

UiO : Institutt for informatikk
Det matematisk-naturvitenskapelige fakultet

IN1020 - Introduksjon til datateknologi

Kryptografi – Nøkkelen til digital samhandling

09.09.2026

Kristin Skar & Håkon Kvale Stensland

simula

Plan for ”nettverksdelen” av IN1020

- 28. august – Introduksjon til datakommunikasjon

- 2. september – Lagdeling i datakommunikasjon
- 3. september – Hvordan fungerer din hjemmeruter?

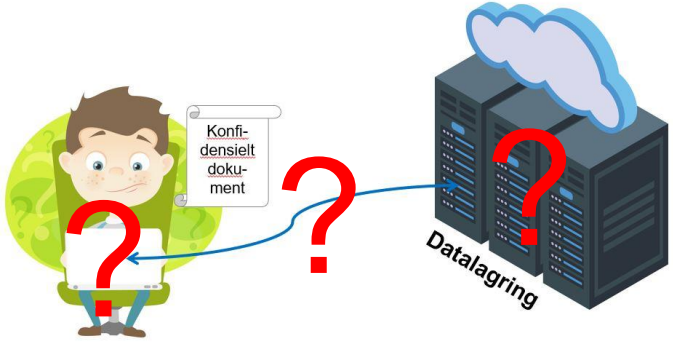
- **9. september – Kryptografi: Nøkkelen til digital samhandling**
- 11. september – Introduksjon til operativsystemer

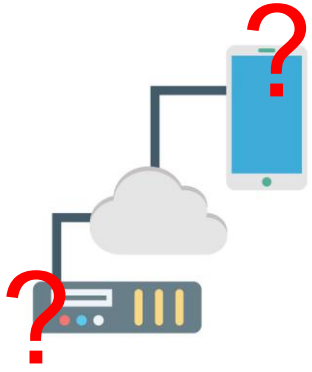
- 30. september – Tjenester i internett

Introduksjon til kryptografi

- Hvorfor trenger vi kryptografi?
- Hva er kryptografi?
- Hvordan ”knekke” en kryptering.
- Hemmelig nøkkelkryptering.
- Hash-algoritmer.
- Offentlig nøkkelkryptering.

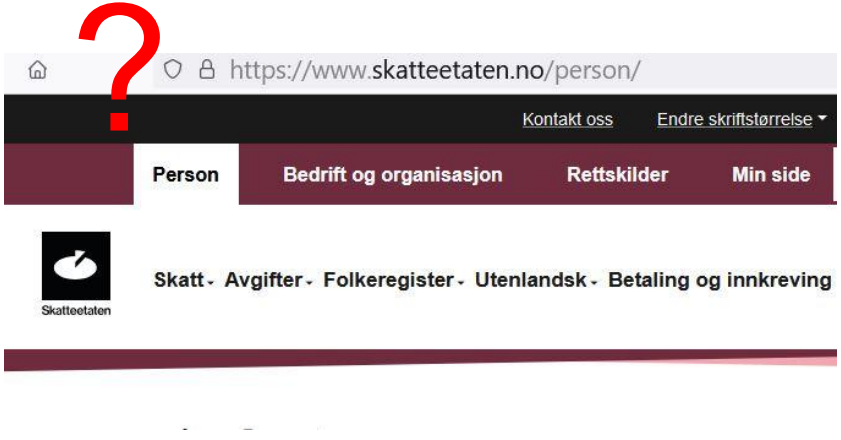
Trenger vi kryptografi?

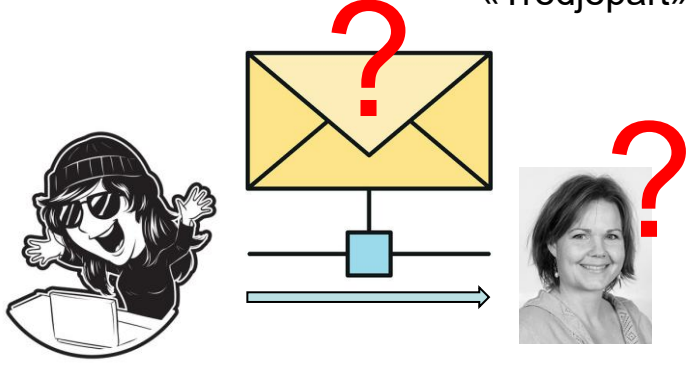






«Tredjepart»





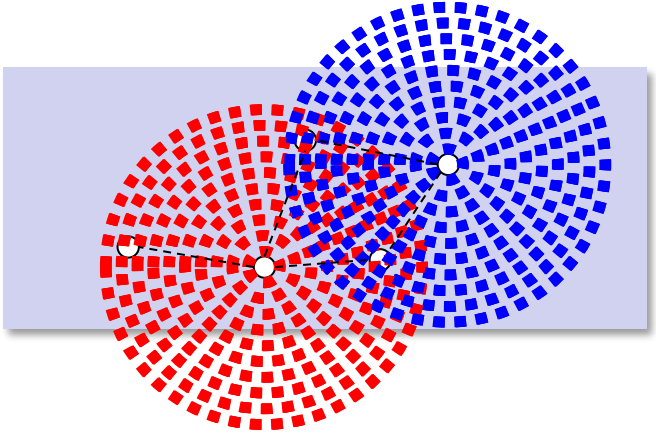
Repetisjon: Nettverksstrukturer

Broadcast-systemer:

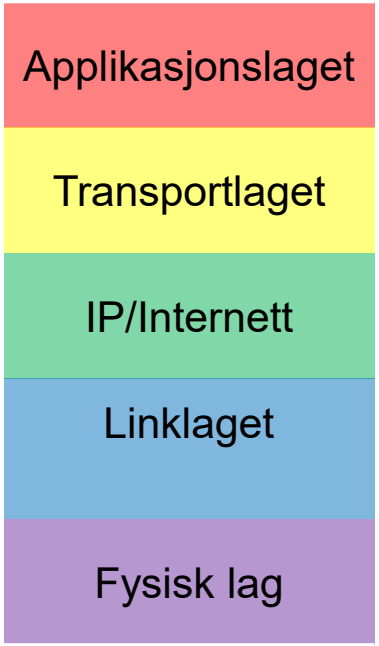


- *Radio:*
  - WiFi (IEEE 802.11)
  - Mobiltelefoni: 2G, 4G, 5G
  - Satellitt: Starlink, Iridium, etc.

- ***Egenskaper:***
  - Kan bli problemer hvis to noder sender samtidig…
  - Feildeteksjon er viktig.
  - **Når en node sender kan alle noder som er i rekkevidde lytte på kommunikasjonen!**



Kryptering og sikkerhet i nettverket



Secure Sockets Layer – (**https**) Kryptering for ende-til-ende Applikasjoner – f.eks. nettbank eller butikker.

