HIS 80907:2009

Helsedirektoratet

Referansearkitektur for web services sikkerhet i helse- og sosialsektoren

Publikasjonens tittel:	Referansearkitektur for web services sikkerhet i helse- og sosialsektoren
Teknisk standard nr.:	HIS 80907:2009
Utgitt:	04/2009
Bestillingsnummer:	
Utgitt av:	Helsedirektoratet
Kontakt:	Seksjon standardisering
Postadresse:	Pb. 7000 St Olavs plass, 0130 Oslo
Besøksadresse:	Universitetsgata 2, Oslo
	Tlf.: 810 20 050 Faks: 24 16 30 01 www.helsedirektoratet.no

1 Innholdsfortegnelse

Revisjonshistorikk

Dato	Kommentar	Ansvarlig
02.04.2009	Til publisering	AV

2 Innledning

Denne rapporten beskriver en referansearkitektur for sikre web services i helsesektoren. Rapporten tar utgangspunkt i samhandlingsarkitekturen for web services i helsesektoren samt forslag til profil for web services i helsesektoren, og beskriver aktuelle organisatoriske og tekniske prosesser og tiltak som kan bidra til å bygge sikre tjenester basert på web services teknologi som ivaretar krav til informasjonssikkerhet i helsesektoren.

Målgruppe for referansearkitekturen er aktører som spesifiserer og utvikler IT-løsninger i helse- og sosialsektoren, system- og virksomhetsarkitekter i helsevirksomheter og leverandører av løsninger.

Referansearkitekturen forsøker å samle "best practices" og tiltak innenfor sikkerhet i en tjenesteorientert arkitektur med vekt på realisering av tjenester i en helse- og sosialsektor som behandler store mengder sensitive helseopplysninger med tilhørende krav til sikkerhet. Sikkerhetsløsninger for en tjenesteorientert arkitektur er et område i løpende utvikling, og det er derfor ennå ikke mulig å peke på én endelig, "korrekt" løsning for å implementere sikkerhetstjenester. Arkitekturen er derfor ment som en oversikt og veiledning for å velge tiltak, og er ikke tenkt som et komplett rammeverk som skal implementeres i en omgang, men skal kunne utvikles gradvis.

Sikkerhetsarkitekturen slik den skisseres her omfatter viktige og relevante deler av en "Identity and Access Management" (IAM) strategi, særlig knyttet til identitetsdrevet kontroll, tilgangskontroll til applikasjoner, web services og mellomvare, Single Sign-on, autorisasjon og "claims"/påstandsbasert tilgangsstyring.

2.1 Utfordringer

Etablering av sikre samhandlende webservices på tvers av virksomheter i helse- og sosial sektoren medfører store sikkerhetsutfordringer:

- Web services overføres ofte over https – som pga. kryptering hindrer effektiv filtrering av trafikk på nettverksnivå.
- Anvendelse av https gir kun punkt-til-punkt og ikke ende-til-ende konfidensialitet (https krypterer kun kommunikasjonskanalen, som ofte termineres før informasjonen er i endesystemet hvor den skal anvendes)
- Anvendelse av webservices i og på tvers av virksomheter stiller store krav til autorisasjonsmodellen som avgjør hvem som får tilgang til funksjoner og informasjon i de bakenforliggende systemene
- Distribuerte og til dels mobile tjenester og brukere stiller utfordringer både ift. tilgjengeliget, pålitelighet og ift. håndheving av sikkerhetspolicy
- Når eksisterende applikasjoner "pakkes inn" som web services oppstår utfordringer med overføring av identitet, sikkerhetsmekanismer og generelt en økt angrepsflate for sikkerhetstrusler
- Infrastrukturen som web services de facto bygger på, som XML, SOAP, web service applikasjonsservere og webtjenere for portaltjenester reiser egne sikkerhetsutfordringer og øker angrepsflaten (f.eks. XDoS, svakheter i xml parsere osv.)

- Standardene som benyttes for sikring av web services, særlig ift. komplekse tjenester (kryssing av sikkerhetsdomener, samhandling/kjeding av tjenester) er umodne og/eller under utvikling, og det er i liten grad etablerte løsninger og rammeverk tilgjengelig utover ”proof-of-concepts”.

For at web services baserte tjenester skal kunne tas i bruk på en sikker måte er det behov for en felles referansearkitektur som sikrer tillit på tvers av virksomhetsgrenser, tjenesteleverandører og tjenestekonsumenter.

Denne referansearkitekturen søker å beskrive tiltak som kan implementeres for å sikre at web service baserte tjenester kan tilbys på tvers av virksomheter og sikkerhetsdomener på en måte som sikrer gjeldende krav til sikkerhet som er beskrevet nedenfor.

2.2 Sikkerhetsbehov

Følgende sikkerhetsbehov (som baserer seg på kravene i bl.a. Bransjenorm for informasjonssikkerhet [Bransjenorm]) er lagt til grunn for referansearkitekturen:

- Autentisering: Arkitekturen skal sikre at kun autoriserte brukere har tilgang til tjenestene
- Autorisasjon: Arkitekturen skal sikre at autoriserte brukere har tilgang til nødvendig funksjonalitet og informasjon i lys av tjenestelige behov og avtaler mellom samarbeidende helsevirksomheter.
- Konfidensialitet: Tjenestene som tilbys skal kunne kommunisere sensitive helseopplysninger, og må derfor sikres slik at kun autoriserte brukere har tilgang til informasjonen. Ved behov må informasjon kunne krypteres ende-til-ende mellom de kommuniserende systemene.
- Integritet: Tjenestene skal sikres slik at informasjon ikke skal kunne endres av uautoriserte, verken via tjenesten eller i kommunikasjon mellom tjeneste og konsument.
- Tilgjengelighet: Tiltak for tilgjengelighet er ikke behandlet direkte av sikkerhetsarkitekturen i denne versjonen.
- Sporbarhet: Handlinger som utføres vha. tjenesten skal logges og kunne henføres til riktig bruker. Sporbarhet skal kunne sikres også igjennom komplekse samhandlingsprosesser, og logging og bevaring av identitetsinformasjon for konsumenten må ivaretas.

