

UiO : Institutt for informatikk
Det matematisk-naturvitenskapelige fakultet

IN1020 - Introduksjon til datateknologi

Kryptografi – Nøkkelen til digital samhandling

11.10.2024

Kristin Skar & Håkon Kvale Stensland

simula

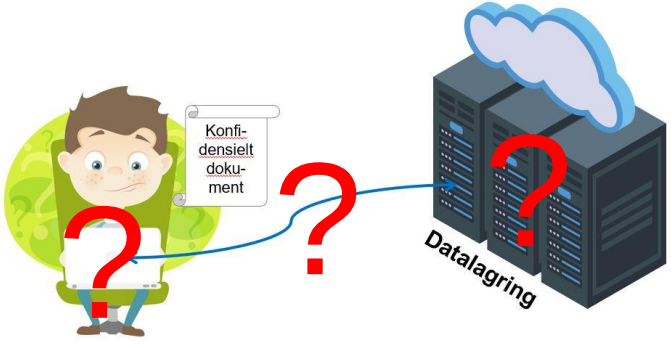
Plan for ”nettverksdelen” av IN1020

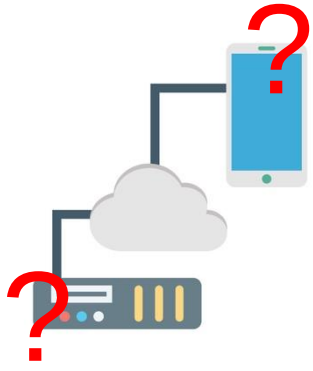
- 21. september - Introduksjon til operativsystemer
- 27. september – Datanett 101 – Introduksjon
- 4. oktober – Lagdeling i datakommunikasjon
- 9. oktober – Hvordan fungerer din trådløse ruter?
- **11. oktober – Kryptografi – Nøkkelen til digital samhandling**
- 16. oktober – Tjenester i Internett

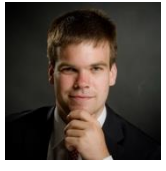
Introduksjon til kryptografi

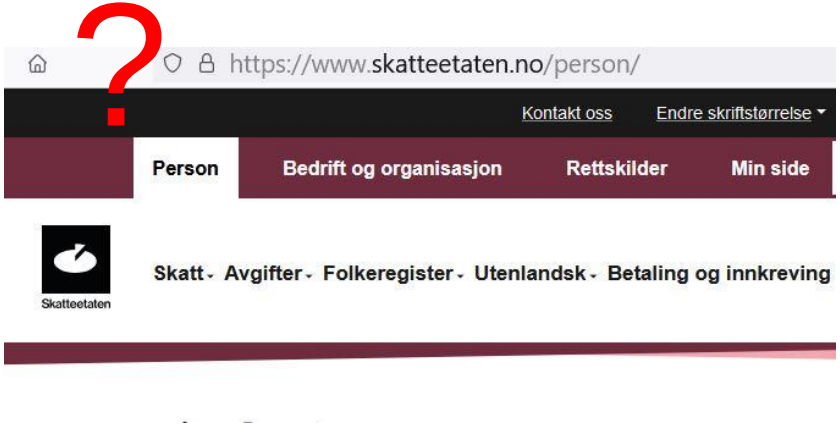
- Hvorfor trenger vi kryptografi?
- Hva er kryptografi?
- Hvordan ”knekke” en kryptering.
- Hemmelig nøkkelkryptering.
- Hash-algoritmer.
- Offentlig nøkkelkryptering.

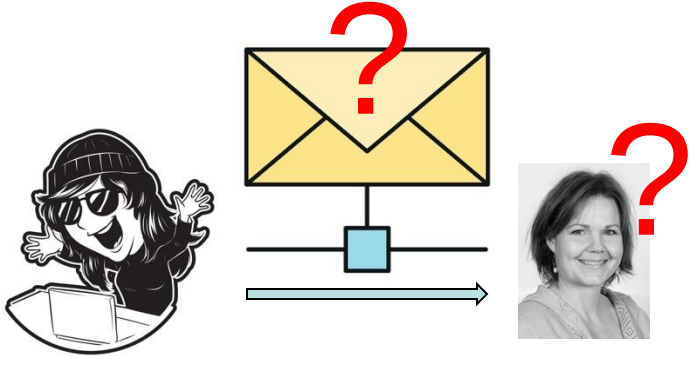
Trenger vi kryptografi?











