Lastline Enterprise SAML 2.0 Single Sign On Integration

January 17, 2024

6950 Hollister Avenue, Suite 100
Goleta, CA 93117
lastline.com | 805.456.7076

1 Introduction

This guide describes the integration of the SAML 2.0 Single Sign On protocol so that it can be used by the Lastline Enterprise.

The purpose of using Single Sign On is to allow a single source of authentication. This is particularly useful in larger networks where an Identity Provider is already used to log into multiple applications on the network. Configuring the Single Sign On system allows an administrator to permit users to log into the Manager / Pinbox / Analyst using an existing Identity Provider. It is also possible to configure more than one Identity Provider if the administrator desires. Users will then be presented with different Single Sign On options when they attempt to login.

In the following sections, this guide will describe how to configure and successfully use the SAML 2.0 Single Sign On protocol.

2 Overview of the SAML 2.0 Single Sign On Integration

2.1 Glossary

This section briefly defines a few terms that will be used in the documentation.

SAML 2.0 Single Sign On

XML based protocol that uses security tokens and assertions to allow information about users to be passed (securely) from one system to another. Information about the user is passed from an Identity Provider (producer) to a Service Provider (consumer).

Identity Provider

Responsible for providing identifiers for users that are interested in interacting with a system. In order for a system to obtain information from an Identity Provider, the user must first successfully authenticate with the Identity Provider. This information is then passed to the Service Provider, asserted, then performs the login action.

Service Provider

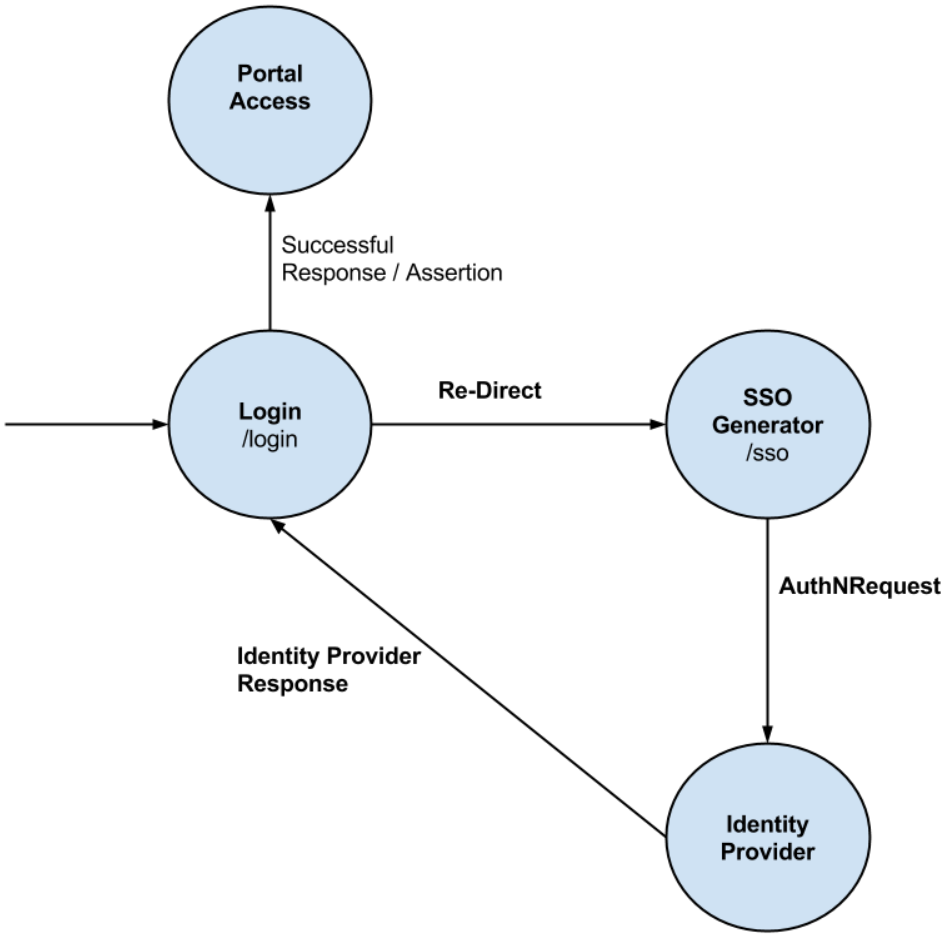
The Service Provider retrieves information about a user from the Identity Provider. This information is then securely asserted (via cryptographic signatures), to assure that user 'Alice' is really user 'Alice'. After successful assertion, the Service Provider is then responsible for performing the authentication process to the system.

