdatum overal waar deze opgeslagen of toegepast werden verwijderd worden. ▪ Zo nodig moet een nieuwe sleutel met dezelfde eisen gegenereerd worden. o De volgende minimale richtlijnen moeten gelden voor het archiveren van sleutels: ▪ Sleutels die gebruikt werden door gebruikers die de organisatie verlaten hebben, moeten versleuteld en gearchiveerd worden. o De volgende minimale richtlijnen moeten gelden voor gecompromitteerde sleutels: ▪ Elke sleutel die gecompromitteerd is of waarvan de verwachting bestaat dat deze gecompromitteerd is, moet direct vervangen worden. ▪ Er moet een procedure zijn vastgesteld voor elk type maatregel waarin is bepaald hoe gehandeld moet worden wanneer een sleutel mogelijk gecompromitteerd is of wanneer een kwetsbaarheid bekend wordt. ▪ Een gecompromitteerde sleutel mag geen data verschaffen die gebruikt kan worden om de vervangende sleutel te bepalen. Voor elke sleutel moet een interne medewerker verantwoordelijk zijn. o Er moet een overzicht bijgehouden worden van alle verantwoordelijken voor sleutels

	○ Er moeten maatregelen toegepast worden om ongeautoriseerde pogingen tot verspreiding, ontcijfering, toegang, gebruik, wijziging of vervanging van sleutels of versleutelde data te detecteren. In overeenkomsten met leveranciers van cryptografische diensten of producten moeten deze richtlijnen ingesloten zijn. ○ Er moeten procedures opgesteld worden die bepalen hoe omgegaan moet worden met de aanvragen voor toegang tot versleutelde data (zoals in het geval van een rechtszaak of in geval van een klacht die ingediend is bij de organisatie). ○ Toegang tot of het gebruik van privésleutels moeten gelogd worden volgens de procedures in het document “BLD Logbeheer”. ○ KSZ Beleidslijn Logbeheer: Elke organisatie onderschrijft de volgende beleidslijnen van informatieveiligheid en privacy voor alle informatie en informatiesystemen onder de verantwoordelijkheid van de organisatie. 1. De organisatie dient een formele procedure van logbeheer op te zetten, te valideren, te communiceren en te onderhouden. 2. De organisatie moet transacties, controlewerkzaamheden, activiteiten van gebruikers, uitzonderingen en informatieveiligheid- en privacy-gebeurtenissen/incidenten gestructureerd vastleggen in afzonderlijke logbestanden, zodat iedere handeling naar de brondocumenten herleid kan worden of uitgevoerde bewerking(en) gecontroleerd kan worden. 3. Logbeheer moet meegenomen worden vanaf het design tijdens de ontwikkeling of bij de bepalingen van aankoopcriteria van toepassingen of systemen om “security/privacy by design” te realiseren. 4. Elke toegang tot gegevens met gevoeligheidsklasse vertrouwelijk of hoger, moet gelogd worden in overeenstemming met de toepasselijke wetgeving en regelgeving. 5. De interne klokken van alle informatiesystemen van de organisatie dienen gesynchroniseerd te worden met een overeengekomen nauwkeurige tijdsbron dat een betrouwbare analyse van logbestanden op verschillende informatiesystemen altijd mogelijk is. 6. De noodzakelijke tools moeten beschikbaar zijn of ontwikkeld worden om log gegevens te kunnen uit te baten en te laten analyseren door de geautoriseerde personen. Via de tools moet het mogelijk zijn om de logs snel, glashelder en eenvoudig te kunnen raadplegen. 7. Zoveel als mogelijk wordt systeemgebruik automatisch gelogd, als dit niet mogelijk is kan ook gebruik gemaakt worden van een manueel logboek door systeembeheerders. 8. Logbestanden dienen beschermd te worden tegen inzage door onbevoegden, wijzigingen en verwijderingen. 9. De logbestanden moeten gedurende een overeengekomen periode worden bewaard, ten behoeve van toekomstig onderzoeken en controles en in overeenstemming met wetgeving en regelgeving . In het bijzonder dienen de privacy logs minstens 10 jaar bewaard worden. 10. De kwaliteit van de privacy log dient een gepast antwoord te bieden om het gebruik te rechtvaardigen (al dan niet gebaseerd op een voorafgaandelijke autorisatie of machtiging). De log dient per verwerking een aanduiding te bevatten van wie wanneer over wie welke persoonsgegevens heeft verwerkt voor welke doeleinden en met welk resultaat (OK,NOK).

	11. De raadpleging van logbestanden is altijd het voorwerp van een georganiseerde procedure binnen de organisatie met een historiek van de verzoeken die werden goedgekeurd/uitgevoerd of die werden afgekeurd. 12. Het resultaat van logbeheer moet regelmatig geanalyseerd, gerapporteerd en beoordeeld worden

2. AANVULLENDE INFORMATIE OVER DE MAATREGELEN

2.1 De technieken

Cryptografie omvat een scala aan technieken die allemaal onder één of andere vorm te maken hebben met encryptie van informatie:

Vanuit gebruikersperspectief:

- Het onleesbaar maken van informatie voor onbevoegden door versleuteling met een encryptiesleutel;
- De integriteit van informatie garanderen door toevoegen van een *hash* of digitale handtekening – deze actie maakt de informatie echter *niet* onleesbaar voor onbevoegden, maar garandeert enkel dat de informatie niet gewijzigd is door onbevoegden;
- Ervoor zorgen dat informatie niet door een bepaalde persoon kan worden weerlegd door middel van digitale handtekening ondersteund door een digitaal certificaat – dit noemt men onweerlegbaarheid of *non-repudiation*;
- Authenticatie van een gebruiker door middel van een aan hem/haar toegekend digitaal certificaat – eID is hiervan een voorbeeld.

Technische activiteiten – deze zijn transparant voor een gebruiker:

- De uitgifte en beheer van digitale certificaten door middel van een PKI (*Public Key Infrastructure*);
- Het sleutelbeheer en de levenscyclus van een sleutel of sleutelpaar;
- De vertrouwelijkheid en integriteit van informatie tijdens transport over een netwerk bijvoorbeeld door het opzetten van VPN (*Virtual Private Network*);
- De afkomst van software garanderen door middel van digitale handtekening;
- De authenticiteit van een website garanderen d.m.v. een (*Secure Sockets Layer*- of SSL-) certificaat; en
- Systeemauthenticatie door middel van digitale certificaten.

Deze technieken bieden een antwoord op de volgende vragen:

- Hoe kan worden vastgesteld met wie er wordt gecommuniceerd en hoe weet de ontvanger zeker dat de verzender ook daadwerkelijk de verzender is en niet iemand anders?
- Hoe kan worden voorkomen dat informatie tijdens transport en opslag onopgemerkt worden gewijzigd, zodat de ontvanger een zekere mate van garantie heeft dat de informatie integer is en dat de informatie inderdaad afkomstig is van de persoon wiens identiteit als ondertekenaar staat vermeld?
- Hoe kan ervoor worden gezorgd dat de inhoud van informatie onleesbaar is voor niet-geautoriseerde personen?
- Waarmee kan worden aangetoond dat informatie tijdens transport of opslag (niet) is gewijzigd?
- Waarmee kan worden aangetoond dat bepaalde gebeurtenissen of handelingen hebben plaatsgevonden, zoals het verzenden en ontvangen van digitale documenten?

Cryptografie is de wetenschappelijke discipline om aan de hand van wiskundige technieken informatie te beveiligen. Door toepassing van cryptografische technieken kunnen een aantal **veiligheidsdiensten** (*security services*) gerealiseerd worden:

- **Vertrouwelijkheid**: waarborgen dat informatie enkel leesbaar is voor degenen die daartoe geautoriseerd zijn; (Zie hoofdstuk: '*Bouwstenen voor vertrouwelijkheid*')

- **Integriteit**: waarborgen dat informatie of functionaliteit niet werd gewijzigd door onbevoegden; (Zie hoofdstuk: '*Bouwstenen voor integriteit'*)
- **Authenticatie** van entiteiten: verifiëren van de identiteit van een entiteit (entiteit= persoon, organisatie, proces, systeem …); (Zie hoofdstuk: *'Bouwstenen voor authenticatie'*)
- **Data-authenticatie**: waarborgen van de oorsprong en integriteit van informatie. Data-authenticatie heeft dus twee elementen: verifiëren dat de gegevens van de juiste entiteit afkomstig is en de integriteit van die gegevens valideren. Verwezenlijken van data-authenticatie is complexer dan verwezenlijken van vertrouwelijkheid.
- **Onweerlegbaarheid** (*non-repudiation*): voorkomen dat eerdere acties of verplichtingen kunnen worden ontkend; (Zie hoofdstuk: *'Bouwstenen voor onweerlegbaarheid'*)
- **Anonimiteit**: het waarborgen van de vertrouwelijkheid van communicerende entiteiten (d.w.z. metadata: wie communiceert met wie) is tevens een beveiligingsdienst waarvoor cryptografische technieken kunnen ingeschakeld worden**.**

Cryptografische technieken kunnen toegepast worden op opgeslagen informatie (*Data at Rest* of DAR), op informatie tijdens gebruik door een toepassing of systeem (*Data in Use* of DIU) of transport van informatie (*Data in Motion* of DIM).

Cryptografie is dus duidelijk meer dan encryptie alleen!

2.2 Encryptie

Encryptie is een belangrijke bouwsteen, zeg maar de basis, voor cryptografische technieken. Zonder encryptie, geen cryptografie!

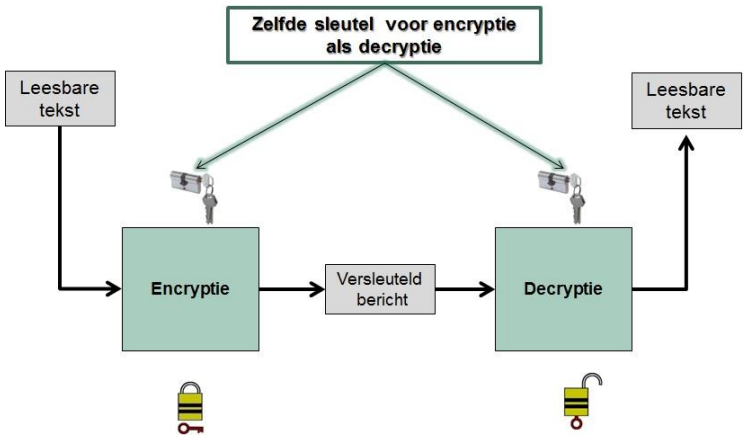
Encryptie is een mechanisme om informatie d.m.v. een wiskundig algoritme en cryptografische sleutel onleesbaar te maken voor onbevoegden. Naast toegangscontrole als beveiligingsmaatregel, is encryptie dus tevens een middel om vertrouwelijkheid van informatie te realiseren. Encryptie voorkomt dat een niet-geautoriseerde partij vertrouwelijke informatie kan lezen.

Het encrypteren van informatie wordt ook wel versleutelen of vercijferen genoemd, decrypteren wordt ontcijferen genoemd.

Er zijn twee encryptie technieken: symmetrische en asymmetrische encryptie.

Symmetrische encryptie

Bij symmetrische encryptie wordt een wiskundig algoritme gebruikt dat dezelfde **geheime sleutel** nodig heeft om zowel te encrypteren als te decrypteren. We spreken dan van een symmetrische geheime sleutel.



Beide partijen dienen over dezelfde geheime sleutel te beschikken. Deze geheime sleutel moet dus vooraf op een veilige manier gedistribueerd worden. In situaties met veel communicerende partijen, kan dit al vrij snel complex worden.

Gekende voorbeelden van algoritmen voor symmetrische encryptie zijn DES (*Data Encryption Standard*) en AES (*Advanced Encryption Standard*).

Voorbeelden hoe sleutelbeheer in de praktijk wordt toegepast:

- KMS: een *Key Management System* beheert de cryptografische sleutels met inbegrip van sleutel generatie, uitwisseling, opslag, gebruik, vernietiging en vervanging van deze sleutels.
- Kerberos: Kerberos is een authenticatiesysteem voor lokale netwerken met een client-serverarchitectuur, gebaseerd op een Trusted Third Party: een derde partij die door alle anderen wordt vertrouwd. Het beschermt *servers* tegen gebruik door niet-geautoriseerde partijen en clients tegen interactie met valse *servers*. Ook is voorzien in het genereren van een sessie-sleutel voor de communicatie tussen client en *server*, zodat indringers geen lopende sessies kunnen overnemen of afluisteren.

