# Recovering from an LDAP expired certificate

Inspired by a real incident, this page suggests workarounds for an expired Active Directory LDAP certificate. The `socat` proxy technique at the end is, in the author's opinion, particularly neat.


- Active Directory SSL failures
- Fix the AD certificate
- Log in via Internal user
- Temporary Fixes
  - Configuring an alternative LDAP
  - Use a non-SSL port 389 LDAP
  - Use a SSL expired-cert-ignoring proxy


## Active Directory SSL failures

It's Friday, and JIRA users report being unable to log in. You check the JIRA logs, and find a colossal stacktrace, the relevant bits of which are:

```
...
Caused by: javax.naming.CommunicationException: activedirectory.example.com:636 [Root exception is javax.net.
ssl.SSLHandshakeException: sun.security.validator.ValidatorException: PKIX path validation failed: java.
security.cert.CertPathValidatorException: timestamp check failed]
    at com.sun.jndi.ldap.Connection.<init>(Connection.java:226)
...
Caused by: java.security.cert.CertificateExpiredException: NotAfter: Fri Jan 29 13:03:00 PST 2016
    at sun.security.x509.CertificateValidity.valid(CertificateValidity.java:274)
...
```

2016-01-29 15:37:34,393 ajp-nio-127.0.0.1-8009-exec-9 ERROR anonymous 937x265x1 1q2ecj2 10.60.11.128 /rest/gadget/1.0/login [c.a.j.security.login.
JiraSeraphAuthenticator] Error occurred while trying to authenticate user 'jsmith'.
com.atlassian.crowd.exception.runtime.OperationFailedException
    at com.atlassian.crowd.embedded.core.CrowdServiceImpl.convertOperationFailedException(CrowdServiceImpl.java:922)
    at com.atlassian.crowd.embedded.core.CrowdServiceImpl.authenticate(CrowdServiceImpl.java:81)
    at com.atlassian.crowd.embedded.core.DelegatingCrowdService.authenticate(DelegatingCrowdService.java:37)
    at com.atlassian.crowd.embedded.core.FilteredCrowdServiceImpl.authenticate(FilteredCrowdServiceImpl.java:51)
    at com.atlassian.jira.security.login.JiraSeraphAuthenticator.crowdServiceAuthenticate(JiraSeraphAuthenticator.java:75)
    at com.atlassian.jira.security.login.JiraSeraphAuthenticator.authenticate(JiraSeraphAuthenticator.java:49)
    at com.atlassian.seraph.auth.DefaultAuthenticator.login(DefaultAuthenticator.java:88)
    ... 12 filtered
    at com.atlassian.plugins.rest.module.servlet.RestSeraphFilter.doFilter(RestSeraphFilter.java:40)
    ... 72 filtered
    at java.util.concurrent.ThreadPoolExecutor.runWorker(ThreadPoolExecutor.java:1142)
    at java.util.concurrent.ThreadPoolExecutor$Worker.run(ThreadPoolExecutor.java:617)
    at org.apache.tomcat.util.threads.TaskThread$WrappingRunnable.run(TaskThread.java:61)
    at java.lang.Thread.run(Thread.java:745)
Caused by: org.springframework.transaction.CannotCreateTransactionException: Could not create DirContext instance for transaction; nested exception is
org.springframework.ldap.CommunicationException: activedirectory.example.com:636; nested exception is javax.naming.CommunicationException:
activedirectory.example.com:636 [Root exception is javax.net.ssl.SSLHandshakeException: sun.security.validator.ValidatorException: PKIX path validation
failed: java.security.cert.CertPathValidatorException: timestamp check failed]
    at org.springframework.transaction.compensating.support.AbstractCompensatingTransactionManagerDelegate.doBegin
(AbstractCompensatingTransactionManagerDelegate.java:90)
    at org.springframework.ldap.transaction.compensating.manager.ContextSourceTransactionManager.doBegin(ContextSourceTransactionManager.java:
126)
    at org.springframework.transaction.support.AbstractPlatformTransactionManager.getTransaction(AbstractPlatformTransactionManager.java:373)
    at com.atlassian.crowd.directory.SpringLDAPConnector.pageSearchResults(SpringLDAPConnector.java:376)
    at com.atlassian.crowd.directory.SpringLDAPConnector.searchEntitiesWithRequestControls(SpringLDAPConnector.java:476)
    at com.atlassian.crowd.directory.SpringLDAPConnector.searchUserObjects(SpringLDAPConnector.java:679)
    at com.atlassian.crowd.directory.SpringLDAPConnector.findUserWithAttributesByName(SpringLDAPConnector.java:628)
    at com.atlassian.crowd.directory.SpringLDAPConnector.findUserByName(SpringLDAPConnector.java:614)
    at com.atlassian.crowd.directory.SpringLDAPConnector.authenticate(SpringLDAPConnector.java:1098)
    at com.atlassian.crowd.directory.DbCachingRemoteDirectory.authenticateAndUpdateInternalUser(DbCachingRemoteDirectory.java:295)
    at com.atlassian.crowd.directory.DbCachingRemoteDirectory.performAuthenticationAndUpdateAttributes(DbCachingRemoteDirectory.java:231)
    at com.atlassian.crowd.directory.DbCachingRemoteDirectory.authenticate(DbCachingRemoteDirectory.java:203)
    at com.atlassian.crowd.manager.directory.DirectoryManagerGeneric.authenticateUser(DirectoryManagerGeneric.java:283)
    at com.atlassian.crowd.manager.application.ApplicationServiceGeneric.authenticateUser(ApplicationServiceGeneric.java:194)
    at com.atlassian.crowd.embedded.core.CrowdServiceImpl.authenticate(CrowdServiceImpl.java:69)
    ... 94 more
Caused by: org.springframework.ldap.CommunicationException: activedirectory.example.com:636; nested exception is javax.naming.
CommunicationException: activedirectory.example.com:636 [Root exception is javax.net.ssl.SSLHandshakeException: sun.security.validator.
