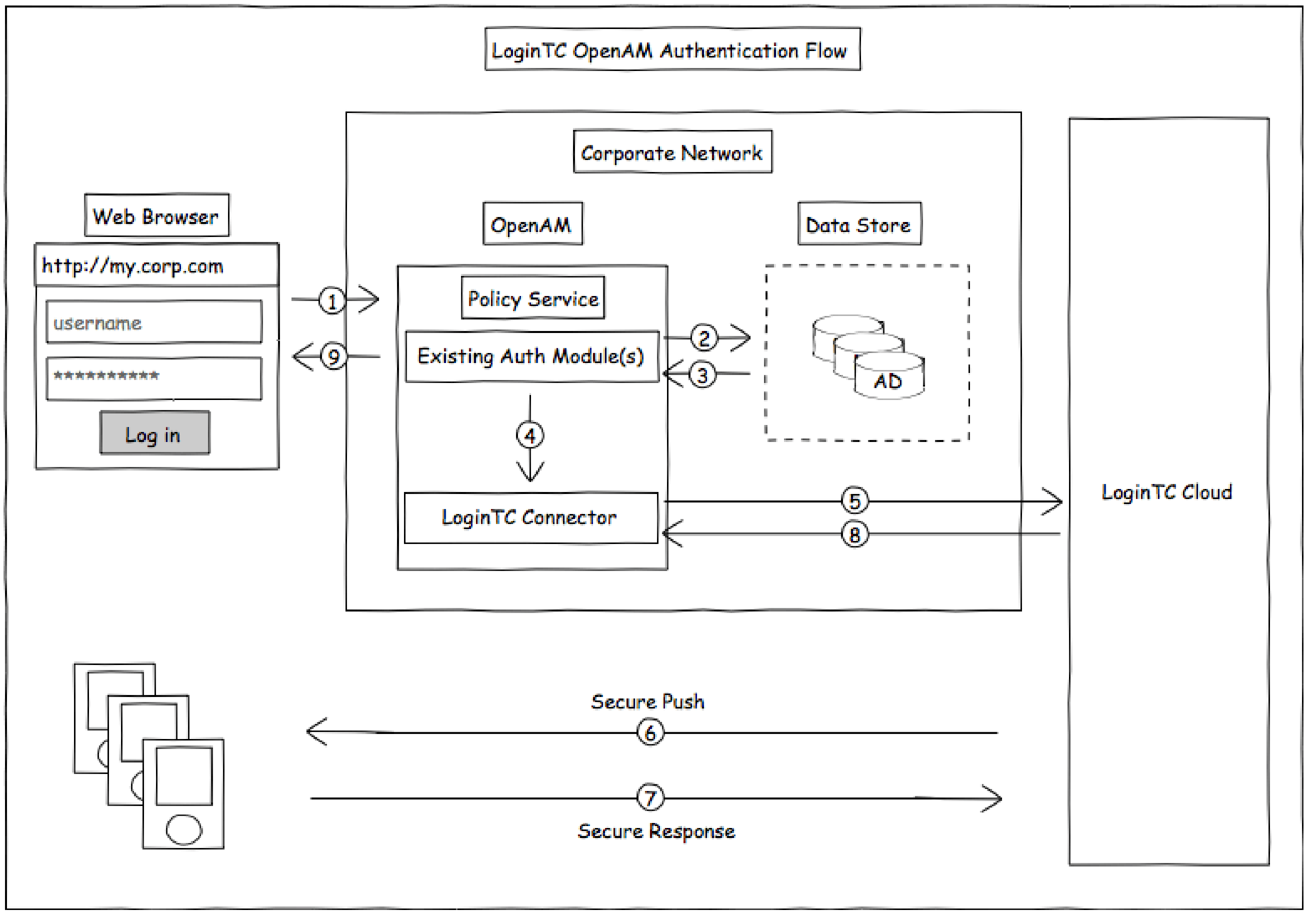
OpenAM Two-Factor Authentication (2FA)

logintc.com/docs/connectors/openam

Introduction

LoginTC OpenAM Connector allows administrators to incorporate two-factor authentication within their OpenAM authentication policies.

The following diagram illustrates a seamless LoginTC OpenAM Connector integration into an existing corporate network protected by OpenAM. The circled numbers indicate the step in a typical authentication flow.



The LoginTC OpenAM Authentication module is installed in your OpenAM server. It can be configured as a stand-alone policy or added to an existing authentication chain. When a user attempts to access a protected resource, the request is intercepted and the configured authentication policy is applied. When the LoginTC Authentication module is activated an out of band request is sent to the users mobile app. The request will launch the LoginTC app and gives the user an opportunity to approve or deny the request and then enter their

PIN or passcode. The user's selection is sent back to module which will succeed in the OpenAM authentication chain if the credential was unlocked, otherwise the authentication will fail.

Enterprise subscription required
Please contact our sales team for trial access to the LoginTC OpenAM Connector.

Prerequisites
Before proceeding, please ensure you have the following:

1. **LoginTC Cloud Administrator Account**An administrator account allows you to perform management, maintenance and monitoring of your own LoginTC organization. An organization is where you manage domains, users, add and remove administrators and various other settings.
   Sign Up for an administrator account at LoginTC Admin Panel.

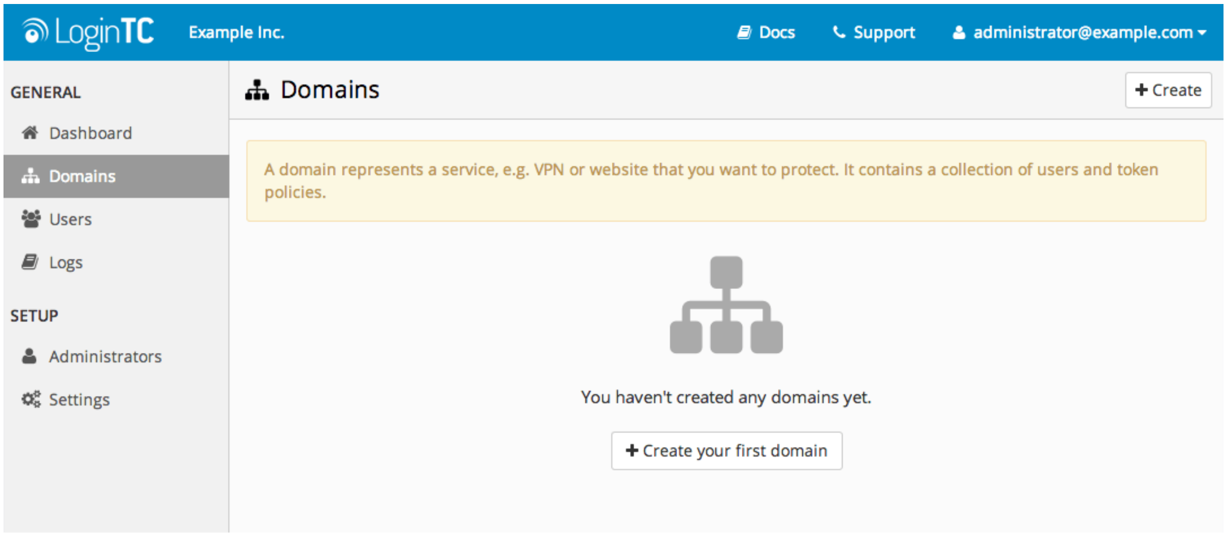
2. **OpenAM Installation**If you do not already have OpenAM, full installation instructions can be found on ForgeRock's wiki: OpenAM Installation. This installation guide was written for OpenAM 10.1.0.
3. **User Data Store, e.g. Active Directory, LDAP Server, OpenDJ**This datastore is the same as the one configured for your OpenAM server. Users from your data store will be synchronized with LoginTC Admin. Passwords are not stored in LoginTC Admin, only information about users like name, username and email.