Met symmetrische systemen kan zowel vertrouwelijkheid van informatie als data-authenticatie verwezenlijkt worden, bijvoorbeeld door eerst encryptie van het document met symmetrische sleutel toe te passen om vervolgens de (*Message Authentication Codes*- of) MAC-waarde te berekenen van versleuteld document (bijlage 5 legt uit hoe MAC werken).

Encryptie wordt niet alleen gebruikt om informatie die verstuurd wordt te beveiligen tegen onbevoegden, maar kan ook worden gebruikt om gegevens op een laptop, externe harde schijf, USB-stick of andere mobiele opslagmedia onleesbaar te maken. Bij verlies of diefstal kan niemand de versleutelde gegevens lezen.

Symmetrische encryptie is snel en wordt daarom nog steeds gebruikt in beveiligingstechnieken, maar heeft een zwak punt, namelijk de doorgifte van de geheime sleutel is nodig om een versleuteld bericht te kunnen ontcijferen. Hoe krijg je die sleutel bij de ontvangende partij zonder dat deze gestolen wordt door onbevoegden? Hier komt asymmetrische encryptie op de proppen.

Asymmetrische encryptie

Het wiskundige algoritme voor asymmetrische encryptie, vaak ook aangeduid als *public key encryptie*, werkt op basis van een sleutelpaar: een publieke (of openbare) sleutel en een private sleutel. Er worden dus verschillende cryptografische sleutels gebruikt voor encrypteren en decrypteren. Van de twee sleutels die nodig zijn om te encrypteren en vervolgens te decrypteren is één sleutel geheim

(private sleutel) en de andere sleutel openbaar (publieke sleutel). De private sleutel moet door de eigenaar van het sleutelpaar geheimgehouden worden.

Het volledige proces encryptie/decryptie is dus afhankelijk van een sleutel*paar*, niet van een enkele sleutel. Zo'n sleutelpaar hoort bij elkaar, je hebt zowel de private sleutel nodig als de bijhorende publieke sleutel. Private en publieke sleutel worden dan ook samen als sleutelpaar gegenereerd.

Een gekend voorbeeld van een asymmetrische encryptie algoritme is RSA (*Rivest–Shamir–Adleman*).

Het grote voordeel van zulke sleutelparen is de eenvoud om de decryptiesleutel bij de ontvangende partij te krijgen: de publieke sleutel is immers openbaar en kan zonder meer gedeeld worden. Met de hele wereld. De focus kan zo gelegd worden op de beveiliging van de private sleutel – wat uiteraard een eenvoudigere klus is dan beveiliging van de hele wereld. Het zwakke punt van symmetrische encryptie is zo van de baan: het is heel eenvoudig om de vereiste sleutel bij de ontvangende partij te krijgen aangezien die gewoon publiek beschikbaar is.

Het nadeel van asymmetrische encryptie is de snelheid – of liever het gebrek eraan. Daarom wordt deze techniek vaak gecombineerd met symmetrische encryptie, waarbij een encryptiesleutel nodig voor encryptie van een omvangrijk stuk informatie versleuteld wordt met asymmetrische encryptie en zo bij de ontvangende partij gebracht wordt, die deze sleutel dan decrypteert met de publieke sleutel van de sturende partij zodat de sleutel bruikbaar is voor ontcijfering van de eigenlijke informatie.

Bij asymmetrische cryptografie geldt namelijk de volgende regel:

Wat men met de ene sleutel encrypteert, kan men decrypteren met de andere sleutel.

Naargelang de gebruikte sleutel om te encrypteren (publieke of private sleutel), kunnen andere securitydiensten gerealiseerd worden:

- Encryptie d.m.v. een publieke sleutel → *vertrouwelijkheid*; en
- Encryptie d.m.v. een private sleutel → *integriteit en onweerlegbaarheid (digitale handtekening)*

Symmetrisch vs. asymmetrisch

Symmetrische encryptie is over het algemeen veel sneller dan asymmetrische encryptie. Voor het snel versleutelen van grote hoeveelheden gegevens wordt dan ook bij voorkeur gebruik gemaakt van symmetrische algoritmen. Het nadeel van symmetrische encryptie is dat communicerende partijen vooraf hun geheime sleutel moeten uitwisselen.

Asymmetrische encryptie wordt enkel gebruikt om beperkte hoeveelheid data te vercijferen (bijvoorbeeld om andere sleutels zoals sessiesleutels te encrypteren, paswoorden, …). Deze techniek kent vooral zijn toepassing in data-authenticatie en sleutelbeheer/-distributie.

In de praktijk wordt meestal de combinatie van beide technieken (symmetrisch/asymmetrisch) gebruikt om informatie te beveiligen. Zo'n veel voorkomende toepassing waarbij de voordelen van beide systemen worden gecombineerd om berichten te vercijferen met behulp van een symmetrisch systeem en een eenmalige sleutel, de sessiesleutel, is de digitale enveloppe. Met deze techniek kan een bericht vercijferd verstuurd worden zonder dat hierbij de verzender en ontvanger van het bericht vooraf over eenzelfde geheime sleutel moeten beschikken. Asymmetrische encryptie wordt toegepast om de sessiesleutel te vercijferen. De techniek wordt toegelicht in bijlage 1.

Pretty Good Privacy (PGP) is een van de bekende voorbeelden van deze gecombineerde techniek.

Door te spelen met de sleutelparen van afzender en ontvanger kan asymmetrische encryptie, eventueel in combinatie met symmetrische encryptie, aldus worden gebruikt om berichten of

informatie onleesbaar te maken of om te garanderen dat een bericht afkomstig is van een bepaalde afzender.

2.3 PKI (*Public Key Infrastructure*)

Om versleutelde berichten tussen afzender en ontvanger uit te wisselen en te lezen moeten beide partijen in het bezit zijn van een sleutelpaar. Een sleutelpaar bestaat uit een private (geheime) sleutel en een publieke (openbare) sleutel. De private sleutel moet de eigenaar goed beveiligen en is ook alleen bekend bij de eigenaar. De publieke sleutel mag aan iedereen worden bezorgd.

Publieke sleutels hebben één nadeel: de ontvanger kan moeilijk controleren of de publieke sleutel afkomstig is van de 'echte' zender. Het zou namelijk ook van iemand kunnen zijn, die zich voordoet als de zender (dit noemt men *spoofing*). In zo'n geval helpt een digitaal certificaat. Een digitaal certificaat kan worden vergeleken met een paspoort of een identiteitskaart. Ze worden gebruikt als officiële legitimatie, om aan te tonen dat je bent wie je zegt dat je bent.

De geloofskracht van een certificaat hangt af van de partij of organisatie die het certificaat uitgeeft en de bijhorende controles die hieraan voorafgaan. Zo is het mogelijk om zelf een certificaat uit te geven, maar dan is de geloofwaardigheid eerder klein. Aan de andere kant kan een certificaat uitgegeven door een overheid als betrouwbaar worden beschouwd. In de praktijk zijn er heel wat organisaties die certificaten uitgeven en beheren: dit zijn commerciële organisaties én overheidsorganisaties.

Daarmee kan de echtheid van een entiteit en de relatie met zijn/haar publieke sleutel worden aangetoond.

Samengevat: welke beveiligingsdienst wordt wanneer toegepast?

Vertrouwelijkheid	Encryptie
Integriteit	Encryptie (*hash*) en digitale handtekening
Onweerlegbaarheid	Digitale handtekening met certificaat
Authenticatie	Digitale handtekening met certificaat

Een andere toepassing van cryptografie is het gebruik van websites. Organisaties stellen steeds meer diensten en informatie beschikbaar via internet. Het is belangrijk dat de gebruiker zeker weet dat de website waarop hij/zij gegevens invult daadwerkelijk van de organisatie is en of de communicatie met de website voldoende beveiligd is. Hiervoor zijn (*Secure Sockets Layer*- of) SSL-certificaten de oplossing. Een certificaat voegt een uniek zegel toe aan een website. Dit zegel is op websites beschikbaar voor controle van de echtheid en beveiliging van de website en garandeert op die manier de integriteit van die website.

De kracht van het certificaat

Een certificaat bevat volgende gegevens:

- Geregistreerde naam van de eigenaar: de certificaathouder;
- Publieke sleutel van de eigenaar;
- Geldigheidsperiode van het certificaat;
- Identiteit van de uitgever van het certificaat: de certificaatautoriteit (CA);
- Locatie van de *Certificate Revocation List* (bij de uitgever van het certificaat); en

- Samenvatting van bovenstaande gegevens, aangemaakt door een 'hash'-functie, en vervolgens vercijferd met de geheime sleutel van de CA. Dit is de digitale handtekening en dient om de geldigheid en authenticiteit van bovenstaande gegevens te waarborgen.

Om bruikbaar te zijn in de meeste toepassingen, worden digitale certificaten opgemaakt volgens een algemeen erkende standaard, zijnde X.509. Eén van de kenmerken van deze standaard is dat zij de mogelijkheid biedt van eenvoudige tot uitgebreide identiteitscontrole van de certificaathouder. Eén van de meest voorkomende types certificaten zijn SSL-certificaten voor de beveiliging van websites (*HyperText Transfer Protocol Secure* of HTTPS) en software om de identiteit van de organisatie die de website of software beheert, aan te tonen.

De kracht van zo'n certificaat naar betrouwbaarheid toe zit in twee criteria:

1. Hoe betrouwbaar is de uitgever van het certificaat?
2. Hoe goed is de identiteit van de eigenaar van het sleutelpaar geverifieerd?

Voor het eerste punt, de uitgever van het certificaat, is het belangrijk om het certificaat zelf te verifiëren:

- Is het certificaat niet uitgegeven door de eigenaar zelf (dan is er immers geen enkele controle op de identiteit van de eigenaar gebeurd)?
- Is het certificaat nog geldig en niet gecompromitteerd (validiteit en revocatie)?
- Is het certificaat uitgegeven door een betrouwbare partij?

Voor het tweede punt, de controle van de identiteit van de certificaathouder, zijn er drie opties bij het aankopen van een X.509-certificaat:

1. **Domeinvalidatie**: certificaten met domeinvalidatie bevatten geen bedrijfsgegevens. Er wordt alleen gecontroleerd of de aanvrager controle heeft over het domein waarvoor het certificaat wordt aangevraagd. Het certificaat wordt door alle browsers vertrouwd en zorgt voor een veilige verbinding d.m.v. encryptie. Bij SSL-certificaten met domeinvalidatie toont de browser een slot-icoontje, bv.:

2. **Organisatievalidatie** (OV): De certificaten met organisatievalidatie bevatten bedrijfsgegevens. De bezoekers van een website kunnen met deze bedrijfsgegevens controleren of ze op de website van het juiste bedrijf zijn. De bedrijfsgegevens worden in het OV-certificaat opgenomen, maar komen niet prominent in beeld zoals dat bij (*Extended Validation* of) EV-certificaten het geval is. Ook hier toont de browser een slot-icoontje én zijn de bedrijfsgegevens in het certificaat opgenomen, maar ze komen niet prominent in beeld op de browser, bv.:

   Uiterlijk zie je geen verschil met domeinvalidatie, maar het certificaat geeft wel meer details vrij.

3. **Uitgebreide validatie** (EV: *Extended Validation*): Naast de domeinvalidatie waarbij de aanvrager laat zien controle te hebben over het domein waarvoor het certificaat wordt aangevraagd, worden ook de bedrijfsgegevens gecontroleerd. Er wordt hiervoor naar een openbaar register gekeken, en er kan ter verificatie naar de organisatie worden gebeld. Soms is het noodzakelijk dat er bijkomende documenten worden ondertekend. Een website die gebruik maakt van EV SSL-certificaten vertoont een groene balk met het slot-icoontje, bv.:

Buiten de groene kleur en het slot-icoontje toont de browser ook de identificatie van de organisatie wanneer op het slot-icoontje geklikt wordt. Dit soort validatie wordt typisch gebruikt door financiële instellingen en webshops.

Specifiek voor de beveiliging van websites, zijn er – naast de drie validatievarianten – ook nog eens drie verschillende types SSL-certificaten:

1. **Enkel-domein**: dit type certificaten beschermt één enkel domeinnaam, bv. 'www.eendomein.com';
2. **Multi-domein**: hiermee is het mogelijk meerdere domeinnamen binnen één SSL-certificaat; en
3. **Wildcard-**certificaten: met een wildcard-certificaat worden alle subdomeinen van één domein beveiligd. Wildcard-certificaten zijn enkel beschikbaar bij domein- en organisatievalidatie, bij uitgebreide validatie (EV-certificaten) moet elk subdomein een eigen certificaat krijgen en kan men dus geen wildcard-certificaten toepassen.

