

User AutoAliasMapping

Jeremy Ma (jeremy.ma@sap.com)
May 16, 2012



Problem Statement

Accounts in two separate user store are different. This create two unique login identity in SAPBI



I want end to end SSO??

-I don't want to maintain two accounts !

-Don't want confuse user inbox\favorites

-Provides end to end SSO from multiple entry point

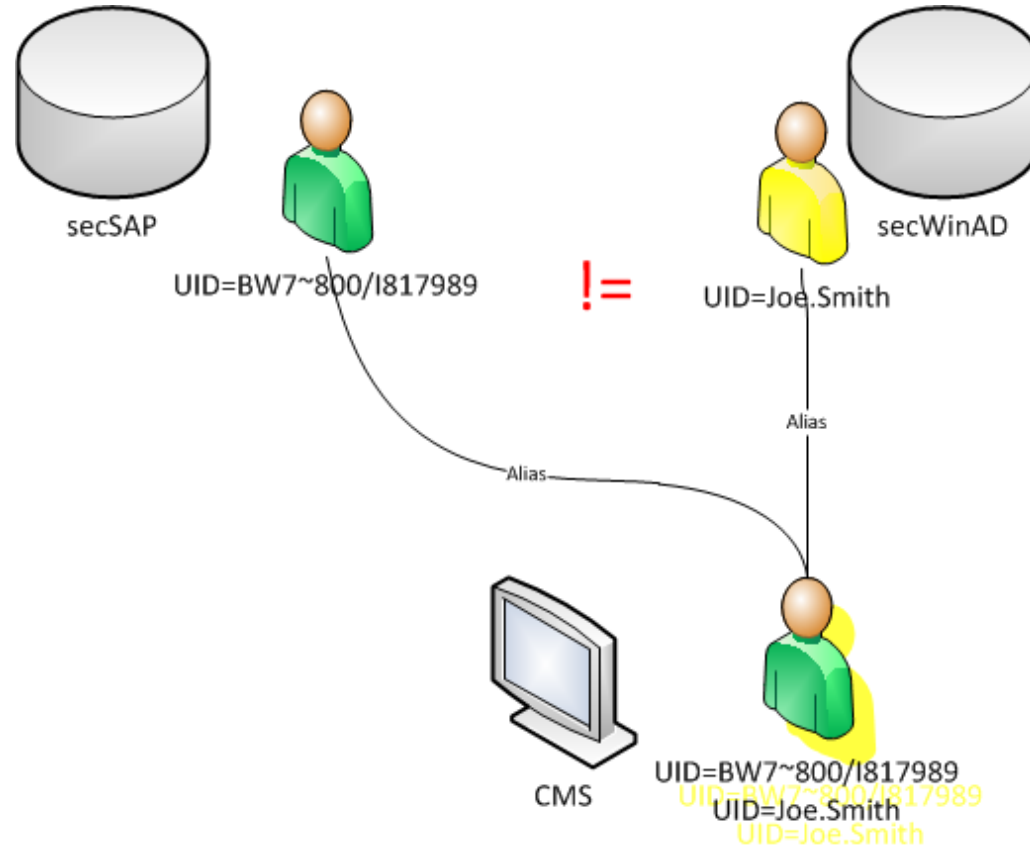
Standard Solution

Below are two out of box configuration options to auto merge the User ID - depends in your deployment scenario:

- If user ID is different by matter of system ID BW~001/ You can apply the SimpleUsernameFormat on the BOE server SAP Note (1343537)
- Instead of using the default User ID populated by WindowAD or LDAP to import the users, Does using another LDAP Attribute will give you the match as SAP\BW system (example employeeID filed in LDAP). This is configure via custom LDAP mapping in CMC>Authenticaiton. Below are some of limitations:
 - User will have to use alternate login ID when prompted in BILaunch Pad or client tools
 - Because Windows standard login ID is not going to be used, there might be challenge in setting up SSO (where to get employeeID in HTTP header)
 - LDAP will be used instead of WinAD this may also further limit Kerberos into DB (HANA ;MsSQL)

3rd Solution ...

Consolidate the accounts by linking Aliases. But this is a manual process and how can I automated for 10K users!!!