OpenAM Domain Creation
Protecting a corporate network or application is done by creating a new domain in your organization. There are various types of domains specific to what you are trying to protect. In this case, since the protected resource is controlled by OpenAM, use an OpenAM domain.

Steps to create a new OpenAM domain:

1. Log in to LoginTC Admin
2. Click **Domains**:

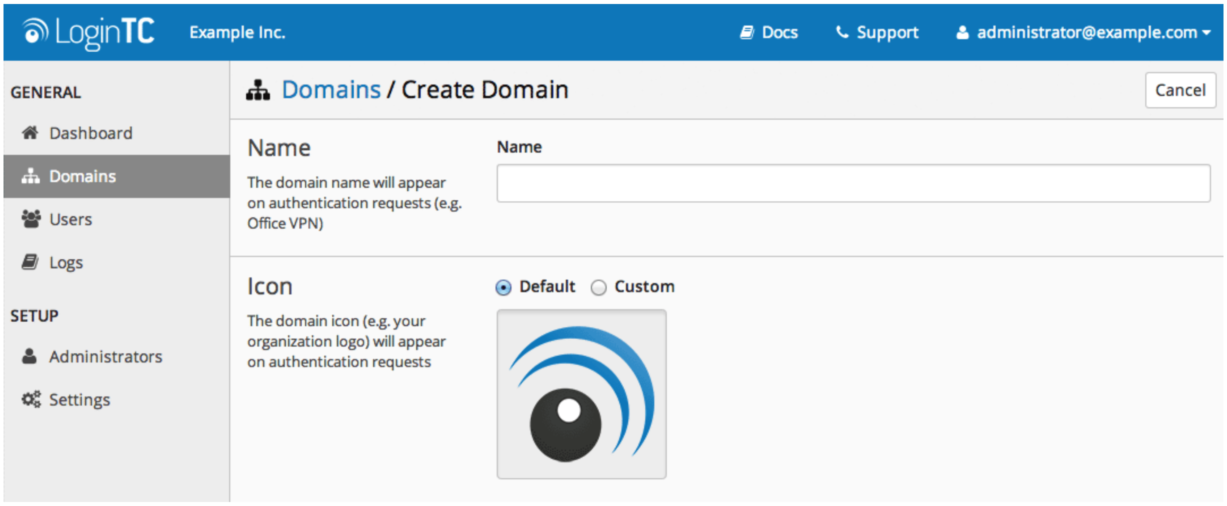
3. Click **Add Domain**:



A domain represents a service, e.g. VPN or website that you want to protect. It contains a collection of users and token policies.

You haven't created any domains yet.

4. Enter a name and optionally pick an icon



Name

The domain name will appear on authentication requests (e.g. Office VPN)

Name

Icon

The domain icon (e.g. your organization logo) will appear on authentication requests

● Default ○ Custom

5. Scroll down and click **Create**

Use Default Domain Settings

Domain settings can be modified at any time by navigating to **Domains > Your Domain > Settings**.

Installation

The LoginTC OpenAM Connector contains:

- OpenAM Authentication module
- OpenAM REST datastore client
- Java WAR utility to manage and sync data stores with LoginTC Admin

Before you begin, log into your OpenAM server via ssh:

1. Create a LoginTC directory:

   ```
   mkdir -p /opt/logintc
               cd /opt/logintc
   ```

2. Copy the latest LoginTC OpenAM Connector to /opt/logintc on your OpenAM server and unzip

   ```
   unzip logintc-openam-connector-x.x.x.zip
   ```

Enterprise subscription required

Please contact our sales team for trial access to the LoginTC OpenAM Connector.

Configuration for OpenAM Authentication

Run Install Script

Before you begin, log into your OpenAM server via ssh:

1. Go to LoginTC directory where the connector was downloaded

   ```
   cd /opt/logintc
   ```

2. Stop your application server, i.e. tomcat

   ```
   sudo service tomcat stop
   ```

3. Run install command

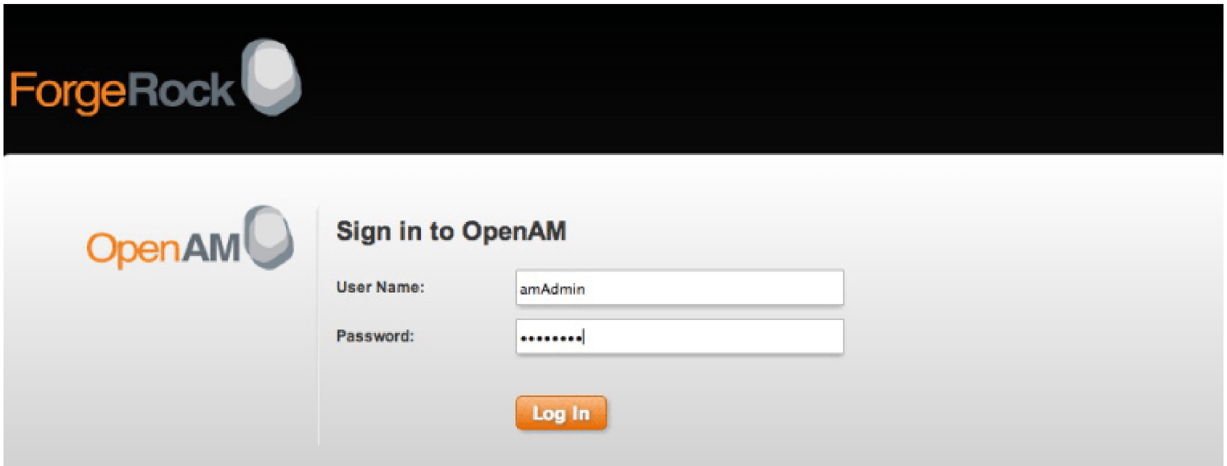
   ```
   ./install.sh /opt/logintc/logintcauth.jar /usr/share/tomcat/webapps/openam

               First argument is the path to the `logintcauth.jar`
               Second argument is the path to the OpenAM exploded WAR
   ```