PKI en CA

Een *Public Key Infrastructure* (PKI) is een set van technische en organisatorische voorzieningen, die een oplossing biedt voor het probleem van koppeling van een eigenaar (persoon of organisatie) aan zijn/haar crypto-sleutelpaar door middel van het digitale certificaat. De uitgifte en het beheer rond deze certificaten wordt op een geformaliseerde wijze uitgevoerd, zodat de status van het certificaat en de eigenaar gegarandeerd is. Daarmee kunnen publieke sleutels in combinatie met de betreffende certificaten gebruikt worden voor authenticatie en het versturen van geheime (sleutel)informatie over een niet vertrouwd netwerk.

Een PKI is dus een uitbreiding op de asymmetrische encryptietechniek die ervoor zorgt dat authenticatie en onweerlegbaarheid aantoonbaar wordt.

Certificaten worden door een (derde) partij uitgegeven, die zowel door zender als ontvanger vertrouwd wordt: een *Certificate Service Provider* (CSP). De CSP beheert de PKI-omgeving en garandeert de echtheid en de oorsprong van de certificaten. Een CSP moet zelf ook aan kwaliteitseisen voldoen. Certificaten kunnen in allerlei vormen worden uitgegeven. De bijbehorende private sleutels moeten beschermd zijn tegen niet-bevoegde toegang en worden bij voorkeur uitgegeven en opgeslagen op afzonderlijk te beveiligen objecten als smartcards, USB-tokens, en *Hardware Security Modules* (HSM).

De componenten van een CSP worden toegelicht in bijlage 2.

Er zijn verschillende soorten van PKI in gebruik en daaraan gekoppeld bestaan verschillende soorten van uitgifte processen van certificaten door CSP's:

- Binnen een **organisatie**: hier worden certificaten uitgegeven door een eigen CSP;
- Binnen het **publieke domein**: uitgifteprocessen moeten hierbij voldoen aan de wet eIDAS en elektronische archivering[1];

[1] Wet van 21 juli 2016 Wet tot uitvoering en aanvulling van de verordening (EU) nr. 910/2014 van het Europees Parlement en de Raad van 23 juli 2014 betreffende de elektronische identificatie en vertrouwensdiensten voor elektronische transacties in de interne markt.

- **Web van vertrouwen** (*web of trust*): in dit model wordt de controle van de identiteit uitgevoerd door de certificaathouders (de eigenaars van de certificaten). Ze staan zelf in voor de authenticiteit van de identiteit van de andere certificaathouders. Voorbeelden hiervan zijn Thawte, CAcert en PGP.

Om een digitaal certificaat te verkrijgen van een CSP, moeten een aantal technische stappen doorlopen worden. Dit wordt uitgelegd in bijlage 3.

2.4 CA en CAA

Het vertrouwen in digitale certificaten is maar zo groot als het vertrouwen dat kan gesteld worden in de uitgevende CA. Niet alle CA's verdienen een even groot vertrouwen. Een organisatie kan afdwingen welke CA's SSL-certificaten mogen uitgeven en welke niet. Ze doen dit via een techniek genoemd CAA (*Certificate Authority Authorisation*). CAA is gebaseerd op en gebruikt DNS ('Domain Name System') door middel van een DNS CAA-record. Hiermee kan de domeineigenaar specifiëren welke root CA's een certificaat voor het domein mogen aanvragen en ondertekenen, in lijn met het veiligheidsbeleid van de organisatie. Dit voorkomt uitgifte van certificaten door CA's die niet door de organisatie werden goedgekeurd.

CAA voegt aldus een extra beveiligingslaag toe door middel van de DNS-omgeving. Maar DNS is van zichzelf geen sterk beveiligd systeem. Hierdoor is het niet ondenkbaar dat onbevoegden DNS-records kunnen aanpassen. Met DNSSEC kan het beveiligingsniveau van de DNS-omgeving verhoogd worden. DNSSEC (*DNS Security Extensions*) is een beveiligingstechniek waarbij de DNS-records gevalideerd worden door gebruik te maken van een digitale handtekening. Door middel van deze digitale handtekening kan een *DNS-server* aantonen dat de informatie afkomstig is van een gevalideerde bron en dat deze niet is gewijzigd.

Blockchain

Blockchain is een systeem dat gebruikt kan worden om gegevens vast te leggen. Dit kunnen bijvoorbeeld overschrijvingen zijn zoals die bij een reguliere bank mogelijk zijn, maar een blockchain kan ook eigendomsaktes, afspraken, persoonlijke berichten of andere gegevens bevatten. Het bijzondere aan blockchain is dat dit mogelijk is zonder centrale autoriteit waardoor het vervalsen van de vastgelegde gegevens vanuit één centraal punt niet mogelijk is. Een blockchain bestaat uit een keten vastgelegde en samengevoegde gegevens, blokken (*blocks*) genoemd. De keten van gegevens wordt bepaald door de volgorde waarin gegevens worden toegevoegd.

Een blockchain is een openbare database, waarin alle transacties die zijn gedaan via het netwerk zijn opgeslagen. Deze database wordt niet beheerd door één bedrijf of persoon, maar door alle computers die deel uitmaken van het netwerk. Iedere computer heeft een kopie van de boekhouding en elke nieuwe transactie wordt gezamenlijk geverifieerd.

Dat heeft een aantal voordelen. Er kan minder snel gesjoemeld worden, aangezien elke wijziging in de database moet worden goedgekeurd door de spelers in het netwerk. Elke transactie wordt in een *block* vastgelegd en aan de bestaande blokketen toegevoegd. Zo is elke wijziging die ooit gemaakt is voor iedereen zichtbaar. Dat is een stuk transparanter en veiliger dan bij een centrale database.

De techniek steunt op de eigenschappen van een *hash*: deze verbindt de blokken van een blockchain als schakels in de keten, als het vorige blok wordt gewijzigd is dit gemakkelijk vast te stellen doordat de *hash* dan niet meer klopt. Een blok wordt dus gekoppeld door middel van een verwijzing naar een vorig blok. Deze verwijzing is eigenlijk een *hash* van de *header* van het vorige blok.

Een blockchain is dus een specifieke toepassing van cryptografie.

De bekendste implementatie van de blockchain is de cryptomunt, waarvan de Bitcoin meest genoemd, maar er is ook sprake van toepassing bij verkiezingen *(e-voting*): stemmen via een decentraal gedistribueerd netwerk waarbij de blockchain-techniek voor de nodige integriteit zorgt. Anonimiteit wordt ingebouwd via een speciaal hiervoor voorzien algoritme.

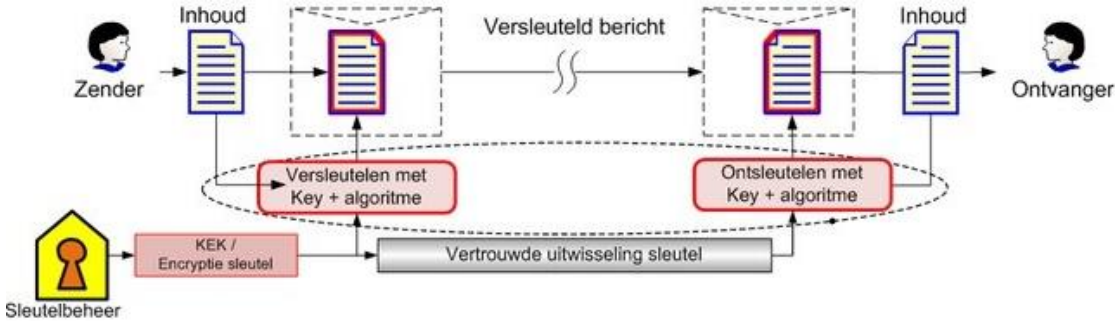
2.5 De bouwstenen

2.5.1 Bouwstenen voor vertrouwelijkheid

Berichten en bestanden worden op verschillende manieren via de niet vertrouwde 'buitenwereld' uitgewisseld:

- Eindgebruikers versturen berichten via e-mail over het interne;
- Eindgebruikers plaatsen bestanden op draagbare media (USB-stick, cd-rom, dvd, SD-kaarten etc.) die ze meenemen buiten de organisatie;
- Berichten worden via openbare netwerken naar een semi-vertrouwde of niet-vertrouwde client verstuurd, bijvoorbeeld voor telewerken of mobiel werken;
- Bij gebruik van draadloze verbindingen binnen de omgeving van de organisatie, waarvan te verwachten is dat ze buiten het gebouw te ontvangen zijn (zoals wifi);
- Er worden berichten uitgewisseld via openbare netwerken tussen systemen van de organisatie op verschillende locaties of met die van vertrouwde partners;
- Berichten en bestanden worden opgeslagen op een mobiele client (laptop, smartphone) die meegenomen wordt buiten de organisatie.

Asymmetrische encryptie maakt het mogelijk dat communicerende entiteiten elkaar geheime berichten toezenden zonder vooraf geheime sleutels uit te wisselen. Zo kan je in theorie in een online-communicatie een bericht encrypteren d.m.v. de publieke sleutel van de bestemmeling. De bestemmeling kan dan het ontvangen vercijferde bericht gaan decrypteren d.m.v. de bijhorende private sleutel. In de praktijk werkt men echter anders omdat asymmetrische encryptie tijdrovend is en dus nadelig voor de performantie: het bericht wordt versleuteld met een symmetrische sleutel die op zijn beurt versleuteld wordt met de publieke sleutel van de ontvanger. Die kan dan de symmetrische sleutel ontcijferen met zijn/haar private sleutel waardoor het originele bericht ontcijferd wordt aan de hand van de symmetrische sleutel.



Bij voorkeur vindt de symmetrische encryptie transparant voor de eindgebruiker plaats. Omdat symmetrische encryptie zich vaak afspeelt op het niveau van verbindingen, netwerken en systemen is dat ook meestal het geval. Bekende voorbeelden hiervan zijn de VPN's (*Virtual Private Network*), TLS (*Transport Layer Security*) en wifi-encryptie (*Wifi Protected Access* of WPA).

Daar waar de encryptie zich op applicatieniveau afspeelt, is vaak interactie van de eindgebruiker vereist. Voorbeelden hiervan zijn e-mailencryptie en aparte software voor encryptie van bestanden om deze op draagbare media op te slaan of als bijlage met e-mail te versturen. De encryptie is uiteindelijk zo sterk als de mate waarin de cryptografische sleutel geheimgehouden kan worden voor onbevoegden. Voor de sterkte van de encryptie spelen de volgende factoren een cruciale rol:

- Encryptiealgoritme;
- Sleutellengte;
- Distributie van sleutels; en
- *Lifecyclemanagement* van sleutels.

Encryptiealgoritme en sleutellengte

Het meest robuuste en vaak toegepaste encryptiealgoritme is AES (*Advanced Encryption Standard*). Het AES-algoritme is geschikt voor de sleutellengtes van 128, 192 of 256 bits. De sleutellengte en de kwaliteit van de sleutel bepalen in belangrijke mate de tijdsduur die nodig is voor het 'kraken' van de encryptie.

Distributie van sleutels

Voor distributie van sleutels worden vaak weer andere encryptiemechanismen ingezet met de bijbehorende sleutels. Samen met de sleutels voor distributie en beheer worden er drie soorten cryptografische sleutels onderscheiden: *Key Encryption Keys* (KEK), periodieke sleutels en sessiesleutels. De eigenschappen van deze sleutels zijn in onderstaande tabel samengevat.

Eigenschap	Key Encryption Key	Periodieke sleutel	Sessiesleutel
Doel	Encryptie van de periodieke of sessiesleutel	Symmetrische encryptie van de gevoelige gegevens	Symmetrische encryptie van de gevoelige gegevens
Soort	Publieke sleutel of geheime sleutel	Geheime sleutel	Geheime sleutel
Levensduur	1 jaar	Vastgestelde periode	1 sessie
Distributie	Fysiek (smartcard, cd-rom, sleutellaadapparaat, papier) over vertrouwd pad	Beveiligd met KEK over communicatiepad zelf of over ander onvertrouwd pad	Beveiligd met KEK over communicatiepad zelf
		Fysiek over vertrouwd pad (geen KEK)	

Afhankelijk van de toepassing zijn er verschillende methoden om sleutels veilig te distribueren en deze worden toegelicht in bijlage 4.

Voorbeelden van versleuteling voor vertrouwelijkheid zijn:

- E-mail encryptie: PGP, S/MIME;
- Encryptie van webverkeer: TLS;
- VPN: IPsec; en
- Wifi-encryptie: WPA, WPA2.

2.5.2 Bouwstenen voor integriteit

Een cryptografische *hash*-functie, kortweg een *hash*, is een andere cryptografische bouwsteen die kan zorgen voor integriteit. Hoewel er geen geheime sleutels worden gehanteerd bij een *hash*, spreekt men vaak ook van encryptie, namelijk één-richtingvercijfering.

Een *hash*-functie neemt als input een bericht van willekeurige lengte en genereert een code, de *hash*-waarde, die specifiek is voor dat bericht. Elke wijziging van het bericht leidt tot een wijziging in de *hash*-waarde. Bovendien is het niet mogelijk om vanuit een bepaalde *hash*-waarde het bericht te reconstrueren (t.t.z. 'niet mogelijk' betekent hier dat het rekenkundig niet haalbaar is om dit in redelijke tijd te doen).

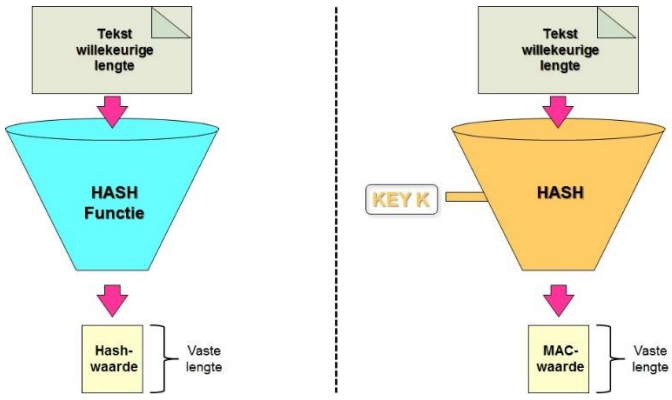
Een *hash*-functie kan dus de integriteit van informatie garanderen op voorwaarde dat de *hash*-waarde zelf correct beschermd is tegen manipulatie. Zo kan bijvoorbeeld in de context van DIM (Data in Motion = transport van data) de integriteit van een verzonden bericht worden aangetoond door de *hash*-waarde via een ander communicatiekanaal te versturen, of geëncrypteerd mee te sturen.

Hash-functies worden gecombineerd met asymmetrische encryptietechnieken om een digitale handtekening te realiseren (zie verder). Een digitale handtekening levert de volgende veiligheidsdiensten: bericht-integriteit, authenticatie van de verzender, data-authenticatie en onweerlegbaarheid.

Bij een MAC (*Message Authenticatie Code*), soms ook gesleutelde *hash*-functie genoemd, wordt een *hash*-waarde of MAC-waarde gegenereerd op basis van een geheime sleutel. Naast bericht-integriteit, wordt ook een (beperkte) vorm van data-authenticiteit gerealiseerd (garanties over de bron van het verzonden bericht). De geheime sleutel dient in dit geval wel vooraf uitgewisseld te worden.

Bij een MAC wordt de bescherming van de authenticiteit van een bericht (informatie-element) dus herleid tot het geheimhouden van een sleutel (zie ook bijlage 5).

Hash-functie vs. MAC:



Sleutelbeheer

Het beheer van cryptosleutels speelt een essentiële rol bij beveiliging op basis van cryptografische technieken. Cryptosleutels zijn alle sleutels voor encryptie als vertrouwelijkheidsmaatregel, digitale handtekening, authenticatie, enz. De mate van bescherming die cryptografie biedt hangt behalve van het gebruikte algoritme of protocol tevens af van de geheimhouding van het sleutelmateriaal (geheime sleutel, private sleutel) en de authenticiteit van de publieke sleutels.

Het sleutelbeheer omvat het aanmaken, registeren, opslaan, distribueren, in gebruik nemen, herroepen, archiveren en vernietigen van sleutels. Voor al deze aspecten zijn processen en procedures nodig. Een combinatie van organisatorische, logische en fysieke beveiligingsmaatregelen moet worden ingezet om van het sleutelbeheer een succesverhaal te maken. Een aantal specifieke maatregelen zijn hierbij nodig, zoals bijvoorbeeld fysieke en logische beveiligingsmaatregelen bij het aanmaken en opslaan sleutels, en functiescheiding om misbruik van sleutels te voorkomen en te detecteren.

Hoe het sleutelbeheer wordt ingericht is voor een belangrijk deel afhankelijk van het gewenste beveiligingsniveau, de schaalgrootte en de verscheidenheid waarop encryptie wordt toegepast plus het belang van de ermee versleutelde gegevens bepaald door de classificatieschaal van de informatie die verwerkt wordt.

- Sleutelbeheer moet antwoord geven op volgende vragen:
  - Hoe moet het aanvragen van een sleutelpaar verlopen?
  - Wie mag sleutels genereren?
  - Op welke manier worden de sleutelparen overgedragen aan de eigenaar?
  - Moet tijdens de overdracht van het sleutelpaar de eigenaar zich legitimeren?
  - Hoe lang zijn de sleutels geldig?
  - Wie kan sleutels intrekken?
  - Hoe worden sleutels geüpdatet?
- Sleutelbeheer omvat volgende activiteiten:
  - Opvolging van de levensduur van sleutelparen;
  - Archivering van verlopen sleutelparen om ontcijfering van informatie mogelijk te houden ook als een sleutelpaar verlopen is;
  - Aanvragen en genereren van sleutelparen en certificaten;
  - Herroepen (*revocation*) van sleutelparen;
  - Veilige bewaring van private en symmetrische sleutels;
  - Back-up van sleutels waar nodig;
  - Distributie van sleutels;
  - Vervangen en update van sleutels; en
  - Vernietigen van sleutels.

Naarmate de informatieclassificatieschaal hoger is, zullen procedures strikter zijn en de mate van functiescheiding toenemen. Een risicoanalyse kan inzicht geven in welke risico's dienen te worden afgedekt met organisatorische en/of technische maatregelen.

Betrouwbare distributie van sleutels is cruciaal. Bij symmetrische systemen worden geheime sleutels gedistribueerd, bij asymmetrische systemen publieke sleutels. In beide gevallen dient de authenticiteit van de sleutels gegarandeerd te zijn:

- Werd de sleutel niet gewijzigd tijdens het distributieproces?
- Is de sleutel authentiek, dus niet afkomstig van iemand die zich als de vermeende afzender voordoet, de zogenaamde *man in the middle*?

De problematiek van betrouwbare sleuteldistributie kan worden opgelost door gebruik te maken van een betrouwbare tussenpersoon.

Sleutelhiërarchie

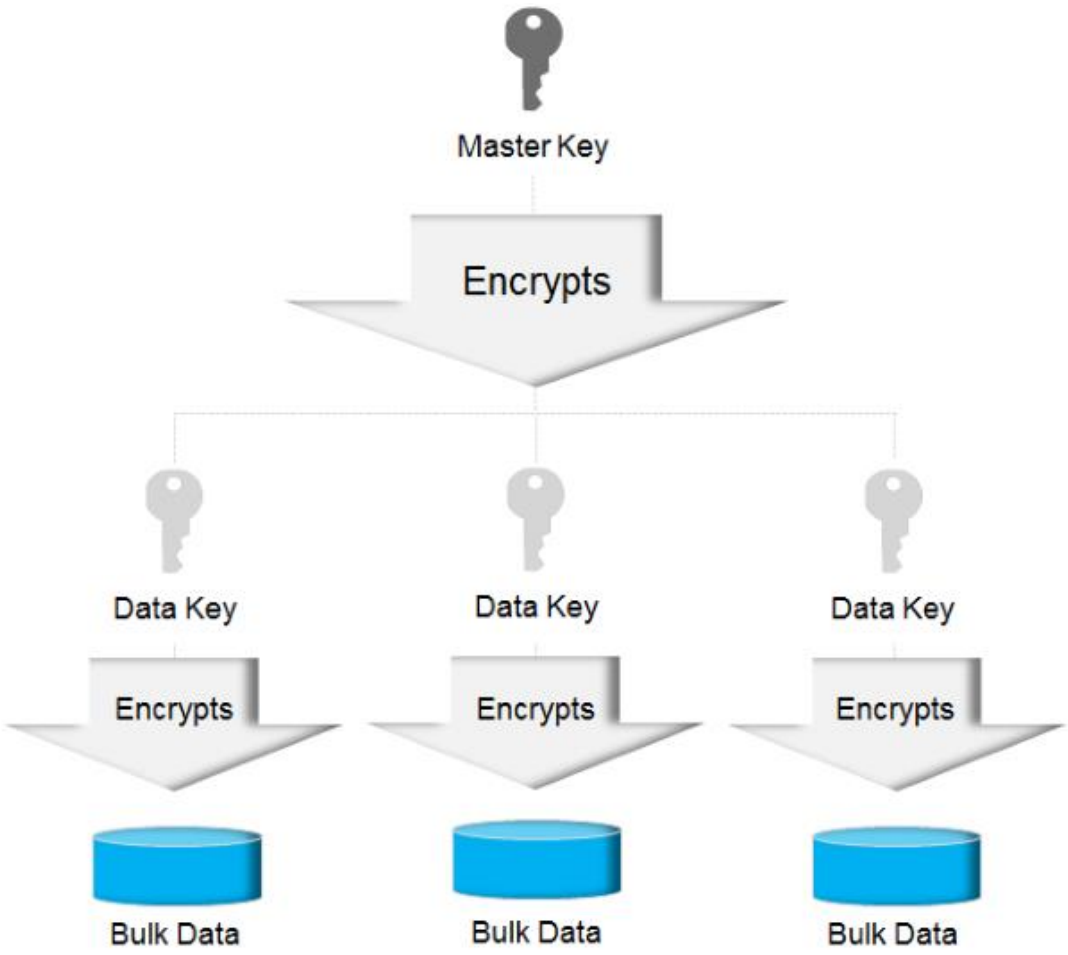
Er zijn verschillende mogelijkheden om cryptosleutels te genereren, sommige zijn *open* source (vaak kosteloos), andere zijn leverancier specifiek (niet kosteloos). Maar zodra een cryptosleutel is gegenereerd en gebruikt, moet deze sleutel veilig bewaard worden voor later gebruik. Opslag van deze sleutel is een lastig punt.

Een oplossing hiervoor kan gevonden worden in sleutelhiërarchie. Dit is een techniek waarbij een *root key* of *master key* (verder 'mastersleutel' genoemd) gebruikt wordt om de cryptosleutel op zijn beurt te versleutelen. Op deze wijze voorziet sleutelhiërarchie in een krachtige methode om andere cryptosleutels te beveiligen. Immers, het is dan afdoende om deze mastersleutel heel goed te beveiligen om de betrouwbaarheid van de andere sleutels te garanderen.

Dit is dan ook het zwakke punt van sleutelhiërarchie: het is van het grootste belang om de mastersleutel zeer goed te beveiligen: als deze sleutel gehackt wordt, zijn alle onderliggende sleutels eveneens gecompromitteerd!

Dit betekent o.a. dat deze mastersleutel best bewaard wordt in een *Hardware Security Module* of HSM-module bijvoorbeeld FIPS 140 gekeurd. Dit keurmerk waarborgt de goede beveiliging van de mastersleutel.

Volgende figuur legt uit hoe sleutelhiërarchie werkt:



Dit zijn de voordelen van sleutelhiërarchie:

- De hoeveelheid sleutels die hoge beveiliging beogen is gereduceerd tot de beveiliging van de mastersleutel (en dit is tevens het zwakke punt: als de mastersleutel gekraakt wordt, zijn alle onderliggende sleutels in gevaar);
- Door gebruik te maken van één mastersleutel is het eenvoudiger om verschillende sleutels te gebruiken voor de beveiliging van verschillende stukken informatie. Daardoor wordt het mogelijk om verschillende sleutels te gebruiken voor bijvoorbeeld verschillende groepen gebruikers en is de impact van het verlies van zo'n sleutel beperkt tot groepen van gebruikers; en
- Gebruik van sleutelhiërarchie heeft ook een gunstig gevolg voor de performantie: encryptie/decryptie moet immers niet langer uitgevoerd worden in de HSM. Enkel de mastersleutel moet behandeld worden in de HSM, waardoor bulk encryptie/decryptie sneller verloopt.

