Power of SAML in SAP NetWeaver

Summary

This document provides an introduction to the **Security Assertion Markup Language** (**SAML**) and its use in SAP NetWeaver.

Author(s): Krishna Sriharikota

Company: HCL Technologies Ltd.

Created on: 17 August 2006

Author Bio

Krishna Srihari is a SAP NetWeaver technical consultant with HCL Technologies, and is proficient in all phases of web application development using JAVA, J2EE , web services, and portal technologies. He also has a sound knowledge of SAP Enterprise Portal, Web Dynpro development, and SAP XI.

Table of Contents

Overview

The **Security Assertion Markup Language**, or SAML, was developed by the Security Services Technical Committee of OASIS as an XML-based framework for communicating user authentication, entitlement, and attribute information. As its name suggests, SAML allows business entities to make assertions regarding the identity, attributes, and entitlements of a subject (an entity that is often a human user) to other entities, such as a partner company or another enterprise application.

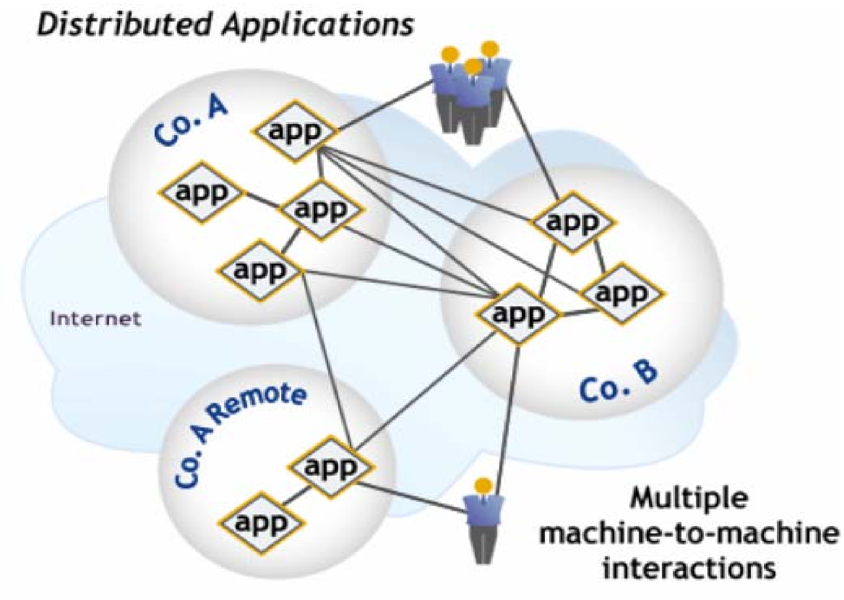
SAP NetWeaver is committed to use SAML 1.0 for Single Sign On, and SAML 1.1 as cornerstone for achieving the security needed for SAP's Enterprise Service Architecture.

Purpose

Distributed computing faces the following security challenges:

1. Security in distributed applications, which span different companies, is crucial
2. Current transport layer security protocols like SSL, TLS, IP Sec are not sufficient
3. Application layer security is a must
4. Security for Man-to-Machine and for Machine-to-Machine interaction is required
5. A standard for distributed authentication and/or authorization is required

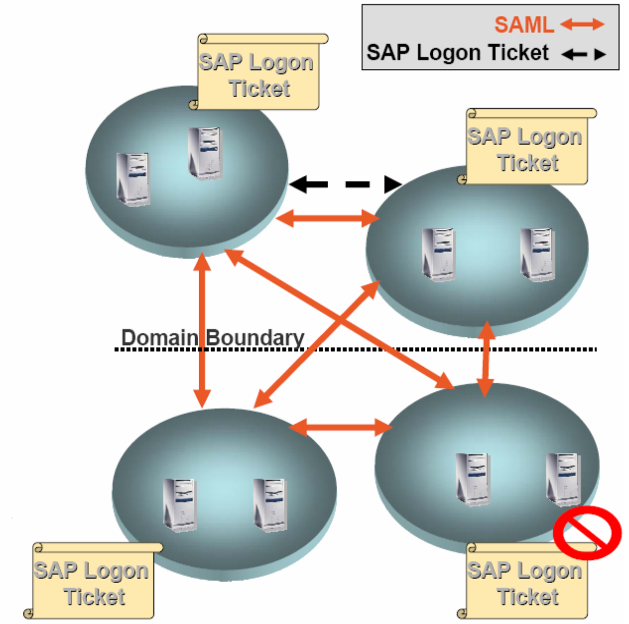
The figure below shows the interaction between Man-to-Machine, and Machine-to-Machine in distributed environments.



