KINS 2021

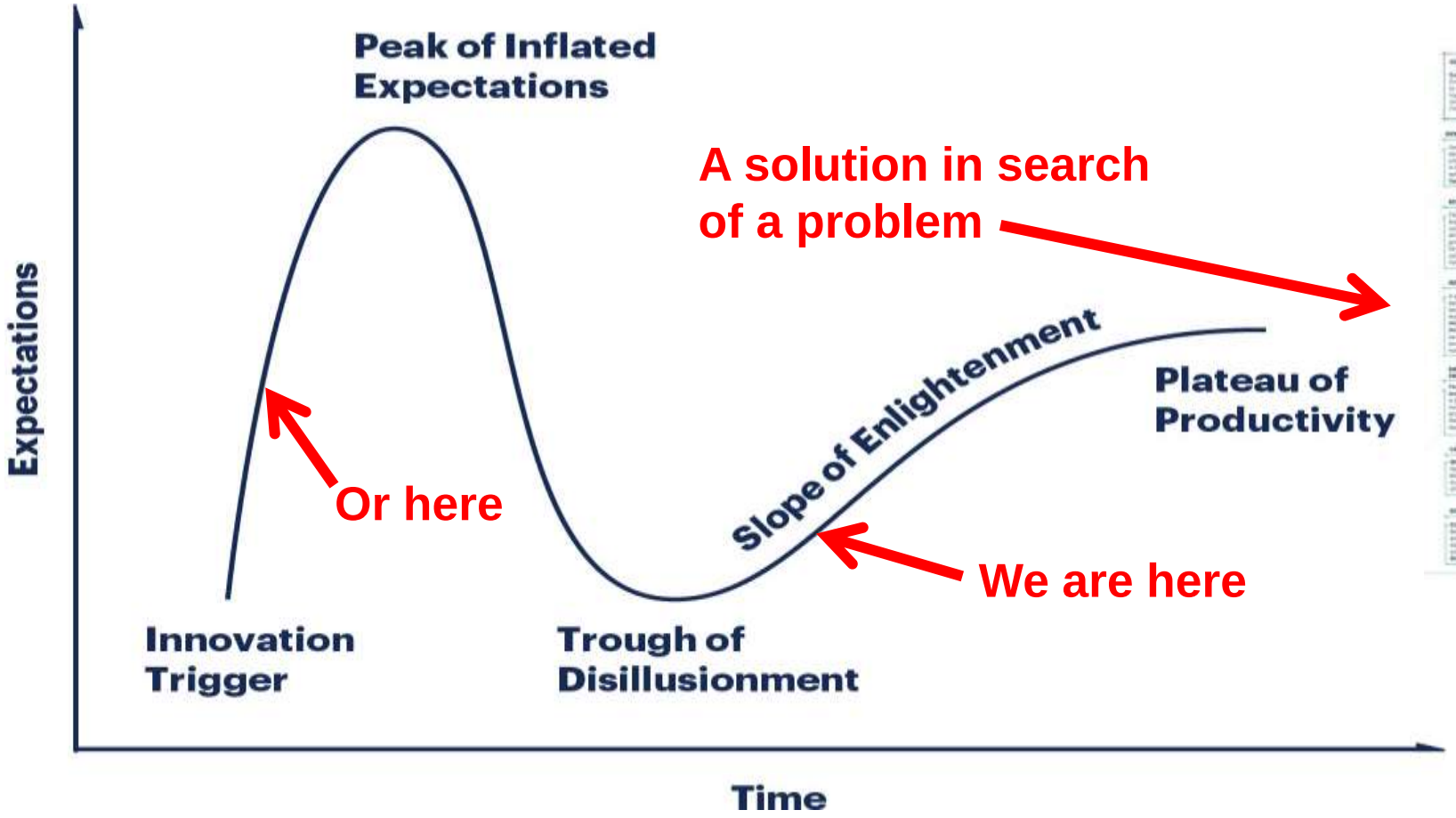
Blokkkjeder - hemmer eller fremmer det personvernet?

