

Validering av ebXML-meldinger



HISD 1172:2017

Publikasjonens tittel:

Validering av ebXML-meldinger

Rapportnummer

HISD 1172:2017

Utgitt:

08/2017

Utgitt av:

Direktoratet for e-helse

Kontakt:

postmottak@ehelse.no

Postadresse:

Postboks 6737 St. Olavs plass,
0130 OSLO

Besøksadresse:

Verkstedveien 1, 0277 Oslo
Tlf.: 21 49 50 70

Publikasjonen kan lastes ned på:

www.ehelse.no

Innhold

1	INNLEDNING	3
1.1	Formål.....	3
1.2	Omfang	3
1.2.1	Antakelser og forutsetninger.....	3
1.3	Dokumenthistorie	4
2	REFERANSER	5
3	TERMER OG DEFINISJONER.....	6
4	Bruksområder for valideringsreglene	9
4.1	Generelt om meldingsvalidatorer.....	9
4.2	Meldingsvalidatoren	9
4.2.1	Adressert mottaker modus	10
4.2.2	Respons fra meldingsvalidatoren i «adressert mottaker» modus.....	11
4.2.3	Kobling av respons i «adressert mottaker» modus	13
4.2.4	Observatør modus.....	14
4.2.5	Krav om resultatlikhet.....	14
5	Regelsett for validering av en ebXML-melding.....	16
5.1.1	Avgrensninger med regelsettet.....	16
5.2	ebXML-melding	17
5.3	Leseveiledning	19
5.4	Oversikt over validering	19
5.5	SMTP/MIME validering.....	21
5.5.1	SMTP-validering.....	21
5.5.2	MIME-validering	21
5.6	XML-skjemavalidering	24
5.6.1	Skjemavalidering av SOAP og ebXML	24
5.7	Validering av innhold i SOAP Envelope.....	25
5.7.1	Validering av påkrevd innhold i SOAP Envelope.	25
5.7.2	SOAP-prosessering.....	25
5.8	Validering av MessageHeader	27
5.8.1	Validering av XML-elementer i MessageHeader.....	27
5.8.2	Validering av eb:MessageId (sjekksum og mottakslogging).....	27
5.8.3	Validering av CPAlid	29
5.8.4	Validering av avsender og mottaker elementer i MessageHeader	29

5.8.5	Validering av Service og Action	30
5.9	Oppslag mot Adresseregisteret (AR)	30
5.9.1	Kontroll av HER-id mot AR	31
5.9.2	Kontroll av CPAId mot AR	31
5.9.3	Kontroll av edi-adresser mot AR	32
5.9.4	Kontroll av sertifikat mot AR	32
5.10	Kontroll av signatur	33
5.10.1	Validering av signatur	33
5.11	Identifikasjon av meldingstype	35
5.11.1	Ping og Pong	35
5.11.2	StatusRequest, StatusResponse eller payloadmelding	35
5.11.3	Payload- eller signal melding	36
5.12	Spesifikke valideringsregler for transportkvittering (Acknowledgment)	37
5.13	Spesifikke valideringsregler for transportfeilmelding (ebXML Error Signal)	38
5.14	Spesifikke valideringsregler for ebXML-konvolutt	39
5.14.1	Kontroll av MessageHeader	39
5.14.2	Kontroll av AckRequested	39
5.14.3	Kontroll av eb:Manifest	39
5.15	Spesifikke regler knyttet til modusen «addressert mottaker»	40
5.15.1	Dekryptering av vedlegget	40
5.15.2	Merking av Testcase	41

1 INNLEDNING

Direktoratet for e-helse har sammen med Norsk Helsenett utviklet en meldingsvalidator for kontroll av elektroniske meldinger som sendes over helsenettet og er basert på ebXML-rammeverket [1] (også omtalt som «norsk profil» og «profil» i dette dokumentet). Kontrollen er basert på et gitt regelsett som er utledet av gjeldende krav og presiseringer. Dette dokumentet beskriver valideringsreglene som ligger til grunn for meldingsvalidatoren.

1.1 Formål

Formålet med dokumentet er skape forutsigbarhet og åpenhet for aktører i helse- og omsorgstjenesten for hvilke regler som gjelder for validering av ebXML-meldinger. Dette skal hjelpe aktører i helse- og omsorgstjenesten som implementerer moduler for sending/mottak av ebXML-meldinger (MSH) til korrekt håndtering av ebXML-meldinger.

Valideringsreglene skal redusere tolkningsrommet i ebXML-rammeverket slik at MSH-leverandører vet hvilke regler som gjelder både ved sending og mottak av ebXML-meldinger. Slik kan man oppnå bedre interoperabilitet og færre feil grunnet ulik tolking av ebXML-rammeverket.

Valideringsreglene er sentrale i meldingsvalidatoren som er implementert av Direktoratet for e-helse. Reglene kan også være nyttige i andre sammenhenger, slik som under implementering eller konfigurering av en MSH eller hvis aktører ønsker å implementere sin egen meldingsvalidator for testformål.

Direktoratet for e-helse ønsker å innføre en praksis hvor man automatisk rapporterer feil, slik at de kan rettes ved kilden og forbedre kvaliteten på kommunikasjonen. Samtidig er det viktig at kommunikasjonsflyten ikke stopper og MSH-er må ikke stoppe behandling av meldinger ved feil på valideringsreglene. Meldingsvalidatoren til Direktoratet for e-helse fungerer som ett verktøy hvor alle aktører kan se på sin statistikk over feil og analysere detaljerte feilmeldinger for å se hvilke valideringsregler som har feilet. På denne måten kan hver aktør følge opp sitt bruk av MSH.

Målgruppen for dokumentet er de som implementerer, og konfigurerer en MSH, de som arbeider med forvaltning, overvåking og drift av infrastruktur for kommunikasjon med ebXML-meldinger og de implementerer sin egen meldingsvalidator.

1.2 Omfang

Reglene som spesifiseres i dette dokumentet dekker validering av ebXML-meldinger i henhold til standarden ebMS 2.0 [2] som utdypet i den norske profilen [1], og veiledning for riktig implementasjon og bruk av ebXML [4]. Regler for validering av Payload i ebXML-meldingene er ikke inkludert i dette dokumentet.

Regler knyttet til kommunikasjonsparametere (CPP/CPA) er utenfor omfanget av dette dokumentet og meldingsvalidatoren forholder seg derfor ikke til CPA.

1.2.1 Antakelser og forutsetninger

Ved bruk av valideringsreglene beskrevet i dette dokumentet antas det følgende:

- Alle MSH-er og meldingsvalidatoren er koblet opp mot Adresseregisteret
- de aktuelle partene er registrert under hver sin HER-id i Adresseregisteret (AR) med endepunkter (adresser) og sertifikater
- API mot Adresseregisteret fungerer som det skal, herunder retur av sertifikater uten ytterligere behov for kontroll, f. eks. av tilbakekallinger (eng. revocation)
- Meldingsvalidatoren sjekker endepunkt og sertifikatinformasjon, basert på HER-id, fra Adresseregisteret.
- For alle MSH-er benyttes det SMTP-mottak (EDI-mottak) som kan produsere eventuelle SMTP-feilmeldinger. I dette dokumentet forutsettes det at prosesseringen i SMTP-mottaket har vært vellykket.

1.3 Dokumenthistorie

Dato	Detaljer
30.08.2017	Versjon 1.0 av dokumentet publisert

2 REFERANSER

Valideringsreglene baserer seg på følgende dokumenter:

- [1] Direktoratet for e-helse, «HIS 1037:2011 Rammeverk for elektronisk meldingsutveksling i helsevesenet basert på ebXML», 2011, URL: <http://ehelse.no/his1037-2011>
- [2] OASIS og UN/CEFACT, «ebXML Message Service Specification. Versjon 2.0», 2002, URL: <http://ehelse.no/ebxml-message-service-specification-v20>
- [3] Direktoratet for e-helse, «HIS 1153:2016 Standard for tjenestebasert adressering», 2016, URL: <http://ehelse.no/his1153-1-2016> og <http://ehelse.no/his1153-2-2016>
- [4] Direktoratet for e-helse, «HISD 1171:2017 Veiledning til riktig implementasjon og bruk av ebXML som rammeverk for meldingsutveksling», 2017, URL: <http://ehelse.no/hisd1171-2017>
- [5] World Wide Web Consortium, «SOAP Messages with Attachments», Microsoft, 2000, URL: <http://www.w3.org/TR/2000/NOTE-SOAP-attachments-20001211>
- [6] Network Working Group, «RFC 2633, S/MIME Version 3 Message Specification», 1999, URL: <https://tools.ietf.org/html/rfc2633>
- [7] Kommunal- og moderniseringsdepartementet, «FOR-2013-04-05-959 Forskrift om IT-standarder i offentlig forvaltning», 2013, URL: <https://lovdata.no/dokument/SF/forskrift/2013-04-05-959>
- [8] World Wide Web Consortium, «Joint W3C/IETF XML-Signature Syntax and Processing specification», 2002, URL: <http://www.w3.org/TR/2002/REC-xmlsig-core-20020212/>
- [9] Direktoratet for e-helse, «IS-2175 Profil for CPP/CPA – partnerprofiler og avtaler, Versjon 1.1», 2014, URL: <https://ehelse.no/profil-for-cppcpa-partnerprofiler-og-avtaler>
- [10] Network Working Group, «RFC 2045, Multipurpose Internet Mail Extensions (MIME)», 1996, URL: <https://tools.ietf.org/html/rfc2045>
- [11] World Wide Web Consortium, «W3C-Draft-Simple Object Access Protocol (SOAP) v1.1», W3C Note, 2000, URL: <https://www.w3.org/TR/2000/NOTE-SOAP-20000508/>

3 TERMER OG DEFINISJONER

Definisjoner av viktige begreper, slik de er brukt i denne spesifikasjonen. Begrepene kan ses i sammenheng med kildedokumenter referert i kapittel 2.