4. Start your application server, i.e. tomcat

   ```
   sudo service tomcat start
   ```

Enable Authentication Module

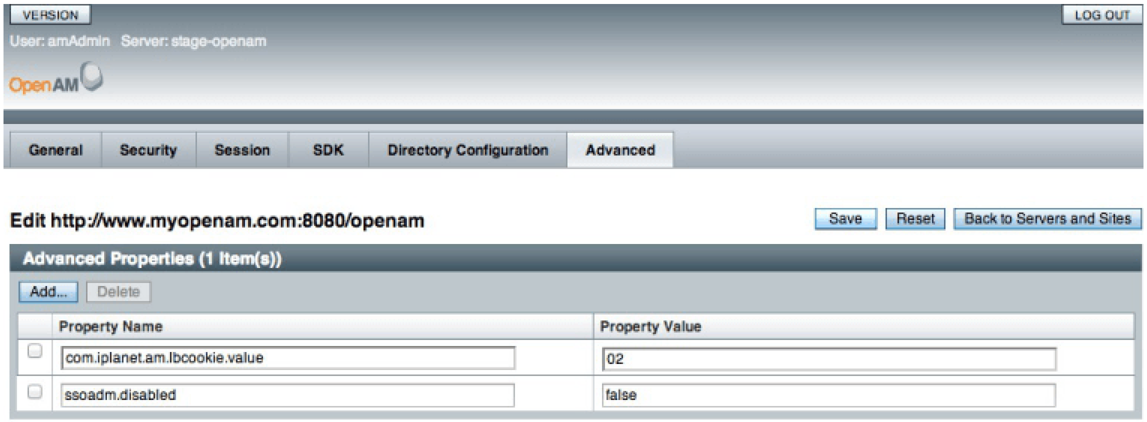
1. Navigate to your OpenAM installation, `http://www.myopenam.com:8080/openam`



User Name: amAdmin

Password: ••••••••

2. Log in using administrator credentials
3. Enable ssoadm, the configuration of core services:
    1. Click on Configuration > Servers and Sites tab
    2. Click on your server under Servers
    3. Click on the Advanced tab
    4. Click the Add button and add a new entry: `ssoadm.disabled` with value `false`
    5. Save



User: amAdmin Server: stage-openam

Edit http://www.myopenam.com:8080/openam

Advanced Properties (1 Item(s))

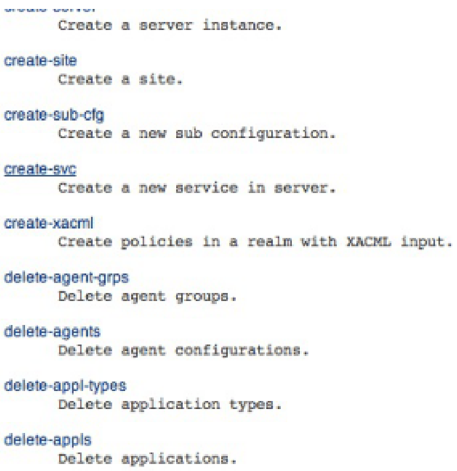
	Property Name	Property Value
□	com.iplanet.am.lbcookie.value	02
□	ssoadm.disabled	false

4. Restart your application server, i.e. tomcat

```
sudo service tomcat restart
```

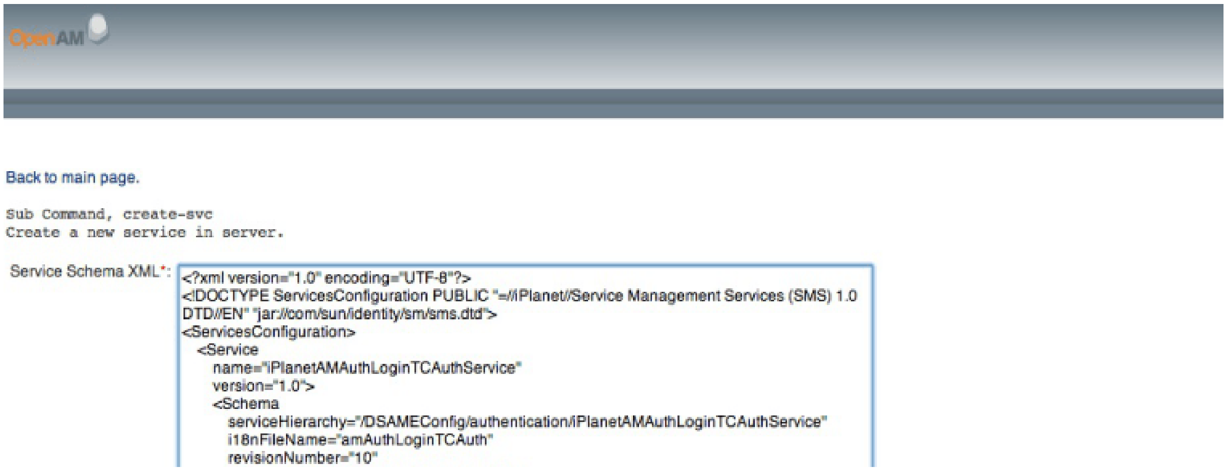
5. Navigate to `http://www.myopenam.com:8080/openam/ssoadm.jsp`

6. Click create-svc



```
        Create a server instance.

create-site
        Create a site.

create-sub-cfg
        Create a new sub configuration.

create-svc
        Create a new service in server.

create-xacml
        Create policies in a realm with XACML input.

delete-agent-grps
        Delete agent groups.

delete-agents
        Delete agent configurations.

delete-appl-types
        Delete application types.

delete-appls
        Delete applications.
```

7. Copy the contents from `/opt/logintc/amAuthLoginTCAuth.xml` and paste it in the form and click Submit



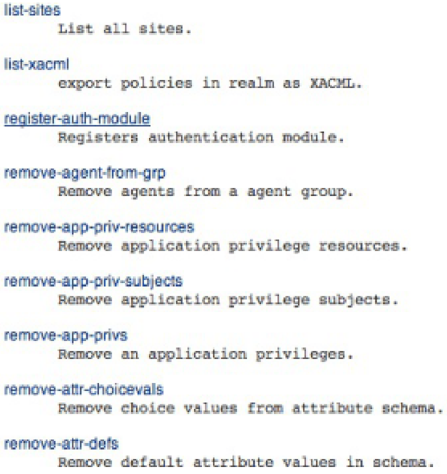
Back to main page.

```
Sub Command, create-svc
Create a new service in server.
```

Service Schema XML*:

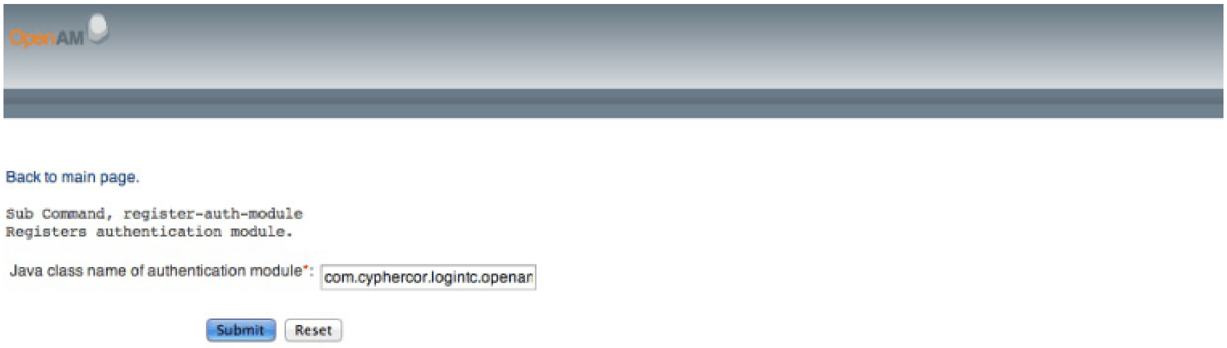
```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE ServicesConfiguration PUBLIC "=//iPlanet//Service Management Services (SMS) 1.0
DTD//EN" "jar://com/sun/identity/sm/sms.dtd">
<ServicesConfiguration>
  <Service
    name="iPlanetAMAuthLoginTCAuthService"
    version="1.0">
    <Schema
      serviceHierarchy="/DSAMEConfig/authentication/iPlanetAMAuthLoginTCAuthService"
      i18nFileName="amAuthLoginTCAuth"
      revisionNumber="10"
```

