SAP Cloud Identity Authentication (IAS/IPS) – Tenant Mapping

Published: 7/17/2020

Overview

This document provides guidance on how to use/map one Identity Authentication Services (IAS) tenant with multiple SuccessFactors tenants.

Terms

- Productive tenant - Tenant that is used for HCM where real candidates and employees of a company login and avail HCM capabilities
- Non-Productive tenant - Tenant that is not a productive tenant. Usually used for development and unit testing, QA, sandbox testing, integration testing, user training etc.
- Development tenant - This is a non-productive tenant provisioned/available in production landscape/environment only for EC customers.
  - Non-EC customers do not get this tenant.
  - This tenant should be typically used for development and unit testing.
  - There is only one development tenant in EC customer's system landscape.
- Test tenant - This is a non-productive tenant provisioned/available in either preview or production landscape/environment.
  - Every customer be it EC or non-EC gets this tenant as a part of standard SF purchase.
  - Customers can have more than one test tenant in either preview and/or production environment
- Production environment/landscape - Environment having productive as well as non-productive (dev and/or test) tenants.
  - This environment is upgraded typically 4-8 weeks after preview environment.
- Preview environment/landscape - Environment having only non-productive(test) tenants only.

Assumptions

- Customers have one dedicated productive tenant (in most cases)
- Customers may have one or more test (non-productive) tenant(s) across preview and production landscapes
- EC customers have one dev tenant (non-productive) in the production landscape

Current Tenant Distribution Model for SuccessFactors (SF) to IAS Migration

- Currently we offer 2 IAS tenants *per region* for free as part of the upgrade process
- 1 IAS tenant is PROD and is linked to the PROD SF tenant
- 1 IAS tenant is non-prod (Test) and is linked to *all* non-prod SF tenants
- While the IAS tenants may be named Test and Production, they are of the same release and the same infrastructure stack
- Current Upgrade Center Process ensures that all non-prod environment in SF (Preview, Dev and Test) are linked ONLY to the Test tenant of IAS and the same applies to PROD tenants

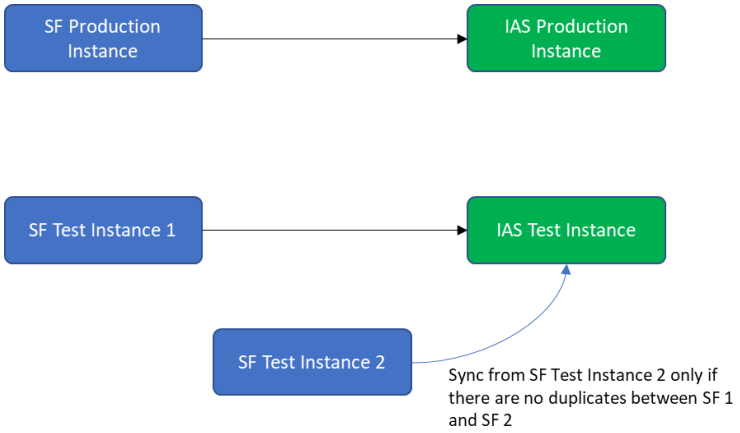
SAP's Recommendation to Customers

- For Production systems, 1-1 mapping must be followed between SF and IAS (i.e.., only one SF prod tenant can be mapped to one prod IAS tenant)
- The revised tenant model for IAS/ IPS tenants with regards to the SF bundle will be as follows (starting 2020-H2 release)
  - Production 1 IAS tenant BY DEFAULT
  - Non-Production 2 IAS tenants BY DEFAULT (Dev & Test typically)
  - In specific cases where customer requires a 1:1 mapping between non-prod SF and IAS systems, additional IAS tenants will be provided at NO Additional Cost
  - Having multiple IAS tenants is not a recommended model as it would result in reduced SSO capabilities across the organization
- For customers with multiple non-prod SuccessFactors tenants, there are 4 possible user data sync scenarios that we have outlined below:

Scenario 1:

If customers have completely different users in each of their non-prod SF tenants, they should load users from each (all) tenants into IAS using IPS.

For this scenario, the user sync is straight forward as users are unique across each tenant and therefore the customer would need to sync users from each tenant to IAS using IPS.