VPN (IPSEC etc.) – Kobler to nettverk sammen så det fungerer som ett LAN selv om de er fysisk adskilt

WPA – WiFi Protected Access - Kryptering på trådløse nettverk, krypterer forbindelsen mellom deg og aksesspunkt/ruter.

Kryptering på flere lag gjør det vanskeligere for uvedkomne å lytte til kommunikasjonen. Adressen (avsender / mottakeren) er vanskelig å kryptere, da router må vite hvor pakken skal leveres.

Kryptografi: Viktige definisjoner



- Beskjeder:
  - Plaintext / Klar tekst
  - Ciphertext / Kryptert tekst
- Ingredienser:
  - Algoritmer
  - Nøkler



Hemmelige koder

- Cæsarchiffer
  - Skifte bokstaver 3 ganger fremover
    - DOZEN → GRCHQ
- Dekoderringer
  - Substituer bokstaver *n* bokstaver videre (*n* = 1..25)
    - HAL → IBM (*n* = 1)
- Monoalfabetisk chiffer
  - Tilfeldig mapping (26! = 4.03291461 × 10^{26})
    - 1 ms / forsøk → 10M år… ***MEN***, bokstavfrekvens er en svakhet...



Hvordan ”knekke” en kryptering

- *Kun Ciphertext / kryptert tekst*
  - *Prøve alle forskjellige nøkler (brute-force)*
  - *Trenger en lang nok chipertext*
- *Deler av plaintext / klar tekst*
  - *Har chipertext, sammenligne med forventede verdier*
  - *Brukt for å knekke Enigma*
- *Finne svakheter i krypteringsalgoritme*
- Finne svakheter i implementasjonen av algoritmen

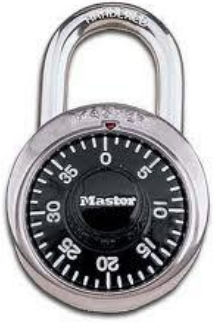
Kompleksitet vs. ”Brute-force”

- Algoritmen bør være effektiv å bruke
- Sikkerheten er avhengig av hvor komplekst koder er å knekke
- Eksempel: Kombinasjonslås
  - 3 tall, mellom 1 og 40 – 10 sekunder per forsøk
  - 4 tall mellom 1 og 40 – 13 sekunder per forsøk



Kompleksitet

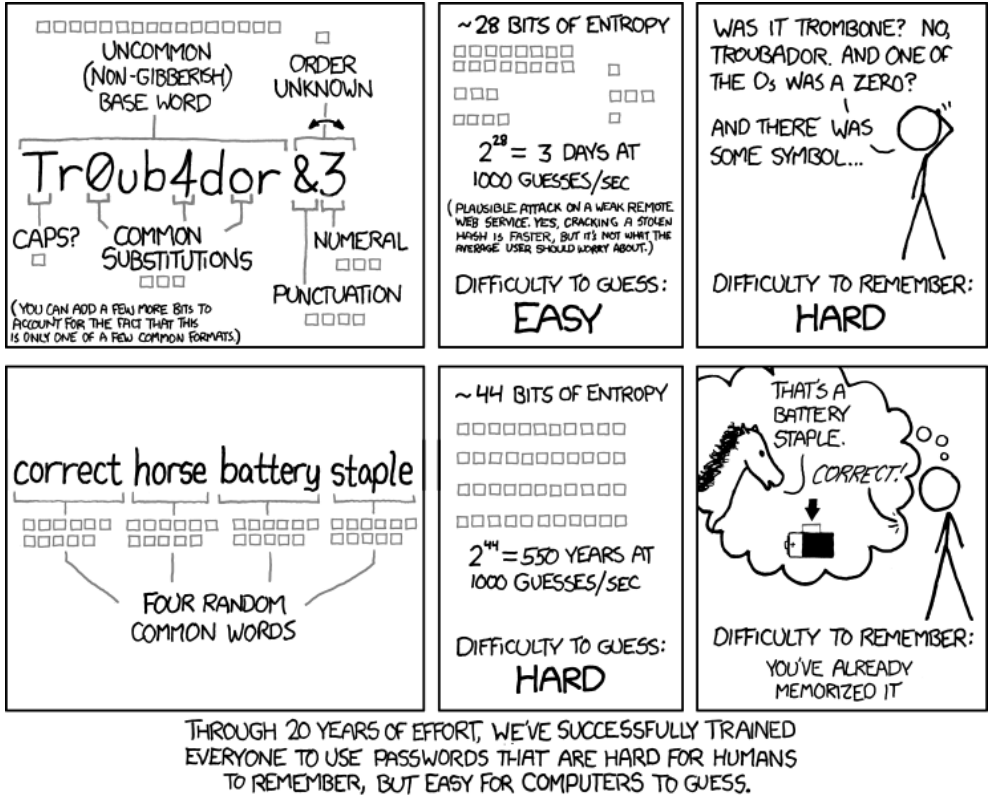
- Kombinasjonslås
  - 3 tall, mellom 1 og 40 – 10 sekunder per forsøk
    - Antall kombinasjoner: $40^3 = 64.000$
    - 178 timer
  - 4 tall, 13 sekunder per forsøk
    - $40^4 = 2,560.000$
    - 9244 timer



Brute-force av kombinasjonslås med utnyttelse av svakheter:
https://youtu.be/09UgmwtL12c

Hva er egentlig et god passord?

- De mest vanlige passordene i 2024:
  - 123456, password, 123456789, 12345678, 12345
- Passord generert med verktøy som 1Password?
  - cdHAY2DIVrX'+3Pt
- «Passphrase»:
  - laptop_Wrapped-bow_Keyboard