8. Navigate to `http://www.myopenam.com:8080/openam/ssoadm.jsp`
9. Click register-auth-module



```
list-sites
        List all sites.

list-xacml
        export policies in realm as XACML.

register-auth-module
        Registers authentication module.

remove-agent-from-grp
        Remove agents from a agent group.

remove-app-priv-resources
        Remove application privilege resources.

remove-app-priv-subjects
        Remove application privilege subjects.

remove-app-privs
        Remove an application privileges.

remove-attr-choicevals
        Remove choice values from attribute schema.

remove-attr-defs
        Remove default attribute values in schema.
```

10. Type in `com.cyphercor.logintc.openam.LoginTCAuth` and click Submit



Back to main page.

```
Sub Command, register-auth-module
Registers authentication module.
```

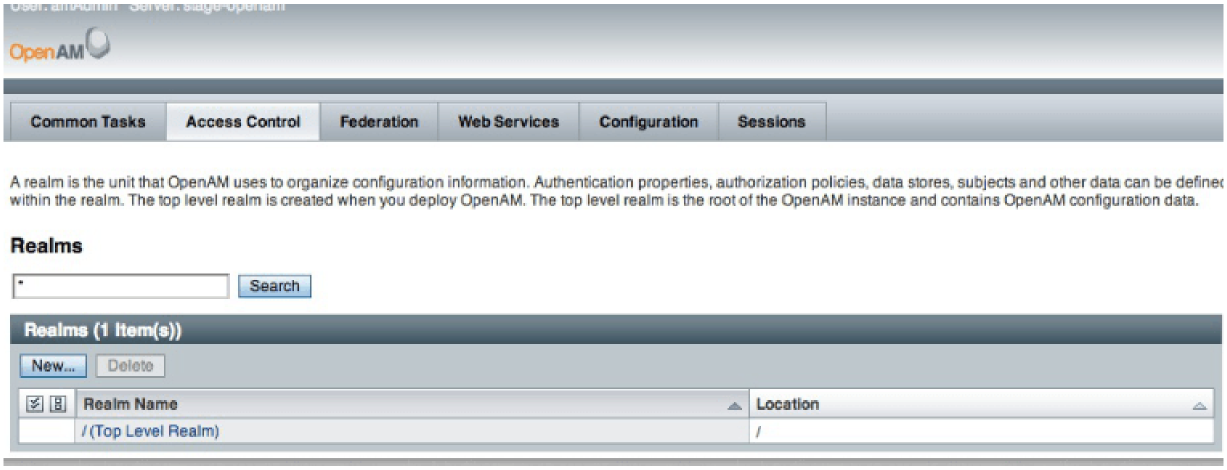
Java class name of authentication module*: com.cyphercor.logintc.openar

11. Restart your application server, i.e. tomcat

```
sudo service tomcat restart
```

Create New Module Instance

1. Navigate to your OpenAM installation, `http://www.myopenam.com:8080/openam`
2. Click Access Control
3. Select and click a Realm



A realm is the unit that OpenAM uses to organize configuration information. Authentication properties, authorization policies, data stores, subjects and other data can be definec within the realm. The top level realm is created when you deploy OpenAM. The top level realm is the root of the OpenAM instance and contains OpenAM configuration data.

Realms

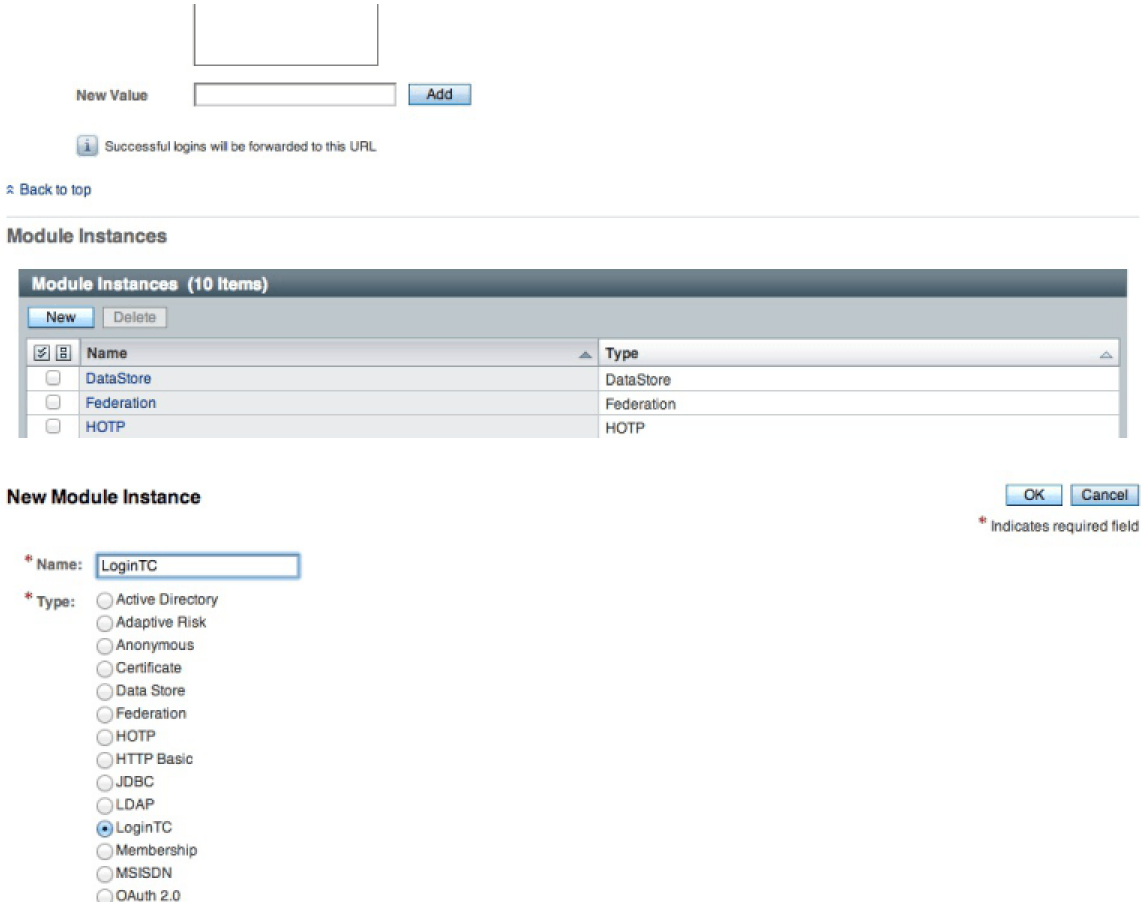
*

Realms (1 Item(s))

Realm Name	Location
/ (Top Level Realm)	/

4. Click Authentication tab

5. Create new Module Instance
    1. Click New under Module Instances
    2. Enter the name `LoginTC`
    3. Select `LoginTC` as Type
    4. Click OK



