

# **Performance Benchmark for SAML2 Protocols Using Federation Manager and Access Manager**

June 2007

This article reports performance numbers yielded from tests of SAML2 Federation protocols. The tests were conducted using Sun Java System Federation Manager 7.0 and Sun Java System Access Manager 7.0. Use the results reported here as a reference point for determining how to optimize performance in your SAML2 Federation environment.

# Contents

- “About the Test Environment” on page 2
- “About the Test Cases” on page 4
- “How to Read the Test Results” on page 6
- “Test Results for SAML2 Protocols Initiated at the Service Provider Site” on page 7
- “Test Results for SAML2 Protocols Initiated at the Identity Provider Site” on page 9
- “Using the Performance Test Results for Guidance” on page 11

## About the Test Environment

The main objective in conducting these tests is to provide benchmark information based on actual SAML2 deployments. You can use this document as a reference to estimate the number of Service Provider subsystems and the number of Identity Provider subsystems to use in your SAML2 environment.

In this test environment, a Service Provider and an Identity Provider form a circle of trust in order to exchange user authentication information using SAMLv2. The circle of trust contains one Identity Provider, a service that maintains and manages identity information. Once the circle of trust is established, single sign-on is enabled between both providers.

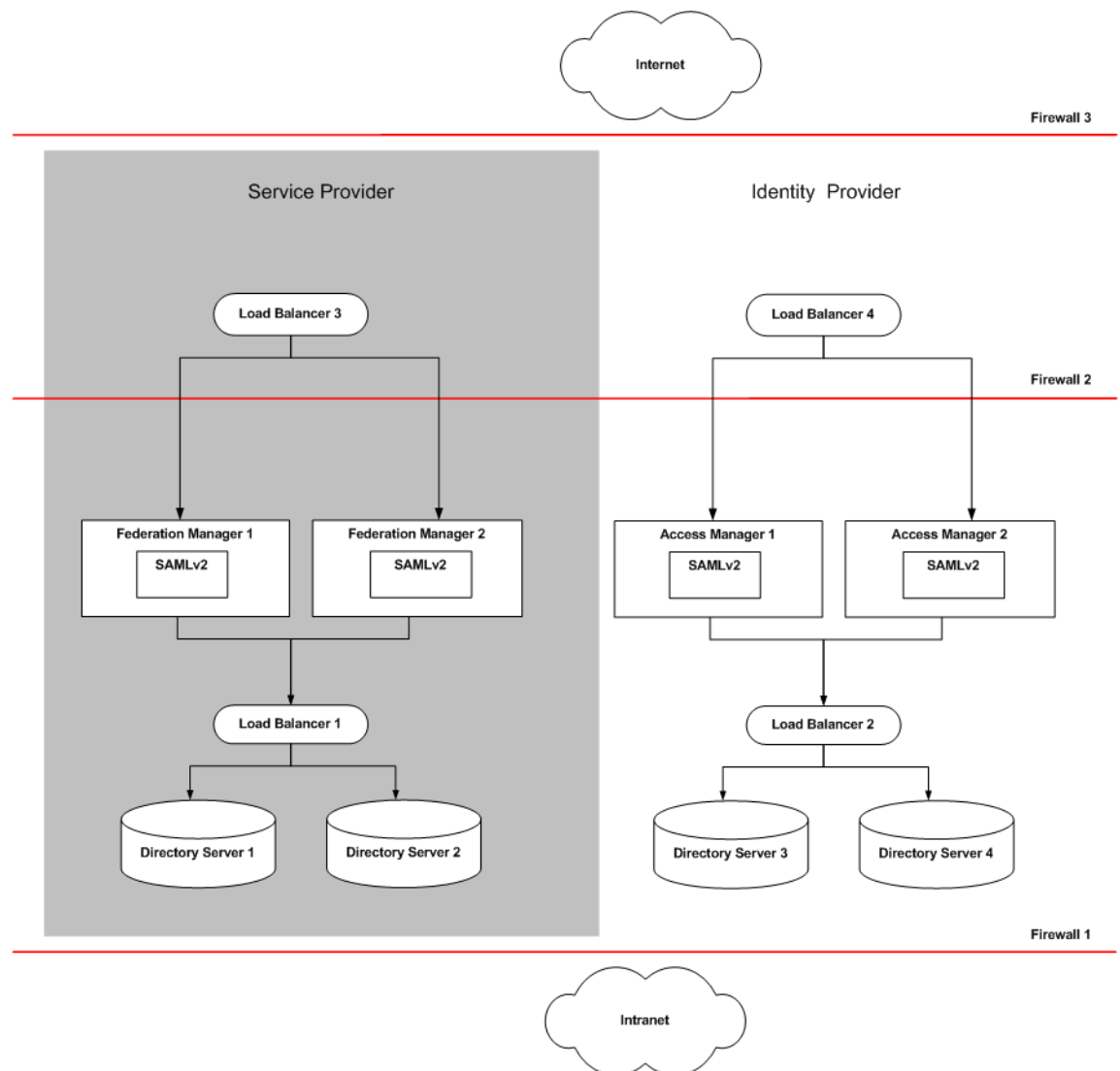


FIGURE 1 System Architecture

Secure Sockets Layer (SSL) is not used anywhere in the environment, but XML signing is enabled to provide a measure of security. Both the Service Provider and the Identity Provider use Sun Java System Directory Server as the backend repository.

