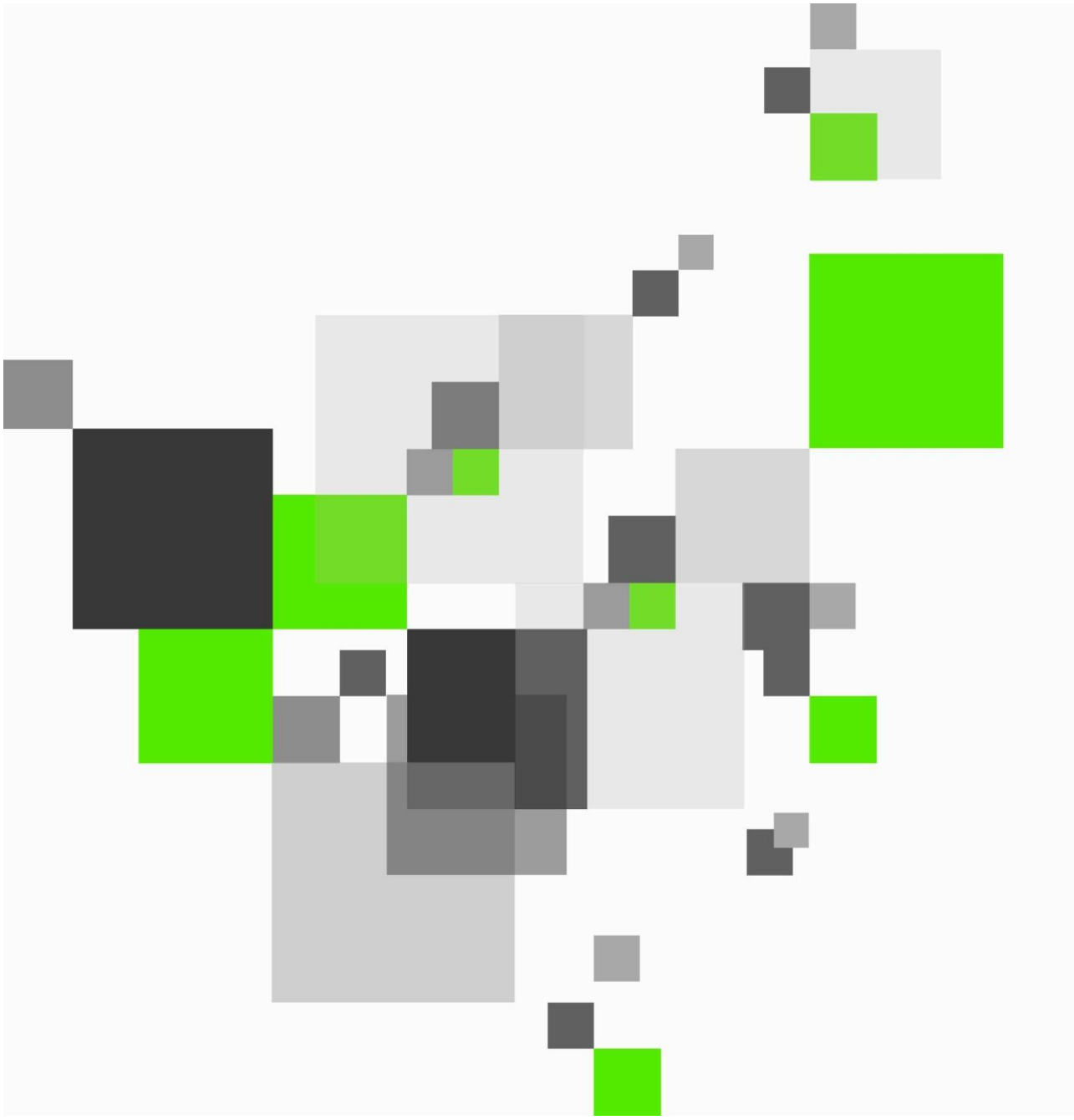
Beskedfordeler-Besked-Afsend-Snitflade



Indholdsfortegnelse

Historik

Dato	Revideret af	Ændring	BF version
2018-06-13	BIW	Eksempler på strukturer tilføjet Eksempler på fejlsituationer tilføjet Illustration af snitflader tilføjet	2.0
2020-05-20	BIW	Exchangenavn opdateret i afsnit "Etablering af AMQP forbindelse" Returkode 42 indsat i afsnittet " AMQP Output" SecurityPolicy indsat fra fællesdokumentet + appendiks 3 med dataafgrænsninger indsat	2.0
2022-08-02	AXB	Snitfladeoversigt og certifikat afsnit opdateret som følge af va-150	2.8
2022-09-28	AXB	Konsekvensrettet og reviewet i forbindelse med generel opdatering af Beskedfordeler dokumentation. Ingen ændringer i funktionalitet.	2.8
2023-01-27	AXB	Beskedfordeler understøtter nu to forskellige amqp-svar for statuskode 20.	3.0

1. Versioner

Snitfladens version er 1.0

2. Målgruppe

Integrationsudviklere der arbejder hjemmevant med SOAP og REST teknologier.