New Value [ ] Add

Successful logins will be forwarded to this URL

Back to top

Module Instances

Module Instances (10 Items)

New Delete

☐	Name	Type
☐	DataStore	DataStore
☐	Federation	Federation
☐	HOTP	HOTP

New Module Instance

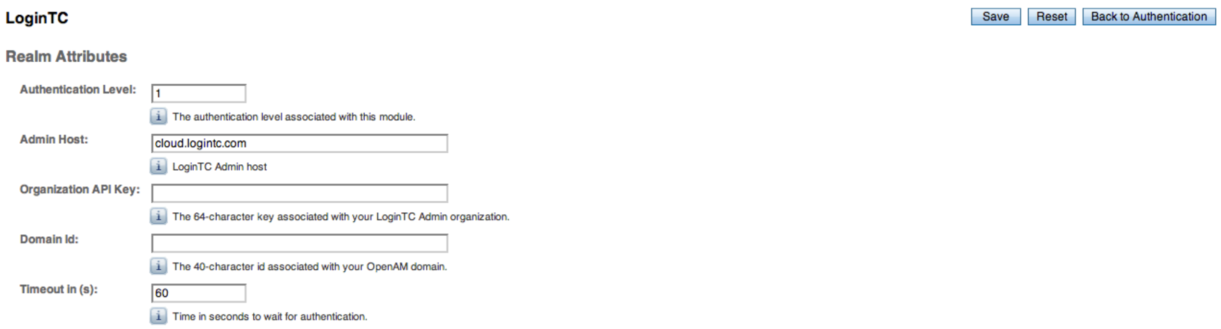
OK Cancel

* Indicates required field

* Name: LoginTC

* Type:
○ Active Directory
○ Adaptive Risk
○ Anonymous
○ Certificate
○ Data Store
○ Federation
○ HOTP
○ HTTP Basic
○ JDBC
○ LDAP
● LoginTC
○ Membership
○ MSISDN
○ OAuth 2.0

6. Click on `LoginTC` in the Modules list to configure the module



LoginTC

Realm Attributes

Authentication Level: 1
The authentication level associated with this module.

Admin Host: cloud.logintc.com
LoginTC Admin host

Organization API Key:
The 64-character key associated with your LoginTC Admin organization.

Domain Id:
The 40-character id associated with your OpenAM domain.

Timeout in (s): 60
Time in seconds to wait for authentication.

Property	Explanation
`Authentication Level`	The authentication level associated with this module
`Admin Host`	LoginTC Admin host
`Organization API Key`	The 64-character API key associated with your LoginTC Admin organization
`Domain ID`	The 40-character ID associated with your OpenAM domain
`Timeout`	Time in seconds to wait for authentication

7. When finished click Save
8. Restart tomcat on this server

```
sudo service tomcat restart
```

User Management

One-way user synchronization of users from your OpenAM datastore directory to your OpenAM domain in LoginTC Admin is done by running `logintc-sync`.

`logintc-sync` will connect via REST to your OpenAM installation and sync users according to settings in `users.cfg` and fetch users from your directory using the `filter` query. If you wish to keep your user directory in sync with the users in your OpenAM domain in LoginTC Cloud, you may periodically run this command (without the `--dry-run` flag).

Go to conf in the installation directory:

```
cd /opt/logintc/conf
```

Copy the sample file as a template for your configuration file:

```
cp sample-users.cfg users.cfg
```

Open the file to modify its contents:

```
vi users.cfg

            # openam
            openam.protocol=http
            openam.host=www.myopenam.com
            openam.port=8080
            openam.path=/openam
            openam.admin.username=amadmin
            openam.admin.password=password
            openam.realm=/
            openam.attr.username=uid
            openam.attr.name=cn
            openam.attr.email=mail
            openam.filter.objectclass=person

            # logintc
            logintc.apikey=
            logintc.domainid=
```

OpenAM configuration values:

Property	**Explanation**	**Examples**
`openam.protocol`	The protocol of your OpenAM server	http or https
`openam.host`	The host of your OpenAM server	www.myopenam.com
`openam.port`	The port of your OpenAM server	8080
`openam.path`	The path to your OpenAM installation	/openam
`openam.admin.username`	The username of a user with admin privileges in the realm	amadmin
`openam.admin.password`	The password of the above account	password
`openam.realm`	The realm in which the module is installed	/
`openam.attr.username`	The attribute containing the user’s username	uid
`openam.attr.name`	The attribute containing the user’s real name	cn

Property	Explanation	Examples
`openam.attr.email`	The attribute containing the user’s email	mail
`openam.filter.objectclass` (optional)	The object class of the users	person

LoginTC configuration values:

Property	Explanation
`logintc.apikey`	The 64-character key associated with your LoginTC Admin organization.
`logintc.domainid`	The 40-character id associated with your OpenAM domain.

The API key is found on the LoginTC Admin Settings page. The Domain ID is found on your domain settings page.

Example:

```
$ cd /opt/logintc/bin
            $ sudo ./logintc-sync /opt/logintc/conf/users.cfg
```

Output:

```
Processing /opt/logintc/users.cfg...
            Querying OpenAM Rest service [http://www.myopenam.com:8080]
            Found 2 users
            +-------+----------+----------+----------------------+
            | Realm | Username | Name     | Email                |
            +-------+----------+----------+----------------------+
            | /     | john.doe | John Doe | john.doe@example.com |
            | /     | jane.doe | Jane Doe | jane.doe@example.com |
            +-------+----------+----------+----------------------+
            Synchronizing 2 users...
            Done.
```

Check that your users were added to your domain by viewing them in LoginTC Admin Panel.

There are several other options for managing your users within LoginTC:

- Individual users can be added manually in LoginTC Admin Panel
- Bulk operations in LoginTC Admin Panel
- Programmatically manage user lifecycle with the REST API

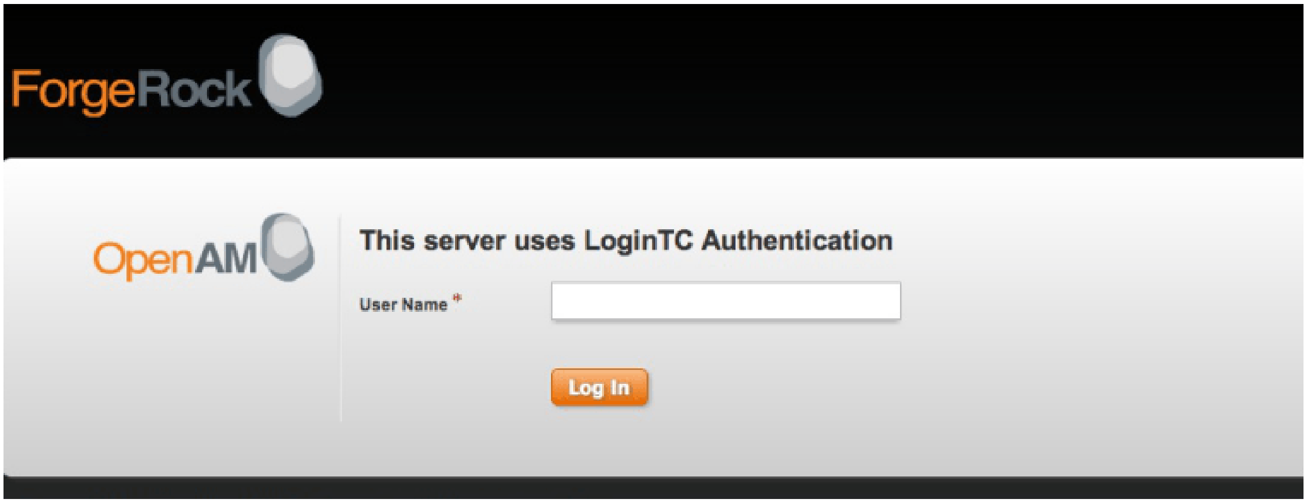
- One-way user synchronization of users to LoginTC Admin is performed using User Sync Tool.