Een diepere hiërarchie voor sleutelbeheer is ook mogelijk: een HSM-beveiligde mastersleutel beveiligt een organisatorische sleutel die op zijn beurt een aantal bulk-encryptiesleutels die doorheen de organisatie gebruikt worden voor de beveiliging van bulkinformatie.

Tijdsstempel (*Time stamping*)

Een elektronische tijdsstempel is een techniek die hand in hand gaat met digitale handtekening.

Digitale handtekeningen zijn in de eerste plaats ontworpen voor validatie direct na ondertekening. Uit veiligheidsoverwegingen heeft een sleutelpaar en dus het bij de publieke sleutel horende certificaat een beperkte geldigheidsduur. Bovendien is de gebruikte techniek continu in ontwikkeling. Gebruikte algoritmes kunnen op den duur gekraakt worden door krachtiger wordende computers. Daarom worden er steeds sterkere algoritmes ingezet. Dit maakt digitale handtekeningen minder geschikt voor de lange termijn. Maar voor bijvoorbeeld digitale facturen en andere archiefdocumenten is juist validatie op lange termijn nodig.

De geldigheidsduur van een digitale handtekening is afhankelijk van de geldigheid van het gebruikte digitale certificaat. Als de geldigheid van dit certificaat is verstreken, geeft de digitale handtekening een foutmelding. Ook het bovenliggende rootcertificaat heeft een geldigheid van bepaalde duur. Een certificaatautoriteit (CA) kan ook ophouden met bestaan. In al deze gevallen is validatie van een ondertekend document niet meer mogelijk. Het gebruik van een digitale handtekening in combinatie met een elektronische tijdsstempel voorkomt foutmeldingen. De elektronische tijdsstempel bewijst dat het certificaat ten tijde van de ondertekening wel degelijk geldig was, hierdoor verloopt een digitale handtekening met elektronische tijdsstempel nooit. Dit maakt validatie op lange termijn ook zonder een geldig certificaat, rootcertificaat of actieve CA mogelijk.

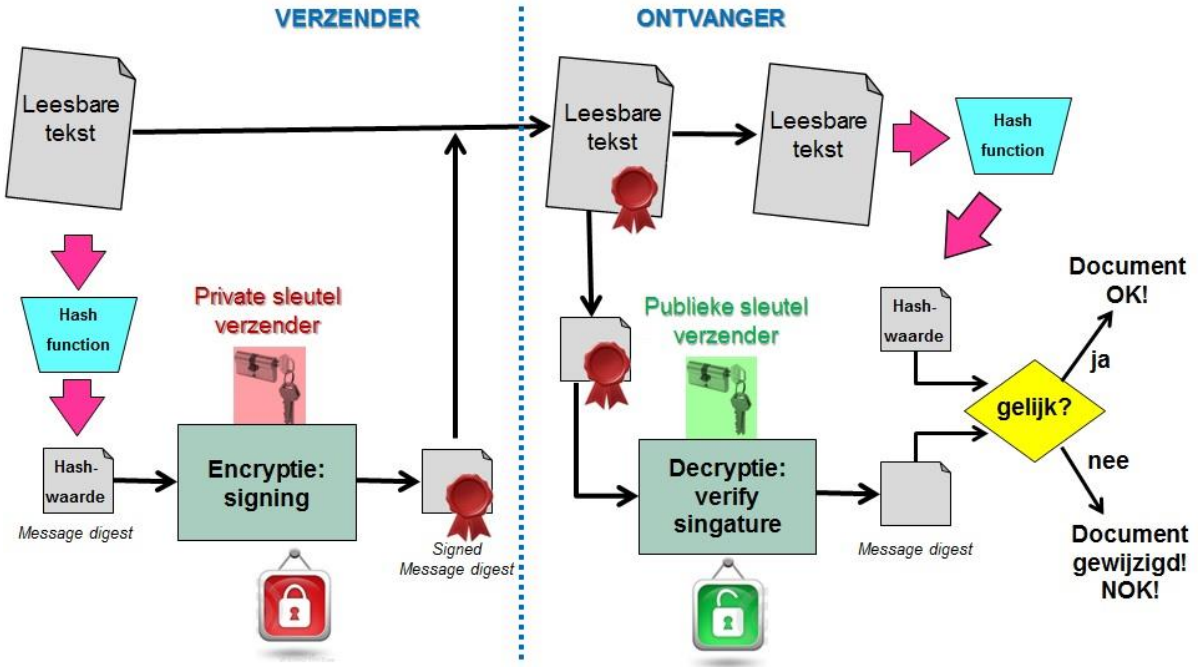
Elektronische tijdsstempels wordt ook gebruik in geavanceerde loggingtechnieken om het tijdsstip van een *log event* vast te leggen.

Het spreekt voor zich dat voor elektronische tijdsstempels een betrouwbare bron moet worden gebruikt. Daarom worden de klokken van *servers* vaak gesynchroniseerd met een externe, betrouwbare tijdsbron, bijvoorbeeld een atomaire klok. Het is mogelijk om in te schrijven op een dienst (*AWS Amazon Time Sync Service* is een voorbeeld) die de tijd van zo'n atomaire klok aanlevert, waardoor de eigen interne klokken eveneens betrouwbaar worden.

2.5.3 Bouwstenen voor onweerlegbaarheid

Een bericht kan versleuteld worden d.m.v. de private sleutel en dan ontcijferd d.m.v. de bijhorende publieke sleutel (dit is dus de omgekeerde beweging van encryptie voor vertrouwelijkheid). Omdat de private sleutel wordt geheimgehouden door de eigenaar ervan, is versleuteling met een private sleutel de basis voor een digitale handtekening. Dit is het meest bekende gebruik van asymmetrische cryptografie geworden. Gezien enkel de eigenaar beschikt over de private sleutel, kan hij niet ontkennen dat hij het bericht heeft versleuteld. Onweerlegbaarheid van data wordt dus gegarandeerd.

Indien men de asymmetrische encryptietechniek nog combineert met het gebruik van een *message digest*/*hash*, kan men tevens berichtintegriteit garanderen en dan spreekt men van een **digitale handtekening**. Alvorens het bericht te versleutelen d.m.v. de private sleutel, wordt het bericht eerst via een cryptografische *hash*-functie gecomprimeerd:



Een digitale handtekening kan gecombineerd worden met asymmetrische encryptie om vertrouwelijkheid van informatie te garanderen: eerst zal de verzender/ondertekenaar zijn/haar digitale handtekening plaatsen op het document met de eigen private sleutel, vervolgens wordt het volledige pakket (document + handtekening) versleuteld met de publieke sleutel van de bestemmeling om het geheel onleesbaar te maken voor deren. In de praktijk echter zal men voor dit laatste meestal symmetrische encryptie gebruiken omwille van de snelheid ervan.

Tijdens transport en opslag vormt het onopgemerkt wijzigen van berichten (of wijzigen door onbevoegden) een risico. De ontvanger heeft geen garantie dat het bericht integer is en dat het bericht afkomstig is van de identiteit, die als ondertekenaar bij het bericht staat vermeld (onweerlegbaarheid).

In het algemeen valt het 'zetten' van de digitale handtekening uiteen in twee delen, wat leidt tot een unieke relatie tussen het bericht en de handtekening en biedt daarmee herleidbaarheid.

- Vastleggen van de unieke kenmerken van het bericht (in een *hash*).
- Verbinden van de unieke identiteit van de zender aan de *hash*.

De unieke identiteit van digitale handtekeningen kan met behulp van verschillende mechanismen worden verbonden met het controlegetal, waarvan de bekendste zijn:

- Symmetrische cryptografische sleutels = vooraf uitgedeeld door regiepartij; en
- Asymmetrische cryptografie op basis van PKI (*Public Key Infrastructure*) = uitgedeeld door een TTP (*Trusted Third Party*).

De mate van zekerheid die uit de toegepaste methode voortvloeit, wordt sterk beïnvloed door de kwaliteit van de aard en toepassing van algoritmen en methoden en vooral van:

- Toevalsgetallen;
- Uniciteit en lengte van sleutels en toegangscodes;
- Sleuteluitgifte-, distributie- en bewaarprocessen en middelen; en
- Kwalificatie van de certificaatuitgifte.

Onderstaande tabel geeft aan welke verbanden er bestaan tussen het zekerheidsniveau, de toegepaste sleutels en wie er door zender en ontvanger wordt vertrouwd.

Zekerheidsniveau	Sleutelmodel	Proceskwaliteit	Zender en ontvanger vertrouwen:
1: Laag, onzekere bewaartermijn	Symmetrisch	Gedeelde sleutels	Eigen organisatie of partner
2: Middel, onzekere bewaartermijn	Asymmetrisch	PKI-service of private PKI	Eigen organisatie of partner
3: Hoog, gegarandeerde bewaartermijn	Asymmetrisch	Gecertificeerde PKI	Overheid of gecertificeerde partij

Integriteit en onweerlegbaarheid worden geïllustreerd in bijlage 6.

2.5.4 Bouwstenen voor authenticatie

De digitale handtekening in combinatie met een certificaat vormt de basis voor authenticatie van entiteiten (personen, apparatuur en organisaties) door de koppeling van de publieke sleutel aan een entiteit en de verificatie van de identiteit. De technieken voor onweerlegbaarheid en authenticatie gaan dus hand in hand, namelijk door gebruik te maken van de digitale handtekening en het certificaat:

- Authenticatie door middel van bewijs van identiteit aan de hand van een digitale handtekening en certificaat;
- Onweerlegbaarheid door verificatie van certificaat en digitale handtekening gebruikt voor het ondertekenen van een document of bericht.

De digitale handtekening in combinatie met een certificaat kan vergeleken worden met een gewone, met de hand gezette, handtekening. Het doel van een met de hand gezette handtekening is authenticatie van de ondertekenaar, een mogelijkheid tot verificatie hiervan door de ontvanger, en de handtekening kan tevens worden gebruikt om onweerlegbaarheid en integriteit te garanderen.

Authenticatie aan de hand van een certificaat behoort tot de sterke authenticatiemiddelen, namelijk door multifactorauthenticatie, waarbij minstens twee authenticatievormen worden afgedwongen:

- Iets wat je weet, bv. een paswoord;
- Iets wat je bezit, bv. een token, USB-sleutel of certificaat;
- Iets wat je bent (een persoonlijke eigenschap), bv. een vingerafdruk of retina scan.

Om deze doelen ook met behulp van een digitale handtekening te garanderen, moet aan de digitale handtekening een aantal eisen worden gesteld, namelijk:

- De handtekening moet uniek zijn om de maker van de handtekening te kunnen verifiëren;
- De handtekening moet de inhoud van het bericht kunnen authentiseren; en
- De handtekening moet door derden kunnen worden gecontroleerd, om eventuele problemen met betrekking tot de onweerlegbaarheid op te lossen.

Om de nodige garanties te kunnen geven zal een derde partij een soort stempel moeten zetten voor het garanderen van de echtheid van de sleutel, en de koppeling van de sleutel aan de juiste persoon. Een voorbeeld uit het niet-elektronische leven is een paspoort. Het aanvragen van een paspoort verloopt via een door de overheid opgestelde procedure. Het paspoort dient bijvoorbeeld in het buitenland als bewijs van de identiteit van de reiziger. Dit is ook de functionaliteit die wordt gezocht voor de digitale handtekening en hiervoor wordt het digitale certificaat gebruikt. Hoe dit praktisch gebeurt, word toegelicht in bijlage 6.

Vaak wordt een apart certificaat gebruikt voor authenticatie en integriteit/onweerlegbaarheid. Dit is bijvoorbeeld zo bij eID. De door de Belgische federale overheid uitgegeven elektronische identiteitskaart eID werkt op basis van twee certificaten: eentje voor authenticatie (bewijs door middel van een digitale handtekening) en eentje voor integriteit/onweerlegbaarheid (door middel van een wettelijk geldende, want gekwalificeerde digitale handtekening).