Here, even though the users from both instances co-exist in IAS, the access would be limited to only the systems in which they are originated. Additionally, there are simple ways to ensure that the users, while being synced from the source SF systems are identified as users from specific source systems. For example, User 1 from SF1 and User 2 from SF2 are both being synced to IAS- during the sync process, using a simple Transformation in IPS, we can put User1 in Group1 and User2 in Group2. Subsequently, using the Conditional and Risk based Authentication in IAS, we can deny access for these groups to any other SF instance. This would ensure that the access of User1 is limited only to SF1 and User2 only for SF2. An indicative transformation template would look something like below:

For the source SuccessFactors system, example Sys1

```
{
 "constant": "Sys1",
```

```
"targetPath": "$.sfinstance"
 },
```

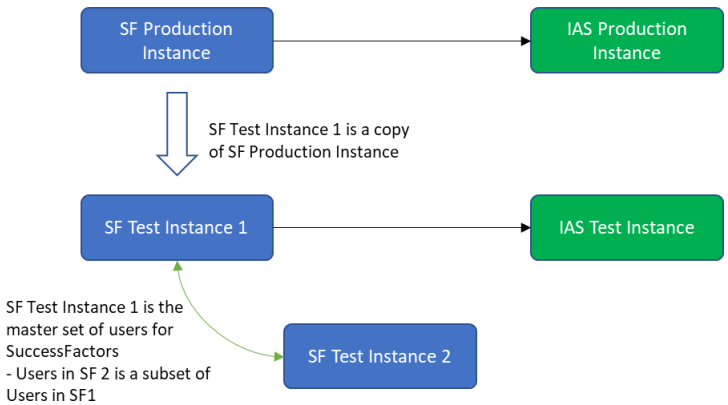
This can be added to every SF Source system that is being integrated. In the target systems, the transformations need to be changed as below:

```
{
      "constant": "Abc",
      "optional": true,
      "targetPath": "$.groups[0].value"
   },
   {
     "sourcePath": "$.sfinstance",
     "optional": true,
     "targetPath": "$.groups[0].value"
  },
```

This sets the users to be placed in Group Abc as a default. Based on the source transformations, this also sets the users from SF Instance Sys1 in the corresponding Group- Sys1. For customers wanting to assign more values, they can use a different source value and group array values. The next step would be to configure Conditional and Risk Based Authentication to deny access for Sys1 user into Sys2 (for example). Under Conditional Authentication in IAS, the customer should select IAS as the default Authenticating IdP. This set of users should now be denied access to the other instances through Risk based authentication. The customer needs to ensure that both User Access Settings *(Allow Identity Authentication users only, Apply Application Configurations)* under Identity Federation are turned ON to ensure restricted Access. Another example of this setup can be found in our Help document: Partial SSO Login using single IdP

Scenario 2:

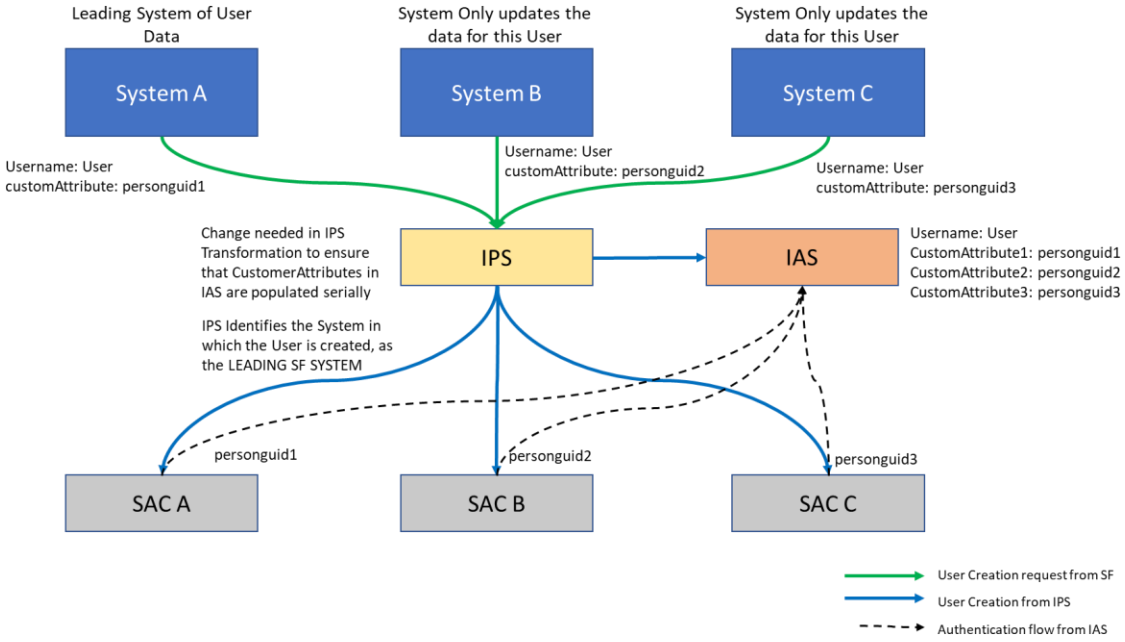
In this scenario, at least one of the customers test/dev tenants has all users and therefore should be considered the master data source for all the other SuccessFactors non-prod test/dev tenants. For this scenario, the customer should only sync data to IAS from the tenant that has all users.



