

# Integrated authentication failed - PingFederate 7.2

Last updated by | Vitor Tomaz | Feb 18, 2021 at 2:30 AM PST

## Integrated authentication Fail - PingFederate 7.2

### Contents

- [Integrated authentication Fail - PingFederate 7.2](#)
  - [Issue](#)
  - [Analysis](#)
  - [Cause](#)
  - [Mitigation](#)
  - [Java Additional Parameters](#)
  - [Classification](#)



### Issue

Unable to login using AD integrated authentication. Customer using PingFederate 7.2 and AD connect.

### Analysis

- First error received is  
Failed to authenticate the user NT Authority\Anonymous Logon in Active Directory (Authentication=ActiveDirectoryIntegrated). Error code 0xCAA90017; state 10. Protocol is not supported by the client library. (Microsoft SQL Server). The reason is WS-Trust is required to be supported by the IdP for the SQL client driver case. PING team helped configure this on PING side to resolve this. Previously using SAML, now using WS Federation with WS-Trust Enabled.
- With the latest SSMS, we can connect using the AD universal authentication which use Modern Authentication (ADAL) workflow. However, it is only supported in SSMS as of now.
- Failed to authenticate the user NT Authority\Anonymous Logon in Active Directory (Authentication=ActiveDirectoryIntegrated).

Error code 0xCAA82F19; state 10 Revocavocor: 0) It is the certification problem as revocation failed. the TLS query is failing with the certificate "SSOLAB.lab.intranet". We got a new self-signed cert as original CA is down and this resolved this error.

- Next error we received:  
The federation server is configured exactly as specified here: <https://azure.microsoft.com/en-us/documentation/articles/active-directory-aadconnect-federation-compatibility/>   
<https://documentation.pingidentity.com/display/PF72/PingFederate+7.2>  Could not discover endpoint for Integrated Windows Authentication. Check your ADFS settings. It should support Integrate Windows

Authentication for WS-Trust 1.3 or WS-Trust 2005. In Marlon's case 116112714988511, adding the SPN of ADFS and restart the ADFS service resolved this error.

- Failed to authenticate the user [mxxxxxd@oya.com](mailto:mxxxxxd@oya.com) in Active Directory (Authentication=ActiveDirectoryPassword).  
Error code 0xCA82EFD; state 10 The attempt to connect to the server failed. (Microsoft SQL Server, Error: 0) From ADAL trace, we see failing to connect to <https://ssolab.lab.intranet/idp/sts.wst?TokenProcessorId=Office365TP> it is failing on it  
d:\bt\48176\sources\product\aal\aalclient.native\log.cpp(78) : atlTraceGeneral - (AdalDll)Sending request failed to <https://ssolab.lab.intranet/idp/sts.wst?TokenProcessorId=Office365TP> <wsdl:port name="CustomBinding\_IWSTrust13Async" binding="tns:CustomBinding\_IWSTrust13Async">  
<soap12:address location="https://ssolab.lab.intranet/idp/sts.wst?TokenProcessorId=Office365TP"/>  
<wsa10:EndpointReference> wsa10:Address<https://ssolab.lab.intranet/idp/sts.wst?TokenProcessorId=Office365TP></wsa10:Address> Internally as well this URL <https://ssolab.lab.intranet/idp/sts.wst?TokenProcessorId=Office365TP> is not accessible and that's the reason for us getting the Error
- Your AAD team checked with Ping tech support and made some changes on our end. Now the URL gets resolved internally. [https://ssolab.lab.intranet:9031/pf/sts\\_mex.ping?PartnerSpId=urn:federation:MicrosoftOnline](https://ssolab.lab.intranet:9031/pf/sts_mex.ping?PartnerSpId=urn:federation:MicrosoftOnline)
- Failed to authenticate the user lab\mmoore in Active Directory (Authentication=ActiveDirectoryPassword).

Error code 0xCA30194; state 10 The server has not found anything matching the requested URI (Uniform Resource Identifier). (Microsoft SQL Server, Error: 0) From fiddler, we see We reached the ping server and authentication worked fine and Ping server has issued a token for the Azure AD.

From ADAL log, we see **d:\bt\48176\sources\product\aal\aalclient.native\atllogger.cpp(33) : atlTraceGeneral - WSTrust response does not have recognized SAML assertion.** Looks like the issue is with the namespace used in the change.

