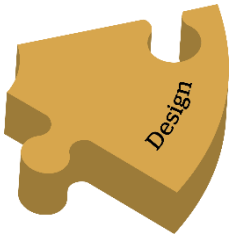
3 Design



Sjekklisten er dynamisk, ikke uttømmende og skal oppdateres regelmessig. Dersom du har innspill til noen av punktene, vil vi gjerne høre fra deg.

Dataorienterte designkrav

Minimere og begrense. Mengden av innsamlet og prosessert data skal begrenses til det tillatte og absolutt nødvendige. Opplysningene skal slettes når det ikke er behov for dem lenger, «Select before you collect».

Eksempler:

- Gå gjennom vurdering av personvernkonsekvenser (DPIA).
- Vær sikker på at behovet for personopplysninger samsvarer med mengde og omfang. Ikke samle inn flere opplysninger enn det som er nødvendig. Begrens mengden av informasjon som behandles på enheter og områder med lavere tillit.
- Unngå, begrens eller minimer behovet for å samle inn og behandle sensitive personopplysninger.
- Begrens og minimer eksponering av unødvendig funksjonalitet og personopplysninger i brukergrensesnittet. Vurder for eksempel om det er nødvendig å lagre direkte identifiserende opplysninger i programvaren eller om pseudonyme opplysninger er tilstrekkelig.

Gjem og skjul. Alle personopplysninger og sammenhengen mellom dem, bør ikke kommuniseres, behandles eller lagres i klartekst. Ved å skjule direkte identifiserende opplysninger fra visning i klartekst, reduseres risiko for misbruk og omfang av hendelser være betydelig. Tiltak for dette kan være kryptering, aggregering og pseudonymisering av personopplysninger.

Eksempler:

- Gjennomfør en trusselmodellering og «analyse og reduksjon av angrepsflaten» ved design av programvare.
- Bruk krypteringsmekanismer for å sikre overføring, behandling og lagring. Dette er særlig viktig dersom det er behov for overføring av personopplysninger over ukontrollerte områder og nettverk.
- Anonymiser eller pseudonymiser personopplysninger dersom det er mulig.
- Unngå unødvendig eksponering av kommunikasjonsmønster og tilkoblinger (som API-er, feeds, gateways, påloggingsgrensesnitt m.m.).

Separere. Personopplysninger bør separeres. Det betyr at de lagres i adskilte databaser, entiteter og områder for hvert formål og behandling. Ved å separere behandlinger og lagring av personopplysninger tilknyttet en enkeltperson, reduseres muligheten for å lage komplette profiler av hver enkelt registrerte. Separasjon er også en god måte å oppnå formålsbegrensning, samt urettmessig kobling og lenking mellom ulike datasett. Tabeller med personopplysninger bør ha kortere lagringstid med tidsfrist for automatisk sletting, mens tabeller uten personopplysninger kan lagres lengre. Tiltak for å oppnå dette kan være tilgangsstyring til tabeller, splitting av databasetabeller, skille mellom enheter med høy tillit og lav tillit, skille tilgang til områder i forhold til behov.

Eksempler:

- Skill sensitive personopplysninger fra mindre sensitive personopplysninger (i databasen, tilgang til områder, på klienter og enheter med mer).
- Splitt tabeller i databasen og tilhørende tilgangsrettigheter til tabeller, samt områder, etter «tjenstlig» behov (minste privilegiums prinsipp).
- Skill enheter og områder med lavere tillit fra enheter med høyere tillit.
- Rader i tabeller bør være vanskelig å lenke til hverandre.

Aggregere. Personopplysninger bør samles inn og behandles mest mulig aggregert for å ivareta den registrertes personvern, uten at dette går utover forretningsmessig verdi og formålet med innsamling og bruk.

Eksempler:

- Reduser bruken av detaljerte og sensitive personopplysninger.
- Fjern unødvendig informasjon og overskuddsinformasjon. For eksempel
  - «løft» tidsenheten ved å bruke uker istedenfor dager eller timer
  - bruk fylkes- og regionsnivå isteden for gateadresser ved allokering av personer eller enheter
  - bruk grupperinger fremfor enkeltpersoner
- Bruk anonymiseringsteknikker.

Personvern som standard. Alle innstillinger skal, som standard, være konfigurert med den mest personvernvennlige innstillingen. Den registrerte skal selv gjøre et bevisst valg om å åpne opp for mindre personvernvennlige innstillinger ved å for eksempel gjøre det mulig å dele mer data om seg selv med andre.

Eksempler:

- Sporing av en enhet skal være deaktivert som standard.
- Blåtann (Bluetooth) skal være deaktivert som standard.
- Sporing fra et nettsted til et annet skal være deaktivert som standard.
- Det skal være standard å ikke gjenbruke opplysninger om hvilke nettsteder den registrerte har gjort oppslag på.