3. Snitfladebeskrivelse for SF1462

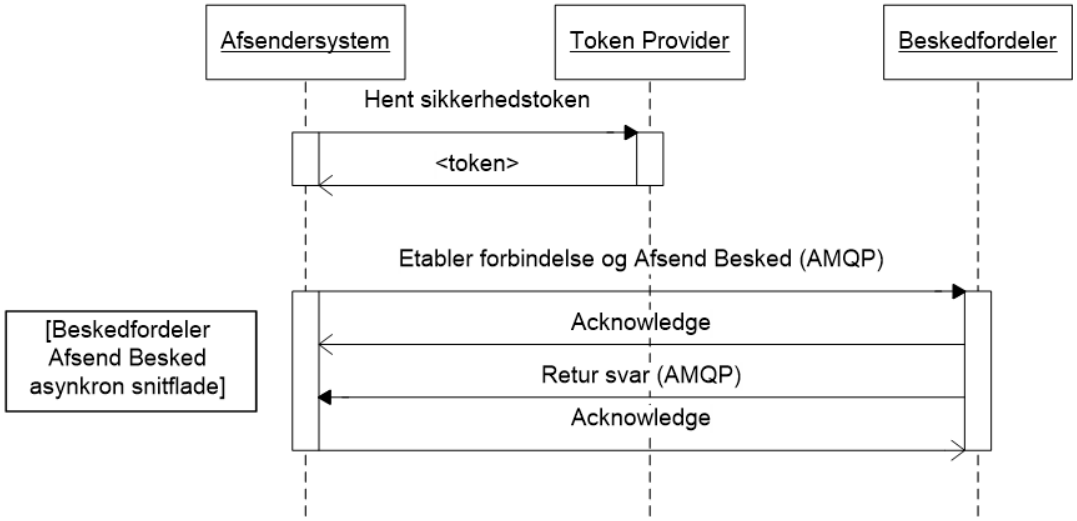
Nærværende dokument beskriver, hvorledes servicen **BeskedAfsend** skal kontaktes af **Afsendersystemet,** for at det kan afsende beskeder til **Beskedfordeler.** Beskeder, der afsendes til **Beskedfordeler,** vil blive distribueret til de **Modtagersystemer**, der via deres abonnenter ønsker og er autoriseret til at modtage disse beskeder.

Figur 1: Beskedfordelerens grænseflader. BeskedAfsend snitfladen er markeret med rød firkant

4. Servicebeskrivelse

Snitfladen kan modtage en besked af typen **Hændelsesbesked. Hændelsesbeskeder** kan indeholde beskeddata, der matcher den beskedtype, der er angivet i **Beskedkuverten**. **Beskedkuverter** struktureres som beskrevet under yderligere oplysninger i underbilag 2O - Beskedkuvert. Beskeddata kan medsendes i en struktur, der afspejler det enkelte objekt, som beskeden omhandler.