Scenario 3:

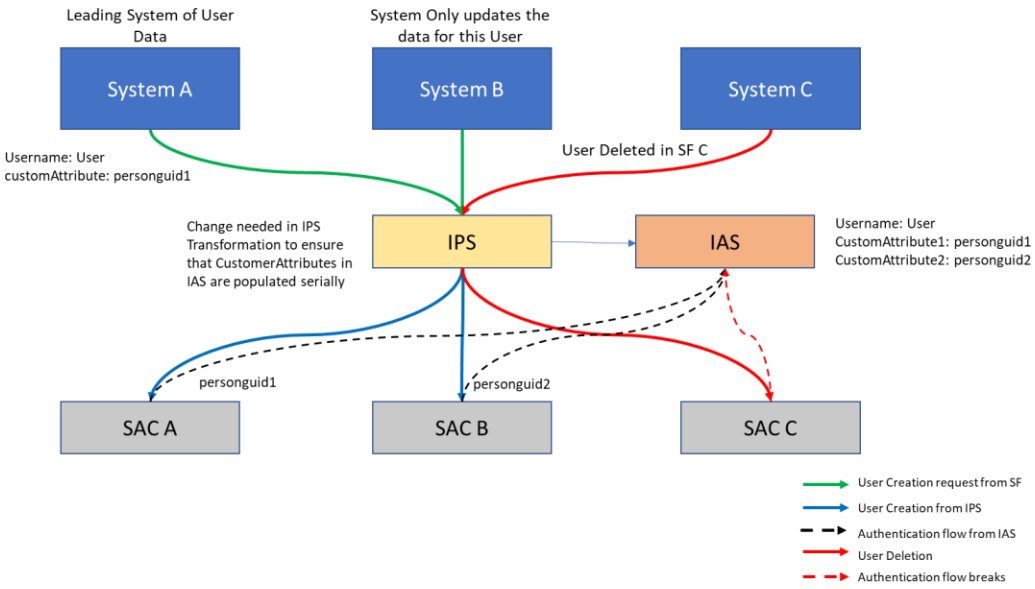
They have some users that reside in all tenants and some that do not. In this scenario, they should load only the net differences. However, this is a much more complex process as the tenants could have same users across multiple tenants of SF.

To tackle this scenario, it is important that one of the SF tenants is deemed as the Leading System of records. A Leading System of Records for IPS is defined as the SF system that CREATES the user(s) in IAS. This Leading System is the only one that can DELETE user(s). All other systems that have the same user(s), can only UPDATE user details. The flow would look something like this:

