Lokal IdP

Tidigare versioner

Innehållsförteckning

Revisionshistorik

Version	Datum	Aktör	Kommentar
0.1	26 Oct 2023	Ehlert, Stefan	• Kopierad från IdP 2.4
0.2	27 Oct 2023	Ehlert, Stefan	▪ Lagt till information kring IdP Administration

Nytt i denna versionen

Ändringar sedan senaste lokala versionen (2.4):

Se 2.5 Release notes - IdP

Checklista inför driftsättning av lokal IdP

En delmängd av de saker som behöver göras inför driftsättning av lokal IdP:

1. Teckna användaravtal med Inera för åtkomst att ladda ner applikationerna
2. Påbörja anslutningsförfarandet mot HSA i god tid innan planerad driftsättning av IdP
3. Se över klusteruppsättning (egna burkar eller virtuella miljöer)
4. Installera/paketera Java 11
5. Sätt upp MongoDB med säkerhetskopiering
6. Sätt upp Redis
7. Sätt upp lastbalanserare
8. Se över portöppningar
9. Certifikat för åtkomst till HSA, förmodligen ett SITHS-utfärdat funktionscertifikat
10. Certifikat för TLS-terminering
11. Certifikat för SAML- och OIDC-meddelandesignering och -kryptering
12. Fastställ behörighetsregler för administrationsgränssnittet
13. Fastställ tolkning av Tillitsnivå (LoA) för olika typer av användarcertifikat

Plattform och tredjepartsprodukter

Plattform

Lokal IdP levereras som en zip-fil med en filstruktur innehållandes konfigurationsfiler tillsammans med två så kallade "fat jarer", d.v.s. .jar-filer som innehåller applikationerna samt webservrar och alla applikationernas kodberoenden. Den ena .jar-filen är själva IdP Application och innehåller själva autentiseringsapplikationen samt det publika användargränssnittet för autentiserings- och signeringsflödet. Den andra .jar-filen är IdP Administration som innehåller administrationsgränssnittet för IdP:n.

Jar-filerna kan köras rakt upp och ner på egna servrar, köras i virtuella maskiner eller paketeras i t.ex. en docker-container och hanteras via en container-orkestreringsplattform. De nationella instanserna av Inera IdP paketeras t.ex. i docker-containers baserade på en enkel RHEL-image med Java 11 installerat och driftsätts sedan m.h.a. OpenShift.

Java

Java 11 krävs för att starta applikationerna. OpenJDK rekommenderas, men även Oracle JDK/JRE bör fungera.

Databaser

IdP:n använder sig av MongoDB och Redis.

Redis-databasen håller enbart temporär lagring (cache, sessioner, et.c.) och behöver således inte säkerhetskopieras.

I MongoDB lagras persistent data (certifikat, klientmetadata, et.c.) och den bör därför säkerhetskopieras regelbundet.

Installation och konfiguration av databaserna ligger utanför scopet för detta dokument.

Följande versioner av databaserna har testats med IdP:

Databas	Version
MongoDB	4.4.5
Redis	5.0.3

MongoDB

IdP:n **kräver** att MongoDB är uppsatt som ett replica set (för att transaktioner ska fungera). Se MongoDB's dokumention för hur man skapar ett replica set. Huruvida det ligger en eller flera noder bakom replica set:et spelar ingen roll för IdP:ns del.

Applikationerna kräver även att det finns en databas och en användare skapad i MongoDB som den kan använda. För att skapa upp detta, anslut till MongoDB med klienten (*mongo*/*mongo.exe*) och ange följande kommandon:

mongo

```
idpdb = db.getSiblingDB("idp")
idpdb.createUser({ user: "idpuser", pwd: "idppassword", roles: [ "readWrite" ]})
quit()
```

Namnet på databasen (**idp** i exemplet ovan) samt användarnamnet och lösenordet (**idpuser** och **idppassword**) kan väljas valfritt, men måste stämma överrens med konfigurationen i *application-custom.properties* filerna för både IdP Application och IdP Administration.

IdP applikationerna kommer sedan att vid anslutning automatiskt skapa upp de kollektioner som den behöver.

Transportkryptering mot MongoDB

Ifall trafiken mellan IdP Application och/eller IdP Administration och MongoDB skall krypteras behöver följande inställning konfigureras:

```
#Lägg till ssl=true som query-parameter i mongodb.uri. T.ex:
spring.data.mongodb.uri=mongodb://user:password@mongodb-node1:27017,mongodb-node2:27017,mongodb-node3:27017,
mongodb-node4:27017/database?replicaSet=mongo-replica-set-name&ssl=true

mongo-ssl-ca-file=file:<sökväg till truststore innehållandes utfärdare av databasens certifikat>
```

Redis

Redis används av IdP som en gemensam cache. Alla IdP-noder behöver alltså anslutas till samma uppsättning av Redis.

IdP applikationerna kan ansluta till sentinel (kluster) eller singelnod av Redis. Redis saknar användare, men kan konfigureras för att kräva lösenord för att ansluta. IdP:n har stöd för båda alternativ. Använder man lösenord måste detta konfigureras i *application-custom.properties* för både IdP Application och IdP Administration.

Transportkryptering mot Redis

> TLS-funktionaliteten mot Redis är inte fullständigt testad och används inte (än) av Nationell IdP.

Redis stödjer TLS från och med version 6. Ifall trafiken mellan IdP Application och/eller IdP Administration och Redis skall krypteras behöver följande inställningar konfigureras:

```
spring.redis.ssl=true
lettuce.client.customizer.trust-store-file=<sökväg till truststore  innehållandes utfärdare av databasens
certifikat>
lettuce.client.customizer.trust-store-pwd=<lösenord för truststore ovan>
lettuce.client.customizer.disable-peer-verification=true
```

Lastbalanserare

IdP:ns applikationer är tänkt att köras med mer än en instans (klustrad). Det innebär att det behövs en extern lastbalanserare som fördelar lasten mellan noderna.

Routes och TLS-terminering

IdP Application går upp med två connectorer, en för TLS-trafik (som skall termineras i lastbalanseraren) och en för mTLS-trafik (som skall släppas igenom av lastbalanseraren och termineras i applikationen).