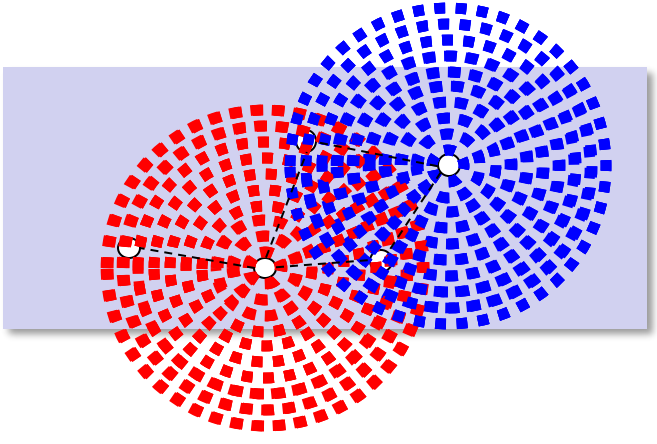
Repetisjon: Nettverksstrukturer

Broadcast-systemer:

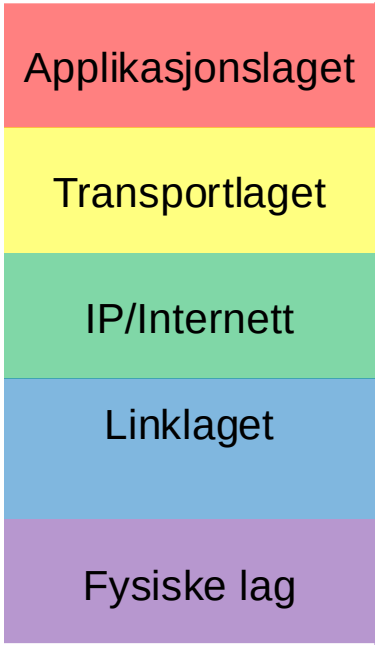
- *Radio:*
  - WiFi (IEEE 802.11)
  - Mobiltelefoni: 3G, 4G, 5G
  - Satellitt



- ***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

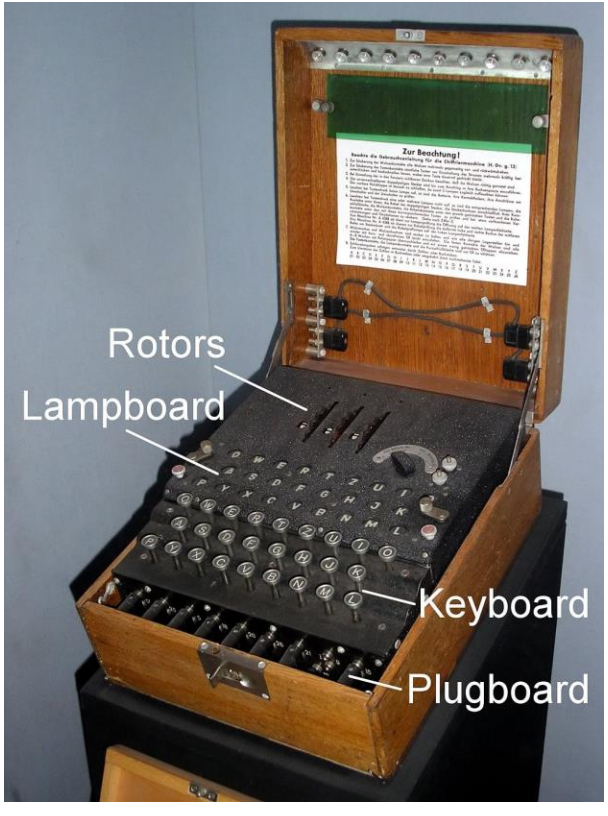


Hemmelige koder

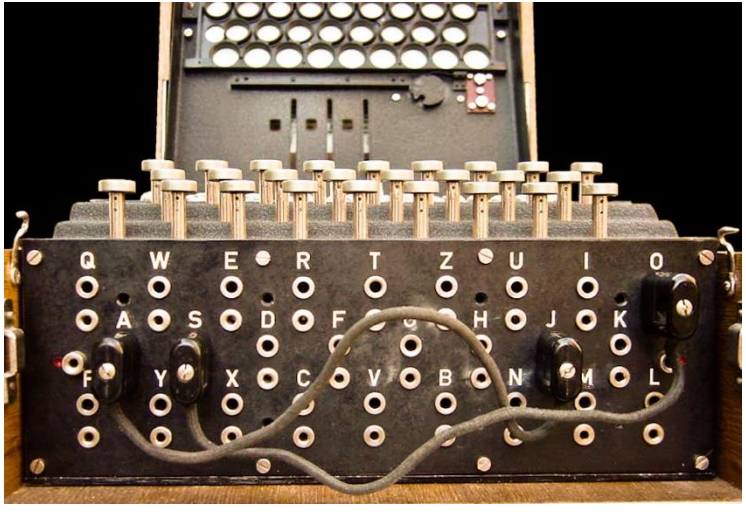
- Cæsarchiffer
  - Skifte bokstaver 3 ganger fremover
    - DOZEN → GRCHQ
- Dekoderringer
  - Substituer bokstaver *n* bokstaver videre (*n* = 1..25)
    - HAL → IBM (*n* = 1)
- Monoalfabetisk chiffer
  - Tilfeldig mapping (26! = 4.03291461 × 10^{26})
    - 1 ms / forsøk → 10M år… *MEN*, bokstavfrekvens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