2.2 Architecture

When using the SAML 2.0 Single Sign On, the Lastline Enterprise performs a cross-domain request to the Identity Provider. The user then must authenticate with the Identity Provider if he/she has not done so already. If the user is successfully logged into the Identity Provider, the Identity Provider will perform a cross-domain request back to the Service Provider with the information about the user. The Service Provider asserts that this information is correct by checking if the signature is valid of the request (using the public x509 cert provided by the Identity Provider).

If the assertion is successful, the Service Provider will then perform the authentication process to allow the user to be logged into the Manager / Pinbox / Analyst. If the user's account does not exist on the Manager / Pinbox / Analyst, that is, if "john_doe@mycompany.com" is returned from the Identity Provider and this user does not exist on the Manager / Pinbox / Analyst, then he/she will not be able to login. The account must be created on the Manager / Pinbox / Analyst and MUST be identical to the email returned from the Identity Provider (see: 3.1).

The following figure summarizes the Single Sign On process.



Summary of SAML 2.0 Single Sign On flow

3 Requirements

3.1 Requirements for the Manager / Pinbox / Analyst

In order to be able to login using SAML 2.0 Single Sign On, the Manager / Pinbox / Analyst must have an account that is identical to the account located on the Identity Provider. For example, if the email returned from the Identity Provider is "john_doe@mycompany.com", then the same account must be created on the Manager / Pinbox / Analyst. If this is not done, an authentication error will occur and the user will not be able to login.

3.2 Requirements for the Identity Provider

The Identity Provider must be configured to work with the SAML 2.0 protocol. It must also provide metadata XML that includes the following values (NOTE: the variables in the parenthesis are the variables that will be configured and the variables before the parenthesis are what they will be in the XML) :

- entityID (entity_id)
  - The unique identifier of the Identity Provider. This is normally the URL for the metadata, such as 'https://example.com/SAM
- X509Certificate (x509_cert)
  - The public X509 Certificate is used to validate the assertion data of the Identity Provider.
- NameIDFormat (name_id_format)
  - The NameIDFormat is used to indicate what SAML name identifier format the Single Sign On service supports.
- SingleSignOnService[Binding] (idp_binding)
  - Standard URI specified in the SAML 2.0 binding specification. This will indicate whether the request to the Identity Provider will be a POST or REDIRECT.
- SingleSignOnService[Location] (sso_service_url)
  - The location that will POST or GET an AuthNRequest for the Identity Provider.

4 Configuration

4.1 Adding SAML 2.0 Single Sign On using example script

4.1.1 How to use the script

The easiest way to configure SAML 2.0 Single Sign On is to use the example script provided here. The script is located in the examples directory as "add_saml_sso_from_metadata.py" and requires the following positional arguments [appliance_uuid] [url_or_file] [display_name] :

- appliance_uuid
  - Needed to determine which appliance to configure. This should be used on the Manager / Pinbox / Analyst that the Single Sign On will be implemented on. If the appliance_uuid is not known, it can be retrieved using the accounting API. Documentation for the method to retrieve all the appliance uuids can be located here.
- url_or_file
  - This can be a URL that contains the metadata xml or it can be the actual metadata xml file. The script uses this XML to parse the necessary information to configure SAML 2.0 Single Sign On.
- display_name
  - Used to identify the Identity Provider when logging in. For example, if the display name is "Foo Bar" the button to login using Single Sign On will be "Login with Foo Bar".

And the following are optional arguments :

- -n / –index
  - The configuration index for SAML 2.0 Single Sign On. Mainly used when multiple configurations are present. For example, if configuration two SAML 2.0 Single Sign On, the first configuration will have the option "-n 0" while the second will have the option "-n 1" and so on.
- –skip-verify-ssl
  - If a URL is specified for the metadata file and the SSL cert is not valid, this option skips checking the validity of said URL.
- -c / –config
  - The papi client configuration file. Where to find this configuration file and how to use it is described in this section: 4.2.2.