Begrep	Definisjon
SOAP with Attachments (SwA)	<p>Overføring av en SOAP-konvolutt sammen med vedlegg.</p> <p><i>Merk1:</i> SOAP with Attachments er definert i et W3C-notat [6].</p> <p><i>Merk2:</i> SOAP-konvolutten og de relaterte vedleggene pakkes i en MIME multipart/related.</p>
ebXML-melding (melding)	<p>MIME-entitet som inneholder en <i>ebXML SOAP Envelope</i>, direkte eller som del av en SwA-innpakning. Eventuell Payload pakkes som et SwA-vedlegg (se kapittel 4.5).</p> <p><i>Merk:</i> SOAP meldingen benytter ebXML extentions til SOAP protokollen derav navnet ebXML-melding. ebMS 2.0 spesifikasjonen [2] omtaler ebXML melding som "message package".</p>
ebXML-konvolutt (SOAP-konvolutt)	<p><i>ebXML SOAP Envelope</i>, en SOAP-konvolutt med ebXML-utvidelser.</p> <p><i>Merk:</i> ebXML-konvolutten inneholder opplysninger om avsender, mottaker, informasjon om forretningsprosessen den er en del av og eventuelle referanse til vedlegget som inneholder Payload.</p>
MSH (Meldingstjener)	<p>Programvarekomponent (eller system) som håndterer meldingsformidling i tråd med ebMS 2.0 spesifikasjonen [2].</p> <p><i>Merk:</i> En meldingstjener kan være sammensatt av flere programvarekomponenter i et større system for meldingsformidling og kommunikasjon med eksterne og interne kommunikasjonsparter.</p>
Fagsystem	<p>Informasjonssystem som mottar og/eller sender fagmeldinger, for eksempel et EPJ-system eller et laboratoriesystem, ved hjelp av en meldingstjener.</p>
ebXML-payload	<p>Forretningsdokument transportert som et vedlegg i en ebXML-melding.</p> <p><i>Merk1:</i> Forretningsdokumentet ligger i en separat MIME del og er i ebXML-meldingen alltid kryptert og signert.</p> <p><i>Merk 2:</i> I dette dokumentet benyttes Payload som synonym for ebXML-payload.</p>

Begrep	Definisjon
Forretningsdokument	Selvstendig dokument som inneholder informasjon beregnet på sluttbrukeren eller sluttbrukerens fagsystem (f.eks. en fagmelding som Henvisning). <i>Merk:</i> Et forretningsdokument er i dette dokumentet alltid enten av typen fagmelding eller applikasjonskvittering.
ebXML signalmelding	Transportkvittering (ebXML Acknowledgment) eller Feilmelding (ebXML Error Signal).
Transportkvittering	ebXML-konvolutt med et eb:Acknowledgment-element i SOAP:Header. <i>Merk1:</i> Sendes som en signalmelding på ebXML-nivå (ikke transportprotokollnivået). <i>Merk2:</i> En transportkvittering forteller avsender av ebXML-meldingen at mottakers meldingstjener har tatt i mot forsendelsen, og at fagmeldingen kan leveres til mottakers fagsystem.
Feilmelding	ebXML-konvolutt med et eb:ErrorList-element i SOAP:Header. <i>Merk1:</i> Sendes som en signalmelding på ebXML-nivå (ikke transportprotokollnivået). <i>Merk2:</i> En feilmelding varsler i hovedsak avsender av ebXML-meldingen at denne er avvist av mottakers meldingstjener. <i>Merk3:</i> ebXML Error Signal kan også benyttes for å varsle avsender av ebXML-meldingen om feil i innhold i ebXML-konvolutt men at meldingen likevel kan behandles av mottakers meldingstjener (Warning).
Signalrespons	Respons som indikerer eventuelle problemer underveis i en meldingsutveksling, i form av en SMTP Bounce, SOAP Fault, ebXML Error Signal eller ebXML Acknowledgment.
Bekreftet mottak (positivt resultat)	Transportkvittering eller en Feilmelding som kun inneholder Warning (ingen Error) tilbake. <i>Merk:</i> Dette forteller avsender av ebXML-meldingen at mottakers meldingstjener har tatt imot forsendelsen, at dekryptering har gått bra, og at vedleggets innhold (payloaden) kan leveres til mottakers fagsystem.
Avvist melding	ebXML-melding med vedlegg som ga en Transportfeilmelding med faktisk feil (Error).

Begrep	Definisjon
Tjenestebasert adressering	Adresseringsmetode hvor det adresseres til og fra kommunikasjonsparter som representerer tjenester, slik det er beskrevet i HIS 1153 [3].
MIME	<i>Multipurpose Internet Mail Extensions</i> er et sett dokumenter som beskriver et meldingsformat for overføring av e-post over Internett som tillater bruk av flere tegnsett, vedlegg og flere typer

4 Bruksområder for valideringsreglene

Denne meldingsvalidatoren benyttes i to ulike modi som nærmere beskrevet i kapittel 4.2. I tillegg kan valideringsreglene tenkes benyttet i følgende områder:

- Som hjelp til å tolke kravdokumentasjon for ebXML-meldinger ved design, implementering og test av MSH-funksjonalitet
- Ved implementering av egen validator hvor det samme sett av valideringsregler som dette dokumentet inneholder skal benyttes.

Under lesing av dette dokumentet må en ta høyde for at den dekker alle modi under ett. Enkelte valideringsregler kan derimot gjelde kun for en av modusene. Dette vil fremkomme av selve regelen.

4.1 Generelt om meldingsvalidatorer

Systemer/løsninger som er i fullt samsvar med ebMS 2.0 [2] omtales som en *Message Service Handler* (MSH). Med de norske tilpasningene omtales de gjerne som *meldingstjenere*.

En meldingsvalidator kan validere ebXML-meldinger som MSH-er har sendt, enten direkte til meldingsvalidatoren i en test eller godkjenningssituasjon eller på ebXML-meldingskopier i produksjon og foreta en kontroll av:

- oppbygningen av ebXML-meldingen
- informasjonen som beskriver selve meldingsutvekslingen, identifikasjon av avsender og mottaker, og eventuell referanse til vedlegget som inneholder Payload

Formålet med valideringen er å registrere alle feilene i meldingen basert på det regelsettet som er dokumentert i kapittel 5.

En reell MSH vil kunne avvise flere tilfeller av feil enn meldingsvalidatoren, da viderebehandling i en MSH kan kreve funksjonalitet ut over ren valideringsfunksjonalitet.

Utgangspunktet for beskrivelsene i denne spesifikasjonen er en ren meldingsvalidator, satt i drift som direkte mottaker (se kapittel 4.2.1).

Riktig bruk av tjenestebasert adressering [3] er en viktig del som valideres og programvaren må, i den grad formålet krever det, kunne gjøre oppslag mot Adresseregisteret.

Dersom man ønsker å implementere en egen meldingsvalidator kan beskrivelsen av Direktoratets meldingsvalidator i kapittel 4.2 benyttes som en modell på hvordan en meldingsvalidator skal virke.

4.2 Meldingsvalidatoren

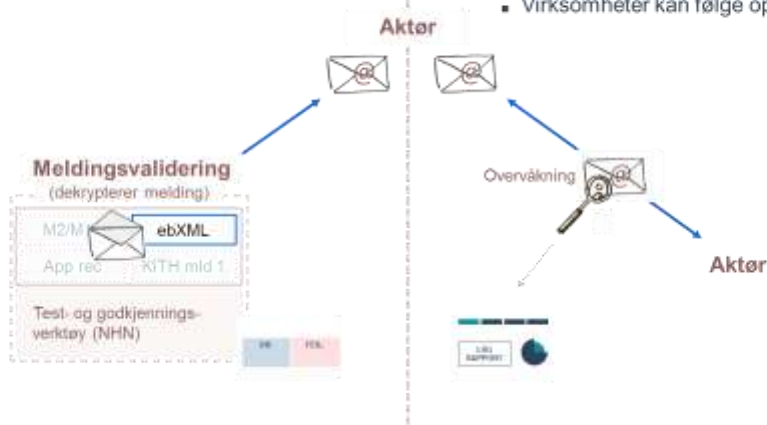
Meldingsvalidatoren kan operere i to ulike modi: enten som adressert mottaker i en test eller godkjenningssituasjon eller som en observatør av ebXML-meldinger i en produksjonssetting.

1. Adressert mottaker

- Fagmelding kan dekrypteres og valideres
- Viser konkret hvor det er avvik fra standardene

2. Observatør

- Fagmeldingen er kryptert og kan ikke valideres
- Viser statistikk om meldingstrafikken på ebXML-nivå
- Virksomheter kan følge opp statistikk for egen virksomhet



I begge modi benytter meldingsvalidatoren de samme valideringsregler som er dokumentert i dette dokument. Alle feil i ebXML-meldingene som valideres blir dokumentert i en feilrapport (logg over oppdagede feil).

Dersom meldingsvalidatoren under validering støter på en feil som ikke stopper muligheten for videre validering, vil meldingsvalidatoren fortsette behandlingen i den hensikt å validere resten av meldingen. Dersom en feil medfører at feilrapporteringen ikke vil bli korrekt vil videre behandling avbrytes, men regnes som gjennomført hvis meldingsvalidatoren var i stand til å finne feil. Meldingsvalidatoren vil sammenholde feilene og føre opp resultatet i en feilrapport.

I en «Adressert mottaker» modus kan meldingsvalidatoren også returnere ebXML Acknowledge eller ebXML Error Signal basert på feilrapporten som blir utarbeidet fra behandlingen av meldingen.

4.2.1 Adressert mottaker modus

I denne modusen vil meldingsvalidatoren operere som en MSH og andre MSH-er må adressere meldinger direkte til meldingsvalidatoren. Denne modusen benyttes kun i test og/eller godkjenningssituasjoner når andre MSH-er skal teste sine implementasjoner/konfigurasjoner.

Som adressert mottaker eksponeres meldingsvalidatoren over et endepunkt, f. eks:

ebxml-validering@nhn.no eller ebms@edi.nhn.no

Meldingsvalidatoren tar imot det som sendes *direkte til installasjonen*, over endepunktet. Meldingsvalidatoren vil kvittere på mottak som en MSH og føre opp eventuelle funn i en rapport.

Som adressert mottaker vil meldingsvalidatoren også dekryptere Payloaden og kontrollere signaturen i ebXML-meldingen.

Meldinger kan ses i sammenheng med andre meldinger (kryss-valideres mot hverandre) når Meldingsvalidatoren brukes som adressert mottaker.

Innkommende responser og meldinger kan fritt sorteres inn under en dialog (trestruktur) i valideringsrapporten hvis de er relatert, og kan berikes med informasjon på tvers av meldingene.

4.2.2 Respons fra meldingsvalidatoren i «adressert mottaker» modus

Responsene fra meldingsvalidatoren er prinsipielt sett tilsvarende en MSH når meldingsvalidatoren benyttes i «adressert mottaker» modus, men vil ikke inneholde detaljert svar. I tillegg forholder den seg til alle signalresponser en mottakende MSH eller SMTP-mottak kan gi.

Hva man kan få i retur (iht. spesifikasjonene) er avhengig av hva man sender som vist i tabellen under. Responsene er gjensidig utelukkende.

Sendt	SMTP-respons	SOAP-respons	ebXML-respons	ebXML-respons
Payload	Bounce	Fault	ErrorList	Acknowledgment
eb:Acknowledgment	Bounce	Fault		
eb:ErrorList	Bounce	Fault		
SOAP Fault	Bounce			

Alle meldinger som sendes, inkludert ebXML-signalene og SOAP Fault, kan potensielt gi en Bounce. Alle unntatt SOAP Fault selv kan gi en Fault.

Meldinger med payload, som ikke resulterer i Bounce eller Fault, skal besvares med én ebXML-respons. Valgt signalmelding skal sendes *som en selvstendig melding (standalone)*, som angitt i den norske profilen [1] av ebMS.

Responsen på en melding med payload, når alt går bra, er Acknowledgment. Feil og advarsler gis med en ErrorList.

Meldingsvalidatoren validerer signalmeldinger (ebXML Acknowledgment eller ebXML Error), men hverken SMTP Bounce eller SOAP Fault skal valideres.

Ved mottak over SMTP plukkes Bounce- og Fault-meldinger ut, før resten av meldingene sendes videre til validering. Dette er gjort for å unngå at ufiltrerte SMTP Bounce-meldinger fremprovoserer en SOAP Fault.

4.2.2.1 ebXML signalmeldinger

ebXML-meldinger med verifiserbar adresseinformasjon kan besvares med ebXML-signaler. Ved feil eller mangler som avsender må varsles om, er responsen fra en normal MSH et ebXML Error Signal (med en eb:ErrorList i SOAP-konvolutten). Man kan formidle både feil og advarsler ved hjelp av dette formatet.

Feil (eb:ErrorList) brukes ved behov for å avvise meldingen helt. Det er et signal om at påkrevd viderebehandling av eb-XML meldingen ikke kan gjennomføres.

Hvis mottatt melding ikke inneholder feil eller mangler avsenderen må varsles om, er responsen en ebXML Acknowledgment (med et eb:Acknowledgment i SOAP-konvolutten).