The Service Provider domain is `siroe.com`. Two Federation Managers are load-balanced for high availability, and each is configured for the SAMLv2 protocol. Immediately following Federation Manager installation, user data was migrated from the default flat files to the Directory Server configuration. Each Federation Manager server uses one Directory Server instance for the user data repository, and a second instance for Federation Manager configuration.

The Identity Provider domain is `example.com`. Two Access Manager servers are configured for the SAMLv2 protocol and load-balanced for high availability, and each is configured for the SAMLv2 protocol. At the Identity Provider site, one Directory Server instance is created for the user data repository, and a second instance is created for Access Manager configuration. A subrealm is created in the Directory Server user data instance.

## Software and Hardware Used in the Testing Environment

The following tables summarize the hardware and software used in the SAML2 Federation testing environment.

TABLE 1 Sun Java System Software Versions

Software	Version
Federation Manager	7.0
Access Manager	7.0 SP5
SAML2 Plug-in	Patch 3
Sun Web Server	6.1 SP7
Directory Server	5.0 Patch 2

TABLE 2 Hardware Specifications

Server Host	CPU	Memory	OS/Platform	Hardware
FederationManager1.siroe.com	4 CPU, 6 cores	16 GB	Solaris 10 U2, 1200MHz	Niagara T2000
FederationManager2.siroe.com	4 CPU, 6 cores	16 GB	Solaris 10 U2, 1200MHz	Niagara T2000
DirectoryServer-1.siroe.com	4 CPU	1 GB	Solaris 10 U2, Ultra-4	Sun Fire 480
DirectoryServer-2.siroe.com	4 CPU	1 GB	Solaris 10 U2, Ultra-4	Sun Fire 480
AccessManager-1.example.com	4 CPU, 8 cores	16 GB	Solaris 10 U2, 1200Mhz	Niagara T2000
AccessManager-2.example.com	4 CPU, 8 cores	16 GB	Solaris 10 U2, 1200Mhz	Niagara T2000
DirectoryServer-3.example.com	2 CPU	4 GB	Solaris 10 U2, X86	X86
DirectoryServer-4.example.com	2 CPU	4 GB	Solaris 10 U2, X86	X86
LoadBalancer.siroe.com				F5 BigIP
LoadBalancer.example.com				F5 BigIP

## Basic Performance Tuning

The instructions in the *Sun Java System Access Manager 7 2005Q4 Performance Tuning Guide* were followed to run the following Access Manager tuning scripts on all servers in the test environment:

Access Manager servers	amtune-os	Tunes the Solaris OS kernel and TCP/IP parameters.
	amtune-identity	Tunes the installed instance of Access Manager.
Federation Manager servers	amtune-os	Tunes the Solaris OS kernel and TCP/IP parameters.
	amtune-identity	Tunes the installed instance of Federation Manager.
Directory Servers	amtune-prepareDSTuner	Generates the amtune-directory script.
	amtune-directory	Tunes the Directory Server that supports Access Manager.
Web Servers	amtune-ws61	Tunes the Sun Java System Web Server 2005Q4 (6.1) Web container.

Federation Manager does not come with its own tuning scripts. So the Access Manager tuning scripts were first copied to the Federation Manager environment, and then used to tune the Federation Manager servers.

For servers running on Niagara hardware systems, the tuning parameters were modified to remedy a known memory issue. For detailed information, see the notes entitled “JVM Tuning Recommendations for Niagara Hardware” on page 13 of this document.

## User Population

The Federation Manager and Access Manager databases are provisioned with 100,000 users and bulk-federated using the saml2bulkfed script that comes with Access Manager and with Federation Manager. The user database is indexed with SAML2 federation attributes for Directory Server search optimization.

## Load Generation

Load Runner software and recorded scripts are used to generate the test load. The users are chosen randomly from user0 to user99999 in the script. One hundred concurrent virtual users were used to achieve maximum throughput and maximum CPU usage in this test environment. In your environment, depending upon your system configuration, this number may be different.