Here is what you have in the token <wst13:RequestSecurityTokenResponseCollection  
xmlns:wst13="http://docs.oasis-open.org/ws-sx/ws-trust/200512"><wst13:RequestSecurityTokenResponse>  
<wst13:TokenType>

Here is what we are expecting. "xmlns:trust='http://docs.oasis-open.org/ws-sx/ws-trust/200512'"

You did following change per PING's suggestion:

- Navigate to ~/pingfederate/server/default/data/config-store folder.
  - Open org.sourceid.websso.profiles.WsTrustStsMetadataRequestCreator.xml in a text editor.
  - Set the 'Enabled' value to true in the urn:federation:MicrosoftOnline section. The end result should look like this:  

```
<con:map name="urn:federation:MicrosoftOnline">
  <con:item name="StsWsdI">sts_o365.xml</con:item>
  <con:item name="UsernamePolicyTemplate">username_o365.xml</con:item>
  <con:item name="Enabled">true</con:item>
</con:map>
```
  - Restart the PingFederate server
- After the above change, we see the final error:

Error code 0xCA20003;state 10 ID3242: The security token could not be authenticated or authorized.  
(Microsoft SQL Server)

From fiddler, we are posting for the request for token but Azure AD is failing with Invalid signature.

## Cause

The response from PING Server - "POST <https://ssolab.lab.intranet:9031/idp/sts.wst?TokenProcessorId=Office365TP>" contains a bunch of line-feed (LF) characters in it. When we read that in to a string on a Windows machine, that LF automatically gets converted to two characters, carriage-return(CR) + line-feed(LF), which is what Windows uses in strings. Because of that change from LF to CR+LF, the signature now no longer matches and it is rejected when sent to [login.windows.net](https://login.windows.net). That is the root of the problem.

We always get the entire Envelope as one long string with no CR or LF in them from federation servers and have never had this issue. If they can prevent their code from doing the formatting (or whatever is adding the LF's) before signing, then it should go through fine

## Mitigation

PING team helps make following change and we are able to login using AD integrated authentication.

### Omit Line Breaks in Digital Signatures

Configure PingFederate to omit line breaks in digital signatures by using one of the following procedures.

**Note:** This change is global, for all cases in which PingFederate may write encoded signatures to XML or log files.

#### On Windows (running PingFederate from the command line):

1. Open <pf\_install>/pingfederate/bin/run.bat in a text editor.
2. Locate the variable PF\_JAVA\_OPTS
3. Add -Dorg.apache.xml.security.ignoreLineBreaks=true as a variable value.

#### On Windows (running PingFederate as a service):

4. Open <pf\_install>/pingfederate/sbin/wrapper/PingFederateService.conf in a text editor.

Example: wrapper.java.additional.9=-Dorg.apache.xml.security.ignoreLineBreaks=true

5. Locate the heading:

## Java Additional Parameters

3. Add -Dorg.apache.xml.security.ignoreLineBreaks=true as a variable value below the heading.

#### On Linux/Unix (running PingFederate from the command line or as a service):

4. Open <pf\_install>/pingfederate/bin/run.sh in a text editor.
5. Locate an instance where the environment variable JAVA\_OPTS is set.
6. Add -Dorg.apache.xml.security.ignoreLineBreaks=true as a variable value.

[https://docs.pingidentity.com/bundle/O365IG20\\_sm\\_integrationGuide/page/O365IG\\_c\\_omitLineBreaksInDigSigs.html](https://docs.pingidentity.com/bundle/O365IG20_sm_integrationGuide/page/O365IG_c_omitLineBreaksInDigSigs.html)

Real Scenario - Customer Case: Issue Connecting with integrated auth

**TITLE: Connect to Server**

Cannot connect to tcp:dixita-west-us.database.windows.net.

**ADDITIONAL INFORMATION:**

Failed to authenticate the user NT Authority\Anonymous Logon in Active Directory (Authentication=ActiveDirectoryIntegrated).

Error code 0xCA20003; state 10

AADSTS70002: Error validating credentials. AADSTS50008: SAML token is invalid.

There are a few reasons why a SAML Token is invalid error occurs:

- Issuer in token doesn't match the configuration of the federated domain
  - The token signing certificate used to sign the token doesn't match the configuration of the federated domain
  - Immutable ID value doesn't match the configuration of the user object in AAD
- This customer had windows data center 2012 which was not domain join complaint with azure AD. Because of which the users with [scalarch.com](https://scalarch.com) domain were not able to access his SQL instances.

**Classification**

Root cause Tree - Connectivity/AAD Issue/AAD user Configuration

**How good have you found this content?**

-