Single Sign-On (SSO) from SAP Enterprise Portal to Lotus Domino – a Comparison of Alternatives

Michael Sambeth, Stefan Thomas

NetWeaver Practice Unit Enterprise Portal, SAP Deutschland AG & Co. KG

Summary

Single sign-on plays an important role in an integration project, e.g. when implementing mySAP ERP and SAP NetWeaver. In addition, SAP NetWeaver serves as an open integration platform for non-SAP technologies like Lotus Notes/Domino. To this end, single sign-on scenarios are also required. SAP thereby offers a choice of technical alternatives with different characteristics. Either way, access to Lotus Domino is granted with single sign-on for SAP users. Users of the SAP Enterprise Portal can now take advantage of an integrated Lotus experience thereby leveraging existing Lotus system, services and data resources.

This paper describes the different alternative solutions SAP offers. It is intended to guide the project team or solution architect to the right choice of technology.

Applies to

- SAP Enterprise Portal 6.0 SP2 Patch 4 and higher (Basis 6.20)
- SAP Enterprise Portal 6.0 SPS5 and higher (Basis 6.40)
- Lotus Domino R5 (5.0.10 and higher) and Lotus 6.x (not Lotus Workplace)

Keywords

Lotus Notes, Lotus Domino, Single Sign-on, SSO, SAP Web Application Server, SAP Enterprise Portal, SAP J2EE

Level of difficulty

Medium

Contents

Introduction

User Management plays a central and vital role in SAP NetWeaver infrastructure and SAP Enterprise Portal projects. User management typically relies on an LDAP Directory which is different from the Lotus Domino Directory, e.g. Microsoft Active Directory or Novell eDirectory.[1]

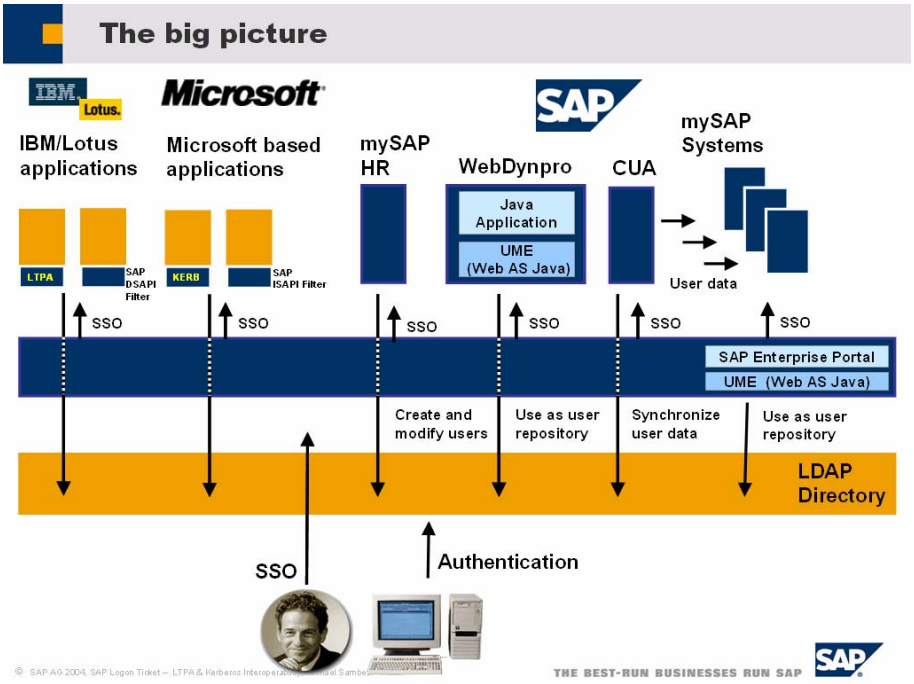

Figure 1: Unified User Management and single sign-on

The key idea for using a Unified User Management is that the resulting landscape leverages both native tokens. This is SAP Logon Ticket and the Lotus LtpaToken. Since the vendor's systems typically accept the vendor's native tokens with no or minor customizing, it is guaranteed that modifications to the system landscape are kept to a minimum.

[1] The Lotus Domino Directory (LDAP task on Domino) can be integrated on project base.