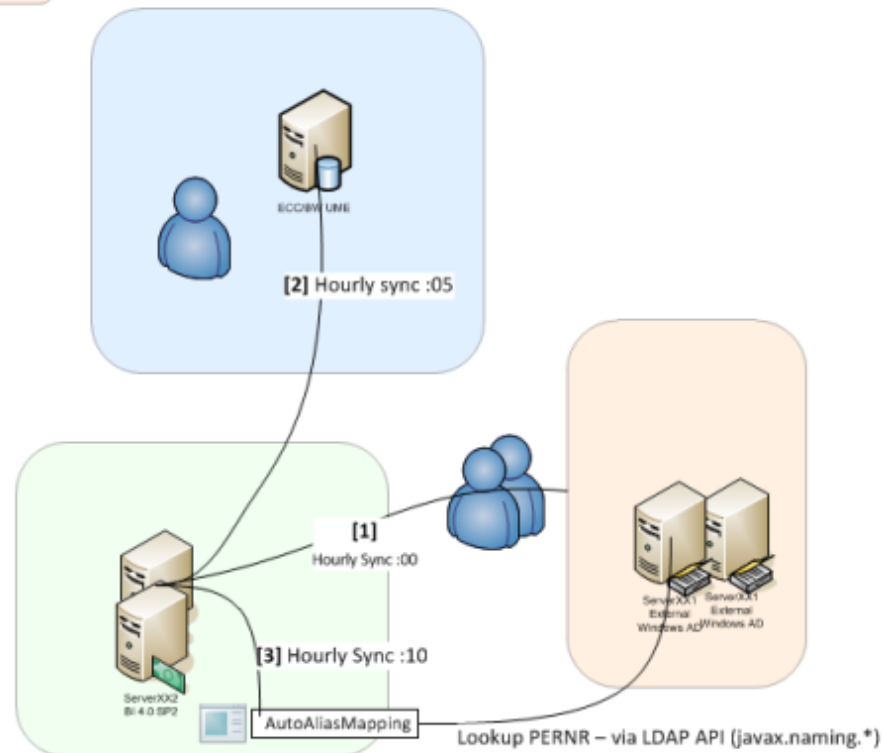


Automate this Solution - Design

SAP BI

SAP BW/ECC

security and WinAD servers

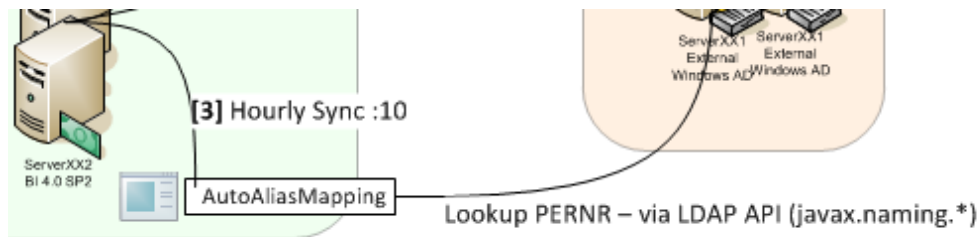


AutoAliasMapping:

Search for All Accounts in CMS that have only SAPAlias:

- 1) Look up employeeID from AD (PERNR) by SAP alias; Search for existing ADAlias in CMS; if found delete SAPAlias; reattached to existing ADAlias
 - 2) Look up First.LastName that match first 12 char of existing ADAlias in AD; if found delete SAP Alias; reattached to existing ADAlias
- Log when successes or no matching.

AutoAliasMapping Process Flow



Below are break out of the AutoAlias Mapping process flow:

- 1) Search in CMS for all account that have only SAPAlias without WinADAlias
- 2) For each of the account identified, Search in WindowsAD and retrieve CN (AD Login id) filter by SAP ID (PERNR) . May need to trim leading zero from the SAP ID in search (BWP~200/00198454)
- 3) Once find the ID needed in AD; Look up CMS to find associated AD acct
- 4) Re-map the SAPAlias to WinADAlais account
- 5) Exception Handling – Log when job fails to identify a WinAD for exception reporting when account is not map after xx hrs – you don't want it to log if it gets pick up after next run.

AutoAliasMapping Matching Rules

Below are example of matching rules to help link users from Windows AD to an SAP ID in creating the Alias.

There are two rules:

- 1) Look up Employee Login ID (CN) from WinAD by employeeID (PERNR) by exact match if not exist search again by last 6 character with a wild card.
- 2) If ECC\BW login ID is not numeric (firstname.last) – 12 char, Search in CMS for like firstname.last% in the SI_Name field.
 - 1) If CMS query return 2+ WinAD record – throw into exception for this entry;
Example Fidencio.Pe1; Fidencio.Pe2



Customizable Matching Rules

To be coming....

Limitations and Known Issues

- 1) Content created by\for the account could be lost during time gaps between SAP accounts AutoAliasMapping - If the AutoAliasMapping program time lags and user logs in to their SAP account, any content either sent to the user's Inbox\ or save in Favorite folder will be removed.
- 2) During this time lag, objects created by the user and save to Corporate folder, Administrator will become owner of the object.
- 3) Any orphan source accounts (SAP Aliases) from disconnected system will not be created in a new system entry with same alias. For example john.s@bw100 is mapped in CMS while bw100 is no longer mapped functional. There is a new source system bw102 with john.s alias. John.s@bw100 will be removed from CMS with this program but John.s@bw102 will not be auto-generated with this process. It's important that your CMS contains only the latest functional accounts.