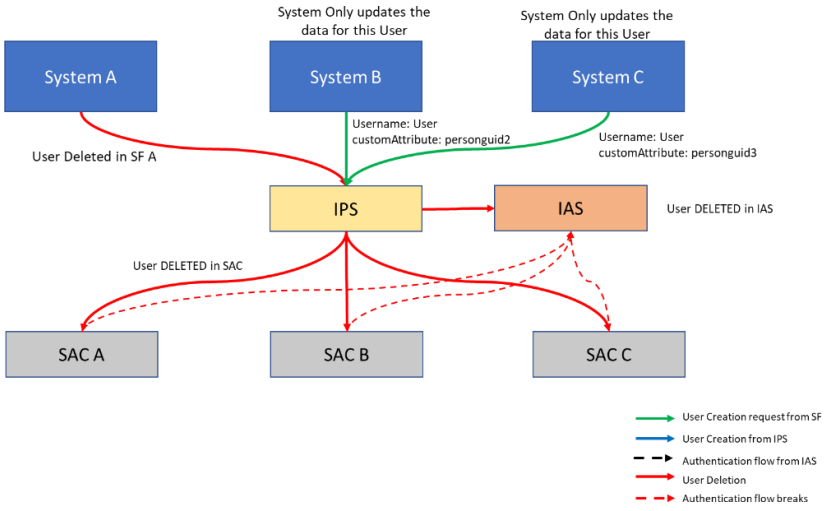


In this scenario, if a user exists in SF Tenant A and SF Tenant B (as illustrated below) and the User is CREATED in SF Tenant A, the user would still exist in IAS even if they are DELETED in SF Tenant B. Once deleted in SF Tenant A, the user is deleted in IAS.

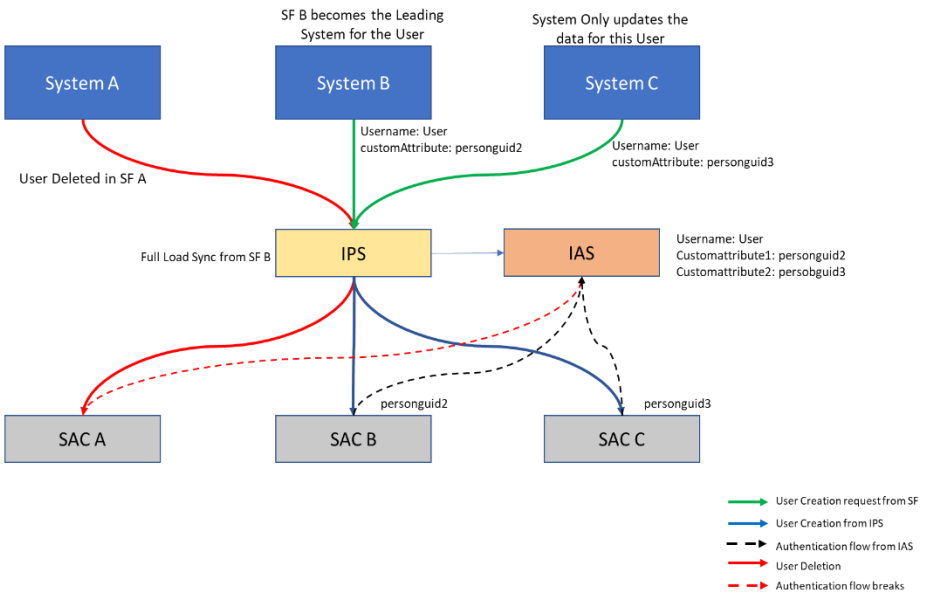
Additionally, if there are changes made to the ***username*** in the Leading System, SF Tenant A, it will update IAS and break all integrations to the other tenants. The only important attribute for SF is the username and the only important attribute for SAC is the personguid.



The recommendation is to have ONE LEADING SF TENANT for all users. If a user is deleted in the leading tenant, ie SF Tenant A, they get deleted in IAS. This means when SF Tenant B is running the IPS Scheduled user sync job, users are synced to IAS; thus, making SF Tenant B the Leading system for the user(s). In a lot of ways this is good especially for customers who do not have 1 SF tenant that acts as a master data source for all.



User Deleted in SF Tenant A (Leading System)



SF Tenant B is now the Leading System for the User

IPS Transformations to realize this setup:

The following section helps detail out the changes/ additions needed in IPS Transformations to ensure that the personGUID from SF A is set in customAttribute1 and that from SF B is set in customAttribute2 and so on.