Profil for web services i helse- og sosialsektoren [WS-Profil] definerer fire sikkerhetsnivå for autentisering, som også benyttes i denne referansearkitekturen:

- Usikrede tjenester: Forholder seg ikke til sikkerhetsmekanismer og er per definisjon åpne for en hvilken som helst konsument som har tilgang til ”nettverket” tjenesten er publisert i
- Brukernavn/passord: enkel autentisering – følger WS-I Basic Security profile 1.0 med username token, evt. SAML 2.0 token basert på autentisering mot en Single Sign-On tjeneste basert på brukernavn/passord.
- Virksomhetsnivå: – følger WS-I Basic Security profile 1.0 med signatur basert på virksomhetssertifikat, evt SAML 2.0 token basert på autentisering mot en Single Sign-On tjeneste signert med et virksomhetssertifikat.

- Personlig nivå: - følger WS-I Basic Security profile 1.0 med signatur basert på personlig sertifikat på nivå Person-Høyt ihht. ”Kravspesifikasjon for PKI i offentlig sektor”. Evt. SAML 2.0 token basert på autentisering mot en Single Sign-On tjeneste med sertifikat på nivå Person-Høyt.

Valg av autentiseringsnivå avhenger av tjenesten, men for tjenester som gir tilgang til helseopplysninger hvor konsumenten befinner seg i et annet sikkerhetsdomene enn tjenesten må sikkerhet på personlig nivå benyttes.

3 Use case

Følgende ”use case” er lagt til grunn for referansearkitekturen og tiltakene som er foreslått.

3.1 Sikring av tjenester internt i virksomheten

En pasient ankommer akuttmottaket. Mottakende lege er pålogget i sitt EPJ. Informasjon om pasienten fylles ut og oppdateres i EPJ. Når informasjonen er utfylt kaller EPJ-systemet helseforetakets sentrale pasientregister og oppdaterer dette med informasjon. Informasjon om brukeren og hans rolle og rettigheter i organisasjonen videreformidles fra det interne sikkerhetssystemet, og logges i pasientregisteret sammen med de oppdaterte opplysningene.

3.2 Samhandling på tvers av sikkerhetsdomener

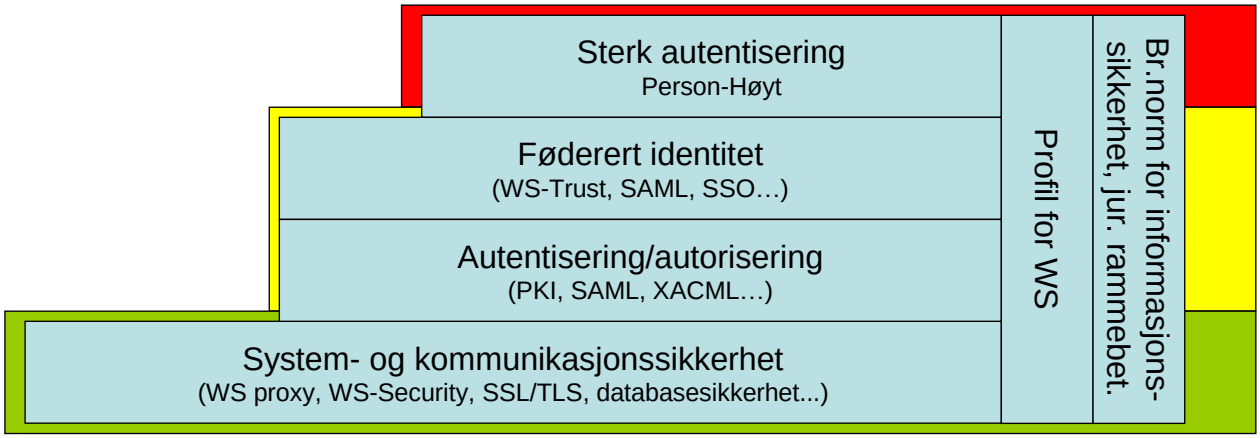
En pasient har ankommet akuttmottaket. Det er mistanke om at pasienten benytter en spesiell type medikament som har betydning for den videre behandlingen. Legen logger seg på sitt lokale EPJ. Derifra gjør han et kall mot en sentral kjernejournal med legemiddelinformasjon. Siden det er nødvendig med tilgang til sensitive helseopplysninger, må legen ha en sterk autentiseringsmekanisme. Legen er pålogget lokalt med en sterk autentiseringsmekansime, og hans identitet med tilhørende autorisasjonspåstander kan derfor videreformidles til den sentrale kjernejournalen. Virksomheten bekrefter videre at legen har et behandlingsforhold til den aktuelle pasienten.

3.3 Samhandling mellom primær- og spesialisthelsetjenste

En pasient har ankommet sin primærlege. Pasienten benytter en spesiell type medikament som kan ha betydning for videre behandlingen hos annet helsepersonell, og primærlegen vil oppdatere den nasjonale kjernejournalen med informasjon om dette.

Legen logger seg på sitt lokale EPJ. Derifra gjør han et kall mot en sentral kjernejournal med legemiddelinformasjon. Kjernejournaltjenesten forutsetter en sterk autentisering, så primærlegen må vha. sitt smartkort logge seg på en sentral sikkerhetstjeneste i helsenettet. Sikkerhetstjenesten gir legen et påloggingsbevis (security token) med informasjon om at legen er autentisert, samt bl.a. informasjon om hpr-nummer, rettigheter fra helsepersonellregisteret osv. Basert på denne informasjonen gis legen tilgang til å oppdatere pasientens kjernejournal.

4 Oversikt - Referansesarkitektur for sikre web services



Referansearkitekturen er oppbygd etter en trappemodell, som gir økende grad av sikkerhet ift. sikkerhetskravene. På det første trinnet (Grønt) er krav til system- og kommunikasjonssikkerhet førende, bl.a. veiledning ift. utvikling og tilgjengeliggjøring av tjenestene, bruk av basisstandarder for kommunikasjonssikkerhet osv. Dette er krav som vil kunne ha relevans også for åpne tjenester uten behov for autentisering.

På det andre trinnet er tjenester med behov for autentisering på person- eller virksomhetsnivå, og trinnet omfatter bl.a. løsninger for føderering av identitet. (Gult)

For tjenester som tilbyr tilgang til helseopplysninger fra eksterne nett er sterk autentisering et krav, og dette utgjør det høyeste trinnet (Rødt).