4.1.2 After using the script

Upon successful configuration and no errors reported, the following figure shows an example of what a successful configuration might look like.

```
root@manager:~# python add_saml_sso_from_metadata.py add [redacted] test.xml "Foo Bar" -c api_config.ini
Adding SSO configuration (index 0) for appliance [redacted]
Logging in...
Starting new HTTPS connection (1): [redacted].lltest.local
"POST /papi/login HTTP/1.1" 200 33
Request took 981 ms
Completed.
Doing POST request to https://[redacted].lltest.local/papi/appliance_mgmt/action/request.json
"POST /papi/appliance_mgmt/action/request.json HTTP/1.1" 200 75
Request took 1810 ms
```

Successful configuration with example script

4.2 Adding SAML 2.0 Single Sign On using papi client

4.2.1 Required Values for Settings

It is possible to configure SAML 2.0 Single Sign On without using the example script as shown above. To do this, the user directly configures the settings using the following values :

- display_name
  - What the Identity Provider will be displayed as in the login form.
    Example: "MyCompany SSO"
- entity_id
  - The unique identifier (URL) of the Identity Provider. Normally, this would contain the metadata of the Identity Provider.
    Example: "https://foobar.idp.example.com/SAML2"
- sso_service_url
  - The URL in which the Service Provider will POST or REDIRECT to.
    Example: "https://foobar.idp.example.com/SAML2/SSO/POST"
- idp_binding
  - The SAML 2.0 binding that the Identity provider will be using.
    Example: "urn:oasis:names:tc:SAML:2.0:bindings:HTTP-POST"
- name_id_format
  - The supported name id format.
    Example: "urn:oasis:names:tc:SAML:2.0:nameid-format:transient"
- x509_cert
  - The public x509 certificate that the Service Provider will use. This x509 cert MUST BE BASE64 ENCODED, exactly as you would see in an Identity Provider metadata file.

These settings must be JSON dictionary under the key "sso_saml2_config{0-3}". Along with the settings, another key "sso_saml2_enabled{0-9}" must be set to true in order for the configuration to work.

NOTE: The "{0-3}" represents that any value between 0 to 3 may be used. This can be used to configure multiple Single Sign On options. If "sso_saml2_enabled3" is set to true, then the corresponding setting must contain the same index, that is, "sso_saml2_config3".

4.2.2 Adding SAML 2.0 Single Sign On Example

To configure a SAML 2.0 Single Sign On, please use the papi-client. To use the papi-client, please insert your credentials to your Manager / Pinbox / Analyst in the "papi_client.ini.template" and rename it to "papi_client.ini". This ini file will be used to authenticate with the Manager / Pinbox / Analyst using the papi-client.

After successfully loading the papi-client, use the following command to configure SAML 2.0 Single Sign On :

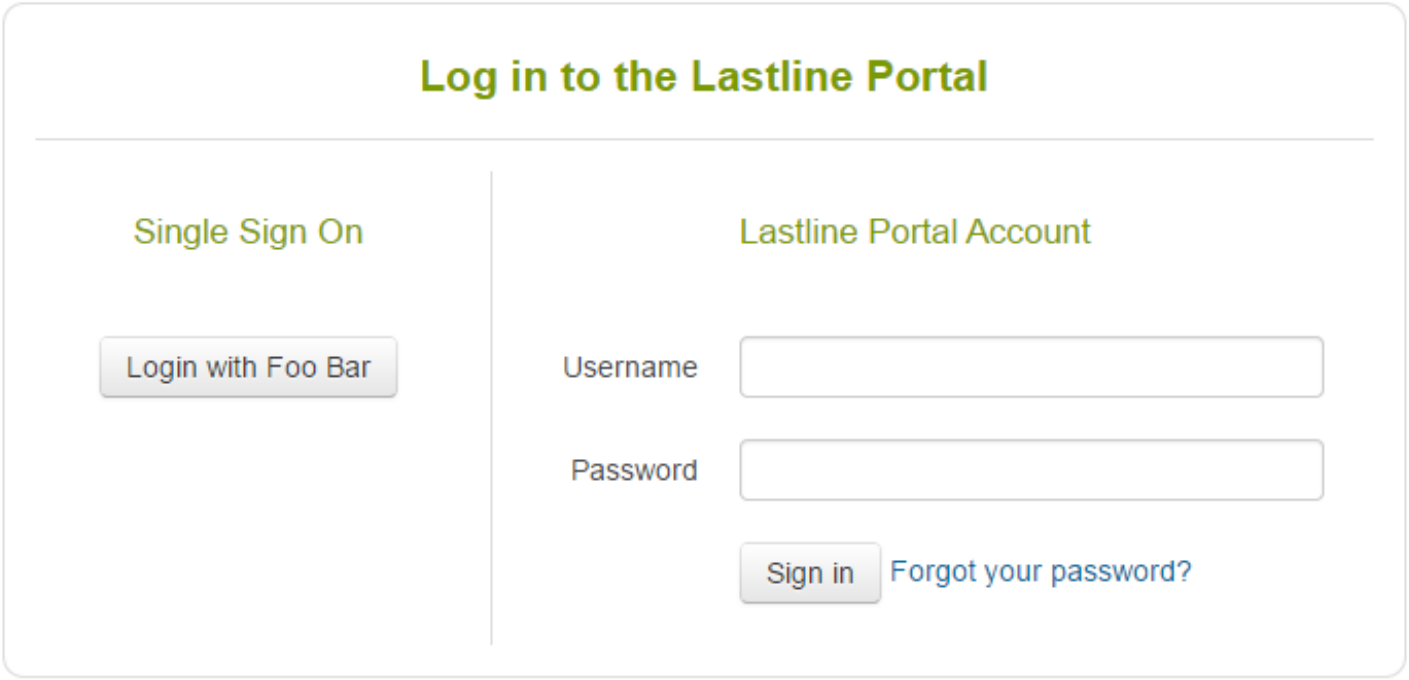
```
appliance_mgmt.action_request(
  appliance_uuid = appliance_uuid,
  action_type = "CONFIGURE",
  action_parameters = {
  "sso_saml2_enabled0": true,
  "sso_saml2_config0": json.dumps({
    "display_name": "MyCompany SSO",
    "entity_id": "https://foobar.idp.example.com/SAML2",
    "sso_service_url": "https://foobar.idp.example.com/SAML2/SSO/POST",
    "idp_binding": "urn:oasis:names:tc:SAML:2.0:bindings:HTTP-POST",
    "name_id_format": "urn:oasis:names:tc:SAML:2.0:nameid-format:transient",
    "x509_cert": "BASE64ENCODEDCERTIFCATEHERE"
  })
})
```