Bruk av warning

Semantikken av eb:AckRequested tilsier at payloadmeldinger som mangler dette elementet besvares med ErrorList (warning).

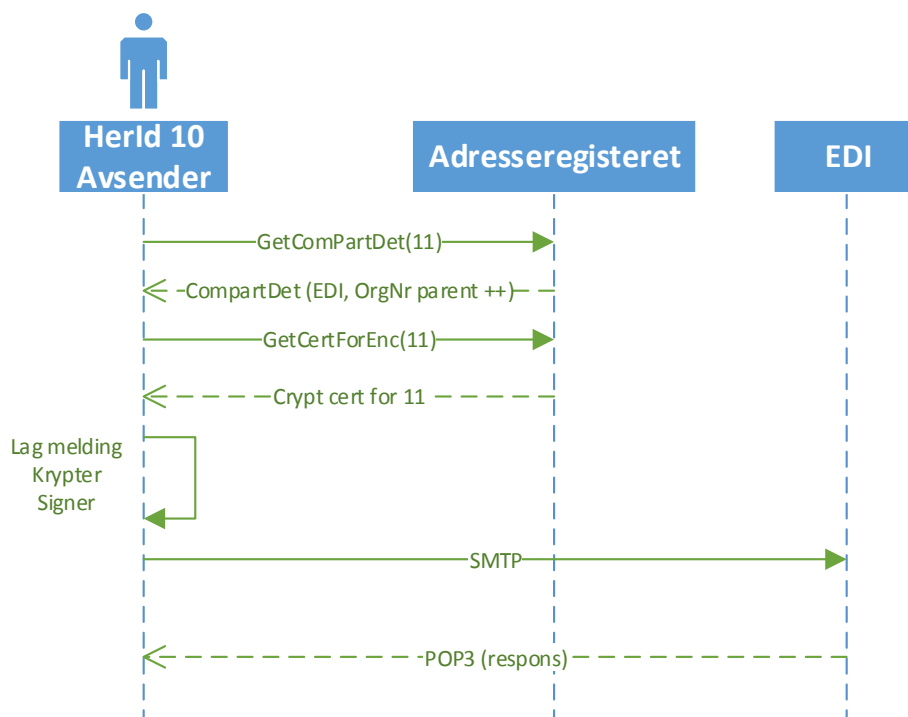
Uten eb:AckRequested skal Acknowledgment ikke brukes. d Samtidig krever den norske profilen [1] at meldinger med payload skal etterspørre bekreftelse, vs. at eb:AckRequested skal benyttes. Innkommende meldinger som mangler eb:AckRequested kan dermed besvares "bekreftende" i form av en advarsel. Slik håndheves den norske profilen samtidig som man respekterer mangelen på dette elementet i den innkommende meldingen.

Validatoren skal returnere Acknowledgment og Error Signal på lik linje med en MSH når den opererer i modus som adressert mottaker.

4.2.2.2 SMTP Bounce

En SMTP Bounce oppstår normalt når en melding ikke når lenger enn til SMTP-serveren kalt "EDI" (Figur 1).

SMTP Bounce produseres av SMTP-serveren foran Meldingsvalidatoren, ikke av den selv. Som en del av reglene i kapittel 5.5.1.1 tar den stilling til om innkommende meldinger er så mangelfulle at de burde ha vært avvist med en Bounce.



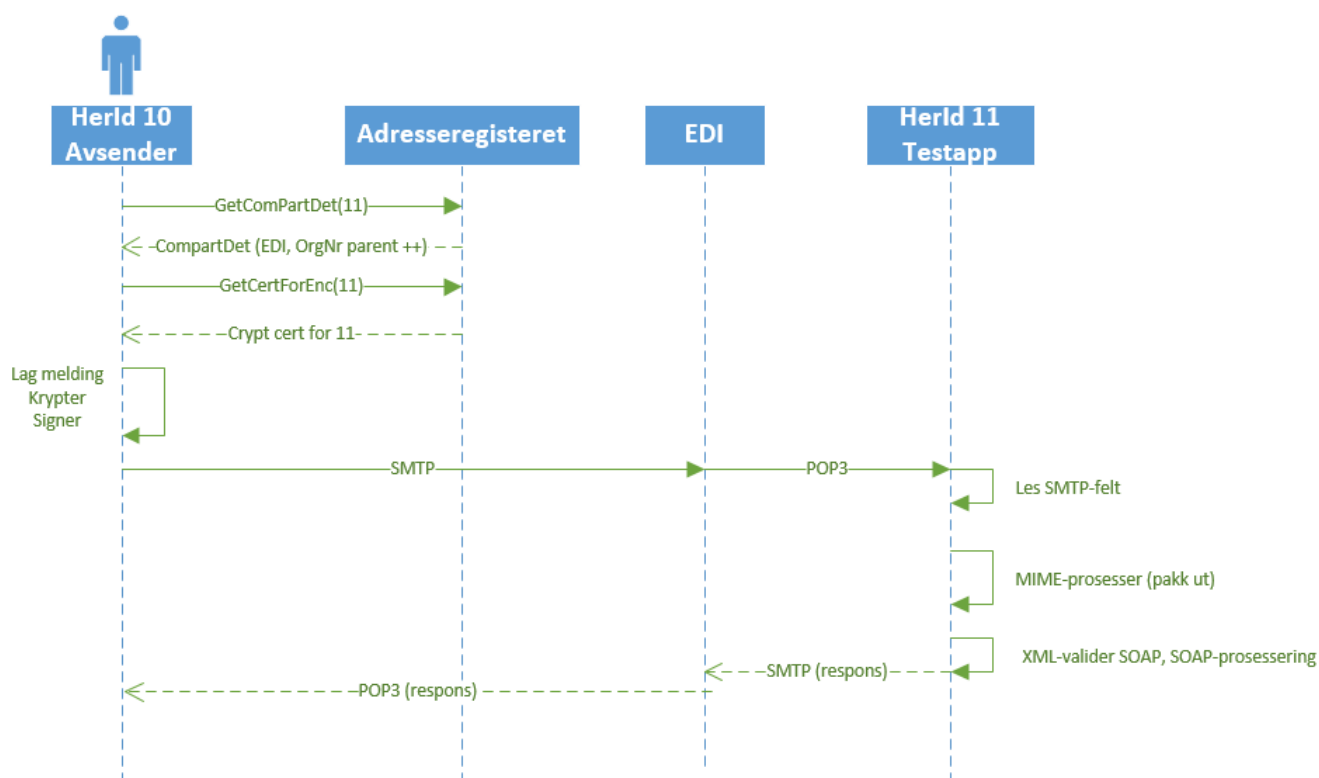
Figur 1: SMTP Bounce

4.2.2.3 SOAP Fault

Hvis SMTP-meldingen når mottakende MSH, men ikke kan tolkes som ebXML, vil man normalt få en SOAP Fault (Figur 2). En SOAP Fault skal returneres til SMTP From: (med mindre SMTP Reply-To: er definert). Alternativt kan den returneres til SMTP Sender:.

Feilsending av helt annen informasjon eller grunnleggende formatfeil forhindrer ikke at meldingen kan plukkes opp fra SMTP-serveren, men den gjenkjennes ikke.

Hvis innholdet kan tolkes som ebXML, men ebXML-meldinger ikke har gjenkjennbar adresseinformasjon (HER-id) så gir det også en SOAP Fault. Hvis gjenkjennbar HER-id er ført opp på begge parter, men oppslag på avsenders HER-id i Adresseregisteret ikke gir en gyldig edi-adresse som man kan returnere et fullverdig ebXML-signal til, så gir det en Fault.



Figur 2: Feil under MIME- eller SOAP-prosessering gir en SOAP Fault som respons

4.2.3 Kobling av respons i «adressert mottaker» modus

4.2.3.1 Kobling på transportnivå

Som avsender vil meldingsvalidatoren notere ned identifikatorer fra mottatte meldinger.

Ut over ebXML-formatets egen identifikator gir sending over SMTP at både Bounce og Fault skal kobles til avsendt meldingsinstans (se kapittel 4.2.2) vha. transportprotokollen og dens meldingsformat (ekstra header-informasjon, i ebXML sitt tilfelle).

Følgende informasjon noteres ned av meldingsvalidator:

- eb:MessageId og eb:ConversationId ved sending av payload (for kobling, alle skal logges)

- SMTP Envelope-Id, tildelt under sende-transaksjonen med serveren, til bruk ved Bounce.
- MAIL Message-ID, som tildelt ved sending, til bruk ved Fault (med mindre en Fault sendes).

4.2.3.2 Kobling på meldingsnivå

Meldingsvalidatoren kobler responsene til opphavsmeldingene, basert på identifikatorer notert under sending.

En SMTP bounce refererer i utgangspunktet ikke direkte til opphavsmeldingen, men til SMTP-transaksjonen opphavsmeldingen ble avlevert i. Meldingsvalidatoren ser etter nednotert SMTP Envelope-Id i Original-Envelope-Id ved Bounce.

Meldingsvalidator sjekker MAIL Message-ID i In-Reply-To ved Fault. En SOAP Fault knyttes via SMTP-meldingen sin MAIL Message-ID som oppgitt i MAIL References og/eller MAIL In-Reply-To. Resending av en og samme ebXML-melding gir opphav til en ny SMTP-melding. Hvis feilen som rapporteres er permanent vil man avslutte resendingsprosessen. Om ebXML-meldingen likevel sendes flere ganger, som ny SMTP-melding, og dermed gir opphav til den samme feilen flere ganger, så er det av liten betydning.

De to signalmeldingene kobles til opphavsmeldingen ved hjelp av dens eb:MessageId, som referert i feltet eb:Acknowledgment/eb:RefToMessageId (hvis signalmeldingen er en ebXML Acknowledgment) eller eb:MessageHeader/eb:MessageData/eb:RefToMessageId (hvis signalmeldingen er en ebXML Error Signal), se ebMS 2.0 [2] for detaljer.

I tillegg til koblingen ved hjelp av MessageId kobles også signalmeldinger iht. norsk profil [1] til opphavsmeldingen ved hjelp av ConversationId.

4.2.4 Observatør modus

I modus som observatør leser meldingsvalidatoren kopi av meldinger som sendes mellom reelle MSHer som opererer i en produksjonssituasjon, dvs. det som sendes over norsk helsenett med reelle data. Meldingsvalidatoren er responsløs i denne modusen og jobber kun på kopier av meldingene. Hensikten med modusen er å hente ut data som underlag for statistikk.

Meldingsvalidatoren kan validere ebXML-meldingen, inkludert signaturen, men kan ikke dekryptere payloaden, siden payloaden er tiltenkt en annen mottaker og kryptert med dennes sertifikat. Pga store volumer av meldinger krever denne driftsmodusen høy kapasitet, og kontrollen avgrenses, generelt sett, til forhold som inngår i én enkelt melding for å holde ytelsen oppe.

All valideringsstatistikk er tilgjengelig via meldingsvalidatoren.

4.2.5 Krav om resultatlikhet

Bruk av valideringsreglene skal alltid gi samme resultat hvis input er lik. Dette vil si at dersom meldingsvalidatoren foretar validering av samme ebXML-melding skal dette gi samme resultat, hvis Adresseregisteret er uforandret.

Hvis forrige beregning av resultatet fortsatt foreligger, så kan det gjenbrukes. Mottaket føres opp i mottaksloggen, MIME-utpakkingen gjentas, og meldingsidentifikatoren plukkes ut, mens resten av beregningen kan kuttes og forrige resultat returneres. Se mer om dette i kapittel 5.8.2.