Enigma



Enigma er basert på en polyalfabetisk chiffer



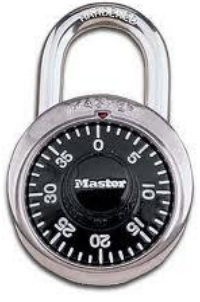
Images from Wikipedia: https://en.wikipedia.org/wiki/Enigma_machine

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 mot den Japanske marinen i WW2*
- *Finne svakheter i krypteringsalgoritme*
- Finne svakheter i implementasjonen av algoritmen

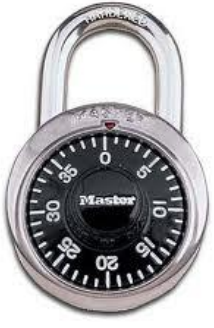
Kompleksitet vs. ”Brute-force”

- 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

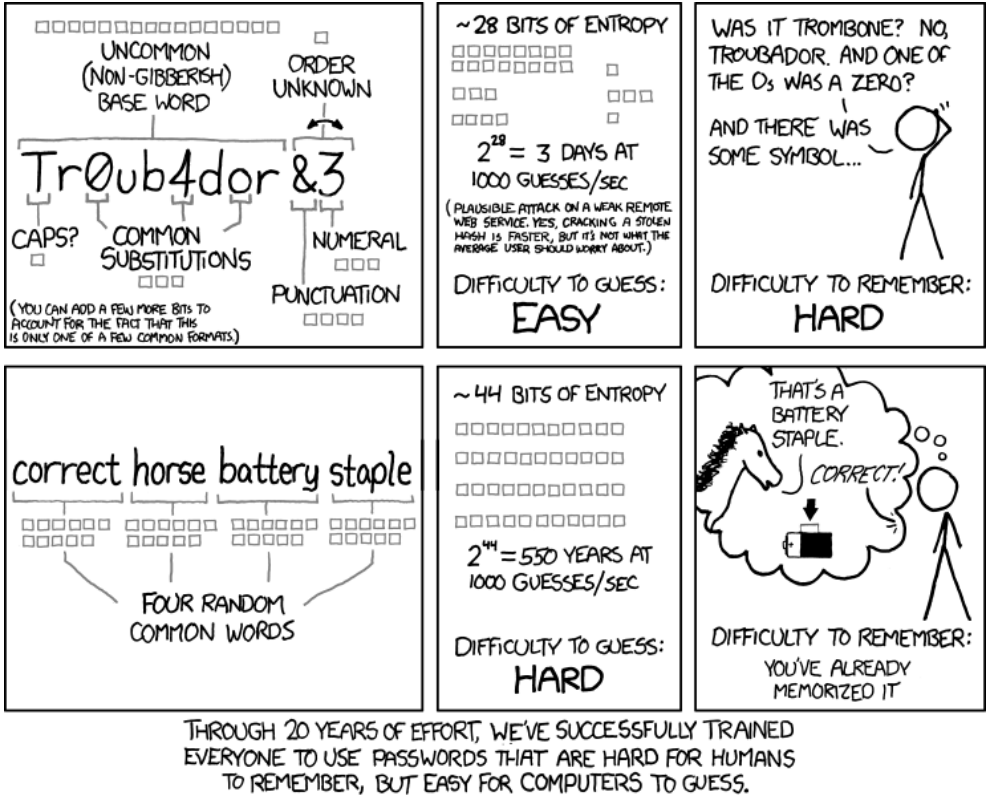
- Kombinasjonslås
  - 3 tall, mellom 1 og 40 – 10 sekunder per forsøk
    - Antall kombinasjoner: $40^3 = 64.000$
    - 178 timer
  - 4 tall, 13 sekunder per forsøk
    - $40^4 = 2,560.000$
    - 9244 timer