Documentation for this command can be found here.

After performing this command, the Manager / Pinbox / Analyst will configure itself and the settings will be applied. If successfully configured, the login page will now change and contain the Single Sign On option(s) that were specified upon configuration (see: 4.3

4.3 Successful Configuration

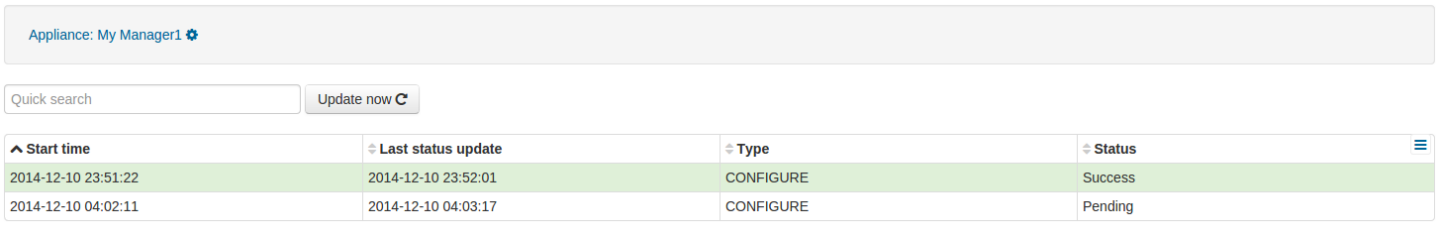
After one of the configuration methods is used (example script or papi-client), the Manager / Pinbox / Analyst will perform a new configuration task on itself that may take a few minutes. After this is completed, a new login page will be displayed and will look like the figure below:



Login page after successfully configuring Single Sign On

If this login page is not displayed, then there was either an error in the configuration or the configuration process is still pending. To view this information, please visit the appliance action logs page, located in the "Appliances->Logs->Action Logs" section. Additional documentation on this page can be found here.

A successful configuration log will look like the first row in following table :



Successful configuration log after configuring Single Sign On

4.3.1 Registering the Service Provider to the Identity Provider

Depending on the Identity Provider, it is sometimes necessary to register the Service Provider to the Identity Provider. This can either be done through manual configuration on the Identity Provider or using the metadata that is generated by the Service Provider. This metadata can be retrieved by accessing the Manager / Pinbox / Analyst user website and going go the "/sso?saml_metadata&cfg={0,3}" route where the cfg GET parameter is the configuration index. By default it will be the first index (0) and is not necessary.

The cfg GET parameter will be necessary to change if there are multiple identity providers configured.