Unntaket er ved feilaktig gjenbruk av eb:MessageId på tvers av forskjellige meldinger, se 5.8.2.

5 Regelsett for validering av en ebXML-melding

Dette kapitlet beskriver valideringsreglene som benyttes for validering av ebXML-meldinger for de ulike bruksområdene beskrevet i kapittel 4.

Meldinger som valideres kan være enten helt nye meldinger, svarmelding relatert til en tidligere sending (del av en konversasjon), resendinger av disse eller responsmeldinger.

Når valideringsreglene benyttes i en meldingsvalidator skal avvisning resultere i en responsmelding tilbake til avsender, hvis mulig. Man bør unngå feilsituasjoner som resulterer i fravær av respons og heller øke bruk av SOAP Fault. Denne strategien er implementert i meldingsvalidatoren til Direktoratet for e-helse.

En meldingsvalidator vil validere så mange valideringspunkter som mulig. Alle resultater av valideringen lagres i en feilrapport, Meldingsvalidatoren kan også sende ebXML signalmelding som resultat av valideringen som er gjennomført, hvis den opptrer i modus som «adressert mottaker» ref kapittel 4.2.1.

De media-typene en MSH må forvente å motta og forventet innhold av disse, er:

- text/xml med rene ebXML SOAP-meldinger eller SOAP Fault (de uten SwA-innpakning)
- multipart/related med en SOAP- og en attachment-del (ebXML-melding iht. SwA [6])

Hvis en meldingsvalidator sender fra seg noe over SMTP har man i tillegg media-typen:

- multipart/report med bounce-meldinger (ved feil på SMTP-nivå)

5.1.1 Avgrensninger med regelsettet

Programvare bygd etter reglene i dette dokumentet har feil hvis programvaren bryter med regelsettet. Programvaren har mangler hvis den unnlater å implementere deler av regelsettet. Dokumentet gir tilleggskrav ut over de underliggende standardene, ebXML-rammeverket [2] og profilen [1].

Hvis dette dokumentet angir noe som gjør det umulig å oppfylle kravene stilt i standardene, så regnes det som en feil i dette dokumentet, med mindre det er angitt som et bevisst brudd. Hvis man kun velger å realisere en ren meldingsvalidator, ikke en fullverdig MSH, så unndrar man seg en del av kravene som stilles i ebXML-rammeverket [2]. Det er hovedsakelig felt- og feltverdikrav som gjenstår.

Nytteverdien av programvare bygd etter dette dokumentet ligger først og fremst i feilrapporten, og hvilke punkter den dekker, mer enn fullt standardsamsvar. Oppgaven til programvaren er å avdekke feil, spesielt forhold som hindrer meldingsutveksling.

Det er spesifisert inn noen bevisste brudd med de underliggende standardene som et resultat av dette:

- Den ignorerer ikke blokker med mustUnderstand="0", på tvers av SOAP [5], i henhold til ebXML- rammeverket [2].

- Den mottar i rollene actor:toPartyMSH og actor:nextMSH, uavhengig av ebXML-rammeverket [2]-samsvar
- Den sjekker at eb:DuplicateElimination er satt, men utfører ikke slik duplikateliminering selv
- Kvitterer stort sett positivt (med eb:Acknowledgment) i stedet for å avvise meldinger med eb:ErrorList. Feil føres heller opp i feilrapporten.

5.2 ebXML-melding

En ebXML-melding er en MIME-entitet som inneholder en SOAP-konvolutt med ebXML-spesifikke utvidelser (ebXML SOAP Envelope extensions), som definert i ebMS 2.0 [2].

Navnerommene er:

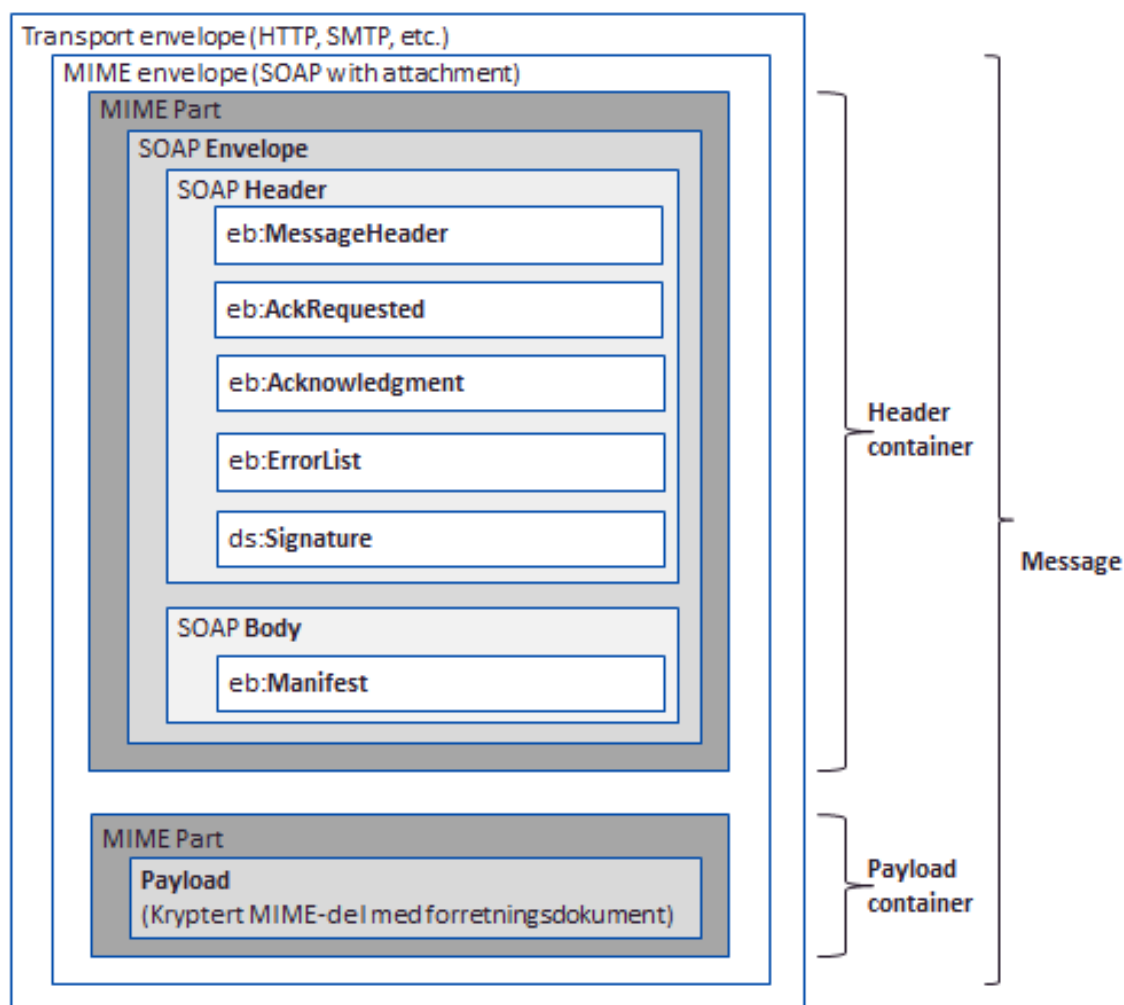
- xmlns:eb="http://www.oasis-open.org/committees/ebxml-msg/schema/msg-header-2_0.xsd"
- xmlns:ds="http://www.w3.org/2000/09/xmldsig#"

Kjennetegnet på en ebXML-melding (2.0) er utvidelsen eb:MessageHeader. Alle meldinger skal ha dette elementet. Meldinger i norsk helsesektor skal i tillegg inneholde en ds:Signature.

Payload transporteres som et vedlegg (attachement) iht. SOAP Messages with Attachments (SwA) [6]. SOAP-konvolutt og vedlegget legges i separate deler som samles i en felles entitet på toppnivå. Ved sending av en fagmelding vil vedlegget være payloaden.

Ved bruk av S/MIME legges payloaden i en MIME-del som krypteres. Dette gir en ny MIME-del hvor innholdet (chifferteksten). Fell tilsvarer det SwA [6] omtaler som vedlegges for klartekst- og kryptotilfellet er samlingen av SOAP-konvolutt og vedleggets MIME-deler i en MIME multipart/related entitet.

Denne entiteten overføres ved hjelp av en transportprotokoll, som SMTP (headere deler plassering).



Figur 3: Korrekt plassering av MIME-delene og XML-elementene (eb: og ds:) i en samlet oversikt.

Figur 3 viser totalstrukturen, med korrekt plassering av de mest aktuelle utvidelsene.

Den norske profilen av ebXML-rammeverket definerer tre meldingsoppsett, omtalt som melding med payload (noen ganger meldingskonvolutt), Transportkvittering og Feilmelding. Et av meldingsoppsettene brukes kun ved overføring av payload. De to resterende oppsettene skal brukes for å signalisere hvordan overføringen gikk. Man velger enten det ene eller det andre av de to, avhengig av resultatet i den konkrete overføringen. Oppsettene dikterer forskjellig innhold i SOAP-konvolutten.

SOAP-konvolutten skal inneholde:

1. én eb:AckRequested og ett eb:Manifest når en melding brukes til å *overføre payload*
2. én eb:Acknowledgment når en melding bekrefter problemfritt mottak (*kvittering*)
3. én eb:ErrorList når en melding videreformidler *feilmeldinger* (errors og warnings) fra mottaket

Som nevnt skal alle de tre variantene *ALLTID* ha én eb:MessageHeader og én ds:Signature.

De to omtalte signalene tilhører ebXML-nivået. De pakkes best som enkel MIME (text/xml). Et signal på forretningsnivå overføres som all annen payload (pakket som en multipart/related).

5.3 Leseveiledning

Valideringsreglene er beskrevet i tabeller, med følgende kolonner:

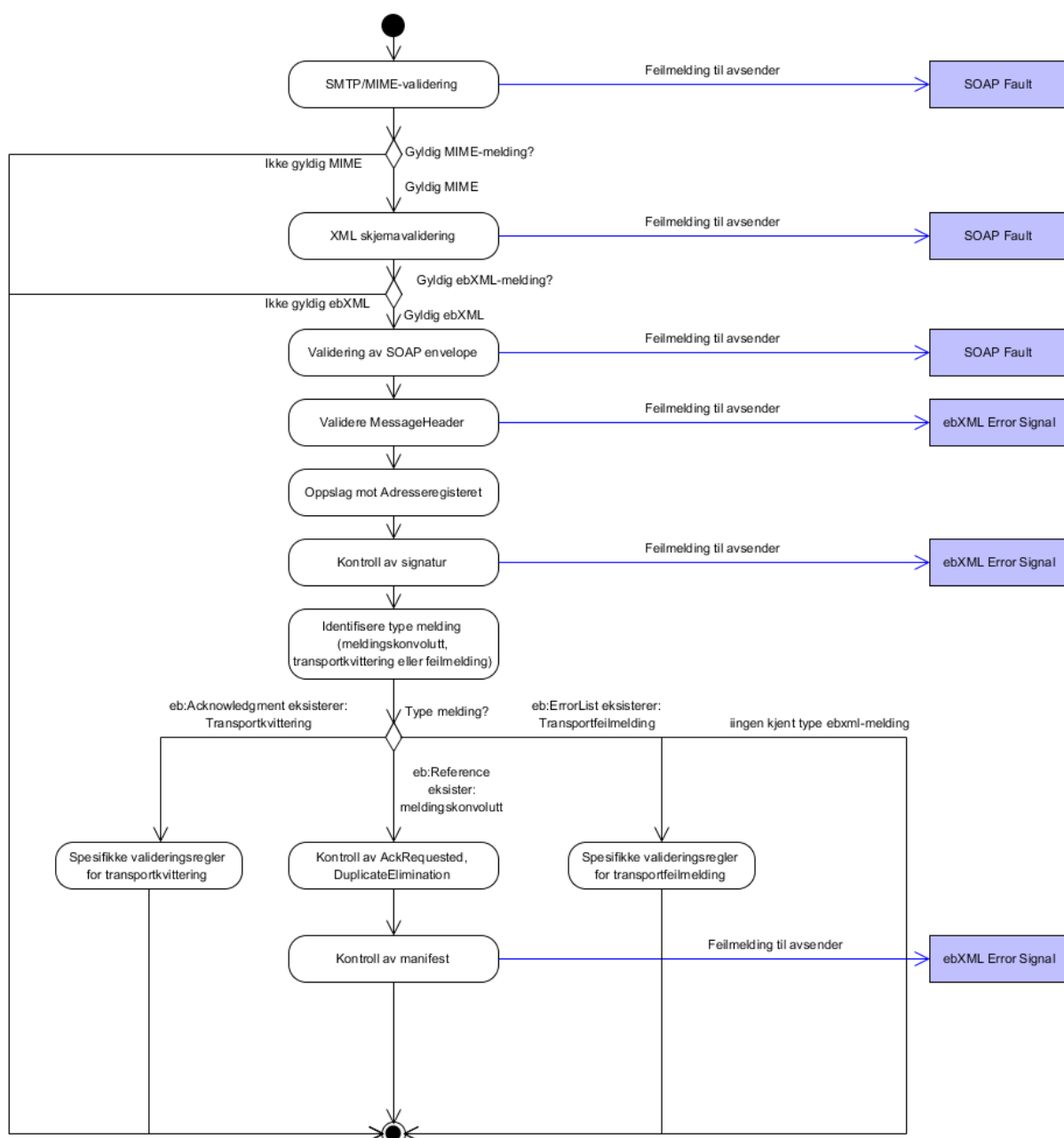
Kolonne	Beskrivelse
Regel Id	Unik identifikator for regelen som tallverdi. Kan benyttes til kryssreferanser mot dette dokumentet under implementasjon.
Beskrivelse	Kort beskrivelse av hva regelen kontrollerer (dens tillagte mening).
Referanse	Referanse til normativt dokument som fastsetter det som kontrolleres.

Et eksempel på de aktuelle header-innslagene i MIME-headeren ved oversendelse over SMTP er:

Message-ID: 1685673502.1308618793518.JavaMail.oas@ketamin
From: reseptformidleren1@edi.nhn.no
To: citylegen@edi.nhn.no
MIME-Version: 1.0
Content-Type: multipart/related; type="text/xml"; boundary="----=_Part_17022"
SOAPAction: "ebXML"
Date: Tue, 21 Jun 2011 03:13:13 +0200 (CEST)

5.4 Oversikt over validering

Figur 4 illustrerer alle grupperingene av valideringsreglene og rekkefølgen i valideringen. De påfølgende kapitlene under detaljerer de enkelte valideringsreglene innenfor hvert trinn.



Figur 4: Valideringsprosessen

Adresseopplysninger valideres tidlig, som grunnlag for responser. En SOAP Fault trenger noen protokollspesifikke headere som sjekkes først. De generelle MIME-feltene kan dermed resultere i en SOAP Fault.

Merk at enkelte valideringsregler kun gjelder for transportkvittering eller feilmelding, mens andre er spesifikke for ordinære utvekslinger med payload.

5.5 SMTP/MIME validering

5.5.1 SMTP-validering

SMTP-validering er validering av protokollspesifikke header-felt. Feil skal kun logges og det skal ikke gis noen respons.

Et HTTP-mottak kan returnere feilkoder basert på egen validering. Her befinner en seg bak en SMTP-server. Mangel av protokollspesifikke opplysningene (felt og verdier) skal forhindres av denne serveren, før meldinger når mottaket.

5.5.1.1 Validering av email-header (protokollspesifikke felt)

Validering av email-header omfatter sjekk av at id-, dato- og adresseelementene i MIME header eksisterer og at de har innhold.

Oppslag mot Adresseregisteret for å kontrollere at adressen er riktig gjøres senere i valideringsprosessen, etter at HER-id er funnet (Se kapittel 5.9.3).

Regel Id	Beskrivelse	Referanse
18	[From:] mangler i MIME header	ebMS [2] kap B.3.2
19	[To:] mangler i MIME header	ebMS [2] kap B.3.2
80	Tom [From:] i MIME header	ebMS [2] kap B.3.2
81	Tom [To:] i MIME header	ebMS [2] kap B.3.2

Retur av en SOAP Fault krever Message-Id for utfylling av In-Reply-To slik at responsen kan kobles:

Regel Id	Beskrivelse	Referanse
250	[Message-ID:] mangler i MIME header	ebMS [2]
20	[Date:] mangler i MIME header	ebMS [2] kap B.3.2
251	Tom [Message-ID:] i MIME header	ebMS [2] kap B.3.2
83	Tom [Date:] i MIME header	ebMS [2] kap B.3.2

5.5.2 MIME-validering

Validering av MIME-felt i meldingspakkens toppnivå, SOAP-del og attachement-del. Feil i MIME valideringen skal besvares med en SOAP Fault.

ebMS 2.0 [2] krever at MIME-feil meldes iht. Soap with Attachments [5], som igjen viser til 4.4.1 SOAP Fault Codes i [11]. De generelle MIME-feltene skal normalt kunne meldes med en SOAP Fault.

MIME-header, SOAP-delen og eventuell attachement-del kontrolleres av disjunkte regelsett. Selv om MIME-header og SOAP-delen refererer til samme MIME-strukturdel i en melding uten multipart, så er reglene enten helt uavhengig av content type (og på toppnivå), relatert til text/xml (SOAP) eller dekker multipart-relaterte attributter (mellomkategori).

5.5.2.1 Validering av MIME-header (toppnivå-entitet)

SOAPAction-feltet kan brukes *av andre* til å skille ut annen trafikk over samme kanal, siden vi validerer at det blir brukt.

Et potensielt unntak kunne vært *bounce*-meldinger (SMTP-feil) som et resultat av egen sending, men de er allerede filtrert vekk fra trafikken som en del av protokollhåndteringen. Krev SOAPAction på alt.

Regel Id	Beskrivelse	Referanse
21	[MIME-Version:] mangler i MIME header	ebMS [2] kap B.3.2
82	MIME-Version i MIME header skal være "1.0"	ebMS [2] kap B.3.2
22	[SOAPAction:] mangler i MIME header	ebMS [2] kap B.3.2
51	SOAPAction i MIME header skal være "ebXML"	ebMS [2] kap 2.1.2
252	[Content-Type:] er "text/plain" eller mangler i MIME header	ebMS [2] kap B.3.2
66	MIME er hverken multipart/related eller text/xml	ebMS [2] kap 2.1.2

Text/xml på toppnivå indikerer en ren signalmelding (eller ufullstendig payloadmelding).

Selv om det kun er meldinger med payload som krever multipart/related-innpakning på toppnivået, så kan det forekomme rene ebXML-signaler som er pakket i en overflødig multipart/related.

5.5.2.2 Validering av multipart-attributter

Content-Type: multipart/related; type="text/xml"; boundary="----=_Part_17022"

Regel Id	Beskrivelse	Referanse
253	Content-type sitt type-attributt i MIME multipart/related skal være "text/xml"	ebMS [2] kap 2.1.2
-	Motsigende start-parameter i toppnivået og Content-id i SOAP-mimedelen	ebMS [2] kap 2.1.2

Den siste regelen kan kontrolleres når man har kommet til SOAP-delen, strukturelt sett, se neste punkt.

5.5.2.3 Validering av MIME-del med SOAP

Eksempel på header-innhold i SOAP-delen:

Content-Id: <soap-part>
 Content-Type: text/xml; charset="UTF-8"
 Content-Transfer-Encoding: 8bit

Gjeldene praksis er at data-innholdet i SOAP mime-delen er (merket vha. XML-deklarasjonen som):

<?xml version="1.0" encoding="UTF-8"?>

Deklarasjonen behandles som en del av selve SOAP-konvolutten, beskrevet i kapittel 5.6. Eventuelle forskjeller mellom content-type charset og encoding oppgitt i deklarasjonen vil fanges opp indirekte.

Regel Id	Beskrivelse	Referanse
67	Text/xml-del mangler i MIME-meldingen	ebMS [2] kap 2.1.2
255	Content-type i SOAP-mimedelen skal være "text/xml"	ebMS [2] kap 2.1.3.1
256	Charset for SOAP-mimedelen skal være "UTF-8"	ebMS [2] kap 2.1.3.2
257	Content-transfer-encoding mangler i SOAP-mimedelen	Offentlig standard [8]
258	Content-transfer-encoding for SOAP-mimedelen er ikke en av følgende: 8bit, binary, quoted-printable, base64	ebMS [2] kap. B.3.2

5.5.2.4 Validering av MIME-del med kryptert innhold (attachement-del)

Eksempel på header-innhold i attachement-delen:

Content-Id: <Vedlegg-0>
 Content-Type: application/pkcs7-mime; smime-type=enveloped-data
 Content-Transfer-Encoding: base64

Regel Id	Beskrivelse	Referanse
259	Content-transfer-encoding i vedleggs-del skal være "base64"	Profil [1] kap 5
260	Content-type i vedleggs-del skal være «application/pkcs7-mime; smime-type=enveloped-data»	Profil [1] kap 5.5

MIME-utpakking, så langt, skal gi oss en SOAP-konvolutt og eventuelle vedlegg. Som observatør er dette så langt validatoren kommer. Ved direktemottak kan vedlegget pakkes videre ut, i form av dekrypteres til en MIME-del med payload, som beskrevet i kapittel 5.15. Her beskrives videre behandling av SOAP-konvolutten.

5.6 XML-skjemavalidering

Selve SOAP-konvolutten starter eventuelt med en XML-deklarasjon:

```
<?xml version="1.0" encoding="UTF-8"?>
```

Deklarasjonen behandles etter reglene gitt i ebMS 2.0 [2] kapittel 2.2.1 og SOAP-konvolutten skjemavalideres (dette leder opp mot reglene gitt i ebMS 2.0 [2] kapittel 4.2.3.4.1). Aktuelle feil er:

- Hvis deklarasjonen er oppgitt, så skal versjonen være 1.0
- XML som ikke er *well formed*
- XML som ikke er *valid*¹

Meldinger med slike feil skal ikke sendes. De kan eventuelt avvisning ved mottak i form av en SOAP Fault.

5.6.1 Skjemavalidering av SOAP og ebXML

Validering mot XSD m.m.

¹ Merk at skjemaet fra Oasis (http://www.oasis-open.org/committees/ebxml-msg/schema/msg-header-2_0.xsd) inneholder en tvetydighetsfeil:

0..* ds:Reference under Acknowledgment-elementet er etterfulgt av en generell åpning for utvidelser ved hjelp av et xs:any med ##other som namespacekrav. For alle ds:Reference som er plassert her, er det ikke entydig om de skal tolkes som en ds:Reference eller en ##other. Det er stilt krav om bruk av signert kvittering i profilen [1], så ds:Reference-feltene skal være i bruk og valideres. Det er antagelig ingen ##other i bruk i norsk helsesektor. Valgt løsning er bruk av et alternativt skjema der den generelle åpningen lukkes (fjernes helt) så ds:Reference er entydig.

