Web Services and Applications Deployment Guide

SAML authentication

12/19/2025

SAML authentication

Contents

Web Services supports Security Assertion Markup Language (SAML) for single sign-on (SSO) authentication to the Agent Desktop and custom integrations.

Configuring SAML

To enable SAML, make the following configuration changes in the serverSettings section of the **application.yaml** file on each of your Web Services nodes:

Start

1. Set the following options in the SSL and CA section:
   - caCertificate — should point to a JKS key storage that includes the SAML encryption key. See Generating security keys for details.
   - jksPassword — should be the password for the **caCertificate** key storage.
2. Set the following option in the SAML section:
   - samlSettings — the following properties are mandatory:
     - encryptionKeyName
     - signingKeyName
     - identityProviderMetadata
3. Save the changes to the file. Your configuration should look something like this:

```
# SSL and CA
caCertificate: /Users/samluser/Documents/Keys/keystore.jks
jksPassword: password

# SAML
samlSettings:
    serviceProviderEntityId: genesys.staging.GWS
    encryptionKeyName: client
    signingKeyName: client
    identityProviderMetadata: /Users/samluser/Documents/Metadata/idp-metadata.xml
```

4. To activate SAML authentication, append the browser URL for Workspace Web Edition with `?authType=saml`.
5. To enable extended SAML logging, add the following string to **logback.xml** file: `<logger name="org.springframework.security.saml2" level="%LEVEL%"/>`, where valid values for LEVEL are INFO (preferred) or DEBUG.

End

Generating security keys

You can use the keytool utility that comes with the Java SDK to generate a JKS key store. Use the following command:

```
keytool -genkey -keystore <path_to_jks_file> -alias <key_name> -keypass <key_password>
-storepass <store_password> -dname <distinguished_name>
```

If you already have a JKS key store, you can add a key to it by executing the command above with the same file name and the new key name and key password. For example:

```
keytool -genkey -keystore /opt/keystore.jks -alias encryption_key -keypass genesys -storepass
genesys -dname "CN=GWS, OU=R&D, O=Genesys, L=Daly City, S=California, C=US"
```

SAML API mappings between GWS 8.5 and GWS 8.6

The SAML API endpoints in GWS 8.5 must be mapped to its equivalent SAML 2.0 API endpoints in GWS 8.6.

SAML (GWS 8.5)	SAML 2.0 (GWS 8.6)
/saml/login	/saml2/authenticate/gws
/saml/SSO	/login/saml2/sso/gws
/saml/logout	/saml2/logout
/saml/SingleLogout	/logout/saml2/slo
/saml/metadata	/saml2/service-provider-metadata/gws

Backward compatibility

To ensure backward compatibility, configure a load balancer such as nginx to map legacy SAML endpoints to the new SAML 2.0 APIs when routing requests to GWS 8.6.

Example configuration:

```
location /saml/login {
  proxy_pass https://gws-host:8443/saml2/authenticate/gws;
  proxy_set_header Host $host;
  proxy_set_header X-Real-IP $remote_addr;
}
```

Next step

- Back to Configuring security