Testing

Once you have synchronized some users you can test the LoginTC module.

Navigate to `www.myopenam.com:8080/openam/UI/Login?module=LoginTC`

Enter the test user’s username. You should now receive an authentication request on your mobile device. After authenticating you will be shown OpenAM user data.



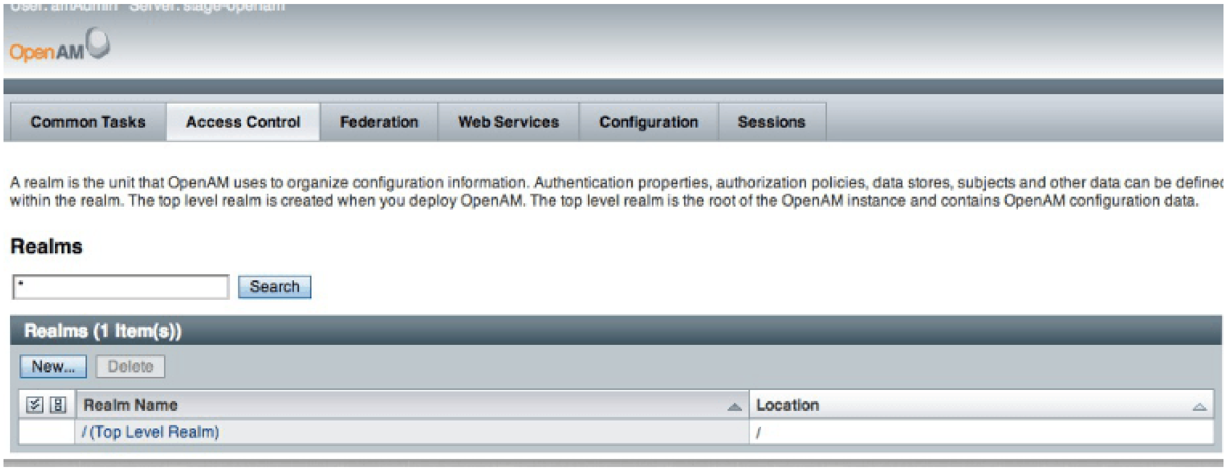
This server uses LoginTC Authentication

User Name *

Authentication Chaining

Now that everything is tested you will want to create or modify an existing authentication chain to include the LoginTC.

1. Navigate to your OpenAM installation, `http://www.myopenam.com:8080/openam`
2. Click Access Control
3. Select and click a Realm



A realm is the unit that OpenAM uses to organize configuration information. Authentication properties, authorization policies, data stores, subjects and other data can be defined within the realm. The top level realm is created when you deploy OpenAM. The top level realm is the root of the OpenAM instance and contains OpenAM configuration data.

Realms

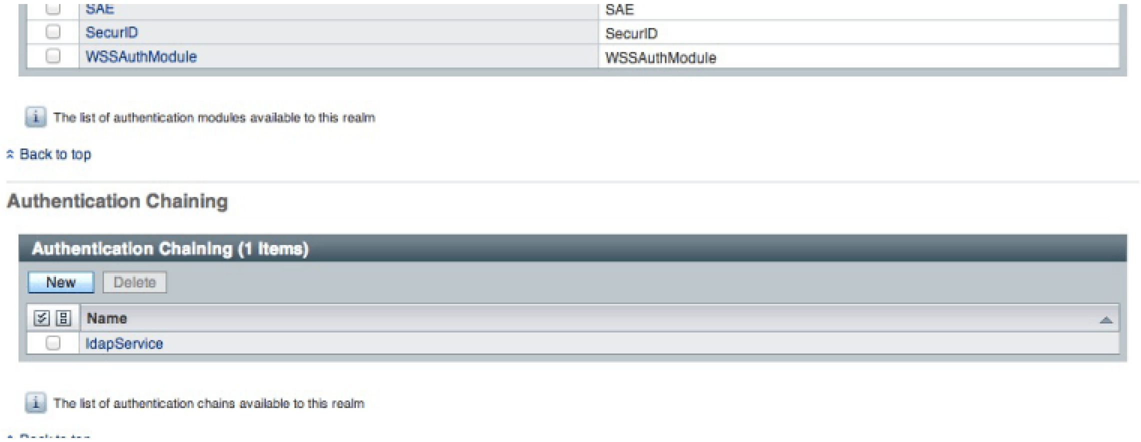
*

Realms (1 Item(s))

	Realm Name	Location
	/ (Top Level Realm)	/

4. Click Authentication tab

5. Create new Authentication Chaining
    1. Click New under Authentication Chaining
    2. Enter `LoginTC` and click OK



□	SAE	SAE
□	SecurID	SecurID
□	WSSAuthModule	WSSAuthModule

The list of authentication modules available to this realm

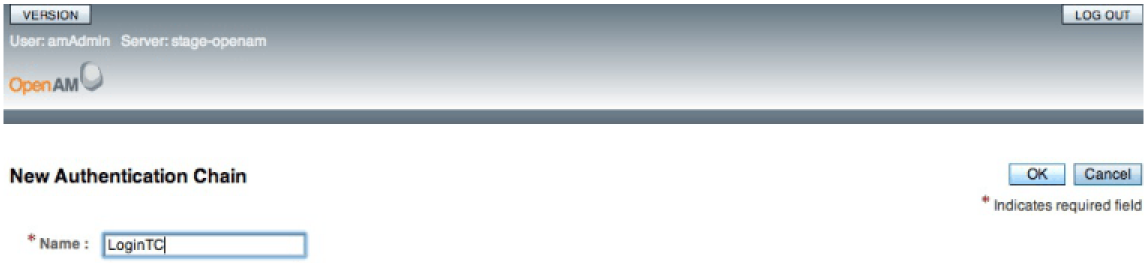
Back to top

Authentication Chaining

Authentication Chaining (1 Items)

