Barbarians at the Gate
OE Authentication
Gateway

Roy Ellis
ellis@progress.com
Agenda

- What is the OpenEdge Authentication Gateway
- Installing and configuring the OpenEdge Authentication Server
- Configuring and Enabling the Database for the OpenEdge Authentication Gateway
- Additional Steps
- The Login Process
And now, for something completely different...
Barbarians at the Gate

NOW GO AWAY

OR I SHALL TAUNT YOU A SECOND TIME
OpenEdge Authentication Gateway is a set of features that ensure trusted identity management. The right users get the right access to the right information.

Authentication and Authorization access to the OpenEdge database.

Authentication for PASOE and OE Management
What are the features?

- The OpenEdge Authentication Server
- Database enabled to enforce use of OpenEdge Authentication Server
- OpenEdge Client-Principal creation and verification
- Client-Server key verification
- Tools to help configure and debug
Security before OpenEdge Authentication Gateway

ABL and SQL database security handled by the clients

ABL & SQL Client
// Directly connects to the db
// Code-enforced authentication
// Code-enforced authorization

Rogue ABL & SQL Clients
// I’m in the database- awesome!
Installing and configuring the OpenEdge Authentication Server
Install the OpenEdge Authentication Gateway Server

- ABL Clients
- Application Server
- SQL Clients

OpenEdge Authentication Gateway Server

OpenEdge Database
Installation checklist

- Bundled with the Progress Production Application Server for OpenEdge listed as a separate control code
Why provide a server instance?

- The following changes were made
  - Both Tomcat and OpenEdge remote administration web applications were removed
  - Standard PAS for OpenEdge administrator user accounts have been hardened by inserting digested passwords in the `conf/tomcat-users.xml` file
  - In this instance, the ROOT web application is replaced with one that always returns a blank response
    - While inserting another ABL application into an Authentication Gateway can be done, it is not recommended
Configuration through properties files

- The `conf` directory contains the following properties files for the instance:
  - `opendge.properties`
  - `logging.properties`
  - `jvm.properties`
  - `catalina.properties`
  - `appservers.properties`

- The `config` directory is for the security token service:
  - `domains.README`
  - `users.properties`
  - `sts.properties`
  - `mydomains.csv`
  - `domains.csv`
  - `domains.keystore`
  - `domainsORIG.json`
  - `domains.schema.json`
  - `domains.json`
OpenEdge Authentication Server

- The OpenEdge Authentication Gateway Server is a PASOE application
  - So you manage and configure much of the server like any PASOE instance
  - If you purchase a PASOE license you can get OE Auth Gateway license

- Installation
  - Install on a separate machine or with the database

- Start and verify that the Authentication Server is working
  - Test with ‘stsclientutil’ tool
      –user test –password test
Starting the Authentication Gateway

- From your instance directory, start the server

```
proenv> bin\tcman start
```

```
proenv> bin\tcman start
Using CATALINA_BASE: "C:\OpenEdge\WRK\oauthserver"
Using CATALINA_HOME: "C:\Progress\OpenEdge\servers\pasoe"
Using CATALINA_TMPDIR: "C:\OpenEdge\WRK\oauthserver\temp"
Using JRE_HOME: "C:\Progress\OpenEdge\jdk"
Using CLASSPATH: ";C:\Progress\OpenEdge\servers\pasoe\bin\bootstrap.jar;C:
:\Progress\OpenEdge\servers\pasoe\bin\tomcat-juli.jar;C:\Progress\OpenEdge\serv
ers\pasoe\bin\tomcat-juli.jar"
PAS started with process IDs: 1888 3084
```
Tools for testing the service - stsclientutil

proenv> stsclientutil -help

usage: stsclientutil [options]
      where [options] (required)
      * -cmd <cmd> = [ping | authenticate | exchange]
      * -url <baseUrl>
      -user <user[@domain]>
      -domain <domain>
      -sslversion <SSL version> = [TLSv1 | SSLv2 | SSLv3 | TLSv1.0 | TLSv1.1 | TLSv1.2]
      -sslCiphers <SSL Ciphers>
      -installpath <oeInstallPath>
      -keyStorepath <keyStorePath>
      -certstorepath <certStorePath>
      -clientkeyHdrc <clientKeyHdrName>
      -logginglevel <loggingLevel>
      -nohostverify
      -help
Pinging the Authentication Gateway

