

WHITE PAPER

Hvordan applikasjoner skal autentisere seg i Helse Vest

Versjon 1.0



Innhold

Versjon.....	2
Oppsummering.....	3
Introduksjon	3
Målgruppe	3
Påloggingsløsning for Helse Vest	4
Tekniske standarder som er støttet	5
Tilgang / Access	5
Autentisering / Authentication.....	5
Provisjonering / Provisioning.....	5
Ikke støttede eller ikke ønskede standarder.....	5
Transisjonsfase for Autentisering	6
Overordnede krav for applikasjoner og tjenester	7
Veiledning for implementering.....	8
Referanse og kilder	8
Ord og uttrykk.....	10

Versjon

Dette dokumentet er av versjon 1.0. Dokumentet vil bli revidert av Helse Vest IKT sitt IAM team årlig

Oppsummering

Helse Vest ønsker å levere effektive identitets- og tilgangsstyringsfunksjoner, sikkerhet og redusere sikkerhetsrisiko.

Vi ønsker at applikasjonene og tjenester skal bruke OAuth 2.0, OpenID Connect, JWTs og SCIM 2.0 for å unngå leverandørlåsing og balansere sikkerhet, personvern, brukervennlighet og skalerbarhet ved utvikling og implementering av applikasjoner og tjenester.

Introduksjon

Dette dokumentet gir en oversikt på hva Helse Vest IKT har som ønsket standard innenfor autentisering i vår infrastruktur. Dokumentet gir også veiledning samt eksempler på hvordan man kan etablere en mer moderne autentisering i Web, mobil, og Desktop/Windows applikasjoner.

Dokumentet vil ha en årlig revisjon av IAM teamet i Helse Vest IKT.

Målgruppe

Dokumentet er rettet mot personell som arbeider med anbudsprosesser, leverandører og team-/personer som utvikler løsninger som skal etableres i Helse Vest sitt miljø, lokalt eller via offentlige skyløsninger, som f.eks. Azure, AWS etc.

Dokumentet gir en innsikt i hva man må forholde seg til når det gjelder regler for autentisering i Helse Vest. Altså hvilke krav Helse Vest stiller til applikasjonen som skal kjøre eller installeres i vår infrastruktur.