## About the Test Cases

The performance test cases were designed to measure throughput (measured in transactions per second), CPU usage, and system response times. To ensure accurate comparisons, each test case followed the same basic protocol.

### Test Case Protocol

Tests were performed with zero think time. No operations were performed and no idle think time elapsed between single sign-on and logout. Zero think time maximizes throughput and maintains uniformity throughout the testing.

#### 1. Perform zero-page authentication.

The test script randomly selects a user from the user database. The script constructs the login URL to include the username and password, eliminating the need to invoke a login page. After successful authentication, the script redirects to a blank page. Redirecting to a blank page further minimizes the login transaction time.

## 2. **Initiate SAML2 single sign-on (SSO) protocol.**

This step can be initiated from the Service Provider site or from the Identity Provider site. The test script invokes the SAML2 single sign-on process by accessing the `idpSSOIniti.jsp` file. The `idpSSOIniti.jsp` file redirects the request to the Identity Provider for authentication. Since the user was authenticated in step 1, the request is redirected to the Service Provider with an SAML2 assertion for that user. At the Service Provider site, Federation Manager validates the assertion and then redirects the request to a page with a Success message or to a page with Failure message.

## 3. **Log out.**

The test script initiates a logout request. The request goes to the Identity Provider, and then returns with a Success response or a Failure response.

The performance tests were first conducted using one Service Provider subsystem and one Identity Provider subsystem. The tests were conducted a second time using two Service Provider subsystems and two Identity Provider subsystems. For these tests, a Service Provider subsystem consists of two load-balanced Federation Manager servers. An Identity Provider subsystem consists of two load-balanced Access Manager servers.

## **Test Case Variables**

The test cases were designed to obtain performance numbers based on three variables: provider type, federation protocol, and SAML2 profile.

## **Initiating SAML2 at the Service Provider Site or at the Identity Provider Site**

Whether you initiate SAML2 at the Service Provider site or at the Identity Provider site is determined by the use case. If your company provides services, but does not have a user data repository or an authentication service, you will initiate SAML2 at the Service Provider site. If your company is invested in storing user information and providing authentication service, your users will initiate SAML2 at the Identity Provider site. Two scenarios illustrate the relationship between Identity Provider and Service Provider.

In the first scenario, a company provides no-cost web mail service, but does not maintain a user database nor any authentication mechanism. Such a company is considered a Service Provider. When an Internet user logs in to read his web-based email, the Service Provider initiates SAML2 and directs the request to an independent Identity Provider site for authentication service.

In the second scenario, the Example Company is a large manufacturing enterprise. Example Company contracts with the Siroe Company to administer health benefits for Example Company employees. When a new employee is hired, the employee is instructed to log in to the Siroe Company website and sign up for health care benefits. In this scenario, Example Company stores its employee information in its own database, and has established its own service for authenticating its employees. Example Company is the Identity Provider, and its employees initiate SAML2 from the Identity Provider site. Once the new employee is authenticated, Siroe Company allows the employee to access its secured website, and then proceeds to provide administration services. Siroe Company is the Service Provider.

## **Using Transient Federation Protocol or Persistent Federation Protocol**

When you use transient federation protocols, user federation information is used for one session and then destroyed at the end of the session. No Directory Server access or user search is necessary because the federation information is stored in memory. When the session times out, the user information is destroyed. When you use persistent federation protocols, at the end of a user session, the user information is persistently stored in the Directory Server for retrieval at subsequent sessions. Determining which SAML2 profile to choose to use depends upon your particular use case.

## **Using Web Browser Artifact or Web Browser POST Profile**

A profile is a set of rules that defines how to embed and extract SAML assertions. The profile describes how the assertions are combined with other objects by an authority, transported from the authority, and subsequently processed at the trusted partner site. Access Manager and Federation Manager support two profiles: the Web Browser Artifact Profile and the Web Browser POST Profile. Both profiles use HTTP. Either can be used in single sign-on between two SAML-enabled entities, allowing an authenticated user to access resources from a trusted partner site.

Each profile has advantages and disadvantages. Here are two examples:

- The Web Browser Artifact Profile works without browsers enabled with JavaScript technology. It is considered more secure than the Web Browser POST Profile.
- The Web Browser POST Profile does not require SOAP. This profile is more firewall-friendly and involves fewer steps and less server-side processing.

Before you can determine which profile to use, you should be familiar with all the various profiles and how they work. For detailed information, see the [Sun Java System Access Manager 7 2005Q4 Federation and SAML Administration Guide](#).

## How to Read the Test Results

Test results are grouped based on whether SAML2 is initiated at the Service Provider or at the Identity Provider. For each test case, the following measurements are reported:

**Transactions Per Second.** This is the number of transactions that can be put through the system per second. This measurement is also known as throughput.