Prosessorienterte designkrav

Informere. Programvaren bør designes slik at den registrerte er tilstrekkelig informert om hvordan programvaren fungerer og hvordan personopplysninger behandles. Den registrerte skal ha informasjon om at profilering og automatisert behandling av personopplysninger foregår. Han eller hun skal også få vite hvordan dette skjer. Det er viktig å huske på at det gjelder spesielle krav dersom programvaren skal rettes mot barn.

Eksempler:
Det etableres en nettside med informasjon om programvaren som er tilgjengelig for den registrerte før vedkommende tar programvaren i bruk:

- Oppgi eventuelt kontaktpunkt, kontaktskjema eller lignende hvor den registrerte skal henvende seg for å få informasjon om
  - hensikt og behovet for innsamlingen

    - sikkerheten i programvaren (hvordan informasjonen er sikkerhetsmessig beskyttet)
    - bruk av underleverandører, og eventuell deling av informasjonen med underleverandører
- Bruk flere tilnærminger og kanaler for å sikre at man når frem til alle brukere og brukergrupper.
- Bruk et enkelt og forståelig språk.
- Bruk flere språk dersom det er nødvendig.
- Berik og varier med bilder, ikoner, lyd, video og så videre.

Kontrollére. Den registrerte har rett til å kontrollere sine personopplysninger. Det innebærer blant annet rett til innsyn, korrigering og sletting av egne opplysninger. Ved automatisert behandling der det tas avgjørelser uten menneskelig innblanding, kan den registrerte kreve manuell behandling. Programvaren bør designes slik at den registrerte kan ivareta disse rettighetene på en enklest mulig måte.

Eksempel:
Etabler og tilgjengeliggjør funksjonalitet for

- å ha oversikt over hvilke behandlinger av personopplysninger som er nødvendige for å oppfylle kontrakten, og hvilke det er frivillig å samtykke til
- at den registrerte kan gi sitt samtykke på en informasjonsside med en sjekkboks før han eller hun tar i bruk programvaren
- å kunne trekke tilbake et samtykke via en meny i programvaren. Husk at innsamling av personopplysninger skal opphøre hvis samtykket trekkes tilbake
- å kunne gi tilgang til å rette, blokkere eller slette personopplysninger for eksempel ved at innsamlet informasjon viser direkte til den registrerte i programvaren
- å kunne sørge for gjennomgående sletting av personopplysninger i databasen og andre steder der de er lagret (eksempelvis backup). Informasjonen kan også eksporteres til en fil eller papirversjon for manuell gjennomgang med en tilhørende rutine for at den registrerte skal kunne rette, blokkere eller slette opplysninger (innen 30 dager).
- å kunne avslutte en kontrakt/avtale, installere, avinstallere, aktivere og deaktivere et program, tjeneste, teknisk komponent eller et system via funksjonalitet i en meny, på en egen side eller manuelt via et skjema
- å kunne sende inn spørsmål eller klager knyttet til personvern og sikkerhet. Alternativt kan informasjonen gjøres tilgjengelig på en nettside med kontaktinformasjon som er knyttet til en bemannet kanal som håndterer henvendelsene (en dokumentert rutine må da etableres).
- å kunne motsette seg profilering ved å tilgjengeliggjøre funksjonalitet for at brukere skal kunne ta et bevisst valg på å motsette seg profilering og videredistribusjon av personopplysninger. Dette kan for eksempel gjøres i en meny med en sjekkboks og et flagg som lagres i databasen. Alternativt manuelt via en bemannet kontaktkanal.
- krav til åpenhet og informasjon om hvordan automatiske avgjørelser tas, samt mulighet for den registrerte til å kreve manuell behandling. Dette er beskrevet i systemdokumentasjonen og tilgjengeliggjøres på en informasjonsside i programvaren.

Håndheve. Programvaren bør designes slik at det kan dokumenteres at den registrertes personvern blir ivaretatt. Dokumentasjonen bør omfatte ansvarsforhold og hvordan personvernregelverket

håndheves. Den må være tilgjengelig ved revisjon og kontroll av behandlingen. Slik dokumentasjon omfatter også programvare knytte til kunstig intelligens, profilering og automatisert behandling.

Eksempler:

- Programvaren skal sette høyeste personverninnstilling som standard:
  - Innstillinger presenteres i en meny hvor den registrerte selv må ta bevisste valg og gjøre en aktiv handling for å «endre» til innstillinger som er mindre personvernvennlig.
  - Dersom programvaren på et senere tidspunkt må endres til en mindre personvernvennlig innstilling, informeres den registrerte tydelig om denne endringen og kan samtykke til den ved å klikke på en sjekkboks etter å ha fått presentert endringen.
- Programvaren skal ivareta krav om dataportabilitet:
  - Det er funksjonalitet for at den registrerte selv kan be om å få utlevert personopplysninger om seg selv eller be om at opplysningene blir overført til en annen tjenestetilbyder i et standardisert og gjenbrukbart format.
  - Retten gjelder når behandling av basert på samtykke eller kontrakt.
  - Den registrerte kan be om at personopplysninger eksporteres og overleveres på en sikker måte (manuelt i tradisjonell post, eller tilsendt på en annen sikker måte) til ny tjenestetilbyder.
- Samtykke skal kreve en aktiv handling fra brukere:
  - Et samtykke, eller endring av samtykke, gjøres ved å fylle ut en tekstboks eller klikke på en sjekkboks, som lagrer et «flagg» i databasen.
  - Programvare rettet mot unge og mindreårige skal ha funksjonalitet som krever samtykke fra foreldrene før det gis tilgang. Funksjonaliteten må sikre eller bekrefte at vedkommende er myndig. Alternativt kan det lages rutiner for å be om et manuelt dokumentert samtykke fra foresatte.
- Den registrerte gis oversikt over hvilke tilganger (inkluderer innsyn, endringer) som er gjort, når og av hvem og hvilke samtykker man har gitt. Oversikt over tilganger fremgår av logg som kan gjøres tilgjengelig for den registrerte.

Demonstrere. Den behandlingsansvarlige skal kunne dokumentere etterlevelse av personvernregelverket og at krav til informasjonssikkerhet er oppfylt. Programvare skal tilrettelegges og utformes slik at den behandlingsansvarlige kan dokumentere og vise hvordan personvernregelverket er implementert og ivaretatt. Slik dokumentasjon kan for eksempel bestå av en rapport som viser at programvaren er utviklet etter metodikk som ivaretar innebygd personvern og informasjonssikkerhet, rapporter fra sikkerhetsrevisjoner, sikkerhetstester som penetrasjonstester og rapporter etter øvelser på hendelseshåndtering knyttet til personvern.

Analysere og redusere angrepsflaten til programvaren som utvikles

- Analyser den ferdigdesignede programvarens angrepsflate for å redusere muligheter til å utnytte svake punkter og sårbarheter i programvaren.
- Gjennomgå designet og analyser hvor det er mulig å ta imot input av data og hvor data sendes.
- Undersøk om samme type informasjon samles inn flere steder (dupliserende funksjonalitet), og vurder om funksjonaliteten kan forenkles.
- Reduserer sannsynligheten for feil ved å forenkle programvaren og fjerne unødvendig funksjonalitet.
- Gjenbruk vurderingene av sikkerhetsrisiko og personvernkonsekvenser som er gjennomført i kravfasen.

- Implementer sårbarhetsreduserende tiltak for å oppnå akseptabelt toleransnivå for personvern og sikkerhet hvis analysen viser at dette ikke er tilfredsstillende.
- Gjenbruk toleransenivå som ble utarbeidet i kravfasen.
- Sørg for å dokumentere analysen og reduksjonen av angrepsflaten.

Trusselmodellering

- Analyserer komponenter, tilgangspunkter, dataflyt og prosessflyt i programvaren.
- Sørg for at de involverte i utviklingsteamet analyserer hvordan programvaren kan misbrukes ved ulike scenarioer.
- Gjennomgå hvert av scenarioene for å se hvordan designet kan forbedres for å unngå trusler som er identifisert. Dette gjøres ved å implementere sårbarhetsreduserende tiltak. Resultatet av dette blir en mer herdet og robust programvare.
- Gjør en risikovurdering av sårbarheter som gjenstår og som må håndteres ved andre tiltak. Sørg også for å føre disse sårbarhetene inn i et risikoregister.

Eksempler på verktøy

- designprinsipper for god sikkerhet, med blant annet:
  - minste privilegiums prinsipp (Principle of least privilege)
  - sikkerhet i dybden (Defense in depth)
  - feil sikkert (Fail Securely)
- threat Modelling
- attack-Surface-Analysis
- use Case og Misuse-Case Modelling
- angrepstre/Attack-Tree
- DREAD
- STRIDE

Hvorfor stille designkrav?

- Personopplysninger skal behandles lovlig, og programvarens design må gjenspeile dette, jfr personvernforordningen artikkel 6. Vær derfor bevisst på hvilke type personopplysninger som skal behandles, formål og vilkår for behandlingen, samt sammenstilling av personopplysninger.
- Den registrertes rettigheter må gjenspeiles i designet, jfr. personvernforordningen artikkel 12-23.