IdP Administrationen går upp med endast en connector för TLS-trafik (som skall termineras i lastbalanseraren).

Huvuddomän

Trafik mot huvuddomänerna SSL-termineras i lastbalanseraren.

Certifikat för domänerna installeras alltså i lastbalanseraren.

Subdomän för mTLS

Trafik mot subdomänerna (*secure*/secure*) (typ secure.idp.inera.se alt secure*0-9*.idp.inera.se, om idp.inera.se är huvuddomänen) skall släppas igenom till applikationen som själv sköter mTLS-termineringen. Nycklar för hantering av mTLS-termineringen läses in i via administrationsgränssnittet i IdP Administration.

Exempelkonfiguration av routes i LB

Givet följande konfiguration i IdP Application application-custom.properties:

IdP Application: application-custom.properties

```
...
idp.server.protocol=https
idp.server.host=idp.domain.test
idp.server.port=443
...
inera.common.server.mtls.port=8443
...
```

IdP Administration: application-custom.properties

```
...
idp.server.protocol=https
idp.server.host=admin.idp.domain.test
idp.server.port=443
...
```

så kommer applikationerna att innanför lastbalanseraren serva två portar: 8080 (default) samt 8443. Samtidigt är adresserna utåt https://idp.domain.test:443 och https://secure.idp.domain.test:443 och https://admin.idp.domain.test:443.

Följande konfiguration skulle då användas i lastbalanseraren:

Inkommande adress	målport hos applikationen	SSL-terminering i LB
https://idp.domain.test:443	8080	Ja
https://secure.idp.domain.test:443	8443	Nej (Passthrough)
https://admin.idp.domain.test:443	8080	Ja

Förslagsvis så redirectas också http-trafik (port 80) till https (port 443).

Headers

Lastbalanseraren måste skicka med följande headers till applikationen:

- X-Forwarded-Proto
- X-Forwarded-Host

- X-Forwarded-Port
- X-Forwarded-For

Certifikat

TLS-trafik

IdP Application går upp med två connectorer, en för okrypterad trafik (som skall termineras i lastbalanseraren) och en för mTLS-trafik (som skall släppas igenom orört av lastbalanseraren och termineras i applikationen).

- Certifikat och nyckel för IdP Applications huvuddomän (ex. idp.domain.test) läses in i lastbalanseraren och används för TLS terminering på all trafik mot huvuddomänen.
- Certifikat och nyckel för subdomänen *secure* (ex. secure.idp.domain.test om idp.domain.test är huvuddomänen) läses in i administrationsgränssnittet i IdP Administration.

IdP Administration går upp med en connector för okrypterad trafik (som skall termineras i lastbalanseraren).

- Certifikat och nyckel för IdP Administrations huvuddomän (ex. admin.idp.domain.test) läses in i lastbalanseraren och används för TLS terminering på all trafik mot huvuddomänen.

Det kan antingen vara tre separata certifikat, eller ett wildcard- eller multi-domain-certifikat, t.ex. ett SAN-cert med både huvuddomänen, secure-subdomänen och/eller admin-subdomänen bland sina Subject Alternative Names.

HSA-kommunikation

För kommunikation med HSA-katalogen krävs i regel (och definitivt vid anslutning till den nationella HSA-katalogen) ett SITHS-utfärdat funktionscertifikat vars HSA-id är registrerat i HSA-katalogen som behörigt att anropa aktuella tjänstekontrakt.

Övriga certifikat

Övriga certifikat är de som används för signering av SAML- och OIDC-meddelanden. Vanligtvis är detta också ett SITHS-utfärdat certifikat, och möjligen samma som används för kommunikation med HSA.

Se användarhandboken samt avsnittet om förstagångskonfiguration nedan för mer information kring installation av certifikat och nycklar.

Portöppningar

Applikationerna behöver åtkomst till

IP/System	IdP Application	IdP Administration
Mongo databas (samtliga noder)	✓	✓
Redis databas (samtliga noder)	✓	✓
HSA	✓	✗
OCSP/CRL	✓	✗
SAMBI, ifall federerat metadata skall hämtas	✓	✓
Autentiseringstjänsten, ifall autentisering med SITHS eID-klienterna skall användas	✓	✗

Beroenden till externa system

HSA

IdP Application nyttjar HSA som attributkälla, specifikt genom de tjänstekontrakt som finns specificerade i SAD:en.

Anslutning till den nationella HSA-katalogen

Anslutning av en tjänst till den nationella HSA-katalogen föregås av en utförlig anslutningsprocess. Läs mer på https://www.inera.se/tjanster/katalogtjanst-hsa/katalogtjanst-hsa/bestall--andra/ och kontakta Inera för att påbörja ett anslutningsförfarande.

Anslutning till regional HSA-katalog

Anslutning till en lokal/regional HSA-katalog (eller annan tjänst som implementerar de aktuella tjänstekontrakten) hanteras av den lokala/regionala förvaltningen.

Konfiguration av HSA-anslutning

1. Certifikat för kommunikation med HSA läses in enligt förstagångs-konfigurationen nedan.
2. Vilken HSA-katalog som IdP skall ansluta till konfigureras med följande parametrar i application-custom.properties (se avsnittet om systemkonfiguration nedan):

IdP Application: application-custom.properties

```
# HSA TK URL (exempel Prod Internet)
inera.common.hsa.host=https://esb.ntjp.se
# Paths
inera.common.hsa.authorization.getadmincredentialsforpersonincludingprotectedperson=/vp/infrastructure/directory
/authorizationmanagement/GetAdminCredentialsForPersonIncludingProtectedPerson
inera.common.hsa.authorization.getcredentialsforpersonincludingprotectedperson=/vp/infrastructure/directory
/authorizationmanagement/GetCredentialsForPersonIncludingProtectedPerson
inera.common.hsa.employee.getemployeeincludingprotectedperson=/vp/infrastructure/directory/employee
/GetEmployeeIncludingProtectedPerson
```

https://esb.ntjp.se/vp/infrastructure/directory/employee/GetEmployeeIncludingProtectedPerson/2/rivtabp21?wsdl
https://esb.ntjp.se/vp/infrastructure/directory/authorizationmanagement/GetCredentialsForPersonIncludingProtectedPerson/2/rivtabp21?wsdl
https://esb.ntjp.se/vp/infrastructure/directory/authorizationmanagement/GetAdminCredentialsForPersonIncludingProtectedPerson/2/rivtabp21?wsdl

Autentiseringstjänsten

Om SITHS eID-autentiseringsmetoderna skall användas behöver IdP anslutas till Autentiseringstjänsten. Se Anslutningsguide till Autentiseringstjänst SITHS för information om anslutningsförfarande, samt Nätverksinställningar för tjänster inom identitet och åtkomst för adresser.

IdP behöver ett SITHS Funktionscertifikat för kommunikation med Autentiseringstjänsten. IdP:ns HSA-id skickas in för registrering i Autentiseringstjänsten efter att anslutningen är godkänd. Detta HSA-id måste sedan matcha subject SERIALNUMBER i det certifikat som IdP använder för kommunikation mot Autentiseringstjänsten.

Konfiguration

Konfigurera url:en till Autentiseringstjänstens api, och aktivera SITHS eID-metoderna.

IdP Application: application-custom.properties

```
# Define which authentication methods should be available at authentication as well as client-registration.
authentication-method.methods.MTLS.enabled=true
authentication-method.methods.SITHS_EID_OTHER_DEVICE.enabled=true
authentication-method.methods.SITHS_EID_SAME_DEVICE.enabled=true

# URL to the RP-API used for SITHS eID.
siths-eid.host=https://secure-authservice.mobiltsiths.ineratest.org/api/rp/v1
```

Egenskaperna nedan hjälper till vid kommunikationssvårigheter mot Autentiseringstjänsten och konfigureras i IdP Application. Egenskaperna är till för att hjälpa systemen återhämta sig vid mindre fel i kommunikationen (ex. ett anrop fick inte ett svar i tid) och är därmed inte en fallback vid eventuella driftstörningar.

Notera att egenskaperna siths-eid.retry.actions.[**/***] ska ha ett värde på en endpoint i sig som IdP:n anropar Autentiseringstjänsten på. **/*** är alltså bara en placeholder för tabellen. Ett exempel på hur detta ska se ut egentligen är siths-eid.retry.actions.[**/auth**]. För varje endpoint (ex. [**/auth**]) bör alla egenskaper från tabellen nedan vara konfigurerade som har samma prefix, det vill säga **.backoff-delay**, **.max-attempts**, **.exception-classes**, **.traverse-causes**.

Egenskap	Defaultvärde	Exempelvärde	Förklaring
siths-eid.retry.enabled	false	true	Avgör ifall funktionaliteten för att göra återförsök vid misslyckade anrop ska vara aktiverad eller inte.
siths-eid.retry.actions.[/*].backoff-delay	0	1000	Fördröjningen i millisekunder för när nästa försök mot endpointen ska genomföras efter föregående försök.
siths-eid.retry.actions.[/*].max-attempts	0	3	Totala antalet försök som kommer genomföras. Värdet inkluderar alltså det initala första försöket. Exempelvis så betyder 3 alltså ett initialt försök och sedan maximalt två återförsök
siths-eid.retry.actions.[/*].exception-classes	[]	org.apache.http.conn.ConnectionPoolTimeoutException,org.apache.http.conn.ConnectTimeoutException	Lista på exception-klasser som triggar att nya försök ska genomföras.
siths-eid.retry.actions.[/*].traverse-causes	false	true	Avgör ifall hela exception-stacken ska kollas ifall det finns ett matchande exception från **siths-eid.retry.actions.[/*].exception-classes** eller om bara sista exceptionet ska matchas.

Egenskaperna nedan möjliggör tilldelningen av fler trådar för anrop mot Autentiseringstjänsten. RestTemplaten som nyttjas för kommunikationen skapar nya trådar i en ny trådpool. Om inget annat anges skapas endast två nya trådar i en trådpool på två trådar. Med konfigurationen kan dessa värden justeras upp så IdP:n inte misslyckas med anropen mot Autentiseringstjänsten pga. saknande trådar.

Egenskap	Defaultvärde	Exempelvärde	Förklaring
siths-eid.max-connections	0	25	Max antalet anslutningar för HTTP-klientens pool
siths-eid.max-conn-per-route	0	20	Max antalet anslutningar per host

Trust för server-kommunikation

Lägg till utfärdandekedjan för Autentiseringstjänstens certifikat i förtroendekällan "siths-eid" i IdP Administration.

SITHS-EID

Editera | Ta bort förtroendekälla

Trust mot AT

Namn	Id	Aktiv
TEST SITHS e-id Function CA v1	a46e4c0c79887f4ac91367ede19f4b1c7b1aea08	Ja
TEST SITHS e-id Root CA v2	fcb322f1863f63e0b3b2ba01def1add8dca2ab0c	Nej

Trust för användarcertifikat

Lägg till utfärdarkedjan för användarcertifikat som skall accepteras vid autentisering via SITHS eID-metoderna i förtroendekällan "user-siths-eid" i IdP Administration. (Alltså separat hantering från förtroendekällan "user" som endast styr mTLS-inloggningen).

USER-SITHS-EID

Editera | Ta bort förtroendekälla

User trusts för SITHS eID.

Namn	Id	Aktiv
TEST SITHS e-id Person HSA-id 3 CA v1	fc6fb340a84d5b3643d13ab38b6360634ca3d67f	Ja
TEST SITHS e-id Person ID 3 CA v1	cb6a39dc642290661694acaa1290ed044e90b272	Ja
TEST SITHS e-id Person HSA-id 2 CA v1	dd7b7d394be740ac6a518246fcbd83a4afa4320f	Ja
TEST SITHS e-id Function CA v1	a46e4c0c79887f4ac91367ede19f4b1c7b1aea08	Ja
TEST SITHS e-id Person ID 2 CA v1	a8fad6eaeba3fd331d1604a35e378cf029b6a2af	Ja
TEST SITHS e-id Person ID Mobile CA v1	b5638117e42ccae071a4b7d770686844bb5964f7	Ja
TEST SITHS e-id Root CA v2	fcb322f1863f63e0b3b2ba01def1add8dca2ab0c	Nej

I bilden ovan så är Function CA v1 inläst för att tillåta automatiserade tester (bilden är från en testmiljö).

Applikationskonfiguration

Installationsspecifik konfigurering görs i filen **config/application-custom.properties**. En exempelfil medföljer, men viss konfigurering i denna måste göras innan uppstart.

IdP Application

Framförallt måste *idp.server.host*, dvs den externa URL som man ansluter till denna instans av IdP:n sättas, samt konfiguration för att ansluta till databaserna (*spring.redis.** och *spring.data.mongodb.uri*) innan uppstart.

IdP Application: config/application-custom.properties

```
######################### IDP APPLICATION CONFIGURATION #########################

##############################################################################
############################## SERVER CONFIGURATION ##############################
##############################################################################
# Outward facing address, should match public address in LB
idp.server.protocol=https
idp.server.host=idp.domain.test
idp.server.port=443


##############################################################################
################################## MTLS CONNECTOR ##################################
##############################################################################
inera.common.server.mtls.port=8443

# Disable before first start, until the identity-group (below) has been configured with certificates
inera.common.server.enable=true

# Certificates and keys, configured in the admin GUI
inera.common.server.mtls.identity-group=idp-secure


##############################################################################
################################### SECURITY ###################################
##############################################################################
# IP ranges allowed to access actuator endpoints
inera.common.security.web.internal-ip-range=127.0.0.1,10.0.0.0/8

# Allow or disallow access with certificates for which OSCP status cannot be verified due to network issues
inera.common.trust.allow-undetermined=true


##############################################################################
############################### DB CONFIGURATION ################################
##############################################################################
# Collection prefix
idp.db.prefix=idp


##############################################################################
############################## REDIS CONFIGURATION ##############################
##############################################################################
# Password, if any
#spring.redis.password=password

# Connection timeout, ISO8601 Duration format
spring.redis.timeout=PT1M

# Redis single node configuration
#spring.redis.password=
spring.redis.host=redis-master
spring.redis.port=6379

## Redis sentinel configuration
#spring.redis.sentinel.master=redis-cluster-name
#spring.redis.sentinel.nodes=redis-sentinel-1:26379,redis-sentinel-2:26379,redis-sentinel-3:26379


##############################################################################
############################## MONGODB CONFIGURATION ############################
```

```
################################################################################
## MongoDB replica set configuration
spring.data.mongodb.uri=mongodb://user:password@mongodb-node1:27017,mongodb-node2:27017,mongodb-node3:27017,
mongodb-node4:27017/database?replicaSet=mongo-replica-set-name&ssl=true
mongo-ssl-ca-file=<file path to truststore containing the CA issuing the certificate used by MongoDB>

# QUARTZ (using MongoDB)
spring.quartz.properties.additionalconfig.uri=${spring.data.mongodb.uri}
spring.quartz.properties.additionalconfig.collection-prefix=${idp.db.prefix}_quartz


################################################################################
########################### AUTHENTICATION METHODS ############################
################################################################################
# Define which authentication methods should be available at authentication as well as client-registration.
authentication-method.methods.MTLS.enabled=true
authentication-method.methods.SITHS_EID_OTHER_DEVICE.enabled=false
authentication-method.methods.SITHS_EID_SAME_DEVICE.enabled=false

# URL to the RP-API used for SITHS eID.
#siths-eid.host=https://secure-authservice.mobiltsiths.ineratest.org/api/rp/v1


################################################################################
##################################### HSA ######################################
################################################################################
# HSA TK URL (exempel, direktanslutning HSA, test, sjunet)
#inera.common.hsa.host=https://wstest.hsa.sjunet.org
# Paths
#inera.common.hsa.authorization.getadmincredentialsforpersonincludingprotectedperson=/getadmincredentials_2
/hsaws/getadmincredentialsforpersonincludingprotectedperson
#inera.common.hsa.authorization.getcredentialsforpersonincludingprotectedperson=/tk2/hsaws
/getcredentialsforpersonincludingprotectedperson
#inera.common.hsa.employee.getemployeeincludingprotectedperson=/tk2/hsaws/getemployeeincludingprotectedperson

# HSA TK URL (exempel, via nationella tjänsteplattformen, Prod, Internet)
inera.common.hsa.host=https://esb.ntjp.se
# Paths
inera.common.hsa.authorization.getadmincredentialsforpersonincludingprotectedperson=/vp/infrastructure/directory
/authorizationmanagement/GetAdminCredentialsForPersonIncludingProtectedPerson
inera.common.hsa.authorization.getcredentialsforpersonincludingprotectedperson=/vp/infrastructure/directory
/authorizationmanagement/GetCredentialsForPersonIncludingProtectedPerson
inera.common.hsa.employee.getemployeeincludingprotectedperson=/vp/infrastructure/directory/employee
/GetEmployeeIncludingProtectedPerson

# Whether or not to include a connectivity check towards HSA in /actuator/health
inera.common.hsa.healthcheck=false
# Personal identity number used in connectivity test
#inera.common.hsa.connectivity-test-person-identity-number = 19121212121212

# Searchbase
#inera.common.hsa.default-search-base = c=SE
# Logical adress
#inera.common.hsa.logical-adress = SE165565594230-1000


################################################################################
############################ HSM CONFIGURATION #################################
################################################################################
inera.hsm.enabled=false

# Prioritized list. Application will try and connect to the first one in the list,
# if that fails it continues with the next in the list until it manages to establish a connection.
# If none of the specified slots work the application will crash.
inera.hsm.slots=1,3

inera.hsm.user.role=CRYPTOOFFICER
inera.hsm.user.pwd=replaceme

inera.hsm.signer.enabled=${inera.hsm.enabled}
```

```
# The aliases in the HSM that the signer service should use to fetch credentials from. The first alias in the
list is the one that will be used.
inera.hsm.signer.key-aliases=idp.domain.test-key


################################################################################
################################## LOG CONFIG ##################################
################################################################################
# External log config, enables updating of log settings in runtime
logging.config=file:/deployments/logging/logback-spring.xml
```