Useful of SAML

SAML provides the following benefits:

1. Interoperable security solutions that simplify systems integration and minimize resource requirements
2. Remote access to protected resources by exchange of
   - Authentication Information
   - Authorization information
3. Standards-based mechanisms to exchange security information using SOAP, HTTP(s)



SAML is a protocol for encoding security related information (assertions) into XML and exchanging this information in a request/response fashion. It does not authenticate users. It relies for message exchange on Standard Security Protocols like SSL, TLS, and uses XML signatures.

SAML authorities produce so called assertions on client requests. An assertion can be either an authentication or an authorization assertion

- Authentication assertion - a piece of data that represents an act of authentication performed on a subject (user) by the authority
- Authorization assertion: a piece of data that represents authorization permissions for a subject (user) on a resource

SAML is an emerging OASIS standard.

Configuring SAML for SSO

1. Configuration Steps

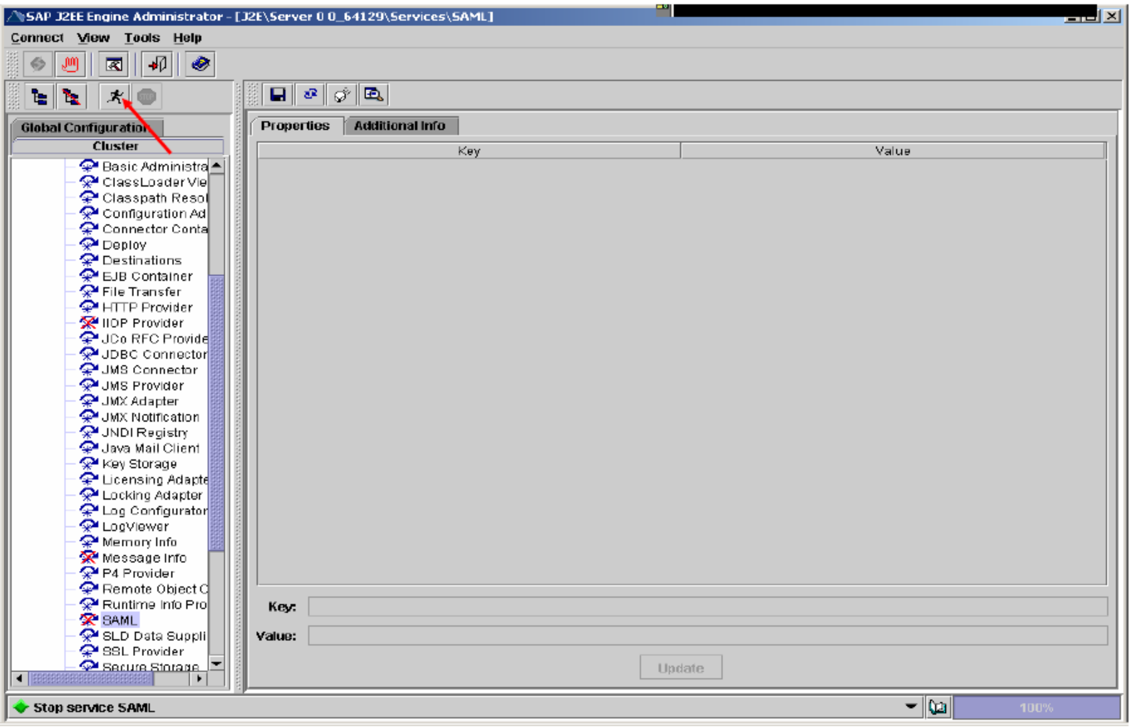
Prerequisites:

- It is strongly recommended that you have configured SSL on the SAP J2EE Engine. SAML relies on secure transport mechanism

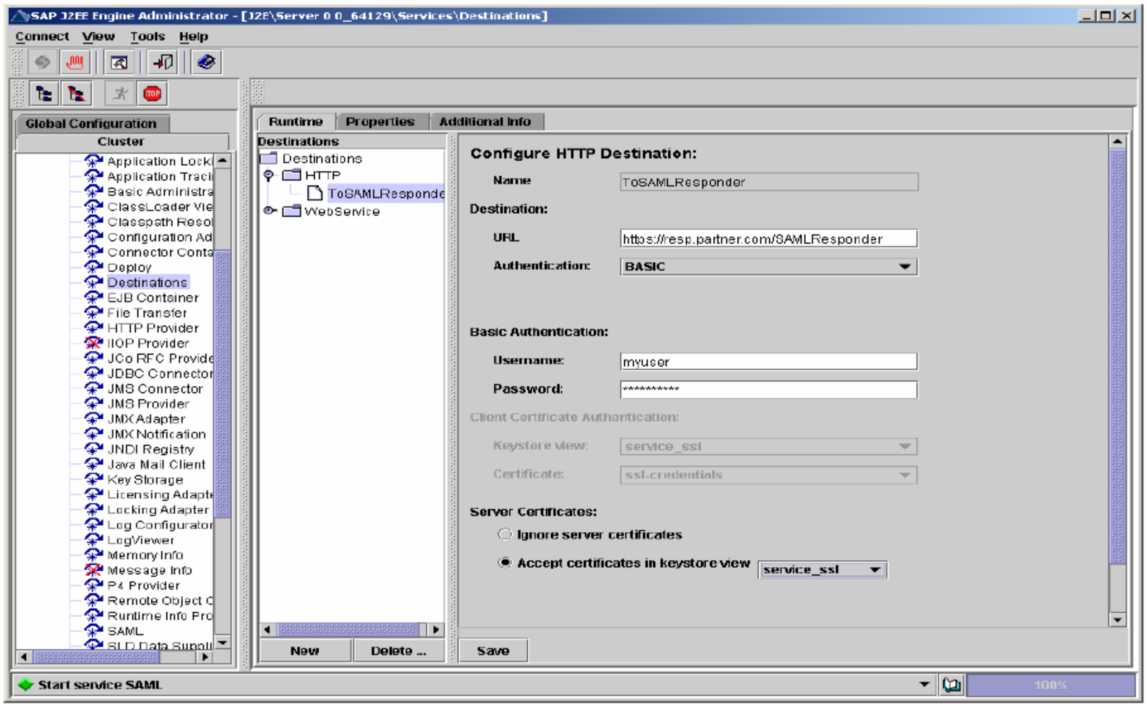
Steps for SAML configuration for the client SSO scenario in SAP J2EE Engine:

- Start the SAML Service
- Define a destination to the SAML responder
- Define SAML Parameters
- Adjust the JAAS login module stack for the services, which should be accessed by SAML

2. Start the SAML Service



3. Create a Destination to the SAML Responder



Configure HTTP Destination:

Name: ToSAMLResponder

Destination:

URL: https://resp.partner.com/SAMLResponder

Authentication: BASIC

Basic Authentication:

Username: myuser

Password: **********

Client Certificate Authentication:

Keystore view: service_ssl

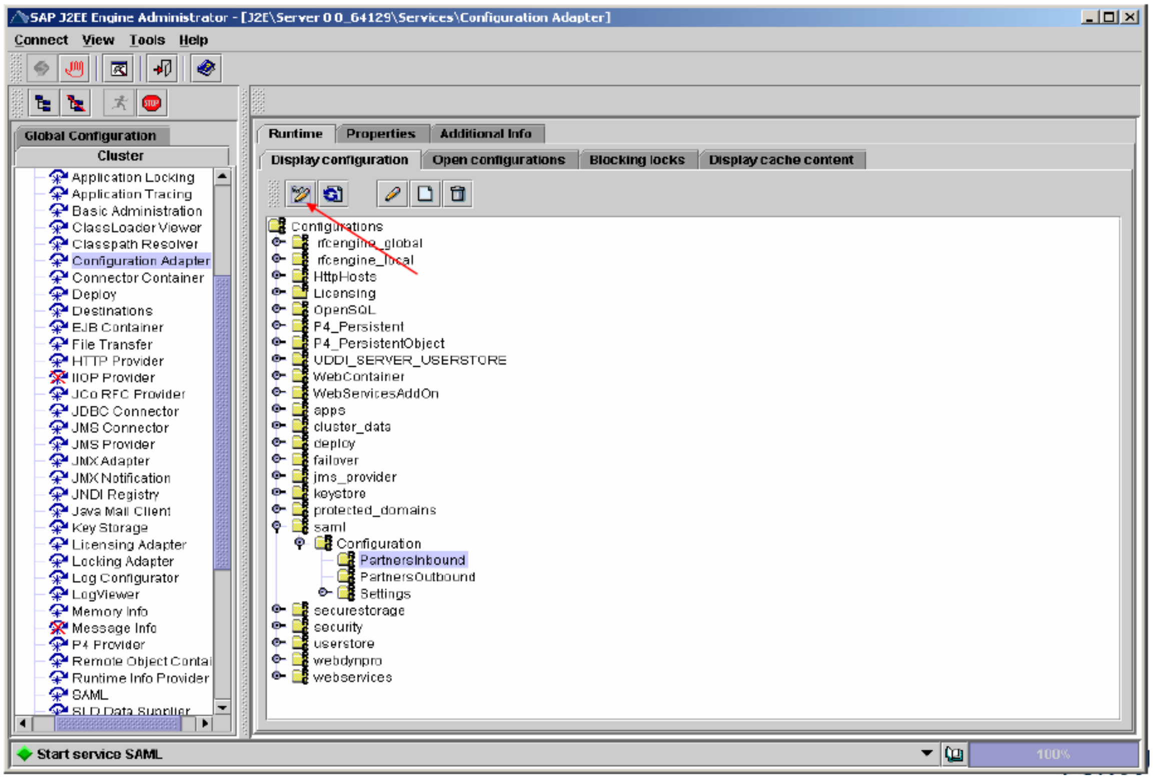
Certificate: ssl-credentials

Server Certificates:

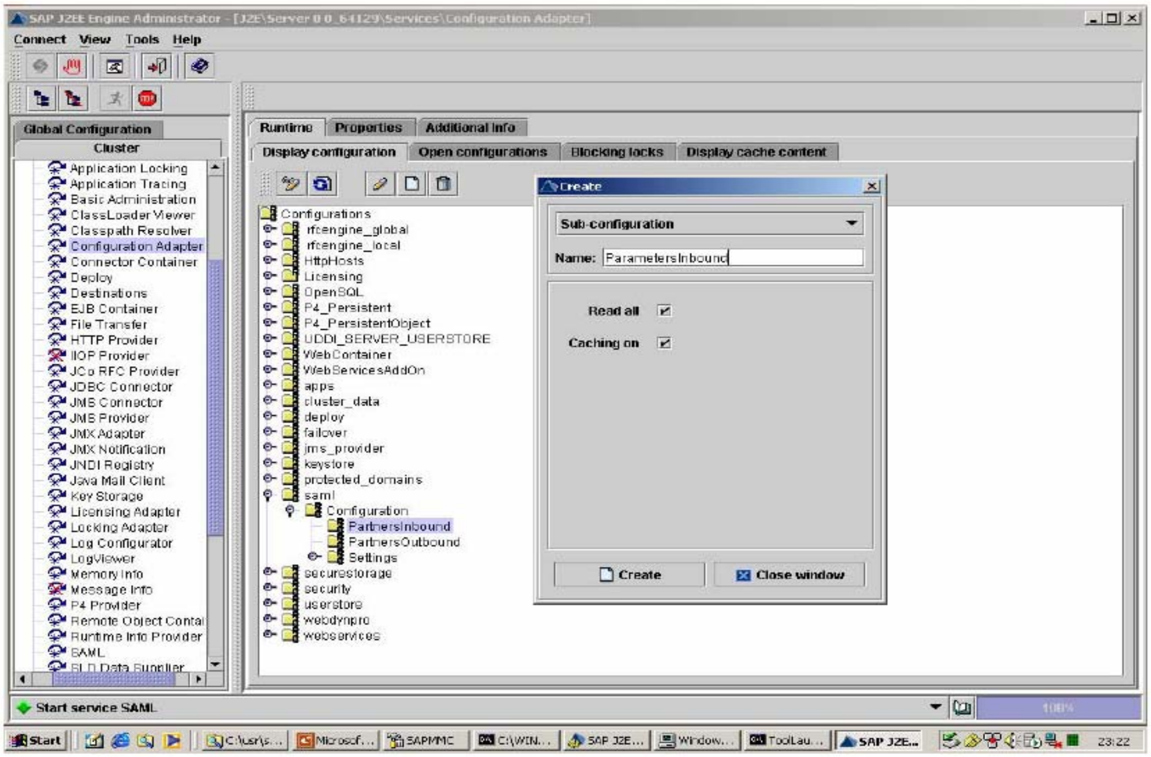
○ Ignore server certificates

● Accept certificates in keystore view service_ssl

4. Enter SAML Parameters –View in the configuration Tree



5. Enter SAML Parameters –Create a Sub-Configuration



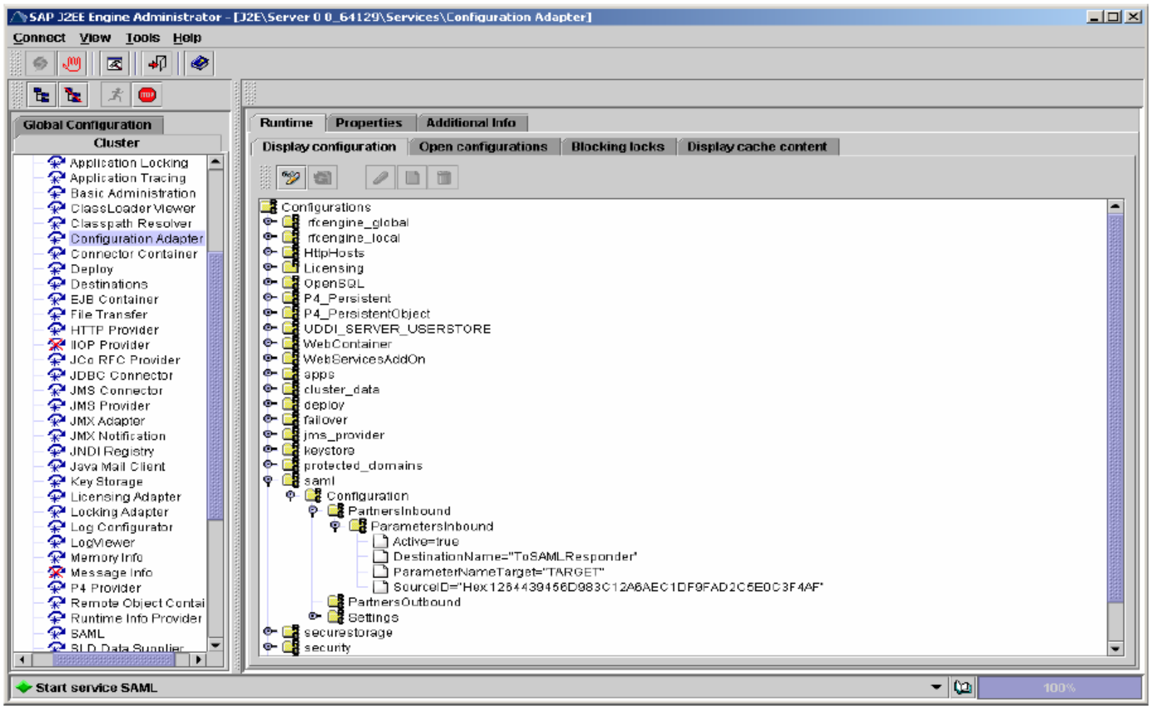
Sub-configuration

Name: ParametersInbound

Read all ☑

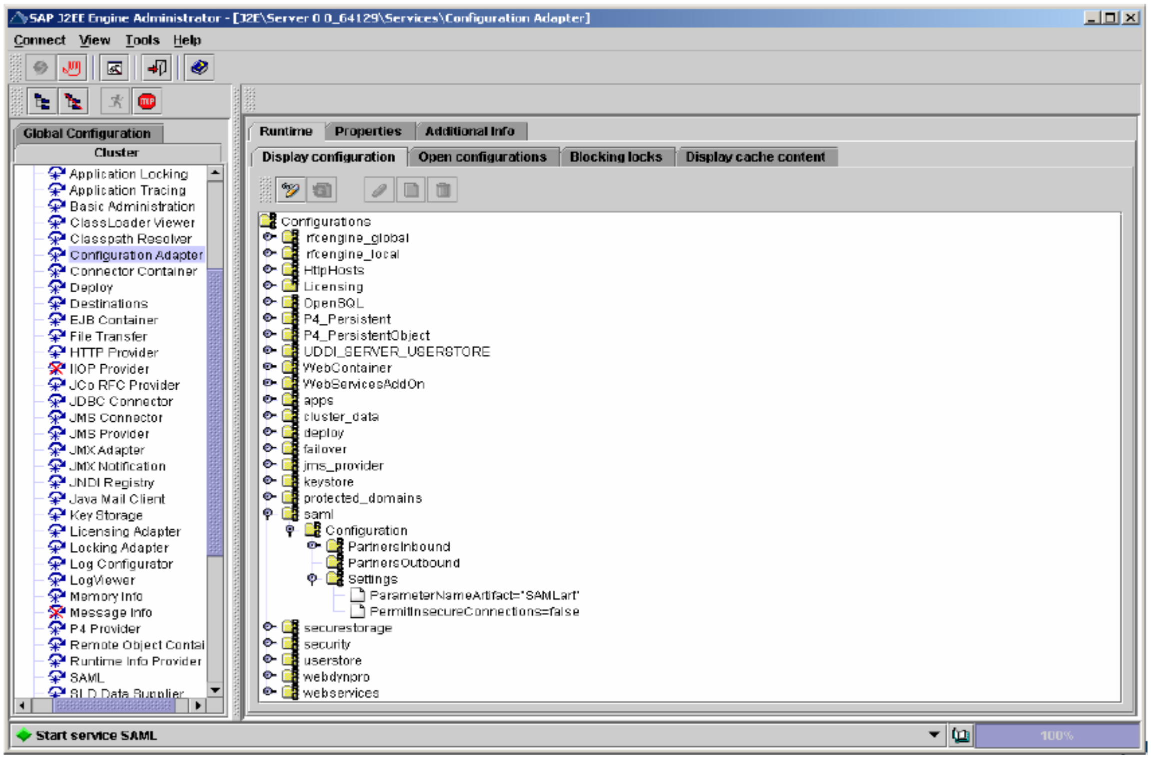
Caching on ☑

6. Enter SAML Parameters –Inbound Parameters



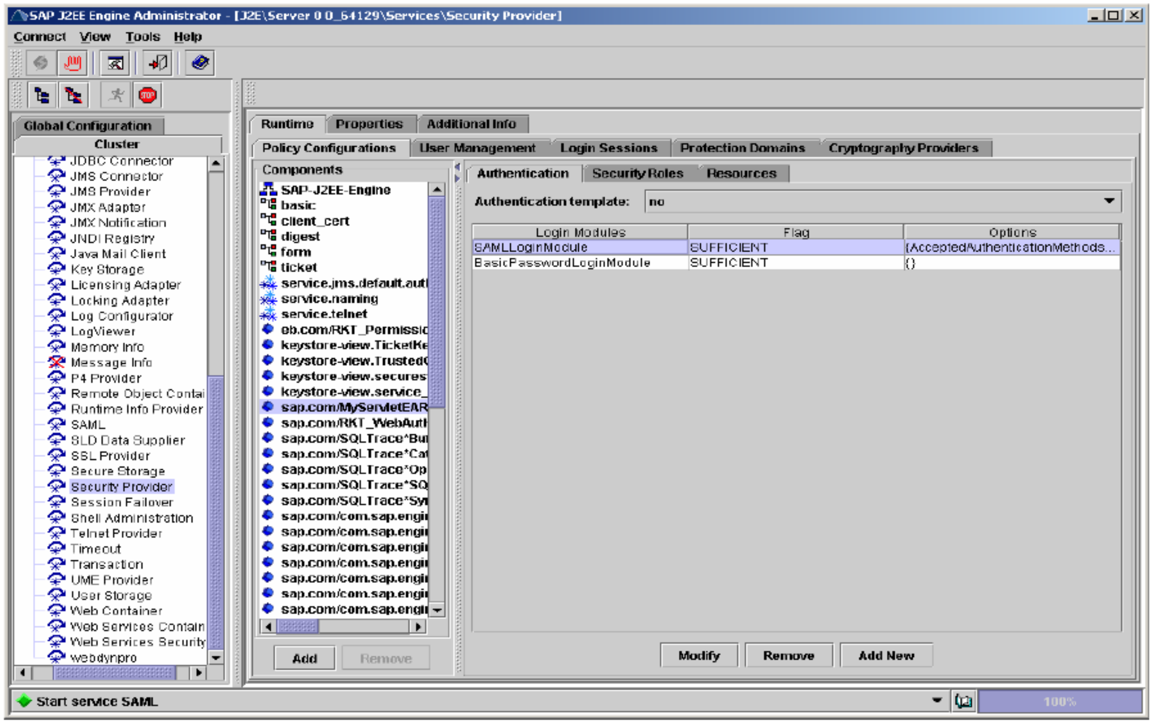
```
ParametersInbound
  Active=true
  DestinationName="ToSAMLResponder"
  ParameterNameTarget="TARGET"
  SourceID="Hex1264439456D983C12A6AEC1DF9FAD2C5E0C3F4AF"
```

7. Enter SAML Parameters –General Parameters



```
Configurations
  rfcengine_global
  rfcengine_local
  HttpHosts