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

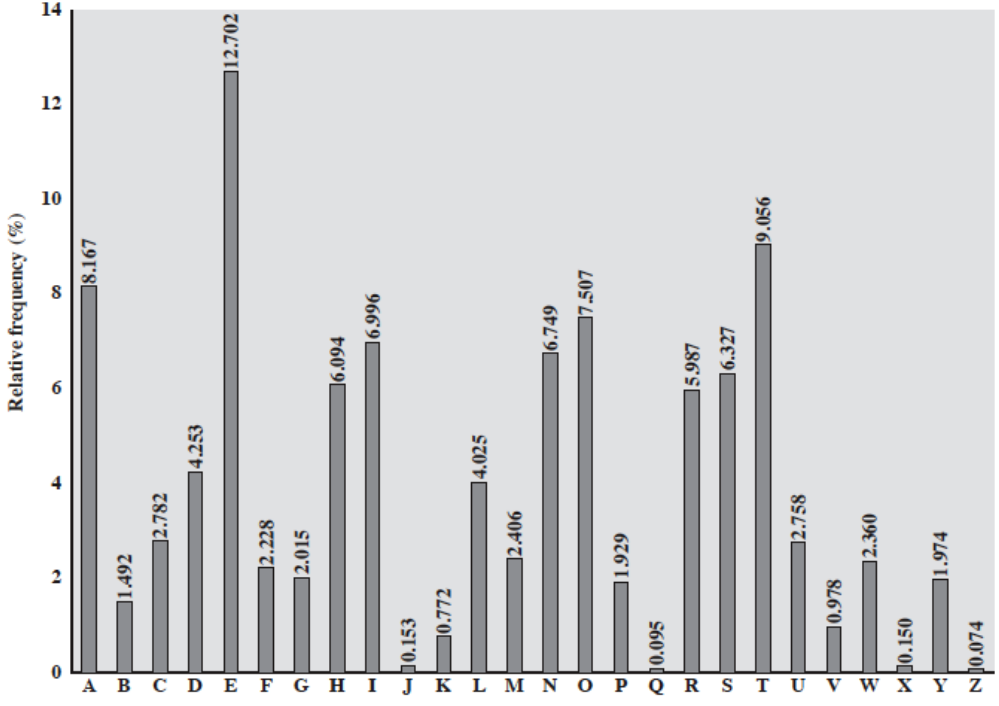


Figure 2.5 Relative Frequency of Letters in English Text

- 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)
- ”Brute-force”
  - Teste alle mulige kombinasjoner

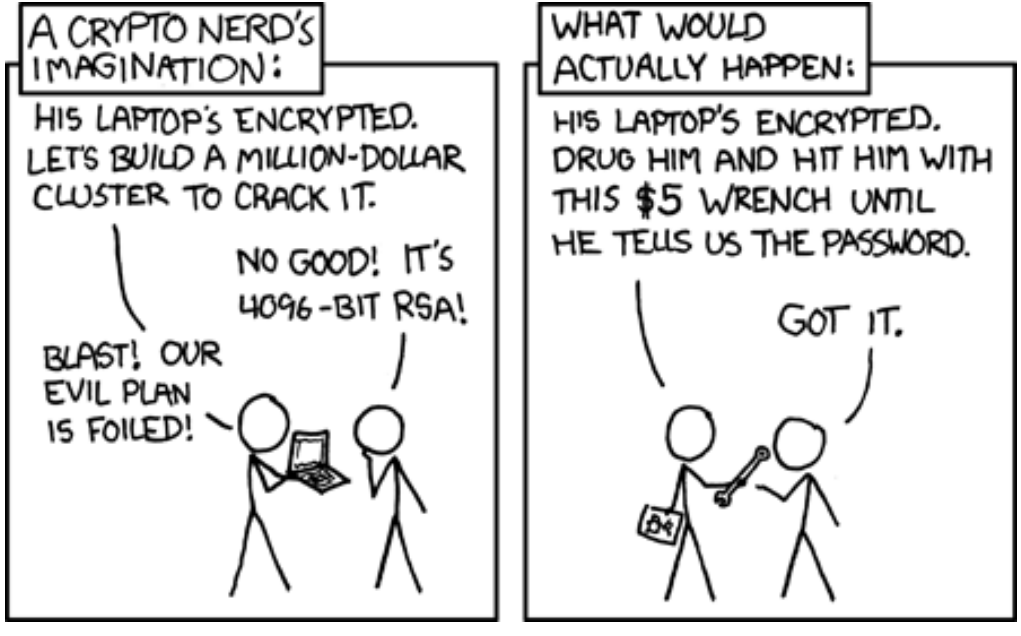


KRACK: Svakheter i WPA:
https://www.krackattacks.com/

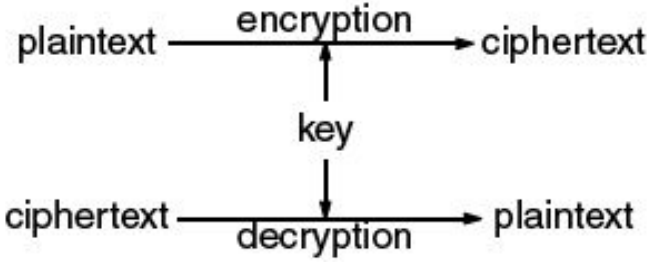
Number of Characters	Numbers Only	Lowercase Letters	Upper and Lowercase Letters	Numbers, Upper and Lowercase Letters	Numbers, Upper and Lowercase Letters, Symbols	Hardware
8	Instantly	6 secs	24 mins	2 hours	4 hours	RTX 2080
8	Instantly	6 secs	13 mins	52 mins	2 hours	RTX 3090
8	Instantly	1 sec	5 mins	22 mins	59 mins	RTX 4090
8	Instantly	Instantly	2 mins	7 mins	19 mins	A100 x8
8	Instantly	Instantly	1 min	5 mins	12 mins	A100 x12
8	Instantly	Instantly	Instantly	Instantly	1 sec	A100 x10,000 (ChatGPT)

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”