Digitale certificaten worden niet alleen gebruikt om personen te authentiseren, maar kunnen ook worden toegekend aan websites en apparatuur, zoals *servers*, *routers*, enz. Digitale certificaten kunnen worden onderverdeeld in twee varianten, server- en clientcertificaten:

1. Een **servercertificaat** wordt gebruikt door een *server*, bijvoorbeeld een *webserver*, om zich te authentiseren en om een versleutelde verbinding tussen client en *server* op te zetten. Op het moment dat een client een veilige https-verbinding op wil zetten, stuurt de *server* een digitaal certificaat naar de client. Dit digitale certificaat bevat als subjectnaam de domeinnaam van de *server*. Bij een correcte authenticatie moet deze domeinnaam in het digitale certificaat overeenkomen met de domeinnaam waar de client een verbinding mee op wil zetten; en
2. Een **clientcertificaat** wordt gebruikt door een eindgebruiker die dit certificaat kan gebruiken om zichzelf te authentiseren met behulp van dit clientcertificaat.

Om rechtsgeldig te zijn in België moeten digitale handtekeningen voldoen aan de wet van 21 juli 2016, de wet ‘eIDAS en elektronische archivering’ genoemd. Deze wet is een concrete vertaling en invulling van de Europese eIDAS-verordening (*electronic IDentification Authentication and trust Services*).

Hierin wordt o.a. de definitie van een digitale handtekening vastgelegd: ‘*gegevens in elektronische vorm bevat die gehecht zijn aan of logisch verbonden zijn aan andere gegevens in elektronische vorm en die door de ondertekenaar worden gebruikt om te ondertekenen’* (art. 3 §10 van de verordening). Deze definitie omvat niet alleen handtekeningen op basis van digitale certificaten, maar ook andere soorten digitale handtekeningen, zoals handgeschreven gescande handtekeningen, biometrische handtekeningen (bijvoorbeeld stemherkenning, irisherkenning, herkenning van vingerafdrukken), digitale handtekeningen of de codes van bankkaarten.

Wanneer een digitale handtekening aan bepaalde eisen voldoet, kan ze ‘geavanceerd’ of ‘gekwalificeerd’ zijn.

Om **geavanceerd** te zijn, moet de digitale handtekening:

- Op unieke wijze met de ondertekenaar verbonden zijn;
- Het mogelijk maken om de ondertekenaar te identificeren;
- Tot stand gekomen zijn met gegevens voor het aanmaken van digitale handtekeningen die de ondertekenaar, met een hoog vertrouwensniveau, onder zijn uitsluitende controle kan gebruiken; en
- Op zodanige wijze aan de daarmee ondertekende gegevens verbonden zijn, dat elke wijziging achteraf van de gegevens kan worden opgespoord (art. 26 van de eIDAS-verordening).

Een digitale handtekening is **gekwalificeerd** indien zij niet alleen geavanceerd is, maar ook aangemaakt is met een gekwalificeerd middel voor het aanmaken van digitale handtekeningen en gebaseerd is op een gekwalificeerd certificaat voor digitale handtekeningen (art. 3 §12 van de eIDAS-verordening).

De verordening specificeert dat een digitale handtekening (ongeacht de gebruikte technologie en het niveau) niet als bewijsmiddel in juridische procedures mag worden geweigerd, louter om de reden dat ze elektronisch is of niet gekwalificeerd. Nochtans wordt alleen de gekwalificeerde handtekening gelijkgesteld met een handgeschreven handtekening (art. 25 van de eIDAS-verordening).

In hoofdstuk '*De kracht van het certificaat*' werd besproken wat EV of uitgebreide validatiecertificaten zijn. Ter herinnering: een EV-certificaat volgt de X.509-standaard en voldoet aan bepaalde eisen omtrent identiteitscontrole volgens de richtlijnen voor uitgave en beheer van EV-certificaten. Deze richtlijnen zijn bedoeld om betere verzekering te bieden omtrent de identiteit van de certificaathouder door het afdwingen van uniforme en gedetailleerde validatieprocedures. De meest voorkomende toepassing van EV-certificaten zijn voor SSL, waarbij de identiteit van de website gecontroleerd is.

Wat is nu het verschil met een gekwalificeerd certificaat? Gekwalificeerde certificaten zijn ontworpen om te voldoen aan de eisen van de eIDAS-verordening. Deze mogen alleen worden uitgegeven aan natuurlijke personen, in hun persoonlijke hoedanigheid als bedrijfsmatige/organisatorische vertegenwoordiger (als deze organisatie, relatie en autoriteit ook zijn geverifieerd). Gekwalificeerde elektronische handtekeningen, waarbij een gekwalificeerd certificaat wordt gebruikt, zijn geldig als bewijs en hebben direct wettelijke effect zoals handgeschreven handtekeningen.

Toepassingen in verschillende data context

DIU (data in use)

Gegevens in tijdelijke opslag: Gevoelige gegevens zoals paswoorden en pincodes worden op systeem- of applicatieniveau versleuteld en de waarden daarvan zijn uitsluitend leesbaar voor bevoegde processen. Deze functies moeten specifiek op applicatie- en systeemniveau zijn ingebouwd.

DIM (data in motion)

Versleuteling op applicatieniveau: Hierbij wordt versleuteling door twee met elkaar communicerende systemen end-to-end uitgevoerd. Alleen het dataveld van een pakket wordt hierbij versleuteld. Een bekende vorm van encryptie op applicatieniveau is de secure-http-techniek, die gebruikt wordt voor versleuteling van http-berichten.

Versleuteling op sessieniveau: zoals *Transport Layer Security* (TLS). Deze techniek wordt gebruikt in combinatie met applicatieprotocollen, zoals http(s), ftp(s), imap(s), pop(s) en smtp(s), herkenbaar aan de 'S'. Als TLS is toegepast bij (http), dan wordt per sessie de webcommunicatie versleuteld (https) en zie je in de statusbalk van de browser een slot-icoontje.

Versleuteling op netwerkniveau: zoals IPSec (*Internet Protocol Security*). Op basis van dit protocol worden versleutelde communicatietunnels gelegd tussen netwerkeindpunten, waarmee veilige communicatie mogelijk is. Met deze tunnels kunnen V*irtual Private Networks* (VPN's) worden gebouwd. Draadloze netwerken worden versleuteld met eigen protocollen als WPA (*Wifi Protected Access*). De versleuteling eindigt op de netwerkcomponent en is dus niet *end-to-end*. Als het eindpunt bijvoorbeeld een *proxyserver* is, dan wordt de communicatie versleuteld tot op de proxy *server*, maar verloopt verder in leesbare tekst tot aan de PC van de gebruiker.

Versleuteling op datalinkniveau: Hierbij wordt op het 'laagste niveau' van het netwerk versleuteling uitgevoerd tussen twee netwerkknooppunten. Alle data die wordt uitgewisseld, dus ook protocolinformatie is daarbij versleuteld. Een voorbeeld van datalink encryptie is het PPTP-protocol (*Point to Point Tunneling Protocol*).

In overzicht:

DAR (data at rest)

Gegevens in opslag: Dit betreft versleuteling van informatie op gegevensdragers op drie niveaus:

1. Encryptie op opslagniveau (*storage*), vaste en mobiele media:
   - Versleuteling van mobiele geheugenmedia, zoals harddisks van laptops, USB-sticks, cd-roms, Tape of insteek-memorymodules maar ook in geheugens van smartphones; en
   - Versleuteling van vaste geheugenmedia, zoals *harddisk arrays* van databases, tape en optische media;
2. Encryptie op database niveau; en
3. Encryptie op applicatie niveau.

2.5.5 Bouwstenen voor beschikbaarheid

Naast het beschikbaar stellen van de nodige apparatuur om te versleutelen en te ontcijferen, is vooral de beschikbaarheid van de cryptografische sleutels en certificaten belangrijk.

Beschikbaar houden van crypto-apparatuur

We hebben het hier over de software en hardware om te kunnen versleutelen, ontcijferen, authenticeren met behulp van cryptografie, digitaal handtekenen, enz.

Om dit te bereiken, worden drie principes toegepast:

- Eliminatie van *single points failure*: ervoor zorgen dat onbeschikbaarheid van één enkele component geen impact heeft op de beschikbaarheid van cryptografie door middel van ontdubbeling en/of inbouwen van redundantie (software en hardware redundantie);
- Betrouwbare *failover* als de hardware of software redundant is opgezet; en

- Snelle en betrouwbare detectie van onbeschikbaarheden: incidenten die te maken hebben met onbeschikbaarheid van componenten moeten tijdig ontdekt en opgelost worden met minimale impact op de eindgebruikers.

Beschikbaar houden van cryptografische sleutels en certificaten

De cryptografische sleutels, nodig om te versleutelen en om te ontcijferen, vormen eveneens belangrijke onderdelen van de cryptografische architectuur, evenals de certificaten gekoppeld aan sleutelparen. Indien de cryptosleutel bij symmetrische encryptie of één dan wel beide sleutels van het sleutelpaar voor asymetrische encryptie ontbreken, is de informatie die versleuteld werd, ontoegankelijk:

- Versleutelde informatie kan niet langer leesbaar gemaakt worden;
- De digitale handtekening kan niet geverifieerd worden; en
- Integriteit en onweerlegbaarheid kunnen niet langer geverifieerd worden.

Sleutel(paren) en certificaten hebben een bepaalde levensduur, ze blijven niet oneindig geldig. Dat heeft gevolgen voor de beschikbaarheid van de versleutelde data: als een sleutel of sleutelpaar niet meer geldig is, is ook de data niet meer toegankelijk. Het is dus belangrijk om een levenscyclus voor het beheer van sleutels en sleutelparen en de daaraan verbonden certificaten in te richten zodat sleutels en certificaten die niet langer geldig zijn tijdig worden vervangen. Maar er moet ook een proces zijn om data die ooit versleuteld werd aan de hand van een eens verlopen sleutel(paar), nadien nog te kunnen ontcijferen.

Een analoge redenering geldt ook wanneer het gaat over verlies van cryptografische sleutels: wanneer de originele sleutel verloren is, kan men de informatie die hiermee versleuteld is, niet meer ontcijferen. In een eerste opwelling zou een organisatie dan kunnen besluiten om back-ups te nemen van cryptografische sleutels, zodat de versleutelde informatie nog kan worden ontcijferd indien de cryptografische sleutel verloren is of ongeldig is geraakt. Maar dit schept potentieel andere problemen zo de versleuteling plaats vond met het oog op authenticiteit en onweerlegbaarheid en kan zelfs in strijd zijn met de wetgeving rond elektronische handtekeningen.

Het is dus belangrijk dat de organisatie goed vastlegt welke sleutel(paren) geback-upt worden, hoelang de levensduur van sleutels en certificaten is per toepassing en wat de *restore*-procedure inhoudt. Indien in het kader van beschikbaarheid geopteerd wordt voor back-up van cryptografische sleutels, is het belangrijk om ook de algoritmen te back-uppen zodat een succesvolle *restore* gegarandeerd wordt.

3 LINK MET ANDERE MAATREGELEN

Toepassen van cryptografie met inbegrip van een adequaat sleutelbeheer levert een essentiële bijdrage om de vertrouwelijkheid van informatie te realiseren, maar mogen niet als geïsoleerde maatregel ingevoerd worden. Vertrouwelijkheid van informatie kan enkel maar optimaal gegarandeerd worden door een geheel van maatregelen toe te passen. Naast cryptografie zijn dit identiteits- en toegangsbeheer (incl. *Access Control* of AC), scheiding van systeemfuncties en traceerbaarheid. Deze maatregelen worden reeds toegelicht in de andere documenten m.b.t informatieclassificatie.

3.1 Link met IAM als maatregel

Verificatie en beheer van identiteiten is een cruciaal onderdeel van het werken met digitale certificaten. Digitale certificaten vormen trouwens een authenticatiemechanisme dat opgenomen is in de minimale maatregelen – IAM (voor meer informatie zie document: *'Vo Informatieclassificatie - Minimale maatregelen – IAM'*). Bovendien is IAM (*Identity and Access Management*) – net als de digitale handtekening – gebaseerd op de eIDAS-verordening.

3.2 Link met functiescheiding als maatregel

Functiescheiding is een organisatorische controlemaatregel. Het implementeert een passend niveau van scheiding van rechten als een veiligheidsprincipe en heeft als voornaamste doelstelling het voorkomen van fraude en fouten. Deze doelstelling wordt bereikt door de taken en bijbehorende rechten voor een specifiek bedrijfsproces over meerdere organisaties, rollen, individuen en/of accounts te spreiden.

3.3 Link met logging als maatregel

Cryptografische maatregelen worden ook toegepast om (gevoelige) gegevens in logbestanden te beschermen:

- Indien de logbestanden gevoelige gegevens (bv. persoonsgegevens) bevatten, dan moeten deze evenzeer beschermd worden. Encryptie kan dan toegepast worden om de vertrouwelijkheid van deze informatie te beveiligen;
- Het is belangrijk dat loginformatie niet gewijzigd wordt. Om de integriteit van logbestanden te garanderen, kan deze beveiligd worden door middel van *hashing* of het plaatsen van een digitale handtekening; en
- Door het toevoegen van een tijdszegel (*time stamp*) is het mogelijk om een correcte tijdssynchronisatie te realiseren, een belangrijk gegeven in de loganalyse.

3.4 Link met netwerken als maatregel

Cryptografie wordt vaak gebruikt om netwerken te beveiligen, hierbij wordt data in transit of *data in motion* (DIM) versleuteld in het kader van:

- Integriteit: door middel van *hashing* of digitale handtekening voorkomen dat 'data in motion' ongecontroleerd gewijzigd wordt; en

- Confidentialiteit: door middel van versleuteling, bijvoorbeeld van niet-publieke data over een publiek netwerk door middel van *tunneling.*

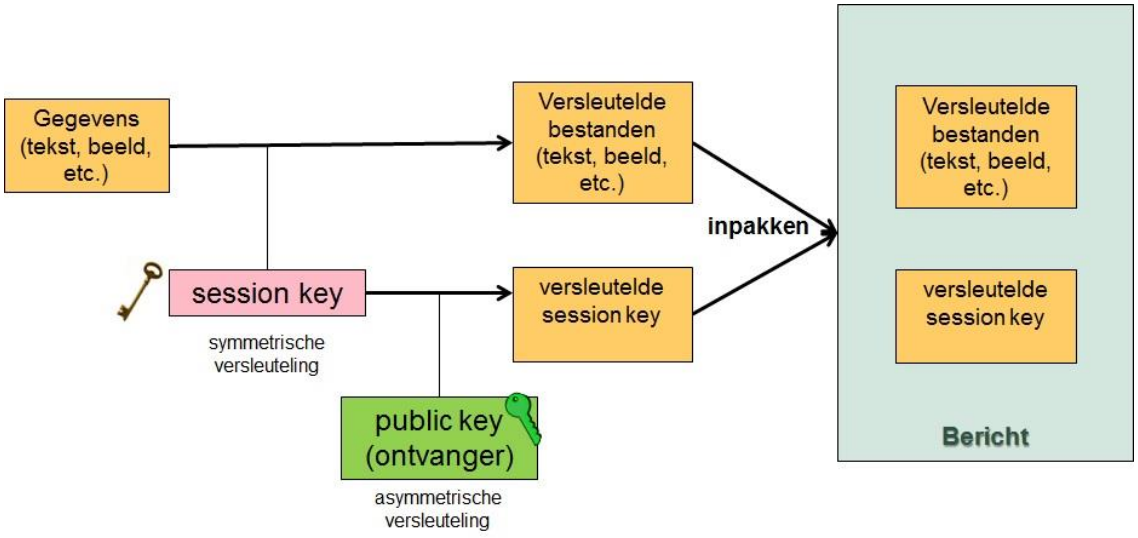
Voor meer informatie zie document*: 'Vo Informatieclassificatie - Minimale maatregelen – netwerken'*).

BIJLAGEN

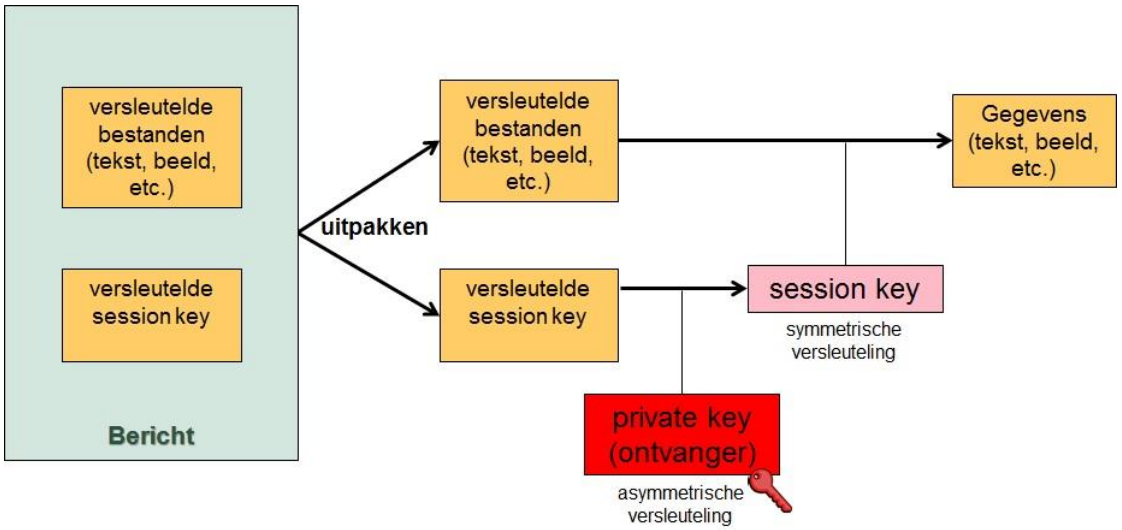
Bijlage 1: digitale enveloppe

In de praktijk wordt meestal de combinatie van symmetrische en asymmetrische encryptie gebruikt om informatie te beveiligen. Zo'n veel voorkomende toepassing waarbij de voordelen van beide systemen worden gecombineerd om berichten te vercijferen met behulp van een symmetrisch systeem en een eenmalige sleutel, de sessiesleutel, is de digitale enveloppe. Met deze techniek kan een bericht vercijferd verstuurd worden zonder dat hierbij de verzender en ontvanger van het bericht vooraf over eenzelfde geheime sleutel moeten beschikken. Asymmetrische encryptie wordt toegepast om de sessiesleutel te vercijferen. De techniek wordt geïllustreerd in onderstaande figuur. De verzender van een bericht doet het volgende:

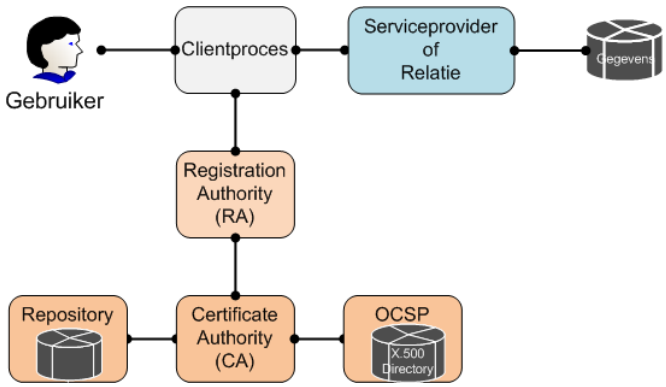
1. Opstellen van een bericht;
2. Genereren van een symmetrische sessiesleutel. Deze wordt éénmalig gebruikt;
3. Encrypteren van het bericht d.m.v. de sessiesleutel;
4. Encrypteren van de sessiesleutel (*one-time key*) met de publieke sleutel van de bestemmeling; en
5. Toevoegen van de geëncrypteerde sleutel aan het geëncrypteerd bericht, en verzenden van het geheel naar de bestemmeling.



De bestemmeling (ontvanger) van het bericht kan de sessiesleutel decrypteren d.m.v. zijn/haar private sleutel om vervolgens het vercijferde bericht opnieuw leesbaar te maken door dit te decrypteren met deze sessiesleutel.



Bijlage 2: de componenten van een PKI



Een PKI (*Public Key Infrastructure*) bestaat uit een aantal basiscomponenten:

- **Entiteiten** (dit kunnen personen of organisaties zijn) die onderling veilige transacties willen uitvoeren. De relatie tussen entiteiten kan van allerlei typen zijn (leverancier – afnemer, werkgever – werknemer, burger – overheid, enz.). Om dit te kunnen doen moeten de partijen vertrouwen op de registratieautoriteit (*Registration Authority* of RA) en de certificaatautoriteit (*Certificate Authority* of CA). Deze twee organisaties horen bij een PKI. De garantie op goed sleutelbeheer, betrouwbare authenticatie en vertrouwelijkheid hangt rechtstreeks af van de kwaliteit van deze componenten;
- **Registration Authority** (RA) is de component waar entiteiten aanvragen kunnen indienen voor het verkrijgen van een certificaat (*certificate request*). Voor een hoog kwaliteitsniveaucertificaat is de kwaliteit van de verificatie van de identiteit van de entiteit belangrijk;
- **Certificate Authority** (CA) geeft certificaten uit op basis van de door de CA zelf goedgekeurde certificaataanvragen (*certificate requests*). Deze aanvragen worden ontvangen door één of meerdere *RA-servers*. De CA beheert de uitgegeven certificaten, publiceert ze en bewaart de certificaten in de *Certificate repository*, een databank van certificaten. Daarnaast publiceert de CA een lijst van ingetrokken certificaten: de *Certificate Revocation List* (CRL). Via het OCSP-protocol (*Online Certificate Status Protocol*) kan online worden bevraagd of een certificaat geldig is;
- **Sleutelpaar**: een paar asymmetrische sleutels, waartussen een bepaalde relatie bestaat. Een sleutel is de private sleutel van de entiteit. Deze moet geheimgehouden worden. De andere sleutel is de publieke sleutel van de entiteit, deze moet juist openbaar beschikbaar zijn;
- **Certificaat**: een digitaal object, het bevat de identiteit van entiteit, ook de certificaathouder genoemd, de publieke sleutel en een verwijzing naar de verantwoordelijke CA;
- **Certificaat status server**: een *server* van de CA, die via het internet bereikbaar is. Deze stelt informatie beschikbaar over de inhoud van de CRL en over de verstrekte certificaten. Een eindgebruiker kan daarmee bijvoorbeeld via zijn browser op internet controleren of zijn relaties beschikken over geldige certificaten en wat de publieke sleutels daarvan zijn. Een voorbeeld is de *OCSP-server*.

Bijlage 3: vier stappen voor het verkrijgen van PKI-certificaten

Per stap worden steeds twee mogelijkheden uitgelegd: bij variant (A) genereert de aanvrager van het certificaat het sleutelpaar zelf, bij variant (B) doet de certificaatautoriteit (*Certificate Authority* of CA) dat.

1. Stap 1: het creëren van het sleutelpaar:

〉 (A) de aanvrager genereert zelf het sleutelpaar. Daarna vraagt hij een certificaat aan bij de registratieautoriteit (*Registration Authority* of RA). De aanvrager geeft daarbij de publieke sleutel van zijn sleutelpaar mee; of

〉 (B) de aanvrager vraagt een certificaat aan bij de RA zonder zelf de sleutel gegenereerd te hebben.

2. Stap 2: de RA doet de verificatie van de aanvrager van het certificaat. De nauwkeurigheid van de verificatie is een belangrijk kwaliteitsaspect van het certificaat. Dit kan variëren van geen verificatie tot een grondige controle van de gegevens. Daarna bestelt de RA het certificaat bij de CA. Als de RA-functie geautomatiseerd verloopt, dan worden correct ingevulde aanvragen automatisch doorgestuurd naar de CA.
3. Stap 3: de CA beoordeelt de authenticiteit van aanvragen en maakt na goedkeuring nieuwe certificaten aan en zet deze in een (lokale) bewaarplaats (*repository*):

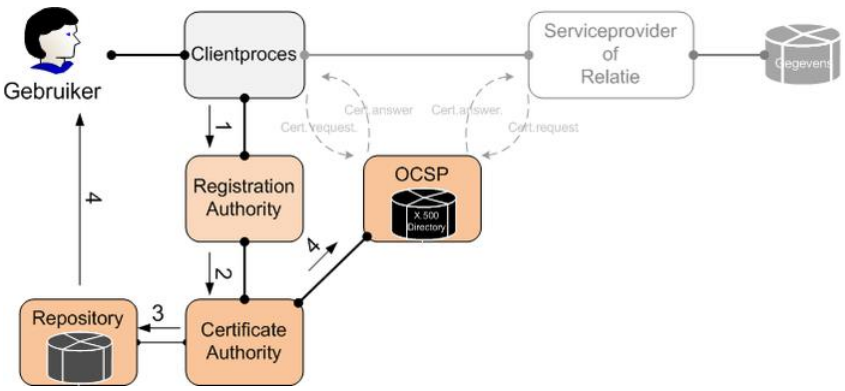
〉 (A) indien de aanvrager zelf een sleutelpaar heeft gegenereerd en de publieke sleutel heeft meegeleverd met de aanvraag, maakt de CA na goedkeuring het certificaat aan en zet deze in een lokale *repository*;

〉 (B) zoniet, genereert de CA een sleutelpaar en gebruikt de nieuwe publieke sleutel om een certificaat aan te maken. Deze wordt in een lokale *repository* gezet. De private sleutel wordt op een zodanige wijze bewaard, dat ook niemand bij de CA kennis kan nemen van de waarde van de sleutel.