**CPU Usage.** This percentage indicates how close the system is to full CPU usage. If any sever at the Service Provider site or at the Identity Provider site reaches 100% CPU usage, then the entire SAML2 federation deployment has reached maximum CPU usage. Directory Server CPU usage is not reported. In all tests, Directory Server CPU usage was negligible and did not impact maximum CPU usage for the whole system.

**Response Times.** The performance tests were designed to measure the time elapsed during each segment of a SAML2 transaction. Response times are reported in milliseconds for the following transaction intervals:

### Identity Provider (IDP) Login

Interval begins when Identity Provider when the user logs in at the Identity Provider site, and ends after the Identity Provider redirects the request to a blank page.

### Identity Provider (IDP) Assertion Generation

Interval begins when the Identity Provider accesses the single sign-on service at the Identity Provider site, and ends after the IDP generates the assertion and sends it back to the Service Provider.

### Service Provider (SP) Assertion Consumption

Interval begins when the Service Provider receives an assertion. The Service Provider processes the assertion, and then generates the user session. The interval ends after a Success message has been displayed.

### Single Sign-On (SSO)

Interval begins when the Service Provider invokes the SAML2 protocol for SSO. The Service Provider processes the assertion, and then generates the user session. The interval ends after a Success message has been displayed.

### HTTP Logout

Interval begins when the HTTP logout request is initiated. The HTTP request goes to the Identity Provider, and comes back with a response. The interval ends when a Success message or a Failure message is displayed.

### SOAP Logout

Interval begins when the SOAP logout request is initiated. The SOAP request goes to the Identity Provider, and comes back with a response. The interval ends when a Success message or a Failure message is displayed.

# Test Results for SAML2 Protocols Initiated at the Service Provider Site

Performance test results are grouped based on transient and persistent federation, and also based on the use of one subsystem or two subsystems.

## Transient Federation Protocols

The following tables summarize test results for transient SAML2 protocols initiated at the Service Provider site:

- Transient Federation Using One Federation Manager Server and One Access Manager Server
- Transient Federation Using Two Federation Manager Servers and Two Access Manager Servers

**TABLE 3** Transient Federation Using One Federation Manager Server and One Access Manager Server

Test Case	Transactions Per Second	CPU Usage	Response Times
Browser POST profile and HTTP Logout	202	Federation Manager 1 – 97% Access Manager 1 – 54%	IDP Login – 38 ms IDP Assertion Generation – 120 ms SP Assertion Consumption – 221 ms HTTP Logout – 105 ms
Browser POST profile and SOAP Logout	170	Federation Manager 1 – 95% Access Manager 1 – 64%	IDP Login – 36 ms IDP Assertion Generation – 118 ms SP Assertion Consumption – 249 ms SOAP Logout – 194 ms
Browser Artifact profile and HTTP Logout	178	Federation Manager 1 – 97% Access Manager 1 – 53%	IDP Login – 38 ms SSO – 425 ms HTTP Logout – 91 ms
Browser Artifact profile and SOAP Logout	152	Federation Manager 1 – 93% Access Manager 1 – 60%	IDP Login – 29 ms SSO – 431 ms SOAP Logout – 178 ms

The following table summarizes performance test results using two Service Provider subsystems and two Identity Provider subsystems. You can compare these results to the results reported in the previous table to see how performance scales as you add more subsystems. When you use two Service Provider subsystems and two Identity Providers subsystems, the throughput is roughly doubled and the response times are reduced by half.

**TABLE 4** Transient Federation Using Two Federation Manager Servers and Two Access Manager Servers

Test Case	Transactions Per Second	CPU Usage	Response Times
Browser POST profile and HTTP Logout	390	Federation Manager 1 – 92% Federation Manager 2 – 70% Access Manager 1 – 60% Access Manager 2 – 53%	IDP Login – 20 ms IDP Assertion Generation – 77 ms SP Assertion Consumption – 99 ms HTTP Logout – 50 ms
Browser POST profile and SOAP Logout	301	Federation Manager 1 – 90% Federation Manager 2 – 72% Access Manager 1 – 53% Access Manager 2 – 51%	IDP Login – 19 ms IDP Assertion Generation – 62 ms SP Assertion Consumption – 131 ms SOAP Logout – 101 ms
Browser Artifact profile and HTTP Logout	332	Federation Manager 1 – 93% Federation Manager 2 – 67% Access Manager 1 – 57% Access Manager 2 – 54%	IDP Login – 20 ms SSO – 245 ms HTTP Logout – 57 ms

TABLE 4 Transient Federation Using Two Federation Manager Servers and Two Access Manager Servers (Continued)

