

SOSI STS Installationsbeskrivelse

Version 0.3

Indholdsfortegnelse

1	Introduktion.....	2
2	Forudsætninger.....	3
2.1	Java.....	3
2.2	JBoss.....	3
2.2.1	Server.....	3
2.3	MySQL.....	3
2.4	Spærrelister.....	3
3	STS konfiguration.....	4
3.1	JBoss.....	4
3.1.1	Admin beskyttelse.....	4
3.1.2	Logging.....	4
3.2	MySQL.....	4
3.3	stsconfig.properties.....	5
3.3.1	Logging.....	5
3.3.2	Database.....	5
3.3.3	CRL check.....	6
3.3.4	CPR opslag.....	6
4	Installation.....	7
4.1	Ny installation.....	7
4.2	Opgradering test af installation.....	7
4.3	Rollback.....	7
4.4	Test af installation.....	7
4.4.1	testconfig.sh.....	7
4.4.2	SOSI testtools.....	9

SOSI STS Installationsbeskrivelse

1 Introduktion

Dette dokument beskriver installation og konfiguration af en STS server. I afsnittene nedenfor gennemgås installation og konfiguration af tredieparts produkter og selve STS.

I det følgende refereres til følgende lokationer:

- STS_BUNDLE: Arkiv med STS, f.eks. `sosi_sts-200612241800.zip`
- STS_HOME: STS folderen på serveren, f.eks. `/opt/sosi`
- STS_CONFIG: STS configurations folder på serveren, f.eks. `/opt/sts/etc`
- JBOSS_HOME: JBoss folderen på serveren, f.eks. `/opt/jboss`
- JBOSS_STS: JBoss STS server konfiguration, f.eks. `/opt/jboss/server/sosi_sts`

Eksempelvis angiver `$STS_BUNDLE/sts.war` en fil i et arkiv, mens `$JBOSS_HOME` angiver en folder i filsystemet.

2 Forudsætninger

2.1 Java

STS kræver Java5, som kan downloades her: http://java.sun.com/javase/downloads/index_jdk5.jsp

Desuden stiller SOSI biblioteket visse krav til konfigurationen af Java, som er beskrevet i ”The SOSI Library: Programmers Guide” (se afsnit 3 ”Set-up”). Dokumentet kan downloades sammen med SOSI biblioteket her: <http://kb.oio.dk/projects/sosi/wiki/Download>

I den komplette distribution af SOSI biblioteket findes en kommandolinietest, der kan bruges til at verificeret at Java er konfigureret korrekt.

2.2 JBoss

STS anvender JBoss 4.0, men stiller ikke krav om en specifik version. JBoss kan downloades her: <http://labs.jboss.com/portal/jbossas/download>

2.2.1 Server

JBoss bruger server konfigurationer, som bl.a. stiller forskellige services til rådighed. STS kræver primært, at Tomcat er til rådighed. Med en server configuration ved navn `sosi_sts` kan JBoss startes som følger:

```
$ run.sh -c sosi_sts
```

I forbindelse med installationen bør man naturligvis sikre sig, at Jboss kører som en dæmon

2.3 MySQL

STS anvender MySQL 5, men stiller ikke krav om en specifik version. MySQL kan downloades her: <http://dev.mysql.com/downloads/mysql/5.0.html>

I forbindelse med installationen bør man naturligvis sikre sig at MySQL kører som en dæmon.

2.4 Spærrelister

STS kan hente spærrelister lokalt eller fra en anden server. Man kan derfor med fordel schedulere (f.eks. med cron) hentning spærrelister, så de ligger lokalt. Derved kan frekvensen af check for nye spærrelister normalt øges.

[TODO: note om check af spærrelister]

3 STS konfiguration

Konfiguration af STS består af tre dele: konfiguration af JBoss, MySQL og STS web applikationen.

3.1 JBoss

3.1.1 Admin beskyttelse

Administrations interfacet til STS er beskyttet af HTTP basic authentication, hvilket i JBoss angives ved at tilføje et `application-policy` element under `policy` rod elementet i `JBOSS_STS/conf/login-config.xml`

Nedenstående sætter fil-baseret bruger og rolle opslag:

```
<!-- STS admin login -->
<application-policy name="stsAdmin">
  <authentication>
    <login-module
      code="org.jboss.security.auth.spi.UsersRolesLoginModule"
      flag="required">
      <module-option name="usersProperties">
        props/stsadmin-users.properties
      </module-option>
      <module-option name="rolesProperties">
        props/stsadmin-roles.properties
      </module-option>
    </login-module>
  </authentication>
</application-policy>
```

Eksempel filerne kan findes i `STES_BUNDLE/examples`.

For at verificere at konfigurationen er korrekte, tilgås STS admin med en browser, f.eks. på <http://pan.certifikat.dk/sts/admin>.

3.1.2 Logging

Logging i STS kan ske ved enten at anvende den medfølgende Log4j konfiguration (`STES_BUNDLE/config/log4j.xml`), eller ved at tilpasse log konfigurationen i `JBOSS_STS/conf/log4j.xml` (se også nedenfor).