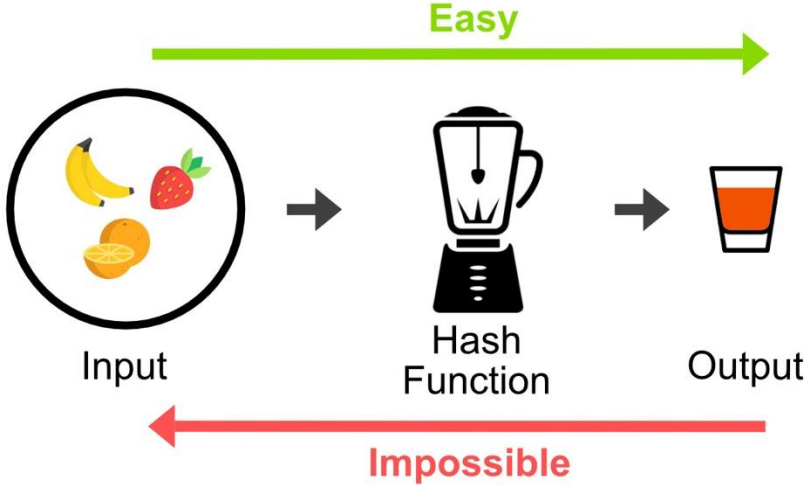
- To typer:
  - Block cipher (DES, 3DES, AES)
  - Stream cipher (RC4)
- Per dags dato er «block cipher» mest brukt, og gjerne i blokker på 128-bits (AES)

Symmetrisk kryptering for (økt) sikkerhet

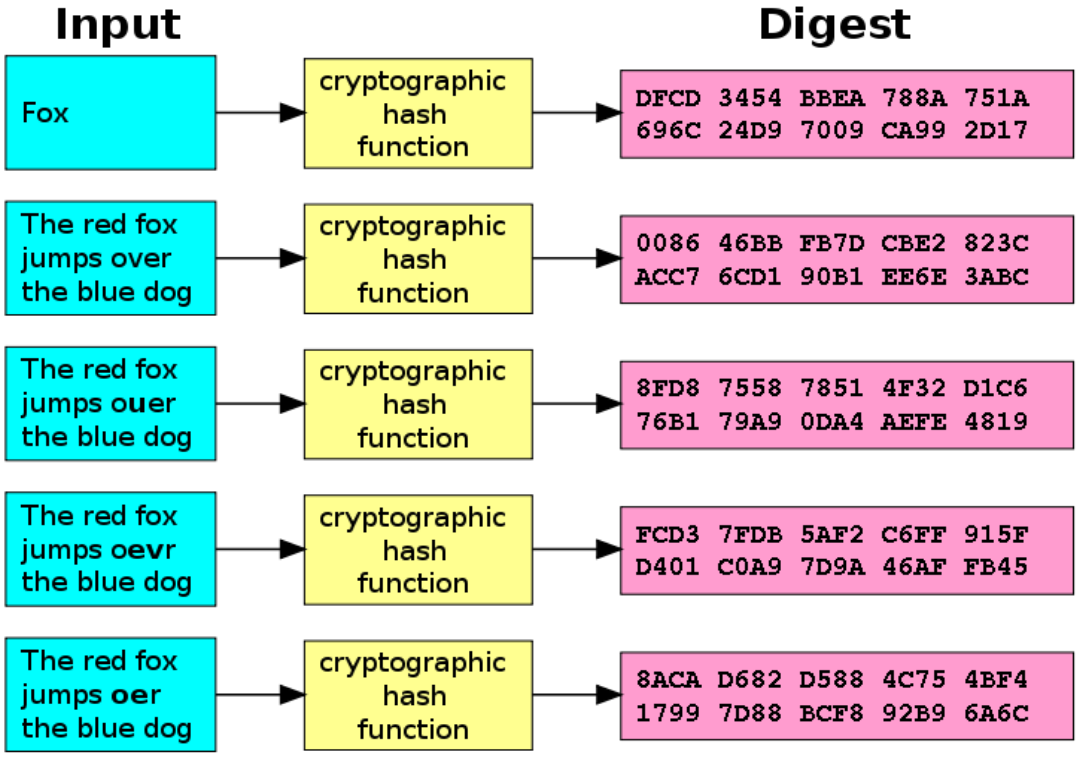
- **Anvendelse**:
  - Kryptering av data
  - Kryptering av meldinger
- **Hvilke sikkerhetsmål kan ivaretas?**
  - Konfidensialitet? **JA**
  - Integritet? I praksis **JA**
  - Autentisitet til dataopprinnelse? **JA**
  - Uavviselighet? **NEI**
  - Mottagers autentisitet **JA**
- **Farer**: Nøkkel på avveie? Nøkkel tapt? Hvordan dele hemmelig nøkkel trygt?