IdP Administration

IdP Application: config/application-custom.properties

```
###################### IDP ADMINISTRATIONCONFIGURATION #######################

##############################################################################
########################### SERVER CONFIGURATION ############################
##############################################################################
# Outward facing address, should match public address in LB
admin-idp.server.protocol=https
admin-idp.server.host=admin.idp.domain.test
admin-idp.server.port=443

##############################################################################
################################### SECURITY ################################
##############################################################################
# Security level for admin GUI
# oidc: Secured with OIDC, default
# password: Secured with formlogin using user/password
# none: Unsecured
inera.common.security.web.level=oidc

# Username and password for admin GUI when security level is set to password
inera.common.security.web.admin-user.user-name=qwerty
inera.common.security.web.admin-user.password=asdfgh

# IP ranges allowed to access actuator endpoints
inera.common.security.web.internal-ip-range=127.0.0.1,10.0.0.0/8

# Allow or disallow access with certificates for which OSCP status cannot be verified due to network issues
inera.common.trust.allow-undetermined=true


##############################################################################
############################## DB CONFIGURATION ##############################
##############################################################################
# Collection prefix
mongo.db-prefix=idp


##############################################################################
########################### REDIS CONFIGURATION ##############################
##############################################################################
# Password, if any
#spring.redis.password=password

# Connection timeout, ISO8601 Duration format
spring.redis.timeout=PT1M

# Redis single node configuration
#spring.redis.password=
spring.redis.host=redis-master
spring.redis.port=6379

## Redis sentinel configuration
#spring.redis.sentinel.master=redis-cluster-name
#spring.redis.sentinel.nodes=redis-sentinel-1:26379,redis-sentinel-2:26379,redis-sentinel-3:26379


##############################################################################
########################### MONGODB CONFIGURATION ############################
##############################################################################
## MongoDB replica set configuration
spring.data.mongodb.uri=mongodb://user:password@mongodb-node1:27017,mongodb-node2:27017,mongodb-node3:27017,
mongodb-node4:27017/database?replicaSet=mongo-replica-set-name&ssl=true
mongo-ssl-ca-file=<file path to truststore containing the CA issuing the certificate used by MongoDB>
```

```
# QUARTZ (using MongoDB)
spring.quartz.properties.additionalconfig.uri=${spring.data.mongodb.uri}
spring.quartz.properties.additionalconfig.collection-prefix=${idp.db.prefix}_quartz


################################################################################
########################## AUTHENTICATION METHODS ##############################
################################################################################
# Define which authentication methods should be available at authentication as well as client-registration.
authentication-method.methods.MTLS.enabled=true
authentication-method.methods.SITHS_EID_OTHER_DEVICE.enabled=false
authentication-method.methods.SITHS_EID_SAME_DEVICE.enabled=false


################################################################################
############################## HSM CONFIGURATION ###############################
################################################################################
inera.hsm.enabled=false

# Prioritized list. Application will try and connect to the first one in the list,
# if that fails it continues with the next in the list until it manages to establish a connection.
# If none of the specified slots work the application will crash.
inera.hsm.slots=1,3

inera.hsm.user.role=CRYPTOOFFICER
inera.hsm.user.pwd=replaceme

inera.hsm.signer.enabled=${inera.hsm.enabled}

# The aliases in the HSM that the signer service should use to fetch credentials from. The first alias in the
list is the one that will be used.
inera.hsm.signer.key-aliases=idp.domain.test-key


################################################################################
################################### LOG CONFIG #################################
################################################################################
# External log config, enables updating of log settings in runtime
logging.config=file:/deployments/logging/logback-spring.xml


################################################################################
##################################### SAMBI ####################################
################################################################################
# Automated job to fetch federated SAML metadata from SAMBI
saml.sambi-job-enabled=false
saml.sambi-job-cron-expression=0 0 0/2 * * ?
#saml.sambi-job-cron-expression=0 * * ? * *
#saml.sambi-job-cron-expression=0 */5 * ? * *

# URI to SAMBI federated metadata
saml.federated-metadata-url=https://fed.sambi.se/trial/md/metadata.xml


################################################################################
#################################### MISC ######################################
################################################################################
# Link to the user manual in GUI
idp.usermanual=https://confluence.cgiostersund.se/x/T6yhCg

################################################################################
```

Utökad konfiguration av IdP

Konfiguration av spärrkontroller

IdP Application stödjer att använda både OCSP och CRL för spärrkontroller med eller utan cache och fallback sinsemellan. Nedan finns de konfigurationsparametrar som kan användas för att styra detta beteende