5 Sikring på transport- og meldingsnivå

WS-I Basic Security profile v. 1.0 [WSI-BSP] legges til grunn for sikring på transport- og meldingsnivå.

For sikring av konfidensialitet bør TLS v. 1.0 eller SSL v. 3.0 benyttes. Dette innebærer punkt-til-punkt sikring av konfidensialitet. I de tilfeller hvor tjenestens oppbygning innebærer at dette er utilstrekkelig, skal XML Encryption iht. [WS-Security] benyttes.

For tjenester som har behov for sikring av autentisering på virksomhetsnivå bør det benyttes XML signatur iht. WS-Security [WS-Security] med virksomhetssertifikater. For samhandling mellom aktører i helsesektoren bør det for slike tjenester benyttes PKI-sertifikater som utstedes av en sertifikatutsteder (CA)/Tiltrodd tredjepart (TTP) som er registrert som utsteder av virksomhetssertifikater hos Post- og teletilsynet, og CA'en må ha en sperretjeneste som er tilgjengelig fra Norsk Helsenett.

For enkelte tjenester vil det kunne stille krav om personlig signatur. Dette bør implementeres i form av XML signatur iht. " XML Signature Syntax and Processing (Second Edition)" [XML-DSIG]. Retningslinjene i [WSI-BSP] bør følges ift. valg av algoritmer for signering og kanonalisering. Alternativt kan personlig signatur implementeres i tråd med [WS-Security] med bruk av personlig sertifikat. Dette forutsetter i så fall en tett integrasjon mellom systemet som genererer SOAP-header og sluttbrukerløsningen for kommunikasjon mot smartkort/nøkkelbærer, noe som kan være problematisk i enkelte sammenhenger. Siden SOAP-header som oftest forkastes etter tjenestekallet, og XML signatur som oftest ønskes benyttet for å bevare signaturen i etterkant, anbefales det første alternativet.

Hvis sikkerhetsbehovet primært er knyttet til autentisering på personnivå av brukeren som aksesserer tjenesten (og ikke langtidslagring av signatur på dokument), anbefales heller bruk av SAML assertion iht. [WSI-BSP] – se kap. 7.2.

6 Autentisering og føderering av identitet

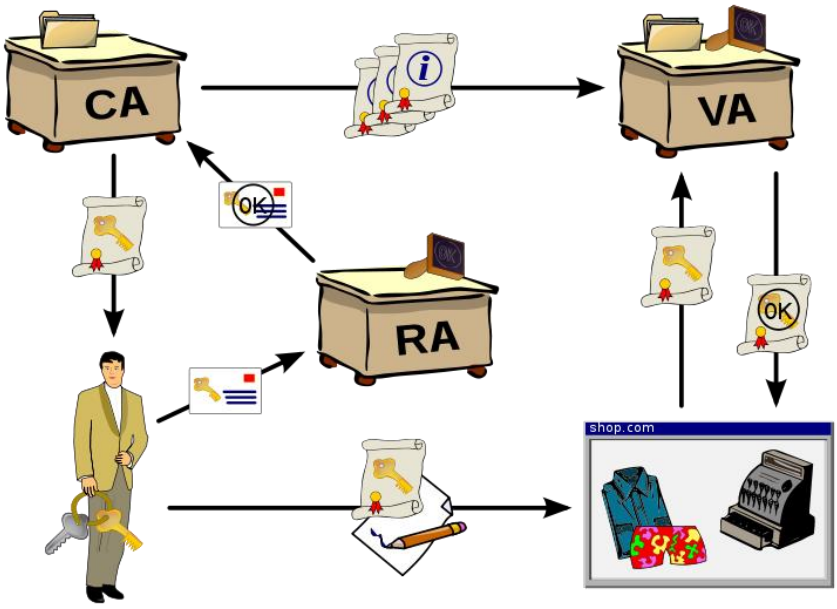
Identitetshåndtering i distribuerte applikasjoner er en stor utfordring:

- ulike applikasjoner håndterer brukeridentiteter på ulike måter, benytter ulike autentiseringsmekanismer og identifikatorer
- identitetene må krysse tillitsgrenser mellom virksomheter
- helsesektoren har ennå ikke standardisert på format for overføring av brukeridentitet og autentiseringsnøkler på tvers av virksomheter

Vi kan skille mellom to hovedlinjer for autentiseringsmekanismer som kan brukes på tvers av virksomheter – PKI m. bruk av felles TTP (som baserer seg på en felles identitetskilde) eller føderering av identitet på tvers av tillitsdomener (som baserer seg på identitetskilder som forvaltes distribuert hos den enkelte virksomhet som inngår i fødereringen). Føderert identitet vil også kunne benyttes som en overbygning over en PKI-basert identitetstjeneste.

6.1 Identitetshåndtering vha. PKI

Kvalifiserte sertifikater som følger kravene i "Kravspesifikasjon for PKI i offentlig sektor" er pt. i bruk av helsepersonell i primærhelsetjenesten for signering av elektroniske sykmeldinger mot NAV, og bruk mot eResept er under veis.



Ved bruk av PKI og sertifikatene som benyttes i helsesektoren er det mulig å bygge felles tjenester hvor brukeren har en sikker identitet autentisert vha. sitt personlige sertifikat utstedt av en tiltrodd tredjepart (TTP).

En PKI-tjeneste fungerer ved at en uavhengig sertifikatutsteder (CA) utsteder et sertifikat som bekrefter brukerens identitet (som registrert av en registreringsautoritet –

RA). Iht. ”Kravspesifikasjon for PKI i offentlig sektor” er denne identiteten brukerens fødselsnummer. Når brukeren skal ha tilgang til en webtjeneste presenterer han sitt sertifikat, og webtjenesten kan verifisere sertifikatet mot en verifikasjonstjeneste (VA) for bl.a. å sjekke at sertifikatet er gyldig. På denne måten vet webtjenesten hvem brukeren er. Webtjenesten har også mulighet til å slå opp i andre tjenester (katalogtjenester o.a.) for å avklare for eksempel brukerens rettigheter og autorisasjoner (er f.eks. brukeren lege, har han forskrivningsrett osv.).