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

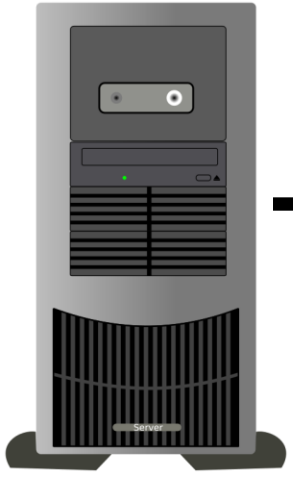


Hash-algoritmer



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)																												

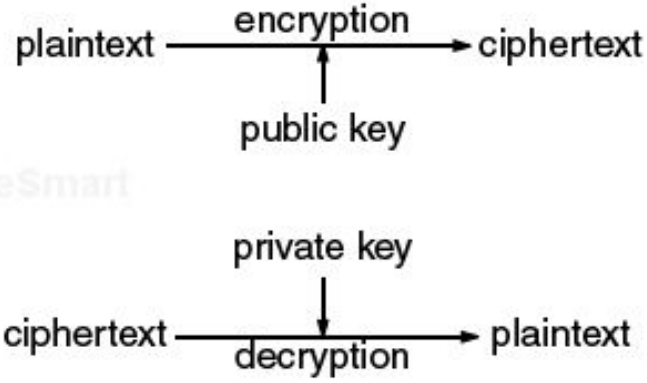
Key	Didn't exist/not public	Under peer review	Considered strong	Minor weakness	Weakened	Broken	Collision found

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

Hash-algoritmer for (økt) sikkerhet

- Anvendelse:
  - Sammenligning av filer (sjekk av integritet)
  - Beskyttelse av passord
  - Integritetssjekk, f.eks. av SW-distribusjoner
  - Integritetssjekk av meldinger (dataintegritet)
- **Hvilke sikkerhetsmål kan ivaretas?**
  - Konfidensialitet? **JA**
  - Integritet til data/melding? **JA**
  - Autentisitet til dataopprinnelse? **NEI**
  - Uavviselighet? **NEI**
- **Farer**?

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 2024

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

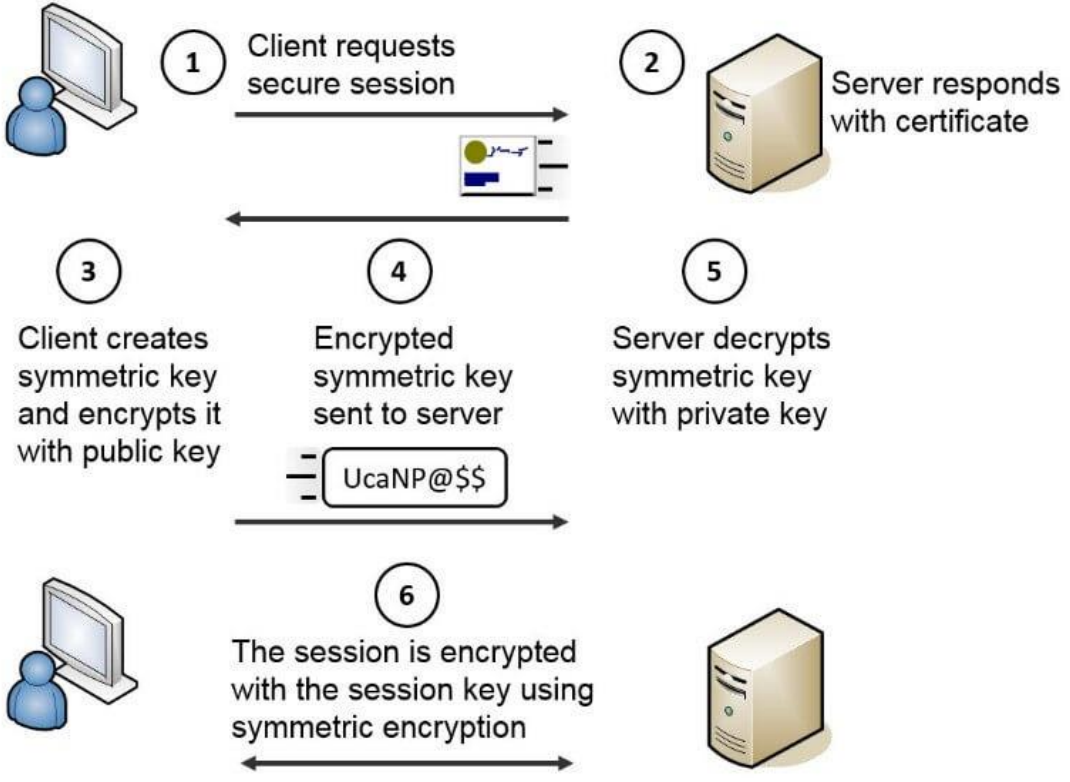
Eksamen i IN1020
Høsten 2024

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



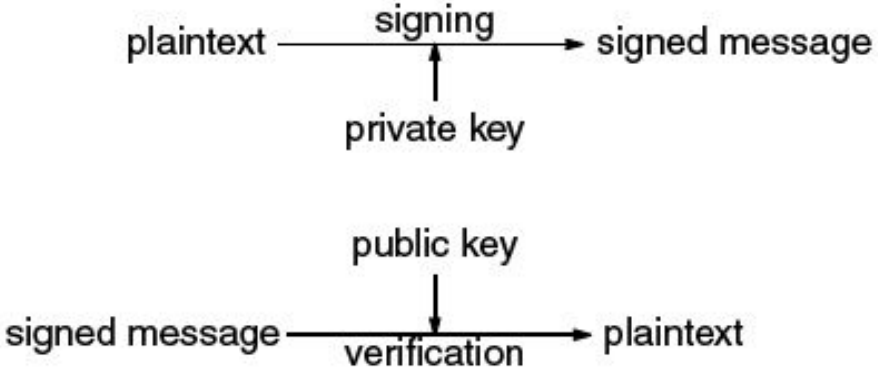
Asymmetrisk kryptering for meldingsutveksling:

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

- **Hvilke sikkerhetsmål kan ivaretas?**
  - Konfidensialitet? **JA**
  - Integritet? I praksis **JA**
  - Autentisitet til dataopprinnelse? **NEI**
  - Uavviselighet? **NEI**
  - Mottagers autentisitet **JA**

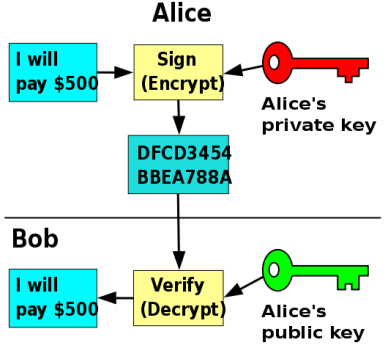
- Farer: Nøkkel på avveie, falske nøkler

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:** Sender krypterer (og dermed signerer) en melding med sin private nøkkel. Mottager dekrypterer (og dermed validerer) meldingen med senders offentlige nøkkel.
- **Anvendelse**: Ivareta tillitt til at meldinger kommer fra en hevdet avsender, autentisering.
- **Hvilke sikkerhetsmål kan ivaretas?**
  - Konfidensialitet? **NEI**
  - Integritet? I praksis **JA**
  - Autentisering av dataopprinnelse? **JA**
  - Uavviselighet? **JA**
  - Mottagers autentisitet? **NEI**
- Farer: Nøkkel på avveie, falske nøkler

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 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

Offentlig-nøkkel infrastruktur oppsummert:

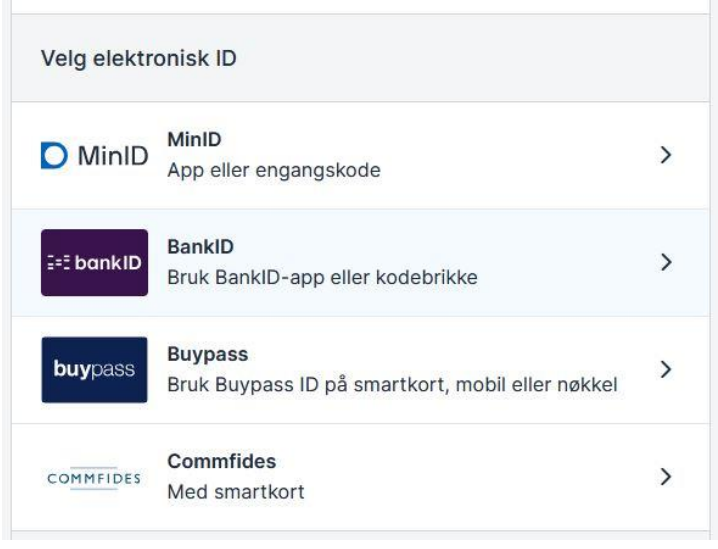
- ✓ Hver eneste offentlige nøkkel bakes inn i hvert sitt *elektroniske sertifikat* som knytter nøkkel og identitet sammen.
- ✓ Et *sertifikat* utstedes av en *tiltrodd sertifikatutsteder* (Certificate Authority), som går god for den offentlige nøkkelens ekthet.
- ✓ Den tilhørende private nøkkelen kan oppbevares (trygt!) på en datamaskin, på SIM-kortet i en 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 ☺)

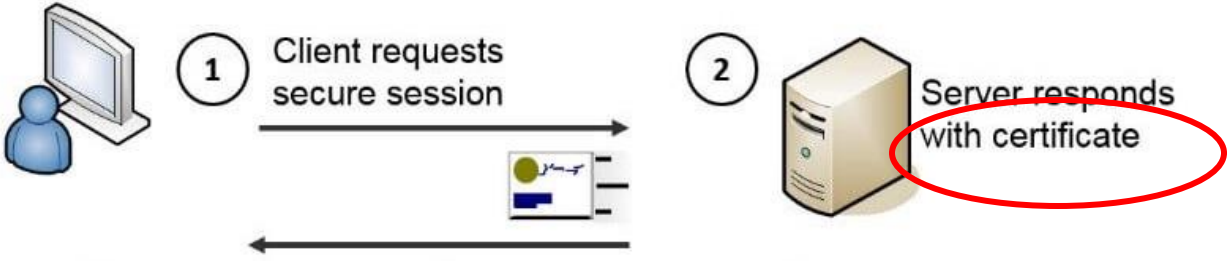
PKI i bruk

- PKI er grunnlag for forskjellige tjenester for elektronisk ID: MinID, BankID, Buypass, Commfide

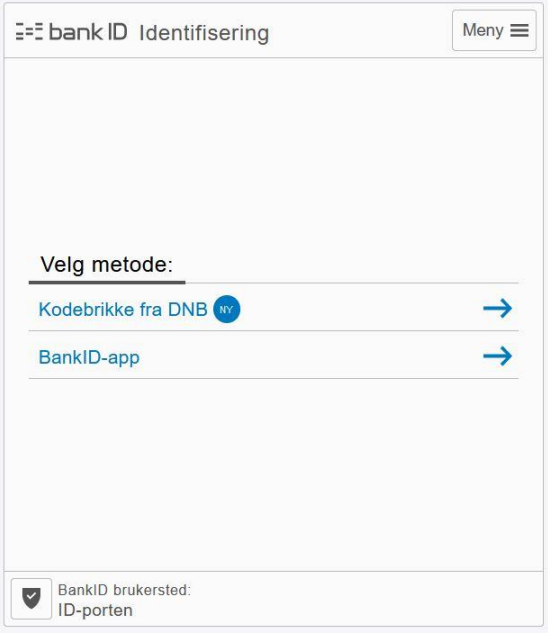
- Sikker kommunikasjon i helsevesenet:
  - eResept, elektronisk sykemelding, etc



- Internett-PKI (web-/nettleser-PKI):



Eksempel: BankID



«BankID er elektroniske sertifikater som en sertifikatholder kan benytte til å sikre elektronisk meldingsutveksling med andre sertifikatholdere ved å bekrefte rett identitet til partene (autentisering), sikre innholdet mot endring (integritet), knytte meldingen til en bestemt part (ikke-benekting) og/eller skjule innholdet for uvedkommende (kryptere)»

For den interesserte:

- https://vipps.no/documents/29/BankID_sertifikatpolicy_v2.0_for-kvalifiserte-sertifikater-fysiske-personer-banklagret.pdf
- https://www.bankid.no/en/bid_pds_personal/

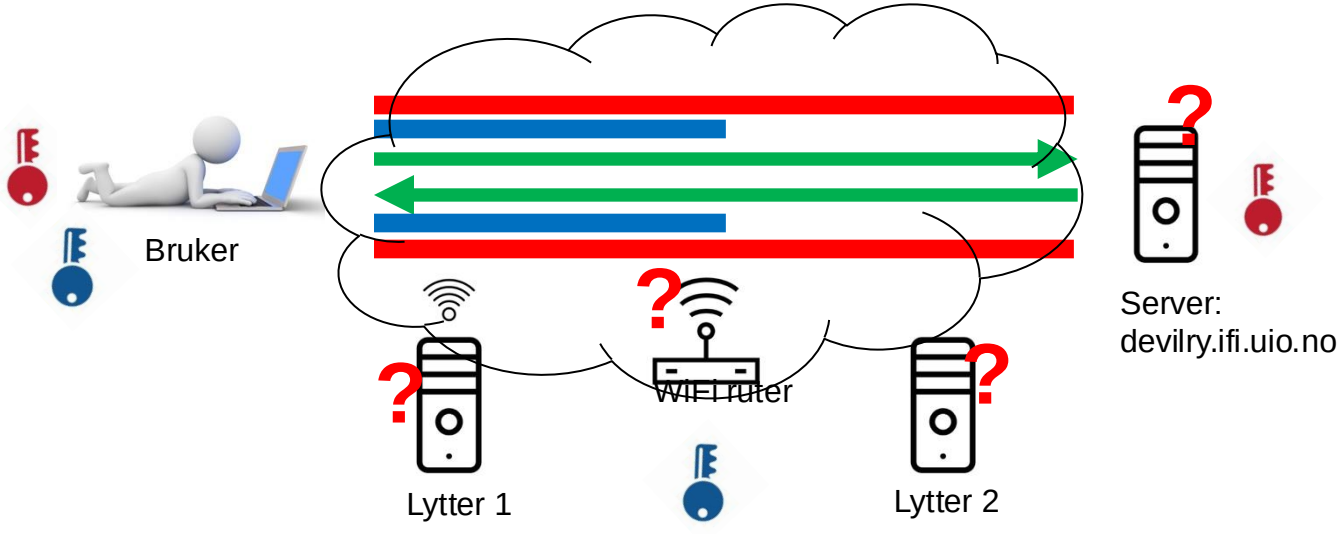
Kryptering oppsummert:

- Hovedmål:
  - Sikre konfidensialitet til data/meldinger – gjøre data uleselig for de som ikke har de riktige nøklene.
  - 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 uavviselighet i avtaleinngåelse (digital signatur)
- Krypteringsformer:
  - Hash-algoritmer (enveis-kryptering)
  - Symmetrisk (én hemmelig delt nøkkel)
  - Asymmetrisk (nøkkelpar: offentlig + privat (hemmelig) nøkkel)

Krypteringsformer, læringsmål:

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

Hvem kan lytte til hva?



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