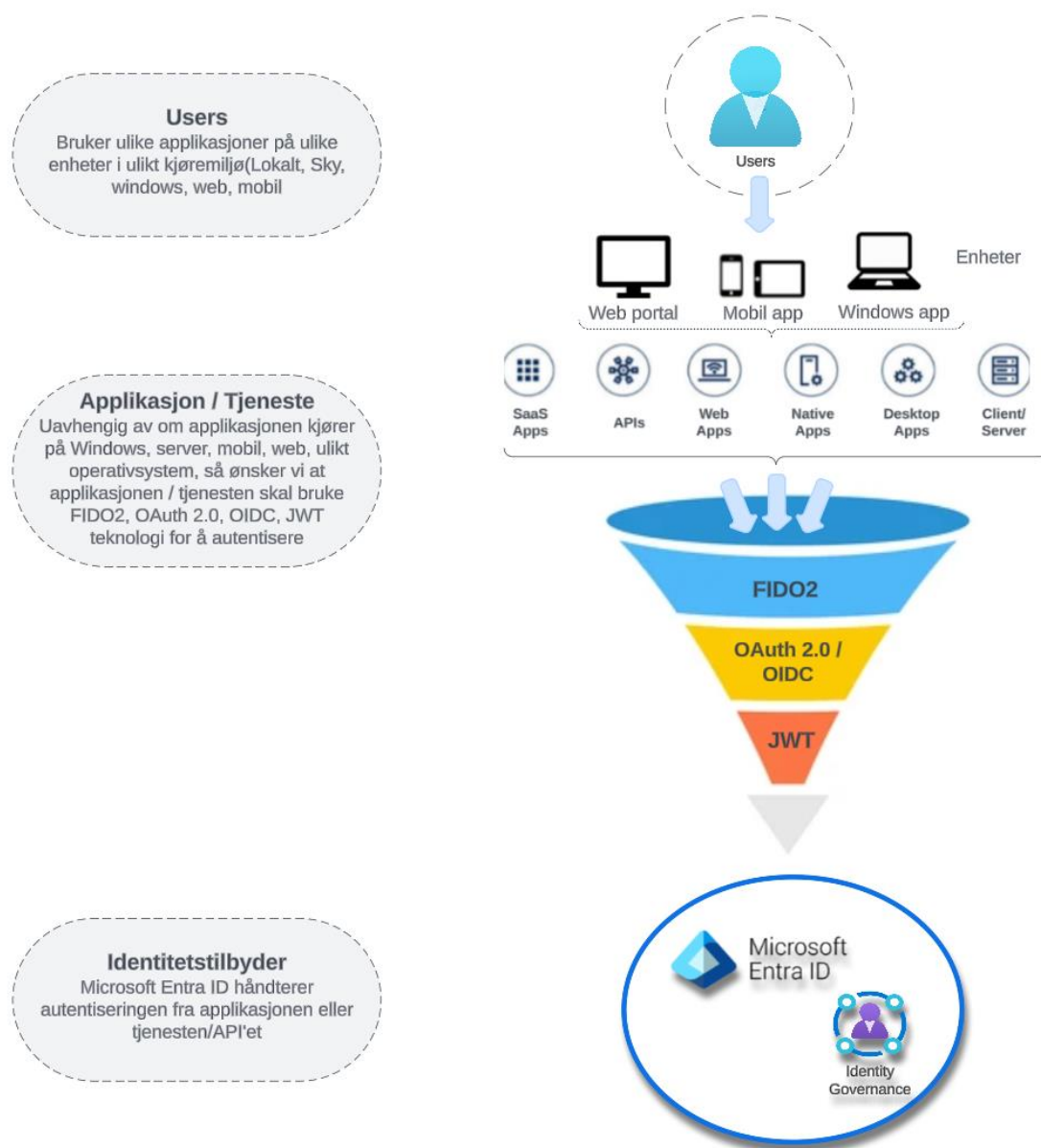
Dokumentet beskriver våre krav samt forslag-/veiledninger på hvordan man kan implementere autentisering i applikasjoner og tjenester.

Påloggingsløsning for Helse Vest

Microsoft Entra ID er Helse Vest sin identitetstilbyder. Applikasjoner, tjenester som skal benyttes i Helse Vest sin infrastruktur skal autentisere mot Microsoft Entra ID. FIDO2 er valgt som Helse Vest IKT sin foretrukne sikkerhetsløsning for pålogging. FIDO2 kommuniserer og autentiserer seg mot Entra ID.

Applikasjoner må kunne autentisere seg mot Helse Vest IdP på våre valgte tekniske standarder, [se kapittel](#). Videre må applikasjonene være beskyttet etter en autentisering med autorisasjon som støtter dette.

Helse Vest IKT sitt IAM team har som mål bilde å bli passordløs. Videre av informasjonssikkerhet hensyn krever vi at det benyttes minimum to faktor-autentisering for tilgang til helse- og personopplysninger og for pålogging på mobile enheter. Fremtidig, vil dette også innebefatte PC eller annen enhet som er tilknyttet Helse Vest sin infrastruktur ved innlogging.



Figur 1 - Forenklet konseptskisse på autentisering

Tekniske standarder som er støttet

Følgende tekniske standarder er støttet og skal brukes i implementering mot Helse Vest sine systemer.

Tilgang / Access

- JSON Web Tokens (JWT)
- OAuth 2.0 / OpenID Connect

Autentisering / Authentication

- FIDO2 – Fast identity Online 2

Provisjonering / Provisioning

- System for cross-domain identity Managment (SCIM 2.0)

Ikke støttede eller ikke ønskede standarder

- Autentisering mot Lokal Active Directory
 - Kerberos
 - NTLM
 - LDAP
- Autentisering mot lokal brukerdatabase i applikasjonen.

INFO! For applikasjoner som har dette implementert, så ønsker Helse Vest IKT at man i fremtiden implementerer våre støttede standarder.