Source system configurations:

For SF-A

```
{
  "sourcePath": "$.personKeyNav.perPersonUuid",
  "targetPath": "$['urn:sap:cloud:scim:schemas:extension:sfsf:2.0:User']['personGUID_A']"
},
```

For SF-B

```
{
  "sourcePath": "$.personKeyNav.perPersonUuid",
  "targetPath": "$['urn:sap:cloud:scim:schemas:extension:sfsf:2.0:User']['personGUID_B']"
},
```

For SF-C

```
{
  "sourcePath": "$.personKeyNav.perPersonUuid",
  "targetPath": "$['urn:sap:cloud:scim:schemas:extension:sfsf:2.0:User']['personGUID_C']"
},
```

Target system configuration for IAS

```
{
  "sourcePath": "$['urn:sap:cloud:scim:schemas:extension:sfsf:2.0:User']['personGUID_A']",
  "optional": true,
  "targetPath": "$['urn:sap:cloud:scim:schemas:extension:custom:2.0:User']['attributes'][0]['value']"
},
{
```

```
  "condition": "$['urn:sap:cloud:scim:schemas:extension:sfsf:2.0:User']['personGUID_A'] EMPTY false",
  "constant": "customAttribute1",
  "targetPath": "$['urn:sap:cloud:scim:schemas:extension:custom:2.0:User']['attributes'][0]['name']"
 },
 {
  "sourcePath": "$['urn:sap:cloud:scim:schemas:extension:sfsf:2.0:User']['personGUID_B']",
  "optional": true,
  "targetPath": "$['urn:sap:cloud:scim:schemas:extension:custom:2.0:User']['attributes'][1]['value']"
 },
 {
  "condition": "$['urn:sap:cloud:scim:schemas:extension:sfsf:2.0:User']['personGUID_B'] EMPTY false",
  "constant": "customAttribute2",
  "targetPath": "$['urn:sap:cloud:scim:schemas:extension:custom:2.0:User']['attributes'][1]['name']"
 },
 {
  "sourcePath": "$['urn:sap:cloud:scim:schemas:extension:sfsf:2.0:User']['personGUID_C']",
  "optional": true,
  "targetPath": "$['urn:sap:cloud:scim:schemas:extension:custom:2.0:User']['attributes'][1]['value']"
 },
 {
  "condition": "$['urn:sap:cloud:scim:schemas:extension:sfsf:2.0:User']['personGUID_C'] EMPTY false",
  "constant": "customAttribute3",
  "targetPath": "$['urn:sap:cloud:scim:schemas:extension:custom:2.0:User']['attributes'][1]['name']"
 },
```

Target System Configurations for SAC

- SAC for SF-A – should be configured with **customAttribute1** as NameID in the SAML assertion
- SAC for SF-B – should be configured with **customAttribute2** as NameID in the SAML assertion
- SAC for SF-C – should be configured with **customAttribute3** as NameID in the SAML assertion

For SAC-A

```
 {
     "sourcePath": "$['urn:sap:cloud:scim:schemas:extension:sfsf:2.0:User']['personGUID_A']",
     "targetPath": "$.userName"
 },
```

For SAC-B

```
 {
     "sourcePath": "$['urn:sap:cloud:scim:schemas:extension:sfsf:2.0:User']['personGUID_B']",
     "targetPath": "$.userName"
 },
```

For SAC-C

```
 {
     "sourcePath": "$['urn:sap:cloud:scim:schemas:extension:sfsf:2.0:User']['personGUID_C']",
     "targetPath": "$.userName"
 },
```

This recommendation will work fine if the data being synced to IAS is consistent across all SF Systems. But if a user has, for example, one email address in SF A and another email address in SF B, IAS will get updated with the system that was last synced.

For this, the customer will need to leverage the customAttributes in IAS to ensure that each system has a different mapping. For example, if the customer requires that user(s) have different divisions in each system, they can add “Division” to customAttributes and ensure that these are maintained separately in IAS, as detailed in the “personGUID” example above.

Scenario 4:

If the customer has the same user, e.g. “cgrant”, being used by different people in different SF Systems, there are no other options than to map the SF tenants with IAS tenants in a 1-1 fashion. While we do not offer the extra tenants (*other than the 3 in total) for free, the customer can request an additional tenant at NO EXTRA COST.

Another example of this scenario would be a case where the customer wishes to use “cgrant” as an SSO user in SF-A but wishes to use the same user as a PWD login user in SF-B. This situation would also require that both SF systems A and B connect to separate IAS tenants- a 1:1 mapping of SF-IAS tenants.

All the above guidance has been under the assumption that these non-prod SF tenants are Test/Dev tenants and hence can have the same IAS tenant to which the users are synced.

If we shift paradigms to Production systems, a 1-1 mapping between the Prod SF system and the Prod IAS tenant is the best recommended scenario.