Test Case	Transactions Per Second	CPU Usage	Response Times
Browser Artifact profile and SOAP Logout	285	Federation Manager 1 – 95% Federation Manager 2 – 71% Access Manager 1 – 60% Access Manager 2 – 58%	IDP Login – 19 ms SSO – 251 ms SOAP Logout – 101 ms

## Persistent Federation Protocols

The following tables summarize test results for persistent SAML2 protocols initiated at the Service Provider site:

- Persistent Federation Using One Federation Manager Server and One Access Manager Server
- Persistent Federation Using Two Federation Manager Servers and Two Access Manager Servers

TABLE 5 Persistent Federation Using One Federation Manager Server and One Access Manager Server

Test Case	Transactions Per Second	CPU Usage	Response Times
Browser POST profile and HTTP Logout	172	Federation Manager 1 – 90% Access Manager 1 – 51%	IDP Login – 35 ms IDP Assertion Generation – 76 ms SP Assertion Consumption – 398 ms HTTP Logout – 58 ms
Browser POST profile and SOAP Logout	158	Federation Manager 1 – 92% Access Manager 1 – 52%	IDP Login – 29 ms IDP Assertion Generation – 97 ms SP Assertion Consumption – 397 ms SOAP Logout – 142 ms
Browser Artifact profile and HTTP Logout	158	Federation Manager 1 – 89% Access Manager 1 – 55%	IDP Login – 32 ms SSO – 503 ms HTTP Logout – 75 ms
Browser Artifact profile and SOAP Logout	146	Federation Manager 1 – 92% Access Manager 1 – 52%	IDP Login – 29 ms SSO – 518 ms SOAP Logout – 139 ms

The following table summarizes performance test results using two Service Provider subsystems and two Identity Provider subsystems. You can compare these results to the results reported in the previous table to see how performance scales as you add more subsystems. When you use two Service Provider subsystems and two Identity Providers subsystems, the throughput is roughly doubled and the response times are reduced by half.

TABLE 6 Persistent Federation Using Two Federation Manager Servers and Two Access Manager Servers

Test Case	Transactions Per Second	CPU Usage	Response Times
Browser POST profile and HTTP Logout	335	Federation Manager 1 – 92% Federation Manager 2 – 77% Access Manager 1 – 57% Access Manager 2 – 53%	IDP Login – 19 ms IDP Assertion Generation – 38 ms SP Assertion Consumption – 192 ms HTTP Logout – 26 ms
Browser POST profile and SOAP Logout	287	Federation Manager 1 – 87% Federation Manager 2 – 77% Access Manager 1 – 61% Access Manager 2 – 51%	IDP Login – 19 ms IDP Assertion Generation – 49 ms SP Assertion Consumption – 193 ms SOAP Logout – 99 ms
Browser Artifact profile and HTTP Logout	293	Federation Manager 1 – 89% Federation Manager 2 – 51% Access Manager 1 – 51% Access Manager 2 – 49%	IDP Login – 17 ms SSO – 270 ms HTTP Logout – 37 ms



TABLE 6 Persistent Federation Using Two Federation Manager Servers and Two Access Manager Servers (Continued)

Test Case	Transactions Per Second	CPU Usage	Response Times
Browser Artifact profile and SOAP Logout	265	Federation Manager 1 – 88% Federation Manager 2 – 57% Access Manager 1 – 63% Access Manager 2 – 57%	IDP Login – 19 ms SSO – 261 ms SOAP Logout – 92 ms

## Test Results for SAML2 Protocols Initiated at the Identity Provider Site

Performance test results are grouped based on transient and persistent federation, and also based on the use of one subsystem or two subsystems.

### Transient Federation Protocols

The following tables summarize test results for transient SAML2 protocols initiated at the Identity Provider site :

- Transient Federation Using One Federation Manager Server and One Access Manager Server
- Transient Federation Using Two Federation Manager Servers and Two Access Manager Servers

TABLE 7 Transient Federation Using One Federation Manager Server and One Access Manager Server

Test Case	Transactions Per Second	CPU Usage	Response Times
Browser POST profile and HTTP Logout	206	Federation Manager 1 – 98% Access Manager 1 – 55%	IDP Login – 35 ms IDP Assertion Generation – 63 ms SP Assertion Consumption – 222 ms HTTP Logout – 141 ms
Browser POST profile and SOAP Logout	172	Federation Manager 1 – 95% Access Manager 1 – 53%	IDP Login – 31 ms IDP Assertion Generation – 59 ms SP Assertion Consumption – 266 ms SOAP Logout – 229 ms
Browser Artifact profile and HTTP Logout	181	Federation Manager 1 – 95% Access Manager 1 – 53%	IDP Login – 35 ms SSO – 358 ms HTTP Logout – 127 ms
Browser Artifact profile and SOAP Logout	155	Federation Manager 1 – 90% Access Manager 1 – 55%	IDP Login – 30 ms SSO – 433 ms SOAP Logout – 225 ms

The following table summarizes performance test results using two Service Provider subsystems and two Identity Provider subsystems. You can compare these results to the results reported in the previous table to see how performance scales as you add more subsystems. When you use two Service Provider subsystems and two Identity Providers subsystems, the throughput is roughly doubled and the response times are reduced by half.

TABLE 8 Transient Federation Using Two Federation Manager Servers and Two Access Manager Servers

Test Case	Transactions Per Second	CPU Usage	Response Times
Browser POST profile and HTTP Logout	398	Federation Manager 1 – 97% Federation Manager 2 – 80% Access Manager 1 – 62% Access Manager 2 – 45%	IDP Login – 18 ms IDP Assertion Generation – 35 ms SP Assertion Consumption – 121 ms HTTP Logout – 72 ms

**TABLE 8** Transient Federation Using Two Federation Manager Servers and Two Access Manager Servers *(Continued)*

Test Case	Transactions Per Second	CPU Usage	Response Times
Browser POST profile and SOAP Logout	335	Federation Manager 1 – 87% Federation Manager 2 – 77% Access Manager 1 – 61% Access Manager 2 – 51%	IDP Login – 17 ms IDP Assertion Generation – 32 ms SP Assertion Consumption – 140 ms SOAP Logout – 120 ms
Browser Artifact profile and HTTP Logout	342	Federation Manager 1 – 62% Federation Manager 2 – 92% Access Manager 1 – 53% Access Manager 2 – 52%	IDP Login – 18 ms SSO – 177 ms HTTP Logout – 68 ms
Browser Artifact profile and SOAP Logout	297	Federation Manager 1 – 63% Federation Manager 2 – 90% Access Manager 1 – 55% Access Manager 2 – 53%	IDP Login – 17 ms SSO – 222 ms SOAP Logout – 127 ms

## Persistent Federation Protocols

The following tables summarize test results for transient SAML2 protocols initiated at the Identity Provider site:

- Persistent Federation Using One Federation Manager Server and One Access Manager Server
- Persistent Federation Using Two Federation Manager Servers and Two Access Manager Servers

**TABLE 9** Persistent Federation Using One Federation Manager Server and One Access Manager Server

Test Case	Transactions Per Second	CPU Usage	Response Times
Browser POST profile and HTTP Logout	173	Federation Manager 1 – 95% Access Manager 1 – 45%	IDP Login – 29 ms IDP Assertion Generation – 42 ms SP Assertion Consumption – 454 ms HTTP Logout – 57 ms
Browser POST profile and SOAP Logout	163	Federation Manager 1 – 91% Access Manager 1 – 53%	IDP Login – 29 ms IDP Assertion Generation – 50 ms SP Assertion Consumption – 397 ms SOAP Logout – 182 ms
Browser Artifact profile and HTTP Logout	153	Federation Manager 1 – 85% Access Manager 1 – 45%	IDP Login – 36 ms SSO – 519 ms HTTP Logout – 74 ms
Browser Artifact profile and SOAP Logout	148	Federation Manager 1 – 94% Access Manager 1 – 53%	IDP Login – 28 ms SSO – 479 ms SOAP Logout – 162 ms

The following table summarizes performance test results using two Service Provider subsystems and two Identity Provider subsystems. You can compare these results to the results reported in the previous table to see how performance scales as you add more subsystems. When you use two Service Provider subsystems and two Identity Providers subsystems, the throughput is roughly doubled and the response times are reduced by half.

**TABLE 10** Persistent Federation Using Two Federation Manager Servers and Two Access Manager Servers

Test Case	Transactions Per Second	CPU Usage	Response Times
Browser POST profile and HTTP Logout	349	Federation Manager 1 – 91% Federation Manager 2 – 81% Access Manager 1 – 49% Access Manager 2 – 48%	IDP Login – 17 ms IDP Assertion Generation – 22 ms SP Assertion Consumption – 241 ms HTTP Logout – 27 ms

**TABLE 10** Persistent Federation Using Two Federation Manager Servers and Two Access Manager Servers  
(Continued)

Test Case	Transactions Per Second	CPU Usage	Response Times
Browser POST profile and SOAP Logout	318	Federation Manager 1 – 87% Federation Manager 2 – 71% Access Manager 1 – 61% Access Manager 2 – 51%	IDP Login – 18 ms IDP Assertion Generation – 27 ms SP Assertion Consumption – 218 ms SOAP Logout – 101 ms
Browser Artifact profile and HTTP Logout	292	Federation Manager 1 – 62% Federation Manager 2 – 92% Access Manager 1 – 53% Access Manager 2 – 52%	IDP Login – 18 ms SSO – 262 ms HTTP Logout – 41 ms
Browser Artifact profile and SOAP Logout	275	Federation Manager 1 – 63% Federation Manager 2 – 90% Access Manager 1 – 55% Access Manager 2 – 53%	IDP Login – 17 ms SSO – 278 ms SOAP Logout – 95 ms

## Using the Performance Test Results for Guidance

As you make decisions regarding hardware and software in your SAML2 federation environment, consider the main factors that impact throughput and response times:

- Provider types
- Federation protocols
- SAML2 profiles

At the conclusion of the performance tests on these variables, a number of generalizations became apparent. Consider these generalizations along with the other information presented in this article when performance tuning your SAML2 environment.

- The throughput is significantly higher on Niagara host systems than on V480 host systems.
- Transient federations achieve higher throughput than persistent federations. An estimated 20% overhead cost occurs in user searches during persistent federations. This occurs more frequently on the Service Provider computer systems. Using a faster Directory Server at the Service Provider site may improve response times.
- Profiles initiated at the Identity Provider site yield better performance results than profiles initiated at the Service Provider site. This is a result of having one less redirection to perform.
- SOAP logout overhead cost is 25% higher than HTTP logout overhead cost.
- The browser POST profiles are more efficient than browser Artifact profiles because the Artifact profiles have an extra SOAP call for the Artifact resolution.
- The Service Provider CPU reaches its maximum usage in most of the test cases.
- The response time for the Assertion Consumer service is higher than the response time for other transactions.
- When you use two Service Provider subsystems and two Identity Providers subsystems, the throughput is roughly doubled, and the response times are reduced by one-half.
- When you scale the number of users in the user repository upward from 100,000 (for example up to one million), the throughput number scales proportionately with negligible impact on response times.
- You can use the tables in this report to estimate the number of subsystems you need in your deployment to reach a transactions/second target. Example:

Target: 800 transactions per second  
Initiated at: Service Provider  
Federation type: persistent  
Profile: browser POST and HTTP logout

For this example, you can use the following formula:

$$800/172 = \sim 5$$

where 800 is the target number of transactions, and 172 (from Table 3) is the number of transactions/seconds yielded in your deployment. The result indicates you will need five Service Provider subsystems and five Identity Provider systems to reach the target number of transactions.

- The performance tests were conducted with high maximum values for session timeout and idle timeout so that the system could run out of memory. The results indicate that the system cannot hold more than 96K concurrent sessions in memory. One can estimate that with a heap size of 3GB and with 96K active user sessions, each active user session is taking approximately  $3\text{GB}/96\text{K}$ , or 34KB, of memory. You can use this as a rough formula for estimating the memory sizing. See the *Sun Java System Access Manager 7.1 Performance Tuning Guide* for more information.

## JVM Tuning Recommendations for Niagara Hardware

If you are using Niagara computer systems in your environment, you may encounter “Out of memory” error messages on servers at the Identity Provider site. This is a known problem that may be due to a problem with the Niagara system garbage collector. As a workaround, you can modify the amtune script, and run the script on all servers hosted on Niagara computer systems.

### Service Provider

#### EXAMPLE 1 Service Provider Example

```
<JVMOPTIONS>-Xms3136M</JVMOPTIONS>
  <JVMOPTIONS>-Xmx3136M</JVMOPTIONS>
  <JVMOPTIONS>-Xloggc:/opt/SUNWwbsvr/https-am-ng-02.red.ipplanet.com/logs/gc.log
    </JVMOPTIONS>
  <JVMOPTIONS>-server</JVMOPTIONS>
<JVMOPTIONS>-Xss128k</JVMOPTIONS>
  <JVMOPTIONS>-XX:NewSize=750M</JVMOPTIONS>
  <JVMOPTIONS>-XX:MaxNewSize=750M</JVMOPTIONS>
  <JVMOPTIONS>-XX:+DisableExplicitGC</JVMOPTIONS>
  <JVMOPTIONS>-XX:+UseParNewGC</JVMOPTIONS>
  <JVMOPTIONS>-XX:+UseConcMarkSweepGC</JVMOPTIONS>
  <JVMOPTIONS>-XX:+PrintClassHistogram</JVMOPTIONS>
  <JVMOPTIONS>-XX:+PrintGCTimeStamps</JVMOPTIONS>
  <JVMOPTIONS>-XX:ParallelGCThreads=18</JVMOPTIONS>
  <JVMOPTIONS>-XX:SurvivorRatio=8</JVMOPTIONS>
```

### Identity Provider

```
<JVMOPTIONS>-Dcom.ipplanet.am.serverMode=true</JVMOPTIONS>
  <JVMOPTIONS>-Xms2048M -Xmx2048M</JVMOPTIONS>
  <JVMOPTIONS>-Xloggc:/opt/SUNWwbsvr/https-am-ng-01.red.ipplanet.com/logs/gc.log
    </JVMOPTIONS>
  <JVMOPTIONS>-server</JVMOPTIONS>
  <JVMOPTIONS>-Xss128k</JVMOPTIONS>
  <JVMOPTIONS>-XX:NewSize=128M</JVMOPTIONS>
  <JVMOPTIONS>-XX:MaxNewSize=128M</JVMOPTIONS>
  <JVMOPTIONS>-XX:+UseConcMarkSweepGC</JVMOPTIONS>
  <JVMOPTIONS>-XX:+UseParNewGC</JVMOPTIONS>
  <JVMOPTIONS>-XX:+CMSPermGenSweepingEnabled</JVMOPTIONS>
  <JVMOPTIONS>-XX:+CMSClassUnloadingEnabled</JVMOPTIONS>
  <JVMOPTIONS>-XX:+CMSParallelRemarkEnabled</JVMOPTIONS>
  <JVMOPTIONS>-XX:SurvivorRatio=1024</JVMOPTIONS>
  <JVMOPTIONS>-XX:MaxTenuringThreshold=0</JVMOPTIONS>
  <JVMOPTIONS>-XX:+UseCMSCompactAtFullCollection</JVMOPTIONS>
  <JVMOPTIONS>-XX:CMSFullGCsBeforeCompaction=0</JVMOPTIONS>
  <JVMOPTIONS>-XX:SoftRefLRUPolicyMSPerMB=0</JVMOPTIONS>
  <JVMOPTIONS>-verbose:gc -XX:+PrintClassHistogram</JVMOPTIONS>
  <JVMOPTIONS>-XX:+PrintGCTimeStamps</JVMOPTIONS>
  <JVMOPTIONS>-XX:+PrintGCDetails</JVMOPTIONS>
  <JVMOPTIONS>-XX:PermSize=128M</JVMOPTIONS>
  <JVMOPTIONS>-XX:MaxPermSize=128M</JVMOPTIONS>
```

Copyright 2007 Sun Microsystems, Inc. All rights reserved. Sun Microsystems, Inc. has intellectual property rights relating to technology embodied in the product that is described in this document. In particular, and without limitation, these intellectual property rights may include one or more U.S. patents or pending patent applications in the U.S. and in other countries.

U.S. Government Rights – Commercial software. Government users are subject to the Sun Microsystems, Inc. standard license agreement and applicable provisions of the FAR and its supplements.

This distribution may include materials developed by third parties.

Parts of the product may be derived from Berkeley BSD systems, licensed from the University of California. UNIX is a registered trademark in the U.S. and other countries, exclusively licensed through X/Open Company, Ltd.

Sun, Sun Microsystems, the Sun logo, the Solaris logo, the Java Coffee Cup logo, docs.sun.com, Java, and Solaris are trademarks or registered trademarks of Sun Microsystems, Inc. in the U.S. and other countries. All SPARC trademarks are used under license and are trademarks or registered trademarks of SPARC International, Inc. in the U.S. and other countries. Products bearing SPARC trademarks are based upon an architecture developed by Sun Microsystems, Inc.

The OPEN LOOK and Sun<sup>TM</sup> Graphical User Interface was developed by Sun Microsystems, Inc. for its users and licensees. Sun acknowledges the pioneering efforts of Xerox in researching and developing the concept of visual or graphical user interfaces for the computer industry. Sun holds a non-exclusive license from Xerox to the Xerox Graphical User Interface, which license also covers Sun's licensees who implement OPEN LOOK GUIs and otherwise comply with Sun's written license agreements.

Products covered by and information contained in this publication are controlled by U.S. Export Control laws and may be subject to the export or import laws in other countries. Nuclear, missile, chemical or biological weapons or nuclear maritime end uses or end users, whether direct or indirect, are strictly prohibited. Export or reexport to countries subject to U.S. embargo or to entities identified on U.S. export exclusion lists, including, but not limited to, the denied persons and specially designated nationals lists is strictly prohibited.

DOCUMENTATION IS PROVIDED "AS IS" AND ALL EXPRESS OR IMPLIED CONDITIONS, REPRESENTATIONS AND WARRANTIES, INCLUDING ANY IMPLIED WARRANTY OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT, ARE DISCLAIMED, EXCEPT TO THE EXTENT THAT SUCH DISCLAIMERS ARE HELD TO BE LEGALLY INVALID.

Part No:

Sun Microsystems, Inc. 4150 Network Circle, Santa Clara, CA 95054 U.S.A.