Frekvensanalyse

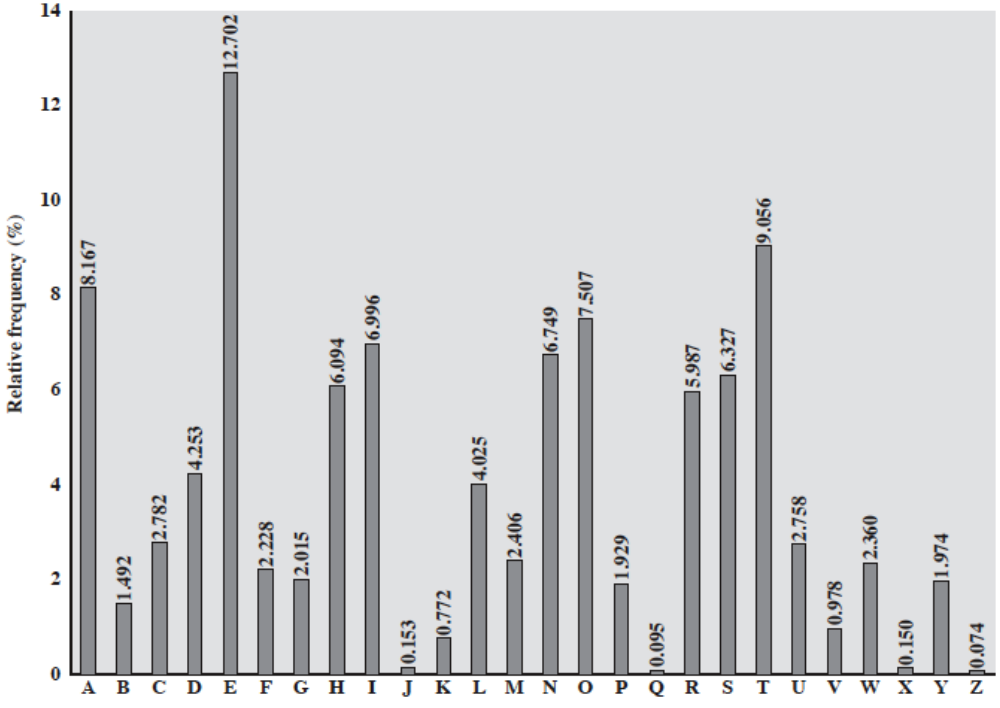


Figure 2.5 Relative Frequency of Letters in English Text

- Følgende passord:
  - Interdisciplinary
    Knekkes på få sekunder
  - Hvorfor? Er jo 18 tegn...
- Bruke av avskjæringer:
  - Frekvensanalyse
  - Bruk av ordlister

[Fra *Stallings, Cryptography & Network Security*]

16. Oktober 2017

Sikkerhet i trådløse nettverk?

- Svakhet i protokoller og algoritmer
  - Svakheter i implementasjoner
    - Key Reinstallation Attacks (KRACK)



KRACK: Svakheter i WPA:
https://www.krackattacks.com/

- ”Brute-force”
  - Teste alle mulige kombinasjoner
  - Ordbokangrep
  - Moderne GPUer kan teste flere milliarder passord i sekundet

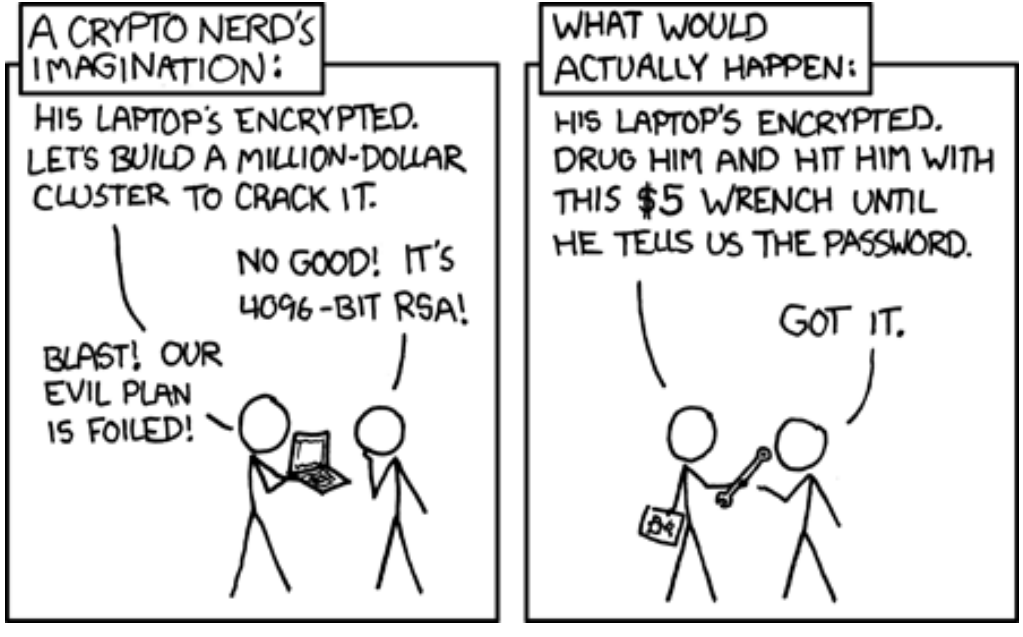
Number of Characters	Numbers Only	Lowercase Letters	Upper and Lowercase Letters	Numbers, Upper and Lowercase Letters	Numbers, Upper and Lowercase Letters, Symbols
4	Instantly	Instantly	Instantly	Instantly	Instantly
5	Instantly	Instantly	57 minutes	2 hours	4 hours
6	Instantly	46 minutes	2 days	6 days	2 weeks
7	Instantly	20 hours	4 months	1 year	2 years
8	Instantly	3 weeks	15 years	62 years	164 years
9	2 hours	2 years	791 years	3k years	11k years
10	1 day	40 years	41k years	238k years	803k years
11	1 weeks	1k years	2m years	14m years	56m years
12	3 months	27k years	111m years	917m years	3bn years
13	3 years	705k years	5bn years	56bn years	275bn years
14	28 years	18m years	300bn years	3tn years	19tn years
15	284 years	477m years	15tn years	218tn years	1qd years
16	2k years	12bn years	812tn years	13qd years	94qd years
17	28k years	322bn years	42qd years	840qd years	6qn years
18	284k years	8tn years	2qn years	52qn years	463qn years

Time it takes a hacker to brute force your password in 2025

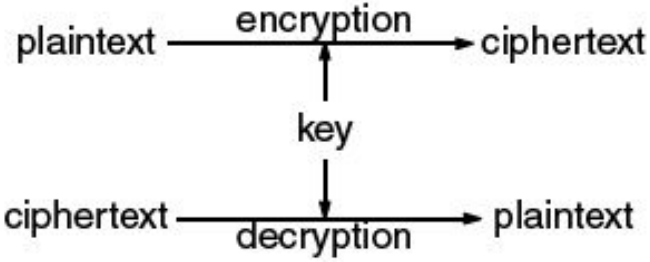
Hardware: 12 x RTX 5090
Password hash: bcrypt (10)

Forskjellige typer kryptering

- *Symmetrisk kryptering*
  - *Hemmelig nøkkelkryptering*
  - En nøkkel: Men må deles mellom sender og mottaker)
- *Hash-algoritmer*
  - Enveis identifikasjon
  - Ingen nøkkel
- *Asymmetrisk kryptering*
  - Offentlig nøkkelkryptering
  - To nøkler:
    - *Privat*
    - *Offentlig*



Symmetrisk kryptering

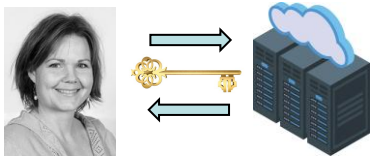


Også kjent som “vanlig kryptering”

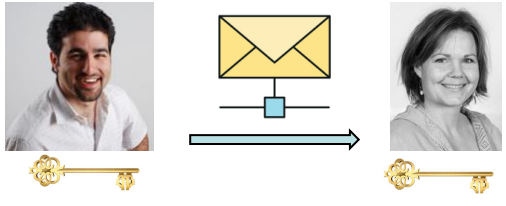
- To typer:
  - Block cipher (DES, 3DES, AES)
  - Stream cipher (RC4)
- Per dags dato er «block cipher» mest brukt, og gjerne i blokker på 128-bits (AES)

Symmetrisk kryptering for (økt) sikkerhet

- **Fremgangsmåte**: Benytter én felles hemmelig nøkkel, avsender bruker nøkkel for å kryptere en melding, mottager bruker samme nøkkel til å dekryptere meldingen.
- **Anvendelse**:
  - Kryptering av data
  - Kryptering av meldinger



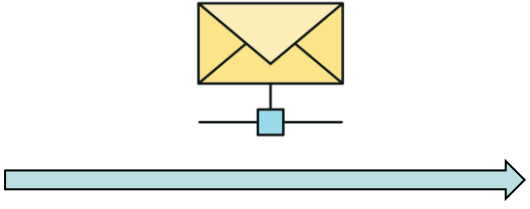
- **Hvilke sikkerhetsmål kan ivaretas?**
  - Konfidensialitet? **JA**
  - Integritet? **NEI**
  - Autentisk avsender (autentisk dataopprinnelse)? **JA**
  - Autentisk mottaker? **JA**
  - Ubenektelighet? **NEI**



- **Farer**: Nøkkel på avveie? Nøkkel tapt? Hvordan dele hemmelig nøkkel trygt?

Symmetrisk kryptering: Ubenektelig?







«Melding er kryptert med vår hemmelige nøkkel, så jeg vet at Omid sendte meldingen.»

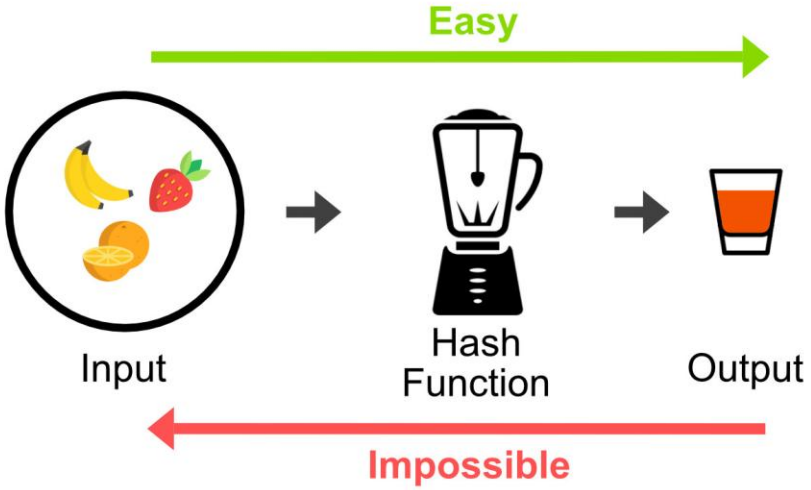
«Men du har jo samme nøkkel, så du kunne sendt meldingen til deg selv.»



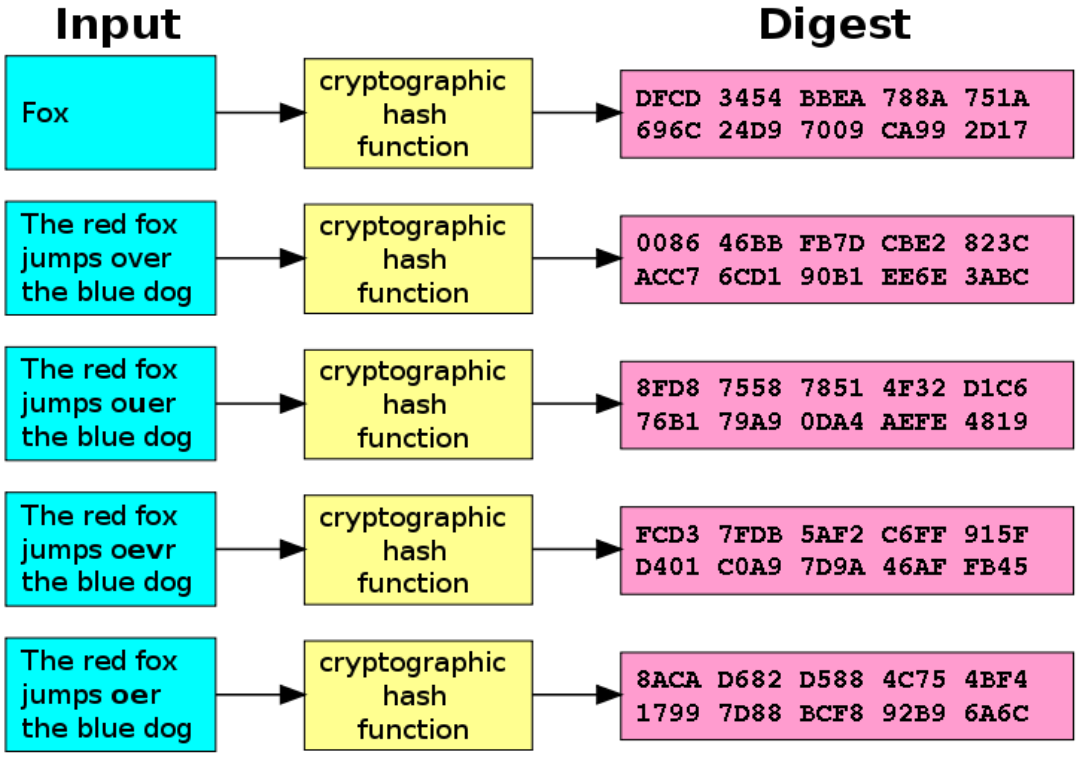
?

Kryptografiske enveisfunksjoner (hash-funksjoner)

- Sjekke dataintegritet / ikke-reverserbar
- Kriterier for en god sjekksumalgoritme:
  - Enkelt å regne ut sjekksum for en gitt beskjed.
  - Ikke mulig (vanskelig) å finne en beskjed for en gitt sjekksum.
  - Ikke mulig (vanskelig) å endre en beskjed uten at sjekksummen blir endret.
  - Ikke mulig (vanskelig) å finne to forskjellige beskjeder med same sjekksum

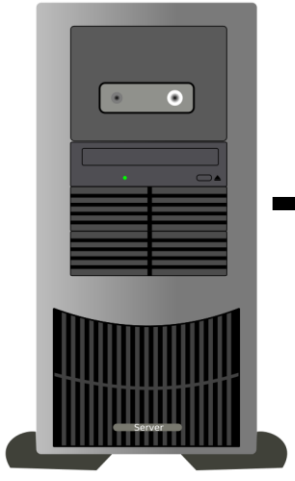


Hash-algoritmer



http://en.wikipedia.org/wiki/Cryptographic_hash_function

Viktigheten av gode tilfeldige tall

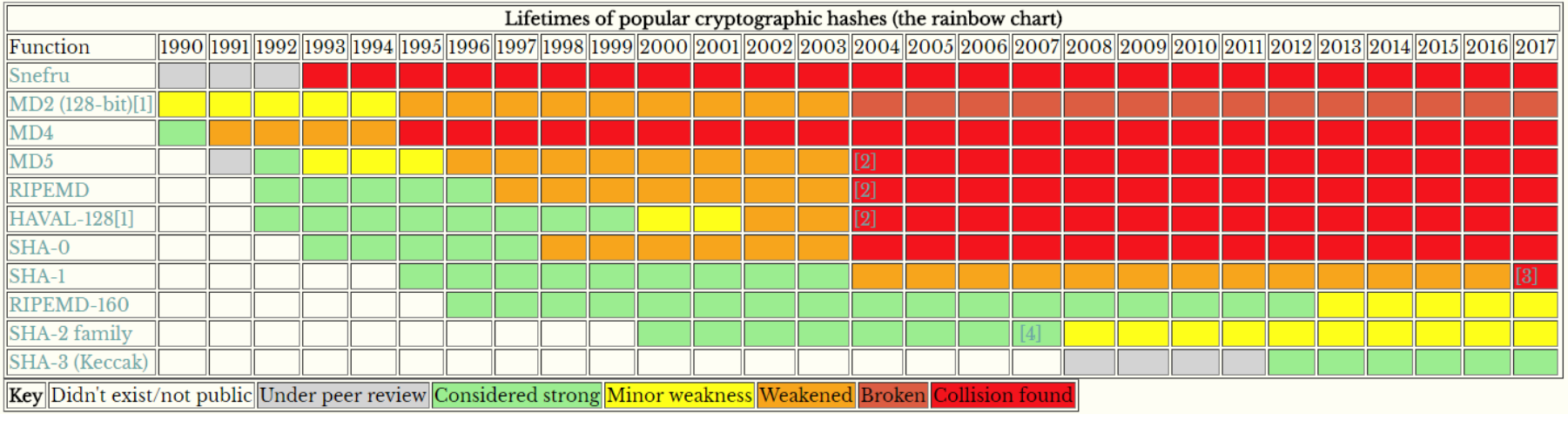




- Brukes til å generere kryptografiske nøkler
- Brukes for å gjøre det vanskeligere å gjette seg frem til kryptografiske nøkler
- Kan f.eks. brukes som «salt» når man genererer nøkler

Tilfeldig tallgenerator: NSA bakdør:
https://en.wikipedia.org/wiki/Dual_EC_DRBG

Kryptografisk Hashlivssykel



Lifetimes of popular cryptographic hashes (the rainbow chart)

Function	1990	1991	1992	1993	1994	1995	1996	1997	1998	1999	2000	2001	2002	2003	2004	2005	2006	2007	2008	2009	2010	2011	2012	2013	2014	2015	2016	2017
Snefru																												
MD2 (128-bit)[1]																												
MD4																												
MD5															[2]													
RIPEMD															[2]													
HAVAL-128[1]															[2]													
SHA-0																												
SHA-1																												[3]
RIPEMD-160																												
SHA-2 family																		[4]										
SHA-3 (Keccak)																												

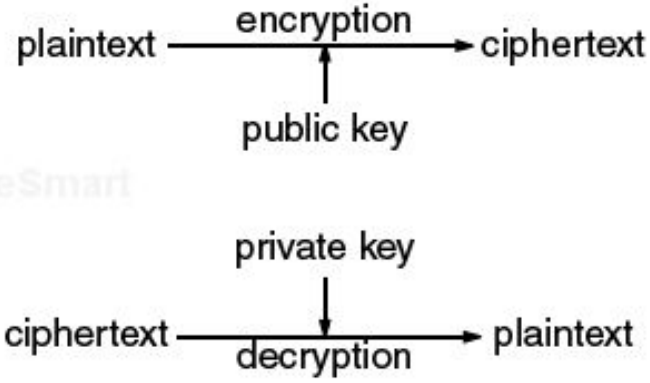
Key: Didn't exist/not public | Under peer review | Considered strong | Minor weakness | Weakened | Broken | Collision found

Status på Hash-algoritmer:
http://valerieaurora.org/hash.html

Hash-algoritmer for (økt) sikkerhet

- **Fremgangsmåte**: Bruker en sjekksumalgoritme på data/melding for å generere en sjekksum (hash-verdi).
- **Anvendelse**:
  - Beskyttelse av passord
  - Sammenligning av filer (sjekk av integritet)
  - Integritetssjekk, f.eks. av programvare-distribusjoner
  - Integritetssjekk av meldinger (dataintegritet)
- **Hvilke sikkerhetsmål kan ivaretas?**
  - Integritet til data/melding? **JA**
  - Konfidensialitet? **JA**
- **Farer**? Svake algoritmer med begrenset levetid.

Asymmetrisk kryptering



Også kjent som «offentlig nøkkelkryptering»

Offentlig nøkkel: publisert, gjerne åpnet på nettet for alle
Privat nøkkel: hemmelig og ikke offentlig

Asymmetrisk kryptering





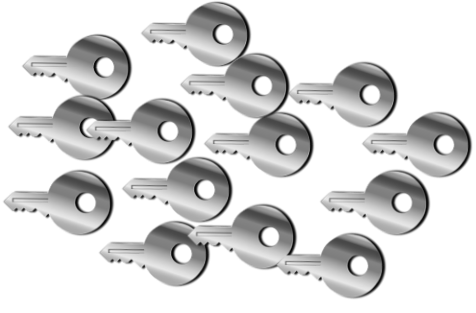
Privat nøkkel



Offentlig nøkkel



Den offentlige nøkkelen kopieres og gjøres lett tilgjengelig for dine kontakter



Asymmetrisk kryptering - kryptere



Omid ønsker å sende en melding til Kristin



Klartekst

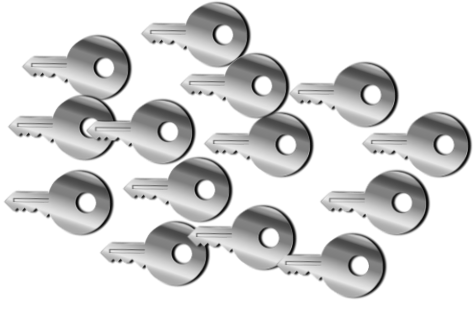
Eksamen i IN1020
Høsten 2025

Chiffertekst

27fdbe64 8588b3d1
21ed8e8e ed01c340
982c50a4 8d87c7b2

Kryptere meldingen med Kristin sin offentlige nøkkel

Kristin sin offentlige nøkkel



Asymmetrisk kryptering - dekryptere



Kristin mottar chiffertekst fra Omid



Chiffertekst

27fdbe64 8588b3d1
21ed8e8e ed01c340
982c50a4 8d87c7b2

Kristin sin private nøkkel



Klartekst

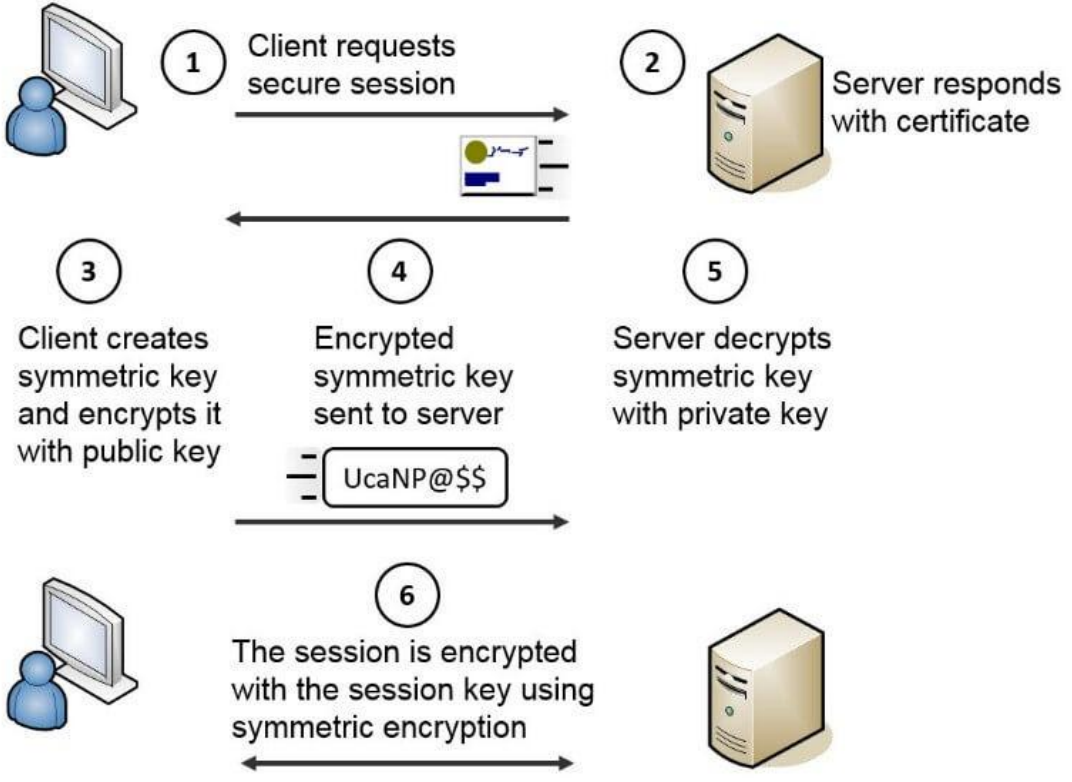
Eksamen i IN1020
Høsten 2025

Utfordringer med asymmetrisk kryptering

- Effektivitet
  - Asymmetriske krypteringsalgoritmer er mange ganger tregere enn symmetriske algoritmer.

- Løsning: hybrid modell
  - Asymmetrisk kryptering brukes for å bli enige om en midlertidig nøkkel.
    - Den vanligste teknikken er kjent som Diffie-Hellman nøkkelutveksling
  - Symmetrisk kryptering brukes under resten av kommunikasjonen.

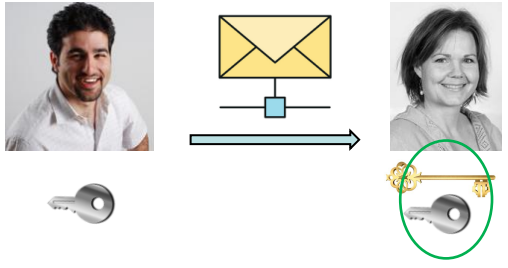
HTTP med SSL (https) – Asymmetrisk kryptering



Asymmetrisk kryptering for meldingsutveksling:

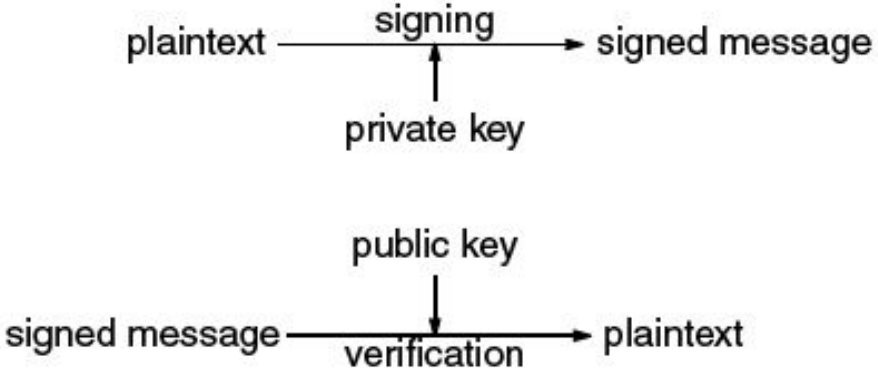
- **Fremgangsmåte**: Avsender bruker mottagers offentlige nøkkel for å kryptere en melding, mottager bruker sin private nøkkel til å dekryptere meldingen.
- **Anvendelse**: Trygg overføring av data/melding, samt autentisering

- **Hvilke sikkerhetsmål kan ivaretas?**
  - Konfidensialitet? **JA**
  - Integritet? I praksis **JA**
  - Autentisk avsender (autentisk dataopprinnelse)? **NEI**
  - Ubenektelighet? **NEI**
  - Autentisk mottager **JA**



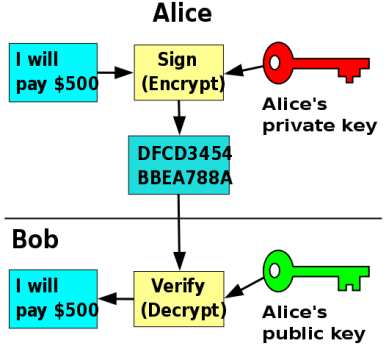
- Farer: Nøkkel på avveie, falske nøkler

Digital signatur



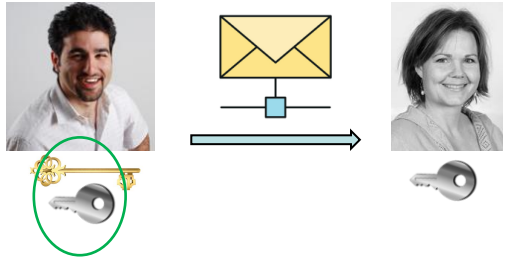
Asymmetri:
Signatur kan kun bli generert av eier eller noen som kjenner privat nøkkel.
Signatur kan bli *verifisert* av alle som har tilgang til offentlig nøkkel.

En korrekt verifisert signatur bekrefter avsenderens identitet, og at meldingen Ikke ble endret fra sending til mottak.



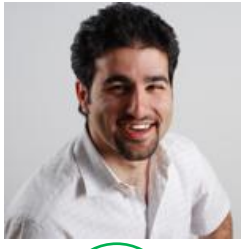
Asymmetrisk kryptering for digital signering:

- **Fremgangsmåte:** Avsender krypterer (og dermed *signerer*) en melding med sin private nøkkel. Mottager dekrypterer meldingen med senders offentlige nøkkel, og har dermed *verifisert* avsender.
- **Anvendelse**: Ivareta tillitt til at meldinger kommer fra en hevdet avsender, samt autentisering.
- **Hvilke sikkerhetsmål kan ivaretas?**
  - Konfidensialitet? **NEI**
  - Integritet? **NEI** (men i praktisk bruk **JA)**
  - Autentisk avsender (autentisk dataopprinnelse)? **JA**
  - Autentisk mottager? **NEI**
  - Ubenektelighet? **JA**

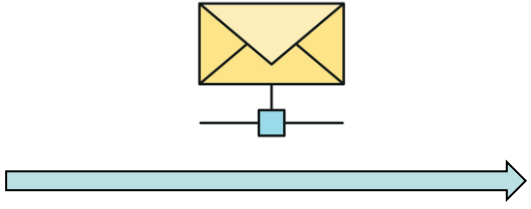


- Farer: Nøkkel på avveie, falske nøkler

Digital signatur: Ubenektelighet?









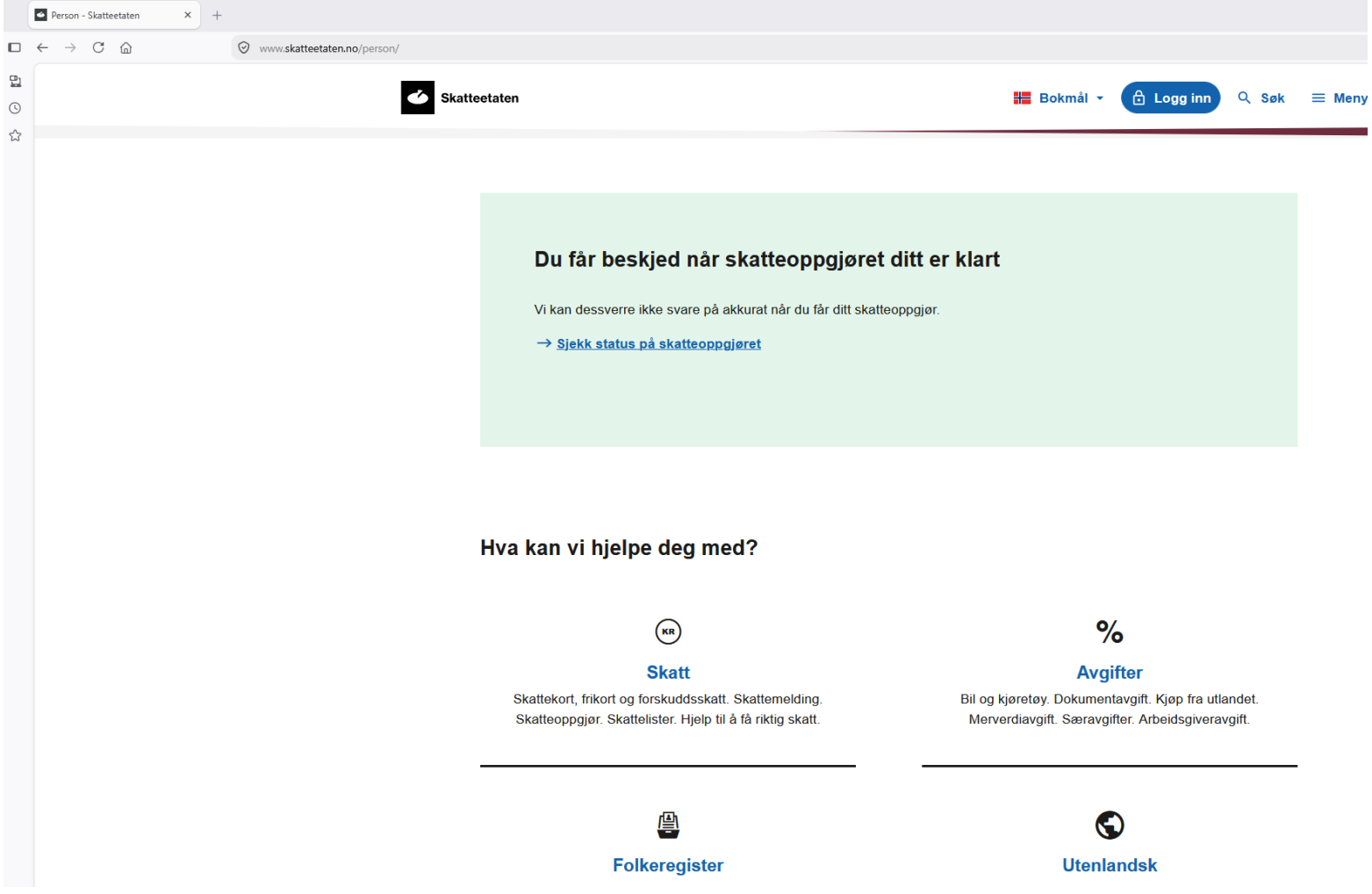


«Digitalt signert med Omids private nøkkel, så jeg vet at Omid sendte meldingen.»



?

«Du er rett, bare Omid kunne signert denne meldingen.»


Person - Skatteetaten
www.skatteetaten.no/person/
Skatteetaten
Bokmål
Logg inn
Søk
Meny
Du får beskjed når skatteoppgjøret ditt er klart
Vi kan dessverre ikke svare på akkurat når du får ditt skatteoppgjør.
→ Sjekk status på skatteoppgjøret
Hva kan vi hjelpe deg med?
KR
Skatt
Skattekort, frikort og forskuddsskatt. Skattemelding. Skatteoppgjør. Skattelister. Hjelp til å få riktig skatt.
%
Avgifter
Bil og kjøretøy. Dokumentavgift. Kjøp fra utlandet. Merverdiavgift. Særavgifter. Arbeidsgiveravgift.
Folkeregister
Utenlandsk

Certificate

www.skatteetaten.no | DigiCert EV RSA CA G2 | DigiCert Global Root G2

Subject Name

Inc. Country	NO
Business Category	Private Organization
Serial Number	974 761 076
Country	NO
State/Province	Oslo
Locality	Oslo
Organization	Skatteetaten
Common Name	www.skatteetaten.no

Issuer Name

Country	US
Organization	DigiCert Inc
Common Name	DigiCert EV RSA CA G2

Validity

Not Before	Wed, 27 May 2026 00:00:00 GMT
Not After	Fri, 11 Dec 2026 23:59:59 GMT

Subject Alt Names

DNS Name	www.skatteetaten.no
DNS Name	skatteetaten.no

Nøkkelutveksling og sertifikater

- Utfordring: Hvordan stole på at en offentlig nøkkel er ekte?

  Løsning: En identitet og tilhørende nøkkel knyttes sammen i et *offentlig-nøkkel sertifikat.*

- Neste utfordring: Hvordan stole på at sertifikatet er ekte?

  Løsning: Sertifikatet utstedes og forvaltes av en *betrodd virksomhet* (sertifikat-myndighet - CA), som går god for at nøklene er autentiske – altså *tilhører identiteten den utgir seg for å tilhøre*.

Men hva er et offentlig-nøkkel sertifikat?

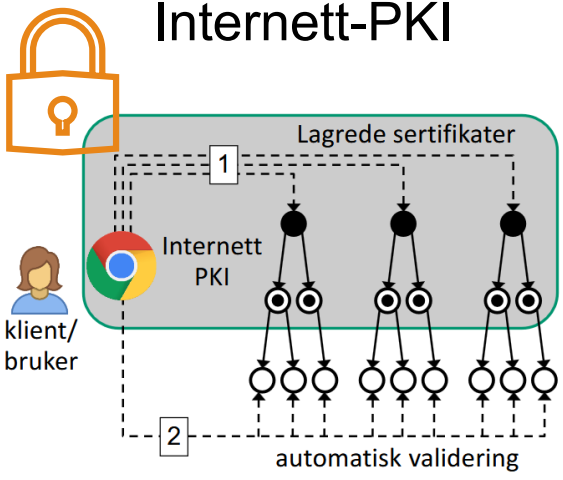


- Et sertifikat består av den offentlige nøkkelen, samt en rekke attributer knyttet til nøkkelen:
  - Eier, gyldighetsperiode, utsteder, kryptoalgoritme brukt, nøkkelstørrelse, etc.
- Attributtene pakkes sammen og signeres av sertifikatutstederen = vi har et sertifikat
- Har *standarder* for hva et sertifikat skal inneholde (f.eks. såkalt X.509)

Og hvordan distribuere offentlige nøkler?

- **Man utarbeider en «Offentlig-nøkkel infrastruktur» (PKI)**
  - PKI er et *rammeverk* for *utstedelse, administrasjon og bruk* av digitale sertifikater med offentlige nøkler
- **Hovedformål**: Sikre *ektheten* av offentlige nøkler, samt trygg og forsvarlig nøkkel-distribusjon.
- **En PKI må bestå av**:
  - En policy for sertifikat-håndtering
  - Teknologi for å implementere policyen
  - Prosedyrer for hvordan håndtere og forvalte nøklene
  - Tillittsmodell for sertifikatene med offentlige nøkler

Eksempler på PKI-er:



bankID

FORSVARET

Norsk helsenett

NB! «Tillitsforvirring»

?



Kryptering oppsummert:

- Bruksområder:
  - Sikre konfidensialitet til data/meldinger – gjøre data uleselig for de som ikke har de riktige nøklene (de som ikke er *autorisert*).
  - Sikre integritet til data – at data er korrekt og ikke endret utilsiktet
  - Autentisering av entiteter som kommuniserer – sikre ektehet/autentisitet til den andre enheten i kommunikasjon (system-/organisasjonsautentisitet, dataautentisitet)
  - For å oppnå ubenektelighet i avtaleinngåelse (digital signatur)
- Krypteringsformer:
  - Hash-algoritmer (enveis-kryptering)
  - Symmetrisk (én hemmelig delt nøkkel)
  - Asymmetrisk (nøkkelpar: offentlig + privat (hemmelig) nøkkel)

Offentlig-nøkkel infrastruktur oppsummert:

- ✓ Hver eneste offentlige nøkkel bakes inn i hvert sitt *elektroniske sertifikat* som knytter nøkkel og identitet sammen.
- ✓ Et *sertifikat* utstedes av en *tiltrodd sertifikatutsteder* (Certificate Authority), som går god for den offentlige nøkkelens ekthet.
- ✓ Den tilhørende private nøkkelen kan oppbevares (trygt!) på en datamaskin/mobil, på et smartkort, i banken (BankID), ...
- ✓ Sertifikatene med de offentlige nøklene gjøres tilgjengelig for alle mottagere som skal kunne kryptere og dekryptere meldinger.

(I praksis mer komplekst enn dette, da man opererer med *hierarki av sertifikater*, *tillitskjeder*, mm. Utenfor IN1020-pensum ☺)

Krypteringsformer, læringsmål:

- Forstå nytteverdien av kryptering i digital samhandling.
- Kjennetegn og vanlige bruksområder for:
  - Hash-algoritmer
  - Symmetrisk kryptering
  - Asymmetrisk kryptering
- Hva er og hvorfor vi trenger offentlig-nøkkel sertifikater og offentlig nøkkel-infrastruktur (PKI)

Ekstramateriale:

- *Bøker og artikler:*
  - W. Stallings. Cryptography and Network Security: Principles and Practice (7th Edition), 2016, Pearson
- *Sikkerhetskurs på IFI og MatNat:*
  - IN2120 – Informasjonssikkerhet
  - IN3210 – Network and Communications Security
  - TEK5510 – Sikkerhet i operativsystemer og programvare