Transisjonsfase for Autentisering

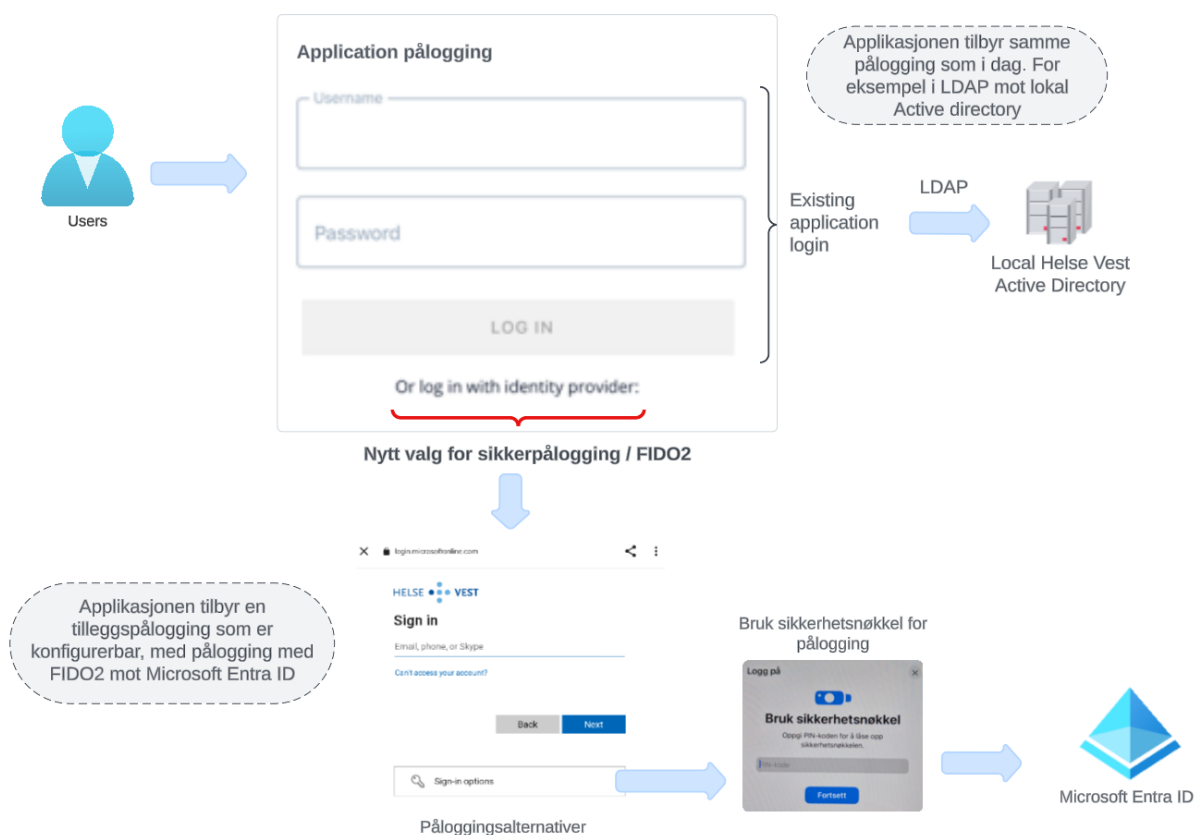
Helse Vest planlegger for en digital transisjon innen autentisering. Det betyr at vi skal standardisere på tekniske standarder innen autentisering, [ref. kapitel](#). I denne transisjonsfasen ønsker vi at applikasjoner skal kunne støtte deres allerede eksisterende autentiseringsløsning i tillegg til vår nye, FIDO2 sikkerhetsnøkkel autentisering. Ergo, vi ønsker ikke å fjerne mulighet for allerede eksisterende autentisering, men at man i denne transisjonen tilbyr den nye i tillegg.