Afsendersystemet kontakter servicen via **Beskedfordelers** AMQP URL. En oversigt over **Beskedfordelers** endpoints kan findes på digitaliseringskataloget under SF1462.



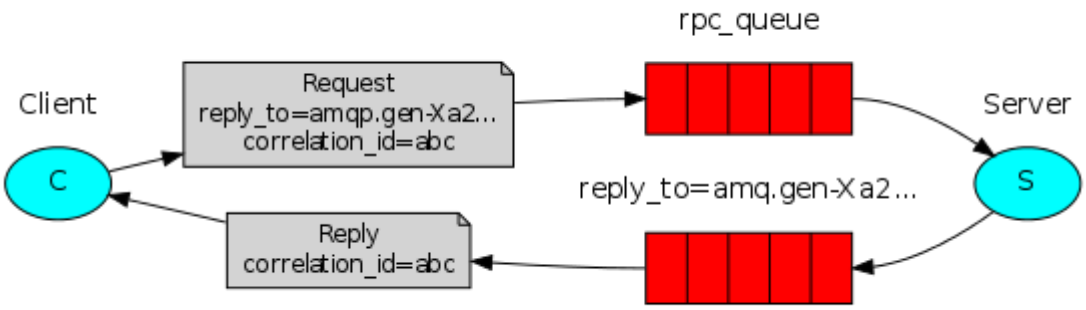
Figur 1 Kaldet til Beskedfordeler-Besked afsend-snitfladen er et asynkront AMQP kald, som udføres direkte til støttesystemet ***Beskedfordeler****. Herved bliver den afsendte besked publiceret til* ***Beskedfordelers*** *abonnenter.* ***Anvendersystemet*** *skaffer et sikkerhedstoken før kaldet til* ***Beskedfordeler****.*

5. Funktionalitet i operationen BeskedAfsend (AMQP)

Anvendersystemet kan anvende snitfladen **BeskedAfsend** over AMQP**. Anvendersystemet** kalder **Beskedfordeler** via AMQP med den token der autoriserer Anvendersystemet til at sende beskeder. Herved etablerer **Anvendersystemet** en TLS sikret kanal til Beskedfordeler. Ved anvendelse af AMQP etableres via mønstret RPC en todelt forbindelse til aflevering af beskeder og til modtagelse af svar fra **Beskedfordeler**. Når først AMQP forbindelsen er etableret, kan der afsendes flere beskeder over samme kanal. Forbindelsen lever så længe at det medsendte token er gyldigt. Til hver besked medsendes et sikkerhedstoken der angiver at Afsendersystemet er autoriseret til at sende netop denne besked.

Mønstret der benyttes ved AMQP er RPC (Remote Procedure Call). Der benyttes RPC direct-reply-to jævnfør, RabbitMQ specifikationen. Derved kan sikkerhedstoken understøttes, og **Afsendersystemet** kan modtage svar på **Beskedfordelers** modtagelse af beskeder uden at give afkald på den fleksibilitet, skalering og performance, som den asynkrone kommunikation giver. Beskeder kan ikke anses for afsendt af **Afsendersystemet,** før at der er modtaget et positivt svar om beskedens modtagelse via forbindelsens svarkanal. Svarkanalen er temporær og eksisterer derfor kun så længe **Afsendersystemets** session med servicen er aktiv. Anvendersystemet skal sørge for at modtage svar på alle afsendte beskeder før at sessionen lukkes. Beskeder der ikke er modtaget svar for skal gensendes.

Beskeden afsendes til køen **'RPC_AFSEND_BESKED_Q'** på den virtuelle host **'BF'**



*Figur 2 Kaldet til Beskedfordeler-Besked afsend-snitfladen via AMQP kald som udføres direkte til støttesystemet **Beskedfordelers** server som RPC (Remote Procedure Call). Herved kan afsendte beskeder publiceres til **Beskedfordelers** abonnenter via den ene kanal, og svaret returneres til **Afsendersystemet** via den anden kanal. Svarkøen er temporær og eksisterer kun så længe som at Afsendersystemet holder sessionen med Servicen aktiv.*

5.1.Etablering af AMQP forbindelse

Etablering af afsendelse af beskeder via AMQP.

Struktur		Beskrivelse
AMQP URL		URL til **Beskedfordelers** AMQP broker. Exchange: "AFSEND_BESKED_EXCHANGE" Virtuel host: BF

5.2.AMQP Input

Når AMQP forbindelsen til **Beskedfordelers** indbakke er etableret, kan **Afsendersystemet** aflevere beskeder gennem snitfladen. Beskeden er ikke afleveret før, at **Beskedfordeler** har afsendt et retursvar til beskeden på svarkanalen. Svarkanalen angives som AMQP properties(i parameteren ”Reply_to”) og er kun gældende så længe sessionen er aktiv(oppe). Ved nedbrud skal de forsendelser der ikke er modtaget svar på gensendes.

I ”Appendiks 1 – eksempler” afsnit ”Inputstruktur – Besked_Afsend” og ”Beriget Inputstruktur – Besked_Afsend” er der eksempler på, hvordan inputstrukturen kan se ud.

Struktur	StatusKode	Fejlbesked
AMQP AfsendBeskedInput		**Afsendersystemet** åbner en session med snitfladen og afleverer beskeder efterhånden, som de er klar. Input gives som en **Hændelsesbesked** , der sendes som et XML dokument i UTF-8 encoding som selve beskedens data (body).HændelsesBeskeder skal overholde det tilhørende skema, som angiver struktur og ved attributter for **Beskedkuvert (se Beskedkuvert.xsd).**

		Med afsendelsendelsen af beskeden skal der sættes en række AMQP headers og properties Header: Sikkerhedstoken fra STS modulet placeres under headernavnet 'token' (bemærk at header placeres i AMQP properties ved afsendelse) **AMQP properties**: MessageId: unik id for afsendelsen af beskeden, der anvendes som id for transaktionen CorrelationsID: identifikation af forespørgslen om afsendelsen af beskeden. Anvendes til identifikation af svaret. Der henvises til RabbitMQ RPC modellen. Reply_to: sættes til 'amq.rabbitmq.reply-to' for at angive at der skal oprettes en sessions afhængig temporær svarkø.

5.3. AMQP Output

Beskedfordeler svarer **Afsendersystemet**, om beskeden er modtaget og valideret korrekt på svarkanalen. Svaret er ikke modtaget før, at **Afsendersystemet** har acknowledged svaret på svarkanalen. Svaret leveres som en AMQP besked, hvis data indeholder et XML dokument.

I "Appendiks 1 – eksempler" afsnit "Outputstruktur - Besked Afsend" er der eksempler på, hvordan outputstrukturen kan se ud. I "Appendiks 2 – fejlsøgning" er der forslag til, hvad der kan forårsage en af nedenstående fejlkoder/statuskoder.

Struktur	StatusKode	Fejlbesked
AMQP		**Beskedfordeler** afleverer returbeskeder efterhånden, som de modtagne hændelsesbeskeder behandles. Returbeskeder afleveres som et XML dokument i selve svarbeskedens data (body), i formatet defineret i StandardRetur elementet i SagDokObjekt.xsd. Svaret er UTF-8 encoded. **AMQP properties**: CorrelationsID: identifikation af forespørgslen om afsendelsen af beskeden. Anvendes til identifikation af svaret. Der henvises til RabbitMQ RPC modellen (se figur 2).
StandardRetur	20 20 30 40 41 42 50	Ok TilladtModtager felt på beskedkuverten ændret grundet manglende serviceaftale. Ny værdi: [<cvr-nr>, <cvr-nr>, etc.] Afsendersystemet er ikke aktivt Forespørgslen har forkert struktur Ikke autoriseret Ugyldig beskedkuvertversion Uventet server fejl

6. Teknisk beskrivelse

Servicen er implementeret som en AMQP (Advanced Message Queuing Protocol) version 0-9-1 service. AMQP er en åben standard applikationslag protokol til besked forsendelse.