3.2 MySQL

STS bruger en database til caching af data og vedligehold af adgangsinformation, så følgende skridt er nødvendige før STS kan startes:

1. Opret database
2. Opret bruger med passende privilegier.
3. Opret tabeller

SOSI STS Installationsbeskrivelse

Se eksempelvis `$STS_BUNDLE/examples/createdb.sql`).

For at testen i SOSI bibliotekets testtool fungerer mod Test STS, skal bestemte certifikater være henholdsvis black- og white listed. Det kan gøre vi administrationsinterfacet, men det kan også gøres ved direkte indsættelse i databasen (se f.eks.

`$STS_BUNDLE/examples/testststdata.sql`)

3.3 stsconfig.properties

Filen `$STS_CONFIG/stsconfig.properties` indeholder konfiguration af selve STS, f.eks. hvilken database der skal bruges og hvor spærrelister skal trækkes fra. En stor del af denne konfigurationsfil indhold vil normalt ikke skulle rettes, da de fleste properties har fornuftige default værdier.

Nedenfor er beskrevet værdier i konfigurationen som man normalt vil have behov for at kunne ændre. Med mindre man er klar over konsekvenserne, bør man ikke rette i de resterende værdier, hvilket man normalt heller ikke vil have behov for.

BEMÆRK: indholdet af denne konfigurationsfil er generelt case-sensitivt og man skal være opmærksom på trailing spaces.

3.3.1 SOSI seal

STS serverens offentlige certifikat og private nøgle angives med følgende properties:

```
# SOSIFactory configuration
#dk.sosi.sts.server.StsFactory.SOSIFactory.sosi\:issuer = STS
#dk.sosi.sts.server.StsFactory.SOSIFactory.sosi\:validate = true
dk.sosi.sts.server.StsFactory.SOSIFactory.keystore = ...
dk.sosi.sts.server.StsFactory.SOSIFactory.password = ...
```

Det er naturligvis vigtigt at sikre sig, at keystore filen findes, men også at den har den rigtige form i forhold til hvad seal biblioteket forventer. Det betyder certifikatet og nøglen skal have et alias der hedder "SOSI:ALIAS_SYSTEM".

De to nøgler `sosi:issuer` og `sosi:validate` (husk '\' er nødvendig i property-filen) sætter egenskaber for SOSI seal biblioteket, og standard værdierne er henholdsvis 'TESTSTS' og 'false'.

BEMÆRK: På produktion skal `sosi:issuer` dog være 'STS'.

3.3.2 Logging

STS logger med Log4j og kan gøres på to måder:

- Brug JBoss log konfiguration
- Brug STS log konfiguration

SOSI STS Installationsbeskrivelse

Dette styres med nedenstående property:

```
# General STS configuraion
sts.log4j.config = ...
```

Hvis ikke denne property findes bruges JBoss log konfiguration, ellers konfigureres Log4j med den XML log konfigurationsfil, som værdien af `sts.log4j.config` peger på.

3.3.3 Database

Database konfigurationen styres af nedenstående properties:

```
# STS db config
sts.db.driverClassName = com.mysql.jdbc.Driver
sts.db.username = ...
sts.db.password = ...
sts.db.url = jdbc:mysql://localhost:3306/sts
```

Her skal angives brugernavn/kodeord for en bruger med passende privilegier. Derudover kan man styre egenskaber ved connection pool (ikke beskrevet her).

3.3.4 CRL check

Hvorfra og hvorofte spærrelister skal hentes, styres af nedenstående properties:

```
# OcesCrlService configuration
dk.sosi.sts.server.service.StsOcesCrlServiceImpl.crl = ...
dk.sosi.sts.server.service.StsOcesCrlServiceImpl.interval = 3600
dk.sosi.sts.server.service.StsOcesCrlServiceImpl.strict = true
```

- `crl`: URL til spærrelisten
- `interval`: max periode i sekunder mellem check for ny spærreliste
- `strict`: hvordan opfører CRL checks sig hvis spærreliste ikke er tilgængelig

3.3.5 CPR opslag

Hvorfra og hvorofte spærrelister skal hentes, styres af nedenstående properties:

SOSI STS Installationsbeskrivelse

```
# OcesCvrRidService configuration
dk.sosi.sts.server.service.StsOcesCvrRidService.endpoint = ...
dk.sosi.sts.server.service.StsOcesCvrRidService.trustStore = ...
dk.sosi.sts.server.service.StsOcesCvrRidService.trustStorePassword =
...
dk.sosi.sts.server.service.StsOcesCvrRidService.keyStore = ...
dk.sosi.sts.server.service.StsOcesCvrRidService.keyStorePassword = ...
dk.sosi.sts.server.service.StsOcesCvrRidService.datasource = sts.db
```

