TDialog – arkitektur

Inledning

Detta dokument beskriver TDialogs arkitektur och övergripande hur TDialog fungerar.

Grundtankar

All känslig information finns i applikationen

När en användare ”skickar” ett meddelande med TDialog stannar meddelandet i applikationen. Snarare än att skicka något tillgängliggörs meddelandet helt enkelt för en mottagare. Rent tekniskt kan man snarare se ”skicka”-funktionen som en behörighetstilldelning. Det gör att all information stannar i applikationen och bara tillgängliggörs för en ny användare.

(Observera dock att i SDK- respektive Mina meddelanden-funktionen skickas meddelanden till mottagaren respektive brevlådeoperatörens infrastruktur.)

Federerad autentisering och behörighetstilldelning

Information i TDialog är normalt sett av en känslighetsgrad att åtkomst till informationen skall föregås av stark autentisering och så fort en autentisering innebär ”något man har” eller ”något man är” snarare än bara ett lösenord behöver den vara anpassad till organisationen för att bli användarvänlig. För vissa organisationer är BankID rätt väg att gå, för andra är det egenutgivna kort eller SITHS, för åter andra är det en mobilapp etc. För att kunna stödja den mängd interna autentiseringsmetoder som finns på marknaden har TDialog federerad autentisering, dvs kunden kan använda vilka metoder som helst så länge de levereras av en identitetsutfärdare (IDP med stöd för SAML v2).

Dessutom kan den federerade autentiseringen användas för att dela information med andra organisationer.

Se nedan för mer information om TDialogs federerade autentisering och behörighetshantering.

Programvara där all information lagras hos kunden

TDialog är byggt för att hantera känslig information. Det finns idag stora osäkerheter knutet till hantering av känslig information i molntjänster, och TDialog är därför en programvara snarare än en molntjänst. Med TDialog blir det enkelt att hantera informationen på ett tillräckligt säkert, samtidigt som informationen hanteras och behandlas hos den som äger informationen snarare än hos tredje part.

Termer

Administratör

En Administratör i TDialog kan utföra uppgifter som att söka/ta bort användare, söka meddelanden och få fram avsändare/mottagare samt sätta vissa globala inställningar som

krypteringsnycklar, och automatrensningsintervall. En administratör har dock inte möjlighet att se andra användares meddelanden eller att ge sig själv sådana behörigheter.

Extern organisation/användare

En organisation/användare som inte äger TDialog-installationen, respektive individer knutna till den organisationen. Kan delas in i betrodda organisationer/användare respektive gästanvändare.

Betrodd organisation/användare

En organisation som inte använder TDialog, men som är betrodd av den interna organisationen. Dessa kan autentisera sig med sina egna autentiseringsmetoder

Gästanvändare

Individer som loggar in som gäster i TDialog. Vanligen loggar de in med BankID/Freja, men andra möjliga kombinationer är Google- eller Microsoft-konto med tvåstegsverifiering.

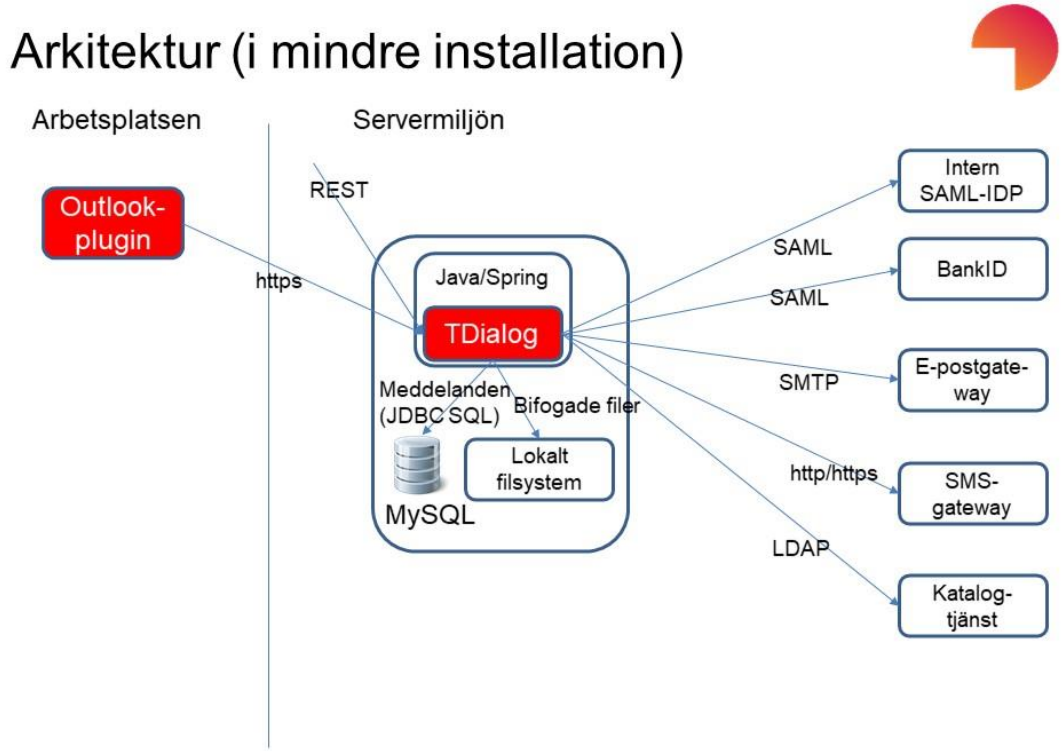
Funktionsbrevlådor

Förutom att kunna kommunicera som sig själv som individ kan en intern eller en betrodd användare ha behörighet till funktionsbrevlådor, dvs en annan inkorg och utkorg som potentiellt kan delas med andra användare av samma typ.

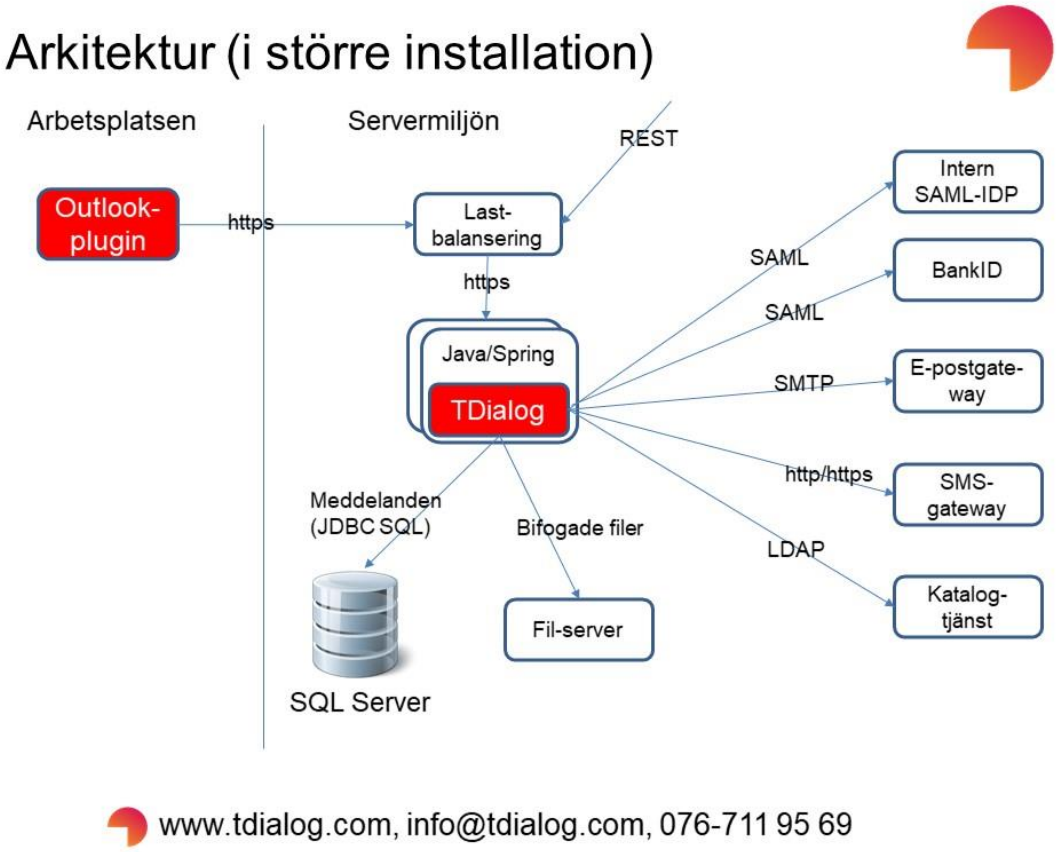
Intern organisation/användare

Den organisation som äger TDialog-installationen, respektive individer knutna till den organisationen.

Grundarkitektur

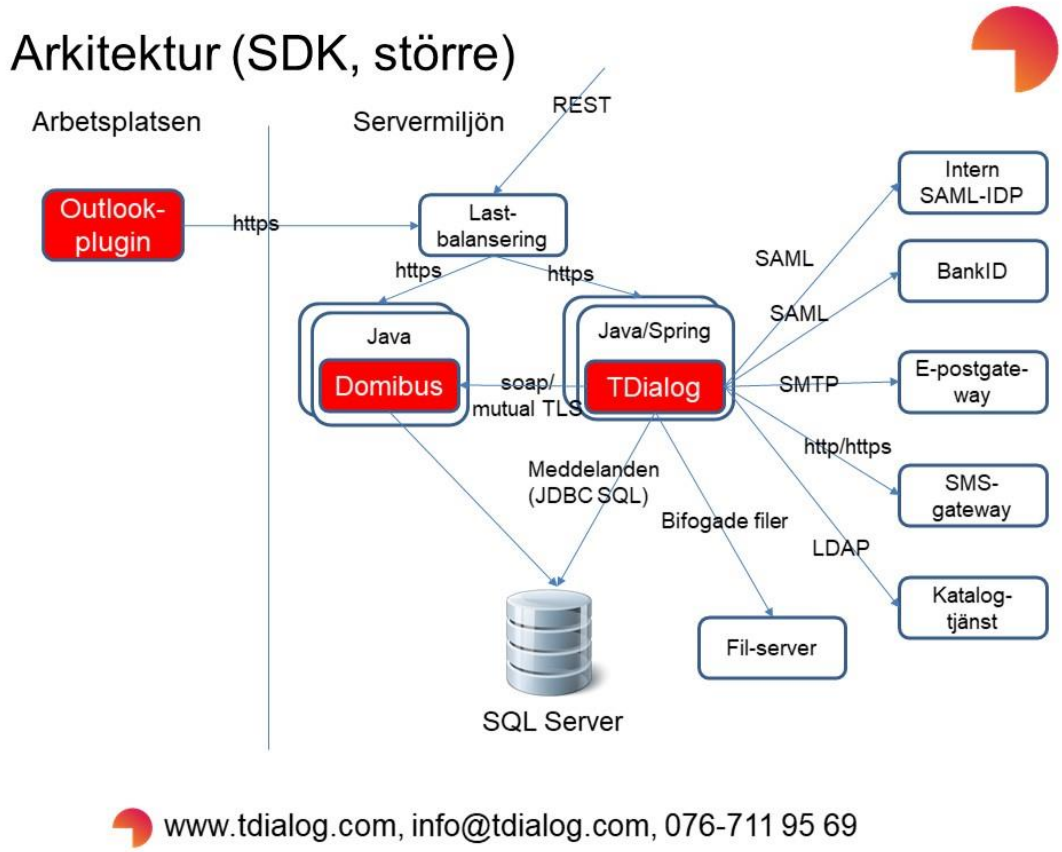


Som synes i skissen är TDialog en Java serverapplikation som använder sig av Spring (primärt Spring Security). Denna ansluter med JDBC till en databas (bland annat för meddelandeinformation) och skriver till ett filsystem (för konfiguration och meddelandebilagor). Vidare kan den ansluta till en mängd andra komponenter för notifieringar, adresslistor autentisering etc.



Bilden ovan visar hur en lastbalanserad arkitektur med extern databas ser ut. Här skalar man genom duplicering upp Java/Spring-miljöerna. Dels ger detta redundans i applikationen och dels ger det faktiskt god ökad prestanda eftersom de kostsamma krypterings-/dekrypteringsfunktionerna som görs varje gång ett meddelande skickas kan fördelas på flera applikationsservrar. Databasservern är i sig inte duplicerad, men kan naturligtvis vara ett SQLServer-kluster.

Arkitektur – SDK



Bilden ovan visar en stor installation med stöd för SDK (Ineras ramverk Säker digital kommunikation). Notera den, potentiellt lastbalanserade, extra javakomponenen med accesspunktsprogramvaran Domibus.

Säkerhetsfunktioner

Autentisering

Som nämns ovan nyttjar TDialog federerad autentisering. Det innebär i korthet att den SAML-IDP autentiserar användaren och skickar ett signerat intyg på godkänd autentisering. Intyget innehåller ett användar-ID och potentiellt även andra attribut som kan användas för auktorisation. Vad som tjänar som användar-ID är konfigurerbart.

TDialog har stöd för autentisering med flera IDP:er, både för intern och extern autentisering. Man kan alltså exempelvis sätta upp TDialog för att autentisera sig antingen mot en BankID-tjänst eller mot Ineras SITHS-tjänst för intern autentisering och antingen BankID eller Freja eID för extern autentisering. Dessutom kan varje betrodd organisation ha en egen IDP.

Auktorisation

Användar-ID:t/identifieraren för en korrekt autentiserad användare används som intern identifierare vid auktorisationen, medan användarens e-postadress används för identifikation av användaren när ett meddelande ska skickas, dvs användaren får behörighet till ny information.

En grundläggande del av auktorisationen är huruvida en användare är extern eller intern (se ”Termer” ovan). En grundläggande princip är att en intern användare alltid måste vara antingen avsändare eller mottagare av meddelandet, dvs en extern användare kan aldrig kommunicera med en annan extern användare.

Nedan beskrivs systemets roller, vilka övergripande behörigheter de har och hur TDialog utför auktorisation av dessa roller. Generellt kan sägas att auktorisation sker med hjälp av attribut vid inloggningen.

- **Gästanvändare** En gästanvändare kan aldrig kommunicera med en annan extern användare. I övrigt är deras behörigheter konfigurerbara. Kan de skicka till vilka interna användare som helst eller bara svara på meddelanden? Kan de självregistrera för att skapa sig ett konto, eller måste de bli inbjudna? En användare auktoriseras som en gästanvändare om den inte har en intern eller betrodd adress och inte loggar in med en intern eller betrodd IDP.
- **Betrodd användare** En användare från en betrodd organisation. De autentiseras genom organisationens IDP och har organisationens domännamn i sin e-postadress.
- **Intern användare** En intern användare har fulla behörigheter till sin egen information och kan kommunicera med vem den vill i lösningen.
- **Funktionsbrevlådor** Auktorisation till funktionsbrevlådor sker med hjälp av inloggningsattribut. En intern användare som har ett inloggningsattribut som säger att den har behörighet till en intern funktionsbrevlåda kommer beredas tillgång till denna funktionsbrevlåda, på motsvarande sätt kan en betrodd användare ha ett inloggningsattribut som medger behörighet till en betrodd funktionsbrevlåda. I båda fallen gäller att en funktionsbrevlåda som inte redan finns skapas upp automatiskt av systemet. Behörigheten till funktionsbrevlådan lagras inte av TDialog, utan vid varje inloggning bestämmer inloggningsattributen vilken/vilka funktionsbrevlådor användaren har tillgång till, och om åtkomst tagits bort (typisk i organisationens katalogtjänst) kommer användaren omedelbart att utestängas från funktionsbrevlådan.
- **Administratör** Liksom med funktionsbrevlådor sätts administratörsbehörigheten av inloggningsattibut, och liksom med funktionsbrevlådor kontrollera behörigheten vid varje inloggning.

Spårbarhet

Så fort en användare, oavsett typ, gör något i TDialog kommer det att loggar i en audit-logg. Tidpunkt, vem den inloggade användaren var och eventuell funktionsbrevlåda som användes loggas alltid, i övrigt är det olika logginformation beroende på vad som utfördes. Den enda information om meddeladet som loggas är avsändare och mottagare, dvs det finns inte känslig

meddelandeinformation i loggarna. Om man däremot kommunicerar med gästanvändare som loggar in med BankID kommer det finnas personnummer i loggarna.

Loggning utförs enligt javastandarden log4j och kan konfigureras enligt denna, dvs det finns stora möjligheter att logga till andra system, specifik loggrotation etc. Loggning påverkas inte av radering eller automatrensning av information.

Skydd av information i rörelse och vila

All kommunikation till och från TDialog, oavsett om det är via en webbläsare eller ett maskinellt REST-gränssnitt, sker över https, och TDialog underhåller en uppsättning ciphers som vid var tid anses tillräckligt säkra för att användas för https-kommunikation.

I vila krypteras meddelandeinformationen med en nyckel som interna organisationen själv sätter. Eftersom lösningen är en programvara på plats finns ingen möjlighet för någon annan aktör att komma över krypteringsinformationen. Interna organisationen kan välja att antingen spara krypteringslösenordet på servern eller att skriva in det varje gång. Sparas krypteringslösenordet på servern ligger det persistent och krypterat, men om en angripare som tagit kontroll över hela lösningen finns risk att den kan dekryptera informationen. Skrivs lösenordet in varje gång sparas det endast i minnet och angriparen kan inte dekryptera, men i gengäld måste lösenordet skrivas in vid varje omstart för att kunna kryptera ny information och dekryptera den befintliga.

Det bästa skyddet av information är att endast spara det som är nödvändigt. TDialog har en funktion för automatrensning av gammalt material, och TDialog rekommenderar att kunder använder den. Kunden sätter helt enkelt en rensningstid för meddelanden (tre månader, ett år eller vad det nu kan vara) och när ett meddelande nått den åldern rensas det bort. Syftet med TDialog är att föra informationen från punkt A till punkt B, inte att vara långtidslagring för information.

Integrationsmöjligheter

TDialog har ett REST-API för skapande och skickande av meddelanden, användarhantering mm. Autentiseringen i REST-API:et sker med JWT-tokens. Se separat dokumentation om REST-API.

Vidare har TDialog även ett native-API för att hantera triggers i systemet, exempelvis när ett meddelande skapas, läses eller är på väg att skickas. Dessa triggers kan sedan i sin tur exekvera kod, exempelvis för automatisk arkivering.

TDialog har även ett gränssnitt för integration via e-post. Det innebär att ett inkommet meddelande (krypterat och från verifierad källa) kan användas för att skapa meddelanden i TDialog. TDialog-gränssnittet kan även användas för att skapa krypterad e-post.