- From your install directory, using the ping command

```
proenv> C:\Progress\OpenEdge\bin\stsclientutil
       -url https://yourmachinename:8443
       -cmd ping
       -nohostverify
```

- You are not using security, just testing that the server is available for testing
Connect to a 3rd Party authentication service

ABL Clients

Application Server

SQL Clients

OpenEdge Authentication Gateway Server

OpenEdge Database

3rd Party

© 2017 Progress Software Corporation and/or its subsidiaries or affiliates. All rights reserved.
OpenEdge Authentication Server - continued

- Configure OpenEdge Authentication Server to use your 3rd party authentication product
  - LDAP
  - Active Directory
  - Operating System
  - Users file (default set up for quick validation)

- How?
  - Configure sts.properties
  - Configure domains in the domains.json
  - Create an encrypted keystore with domains access code
Authenticating a user through the Authentication Gateway before locking the database

c:\Progress\OpenEdge\bin\stsclientutil
-url https://HOSTNAME:8443
-cmd authenticate
-user test
-password test
-nohostverify

Authentication succeeded
Configuring and enabling the database for the OpenEdge Authentication Server
Configure your database to use OpenEdge Authentication Gateway Server

ABL Clients

Application Server

SQL Clients

OpenEdge Database

OpenEdge Authentication Gateway Server

3rd Party
Configure the OpenEdge Database

- Test the connection from database machine to OpenEdge Authentication Server
  - `stsclientutil --url https://hostname:port --nohostverify --cmd ping`
  - `stsclientutil --url https://hostname:port --nohostverify --cmd authenticate
    --user user@domain --password password`

- Add a security domain
  - Add at least 2 security administrators in the security domain
    - Including one that matches your LDAP, Active Directory, or Operating System
  - Add domains that match the domains in your OpenEdge Authentication Server `domains.json`
  - Enter the access code for the domain that matches the domains keystore in your OpenEdge Authentication Server
Configure the OpenEdge Database - continued

- Add the OpenEdge Authentication Server URL to the database

- Enable the database to use the OpenEdge Authentication Gateway
  - proutil database –C enableauthgateway

- Start the database
  - proserve dbname –nohostverify –S –H

- Verify you can log into the database
  - mpro dbname –U user@domain –P password
Additional steps
Create a Server Key and install Client Keys

ABL Clients

Application Server

SQL Clients

OpenEdge Database

OpenEdge Authentication Gateway Server

HTTPS

3rd Party
More Security Features

- Create a web server certificate with valid hostname
  - Now you can stop using \texttt{--nohostverify}
- Create a client-server key set for the OpenEdge Authentication Server
- Server key
  - \texttt{stskeyutil create \textbar url} \url{https://hostname:port}
    - Enter password
    - Enter access-code
  - Modify \texttt{sts.properties}
    - Enable server key
    - Set path and file name
    - Enter access-code
  - Restart OpenEdge Authentication Server
Creating a domain that will use the key

- Edit the reserved `oests.server` domain defined in `domains.json` to enabled
  - Set `enabled` to true
  - Confirm `sso` is true
  - Confirm `authProvider` is anonymous
Setting `sts.properties` to use the keystore

- **Edit `sts.properties`**
  - Enable the use of server keys
    ```properties
    # set this to true to require the use of the server key file configuration. Defaults to false.
    sts.server.key.enabled=true
    ```
  - Tell it where to find the key file
    ```properties
    # STS server key file location. Specified if client key SSO validation is enabled. Default name is oests-key.ecp.
    # Specific full path to file to ensure it can be found properly.
    # make sure to use forward slashes even on windows. Must be provided if server key enabled is true.
    sts.server.keyfile=c:/OpenEdge/WRK/oeauthserver/conf/oests-key.ecp
    ```
  - Provide the password for the key, encryption can be added
    ```properties
    # STS server key file password. Must be provided if server key file is specified. May be be oech1:: encoded
    sts.server.keyfilepassword=password
    ```
More Security Features - continued