ValidatorException: PKIX path validation failed: java.security.cert.CertPathValidatorException: timestamp check failed]
    at org.springframework.ldap.support.LdapUtils.convertLdapException(LdapUtils.java:108)

```
    at org.springframework.ldap.core.support.AbstractContextSource.createContext(AbstractContextSource.java:356)
    at org.springframework.ldap.core.support.AbstractContextSource.doGetContext(AbstractContextSource.java:140)
    at org.springframework.ldap.core.support.AbstractContextSource.getReadWriteContext(AbstractContextSource.java:175)
    at org.springframework.ldap.transaction.compensating.manager.ContextSourceTransactionManagerDelegate.getNewHolder
(ContextSourceTransactionManagerDelegate.java:96)
    ... 109 more
Caused by: javax.naming.CommunicationException: activedirectory.example.com:636 [Root exception is javax.net.ssl.SSLHandshakeException: sun.
security.validator.ValidatorException: PKIX path validation failed: java.security.cert.CertPathValidatorException: timestamp check failed]
    at com.sun.jndi.ldap.Connection.<init>(Connection.java:226)
    at com.sun.jndi.ldap.LdapClient.<init>(LdapClient.java:137)
    at com.sun.jndi.ldap.LdapClient.getInstance(LdapClient.java:1613)
    at com.sun.jndi.ldap.LdapCtx.connect(LdapCtx.java:2746)
    at com.sun.jndi.ldap.LdapCtx.<init>(LdapCtx.java:319)
    at com.sun.jndi.ldap.LdapCtxFactory.getUsingURLs(LdapCtxFactory.java:210)
    at com.sun.jndi.ldap.LdapCtxFactory.getLdapCtxInstance(LdapCtxFactory.java:153)
    at com.sun.jndi.ldap.LdapCtxFactory.getInitialContext(LdapCtxFactory.java:83)
    at javax.naming.spi.NamingManager.getInitialContext(NamingManager.java:684)
    at javax.naming.InitialContext.getDefaultInitCtx(InitialContext.java:313)
    at javax.naming.InitialContext.init(InitialContext.java:244)
    at javax.naming.ldap.InitialLdapContext.<init>(InitialLdapContext.java:154)
    at org.springframework.ldap.core.support.LdapContextSource.getDirContextInstance(LdapContextSource.java:42)
    at org.springframework.ldap.core.support.AbstractContextSource.createContext(AbstractContextSource.java:344)
    ... 113 more
Caused by: javax.net.ssl.SSLHandshakeException: sun.security.validator.ValidatorException: PKIX path validation failed: java.security.cert.
CertPathValidatorException: timestamp check failed
    at sun.security.ssl.Alerts.getSSLException(Alerts.java:192)
    at sun.security.ssl.SSLSocketImpl.fatal(SSLSocketImpl.java:1937)
    at sun.security.ssl.Handshaker.fatalSE(Handshaker.java:302)
    at sun.security.ssl.Handshaker.fatalSE(Handshaker.java:296)
    at sun.security.ssl.ClientHandshaker.serverCertificate(ClientHandshaker.java:1478)
    at sun.security.ssl.Handshaker.processLoop(Handshaker.java:979)
    at sun.security.ssl.Handshaker.process_record(Handshaker.java:914)
    at sun.security.ssl.SSLSocketImpl.readRecord(SSLSocketImpl.java:1050)
    at sun.security.ssl.SSLSocketImpl.performInitialHandshake(SSLSocketImpl.java:1363)
    at sun.security.ssl.SSLSocketImpl.startHandshake(SSLSocketImpl.java:1391)
    at sun.security.ssl.SSLSocketImpl.startHandshake(SSLSocketImpl.java:1375)
    at com.sun.jndi.ldap.Connection.createSocket(Connection.java:376)
    at com.sun.jndi.ldap.Connection.<init>(Connection.java:203)
    ... 127 more
Caused by: sun.security.validator.ValidatorException: PKIX path validation failed: java.security.cert.CertPathValidatorException: timestamp check failed
    at sun.security.validator.PKIXValidator.doValidate(PKIXValidator.java:352)
    at sun.security.validator.PKIXValidator.engineValidate(PKIXValidator.java:260)
    at sun.security.validator.Validator.validate(Validator.java:260)
    at sun.security.ssl.X509TrustManagerImpl.validate(X509TrustManagerImpl.java:324)
    at sun.security.ssl.X509TrustManagerImpl.checkTrusted(X509TrustManagerImpl.java:229)
    at sun.security.ssl.ClientHandshaker.serverCertificate(ClientHandshaker.java:1460)
    ... 136 more
Caused by: java.security.cert.CertPathValidatorException: timestamp check failed
    at sun.security.provider.certpath.PKIXMasterCertPathValidator.validate(PKIXMasterCertPathValidator.java:129)
    at sun.security.provider.certpath.PKIXCertPathValidator.validate(PKIXCertPathValidator.java:212)
    at sun.security.provider.certpath.PKIXCertPathValidator.validate(PKIXCertPathValidator.java:140)
    at sun.security.provider.certpath.PKIXCertPathValidator.engineValidate(PKIXCertPathValidator.java:79)
    at java.security.cert.CertPathValidator.validate(CertPathValidator.java:292)
    ... 142 more
Caused by: java.security.cert.CertificateExpiredException: NotAfter: Fri Jan 29 13:03:00 PST 2016
    at sun.security.x509.CertificateValidity.valid(CertificateValidity.java:274)
    at sun.security.x509.X509CertImpl.checkValidity(X509CertImpl.java:629)
    at sun.security.provider.certpath.BasicChecker.verifyTimestamp(BasicChecker.java:190)
    at sun.security.provider.certpath.BasicChecker.check(BasicChecker.java:144)
    at sun.security.provider.certpath.PKIXMasterCertPathValidator.validate(PKIXMasterCertPathValidator.java:119)
2016-01-29 15:37:47,070 ajp-nio-127.0.0.1-8009-exec-12 ERROR anonymous 937x271x1 1q2ecj2 10.60.11.128 /rest/gadget/1.0/login [c.a.c.manager.
application.ApplicationServiceGeneric] Directory 'ActiveDirectory' is not functional during authentication of 'jsmith'. Skipped.
```