Regel Id	Beskrivelse	Referanse
78	XML-deklarasjon mangler (<?xml version="1.0" encoding="UTF-8"?>)	ebMS [2] kap 2.2.1
68	XML er ikke encodet som UTF-8 (https://www.difi.no/artikkel/2015/10/tegnsett#utveksling)	Veiledning ebXML [4] kap 7.2
86	Innholdet i meldingen kan ikke identifiseres som ebXML (ut fra rot-namespace)	ebMS [2] kap. 2.2.2
16	XML følger ikke SOAP-skjema	Offentlig standard [8]
17	XML følger ikke ebXML-skjema	ebMS [2] kap 2.3

Hvis skjemavalideringen ikke kan utføres, så stoppes valideringen på dette punktet. Hvis skjemavalideringen finner feil fortsetter en ren validator. Det er, med hensikt, delvis overlap mellom feil funnet av skjema og resterende regler.

5.7 Validering av innhold i SOAP Envelope

5.7.1 Validering av påkrevd innhold i SOAP Envelope.

Regel Id	Beskrivelse	Referanse
43	[SOAP:Envelope] mangler i meldingen	SOAP [5] kap 4
2	[SOAP:Header] mangler i SOAP:Envelope	SOAP [5] kap 4 Profil [1] kap 6.1
3	[SOAP:Body] mangler i SOAP:Envelope.	SOAP [5] kap 4

5.7.2 SOAP-prosessering

En MSH kan betraktes som en *ebXML SOAP Processor*. Den skal være i samsvar med både ebMS og SOAP. Ut fra dette er det lagt til noen regler.

SOAP Processing baserer seg på de to feltene:

- soap:actor (angir roller)
- soap:mustUnderstand (angir tvungen prosessering)

Uttrykt med SOAP-semantikken vil validatoren innta alle disse rollene:

- SOAP:next (krever egentlig SOAP-samsvar)

- SOAP:ultimateReceiver² (ditto, denne rollen er tiltenkt mottaker i fravær av en actor-oppføring)
- actor:toPartyMSH (krever egentlig ebMS-samsvar [2])
- actor:nextMSH (krever egentlig ebMS-samsvar [2])

ebXML-elementer med mustUnderstand="0" skal valideres, på tvers av SOAP 1.1 [11].

For elementer med egne tabeller er eksistens- og verdisjekk av attributtet mustUnderstand listet i hver enkelt tabell. Her listes det kun to regler, som en generell hvitvask av innholdet i SOAP-konvolutten.

En MSH skal utføre reell SOAP-prosessering med *eb:MessageHeader* og *ds:Signature* som naturlige førstevalg. Hvis signaturen sjekkes aller først, og ikke stemmer, skal prosesseringen fortsette med *eb:MessageHeader* inntil man har nok informasjon til å returnere en *ErrorList* (som angir at det var en signaturfeil), eller feile med en *Fault* før man kommer så langt, uten at en slik *Fault* relateres til signaturen.

En meldingsvalidator regner kun opp innholdet av Header og Body, for å reagere på uventet innhold.

Elementene som forventes å forekomme i en ebXML-melding sin SOAP:Header er

- MessageHeader
- AckRequested
- Acknowledgment
- ErrorList
- ds:Signature
- MessageOrder

SyncReply skal ikke forekomme i Header ved mottak over SMTP.

Regel Id	Beskrivelse	Referanse
100	Ukjent eller feilplassert XML-blokk (med SOAP:mustUnderstand="1") i SOAP:Header	SOAP [11] kap 4.2.3 ebMS [2] kap 2.3.5 og 6.3
101	Ukjent eller feilplassert XML-blokk i SOAP:Body	SOAP [11] kap 4.2.3 ebMS [2] kap 2.3.5 og 6.3

Elementene som generelt kan forekomme i en ebXML-melding sin SOAP:Body er Manifest, StatusRequest og StatusResponse.

² Klart definert først i SOAP 1.2, ikke i SOAP 1.1 som egentlig er underlaget for denne spesifikasjonen (via ebMS 2.0 [2]).

For elementer med egne tabeller er eksistens- og verdisjekk av attributtet mustUnderstand listet i hver enkelt tabell. Her lister vi kun to regler, som en generell hvitvask av innholdet i SOAP-konvolutten.

5.8 Validering av MessageHeader

I MessageHeader skal blokken eb:MessageHeader alltid være med.

Dersom det ikke er mulig å lese ut korrekt kommunikasjonspart så Brukes SOAP Fault til MAIL From. Straks det er grunnlag for ebXML Error Signal (mest aktuelt for en MSH), skal den brukes for avvisning eller advarsler.

5.8.1 Validering av XML-elementer i MessageHeader

Regel Id	Beskrivelse	Referanse
4	[MessageHeader] mangler i SOAP:Header	ebMS [2] kap 3.1
8	[@version] mangler i MessageHeader	ebMS [2] kap 3.1 ebMS [2] kap 2.3.8
70	MessageHeader/@version skal være "2.0"	ebMS [2] kap 3.1 ebMS [2] kap 2.3.8
7	[@SOAP:mustUnderstand] mangler i MessageHeader	ebMS [2] kap 3.1
69	MessageHeader/@SOAP:mustUnderstand skal være "1" (sann)	ebMS [2] kap 3.1

5.8.2 Validering av eb:MessageId (sjekksum og mottakslogging)

eb:MessageId gir, alene, en unik identifikator av en melding. Enhver endring i en melding skal gi ny verdi, slik at en MSH kan bruke eb:MessageId til duplikateliminerings og korrekt sammenkobling av meldinger.

Påstanden kan etterprøves ved å sammenholde eb:MessageId og sjekksummer. Hvis en melding mottas på nytt, med samme sjekksummer, har vi et ekte gjensyn. Forskjeller indikerer falske gjensyn.

Id og sjekksummer lagres i mottaksloggen sammen med det reelle mottakstidspunktet (en avgrenset tid).

Regel Id	Beskrivelse	Referanse
76	[Messageld] mangler i MessageHeader/MessageData/	ebMS [2] kap 3.1.6.1
102	Messageld skal være en UUID.	Profil [1] kap 6.2.1
77	[Timestamp] mangler i MessageHeader/MessageData	ebMS [2] kap 3.1.6.2

Ved ekte gjensyn kan videre validering kuttes. Da holder det å notere ned det nye mottaket (tidspunkt etc.) og gi korrekt respons på nytt, så lenge oppføringene i Adresseregisteret fortsatt er de samme.

Normal duplikathåndtering tilsier at man stoler på eb:Messageld, og eliminerer duplikater. Her kan denne sjekken utvides, og kun eliminere *ekte* duplikater.

Skjema-valideringen gir at det finnes én og kun én eb:Messageld, med en verdi. Den skal være en UUID.

/SOAP:Envelope/SOAP:Header/eb:MessageHeader/eb:MessageData/eb:Messageld

Allerede utført kontroll skal ha sikret at det ligger en signatur i meldingen:

/SOAP:Envelope/SOAP:Header/ds:Signature

Skjema-valideringen sikrer at det ligger minst én Reference i en signatur, med en DigestValue

/SOAP:Envelope/SOAP:Header/ds:Signature/ds:SignedInfo/ds:Reference[1]/ds:DigestValue

/SOAP:Envelope/SOAP:Header/ds:Signature/ds:SignedInfo/ds:Reference[2]/ds:DigestValue

Plukk ut alle DigestValues, en fra hver referanse i signaturen (de utgjør en sjekksum for hver del av meldingen). Hvis en melding mottas med samme Id og forskjellig(e) sjekksum(mer) så varsles dette. Hvis instanser av samme melding har samme id, så skal ny sjekksum gi treff på notert id også.

Den reelle sjekken av sjekksummenes korrekthet skjer som en del av signatursjekken.

Om mulig noteres det også en advarsel på impliserte meldinger. I observatørmodus (statistikk) kan denne handlingen utelates helt, av ytelseshensyn, siden den indirekte forholder seg til flere meldinger.

5.8.3 Validering av CPAId

Regel Id	Beskrivelse	Referanse
11	[CPAId] mangler i MessageHeader	ebMS [2] kap 3.1.2
12	CPAId skal være formatert på en av følgende måter: <ul style="list-style-type: none"> • HER(minst)_HER(størst). Eksempel: 1001_1212 • UUID 	Profil [1] kap 6.2.1

Merk at HER(minst)_HER(størst)_løpenummer kan forekomme og bør ikke resultere i feil.

En UUID kan også brukes, hvis man har registrert inn en cpa-avtale i Adresseregisteret.

5.8.4 Validering av avsender og mottaker elementer i MessageHeader

Avsender og mottaker skal identifiseres med hver sin HER-id, plassert i henholdsvis From og To.

Selv andre id-typer enn HER-id og organisasjonsnummer (angitt i profilen [1]) får ligge i meldingen.

Regel Id	Beskrivelse	Referanse
10	[From] mangler i MessageHeader	ebMS [2] kap 3.1.1
9	[To] mangler i MessageHeader	ebMS [2] kap 3.1.1
58	[PartyId] mangler i From	ebMS [2] kap 3.1.1
59	[PartyId] mangler i To	ebMS [2] kap 3.1.1
25	Tom PartyId i From	ebMS [2] kap 3.1.1
55	Tom PartyId i To	ebMS [2] kap 3.1.1
60	From/PartyId av type "HER" mangler	Tjenestebasert adressering (HIS 1153-2:2016) [3] kap 3.2, krav AD2.1
61	To/PartyId av type "HER" mangler	Tjenestebasert adressering (HIS 1153-2:2016) [3] kap 3.2, krav AD2.1
14	Det skal være eksakt én HER-id i From-elementet	ebMS [2] kap 3.1.1 Tjenestebasert adressering (HIS 1153-2:2016) [3] kap 3.2

Regel Id	Beskrivelse	Referanse
13	Det skal være eksakt én HER-id i To-elementet	ebMS [2] kap 3.1.1 Tjenestebasert adressering (HIS 1153-2:2016) [3] kap 3.2
31	Avsenders HER-id skal bestå av tall	Tjenestebasert adressering (HIS 1153-2:2016) [3] kap 3.2, krav AD2.1
56	Mottakers HER-id skal bestå av tall	Tjenestebasert adressering (HIS 1153-2:2016) [3] kap 3.2, krav AD2.1

5.8.5 Validering av Service og Action

Her begrenser en seg til å sjekke at verdien i Service og Action er satt. Ytterligere kontroll skjer først som del av de særskilte reglene. Verdiene kan tas vare på, og senere sjekkes som en del av validering av payload.

Regel Id	Beskrivelse	Referanse
47	[Service] mangler i MessageHeader	ebMS [2] kap 3.1.4
71	Tom Service i MessageHeader	ebMS [2] kap 3.1.4
48	[Action] mangler i MessageHeader	ebMS [2] kap 3.1.5
72	Tom Action i MessageHeader	ebMS [2] kap 3.1.4

5.9 Oppslag mot Adresseregisteret (AR)

Oppslag mot Adresseregisteret (eller lokal cache) på avsenders HER-id gjøres nå.

Bakgrunn for kontroll av dette er at det viktigste oppslaget mot Adresseregisteret før sending av en melding er avsenders MSH sitt oppslag i forhold til mottakers sertifikatinformasjon og kommunikasjonsparametere.