En svakhet ved denne modellen er at den gir en sterk binding mellom tjenesteleverandør, tjenestekonsument og autentiseringen. PKI-modellen støtter også kun identitetsformidling, men kan i liten grad brukes direkte for å gi annen informasjon om brukeren som kan brukes i autorisasjonssammenheng, noe som håndteres bedre av en føderert identitetstjeneste.

6.1.1 Eksempel - Bruk av PKI i eResept

Reseptformidleren som benyttes i eResept-prosjektet er et eksempel på en tjenesten som baserer seg på en sentral PKI-tjenesten for autentisering.

eResept-løsningen (reseptbanken) bruker signering og kryptering iht. WS Security. Løsningen krever at alle meldinger (Message Body) er signert med virksomhetens sertifikat, utstedt av en godkjent TTP. Meldinger til reseptformidleren krypteres tilsvarende med reseptformidlerens offentlige sertifikat.

For meldinger som krever personlig signatur, foregår autentiseringen i foretningslaget og ikke iht. WS security – noe som hovedsakelig kan begrunnes med at eReseptmeldingene har behov for å oppbevare signaturen knyttet til meldingen over tid (persistent sikkerhet), samt at de samme meldingen kan utveksles både over WS-grensesnitt og via ebXML over SMTP. Personlig signatur er implementert ved bruk av XMLDSig i den aktuelle meldingen.

```
<soap>
  <header>
    <signatur_over_body/>
  </header>
  <body>
    <!--Kryptert m. virksomhetssertifikat -->
    <eResept>
      <innhold>Resept</innhold>
      <personlig_signatur/>
    </eResept>
    <!--Kryptert m. virksomhetssertifikat -->
  </body>
</soap>
```

Når en melding med personlig signatur ankommer reseptformidleren, verifiseres sertifikatet som er benyttet mot sertifikatutstederen. I tillegg gir sertifikatutstederen opplysninger om brukerens fødselsnummer som benyttes for å verifisere autorisasjoner i helsepersonellregisteret.

6.2 Identitetshåndtering vha. føderering

Identitetsføderering gjør det mulig for virksomheter å dele tjenester basert på en felles infrastruktur identitetstjenester. Ved føderering kan brukeren autentisere seg som normalt i sin egen virksomhet. Når brukeren skal ha tilgang til en tjeneste ber han sin lokale identitetstjeneste om et bevis på sin identitet og påstander (”assertions”) som han fremmer i forbindelse med forespørselen, som han videreformidler til tjenesten i den andre virksomheten, eller internt. Dette forutsetter at de to virksomhetene er enige om å føderere identiteter, og stoler på hverandres systemer og prosedyrer for

identitetshåndtering og informasjonssikkerhet. Dette kan for eksempel innebære enighet om krav til sikkerhetspolicy og retningslinjer for virksomhetene, krav til sertifisering eller lignende. Tillitsforholdet er uttrykt gjennom en standardisert ”bevis”– et såkalt SAML 2.0 token.

Føderering av identitet kan benyttes uavhengig av hvilken mekanisme som brukes for autentisering (brukernavn/passord, pki, biometri osv.) – så pki-basert autentisering som beskrevet ovenfor kan benyttes som et element i en føderert identitetsarkitektur.

Det eksisterer en rekke bruksområder hvor identitetsfederering kommer til anvendelse, bl.a. knyttet til webportaler og single sign-on (SSO). Sikkerhetsarkitekturen vil i denne utgaven fokusere på maskin-til-maskin kommunikasjon ved kall mot en web service i et annet sikkerhetsdomene/annen virksomhet som har etablert føderering. En tilsvarende prosess vil gjelde ved bruk av føderert autentisering internt i virksomheten, slik at tjenester kan bygges opp på samme måte om de tilbyr tjenesten internt eller eksternt. En annen viktig fordel ved dette er at man får en sikkerhetsarkitektur som bidrar til løse koblinger for autentisering / sikkerhetsnivå og autorisasjon mellom tjenestekonsument, identitetskilder og tjenestetilbyder.

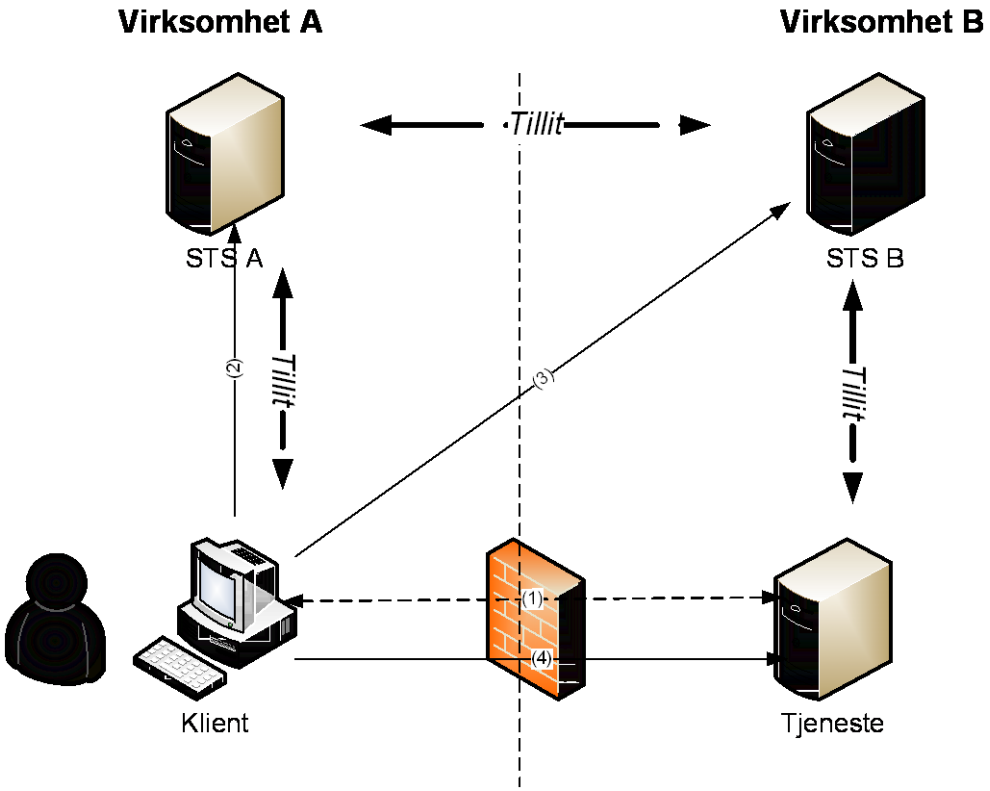
Fødereringsteknologi kan understøtte autentisering i en rekke brukssammenhenger, både internt og ift. ekstern samhandling. Den primære gevinsten internt vil ligge i muligheten for å ha en løs kobling mellom autentiseringtjenesten og foretningstjenesten denne skal understøtte, slik at den enkelte foretningstjeneste ikke trenger å implementere logikk for å håndtere autentisering av brukeren.