- endpoint: URL til spærrelisten
- trustStore: giver sig selv
- trustStorePassword: giver sig selv
- keyStore: giver sig selv
- keyStorePassword: giver sig selv

4 Installation

4.1 Ny installation

Hvis STS *ikke har* været installeret tidligere på en server gøres følgende:

1. Pak `$STS_BUNDLE` ud og placer filerne i `$STS_HOME` og `$STS_CONFIG` som ønsket.
2. Gennemgå ovenstående afsnit
3. `$STS_BUNDLE/sts.war` til `$JBOSS_STS/deploy`
4. Start JBoss

4.2 Opgradering test af installation

Hvis STS *har* været installeret tidligere på en server gøres følgende:

1. Check konfigurationsfiler i `$STS_BUNDLE/config`
2. Check evt database skemaændringer
3. Kopier `$STS_BUNDLE/sts.war` til `$JBOSS_STS/deploy`
4. Hvis autodeploy er slået fra skal JBoss genstartes

4.3 Rollback

For at kunne rulle tilbage til en tidligere installeret version, skal man naturligvis have en backup. Følgende er i den sammenhæng relevant:

- Database: backup/restore data, f.eks. med `mysqldump/mysql`
- Konfiguration: `$STS_CONFIG`
- Webapplikationen: `$JBOSS_STS/deploy/sts.war`

Såfremt der ikke er skemaændringer i databasen, er det ikke nødvendigt at lave backup/restore. Har man lavet væsentlige ændringer i JBoss serverkonfiguration (`$JBOSS_STS`), som man ønsker at kunne rulle tilbage, kan man også lave backup/restore af denne folder, evt. uden temporære filer i log, tmp og work.

4.4 Test af installation

4.4.1 testconfig.sh

For at verificere at konfigurationen af STS (`stsconfig.properties`) er delvis korrekt, kan det medfølgende shell script bruges. Før det kan køres skal environment variabelen `$STS_WAR` sættes til at pege på en udpakket udgave af `$STS_BUNDLE/sts.war`.

For at se hvilke kommandoer testen scriptet understøtter køres følgende:

SOSI STS Installationsbeskrivelse

```
$ testconfig.sh
TestConfig stsconfig.properties command [args...]
  command
  arg1..n

example commands:
  FindCpr CVR:19343634-RID:1165813849809
  IsRelated CVR:19343634-RID:1165813849809 1111111118
  CrlTimestamp
  Keystore
```

Check om CPR opslag virker, eksempelvis:

```
$ testconfig.sh stsconfig.properties FindCpr CVR:19343634-
RID:1165813849809
...
[CVR:19343634-RID:1165813849809]->1111111118
```

Check om CPR opslag virker og der er adgang til databasen, eksempelvis:

```
$ etc/testconfig.sh target/config/stsconfig.properties IsRelated
CVR:19343634-RID:1165813849809 1111111118
...
[[CVR:19343634-RID:1165813849809] , [11111xxxx]]->>true
```

Såfremt kombination af CVR-RID og CPR findes vil den blive tilføjet til databasen (SELECT * FROM cprhash).

Check spærreliste konfiguration, eksempelvis:

```
$ etc/testconfig.sh target/config/stsconfig.properties CrlTimestamp
...
timestamp = Sun Mar 18 10:40:13 CET 2007
```

Hvis spærrelisteopslaget er korrekt konfigureret, vises spærrelistens tidsstempel.

Check at keystore opfylder SOSI seal kravene til et keystore, dvs. at certifikat og privat nøgle findes under det rigtige alias:

```
$ etc/testconfig.sh target/config/stsconfig.properties Keystore
...
STS system certificate: CN=Danske Regioner - SOSI STS +
SERIALNUMBER=CVR:55832218-UID:1163447368627, O=Danske Regioner //
CVR:55832218, C=DK
```

Hvis det konfigurerede keystore opfylder kravene skrives certifikatets distinguished name ud.

SOSI STS Installationsbeskrivelse

4.4.2 SOSI testtools

Med SOSI biblioteket følger testtool, som kan anvendes til at test klient webservicekald mod en STS installation. Det kan downloades med den komplette SOSI pakke fra:

<http://kb.oio.dk/projects/sosi/wiki/Download>

Som default benytter testtools test STS på pan.certifikat.dk. Ønsker man at teste en anden STS, skal følgende Java system property sættes:

```
-Dststs.url=http://<hostname>/sts/services/SecurityTokenService
```

Der følger et Windows cmd script med testtools, som umiddelbart kan bruges som udgangspunkt for et Unix shell script. Test afvikles som:

```
$ ./runtesttools.cmd
.
Running testtools ... please wait
.
log4j:WARN No appenders could be found for logger \
(org.apache.xml.security.Init)
.
log4j:WARN Please initialize the log4j system properly.
.....
Time: 10,641

OK (39 tests)

Please refer to the 'output/*' subfolders to see generated requests \
and responses.
```

BEMÆRK: da testtools bruger TDC OCES testcertifikater, kan den ikke bruges på productions STS.