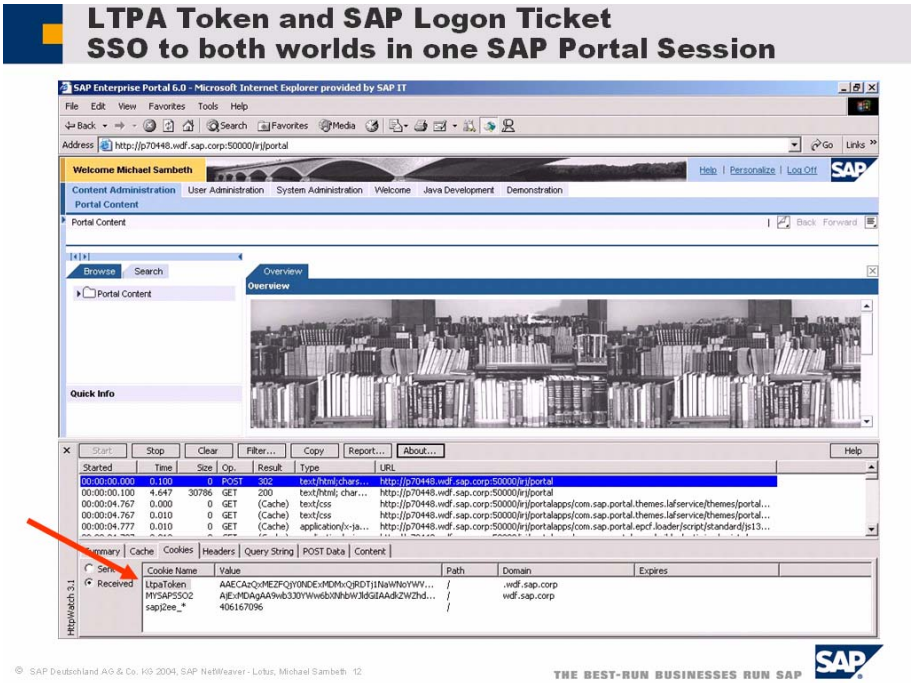

Figure 2: Token of SAP and Lotus world in one SAP Enterprise Portal Session

The goal of a Unified User Management is to give the interacting portal user access to all integrated systems with single sign-on (SSO). SAP Logon Tickets can serve as authentication tokens against Lotus Systems. All SAP applications and various non-SAP applications support SAP Logon Tickets as SSO mechanism. The user credentials that are contained in a valid SAP Logon Ticket can be used by an external application using SAP's Lotus Domino Web Server Filter (SAP Ticket verifier for Lotus Domino).

The SAP Enterprise Portal issues a SAP Logon Ticket to a user after successful initial authentication at the portal against a user persistence specified in the SAP Web Application Server User Management Engine (UME). The SAP Logon Ticket that contains the portal user id of the authenticated user is stored as session cookie on the client browser. The authenticity and integrity is protected using digital signatures whereas the confidentiality of the token is protected through the use of the SSL protocol while in transport. As a third measure the SAP Logon Ticket contains a validity period that can be configured in the security settings of the SAP Enterprise Portal.

Alternatively, SAP Enterprise Portal users can directly authenticate against Lotus Systems with a username and password. This can be done either against the HTTP task (web authentication) or against the DIIOP task (IIOP authentication).

Scenarios

This section describes the different technical scenarios. For all options it is necessary that Lotus Domino is configured for Multi-Server-Session Authentication with LtpaToken.

Since the SAP Logon Ticket as well as the LtpaToken are implemented as session cookies in the browser request, they can be leveraged in both communication channels:

- Browser → Lotus Domino
- SAP Enterprise Portal Server → Lotus Domino

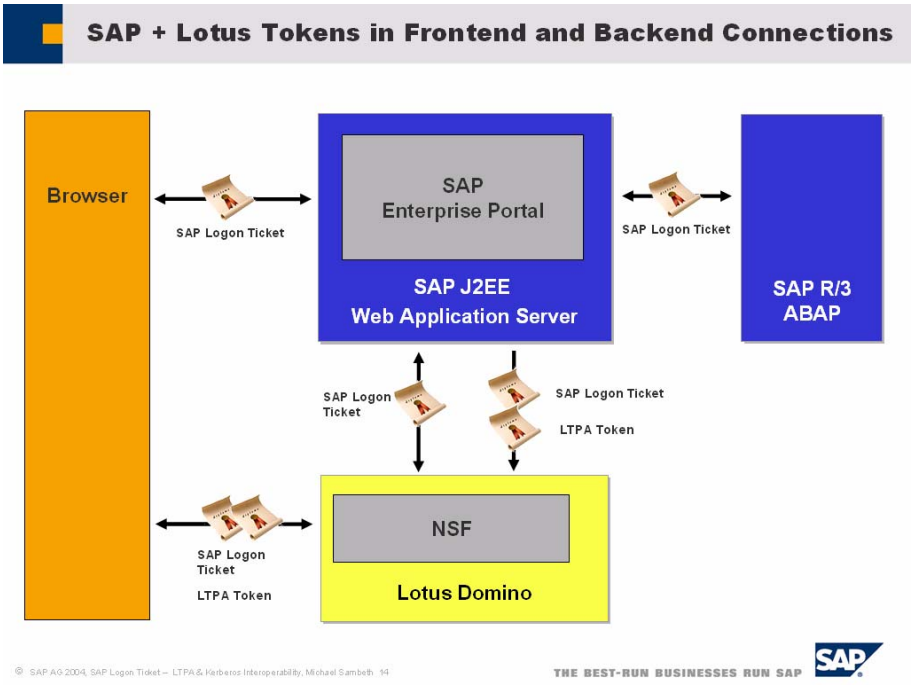

Figure 3: Architectural Setup: SSO in frontend and backend connections

SAP Logon Ticket as well as LtpaToken are supported for single sign-on (SSO) for HTTP access as well as non-HTTP access:

- SAP Logon Ticket: HTTP, RFC, DIAG
- LtpaToken: HTTP, DIIOP

Please note that the LtpaToken is also supported by:

- Lotus Sametime
- Lotus Quickplace
- Domino Web Access

This makes the tokens a flexible and secure way for implementing single sign-on.

Scenario 1: SAP Ticketverifier for Lotus Domino for LtpaToken Generation

Logical Components

The solution comprises of the following components:

- SAP Enterprise Portal 5.0 / 6.0 or SAP Web Application Server 6.20 / 6.30 / 6.40
- Lotus Domino Server R5/R6
- SAP Ticketverifier for Lotus Domino DSAPI Filter
- Security certificate(s) key store

The following picture outlines the scenario:

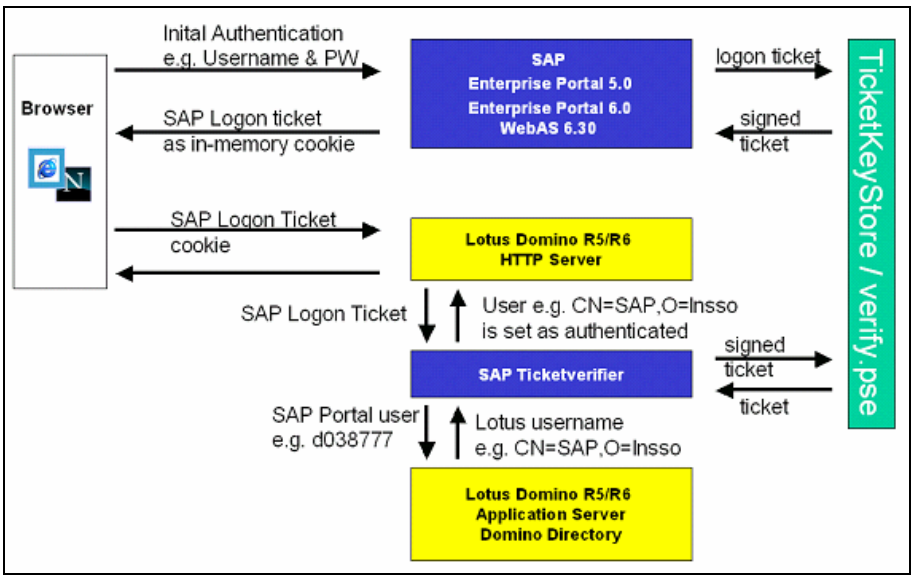

Figure 4: SAP Ticketverifier for Lotus Domino

Single-sign-on information is carried within the SAP Logon Ticket that is stored as an encrypted cookie in the web browser. SAP Logon Ticket can be described as a piece of information used for user authentication and single sign-on with SAP Systems. The logon ticket is issued to a user when he or she logs onto an SAP System that is configured to create tickets (for example, the SAP Web Application Server or SAP Enterprise Portal).

To achieve SSO, the SAP Ticketverifier reads the cookie, gets the SAP Logon Ticket and performs a user lookup with the SAP user stored in the logon ticket in the Domino directory (a.k.a. public name and address book). Finally, the SAP Ticketverifier logs the SAP user on to Lotus Domino using the user's full canonical name. There is no need for a Lotus Domino password.

Attention: If there is no SAP Logon Ticket cookie in the HTTP request header, the SAP Ticketverifier passes the authentication request back to the Lotus Domino server → access to Lotus Domino resources remains unchanged. The SAP Ticketverifier is

triggered only by the existence of a SAP Logon Ticket cookie in an HTTP request that requires authentication.

Flow diagram

The following sequence illustrates how the SAP Ticketverifier for Lotus Domino works.

The functionality of the SAP Ticketverifier is to catch Lotus Domino Webserver authentication requests and to handle these requests, so that the authentication is not on behalf of the Domino Server anymore. Given an HTTP Request to an access-protected Lotus Notes DB / URL, the SAP Ticketverifier:

1. Parses the HTTP request header
2. Extracts the SAP Logon Ticket cookie from the request header
3. Decrypts that cookie using the SAP Enterprise Portal's "verify.pse" key store
4. Verifies the validity of the SAP Logon Ticket (expiration etc.)
5. Gets the SAP username stored in the ticket
6. Performs a lookup in the Lotus Domino directory (names.nsf) with the SAP username
7. Logs the matching Domino user on (full canonical name), if a match can be found
8. Continues with processing of the requested URL

Summarizing: the SAP Ticketverifier replaces the Lotus Domino authentication mechanism to enable SSO to SAP Enterprise Portal users. It can be applied to all Domino Server releases SAP supports.

Technical Components

The technical components of this solution consist of two parts:

1. Runtime components: components that are executed during operation
2. Security components: information that is shared between SAP Enterprise Portal and Lotus Domino to ensure security

In the following sections, these two component classes are explained.

Runtime components

The runtime environment of the SAP Ticketverifier consists of a Microsoft Windows DLL or Unix shared library. In detail, all the needed libraries are as follows:

Library	Filename	Description
SAP Ticketverifier for Lotus Domino R5/R6	ds_ticket_204.dll	Implementation of the SAP Ticketverifier for Lotus Domino
SAP Seculib	sapsecu.dll	Functions for working with the verify.pse key store (public key infrastructure)
MySAP.com-SSO	wpsso_v3.dll	Implementation of mySAP.com logon ticket handling functions

As mentioned before, the server running the SAP Ticketverifier for Lotus Domino has to run all these libraries. The SAP Ticketverifier will be registered within the Lotus Domino server and the other libraries will be put in the file system of the operating system.

Security components

The security of the SAP Ticketverifier solution is based on a public key infrastructure (PKI) and a trust relationship:

- The trust relationship between the mySAP Enterprise Portal and Lotus Domino is established by copying the SAP Enterprise Portal certificate(s) from the SAP Enterprise Portal to Lotus Domino.
- Security is achieved through digitally signing the SAP Enterprise Portal user's credentials

The SAP Ticketverifier needs to have access to the SAP Enterprise Portal's PKI certificate/key store that is stored in a file called "verify.pse". You have to copy that file from the SAP Enterprise Portal machine to the Lotus Domino machine.

- Enterprise Portal 5.0: Verify.pse file can be found under the IRJ tree of the Enterprise Portal in the "data" directory of the usermanagement service.
- Enterprise Portal 6.0: Verify.pse file can be found in the "cluster/server/ume" directory of the J2EE Engine (j2ee\j2ee_<instance_number>\ume).
- Enterprise Portal 6.0 SP2 and higher: Verify.pse has to be downloaded from the portal by choosing: System administration → System configuration → Keystore administration → Download verify.pse file