  Licensing
  OpenSQL
  P4_Persistent
  P4_PersistentObject
  UDDI_SERVER_USERSTORE
  WebContainer
  WebServicesAddOn
  apps
  cluster_data
  deploy
  failover
  jms_provider
  keystore
  protected_domains
  saml
    Configuration
      PartnersInbound
      PartnersOutbound
      Settings
        ParameterNameArtifact="SAMLart"
        PermitInsecureConnections=false
  securestorage
  security
  userstore
  webdynpro
  webservices
```

8. Adjust Login Module Stack



Authentication template: no

Login Modules	Flag	Options
SAMLLoginModule	SUFFICIENT	{AcceptedAuthenticationMethods...
BasicPasswordLoginModule	SUFFICIENT	{}

Disclaimer and Liability Notice

This document may discuss sample coding or other information that does not include SAP official interfaces and therefore is not supported by SAP. Changes made based on this information are not supported and can be overwritten during an upgrade.

SAP will not be held liable for any damages caused by using or misusing the information, code or methods suggested in this document, and anyone using these methods does so at his/her own risk.

SAP offers no guarantees and assumes no responsibility or liability of any type with respect to the content of this technical article or code sample, including any liability resulting from incompatibility between the content within this document and the materials and services offered by SAP. You agree that you will not hold, or seek to hold, SAP responsible or liable with respect to the content of this document.