For ekstern samhandling innebærer føderering at virksomheten som tilbyr tjenesten ikke må håndtere brukere (utstede brukernavn/passord eller andre autentiseringsmekanismer, håndtere aktivering/terminering av bruker osv.) – men baserer seg på et tillitsforhold til virksomheten brukeren tilhører, og mottar nødvendig informasjon om brukeren fra identitetstjenesten i brukerens egen virksomhet.

En fødereringstjeneste baserer seg på en rekke WS-standarder, særlig SAML og WS-Trust. WS-Trust benyttes for å be om og utstede ”security tokens”, som igjen typisk er beskrevet i SAML.

En fødereringstjeneste består av flere komponenter. En **Security Token Service** (STS) utsteder og konverterer **Tokens**. Tokens er et sett med ”påstander” (”assertions”) om et subjekt (typisk en bruker), dette kan være brukerens navn, e-postadresse, rolle i organisasjonen og hvorvidt brukeren er autentisert.

En **Relying party** er typisk en web service som har behov for autentiseringstjenester, og som baserer seg på tokens mottatt fra en STS. Ved at STS’en håndterer autentiseringsfunksjonen kan tjenesten fokusere på foretningslogikken og overlate autorisasjon og autentisering til sikkerhetstjenesten.



Hovedtrinnene ved bruk av en føderert identitetstjeneste[1] er:

1. Klienten gjør (opsjonelt) oppslag mot tjenesten for å avdekke krav til autentisering (for eksempel via WS-MetadataExchange).
2. Klienten autentiserer seg mot lokal STS og mottar et token.
3. Klienten overfører sitt autentiseringstoken til STS hos virksomhet B, samt informasjon om hvilken tjeneste som forsøkes aksessert, og mottar et token som gir tilgang til tjenesten.
4. Klienten aksesserer tjenesten vha. token mottatt fra STS B, og tjenesten validerer token'et og gir tilgang.

Tjenester basert på føderert identitet bør:

- kunne håndtere autentisering vha. SAML v.2.0 token utstedt av en STS
- STS bør implementere WS-Trust v. 1.3
- Løsningene bør følge relevante Liberty-spesifikasjoner [Liberty].
- For tjenester som tilbyr tilgang til helseopplysninger mot eksterne nett, bør STS kreve autentisering med sertifikat på nivå Person-Høyt iht. [OffPKI] eller tilsvarende sikkerhet.

Innen det enkelte sikkerhetsdomenet må det etableres en STS – en sikkerhetstjeneste med formål å videreformidle sikkerhetsinformasjon om den enkelte bruker. Tjenestene bør etableres innenfor de samme rammer som dagens lokale autentiseringstjenester, for

[1] Det eksisterer ulike "federation patterns" som kan benyttes, dette er et eksempel , såkalt "inter-domain token exchange". Det er også mulig for tjenesten å validere token fra konsumenten direkte mot egen STS. Se f.eks. http://msdn.microsoft.com/en-us/library/cc836393.aspx

noen vil dette innebære at STS etableres pr. helseforetak, mens enkelte vil ha regionale tjenester for dette.

Det må standardiseres et felles sett med sikkerhetspåstander (”claims”) som har relevans for utveksling av informasjon om den enkelte bruker på tvers av virksomheter. Mulige påstander kan være:

- Identifikator for virksomheten (HER-id, enhetsnummer)
- HPR-nummer
- Navn, stilling, e-post, mobilnummer
- Roller i organisasjonen
- Roller ift. en aktuell behandlingssituasjon/pasient (dynamiske og kontekstavhengige)

Endelig avklaring av aktuelle påstander bør utarbeides som ledd i et prosjekt som skal implementere en slik løsning.

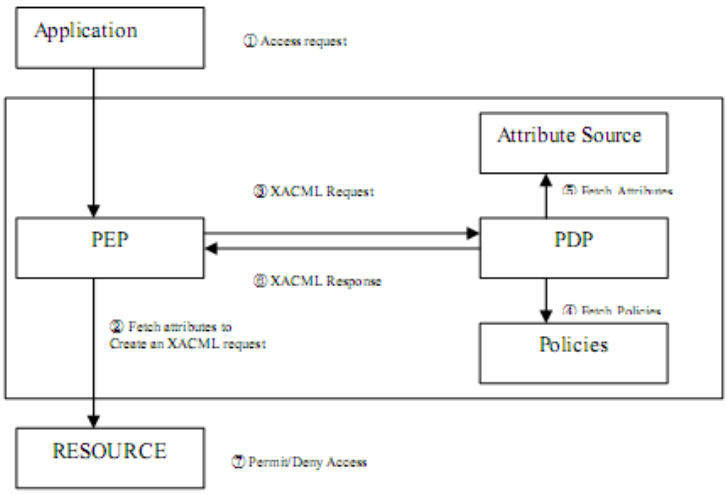
7 Løs kobling mellom tjenestelogikk og autorisasjon

På samme måte som føderert identitet og bruk av sikkerhetstjeneste (STS) frikobler identitet og autentisering fra tjenesten, kan også autorisering frikobles som en tjeneste som kalles for å avgjøre autorisasjonsrettigheter.

XACML kombinert m. SAML 2.0 gjør dette mulig på en standardisert måte.

XACML (Extensible Access Control Markup Language) er et XML-basert språk for tilgangskontroll. XACML beskriver både policy'er for tilgangskontroll og en forespørsel/svar-protokoll for å gjøre forespørsler om tilgang mellom et "policy enforcement point" og et "policy decision point".