4. Stap 4: De CA zorgt voor de publicatie van een up-to-date *Certificate Revocation List* (CRL), waarop alle ingetrokken certificaten staan vermeld. De CRL wordt gepubliceerd op de 'OCSP-server' (*Online Certificate Status Protocol server*), die computers antwoord geeft over de geldigheidsstatus van een certificaat:

〉 (A) de aanvrager heeft zelf het sleutelpaar gegeneerd. De CA stuurt een bevestiging van certificaat naar de aanvrager. De aanvrager is verantwoordelijk voor de beveiligingsmaatregelen rond het gebruik van de private sleutel; of

〉 (B) de aanvrager ontvangt van de CA de private sleutel op een beveiligde wijze. De aanvrager is verantwoordelijk voor een beveiligde opslag van deze sleutel en voor de beveiligingsmaatregelen rond het gebruik van de private sleutel.

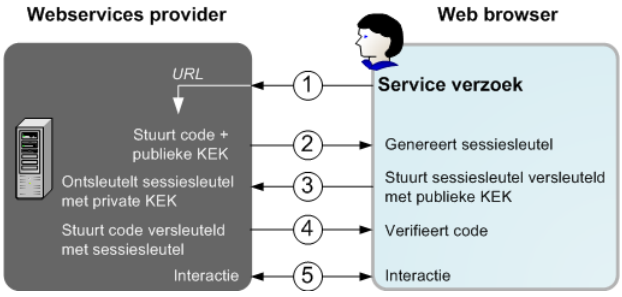


Bijlage 4: distributie mechanismen voor crypto-sleutels

Afhankelijk van de toepassing zijn er verschillende methoden om sleutels veilig te distribueren:

- Distributie van de *Key Encryption Keys* (KEK): de KEK moet voorafgaand aan de ingebruikname van de encryptie over een vertrouwd pad gedistribueerd worden. De KEK wordt meestal op een fysiek medium opgeslagen, zoals een smartcard, dat bijvoorbeeld met een speciale koerier verstuurd kan worden of bij een loket afgehaald moet worden op basis van een bewijsstuk en legitimatie. Bij specifieke apparatuur wordt de asymmetrische sleutel al bij de fabriek geïnstalleerd;
- Distributie van periodieke sleutels: voor beveiliging van de distributie van periodieke sleutels kan men zowel kiezen voor asymmetrische encryptie (de KEK is een publieke sleutel afkomstig van een PKI) als voor symmetrische encryptie (de KEK is geheim). Een voorbeeld van een systeem met een geheime KEK is de (*Advanced Encryption Standard*) AES Key Wrap-methode, zie onderstaande figuur.

  Symmetrische encryptie met de KEK als sleutel zorgt hier voor de beveiliging van de periodieke sleutel, zodat deze via een niet vertrouwd pad verzonden kan worden of veilig op een systeem kan worden opgeslagen. De periodieke sleutel kan weer verkregen worden met de omgekeerde bewerking (*Key Unwrap*) en dezelfde KEK. Een bekende toepassing is *Over-The-Air-Keying* (OTAK) voor encryptie van draadloze systemen voor spraak;
- Distributie van periodieke sleutels met een publieke KEK verloopt bijvoorbeeld op een vergelijkbare manier als hierna beschreven wordt voor distributie van een sessiesleutel; of
- Distributie van een sessiesleutel: onderstaande figuur laat zien hoe een publieke KEK (afkomstig van een PKI) wordt gebruikt bij het uitwisselen van een sessiesleutel voor het beveiligen van webverkeer (https) tussen client en *server* op het internet.



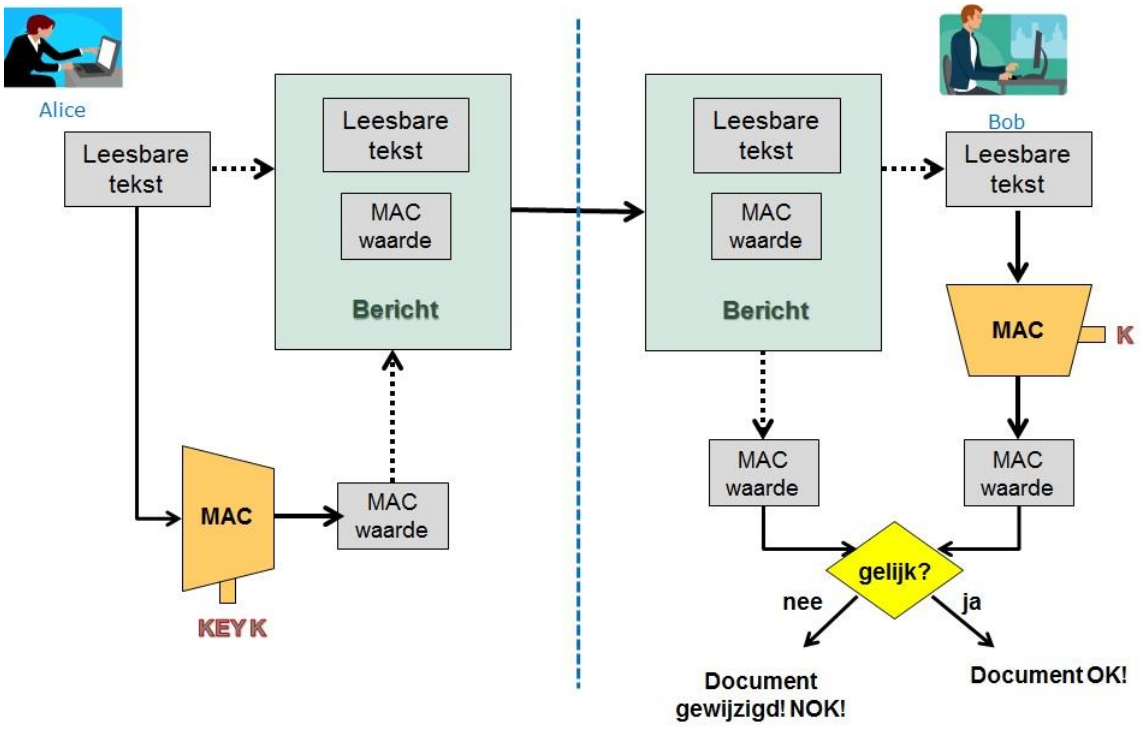
De berichtuitwisseling verloopt als volgt:

1. Vanuit de client (*webbrowser*) wordt een serviceverzoek gedaan naar een *webserver*;
2. De *server* stuurt een willekeurige code samen met de publieke KEK van de *server*. De webbrowser checkt de geldigheid van het certificaat van de *server* aan de hand van een tabel in de browser;
3. De client genereert een willekeurige (*random*) sessiesleutel, versleutelt deze met de publieke KEK van de *server* en stuurt die terug naar de *server*;
4. De *server* decrypteert de sessiesleutel met behulp van de geheime private KEK van de *server* en stuurt de met de nieuwe sessiesleutel versleutelde code opnieuw naar de client, zodat die kan verifiëren of er nog steeds met dezelfde server gecommuniceerd wordt;
5. De sessiesleutel is nu bij beide partijen bekend en wordt vervolgens voor de symmetrische encryptie van het webverkeer gebruikt. Veilige interactie is nu gegarandeerd voor de duur van de sessie.

Omdat de KEK van alle sleutels de langste levensduur heeft, is dat ook de meest gevoelige sleutel. De KEK zelf wordt niet versleuteld en dient dus op een andere wijze fysiek en/of logisch te worden afgeschermd. Toegang tot de KEK is alleen toe te staan na identificatie en authenticatie van de beheerder dan wel de gebruiker van de KEK. Er bestaan ook oplossingen waarbij een KEK ontbreekt. Een bekend voorbeeld is wifi-encryptie in de *pre-shared key* (PSK) modus. Daarbij worden de (periodieke) wifisleutels ter plaatse door een beheerder ingevoerd. De beheerder moet ervoor zorgen dat de sleutels niet bekend worden.

Bijlage 5: MAC

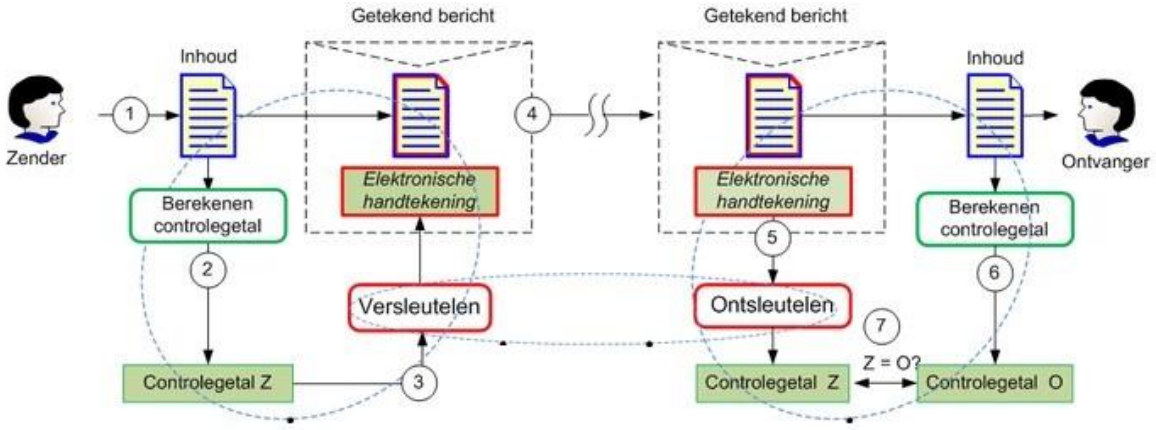
De volgende figuur illustreert het gebruik van een (*Message Authentication Code* of) MAC om bericht-authenticatie te realiseren bij het verzenden van berichten:



Message Authentication Codes worden in diverse protocollen toegepast. Bijvoorbeeld in (*Secure Sockets Layer/ Transport Layer Security*) SSL/TLS om het kanaal tussen twee communicerende partijen (*server*/*server* of *client*/*server*) te beveiligen: bericht-integriteit (Werd de informatie niet gewijzigd in transit?) en authenticatie (Is de informatie wel afkomstig van de juiste *server*?). Dit toon het belang aan om websites niet louter omwille van vertrouwelijkheidsdoeleinden via (*HyperText Transfer Protocol Secure*) https (dus SSL/TLS over http) te ontsluiten.

Bijlage 6: integriteit, authenticatie en onweerlegbaarheid

Integriteit, authenticatie en onweerlegbaarheid worden aan een bericht toegevoegd in volgende stappen:



1. De zender stelt een bericht op;
2. Over de inhoud van het bericht wordt een controlegetal berekend, de *hash*. Voor de berekeningsmethode van de *hash* wordt een standaard algoritme gebruikt, dat door elke infrastructuur die deze standaard ondersteunt, is toe te passen;
3. Het controlegetal wordt versleuteld met de private sleutel van de zender en bij de al dan niet versleutelde berichtinhoud gevoegd als een digitale handtekening. Versleutelen van de inhoud van het bericht voor vertrouwelijkheid is mogelijk;
4. Het bericht wordt samen met de digitale handtekening verstuurd;
5. Van het bericht wordt de handtekening ontcijferd met de publieke sleutel van de zender, waarna het controlegetal (Z) voor de zender leesbaar wordt;
6. De publieke sleutel van de zender is gekoppeld aan een geldig certificaat, uitgegeven door een betrouwbare certificaatautoriteit (CA). Dit garandeert de identiteit van de zender;
7. Van de inhoud van het bericht wordt aan de ontvangkant opnieuw een controlegetal (O) berekend; en
8. Tenslotte wordt het meegestuurde controlegetal (Z) vergeleken met controlegetal (O). Wanneer deze getallen precies gelijk zijn, dan is daarmee bewezen dat:

- De inhoud van het bericht niet is gewijzigd
- Het bericht afkomstig is van de zender van de overeenkomende publieke sleutel, gekoppeld aan het certificaat van de zender en daarmee heeft de zender zich bij de ontvanger geauthentiseerd.