For Helse Vest vil vi ha en situasjon hvor noen vil bli passordløs i transisjonen og noen må bruke passord da det vil være ulik implementeringstakt på personell og system

Scenario:

1. Bruker/Users starter applikasjon
2. Man kan velge mellom å logge inn på vanlig måte som i dag, eller man kan velge en annen identitetstilbyder (i vårt tilfelle Microsoft Entra ID).
3. Vanlig pålogging vil være som før, mens for ny løsning vil man måtte velge konto som Helse Vest benytter for pålogging mot Entra, samt velge sikkerhetsnøkkel

***INFO!** Figur må ikke bli behandlet som et krav, men som et eksempel på autentisering med gammel og ny påloggingsmetode*



Figur 2 Eksempel på autentisering i en applikasjon med FIDO2 og gjeldene innlogging

Overordnede krav for applikasjoner og tjenester

Krav	Forklaring
Nye applikasjoner må støtte autentisering mot <u>Microsoft Entra ID</u> . Kravet gjelder uavhengig av hvilken plattform applikasjonen kjører på.	Microsoft Entra ID er Helse Vest sin identitetsleverandør og autentiseringstjeneste. Applikasjonen må være registrert i Entra ID og kan bruke Microsoft Authentication Library (MSAL) eller annet Microsoft supportert bibliotek, som certified OpenID Connect implementation for håndtering av autentisering.
Nye applikasjoner må støtte protokollen <u>OpenID Connect (OIDC)</u> .	OpenID Connect er en standard som ivaretar en sikker og fleksibel identitetshåndtering på tvers av applikasjonene.
Nye applikasjoner må støtte autentisering gjennom <u>FIDO2</u> .	FIDO2 er en Phishing-resistent MFA autentisering basert på kryptografiske nøkler og er en standard som gir en sikker, brukervennlig og plattformuavhengig autentisering.
Alle applikasjoner skal tilrettelegge for <u>SSO</u> fra enhetsinnlogging.	Applikasjoner som er konfigurert med OpenID Connect, vil relativt enkelt kunne implementere SSO fra enhetsinnlogging. Applikasjoner som er installert på enheten kan tilrettelegge for SSO gjennom Entra Connect.
Eksisterende applikasjoner som autentiserer med brukernavn og passord mot on-prem AD, bør også støtte FIDO2.	On-prem AD fungerer nå som en sekundær autentiseringstjeneste. Entra ID støtter eldre autentiseringsteknologier slik at det ansatte kan logge inn med FIDO2, men da må applikasjonen tillate både passord/brukernavn og FIDO2 fram til alle har fått FIDO2. Dette forutsetter at passordautentiseringen går mot on-prem AD.
Ingen applikasjoner skal benytte proprietære autentiseringsløsninger.	Helse Vest ønsker en enhetlig påloggingsprosess på tvers av applikasjoner og enheter for å forbedre brukervennligheten og redusere kompleksiteten.

Veiledning for implementering

Felles for alle tjenester som skal autentisere seg mot Microsoft Entra ID i Helse Vest er at de først må få registrert tilgang til dette ved å registrere Applikasjonen. Dette gjøres med en henvendelse mot Team Skytjenester. Regler og prosess for dette eies av denne gruppen. De som implementerer løsningen i Helse Vest må derfor kontakte denne gruppen for å få gjort en slik registrering av applikasjonen.




Helse Vest har utarbeidet kode eksempler i .NET som viser hvordan man kan få til en mer moderne autentisering i henholdsvis:

- Web applikasjoner
- Mobil applikasjoner
- Desktop / Windows applikasjoner

Kodeeksemplene er publisert og tilgjengelig på Helse Vest sin GitHub her:

[GitHub - HelseVestIKT/HVI-UTV-EntraEksempler: Eksempler for implementasjon av AD/Entra-pålogging for Web, Flutter, og Desktop.](#)

Vi har delt eksemplene opp i 3, som reflekterer ulike applikasjonstyper.

 angular	angular SPA eksempel
 desktop/EntraDemo	Desktop eksempel
 flutter/hviktentra	angular SPA eksempel

Vi minner på om at dette er eksempler og andre verktøy, og komponenter kan brukes for å gjøre det samme.

VIKTIG ! Hver enkelt leverandør/team, eller utvikler har ansvar for sikkerheten i applikasjonen. Eksemplene som det henvises til viser ikke hvilket sikkerhetsnivå som er gjeldende. Applikasjonen må i tillegg til god autentisering ha god tilgangskontroll og autorisasjonskontroll på applikasjonen.