I web services sammenheng kan dette benyttes til å knytte tilgangspolicy'er til ressurser og tjenester, for eksempel ved å si at kun en autentisert bruker med en gitt rolle (Andersen, som har logget seg på med et smartkort, er "pasientansvarlig lege"), skal kunne ha tilgang til en gitt tjeneste (spørre om oppdaterte legemiddelopplysninger hos en annen virksomhet).



XACML gjør det mulig å skille mellom tjeneste/foretningslogikken og tilgangsstyringen og styring av autorisasjon. PEP kan fungere som en gateway mellom tjenestekonsument (for eksempel som den av en web services proxy/brannmur) og tjenesten, og gjøre kall mot PDP for å verifisere om en gitt bruker skal gis tilgang til tjenesten.

Policy'er beskrevet i XACML kan beskrives uavhengig av tjenestelogikken og gjøres tilgjengelig for PDP'en, som parser disse for å gi svar på forespørselen om tilgang.

Bruk av XACML vil foregå internt i virksomheten som tilbyr tjenesten, og er ikke synlig eksternt, men foregår mellom autoriseringsfunksjonalitet i tjenesten som kaller policy enforcement-tjenesten. XACML vil derfor ikke være synlig som en del av kontrakten med tjenestekonsumenten, med unntak av eventuelle krav til innhold i SAML 2.0 token som konsumenten må benytte for å få tilgang til tjenesten.

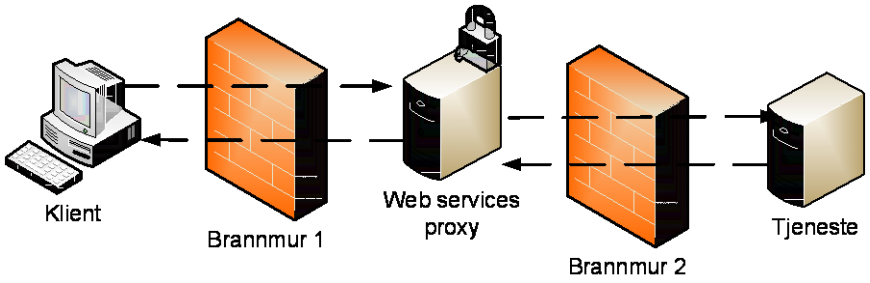
Typisk implementasjonsscenario er:

- Tjenesten mottar et kall fra en konsument. Med kallet følger SAML token med informasjon om konsumenten.
- PEP fanger opp kallet til tjenesten (i form av interception av tjenesten eller kall til PEP-modul før tjenestelogikken)
- PEP videresender konsumentens påstander (fra SAML token) samt informasjon om ressurs, handling og annen relevant informasjon til PDP (for eksempel v. bruk av SAML 2.0 binding mot en STS).
- PDP evaluerer forespørselen om autorisering mot registrerte policy'er og svarer tilbake hvorvidt tilgang skal gis.
- PEP avviser eller videreformidler kallet til tjenesten.

Anbefaling:

Hvis tjenesten har behov for sentralisert styring av tilgang og autorisering, bør en slik løsning baseres på bruk av XACML over SAML, iht. [XACML]

8 Sikring av grensesnitt mot omverdenen – Web service proxy



Når web tjenester skal eksponeres utenfor eget nett, tilbys tjenesten til et potensielt stort omfang av ulike aktører, noe som stiller store krav til tjenesten på områder som sikkerhet, tilgjengelighet, versjonering og endringshåndtering, tjenestekvalitet og teknisk interoperabilitet.

En web service proxy kan dekke en rekke ulike funksjoner, som:

- Frikoble grensesnitt mot omverdenen fra implementasjonen internt i virksomheten
- Transformering av protokoller (f.eks. SOAP over JMS til SOAP over http)
- Sikkerhet og drift/overvåkning

Viktigst i denne sammenheng er funksjonalitet ift. sikkerhet:

- Endestasjon for transportsikkerhet: En WS proxy kan tjene som endestasjon for kryptering på transportlaget (TLS/SLL). Løsningen kan ha spesialhardware for å håndtere kryptering/dekryptering slik at ytelsen i tjenesten ikke begrenses.
- Filtrering av XML innhold, f.eks. validering av innkommende tjenestekall mot XML schema for tjenesten. Proxy'en kan også implementere andre XML-baserte filter som gjør det mulig å sentralisere styringen med sikkerhet for

tjenesten, for eksempel basert på innholdet i tjenestekallet eller andre parametere.

- Filtrering av annet innhold, for eksempel vedlegg til tjenestekallet mot virus og ondsinnet kode.
- Legge på og fjerne sikkerhetsinformasjon: Slik WS-protokollene er oppbygd, er det mulig å legge til og fjerne informasjon i headere, og dette kan benyttes til å frikoble applikasjonslogikken fra sikkerhetslogikken. WS proxy'en kan for eksempel legge på signatur på virksomhetsnivå på utgående trafikk, og validere signatur og evt. dekryptere informasjon i kall mot tjenesten.
- Beskyttelse mot tjenestenekt-angrep: XML-strukturer kan være store og bl.a. åpne for rekursjon og andre egenskaper som gjør at en ondsinnet angriper kan utføre et tjenestenektangrep. En we services proxy kan legge på filter som skjermer tjenesten for slike angrep.

Krav til løsningene:

For tjenester som skal eksponeres mot eksterne nett og andre sikkerhetsdomener, bør tjenesten sikres vha. web services proxy m. bl.a. xml brannmurfunksjonalitet.

9 Sikkerhet på tjeneste/applikasjonsserver

Mye av sikkerhetsfunksjonaliteten som beskrives i denne arkitekturen er løskoblet fra tjenestene de skal beskytte. Sikkerhet må likevel også håndteres som en del av funksjonaliteten i den enkelte tjenesten, samt som en del av utviklingsprosessen. Informasjon om identitet knyttet til bruken av tjenesten må følge med gjennom hele tjenestekjeden, tilgang til informasjon i databaser må implementeres på en sikker måte, autorisasjon må håndteres slik at riktig bruker får tilgang til riktig funksjonalitet, samt at bruk av tjenesten må logges på en tilfredsstillende måte.