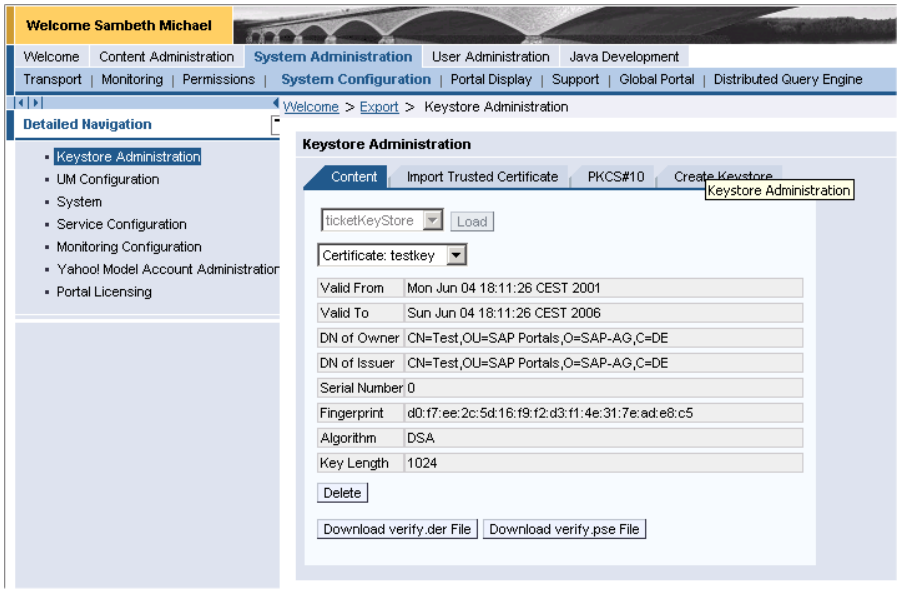

Figure 5: Keystore Administration in EP6.0 SP2

As mentioned before, Lotus Domino ACLs and user privileges are not affected by the SAP Ticketverifier.

Scenario 2: SAP JAAS Module for LtpaToken Generation

The SAP JAAS Login Module for LtpaToken generation runs as a component in the User Management Engine of the SAP Web Application Server. The JAAS Module is currently available for SAP Web Application Server 6.20 and 6.40.[2] The Module requires the DIIOP task and connectivity to be configured on the Lotus Domino Server.

Logical Components

The solution comprises of the following components like

- SAP Enterprise Portal 6.0 (Basis 6.20 / 6.40)
- Lotus Domino Server R6

The following picture outlines the scenario:

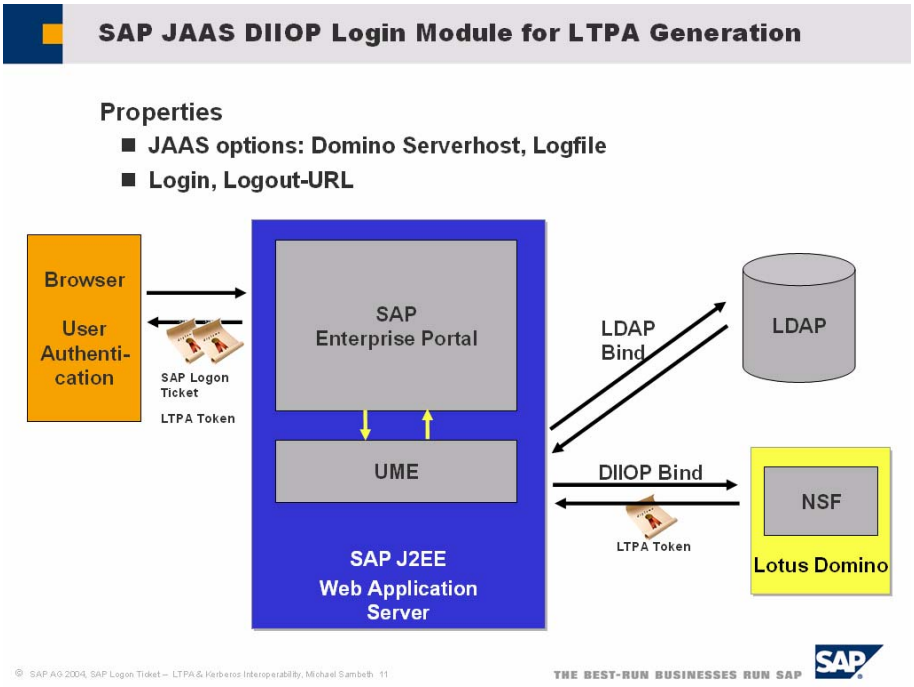

Figure 6: SAP JAAS Module for LTPA Generation

Flow diagram

The following sequence illustrates how the authentication and SSO Token retrieval process for Lotus Domino works.

The functionality of the SAP JAAS LTPA Module for Lotus Domino is to let Lotus Domino generate an LTPA SSO Token for the user that is authenticated against SAP Enterprise Portal. Given an HTTP Request to the SAP Enterprise Portal Authentication, the SAP JAAS LTPA Module for Lotus Domino:

[2] The SAP JAAS Module for LtpaToken Generation can be integrated on project base.

1. Gets the typed credentials (UID, PW) from the Callback Handler to the Portal
2. Reads properties about host and port from the UME properties
3. Initiates an DIIOP connection to Domino using the credentials for authentication
4. Gets the LTPA Token from the Domino Session
5. Returns the LTPA Token as a cookie to the portal server and end-user browser

Summary: The SAP JAAS LTPA Module for Lotus Domino Ltpa Generation enables subsequent SSO to Domino for SAP Enterprise Portal users. It can be applied to all available Domino and SAP releases if DIIOP is configured accordingly and if SAP Enterprise Portal and Domino credentials match.

Technical Components

The technical components of this solution consist of:

- Runtime components: components that are executed during operation

In the following sections, these component classes are explained.

Runtime Components

The runtime environment of the SAP JAAS LTPA Module for Lotus Domino consists of a JAR file and the Domino Notes Client-side Objects for Java (NCSO). In detail, all the needed libraries are as follows:

Library	Filename	Description
SAP JAAS LTPA Module for Lotus Domino Ltpa Generation	com.sap.consulting. portal.ltpasso. LtpaSSOLoginModule.jar	Implementation of the SAP JAAS LTPA Module for Lotus Domino
Notes Client-side objects for Java	NCSO.jar	CORBA Stubs for DIIOP communication of Java application Server with Domino

As mentioned before, the Domino has to run the DIIOP task. Lotus NCSO has to be deployed on SAP Web Application Server. NCSO is shipped with the Lotus Domino Server and can be found in the domino/java directory of a Domino server.

Scenario 3: SAP User Mapping iView for LtpaToken Generation

If for any reason SAP Logon Tickets as well as the SAP JAAS Module are not applicable SAP NetWeaver offers a feature called User Mapping/Account Aggregation. The User Management database instance of the User Management Engine of SAP Enterprise Portal Server offers a secure way to store usernames and passwords which can be used to access backend as well as frontend systems. User Mapping can be maintained either by the end user or by the administrator or by both.

Logical Components

The solution comprises of the following components like

- SAP Enterprise Portal 5.0 / 6.0 or SAP Web Application Server 6.20 / 6.30 / 6.40
- Lotus Domino Server R5/R6

Once the User Mapping is maintained it can be applied to any iView. SAP Consulting Germany developed an iView which perfectly suits Lotus Domino so that User Mapping can be applied when accessing Lotus Domino via the standard Domino HTTP form.[3]

The following picture outlines the scenario:

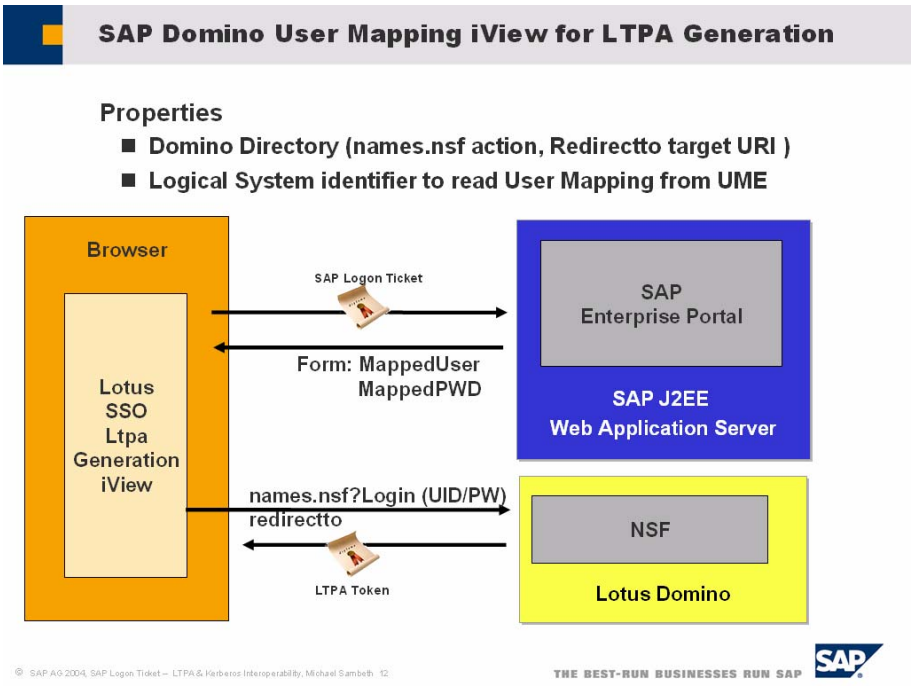

Figure 7: Application of SAP User Mapping to authenticate to Domino and to generate the Ltpa Token

[3] The SAP Domino User Mapping iView for LtpaToken Generation can be integrated on project base.

A sample instance of this SAP iView can look like the following example which authenticates against names.nsf with the HTTP task and then redirects to log.nsf:

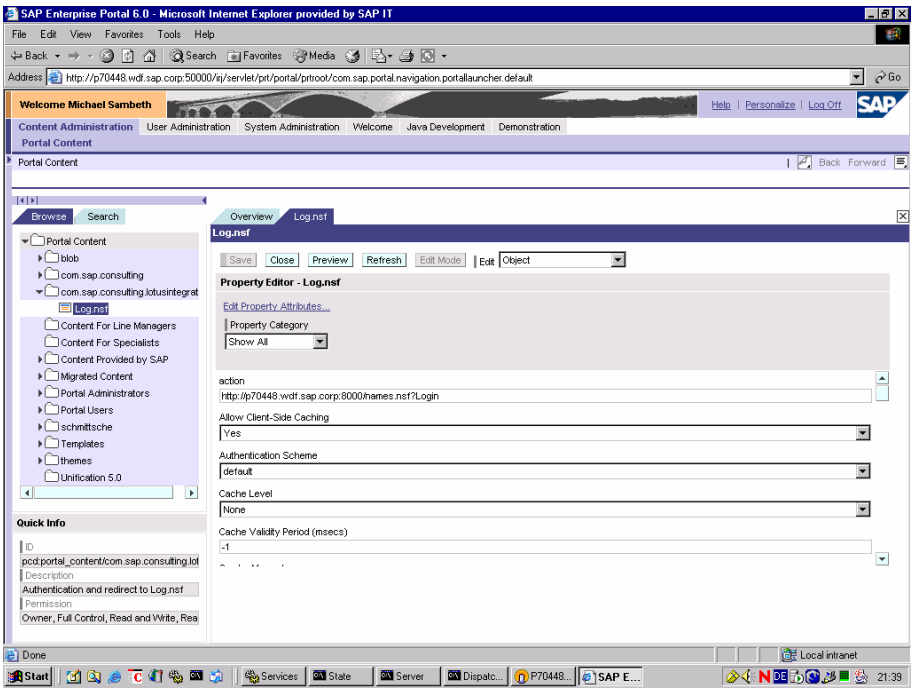

Figure 8: SAP iView for Ltpa Generation with User Mapping

Technically, SAP Enterprise Portal automatically fills and submits the standard Lotus Domino Login form on behalf of the authenticated portal user. Domino returns its session and/or LtpaToken if configured accordingly. Since the communication relies on the standard HTTP forms it can be applied without any modification on SAP or Lotus side. In addition, SAP Enterprise Portal User Mapping allows maintaining user names and credentials different from the SAP or LDAP credentials used when authenticating against SAP Enterprise Portal / SAP Web Application Server.

Summary: the SAP Domino User Mapping iView for Ltpa Generation enables subsequent SSO to Domino for SAP Enterprise Portal users. It can be applied to all Domino and SAP releases.

Scenario 4: Using IBM Tivoli WebSeal Access Manager to generate LtpaToken

If an IBM Tivoli WebSeal Access Manager is in place, e.g. to serve as authentication gateway to SAP Enterprise Portal or SAP Web Application Server, it can be leveraged to generate the LtpaToken as well.

Logical Components

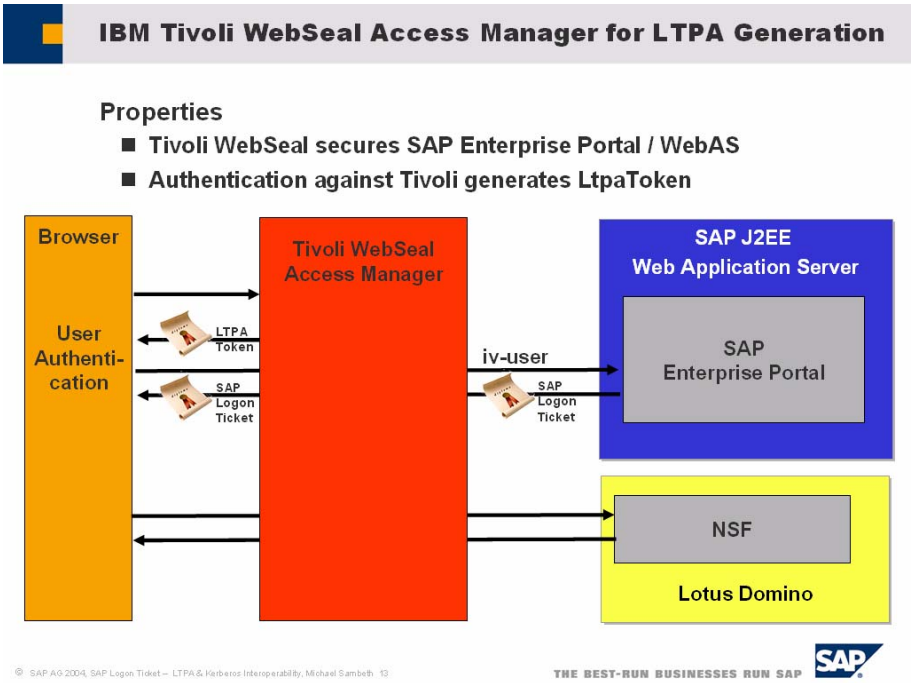

Figure 9: LtpaGeneration with IBM Tivoli WebSeal

The Domino Server has to be configured to accept the LtpaToken that is generated by Tivoli WebSeal.

Comparison of Alternatives

The three options SAP offers apply for different scenarios and there is no scenario where SAP's technology is not an option. However, often two or all three options can be applied and that it is a matter for the customer solution architect to decide.

The closer differentiation has to take into account two aspects:

1. The way users authenticate to SAP Enterprise Portal or SAP Web Application Server
2. Constraints regarding the Domino Server configuration

Applying the SAP Ticketverifier for Lotus Domino which is based on SAP Logon Tickets gives the maximum flexibility regarding 1). This means that the SAP Ticketverifier for Lotus Domino can always be applied no matter how authentication to SAP Enterprise Portal / SAP Web Application Server took place.

When using the SAP JAAS Login Module for LtpaToken generation it is necessary to configure DIIOP access on Domino. So the SAP JAAS module has constraint 2). In addition, the Username+Password pair which is used to authenticate against SAP Enterprise Portal/SAP Web Application Server has to be also valid for authentication against Lotus Domino. If the passwords differ one has to apply User Mapping.

Applying a Web Access Management Product like Tivoli WebSeal Access Manager puts the responsibility for LtpaToken back to IBM. In that case SAP relies on the IBM and Lotus products to be configured accordingly.

Conclusion

SAP NetWeaver (SAP Enterprise Portal and SAP Web Application Server) serves as an end to end solution for SSO for both SAP and Lotus backend and frontend applications. No mandatory 3rd party software is needed to set up a Single Sign-On solution to access SAP and Lotus Domino based backend systems. However, if such a solution is in place (e.g. IBM Tivoli WebSeal) it can be leveraged. Each solution offering thereby features different characteristics. Please contact the SAP – IBM CTSC Collaboration Technology Support Center or SAP Consulting Germany to find out more.

SAP NetWeaver serves as an open integration platform for non-SAP technologies like Lotus Notes/Domino. SAP thereby ensures that the investments in a Lotus infrastructure can be preserved. Lotus content, applications and data can be made available in backend (data), frontend (user interface) and services of portal applications. Lotus Domino comes into play in a wide set of scenarios – integrated with even SAP R/3 ABAP. And there exist many more scenarios (e.g. portalized applications, self-developed Lotus applications, Lotus collaboration products like Sametime) which will be covered in additional documents, collaboration briefs and blogs on the SAP Developer Network (sdn.sap.com).

References

- SAP-IBM Collaboration Technology Support Center: http://service.sap.com/ibm
- SAP Developer Network IBM Interoperability Area
  http://www.sdn.sap.com/sdn/developerareas/ibm.sdn
- SAP Ticketverifier for Lotus Domino:
  http://help.sap.com/saphelp_erp2004/helpdata/en/db/b42e90fb94534687ef57991a6fcaf9/frameset.htm
- SAP JAVA Ticketverifier Classes:
  http://media.sdn.sap.com/html/submitted_docs/60_sp2_javadocs/ume/com/sap/security/api/ticket/TicketVerifier.html
- SAP Note: 696294
  http://service.sap.com/notes