Oppslag i Adresseregisteret på HER-iden til kommunikasjonspartneren gir riktig endepunkt for oversending av meldinger og sertifikater som skal brukes.

Feltverdien indikerer om utvekslingen er iht. en CPA, generelle oppføringer eller lokale opplysninger. Den speiler dermed hvordan man skal forholde seg til Adresseregisteret.

Det er per i dag flere ulike varianter for definisjon av CPAId. UUID er formatet som benyttes i Adresseregisterets løsning for utveksling av kommunikasjonsparametere. Meldingsvalidatoren sjekker i Adresseregisteret om det finnes en registrert CPA mellom partene i de tilfellene hvor CPAId er definert som en UUID.

Man kan også kontrollere verdiene i SMTP-feltene From: og To.

5.9.1 Kontroll av HER-id mot AR

Regel Id	Beskrivelse	Referanse
29	Finner ikke avsenders HER-id (From/PartyId) i Adresseregisteret.	Tjenestebasert adressering (HIS 1153-1:2016) [3] kap 4.2, krav AD1.19
54	Finner ikke mottakers HER-id (To/PartyId) i Adresseregisteret.	Tjenestebasert adressering (HIS 1153-1:2016) [3] kap 4.2, krav AD1.18
30	Avsenders HER-id (From/PartyId) er ikke av riktig type i Adresseregisteret (AR). Lovlige typer i AR er Service og OrganizationPerson.	Tjenestebasert adressering (HIS 1153-2:2016) [3] kap 3.2, krav AD2.1
54	Mottakers HER-id (To/PartyId) er ikke av riktig type i Adresseregisteret (AR). Lovlige typer i AR er Service og OrganizationPerson.	Tjenestebasert adressering (HIS 1153-2:2016) [3] kap 3.2, krav AD2.1

Ikke send meldinger uten korrekte kommunikasjonsparter. Avvis dem eventuelt med en SOAP Fault.

5.9.2 Kontroll av CPAId mot AR

Hvis man aktivt bruker en avtale, så kontrolleres verdien her.

Hvis CPAId var en UUID, så indikerer det aktiv bruk av en avtale som er registrert inn i Adresseregisteret.

Alle kan fortsatt basere seg på standardverdiene, også de med avtale, ved å bruke CPAId uten UUID.

Regel Id	Beskrivelse	Referanse
110	Avtalen som er registrert i Adresseregisteret stemmer ikke med UUID oppgitt i CPAId.	Med avtale og en validator som kun implementerer minimalt samsvar (kap 4.7).

Regelteksten "Avtalen som er registrert i Adresseregisteret stemmer ikke med UUID oppgitt i CPAId." brukes hvis validatoreren ikke implementerer mer enn minimumskravet som stilles til den i denne spesifikasjonen (om merking av feilrapporten som helhet). Validatoren bør forholde seg til minst to deler av CPA-innholdet: de aktuelle partene og gyldighetsperioden. I så fall erstattes regeldelen med følgende fire regler som forholder seg til den returnerte avtalen (UUID mot AR):

Regel Id	Beskrivelse	Referanse
111	Avsenders HER-id (From/PartyId) gjenfinnes ikke i oppgitt avtale (UUID i CPAId)	CPP/CPA [9] kapittel 3.3.1.1.1
112	Mottakers HER-id (To/PartyId) gjenfinnes ikke i oppgitt avtale (UUID i CPAId)	CPP/CPA [9] kapittel 3.3.1.1.1
113	Meldingens eb:Timestamp inntreffer tidligere enn gyldighetsperioden til oppgitt avtale (UUID i CPAId)	CPP/CPA [9]
114	Meldingens eb:Timestamp inntreffer senere enn gyldighetsperioden til oppgitt avtale (UUID i CPAId)	CPP/CPA [9]

5.9.3 Kontroll av edi-adresser mot AR

Hvis HER-id var i Adresseregisteret kan edi-adresser også kontrolleres mot registeret nå.

Regel Id	Beskrivelse	Referanse
261	Verdien i [From:] i Mime headeren stemmer ikke overens med avsenders edi-adresse i Adresseregisteret	Tjenestebasert adressering (HIS 1153-2:2016) [3] kap 3.2, krav AD2.1
262	Verdien i [To:] i Mime headeren stemmer ikke overens med mottakers edi-adresse i Adresseregisteret	Tjenestebasert adressering (HIS 1153-2:2016) [3] kap 3.2, krav AD2.1

5.9.4 Kontroll av sertifikat mot AR

Regel Id	Beskrivelse	Referanse
79	Fant ikke avsender (From/PartyId[@type="HER"]) sitt signeringssertifikat	Veiledning ebXML [4], kap 5.2.5
44	Sertifikatet som er benyttet for signering (Signature/KeyInfo/X509Data/X509Certificate) stemmer ikke med signeringssertifikat i Adresseregisteret	Veiledning ebXML [4], kap 5.2.5

Den siste regelen kan slå til hvis man revaliderer en gammel melding (sendt uten bruk av CPA), hvis AR er oppdatert etter at meldingen ble generert, selv om den opprinnelig var riktig.

5.10 Kontroll av signatur

Formatet for ds:Signature [8] er anvendt i ebMS 2.0 [2], tilpasset i den norske profilen [1] og videre presisert i veiledningen [4].

Sertifikatet assosiert med nøkkelen brukt til å signere meldingen med, skal ligge i:

/SOAP:Envelope/SOAP:Header/ds:Signature/ds:KeyInfo/ds:X509Data/ds:X509Certificate

Signaturen sjekkes mot dette sertifikatet. Meldingen skal være generert i sertifikatets gyldighetsperiode. Sertifikatet skal være det samme som ligger i adresseregisteret (så lenge meldingen er generert etter det forrige sertifikatskiftet), dette kontrolleres senere.

5.10.1 Validering av signatur

Regel Id	Beskrivelse	Referanse
45	[Signature] mangler i SOAP:Header	Profil [1] kap 6.2
52	Meldingen har mer enn ett Signature element	Profil [1] kap 6.2 Profil [1] kap 6.2.3
363	[SignedInfo] mangler i Signature	Profil [1] kap 6.2.3
42	[SignatureValue] mangler i Signature	Profil [1] kap 6.2.3
32	[KeyInfo] mangler i Signature	Profil [1] kap 6.2.3
39	Det må være minimum en [Reference] i SignedInfo.	Profil [1] kap 6.2.3
40	Rot-referansen mangler i SignedInfo (ingen Reference med @URI="").	Profil [1] kap 6.2.3
34	Minst en payloadreferanse i SignedInfo begynner ikke med [cid:]	Profil [1] kap 6.2.3
33	[@Algorithm] i minst en Reference/DigestMethod er ikke "http://www.w3.org/2000/09/xmldsig#sha1"	Profil [1] kap 6.2.3
37	Transforms/Transform[1]/@Algorithm i rotreferansen skal være "http://www.w3.org/2000/09/xmldsig#enveloped-signature"	Profil [1] kap 5.5
36	Transforms/Transform[2]/@Algorithm i rotreferansen skal være "http://www.w3.org/TR/1999/REC-xpath-19991116"	Profil [1] kap 5.5

Regel Id	Beskrivelse	Referanse
35	Transforms/Transform[3]/@Algorithm i rothereferansen skal være "http://www.w3.org/TR/2001/REC-xml-c14n-20010315"	Profil [1] kap 5.5
38	Det skal være 3 (tre) transforms i denne rekkefølgen, DsigEnvelopedSignature, Xpath, DsigC14N	Profil [1] kap 5.5 [X509Certificate] element er ikke funnet i ds:Signature
41	[X509Data] mangler i KeyInfo.	Profil [1] kap 6.2.3 (side 38)
64	[DigestValue] mangler i Reference	Profil [1] kap 6.2.3 (side 36-37)
85	[@URI] mangler på Reference	Profil [1] kap 6.2.3 (side 36-37)
103	[X509Certificate] mangler i X509Data	Profil [1] kap 5.5[X509Certificate] element er ikke funnet i ds:Signature
65	Mer enn ett X509Certificate i meldingens X509Data	Profil [1] kap 6.2.3 (side 38)
46	Meldingen inneholder feil type sertifikat (ugyldige data i X509Certificate)	Profil [1] kap 6.2.3 (side 38)
104	Meldingens eb:Timestamp inntreffer tidligere enn signeringssertifikatets gyldighetsperiode	Profil [1] kap 6.2.3, veileder [4] kap 5.2.5 ³
105	Meldingens eb:Timestamp inntreffer senere enn signeringssertifikatets gyldighetsperiode	Profil [1] kap 6.2.3, veileder [4]kap 5.2.5
84	Metoden angitt i ds:SignatureMethod/@Algorithm støttes ikke, anbefalt verdi er " http://www.w3.org/2000/09/xmldsig#rsa-sha1 " (merk: RSA, ikke DSA)	Profil [1] kap 6.2.3
50	Meldingen er ikke korrekt signert eller har blitt endret etter signaturen	Profil [1] kap 5.5 Profil [1] kap 6.2.3

³ Når alt er rett har vi Valid From < eb:Timestamp < Valid To

Selve signatursjekken kan utføres tidligere. En norsk MSH kan starte SOAP-prosesseringen med signaturen som første element. En ugyldig signatur gir kutt av videre behandling og avvisning.

Her skal valideringen fortsette selv om signatursjekken feiler.

5.11 Identifikasjon av meldingstype

Validatoren skal identifisere meldingstypen ut fra elementene i SOAP:Header og SOAP:Body.

5.11.1 Ping og Pong

Hittil er MessageHeader og ds:Signature kontrollert. Hvis det ikke finnes flere elementer må meldingen være en Ping, en Pong eller mangelfull. Vask for Ping og Pong skjer vha. Service og Action:

eb:MessageHeader/eb:Service skal være «urn:oasis:names:tc:ebxml-msg:service»

eb:MessageHeader/eb:Action skal være «**Ping**» eller «**Pong**»

Regel Id	Beskrivelse	Referanse
-	Meldingen hadde ingen elementer ut over MessageHeader og ds:Signature (og var hverken Ping eller Pong).	ebMS [2] kap 8

Hvis denne regelen slår til, skal validatoren avslutte all ytterligere validering.

5.11.2 StatusRequest, StatusResponse eller payloadmelding

Elementene som generelt kan forekomme i en ebXML-melding sin SOAP:Body er

- Manifest
- StatusRequest
- StatusResponse.

Regel	Konklusjon
Elementet <eb:Manifest> eksisterer	Meldingen antas å være en payloadmelding. Valideringsregler i kapittel 5.14 utføres.
Elementet <eb:StatusRequest> eksisterer	Meldingen er en StatusRequest. Ingen ytterligere validering utføres.
Elementet <eb:StatusRespon> eksisterer	Meldingen er en StatusRequest. Ingen ytterligere validering utføres.

Hvis meldingen er en StatusRequest eller StatusResponse skal validatoren avslutte all valideringen.

Hvis det forekommer mer enn ett element i Body, så er det en feil.

Regel Id	Beskrivelse	Referanse
-	Mer enn ett element i SOAP:Body	ebMS [2]

5.11.3 Payload- eller signal melding

Ping, Pong, StatusRequest og StatusResponse meldinger er sortert ut. Resten skal beholdes inntil videre. Noen er typet som payloadmeldinger. Resten skal være signalmeldingene

Acknowledgment eller Error Signal. Forventet innhold i SOAP:Header er

- MessageHeader
- AckRequested
- Acknowledgment
- ErrorList
- ds:Signature
- MessageOrder.