Oops. The SSL certificate expired. You can't log into JIRA either. Now what?

# Fix the AD certificate

Get your friendly Windows administrator to wrestle with AD to fix the cert. I'm told that Atlassian's page (https://confluence.atlassian.com/display/CROWD/Configuring+an+SSL+Certificate+for+Microsoft+Active+Directory) is out-of-date, but perhaps better than nothing.

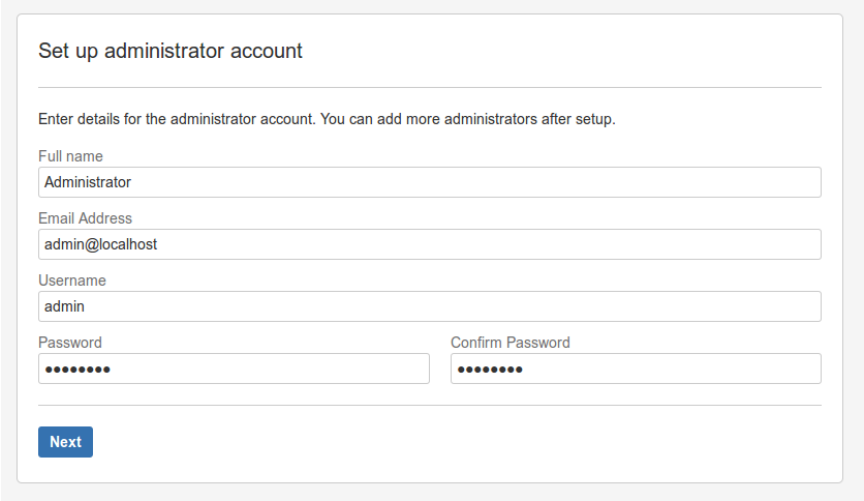
In the meanwhile, there's things we can do:

1. Ensure we can still log in as a non-LDAP admin user.
2. Either:
    a. If an alternative (e.g. replicated) LDAP/AD is available, switch JIRA to using it.
    b. If a non-SSL port 389 variant of LDAP is available, use that temporarily.
    c. Use a SSL-decrypting, expired-cert-ignoring LDAP wrapper using `socat`

Each of these is discussed below.