9.1 Autentisering mot lokale databaser/fagsystemer

På tjeneste/applikasjonsserver vil logikk på foretningsnivå håndtere kall mot bakenforliggende systemer og databaser, enten i form av direkte tilgang mot databasene eller via tjenestegrensesnitt som de bakenforliggende systemene eksponerer.

Det er viktig at brukerens identitet ikke "forsvinner" på tjenestenivået, for eksempel ved at tjenesten aksesserer bakenforliggende databaser med en egen databasebruker med vide tilgangsrettigheter, uten at identiteten knyttet til forespørselen lagres sammen med forespørselen. Ideelt sett bør også tilgang til data sikres ved at brukerens identitet i stedet propageres videre bakover slik at sporbarheten opprettholdes, og rettigheter kan styres på samme måte på tvers av lag og tjenester.

9.2 Sikring av datatilgang

Normale tiltak for beskyttelse mot datadrevne angrep bør implementeres mellom web service og database, inkl. beskyttelse mot SQL-injection.

Viktige tiltak er:

- Sikker lagring av database-tilgangsinformasjon (brukernavn/passord osv. som tjenesten benytter for å aksessere databasen) – denne informasjonen må lagres på en betryggende måte iht. det rammeverket som benyttes for å utvikle og tilby tjenesten.
- Riktig valg av identitet for databasetilgang – datatilgang kan foregå via bruker knyttet til tjenesten eller som opprinnelig bruker (impersonation/delegation) – riktig valg avhenger av ulike faktorer knyttet til tjenesteoppbygning, ytelse osv. som må vurderes opp mot hverandre – best practice for aktuelle arkitektur bør følges.
- Sikring av kommunikasjon mellom database og tjeneste – kryptert kommunikasjon og beskyttelse av brukernavn/passord.
- Databasens funksjoner for autentisering og autorisering bør benyttes for å sikre minimal tilgang til databasen, slik at tjenesten ikke kan misbrukes til å få ut unødvendig informasjon – dette kan gjøres gjennom rettigheter og bruk av roller i den aktuelle databasen. Det er også mulig å fjerne direkte databasetilgang til fellesbruker som benyttes i tjenesten ved kun å tillate tilgang gjennom executerettigheter på database prosedyrenivå, dvs. at fellesbruker ikke har direkte skrive eller lese rettigheter i databasen
- Beskyttelse mot SQL-injection – tjenesten må sikre at kall mot databasen ikke kan misbrukes til SQL injection-angrep, for eksempel ved bruk av paramatere eller lagrede prosedyrer.
- Informasjon om den opprinnelige brukeren som er opphav til databasekallet må lagres, enten i form av logging i databasen basert på at databasen kalles med brukerens rettigheter (impersonation) eller på applikasjonsnivå.

9.3 Logging og sporbarhet

Logging og sporbarhet må ivaretas på flere steg igjennom prosessen fra kall hos lokalt system til kall av tjenesten og tilgang til bakenforliggende databaser.

Logging bør omfatte:

- All individuell tilgang til helseopplysninger
- Alle aktiviteter utført av brukere med administrator-tilgang
- Pålogging og forsøk på pålogging
- Logg bruker, tidspunkt, type hendelse, opprinnelsen til hendelsen (for eksempel ip-adresse)

Logging kan med fordel etableres som en sentral tjeneste som kan nås via web services-basert grensesnitt og kalles på ulike trinn i prosessen.

10 Organisatoriske tiltak

10.1 Risikovurderinger

Tjenester som vil eksponere helseopplysninger for eksterne nett skal underlegges risikovurdering for å kartlegge trusler og nødvendige tiltak for å kunne tilby tjenesten på et akseptabelt risikonivå.

Retningslinjer for gjennomføring av risikovurderinger kan finnes i bl.a. faktark til Bransjenorm for informsjonssikkerhet i helsesektoren [Bransjenorm].

Risikovurderingen må også omfatte valg av riktig sikkerhetsnivå for personlig autentisering, jfr. Avsnitt 2.2.

10.1.1 Sikkerhetstesting

For tjenester som er eksponert mot eksterne nett, og som kan gi tilgang til helseopplysninger, bør det gjennomføres jevnlige (f.eks. kvartalsvise) sårbarhetsscanning (bruk av automatiserte verktøy for å avdekke potensielle sikkerhetssvakheter).

Tjenester med mulighet for tilgang til helseopplysninger som potensielt kan nås fra internett bør gjennomgå angrepstesting (penetration test) jevnlig (f.eks. årlig). Testingen bør foregå av kvalifisert personell, fortrinnsvis fra ekstern virksomhet (PCI QSA og PCI ASV kan være aktuelle kvalifikasjoner) [PCI-DSS]. Testingen bør gjøres både på applikasjonsnivå og nettverksnivå (infrastruktur og operativsystemer).

10.2 IDS/IPS

Virksomheter som eksponerer tjenester med helseopplysninger mot eksterne nett bør etablere IDS/IPS-systemer som overvåker og sikrer trafikk inn mot tjenesten og varsler om mulige angrep mot tjenesten.