Egenskap	Defaultvärde	Exempelvärde	Förklaring
inera.common.trust.do-revocation-check	true	true	Avgör ifall spärrkontroller ska genomföras.
inera.common.trust.no-fallback	ej definierat, resulterar i false	false	Avgör om fallback-tekniken ska användas när revokeringsstatus inte kan avgöras med hjälp av den primära källan. Exempelvis ifall ingen revokeringsstatus kunde avgöras med hjälp av OCSP ifall applikationen ska falla tillbaka till CRL:erna eller vice versa.
inera.common.trust.only-end-entity	ej definierat, resulterar i false	true	Avgör ifall enbart "sista" certifikatet (ex. slutanvändarcertifikatet) i certifikatskedjan ska genomgå spärrkontrollen.
inera.common.trust.prefer-crls	ej definierat, resulterar i false	false	Avgör ifall CRL ska användas som primär källa för revokeringsstatus.
inera.common.trust.soft-fail	ej definierat, resulterar i false	false	Avgör ifall revokeringskontrollen kan anses som lyckad ifall revokeringsstatusen inte kan avgöras p.g.a. exempelvis nätverksfel. När flaggan är satt till **false** kan revokeringsstatusen resultera i **UNDETERMINED_REVOCATION_STATUS**.
inera.common.trust.allow-undetermined	ej definierat, resulterar i false	false	Avgör ifall ett certifikat som fått revokeringsstatusen **UNDETERMINED_REVOCATION_STATUS** ska anses som giltigt eller inte. Har endast effekt ifall **inera.common.trust.soft-fail** är satt till **false**.
inera.common.trust.max-path-length	10	10	Max antalet certifikat som får finnas i en certifikatskedja
inera.common.trust.use-only-cached-revocation-data	ej definierat, resulterar i false	false	Avgör ifall revokeringskontrollen ska genomföras ifall ingen revokeringsstatus finns tillgängligt för certifikatet.
inera.common.trust.dynamically-sort-fallback-order	ej definierat, resulterar i false	false	Avgör ifall prioritetsordningen på cachead revokeringsstatus ska sorteras dynamiskt. Förutsätter att **inera.common.trust.no-fallback** är satt till **false**.
inera.common.trust.verifier.enable-explicit-ocsp-lookup	true	true	Avgör ifall OCSP data ska hämtas i förväg och komplettera befintlig data innan revokeringskontrollen ska genomföras
inera.common.revocation-check-executor.only-end-entity	false	false	Avgör ifall enbart slutanvändarcertifikatet ska kontrolleras i RevocationCheckExecutor-klassen.
inera.common.ocsp-data-service.enable-cache	false	true	Avgör ifall revokeringsdata ska sparas till cachen och hämtas från cachen ifall direktuppslag mot OCSP-respondern misslyckas.
inera.common.crl-data-service.max-age-seconds	null	259200	Anges i sekunder. Avgör maximala livslängden på cachen för CRL:er innan dom anses för gamla.
inera.common.crl-data-service.use-old-crls	false	false	Avgör ifall utgångna CRL:er ska användas för spärrkontroll. Utgångstiden styrs av **inera.common.crl-data-service.max-age-seconds**
inera.common.crl-data-service.local-cache-ttl-seconds	1800	1800	Anges i sekunder. Avgör hur länge minnescachen är giltig innan minnescachen behöver uppdateras med uppgifter från Redis-cachen. Prestandahöjande funktion som används för att minska antalet slagningar mot Redis-cachen.
inera.common.crl-data-job.fixed-rate-millis	ej definierat	1800000	Anges i millisekunder. Avgör frekvensen med hur ofta CRL-jobbet ska köras.
inera.common.crl-data-job.end-entity-crl-urls	ej definierat, resulterar i tom lista	http://crl1pp.siths.se/testsithseidpersonhsaid3cav1.crl, \ http://crl1pp.siths.se/testsithseidpersonid3cav1.crl,	Anges som kommaseparerad lista. Avgör vilka extra CRL:er som ska hämtas utöver de CRL:er som återfinns i applikationens inlästa certifikatsutfärdare.

Loggning

Inställningar för loggning kan göras i filen **logging/logback-spring.xml**.

Per default skrivs loggarna till fil (logs/auth-application.log), detta går att ändra till att skrivas till standard out (konsoll) genom att ändra raden *<appender-ref ref="FILE" />* till *<appender-ref ref="CONSOLE" />*.

logging/logback-spring.xml

```
<?xml version="1.0" encoding="UTF-8"?>
<configuration scan="true" scanPeriod="60 seconds">

  <property name="LOG_FILE" value="logs/auth-application.log" />
  <property name="LOG_FILE_MAX_SIZE" value="10MB" />
  <property name="LOG_FILE_MAX_HISTORY" value="7" />

  <include resource="org/springframework/boot/logging/logback/defaults.xml" />
  <include resource="org/springframework/boot/logging/logback/console-appender.xml" />
  <include resource="org/springframework/boot/logging/logback/file-appender.xml" />

  <!-- Global logging level of application -->
  <logger name="com.cgi.se.inera" level="INFO" />

  <!-- Supress verbose loggers -->
  <logger name="com.novemberain.quartz.mongodb" level="WARN" />

  <!-- INFO level needed to log the SOAP messages -->
  <logger name="org.apache.cxf" level="WARN" />
  <logger name="org.apache.cxf.services" level="WARN" />

  <!-- WARNING! -->
  <!-- Enabling message logging can and will expose personal information about end users in the logs! -->
  <!-- Only enable message logging if it is needed for debugging purposes, and only for limited times. -->

  <!-- OpenSaml logger for SAML request/response. DEBUG for SAML messages -->
  <logger name="PROTOCOL_MESSAGE" level="DEBUG" />

  <!-- fine tune individual service logging -->
  <logger name="org.apache.cxf.services.GetEmployeeIncludingProtectedPersonResponderInterface.REQ_OUT" level="
WARN" />
  <logger name="org.apache.cxf.services.GetEmployeeIncludingProtectedPersonResponderInterface.RESP_IN" level="
WARN" />
  <logger name="org.apache.cxf.services.GetEmployeeIncludingProtectedPersonResponderInterface.FAULT_IN" level="
INFO" />

  <logger name="org.apache.cxf.services.GetCredentialsForPersonIncludingProtectedPersonResponderInterface.
REQ_OUT" level="WARN" />
  <logger name="org.apache.cxf.services.GetCredentialsForPersonIncludingProtectedPersonResponderInterface.
RESP_IN" level="WARN" />
  <logger name="org.apache.cxf.services.GetCredentialsForPersonIncludingProtectedPersonResponderInterface.
FAULT_IN" level="INFO" />

  <logger name="org.apache.cxf.services.GetAdminCredentialsForPersonIncludingProtectedPersonResponderInterface.
REQ_OUT" level="WARN" />
  <logger name="org.apache.cxf.services.GetAdminCredentialsForPersonIncludingProtectedPersonResponderInterface.
RESP_IN" level="WARN" />
  <logger name="org.apache.cxf.services.GetAdminCredentialsForPersonIncludingProtectedPersonResponderInterface.
FAULT_IN" level="INFO" />

  <logger name="org.apache.cxf.services.NetiDAccessServerSoap.REQ_OUT" level="WARN" />
  <logger name="org.apache.cxf.services.NetiDAccessServerSoap.RESP_IN" level="WARN" />
  <logger name="org.apache.cxf.services.NetiDAccessServerSoap.FAULT_IN" level="INFO" />

  <logger name="com.cgi.se.inera.common.pkix.server.X509HeaderFilter" level="WARN" />
  <logger name="com.cgi.se.inera.common.pkix.TrustServiceImpl" level="WARN" />
  <logger name="com.cgi.se.inera.auth.oidc.endpoint.advice.OIDCExceptionHandler" level="DEBUG" />
  <logger name="com.cgi.se.inera.common.job.NonSystemExitMongoDBJobStore" level="WARN" />

  <logger name="com.cgi.se.inera.auth.core.logging.ResponseLoggingFilter" level="WARN" />
  <logger name="org.springframework.web.filter.CommonsRequestLoggingFilter" level="WARN" />

  <root level="INFO">
    <!-- Log to file or to console -->
    <appender-ref ref="FILE" />
    <!-- <appender-ref ref="CONSOLE" /> -->
  </root>

</configuration>
```

Inför första uppstart: Konfiguration av nycklar, cert och behörighet

Ställ ner säkerheten på IdP Administration och stäng av mTLS-connectorn

När applikationen skall startas första gången så måste säkerheten på administrationsgränssnittet sättas ner för att kunna konfigurera nycklar, certifikat och behörigheter.

IdP Administration: application-custom.properties

```
inera.common.security.web.level=password
inera.common.security.web.admin-user.user-name=qwerty
inera.common.security.web.admin-user.password=asdfgh
```

Samtidigt måste mTLS-connectorn vara avstängd tills det finns en nyckelkollektion den kan använda.

IdP Application: application-custom.properties

```
inera.common.server.enable=false
```

Starta sedan IdP Administration enligt uppstarts-instruktionerna.

Konfigurera systemet via IdP Administration

Åtkomst till administrationsgränssnittet sker genom att gå mot /admin-endpointen som finns i IdP Administration (t.ex. https://admin.idp.domain.test/admin).

Se användarhandboken för information om hur gränssnittet används.

Konfigurera applikationens certifikat och nycklar

Lägg upp alla nyckelgrupper som behövs och läs in certifikat och nycklar.

Grupp-ID	Beskrivning
idp	Anger de certifikat och nycklar som används av IdP Application för SAML och OIDC. Det aktiva certifikatet används för signering och övriga certifikat ingår som en del av IdP metadata (inom både SAML och OIDC).
idp-authentication	Anger de certifikat och nycklar som används av IdP Administrations gränssnitt för anslutning mot IdP Application.
idp-secure	Anger de certifikat och nycklar som används av mTLS-connectorn på *secure*-subdomänen för användarautentisering via mTLS.
hsa	Anger de certifikat och nycklar som används för anslutning till HSA. Är typiskt sett ett SITHS funktionscertifikat vars HSA-id är registrerat i HSA-katalogen som betrott att anropa aktuella tjänstekontrakt.

Registrera en OIDC-klient för admin-gui

Skapa en OIDC-klient för IdP Administration. Kopiera värden från fliken "RP Information" i administrationsgränssnittet. Dubbelkolla att nyckelgruppen som anges under "RP Information" är skapad enligt ovan.

Konfigurera behörighet för admin-gui

Sätt upp behörighetsregler för vilka HSA-attribut som krävs för att komma åt admin-gui.

1. Gå in på "Behörighet"
2. Klicka "Ny resurs"
3. Fyll i "ADMIN"
4. Lägg till en "READ" eller "WRITE"-Action
5. Klicka på respektive action under ADMIN-noden som dyker upp i behörighetsvyn i mitten
6. Lägg till önskade Conditions
    a. Namnsättningen är enligt OIDC-attributen på Attributlistan (t.ex. "employeeHsaId" om ni vill lägga till administratörer en och en, eller "systemRole" och "healthCareProviderHsaId" om alla med en viss roll i en organisation skall ha åtkomst)
    b. Tillgängliga OIDC-attribut är [name, employeeHsaId, commissionHsaId, commissionName, healthCareProviderHsaId, organizationName, mail, mobileTelephoneNumber, systemRole]
7. Klicka på respektive Condition och lägg till önskade värden

Läs in betrodda certifikat

Läs in betrodda certifikatsutfärdare för server-2-server kommunikation, användarcertifikat, sambi-federationen och eventuellt övriga metadatautfärdare.
Certifikatsutfärdare för IdP:s egna certifikat måste finnas inlästa för att IdP Administrations OIDC-klient och IdP skall kunna kommunicera med varandra.
Se Användarhandbok för IdP-administration för information om vilka förtroendekällor som behövs.
OBS: Notera under 2.4.1 i användarhandboken att loa-regler nu hämtas från en förtroendekälla som heter "loa".

Lägg in organisations- och kontaktuppgifter

Under "Konfiguration" i administrationsgränssnittet: Lägg till organisationsuppgifter samt minst två kontaktpersoner (en av Typ: technical och en av Typ: support). Denna information kommer med i IdP:s SAML-metadata.

Ställ upp säkerheten på IdP Administration och aktivera mTLS-connectorn

Säkra IdP Administration med OIDC-inloggning

När sedan trust och identiteter satts upp så ställs säkerheten på administrationsgränssnittet upp till att skyddas genom normal inloggning.

IdP Administration: application-custom.properties

```
inera.common.security.web.level=oidc
```

Aktivera mTLS-connectorn

Aktivera mTLS-connectorn nu när det finns en nyckelgrupp för den att använda.

IdP Application: application-custom.properties

```
inera.common.server.enable=true
```

Starta om applikationerna

Uppstart

Följande är ett exempel på hur applikationen kan startas med nödvändiga JVM-parametrar och environment-variabler.