# Log in via Internal user

When JIRA/Confluence was initially set up, one of the setup steps was to configure an administrator user:




Let's call this the `admin` user (the actual name will depend on whatever your initial JIRA configurer chose). This `admin` account should be able to log in regardless of the status of AD/LDAP.

There are three situations where this account won't work:

- You've forgotten the `admin` user's password because it was so long ago.
  - If so, see reset a user password in the database..
- The 'Internal' directory has been disabled, or there is an identically named user (here, `admin`) in AD/LDAP, and the AD/LDAP directory is configured to be checked before the Internal directory.
  - In this case you'll need to reorder your user directories.

Once logged in as `admin` you'll be able to tweak settings for the User Directories.

# Temporary Fixes

Hopefully one of the next three alternatives will be available to you.

## Configuring an alternative LDAP

If your AD replicates elsewhere, your job is simple: go to the **User Directories** admin screen, edit the relevant directory (or add another) and use the replicated AD URL.

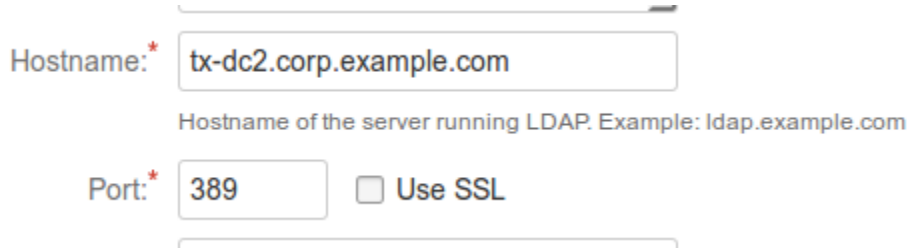
If LDAP breaks, it is sometimes useful to use ldapsearch from the command-line to verify queries. See Testing LDAP connectivity with ldapsearch.

## Use a non-SSL port 389 LDAP

Your LDAP server may be able to talk unencrypted on port 389 rather than port 636. If your network policy permits. Check that port 389 is open with `telnet` (and perhaps ldapsearch):

```
jturner@jturner-desktop ~ $ telnet tx-dc2.corp.example.com 389
Trying 10.0.10.100...
Connected to tx-dc2.corp.example.com.
Escape character is '^]'.
```

If this works, untick the 'Use SSL' box and try your luck:

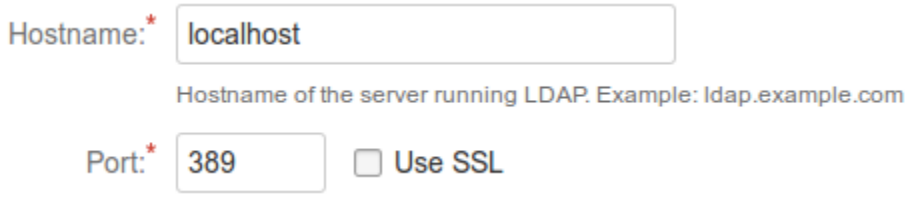



## Use a SSL expired-cert-ignoring proxy

You *know* the certificate is correct (just expired), and in a better world, JIRA would have an 'Ignore certificate errors' checkbox. As it doesn't, we have to get creative.

The magical `socat` Unix utility is able to create a `localhost` proxy LDAP that forwards requests to the real encrypted LDAP, but ignores the certificate errors.

```
openssl genrsa -out /tmp/client.key 1024
openssl req -new -key /tmp/client.key -x509 -days 2654 -out /tmp/client.crt
cat /tmp/client.key /tmp/client.crt > /tmp/client.pem
chmod 600 /tmp/client.pem
socat -v TCP4-LISTEN:389,reuseaddr,fork OPENSSL:tx-dc2.corp.example.com:636,cert=/tmp/client.pem,capath=/etc/ssl
/certs,verify=0
```

(This command assumes Debian/Ubuntu's `/etc/ssl/certs` directory of CA certs.)

Then we configure the User Directory to use the proxy: