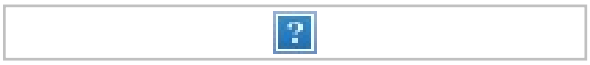
API-flyt for synkrone signeringsoppdrag

Dette integrasjonsmønsteret vil passe for større tjenesteeiere som har egne portaler og nettløsninger, og som ønsker å tilby signering sømløst som en del av en prosess der brukeren allerede er innlogget i en sesjon på tjenesteeiers nettsider. Signeringsprosessen vil oppleves som en integrert del av brukerflyten på tjenesteiers sider, og brukeren blir derfor sendt tilbake til tjenesteeiers nettsider etter at signeringen er gjennomført.

Relevante typer for denne delen av APIet finnes i filen `schema/xsd/direct.xsd`.



Flytskjema for det synkrone scenariet: *skjemaet viser flyten helt fra en bruker logger inn på tjenesteeiers nettsider til oppdraget er fullstendig signert. Heltrukne linjer viser brukerflyt, mens stiplede linjer viser API-kall*

Steg 1: Opprette signeringsoppdraget

Flyten begynner ved at tjenesteeier gjør et bak-kanal-kall mot APIene for å opprette signeringsoppdraget. Dette kallet gjøres som en multipart-request, der den ene delen er dokumentpakken og den andre delen er metadata.

- Kallet gjøres som en `HTTP POST` mot ressursen `<rot-URL>/direct/signature-jobs`
- Dokumentpakken legges med multipart-kallet med mediatypen `application/octet-stream`. Se forrige kapittel for mer informasjon om dokumentpakken.
- Metadataene som skal sendes med i dette kallet er definert av elementet `direct-signature-job-request`. Disse legges med multipart-kallet med mediatypen `application/xml`.

En del av metadataene er et sett med URLer definert i elementet `exit-urls`. Disse adressene vil bli benyttet av signeringstjenesten til å redirecte brukeren tilbake til din portal ved fullført signering. Følgende tre URLer skal oppgis:

- completion-url: Hit sendes brukeren dersom signeringen er vellykket. Du kan da be om status for å få URLer til nedlasting av signert dokument.
- rejection-url: Hit sendes brukeren dersom hun selv velger å avbryte signeringen. Dette er en handling brukeren *selv valgte* å gjennomføre.
- error-url: Hit sendes brukeren dersom det skjer noe galt under signeringen. Dette er noe brukeren *ikke* valgte å gjøre selv.

Følgende er et eksempel på metadata for et signeringsoppdrag:

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<direct-signature-job-request xmlns="http://signering.posten.no/schema/v1">
    <reference>123-ABC</reference>
    <exit-urls>
        <completion-url>https://www.sender.org/completed</completion-url>
        <rejection-url>https://www.sender.org/rejected</rejection-url>
        <error-url>https://www.sender.org/failed</error-url>
    </exit-urls>
    <polling-queue>custom-queue</polling-queue>
</direct-signature-job-request>
```

Følgende er et eksempel på `manifest.xml` fra dokumentpakken:

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<direct-signature-job-manifest xmlns="http://signering.posten.no/schema/v1">
    <signer>
        <personal-identification-number>12345678910</personal-identification-number
        <signature-type>ADVANCED_ELECTRONIC_SIGNATURE</signature-type>
        <on-behalf-of>SELF</on-behalf-of>
    </signer>
    <sender>
        <organization-number>123456789</organization-number>
    </sender>
    <document href="document.pdf" mime="application/pdf">
        <title>Tittel</title>
        <description>Melding til undertegner</description>
    </document>
    <required-authentication>3</required-authentication>
    <identifier-in-signed-documents>PERSONAL_IDENTIFICATION_NUMBER_AND_NAME</identi
</direct-signature-job-manifest>
```

Undertegner

I dette eksempelet er fødselsnummer (`personal-identification-number`) brukt for å identifisere undertegner. Man kan også benytte en selvvalgt identifikator, som eksemplifisert i `/schema/examples/direct/manifest-signer-without-pin.xml` .

Merk at `signature-type` spesifiseres per undertegner, hvilket vil si at det i praksis er mulig å innhente ulike typer signaturer fra ulike undertegnere i et multiundertegner-case. Dette er imidlertid antatt å være et såpass sjeldent use-case at det ikke er mulig via grensesnittet i web-portalen – der spesifiseres signaturtype på jobbnivå.

For offentlige avsendere kan man for elementet `on-behalf-of` under `signer` sende inn verdien `OTHER` for å angi at man signerer på vegne av en tredjepart (f.eks. signering av anskaffelseskontrakt på vegne av arbeidsgiver). Elementet er valgfritt, og verdien `SELF`, altså signering på egne vegne, benyttes om man ikke angir noe. I første omgang benyttes denne verdien kun til å deaktivere videresending av signerte dokumenter til digital postkasse; signerer man på vegne av noen andre (`OTHER`) vil videresending deaktiveres. Videresending er altså aktivert som standard.

Andre attributter

Sikkerhetsnivå (`required-authentication`) spesifiseres på jobbnivå ettersom dette også er knyttet til dokumentets sensitivitetsnivå.

`identifier-in-signed-documents` brukes for å angi hvordan undertegneren(e) skal identifiseres i de signerte dokumentene.
Tillatte verdier er `PERSONAL_IDENTIFICATION_NUMBER_AND_NAME`, `DATE_OF_BIRTH_AND_NAME` og `NAME`, men ikke alle er gyldige for alle typer signeringsoppdrag og avsendere.
Disse begrensningene og standardverdier er beskrevet i den funksjonelle dokumentasjonen.

Som respons på dette kallet vil man få en respons definert av elementet `direct-signature-job-response`.

- Denne responsen inneholder en URL (`redirect-url`) som man redirecter brukerens nettleser til for å starte signeringsseremonien.
- I tillegg inneholder den en URL du benytter for å spørre om status på oppdraget. Her skal man IKKE benytte seg av polling, man skal derimot vente til brukeren returneres til en av URLene definert i requesten, for deretter å gjøre kallet. For å forhindre polling kreves det et token som du får tilbake ved redirecten, se Steg 3 for nærmere forklaring.

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<direct-signature-job-response xmlns="http://signering.posten.no/schema/v1">
    <signature-job-id>1</signature-job-id>
    <redirect-url>
        https://signering.posten.no#/redirect/421e7ac38da1f81150cfae8a053cef62f9e74
    </redirect-url>
    <status-url>https://api.signering.posten.no/api/{sender-identifier}/direct/sign
</direct-signature-job-response>
```

Steg 2: Signeringsseremonien

Hele dette steget gjennomføres i signeringsportalen. Du redirecter brukeren til portalen ved å benytte URLen du får som svar på opprettelsen av oppdraget. Denne linken inneholder et engangstoken generert av signeringstjenesten, og det er dette tokenet som gjør at brukeren får tilgang til å lese dokumentet og gjennomføre signeringen.

Sikkerhet i forbindelse med engangstoken:
For å håndtere sikkerheten i dette kallet vil tokenet kun fungere én gang. Brukeren vil få en cookie av signeringstjenesten ved første kall, slik at en eventuell refresh ikke stopper flyten, men du kan ikke bruke denne URLen på et senere tidspunkt. Årsaken til at vi kun tillater at den brukes én gang er at URLer kan fremkomme i eventuelle mellomtjeneres logger, og de vil dermed ikke være sikre etter å ha blitt benyttet første gang.

Brukeren gjennomfører signeringsseremonien, og blir deretter sendt tilbake til din portal via URLen spesifisert av deg i `completion-url`. På slutten av denne URLen vil det legges på et query-parameter (`status_query_token`) du senere skal benytte når du spør om status.

Steg 3: Hent status

Når brukeren blir sendt tilbake til din portal skal du gjøre et bak-kanal-kall (`HTTP GET`) for å hente ned status. Dette gjøres ved å benytte `status-url` du fikk i steg 1, pluss query-parameter (`status_query_token`) du fikk i steg 2.

Hvis signeringsoppdraget er lagt på en spesifikk kø, så må query-parameteret `polling_queue` settes til navnet på køen. Dette er kun relevant når `status-retrieval-method` er satt til `POLLING`.

Du skal ikke sende med noen andre data i dette kallet.

Responsen fra dette kallet er definert gjennom elementet `direct-signature-job-status-response`. Et eksempel på denne responsen ved et suksessfullt signeringsoppdrag vises under:

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<direct-signature-job-status-response xmlns="http://signering.posten.no/schema/v1">
    <signature-job-id>1</signature-job-id>
    <signature-job-status>COMPLETED_SUCCESSFULLY</signature-job-status>
    <status since="2017-01-23T12:51:43+01:00">SIGNED</status>
    <confirmation-url>https://api.signering.posten.no/api/{sender-identifier}/direc
    <xades-url>https://api.signering.posten.no/api/{sender-identifier}/direct/signa
    <pades-url>https://api.signering.posten.no/api/{sender-identifier}/direct/signa
</direct-signature-job-status-response>
```

Steg 4: Laste ned PAdES eller XAdES

I forrige steg fikk du to lenker: `xades-url` og `pades-url`. Disse kan du gjøre en `HTTP GET` på for å laste ned det signerte dokumentet i de to formatene.

XAdES er et format som brukes til å styrke og standardisere signaturene som kommer fra e-ID-leverandørene. Formatet har støtte for langtidsvalidering, og gjør samtidig at man får ett format å forholde seg til, uavhengig av hvilken e-ID-leverandør som er brukt til signering.

PAdES er et signaturformat som inneholder originaldokumentet, alle signaturer og all informasjon som er nødvendig for å validere signaturen. Formatet er spesifisert av ETSI, og bygger på PDF. En unik egenskap med PAdES er at dokumentet kan åpnes i en vilkårlig PDF-leser. Adobe Reader (og eventuelle andre avanserte PDF lesere) vil også kunne vise frem deler av valideringsinformasjonen slik at sluttbrukeren selv kan se at dokumentet er gyldig signert. I tillegg ligger også XAdES-dokumentet vedlagt denne PDFen.

Steg 5: Bekrefte ferdig prosessering

Til slutt gjør du et `HTTP POST`-kall mot `confirmation-url` for å bekrefte at du har prosessert jobben ferdig. Hvis langtidslagring benyttes vil dette markere oppdraget som ferdig og lagret. I motsatt fall vil oppdraget slettes i signeringsportalen.