Deliverables

Deliverables:

- 1) Design doc**
- 2) Binary (Compiled Runtime)**
- 3) Operational Document**
- 4) Documentation**
- 5) Post\Production support**



Thank you

Contact information:

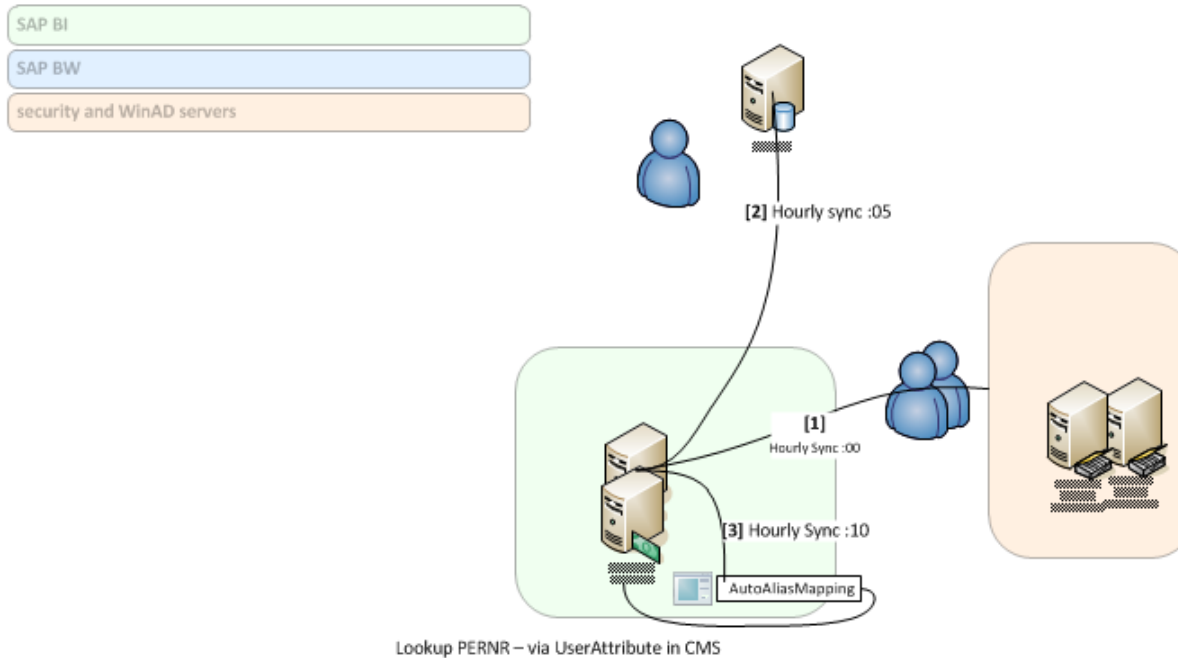
Jeremy.Ma@sap.com

Platinum Consultant

Address

Phone number

Appendix - Design Alternate (BI 4.03 required)



AutoAliasMapping:

Search for All SAPAlias in CMS that have single Alias:

- 1) Look up employeeID from UserAttribute in CMS; Search for existing ADAlias in CMS; if found delete SAPAlias; reattached to existing ADAlias
 - 2) Look up First.LastName that match first 12 char of existing ADAlias in AD; if found delete SAP Alias; reattached to existing ADAlias
- Log when successes or no matching.

Custom User Attributes (1/2)

Ability to define attributes in the CMC that will be attached to each user

These attributes can be filled manually in the CMC (Enterprise)

These attributes can be filled from a LDAP or a SAP data source

- The CMS must have been configured to authenticate with this data source
- Define the attribute from LDAP / SAP data source to provide actual values

Custom User Attributes (2/2)

New Page in the CMC to list and administer Custom User Attributes

Central Management Console

User Attribute Management

Welcome: Administrator

Name	Internal Name	Source	Sources Attribute Source Name
COUNTRY	SI_COUNTRY	Enterprise	
CITY	SI_CITY	LDAP	city

Value of Custom User Attributes for each user is displayed in the CMC, in the user properties dialog

Administrator can explicitly enter the value for Custom User Attributes defined in the CMS repository

Values retrieved from LDAP and SAP data source are displayed

Properties: annie

Hide Navigation

Properties

User Security

Member Of

Profile Values

Account Manager

Account Name: annie

Full Name: annie

Email: annie@company.com

ID, CUID: 6206 , AdfMIHjhQ05CkGVtjw5T6HQ

Description:

Additional User Properties

COUNTRY

CITY London

Database Credentials

Enable