6.1.Sikkerhed

Servicen er sikret via TLS version 1.2 samt SAML tokens, der hentes via Støttesystemet **Adgangsstyring for systemer** services. Der medsendes et token ved etablering af forbindelse til Beskedfordeleren der viser at systemet har rettigheder til at etablere forbindelsen. Forbindelsen kan eksistere så længe dette token er gyldigt. Endvidere sendes et token pr besked der afsendes over snitfladen. Sikkerhedstokenet anvendes til at validere at afsenderen har rettighed til at sende netop denne besked.

7. SecurityPolicy

Der anvendes i Beskedfordeleren's snitflader en blanding af Tokensikkerhed og certifikatsikkerhed. Denne service er af type "**Anden fælleskommunal service**", hvor Servicen autentificerer anvendersystemet dels via det modtagne security token, dels ved at tjekke at kaldet er signeret med et OCES certifikat, angivet i security tokenet (såkaldt holder-of-key).

7.1. Certifikater

Afsendersystemet identificeres i **Beskedfordeler** via det medsendte anvender-certifikat, som er et OCES 2 funktionscertifikat. Certifikatet afsendes ved etableringen af den TLS sikrede forbindelse. Der **skal** anvendes samme certifikat som ved anmodning om sikkerhedstoken fra støttesystemet **Adgangsstyring for systemer**.

For at kunne afsende beskeder til **Beskedfordeler** er det vigtigt at afsendersystemet truster **Beskedfordelers** funktionscertifikat. Hvilket certifikat der her er tale om afhænger af det miljø som anvendersystemet befinder sig i. Der findes separate certifikater til miljøerne Ekstern Test og Produktion. De nyeste certifikater til begge miljøer kan til enhver tid findes på Digitaliseringskataloget

7.2. Autorisation

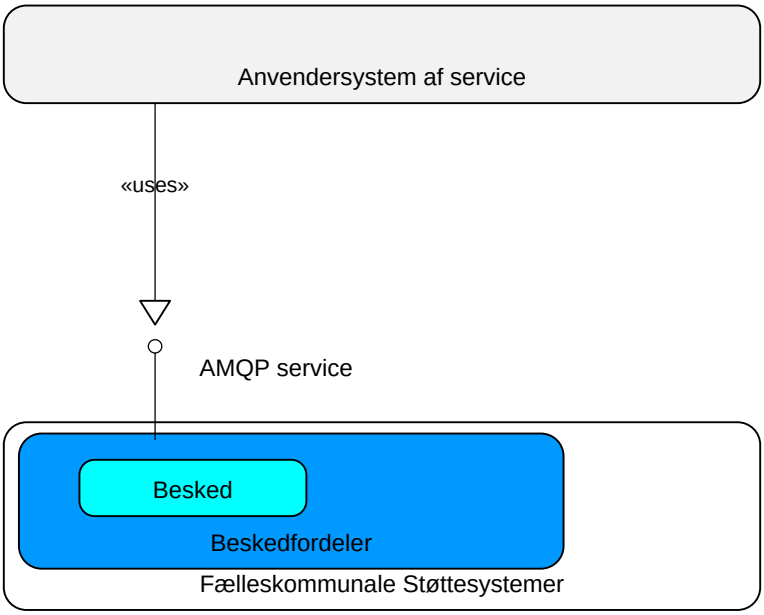
Afsendersystemet autoriseres til at kalde snitfladen via en serviceaftale i Støttesystemernes **Administrationsmodul**. Mulighederne for dataafgrænsninger i anvendelsen af snitfladen er beskrevet i **Fejl! Henvisningskilde ikke fundet.**

7.3. Skemavalidering

Der foretages en skemavalidering af XML for overholdelse af bl.a. version, format og gyldighed af tags i henhold til den givne version af **Beskedkuverten** og beskedtypen.

