

v1.19_NBI

-
-
- - /
 - [/etc/eltex-radius-nbi/radius_nbi_config.txt](#)
 - [/etc/eltex-radius-nbi/hikari.properties](#)
-

```
<div class="lui-message error lui-message-error">
<p class="title">
<span class="lui-icon icon-error"></span>
<strong>Comala Metadata License Details</strong>
</p>
<p>Invalid commercial evaluation license with a expired error. Please click <a href="https://marketplace.atlassian.com/plugins/org.andya.confluence.
plugins.metadata" target="_blank">here</a> to purchase a commercial license.</p>
</div>
```

SoftWLC, SOAP. : , , PCRF, OSS/BSS. TLS-.

```
apt-get install eltex-axis
apt-get install eltex-radius-nbi
```

 eltex-axis, eltex-radius-nbi ().

/
tomcat (7), / .

/etc/eltex-radius-nbi/radius_nbi_config.txt

SoftWLC.

- MySQL. (ELTEX_PORTAL, !), radius.

```
# DB radius(alias=radius)
radius.jdbc.driver=org.gjt.mm.mysql.Driver
radius.jdbc.dbUrl=jdbc:mysql://localhost/radius?
zeroDateTimeBehavior=convertToNull&useUnicode=true&characterEncoding=utf8&relaxAutoCommit=true&connectTimeout=5000
radius.jdbc.username=javauser
radius.jdbc.password=javapassword
radius.jdbc.maxPoolSize=48
radius.jdbc.inUse=yes
```

- **northbound EMS, eltex-ems:**

```
# ems-northbound
ems.nbi.host=127.0.0.1
ems.nbi.port=8080
ems.nbi.path=northbound
ems.nbi.protocol=http
```

- **eltex-auth-service ssh, NAS :**

```
# eltex_auth_service
auth.port=22
auth.host=127.0.0.1
auth.username=username
auth.password=password
```

- **eltex-radius ssh, NAS:**

```
# freeradius-domain-1
freeradius-domain-1.port=22
freeradius-domain-1.host=192.168.0.1
freeradius-domain-1.username=username
freeradius-domain-1.password=password
```

- **Tomcat, . TLS- .**

```
# tomcat url
tomcat.host=127.0.0.1
tomcat.port=8080
```

- **PCRF:**

```
# pcrf stuff
pcrf.enabled=true
pcrf.url=http://localhost:7070
pcrf.username=admin
pcrf.password=password
pcrf.readtimeout=11
pcrf.writetimeout=1
```

- **MongoDB pcrf:**

```
# pcrf mongodb connector
pcrf.mongodb.enabled=true
pcrf.mongodb.uri=mongodb://localhost:27017/pcrf
```

- **MongoDB wifi-customer-cab:**

```
# wifi-customer-cab mongodb connector
wificab.mongodb.enabled=true
wificab.mongodb.uri=mongodb://localhost:27017/wifi-customer-cab
```

- **SORM2 (Mongo):**

```
# Eltex.SORM2.replicator MongoDB 'sorm2' connect
sorm2.mongodb.enabled=false
sorm2.mongodb.uri=mongodb://localhost:27017/sorm2
```

- **SORM2:**

```
# Eltex.SORM2.replicator host to use API
sorm2.enabled=false
sorm2.url=http://localhost:7071
sorm2.username=admin
sorm2.password=password
```

- **-3. " " " " Wi-Fi. " " " " Wi-Fi.**

```
#It enables records export to SORM3 while editing wifi users
sorm3.enabled=false
```

- **. 90000.**

```
# wifi-customer-cab request settings
wificab.timeout=90000
```

- **mongo, OTT:**

```
# ott mongodb connector
ott.mongodb.enabled=true
ott.mongodb.uri=mongodb://localhost:27017/ott
```

- **NBI SSO:**

```
# SSO settings
sso.enabled=false
sso.clientSecret=
sso.clientId=

# SSO REST
sso.rest.server.protocol=http
sso.rest.server.address=
sso.rest.server.port=80
sso.rest.server.timeout.sec=10
sso.rest.protocol.version=2.0
sso.rest.username=
sso.rest.password=

sso.rest.getToken.path=/apiman-gateway/b2b_test/getToken
sso.rest.getUserInfo.path=/apiman-gateway/b2b_test/getUserInfo
sso.rest.addUser.path=/apiman-gateway/b2b_test/addUser
sso.rest.updateUser.path=/apiman-gateway/b2b_test/updateUser
sso.rest.delUser.path=/apiman-gateway/b2b_test/delUser
sso.rest.addUserParam.path=/apiman-gateway/b2b_test/addUserParam
sso.rest.delUserParam.path=/apiman-gateway/b2b_test/delUserParam
sso.rest.getUserByName.path=/apiman-gateway/b2b_test/getUserByName
sso.rest.getUserByEmail.path=/apiman-gateway/b2b_test/getUserByEmail
sso.rest.resetPassword.path=/apiman-gateway/b2b_test/resetPassword
sso.rest.getUserByParam.path=/apiman-gateway/b2b_test/getUserByParam
```

- **eltex-portal:**

```
#####
#####DB ELTEX_PORTAL settings#####
#####
portal.db.driver=com.mysql.jdbc.Driver
portal.db.url=jdbc:mysql://localhost:3306/ELTEX_PORTAL?
max_allowed_packet=32362048&useUnicode=true&characterEncoding=utf8
portal.db.username=javauser
portal.db.password=javapasswor
```

- **eltex-ngw:**

```
# NGW
ngw.url=http://127.0.0.1:8040
```

- **eltex-doors:**

```
# DOORS
doors.url = http://localhost:9097/
doors.timeout = 60
doors.username = user
doors.password = password
```

- **API-Elvis (eltex-portal-constructor):**

```
# ELVIS
elvis.url=http://localhost:9001/epadmin/
elvis.timeout=60
```

/etc/eltex-radius-nbi/hikari.properties

```
eltex-radius-nbi  mysql  hikari cp.
```

- :

```
# ConnectionTimeout property controls the maximum number of milliseconds that a client (that's you) will wait for a connection from the pool.
connectionTimeout=15000
```

- , , connectionTimeout:

```
# ValidationTimeout property controls the maximum amount of time that a connection will be tested for aliveness.
validationTimeout=3000
```

- , . 10000(10), 600000 (600):

```
# IdleTimeout property controls the maximum amount of time that a connection is allowed to sit idle in the pool.
idleTimeout=30000
```

- , hikari cp :

```
# MinimumIdle property controls the minimum number of idle connections that HikariCP tries to maintain in the pool.
minimumIdle=1
```

eltex-radius-nbi. <http://localhost:8080/eltex-radius-nbi/asciidoc/>.

WSDL <http://localhost:8080/axis2/services/RadiusNbiService?wsdl>.

localhost ip-, NBI.