	Name
□	ldapService

The list of authentication chains available to this realm



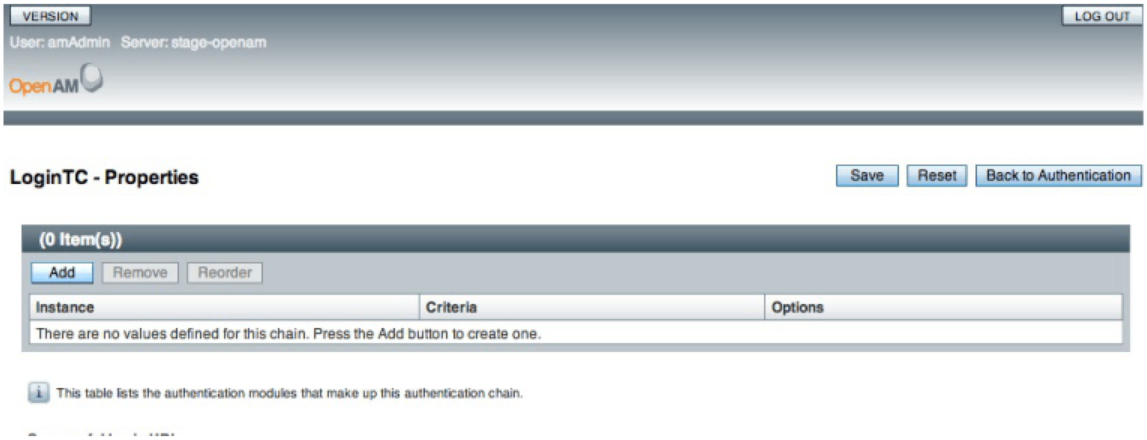
User: amAdmin Server: stage-openam

New Authentication Chain

* Indicates required field

* Name : LoginTC

6. Configure the new Authentication Chain
    1. Click Add
    2. Selet LoginTC
    3. Set Criteria to `REQUIRED`
    4. Click Save
    5. Click Back to Authentication

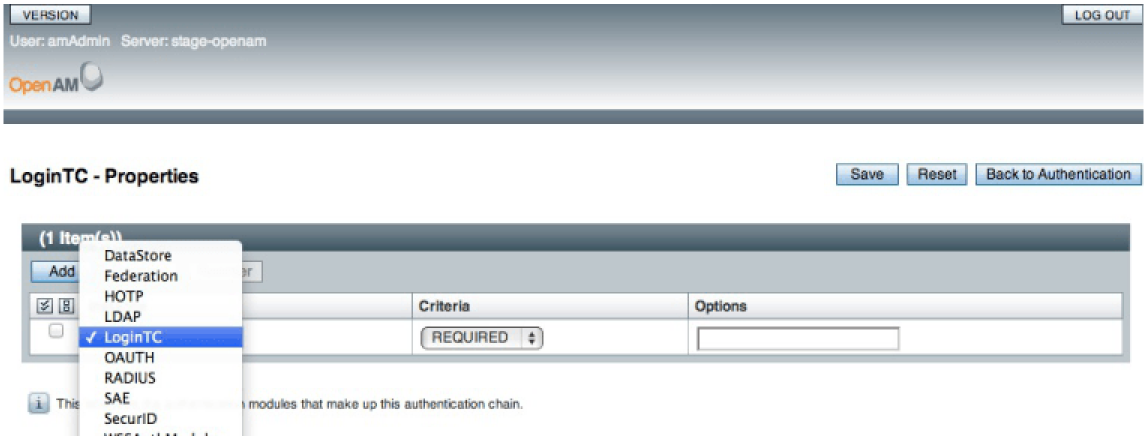


LoginTC - Properties

(0 Item(s))

Instance	Criteria	Options
There are no values defined for this chain. Press the Add button to create one.		

This table lists the authentication modules that make up this authentication chain.



LoginTC - Properties

(1 Item(s))

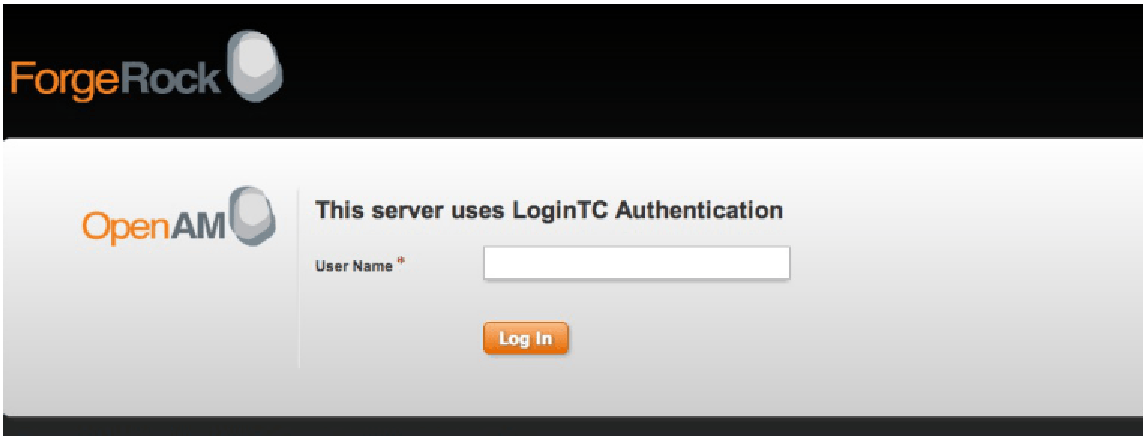
	Criteria	Options
☐ LoginTC	REQUIRED	

This table lists the authentication modules that make up this authentication chain.

7. Restart your application server, i.e. tomcat

```
sudo service tomcat restart
```

8. Test the new Authentication Chain
    1. Navigate to `www.myopenam.com:8080/openam/UI/Login?service=LoginTC`
    2. Enter the test user's username. You should now receive an authentication request on your mobile device. After authenticating you will be shown OpenAM user data.

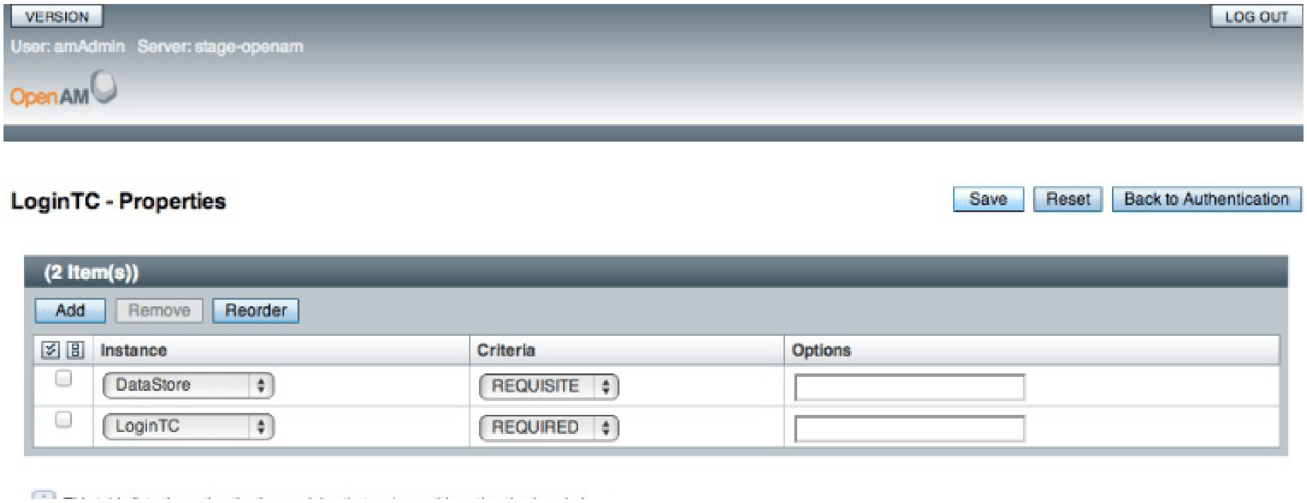


This server uses LoginTC Authentication

User Name *

Log In

You can add other authentication modules to make chains. For example the following will first prompt for a username and password and then request a 2nd factor using the LoginTC:



User: amAdmin Server: stage-openam

LoginTC - Properties

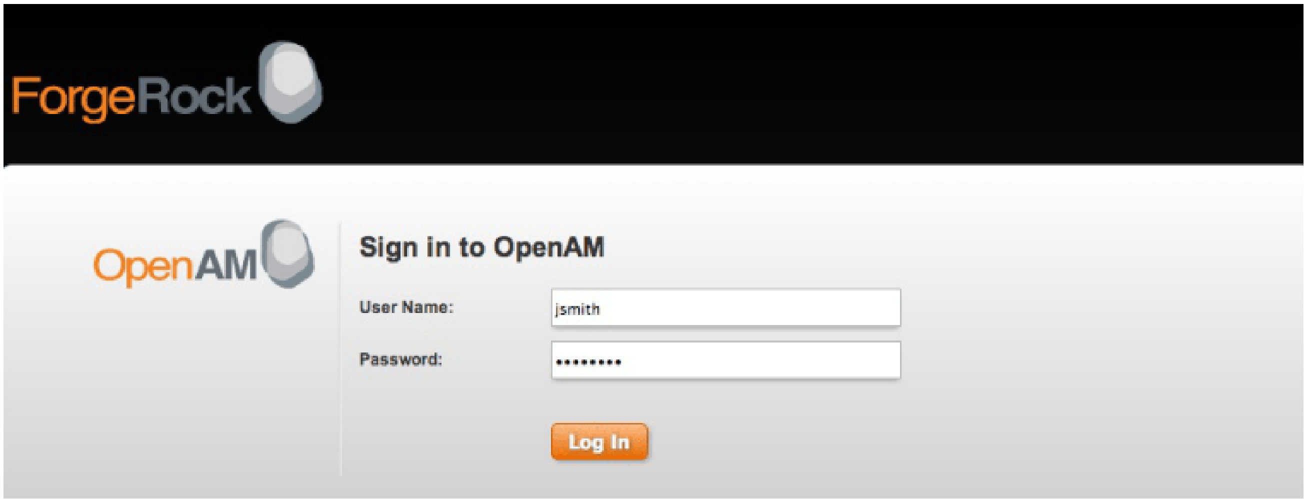
Save | Reset | Back to Authentication

(2 Item(s))

Add | Remove | Reorder

☑	Instance	Criteria	Options
□	DataStore	REQUISITE	
□	LoginTC	REQUIRED	

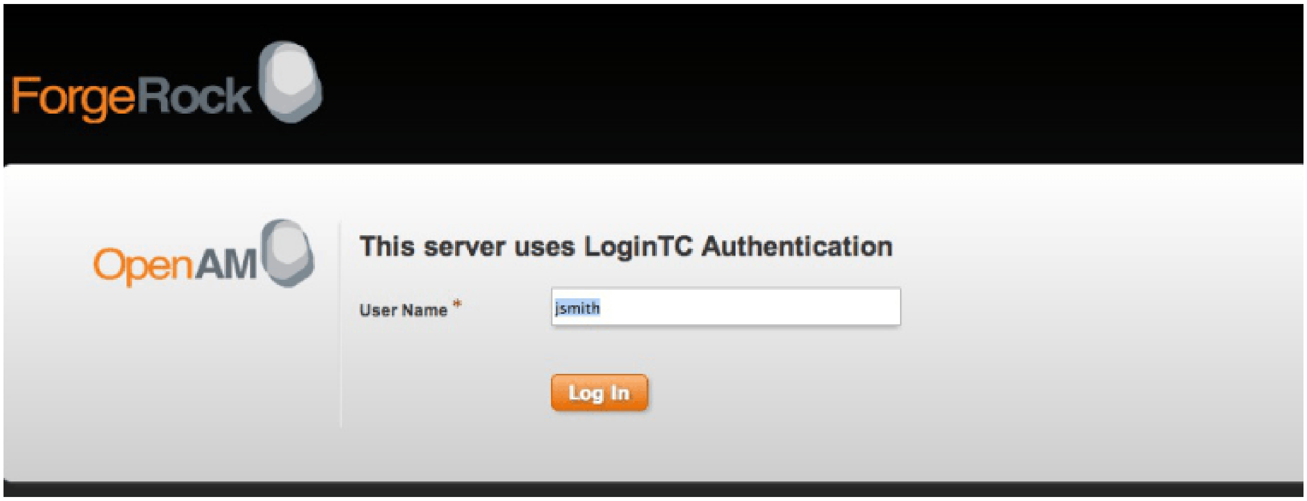
Sample flow:



Sign in to OpenAM

User Name: jsmith

Password: ••••••••



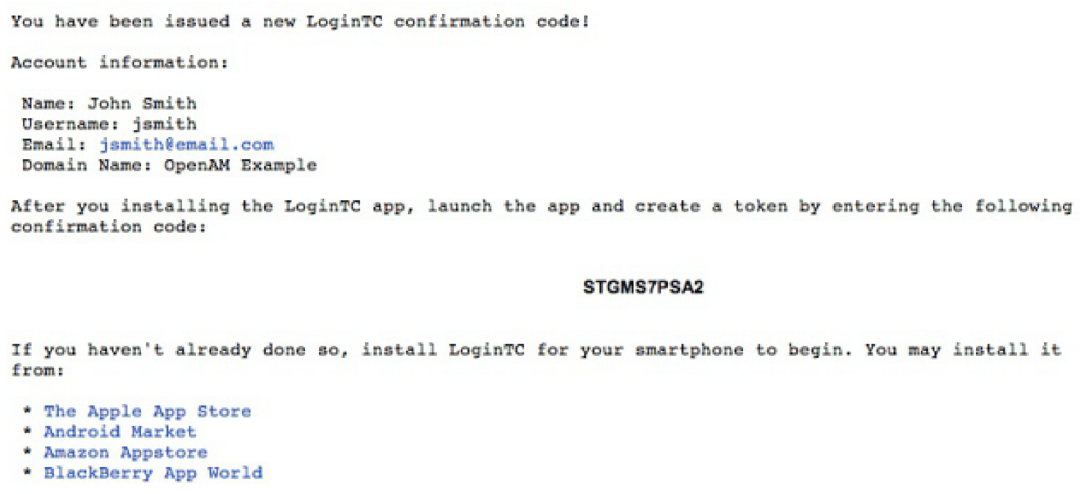
This server uses LoginTC Authentication

User Name * jsmith

User Enrolment

Once users have been synched to your OpenAM domain and you have tested your setup you can begin the process of user provisioning.

As seen in the Testing Installation section, users are issued a confirmation code which they use to load a new token on their mobile app. The Issue Token button will send an email to your user with full instructions on how to load a token. Sample email:



```
You have been issued a new LoginTC confirmation code!

Account information:

 Name: John Smith
 Username: jsmith
 Email: jsmith@email.com
 Domain Name: OpenAM Example

After you installing the LoginTC app, launch the app and create a token by entering the following
confirmation code:

                                        STGMS7PSA2

If you haven't already done so, install LoginTC for your smartphone to begin. You may install it
from:

 * The Apple App Store
 * Android Market
 * Amazon Appstore
 * BlackBerry App World
```