8. Kommunikation

Snitfladen er et AMQP service kald, der udføres direkte ved kald til det fælleskommunale støttesystem **Beskedfordeler**. Ved kald til AMQP servicen BeskedAfsend skal Anvendersystemet for at publicere beskeder, forbinde sig til Beskedfordelers afsend kø: RPC_AFSEND_BESKED_Q



*Figur 3 Kaldet til Beskedfordeler-Besked afsend-snitfladen er et AMQP kald, som udføres direkte til støttesystemet **Beskedfordeler**.*

8.1. Håndtering af utilgængelighed i snitfladen

Servicen er sikret mod nedbrud. Som for alle opdaterede services vil det kunne ske, at data opdateres uden klienten modtager et svar. Det kan f.eks. skyldes netværksfejl. Ved nedbrud skal de forsendelser der ikke er modtaget svar på gensendes.

8.2. Indlejring af tokens og sikkerhedspolitik i AMQP

Det udpakkede Token skal indlejres i AMQP via SASL i hver kald til Beskedfordeler, via SASL metoden EXTERNAL. Sikkerhedstoken fremskaffes via snitfladen til Støttesystemet Sikkerhed for Systemer. Det afsendende system skal opsætte sin AMQP klient til at medlevere det indhentede sikkerhedstoken ved at konfigurere AMQP klienten hertil, og hertil skal AMQP klientens API dokumentation konsulteres.
Et eksempel på anvendelse af sikkerhedstoken via SASL i en AMQP forbindelse med ses her i Java kode:

```
public class TokenConnectionFactory {
   public static ConnectionFactory getConnectionFactory(SSLContext sslContext, String token)
{
      final ConnectionFactory connectionFactory = new ConnectionFactory();
      connectionFactory.useSslProtocol(sslContext);
      connectionFactory.setSaslConfig(new TokenSaslConfig(token));
      return connectionFactory;
   }

   private static class TokenSaslConfig implements SaslConfig, SaslMechanism {
      final String token;

      public TokenSaslConfig(final String token) {
         this.token = token;
      }

      @Override
      public SaslMechanism getSaslMechanism(final String[] mechanisms) {
         assert Arrays.asList(mechanisms).contains(this.getName());
         return this;
      }

      @Override
      public String getName() {
         return "EXTERNAL";
      }

      @Override
      public LongString handleChallenge(final LongString challenge, final String username, final
String password) {
         assert challenge == null;
         return LongStringHelper.asLongString(token);
      }
   }
}

{
   ConnectionFactory factory = new ConnectionFactory();
   ...
   factory.setSaslConfig(new TokenSaslConfig(decodedToken));
}
```

8.3.Angivelse af transaktionsId i snitfladen

Beskedkuverten indeholder flere ID'er der identificere forskellige dele af den komplekse hændelse som beskeden udgør. Ved generering af beskeden vil det transaktionsgenererende system, Afsendersystemet udfylde "Haendelsesbesked.BeskedId" som udgør et Transaktions ID. Beskedfordeler påsætter ved modtagelsen af beskeden der ud over et "Leveranceinformation.TransaktionsId" på beskedkuverten.

Endvidere skal Afsendersystemet for hver enkelt forsøg på at afsende en besked via snitfladen udfylde AMQP property messageID med et for afsendelsen af beskeden unikt ID.

Et Modtagersystem kan således altid være sikker på at en besked er unik. Modtagersystemet kan via Haendelsestbesked.BeskedId registrere den samlede transaktion for beskeden.

Modtagersystemet kan også via Leveranceinformation.TransaktionsId registrere transaktionen for aflevering af beskeden.

9. Appendiks 1 – eksempler

Dette appendiks indeholder simple eksempler på, hvordan input og outputstrukturer kan se ud.

9.1. Inputstruktur – Besked_Afsend

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<ns2:Haendelsesbesked xmlns="urn:oio:sagdok:3.0.0" xmlns:ns2="urn:oio:besked:kuvert:1.0"
xmlns:ns3="http://www.w3.org/2000/09/xmldsig#">
    <ns2:BeskedId>
        <UUIDIdentifikator>20000000-0000-0000-0000-000000000000</UUIDIdentifikator>
    </ns2:BeskedId>
    <ns2:BeskedVersion>1.0</ns2:BeskedVersion>
    <ns2:Beskedkuvert>
        <ns2:Filtreringsdata>
            <ns2:Beskedtype>
                <UUIDIdentifikator>00000000-1111-0000-0000-000000000000</UUIDIdentifika-
tor>
            </ns2:Beskedtype>
        </ns2:Filtreringsdata>
        <ns2:Leveranceinformation>
            <ns2:TransaktionsId>
                <UUIDIdentifikator></UUIDIdentifikator>
            </ns2:TransaktionsId>
        </ns2:Leveranceinformation>
    </ns2:Beskedkuvert>
    <ns2:Beskeddata>
        <lol:dfd xmlns:lol="http://lol.com" xmlns:bfr="urn:oio:sts:beskedfordeler:1.0.0"
xmlns:ns4="urn:oio:sts:beskedfordeler:vaerdiliste:1.0.0" xmlns:soapenv="http://sche-
mas.xmlsoap.org/soap/envelope/" xmlns:urn1="urn:oio:besked:kuvert:1.0"
xmlns:urn2="urn:oio:sagdok:3.0.0" xmlns:xd="http://www.w3.org/2000/09/xmldsig#">
            TEST MESSAGE
        </lol:dfd>
    </ns2:Beskeddata>
</ns2:Haendelsesbesked>
```

9.2. Beriget Inputstruktur – Besked_Afsend

```
 <ns2:Haendelsesbesked xmlns="urn:oio:sagdok:3.0.0" xmlns:ns2="urn:oio:besked:kuvert:1.0"
xmlns:ns3="http://www.w3.org/2000/09/xmldsig#">
    <ns2:BeskedId>
        <UUIDIdentifikator>c8693551-981e-4be1-b1a6-180cf8fad1f0</UUIDIdentifikator>
    </ns2:BeskedId>
    <ns2:BeskedVersion>1.0</ns2:BeskedVersion>
    <ns2:Beskedkuvert>
        <ns2:Filtreringsdata>
            <ns2:Beskedtype>
                <UUIDIdentifikator>00000000-1111-0000-0000-000000000000</UUIDIdentifika-
tor>
            </ns2:Beskedtype>
            <ns2:TilladtModtager>
                <URNIdentifikator>12345678</URNIdentifikator>
            </ns2:TilladtModtager>
            <ns2:ObjektRegistrering>
                <ns2:ObjektAnsvarligMyndighed>
                    <URNIdentifikator>12345678</URNIdentifikator>
                </ns2:ObjektAnsvarligMyndighed>
                <ns2:ObjektType>
                    <UUIDIdentifikator>156e8146-e5ac-46ce-a3af-f20316763055</UUIDIdentif-
ikator>
                </ns2:ObjektType>
```

```
            </ns2:ObjektRegistrering>
            <ns2:ObjektRegistrering>
                <ns2:ObjektType>
                    <UUIDIdentifikator>b764c528-8198-4afe-b023-8c1b1350458e</UUIDIdentif-
ikator>
                </ns2:ObjektType>
                <ns2:OpgaveEmne>
                    <URNIdentifikator>URN:OIO:KLE:54.33.11</URNIdentifikator>
                </ns2:OpgaveEmne>
            </ns2:ObjektRegistrering>
        </ns2:Filtreringsdata>
        <ns2:Leveranceinformation>
            <ns2:TransaktionsId>
                <UUIDIdentifikator></UUIDIdentifikator>
            </ns2:TransaktionsId>
        </ns2:Leveranceinformation>
    </ns2:Beskedkuvert>
    <ns2:Beskeddata xmlns:ns4="urn:oio:sts:beskedfordeler:vaerdiliste:1.0.0">
        <lol:dfd xmlns:lol="http://lol.com" xmlns:bfr="urn:oio:sts:beskedfordeler:1.0.0"
xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:urn1="urn:oio:besked:ku-
vert:1.0" xmlns:urn2="urn:oio:sagdok:3.0.0" xmlns:xd="http://www.w3.org/2000/09/xmld-
sig#">
            TEST MESSAGE
        </lol:dfd>
    </ns2:Beskeddata>
</ns2:Haendelsesbesked>
</Besked></AfsendBeskedInput>
```

9.3.Outputstruktur - Besked Afsend

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<StandardRetur xmlns="urn:oio:sagdok:3.0.0" xmlns:ns2="urn:oio:besked:kuvert:1.0"
xmlns:ns3="http://www.w3.org/2000/09/xmldsig#">
    <StatusKode>20</StatusKode>
    <FejlbeskedTekst>OK</FejlbeskedTekst>
</StandardRetur>
```

10. Appendiks 2 – fejlsøgning

Dette appendiks indeholder samlet oversigt over fejlkoder inkl. forslag til, hvad der kan have forårsaget en given fejlkode.

Fejlkode	Fejltekst	Potentiel årsag
30	Afsendersystemet er ikke aktivt	Anvendersystemet er ikke aktivt i beskedfordeleren. Status ændres via beskedfordelerens brugergrænseflade.
40	Forespørgslen har forkert struktur	Beskeden der afsendes har en forkert struktur og kan ikke behandles eller der beskedtypens gyldighedsperiode er i modstrid aktuel besked. Løsningsforslag: • tilføje et namespace til test elementet. <fp:test xmlns=" http://www.testfirma.dk/testkuvert">Hello world</fp:test> • BeskedId er længere end de tilladte 36 tegn

41	Ikke autoriseret	Afsendersystemet kan ikke godkendes/er ikke autoriseret. Det kan skyldes:
		• Anvendersystemet autentificerer sig ikke med gyldigt certifikat
		• Det anvendte certifikat er ukorrekt og er ikke knyttet til et Anvendersystem der er kendt af Beskedfordeleren
		Forslag til fejlsøgning:
		• Kontroller at ordet "token" er skrevet med småt, som I eksemplet nedenfor:
		// Build header with token Dictionary<String, Object> headers = new Dictionary<String, Object>();
		headers.Add("token", theXmlToken);
		• Kontroller at der er overensstemmelse mellem listen over "tilladte modtagere" i beskeden og serviceaftalen. Beskeden afvises, hvis den indeholder tilladte modtagere, der ikke er i serviceaftalen
		• Kontroller at der ikke er overflødige blanktegn eller linjeskift i feltet med beskedId.

42	Skemaet til beskedversion findes ikke	Ikke muligt at identificere beskedkuvertens version Beskedkuvertversionen er ikke supporteret af en XSD validering
50	Uventet serverfejl	Uventet fejl opstod under behandlingen af beskeder til fejlkøen.
	Forbindelse kan ikke oprettes til beskedfordeleren	Kontroller at firewall/port er korrekt opsat

	Fault occurred while processing	Kald af service uden token
	Cannot read security of the token	Kald af service med et forkert token (eks. Token til et andet system end det system man vil kalde)
	"ACCESS_REFUSED - Login was refused using authentication mechanism EXTERNAL. For details see the broker logfile"	Kontroller, at der bruges den korrekte type certifikat. Dvs testcertifikat i ekstern test og produktionscertifikat i produktion Hvis i tvivl kan leverandøren af Beskedfordeleren undersøge ved hjælp af denne streng "EXTERNAL login refused: no peer certificate"