Referanse og kilder

Følgende linker kan være til hjelp i implementering av mer moderne pålogging.

- <https://learn.microsoft.com/en-us/entra/identity-platform/support-fido2-authentication>
- <https://learn.microsoft.com/en-us/entra/identity/authentication/howto-authentication-passwordless-deployment>
- <https://learn.microsoft.com/en-us/entra/identity/authentication/howto-authentication-passwordless-faqs>
- <https://learn.microsoft.com/en-us/entra/identity/authentication/howto-authentication-passwordless-security-key>

- **OAuth 2.0:** Designet for å delegere tilgang til applikasjoner og tjenester, samt å beskytte API-er. Det gjør det mulig for alle typer applikasjoner (dvs. OAuth 2.0-klient), som for eksempel en applikasjon, en JavaScript-basert applikasjon som kjører i en nettleser, en intern eller ekstern tjeneste, eller en webserverapplikasjon å bevise at de har autorisasjon til å få tilgang til tjenester. Det kan være en bruker til stede, eller klienten kan handle på egne vegne. OAuth 2.0 er en identitetsprotokoll, og den danner grunnlaget for mange andre moderne identitetsprotokoller.
- **OpenID Connect:** SAML 2.0 har lenge vært protokollen som foretrekkes for autentisering på tvers av domener; imidlertid har implementeringer manglet støtte for enkel etablering av tillit på tvers av systemer og støtte for mobile og API-brukstilfeller. SAML bruker også komplekse XML-baserte påstander som kan være utfordrende å analysere og validere. OIDC bygger på OAuth 2.0, men de to løser ikke de samme bruksområdene. Sammenlignet med OAuth 2.0, som beskrevet ovenfor, muliggjør OIDC autentisering på tvers av domener, og det er den moderne ekvivalenten til SAML 2.0, som forbedrer SAMLs mangler.
- **SCIM 2.0:** OIDC og OAuth 2.0 er tilgangskontrollmekanismer for applikasjoner. SCIM 2.0 derimot muliggjør håndteringen av automatisert identitetslivssyklusstyring. SCIM 2.0-standard er utviklet for å hjelpe med å ta i bruk, oppdatere og avvile identiteter i målrettede applikasjoner som er distribuert lokalt og i skyen.
- **JWT:** En JavaScript Object Notation (JSON) Web Token er nå den faktiske legitimasjonen som brukes ved tilgang til API-er og tjenester. Applikasjoner skaffer JWT-er med hjelp av en OAuth 2.0-flyt, og API-er og tjenester validerer og autoriserer tilgang med hjelp av JWT-en. JWT-er er signerte og potensielt krypterte JSON-dokumenter som inkluderer påstander som kan brukes som grunnlag for autorisasjonsbeslutninger.
- **SaaS apps:** Programvare som en tjeneste (SaaS) lar brukere koble til og bruke skybaserte apper over internett.
- **APIs:** API står for Application Programming Interface, som er en samling definisjoner og protokoller for å bygge og integrere applikasjonsprogramvare
- **Single Page Apps:** En enkeltsideapplikasjon er en nettside eller webapplikasjon som dynamisk omskriver en gjeldende nettside med nye data fra webserveren, i stedet for standardmetoden der en nettleser laster inn hele nye sider
- **Web Apps:** er et applikasjonsprogram som er lagret på en ekstern server og levert over internett gjennom et nettlesergrensesnitt.
- **Native Apps:** Begrepet native app antyder en app du kan laste ned og installere på enheten din. En native mobilapp er utviklet spesielt for en mobil enhet
- **Desktop Apps:** Skrivebordsapplikasjoner er programvareprogrammer som kjøres lokalt på datamaskinenheter. De er ikke tilgjengelige fra en nettleser, som webbaserte apper, og krever distribusjon på en personlig datamaskin eller bærbar PC.

- **Client / Server:** Client-server denotes a relationship between cooperating programs in an application, composed of clients initiating requests for services and servers providing that function or service.