Starta IdP Application med nödvändiga JVM-parametrar och environment-variabler

```
java -jar \
-Dfile.encoding=UTF-8 \
-Duser.country=SE \
-Duser.language=sv \
-Dspring.profiles.active=custom \
-Xms256m \
-Xmx1024m \
idp-application-*.jar
```

Lägg dessutom till följande konfig för att peka ut var Thales LunaProvider-jar (LunaProvider.jar) samt bibliotek (libLuna.so or LunaAPI.dll) ligger ifall IdP skall använda HSM för nyckelhantering.

HSM-konfig

```
-Djava.library.path=/usr/local/luna/jsp/64
-Dloader.path=/usr/local/luna/jsp/LunaProvider.jar
```

IdP-metadata

IdP tillhandahåller SAML- och OIDC-metadata på följande endpoints:

SAML-metadata	OIDC-metadata
<idp url>/saml	<idp url>/oidc/.well-known/openid-configuration

Administration (GUI)

Inloggning till administrationsgränssnittet sker genom att gå mot /admin som finns i IdP Administration (t.ex https://admin.idp.domain.test/admin).

Se användarhandboken för instruktioner kring hur administrationsgränssnittet används.

Externa endpoints

https://<idp url>/external/clients

Denna endpoint aktiveras av att sätta följande properties:

IdP Application: properties för /external/clients

```
inera.common.security.external.clients.username=
inera.common.security.external.clients.password=
```

Endpointen tillåter externa tjänster hämta viss information om vilka OIDC- och SAML-klienter som finns registrerade på IdP:n. Ett användningsfall är ifall anslutningarna från flera IdP:er ska aggregeras på ett och samma ställe.

Fullständig adress för endpointen: https://<*idp url*>/external/clients

Exempel /external/clients


```
{
  "oidcClients": [
    {
      "clientId": "https://idp.dev.inera.test:8443/oidc",
      "systemName": "Lokal IdP :8443",
      "customerName": "CGI AB",
      "authenticationMethods": [
        "MTLS",
        "SITHS_EID_SAME_DEVICE",
        "SITHS_EID_OTHER_DEVICE"
      ],
      "latestLogin": "2022-09-06T14:23:36.723",
      "prestudies": [],
      "clientType": "ORDINARY",
      "active": true
    }
  ],
  "federatedSamlClients": [],
  "nonFederatedSamlClients": [
    {
      "clientId": "https://sp.dev.inera.test:8881",
      "systemName": "Inera Test SP :8881",
      "customerName": "CGI AB",
      "authenticationMethods": [
        "SITHS_EID_OTHER_DEVICE",
        "SITHS_EID_SAME_DEVICE",
        "MTLS"
      ],
      "latestLogin": "2022-09-06T08:53:30.550",
      "prestudies": [],
      "clientType": "ORDINARY",
      "active": true,
      "federated": false
    }
  ]
}
```

https://<idp url>/external/statistics/*

Dessa endpoints aktiveras av att sätta följande properties:

IdP Application: properties för /external/statistics/*
`inera.common.security.external.statistics.username=` `inera.common.security.external.statistics.password=`

Dessa endpoints tillåter externa tjänster att hämta statistik från IdP:n om inloggings- och utloggningsförsöken som genomförts under ett angivet tidsintervall. Resultatet levereras som en .zip-fil som i sin tur innehåller en .csv-fil med all statistik.

Endpointsen tar emot två requestparametrar: **startDate** och **endDate**. Båda parametrarna tar emot ett ISO-formatterat datum, exempelvis det här: 2022-12-31

Idag finns tre versioner av dessa statistikendpoints:

- https://*<idp url>*/external/statistics/export_all?startDate=*<iso date>*&endDate=*<iso date>*
- https://*<idp url>*/external/statistics/export_all_v2?startDate=*<iso date>*&endDate=*<iso date>*
- https://*<idp url>*/external/statistics/export_all_v3?startDate=*<iso date>*&endDate=*<iso date>*

https://<idp url>/external/numericalstatistics/*

Denna endpoint aktiveras av att sätta följande properties:

IdP Application: properties för /external/numericalstatistics/*

```
inera.common.security.external.numericalstatistics.username=
inera.common.security.external.numericalstatistics.password=
```

Dessa endpoints tillåter externa tjänster att hämta numerisk statistik från IdP:n om inloggnings- och utloggningsförsöken som genomförts för en specifik dag med alternativ att specifiera resultatet för en specifik anslutning.

Resultaten är grupperade per hel timme, exempelvis 01:00:00 - 01:59:59. Hämtas statistiken för det pågående dygnet innehåller svaret fortfarande statistik från timmarna som ligger i framtiden för att vara konsistent i hur svaren levereras.

Idag finns två versioner av dessa statistikendpoints:

- https://<idp url>/external/numericalstatistics/hourly?date=<*iso date*>
- https://<idp url>/external/numericalstatistics/hourly-by-client?date=<*iso date*>&clientId=<*client id*>

Exempel för /external/numericalstatistics/hourly?date=2023-02-16


```
{
    "date": "2023-02-16",
        "clientId": null,
    "statistics": [
        {
            "timeRangeStart": "00:00:00",
            "timeRangeEnd": "00:59:59",
            "oidc": {
                "login": {
                    "success": 12,
                    "fail": 1
                },
                "logout": {
                    "success": 23,
                    "fail": 2
                }
            },
            "saml": {
                "login": {
                    "success": 34,
                    "fail": 3
                },
                "logout": {
                    "success": 45,
                    "fail": 4
                }
            }
        },
                {.....}
        }
}
```

Exempel för /external/numericalstatistics/hourly-by-client?date=2023-02-16&clientId=https://sp.dev.inera.test:8881


```
{
    "date": "2023-02-16",
        "clientId": "https://sp.dev.inera.test:8881",
    "statistics": [
        {
            "timeRangeStart": "00:00:00",
            "timeRangeEnd": "00:59:59",
            "oidc": {
                "login": {
                    "success": 12,
                    "fail": 1
                },
                "logout": {
                    "success": 23,
                    "fail": 2
                }
            },
            "saml": {
                "login": {
                    "success": 34,
                    "fail": 3
                },
                "logout": {
                    "success": 45,
                    "fail": 4
                }
            }
        },
                {.....}
        }
}
```