- Install the client key on the database machine
  - Copy the server key on the database machine
  - Install the client key from the server key
    - `stskeyutil install -url https://hostname:port -file serverkeyname`
    - Enter password when prompted
- Test OpenEdge Authenticate server connection using `stsclienutil`
- **Delete the server key from `client/database machine`**
- Copy the new web certificate to the database machine
- Restart your database
- Follow the steps above to add the client key to all client installs
Limit database access through connection roles

- ABL Clients
- Application Server
- SQL Clients
- OpenEdge Database
- HTTPS
- OpenEdge Authentication Gateway Server
- 3rd Party
Even more security features - continued

- Add connection role authorization to database

- Enable
  - stsconnroleutil enable –db dbname –U user@domain –P password
    - Now only user@domain can connect to the database

- Add another user
  - stsconnroleutil grantuser –grantee anotheruser@domain –db dbname
    –U user@domain –P password

- Add a list of users
  - stsconnroleutil grantuser –file addusers.list –db dbname –U user@domain –P password
Using Roles

- Add users
- Enable the use of roles

```
proenv> stsconnroleutil enable
    -db sports
    -U administrator@local
    -P NAPug2017
```

- Grant roles

```
proenv> stsconnroleutil grantuser
    -grantee test1@local
    -can yes
    -db sports
    -U administrator@local
    -P NAPug2017
```
Even more security features - continued

- Add policies
  - Using 4GL code refine policies even further

- Add events
  - Fire events using 4GL code

- Invoke –processid
  - Use current login user
  - mpro –db dbname –OSUser –domain domain
The login process
The OpenEdge Authentication Gateway Process

ABL Clients → Application Server → OpenEdge Database → OpenEdge Authentication Gateway Server

SQL Clients → Application Server → OpenEdge Database

HTTPS

3rd Party
The OpenEdge Authentication Gateway Process

- ABL Clients
- Application Server
- SQL Clients
- OpenEdge Database
- OpenEdge Authentication Gateway Server

SSL/TLS

3rd Party
The OpenEdge Authentication Gateway Process

ABL Clients → Application Server → OpenEdge Database → OpenEdge Authentication Gateway

SQL Clients → Application Server → OpenEdge Database → OpenEdge Authentication Gateway Server

3rd Party

HTTPS
The OpenEdge Authentication Gateway Process

ABL Clients -> Application Server -> OpenEdge Database

SQL Clients

OpenEdge Authentication Gateway Server

HTTPS

3rd Party
The OpenEdge Authentication Gateway Process

ABL Clients → Application Server → OpenEdge Database → HTTPS → 3rd Party

OpenEdge Authentication Gateway Server

SQL Clients
The OpenEdge Authentication Gateway Process

- ABL Clients
- Application Server
- SQL Clients
- OpenEdge Authentication Gateway Server
- OpenEdge Database
- 3rd Party

HTTPS
The OpenEdge Authentication Gateway Process

- ABL Clients
- Application Server
- SQL Clients

OpenEdge Database

 строки

3rd Party

_ db.connect

OpenEdge Authentication Gateway Server

HTTPS

© 2017 Progress Software Corporation and/or its subsidiaries or affiliates. All rights reserved.
The OpenEdge Authentication Gateway Process

- ABL Clients
- Application Server
- SQL Clients
- OpenEdge Authentication Gateway Server
- OpenEdge Database

ABL Clients → Application Server → SQL Clients

Encryption via HTTPS:

- HTTPS connection
- STS (Security Token Service)

_A_db.connect_

3rd Party
New for 11.7.2
OpenEdge Authentication Gateway - PASOE authenticates to OE AuthGateway

- PASOE
- ABL Clients
- OpenEdge SQL Server
- OpenEdge Database
- Directory Information Service (e.g. LDAP, Active Directory)
- STS
- OpenEdge Authentication Gateway
- ODBC
- JDBC
- © 2017 Progress Software Corporation and/or its subsidiaries or affiliates. All rights reserved.
PASOE to OEAuhtGateway Server

- PASOE can use the OEAuhtGateway Server for authentication
- oeablSecurity.properties(.README)
Summary
OpenEdge Authentication Gateway

- Protects your database from unauthorized access
- Separation of duties
- Takes creating and validating client principals out of the ABL
- Allows database administrators the ability to manage users authorization
- Auditing via Events
- Ability to refine authorization through Policies
- Never have to write authentication code again!