Eirik Gulbrandsen, senioringeniør, Datatilsynet

21. oktober 2021

Datatilsynet

Gartner Hype Curve



https://www.gartner.com/en/research/methodologies/gartner-hype-cycle

50+ BLOCKCHAIN
REAL WORLD USES CASES

THE INTERNET OF BLOCKCHAIN FOUNDATION

Blokkjeder - hemmer eller fremmer det personvernet?

Hva er blokkjeder egentlig?

Kryptovaluta; Bitcoin, Ethereum og «10 000» andre…

Ethereum; Smart Contracts

NFT – Non-Fungible Tokens…

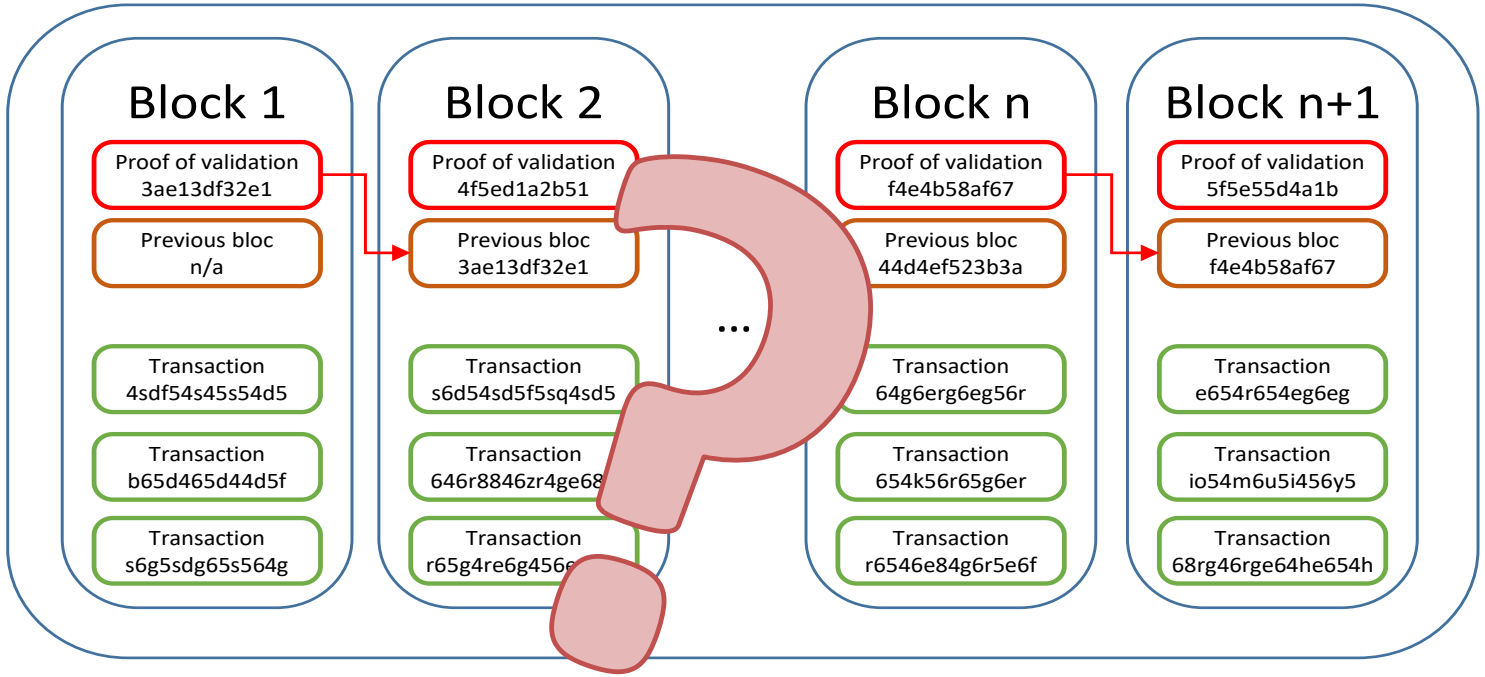
Hva må man tenke på i et personvernperspektiv?

Hva er en blokkjede?

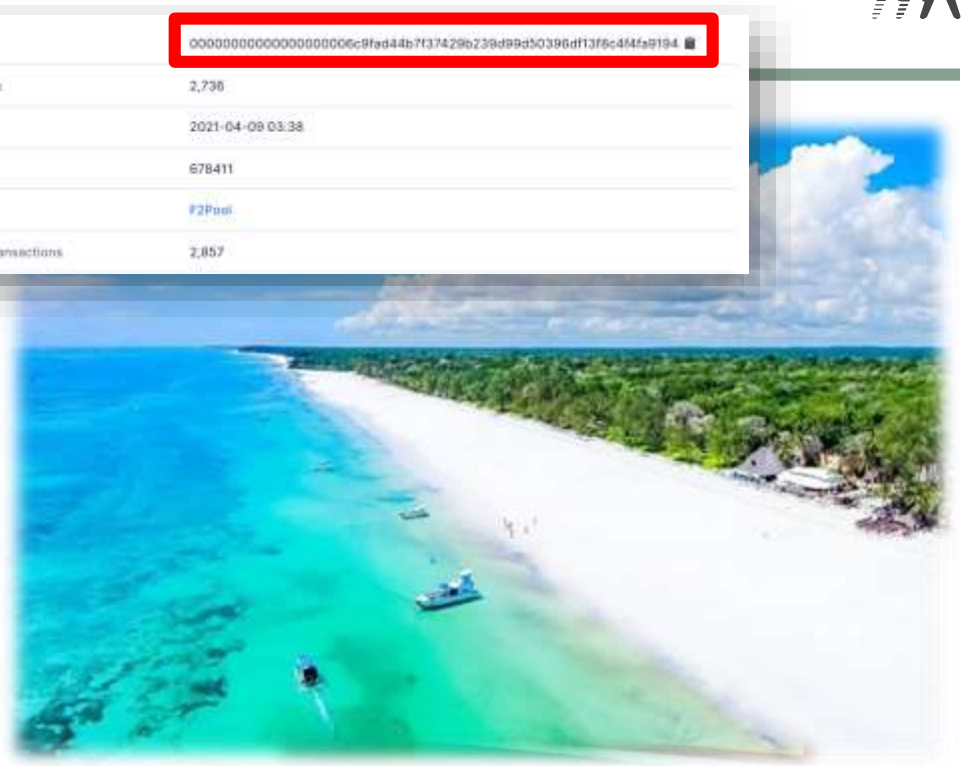
A decentralized, distributed ledger wherein transactions are verified by and immutably* recorded across the network, not a single central authority.

* unchangeable object

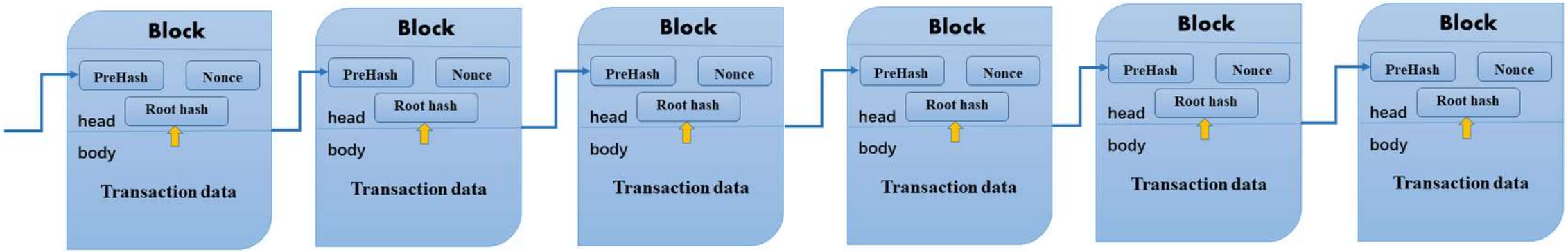
En desentralisert, skrivbar database som ikke kan endres i ettertid.



Hva er en blokkjede?



- digitale signaturer signerer transaksjoner
  - privat nøkkel
  - offentlig nøkkel
- → «body» (defineres av «token»-format)
- hashfunksjoner (avtrykksfunksjon) + tidsstempling = blokkjede / tidskjede / tillitskjede
  - «låser» et sett med transaksjoner (SHA256)
- proof-of-work (bevis på benyttet datakraft)
  - en hash med en definert egenskap
    - ledende nuller – «000000000000000000006c9fa…» (19)
  - gjetting av riktig «nonce» (et tilfeldig tall) er «diamanten i gruven»



Blokkjeder og smarte kontrakter

Blokkjeder → smarte kontrakter

Difficult to change

Changing smart contract processes is almost impossible, any error in the code can be time-consuming and expensive to correct.

Possibility of loopholes

According to the concept of good faith, parties will deal fairly and not get benefits unethically from a contract. However, using smart contracts makes it difficult to ensure that the terms are met according to what was agreed upon.

Third party

Although smart contracts seek to eliminate third-party involvement, it is not possible to eliminate them. Third parties assume different roles from the ones they take in traditional contracts. For example, lawyers will not be needed to prepare individual contracts; however, they will be needed by developers to understand the terms to create codes for smart contracts.

Vague terms

Since contracts include terms that are not always understood, smart contracts are not always able to handle terms and conditions that are vague.

Autonomy and savings

Smart contracts do not need brokers or other intermediaries to confirm the agreement; thus, they eliminate the risk of manipulation by third parties. Moreover, the absence of intermediary in smart contracts results in cost savings.

Redundancy

All the documents stored on blockchain are duplicated multiple times; thus, originals can be restored in the event of any data loss.

Safety and security

Smart contracts are encrypted, and cryptography keeps all the documents safe from infiltration.

Speed

Smart contracts automate tasks by using computer protocols, saving hours of various business processes.

Accuracy

Using smart contracts results in the elimination of errors that occur due to manual filling of numerous forms.

Smart Contract Platforms

- Ethereum – Simple contracts using Solidity; widely adopted
- EOS – High throughput; removal of transaction fees; scalability; DPOS (democracy as a POS); WASM language
- Stellar – Virtually no cost; connects banks, payment systems and people; five-second confirmation time; DTI issues
- Cardano – functional programming approach; robust platform; Plutus/Haskell
- Neo – “Ethereum of China”; smart economy; javascript
- Hyperledger Fabric – Linux Foundation; modular architecture; scalable

https://blockgeeks.com/guides/different-smart-contract-platforms/

Example of a Smart Contract

```
contract Mortal {
    /* Define variable owner of the type address */
    address owner;

    /* This function is executed at initialization and sets the owner of the contract */
    function Mortal() { owner = msg.sender; }

    /* Function to recover the funds on the contract */
    function kill() { if (msg.sender == owner) selfdestruct(owner); }
}

contract Greeter is Mortal {
    /* Define variable greeting of the type string */
    string greeting;

    /* This runs when the contract is executed */
    function Greeter(string _greeting) public {
        greeting = _greeting;
    }

    /* Main function */
    function greet() constant returns (string) {
        return greeting;
    }
}
```

https://www.ethereum.org/greeter

Solidity is a contract-oriented, high-level language for implementing smart contracts. It was influenced by C++, Python and JavaScript and is designed to target the Ethereum Virtual Machine (EVM)

Features of a Smart Contract

Smart Contracts are automated, self-executing contracts with specific instructions written on its code (in the blockchain) which get executed when certain conditions are met:

- Deterministic – Same output given the same inputs
- Terminable – Predetermined steps, fees or completeness
- Isolated – Preventing tainting of entire ecosystem

- Electronic agreement
- Uses blockchain technology and coding
- No third party (TI) required to validate contract

https://blockgeeks.com/guides/different-smart-contract-platforms/

Blokkjeder og NFT (non-fungible tokens)

Blokkjeder og NFT (non-fungible tokens)

ERC20 / ERC 721 / ERC 1155

Blokkjeder og NFT (non-fungible tokens)

ERC20 / ERC 721 / ERC 1155

Blokkjeder og personvern

Blokkjeder og personvern

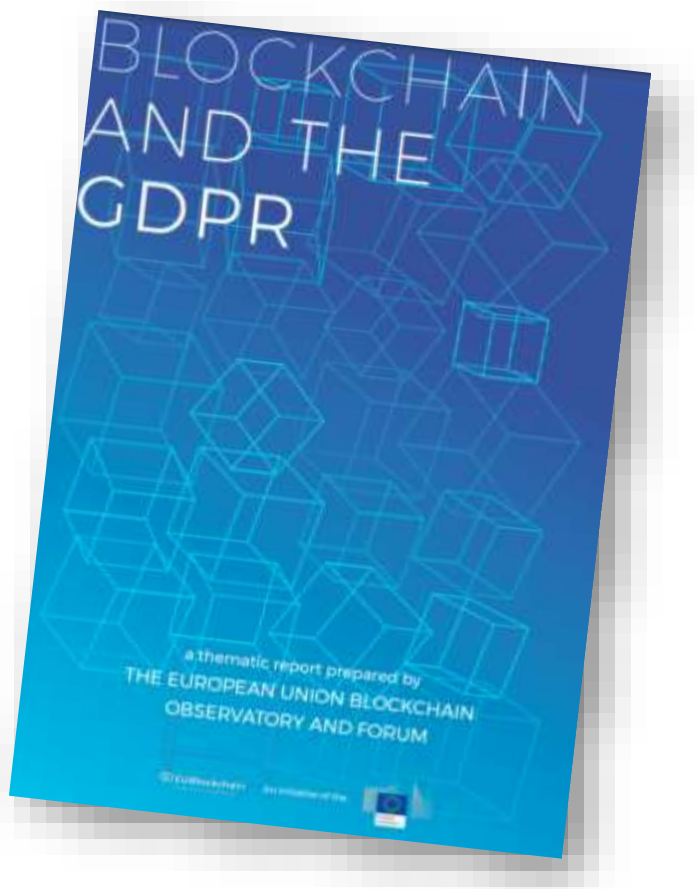
Risiko knyttet til blokkjedeteknologien:

- Generelle (standardisering)
- Sikkerhetsrelaterte (KIT)
- Juridiske (databehandler, håndhevelse av rettigheter)

Arbeid i EDPB*

- EDPBs Technology Expert Subgroup jobber med å utarbeide en veileder om Blockchain-teknologien

- Sentrale problemstillinger:
  - Behandling og ansvar
  - Forenlighet med personvernprinsippene
  - Innebygd personvern
  - Risiko og DPIA
  - De registrertes rettigheter



https://www.eublockchainforum.eu/sites/default/files/reports/20181016_report_gdpr.pdf

*European Data Protection Board

Behandling og ansvar

- Personopplysninger kan lagres i blockchain:
  - I brukeridentifikator (f eks digitale signaturer)
  - I payload data («body»)

- Utfordring: Personopplysninger lagret på en åpen blokkjede vil alltid være synlig for alle brukere
  - Vurder nødvendigheten av de ulike arkitektoniske elementene i blokkjeder
  - Personopplysninger bør i størst mulig grad lagres «off-chain»

- Hvem er behandlingsansvarlig?
  - Noder verifiserer gyldigheten av «blokker»
  - Aktører samler inn opplysninger brukt i transaksjoner og legger det inn i blokkjeden

Blokkjeder og personvernprinsippene

- Teknologien er nøytral ift. flere prinsipper:
  - Rettferdighet, åpenhet, formålsbestemthet, riktighet
  - Ikke teknologien i seg selv som utgjør en utfordring, opp til behandlingsansvarlige

- Utfordrende ift.:
  - Dataminimering (lagring over tid)
  - Lagringsbegrensning (sletting, f eks når formål er oppnådd)
  - Integritet og tilgjengelighet (inkl nødløsninger)

- Om lovlighetsprinsippet:
  - Ikke samme behandlingsgrunnlag for alle transaksjoner
  - Bevisstgjøre brukere av konsekvensen av «åpen bok»

De registrertes rettigheter

- Innsynsrett og dataportabilitet ikke problematisk

- Rett til sletting og rett til å protestere
  - Behandles samlet fordi ved protest må opplysninger slettes
  - Sletting kan muligens oppnås ved å gjøre personopplysningene praktisk utilgjengelige
    - Slette referansenøkkel – spørsmål om dette tilsvarer reell sletting

- Rett til retting
  - Ikke mulig å endre opplysninger som allerede er i blokkjeden
  - Opplysninger må eventuelt «slettes» og så lage ny transaksjon med korrigerte opplysninger.

Innebygd personvern

- Blokkjedeteknologien har iboende egenskaper som kan være vanskelig å forene med kravet om innebygd personvern

- Kan være mulig å løse utfordringene case by case ved å ta i bruk key design and default elements fra veileder om innebygd personvern.

Programvareutvikling med innebygd personvern

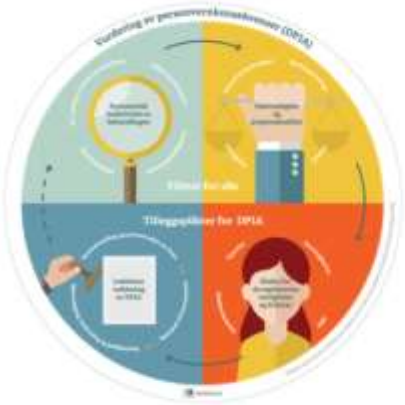
Denne veilederen skal hjelpe norske virksomheter å forstå og etterleve kravet om *innebygd personvern* i personvernregelverket. Den er utarbeidet i samarbeid med sikkerhetseksperter og programutviklere i privat og offentlig sektor. Veilederen har også vært på høring i flere virksomheter og organisasjoner.

https://www.datatilsynet.no/rettigheter-og-plikter/virksomhetenes-plikter/innebygd-personvern/programvareutvikling-med-innebygd-personvern/

Risiko og DPIA (personvernkonsekvensutredning)

Blockchain krever DPIA (Data Protection Impact Assessment)

- "Innovative use or applying new technological or organisational solutions" utgjør en sannsynlighet for høy risiko jf. DPIA-veilederen



Vurdering av personvernkonsekvenser (DPIA)

En vurdering av personvernkonsekvenser (Data Protection Impact Assessment - *DPIA*) skal sikre at personvernet til de som er registrert i løsningen ivaretas. Dette er en plikt etter personvernregelverket. Artikkel 35 definerer når det er påkrevd å gjøre en DPIA, hva den skal inneholde og hvem som skal gjennomføre den.

https://www.datatilsynet.no/rettigheter-og-plikter/virksomhetenes-plikter/vurdere-personvernkonsekvenser/vurdering-av-personvernkonsekvenser/

Innhold

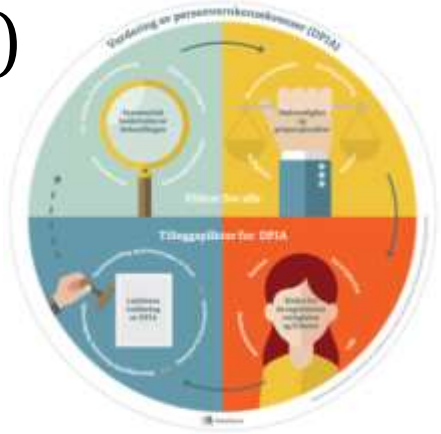
Blokkjeder og personvern

Blokkjeder har egenskaper som både fremmer og utfordrer personvern

Utfordringer kan adresseres gjennom arkitektur og løsningsdesign

Bruk prinsippene beskrevet i veileder «Innebygd personvern»

Bruk av blokkjeder vil høyst sannsynlig kreve personvernkonsekvensutredning (DPIA)



Eirik Gulbrandsen, Senioringeniør, Datatilsynet
eirik.gulbrandsen@datatilsynet.no

Datatilsynet

postkasse@datatilsynet.no
Telefon: +47 22 39 69 00

datatilsynet.no
personvernbloggen.no