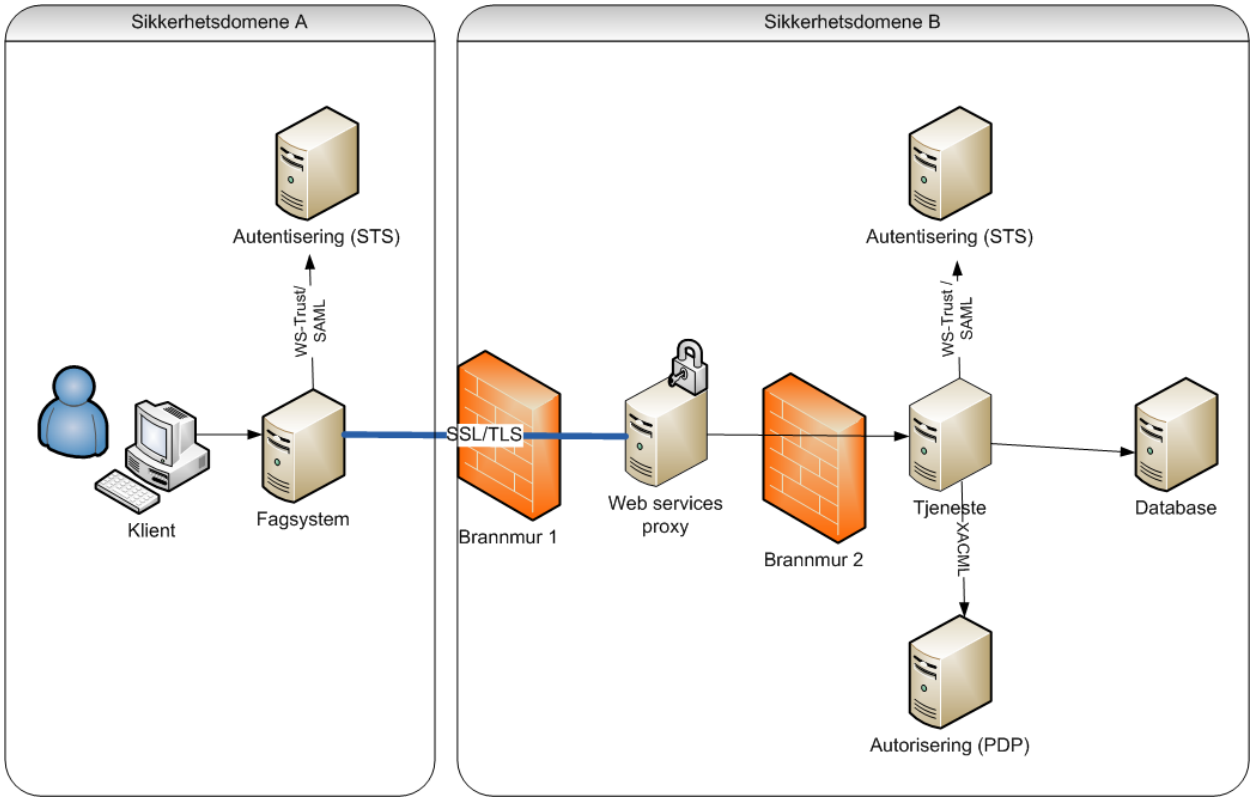
10.3 Rutiner for sikker utvikling

Virksomheter som tilbyr web services som eksponerer helseopplysninger bør etablere rutiner for å sikre at krav til informasjonssikkerhet defineres og følges opp gjennom utviklingsprosessen. Dette kan innebære krav til for eksempel:

- Opplæring i sikkerhet for utviklere
- Felles ”best practices” for sikker utvikling
- Gjennomgang av sikkerhetsarkitektur for løsningen og modellering av trusler
- Bruk av verktøy for testing av sikkerhet i løsningen

I tillegg bør løsningen testes for kjente web services trusler, f.eks basert på kjente lister (for eksempel ”OWASP Top Ten” [OWASP] eller lignende lister over utbredte sikkerhetstrusler)

11 Eksempel - Teknisk implementasjon av sikkerhetssarkitekturen



Det følgende gir *ett* eksempel på en mulig komplett teknisk implementasjon av sikkerhetsarkitekturen slik den er foreslått i dette dokumentet. Som beskrevet tidligere er arkitekturen ment som en oversikt og ”best practice” og ikke som en komplett og endelig spesifikasjon av hvilke elementer og tiltak som skal implementeres. Enkelte tiltak er komplimentære mens andre kan overlappe hverandre for å gi sikkerhet i dybden/flere lag.

1. Bruker logger seg på lokal klient vha. sterk autentiseringsmekanisme.
2. Bruker utfører handling som medfører kall til tjenesten i et annet sikkerhetsdomene.
3. Fagsystemet gjør kall mot tjenesten for å avdekke krav til tilgangen, for eksempel at virksomheten må bekrefte at brukeren har medisinsk behandlingsansvar for en gitt pasient.
4. Fagsystemet gjør kall mot lokal STS vha. WS-Trust, og får utstedt et SAML 2.0 token som bekrefter at brukeren er autentisert med sterk autentisering, samt at brukeren har medisinsk behandlingsansvar til den gitte pasienten som tjenestekallet omhandler.
5. Fagsystemet gjør kall mot tjenesten med SAML token vedlagt. Kallet inneholder helseopplysninger og kallet foregår derfor kryptert over SSL.
6. Kallet fanges opp av virksomhet B’s web services proxy, som terminerer SSL-kanalen. WS proxy validerer struktur og innhold i kallet og godkjenner dette, slik at kallet går videre til tjenesten.

7. Tjenesten mottar kallet med SAML token. SAML token valideres mot lokal STS. Når brukeren er autentisert gjør tjenesten tilsvarende kall mot Autoriseringstjenesten (PDP) – kallene kan sammenkobles, og være implementert som en felles tjeneste.)
8. Hvis brukeren autoriseres utfører tjenesten implementert foretningslogikk inkl. tilgang til databaser. Informasjon om brukeren og rettigheter logges i databasen.

12 Referanser

[WS-Security] Web Services Security: SOAP Message Security 1.0 (WS-Security 2004) - http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-soap-message-security-1.0.pdf

[Bransjenorm] – Norm for informasjonssikkerhet i helsesektoren - http://www.nhn.no/informasjonssikkerhet/bransjenormen-1

[XML-DSig] - XML Signature Syntax and Processing (Second Edition) - www.w3.org/TR/xmldsig-core/

[WSI-BSP] WS-I Basic Security Profile v. 1.0 - http://www.ws-i.org/Profiles/BasicSecurityProfile-1.0.html

[Liberty] - The Liberty Alliance - http://www.project-liberty.org/

[XACML] - OASIS eXtensible Access Control Markup Language (XACML) TC http://www.oasis-open.org/committees/tc_home.php?wg_abbrev=xacml

[OWASP] – Open Web Application Security Project - http://www.owasp.org/

[PCI-DSS] - - https://www.pcisecuritystandards.org/security_standards/pci_dss.shtml

[WS-Profil] – Profil for web services i helse- og sosialsektoren – http://www.kith.no/

[OffPKI] – Kravspesifikasjon for PKI i offentlig sektor - http://www.regjeringen.no/nb/dep/fad/dok/rapporter_planer/Rapporter/2004/Kravspesifikasjon-for-PKI-i-offentlig-se.html?id=106067