SyncReply skal ikke forekomme ved mottak over SMTP.

Det tredje elementet vi finner lar oss type-bestemme signalene.

Regel	Konklusjon
Elementet <eb:Acknowledgment> eksisterer	Meldingen er en transportkvittering. Valideringsregler i kapittel 5.12 må utføres.
Elementet <eb:ErrorList> eksisterer	Meldingen er en transportfeilmelding. Valideringsregler i kapittel 5.13 må utføres
Elementet <eb:AckRequested> eksisterer	Dette bekrefter antagelsen om payloadmelding. Valideringsregler i kapittel 5.14 må utføres.
Elementet <eb:MessageOrder> eksisterer	Validatoren skal ignorere dette elementet

Ved sending over en MSH som har implementert det norske ebXML-rammeverket [1] skal Acknowledgment, Error Signal og payloadmeldinger sendes hver for seg, så Acknowledgment, ErrorList og (AckRequested | Manifest) skal ikke forekomme i samme melding.

Regel Id	Beskrivelse	Referanse
-	Elementene Acknowledgment og ErrorList funnet i en og samme melding.	ebMS [2] kap 6.3.2
-	Elementene Acknowledgment og AckRequested funnet i en og samme melding.	ebMS [2] kap 6.3.2
-	Elementene Acknowledgment og Manifest funnet i en og samme melding.	ebMS [2] kap 6.3.2
-	Elementene ErrorList og AckRequested funnet i en og samme melding.	ebMS [2] kap 6.3.1
-	Elementene ErrorList og Manifest funnet i en og samme melding.	ebMS [2] kap 5.1.2
-	Meldingen skal ha eksakt ett {...} element	ebMS [1] kap 2.1.4 og 3.2 Profil [1] kap 7.2

Hvis noen av elementene dukker opp mer en gang, så skal det varsles med en konkret feilmelding, for eksempel "Meldingen skal ha eksakt ett eb:Acknowledgment element", ikke en generell feilmelding.

Hvis vi har en potensiell dobbel-typing av meldinger, på grunn av feil sammensetning, så skal den videre valideringen skje som en av typene hvis innholdet i Service og Action bekrefter en av dem:

eb:MessageHeader/eb:Service : «urn:oasis:names:tc:ebxml-msg:service»
 eb:MessageHeader/eb:Action : «**Acknowledgment**» eller «**MessageError**»

Alle andre verdier i Service bekrefter meldingen som en payloadmelding. Hvis man ikke kan løse type-konflikten ved hjelp av dette, så skal ikke særvalideringen basert på type utføres, kun de andre sjekkene frem til den.

5.12 Spesifikke valideringsregler for transportkvittering (Acknowledgment)

Følgende regler gjelder særskilt for validering av transportkvittering. Disse skal kun kontrolleres når det er fastslått at meldingen er en transportkvittering (Acknowledgment).

Regel Id	Beskrivelse	Referanse
106	eb:MessageHeader/Service for en Acknowledgment skal være "urn:oasis:names:tc:ebxml-msg:service"	ebMS [2] kap 6.3.2.7
107	eb:MessageHeader/Action for en	ebMS [2] kap 6.3.2.7

Regel Id	Beskrivelse	Referanse
	Acknowledgment skal være "Acknowledgment"	
108	eb:Acknowledgment/RefToMessageId skal ha en verdi i form av en UUID	ebMS [2] kap 6.3.2.3 Profil [1] kap 7.2.3 (side 44)
109	eb:MessageHeader/MessageData/RefToMessageId skal ikke ha noen verdi i transportkvittering	Profil [1] kap 4.2 (tabell side 18)
115	Meldingen skal ikke inneholde elementet eb:ErrorList	Profil [1] kap 4.2 (side 18)
116	eb:Acknowledgment/ds:Reference med URI="" mangler	ebMS [2] kap. 6.3.2.5 Profil [1] kap 5.5 (side 24) Profil [1] kap 6.2.3 (side 36)
117	eb:Acknowledgment/ds:Reference med URI="cid:xxx" mangler	ebMS [2] kap. 6.3.2.5 Profil [1] kap 5.5 (side 24) Profil [1] kap 6.2.3 (side 36)

De to siste kan fange opp følgefeil av feil i payloadmeldingen.

5.13 Spesifikke valideringsregler for transportfeilmelding (ebXML Error Signal)

Følgende regler gjelder særskilt for validering av transportfeilmeldinger. Disse skal kun kontrolleres når det er fastslått av meldingen inneholder elementet eb:ErrorList.

Regel Id	Beskrivelse	Referanse
120	eb:MessageHeader/Service for en ErrorList skal være «urn:oasis:names:tc:ebxml-msg:service»	ebMS [2] kap 4.2.4.3
121	eb:MessageHeader/Action for en ErrorList skal være « MessageError »	ebMS [2] kap 4.2.4.3
122	eb:MessageHeader/MessageData/RefToMessageId skal ha en verdi i form av en UUID	Profil [1] kap 7.2.1 (side 42-43)
123	Meldingen skal ikke inneholde elementet eb:Acknowledgment	Profil [1] kap 4.2 (side 18)

5.14 Spesifikke valideringsregler for ebXML-konvolutt

I Figur 4: Valideringsprosessen så benyttes begrepet meldingskonvolutt om ebXML-konvolutten.

5.14.1 Kontroll av MessageHeader

DuplicateElimination skal være satt i meldinger med payload (gir håndtering av resending).

Regel Id	Beskrivelse	Referanse
74	[eb:DuplicateElimination] mangler i eb:MessageHeader	Profil [1] kap 4.2 Profil [1] kap 6.2.1

Validatoren selv utfører ikke normal duplikateliminerings, den har sin egen mekanisme, se 5.8.2.

5.14.2 Kontroll av AckRequested

Transportkvitteringer skal etterspørres med elementet eb:AckRequested.

Regel Id	Beskrivelse	Referanse
75	eb:AckRequested mangler i SOAP:Header	Profil [1] kap 4.2 Profil [1] kap 6.2.2
-	[@SOAP:mustUnderstand] mangler i eb:AckRequested.	ebMS [2] kap 6.3.1
-	eb:AckRequested /@SOAP:mustUnderstand skal være «1» (true)	ebMS [2] kap 6.3.1
94	eb:signed attribute må være tilstede og skal være (true 1) i eb:AckRequested elementet	Profil [1] kap 6.2.2

Merk at AckRequested ikke skal eksistere i transportkvitteringer og feilmeldinger.

5.14.3 Kontroll av eb:Manifest

Soap-body skal inneholde et manifest – hvis det følger med payload i ebXML-meldingen. Eksempel:

```
<SOAP:Body>
  <eb:Manifest eb:version="2.0">
    <eb:Reference xlink:href="cid:b351f2c2-a0c5-4e53-a59f-8ae7bb2c29b9"/>
  </eb:Manifest>
</SOAP:Body>
```

Her skal cid-referansen (eb:Reference) samsvare med Content-ID i mime-multiparten som vedlegget er pakket i.

Transportkvitteringer og feilmeldinger skal ikke inneholde manifest.

Regel Id	Beskrivelse	Referanse
5	eb:Manifest/@version skal være "2.0"	ebMS [2] kap 2.3.8
6	[eb:Reference] mangler i eb:Manifest	ebMS [2] kap 3.2
27	eb:Manifest/eb:Reference oppgitt i melding uten faktisk vedleggs-del	ebMS [2] kap 3.2.2
-	Tom @xlink:href i eb:Manifest/eb:Reference	ebMS [2] kap 3.2.2
23	Finner ikke Content-Id som referert i eb:Manifest noe sted i multipart-strukturen	ebMS [2] kap 3.2.2 og kap 2.1.4

5.15 Spesifikke regler knyttet til modusen «adressert mottaker»

Når en meldingsvalidator benyttes i modusen «adressert mottaker» ref kapittel 4.2.1 kan meldingsvalidatoren dekryptere vedlegget, headere i klartekst kontrolleres og payloaden avleveres.

Den norske profilen [1] av ebMS beskriver kryptering og dekryptering av payloaden som en del av ansvaret til meldingstjeneren. En normal meldingstjener skal produsere en feilmelding (ErrorList med Error) hvis dekrypteringen feiler. Meldingsvalidator skal kun notere ned feilen.

Payload overføres som et vedlegg. Overført som klartekst legges payloaden rett inn i MIME-delen som inneholder vedlegget.

Ved kryptering vil vedlegget, forenklet sagt, inneholde en kryptert MIME-del. Vedlegget er egentlig en CMS-struktur som inneholder en kryptert MIME-del. Strukturen rommer både innholdet og en del metainformasjon. Den gir at innholdet er kryptert med en symmetrisk nøkkel som også ligger i strukturen, asymmetrisk nedkryptert. Den inkluderte nøkkelen dekrypteres ved hjelp av mottakerens privatnøkkelen, slik at innholdet kan dekrypteres. Klarteksten er en MIME-del som inneholder den egentlige payloaden.

Alt dette er i henhold til S/MIME, og deklarerert som det, ved hjelp av header-informasjonen i MIME-delen som vedlegget lå i. Implementer dette ved hjelp av anerkjente tredjepartsbiblioteker, ikke kod selv.

5.15.1 Dekryptering av vedlegget

Innholdet i MIME-delen som ikke inneholder SOAP (les: vedlegget) er et CMS-objekt.

Pakk ut (dekrypter) innholdet i CMS-objektet.

Regel Id	Beskrivelse	Referanse
263	Vedlegget skal være et CMS-objekt.	Profil [1] kap. 5.5

Vedlegget skal være en kryptert MIME-del. Klartekst-varianten er en MIME-del med payload. Klartekst-MIME skal dermed utformes i henhold payloaden. Selv om utformingen er styrt av payloaden, kan vi anta at den er XML i UTF-8 og kontrollere header-informasjon ut fra disse antagelsene.

Uten reell innsikt i payloaden ut over dette vil et eksempel på header-innhold i klartekst-MIME (payload-delen)

Content-Id: <Payload-0>
 Content-Type: application/xml
 Content-Transfer-Encoding: 8bit

Regel Id	Beskrivelse	Referanse
-	[Content-type] mangler i payload-del.	RFC 2045
264	[Content-transfer-encoding] mangler i payload-del.	RFC 2045
-	Content-type i payload-del skal antagelig være «application/xml»	RFC 2045
259	Content-transfer-encoding i payload-del er ikke en av følgende: <ul style="list-style-type: none"> • "8bit" • "binary" • "quoted-printable" • "base64" 	ebMS [2] kap. B.3.2 RFC 2045 [11] kap 6

5.15.2 Merking av Testcase

Meldinger som sendes fra en meldingsvalidator skal merkes som testmeldinger.

Meldinger med feil kan slippes (ut) i testøyemed, så lenge de er merket.

5.15.2.1 SOAP-konvolutt

Merk alle utgående konvolutter. Første innslag under <SOAP:Header> skal være:

<!-- TEST. TEST. Sendt i testøyemed. Har ingen andre formål. -->


Hvis meldingen er text/xml på toppnivå er dette tilstrekkelig merking.

5.15.2.2 MIME-innpakning

Ved bruk av MIME skal tilsvarende tekst stå oppført i MIME multipart preamble:

TEST. TEST. Sendt i testøyemed. Har ingen andre formål.

Dette gir en oppføring av teksten i preamble og en som XML-kommentar i SOAP-delen.

 Direktoratet for e-helse

Besøksadresse

Verkstedveien 1
0277 Oslo

Postadresse

Postboks 6737
St. Olavs plass
0130 OSLO