Paper 369-2011

Using SAS® on Unix with multiple active directories as authentication providers

Jan Bigalke, Allianz Managed Operations & Services SE, Business Unit ASIC

ABSTRACT

Using SAS on Unix / Linux servers with active directories as authentication providers is now a common approach. This paper explains the basics of Unix authentication against active directories. Limitations of using standard PAM modules for authenticating users stored in multiple domains are explained. Various solutions are required to address challenges for the different access methods used by clients to connect to SAS, e.g. web access and client applications like Enterprise Guide®.

INTRODUCTION

More and more the Microsoft active directory is becoming a common solution as authentication system for UNIX and Linux based systems. Active Directory is used to simplify the authentication process for business users reducing the number of authentication systems required. Since the distribution of Windows Server 2003 R2 Microsoft ships schema additions to the active directory for UNIX authentication. These schema enhancements provide attributes that map closely enough to RFC 2307[1] to be generally usable for UNIX authentication. The standard implementations to connect UNIX systems to the active directory are nss_ldap and pam_ldap.
This paper discusses a scenario using SAS on a UNIX system with an active directory as authentication provider. The SAS users in the discussed enterprise organization are based in different active directories. The requirement for the discussed solution is that a user only has a single identity.

Workarounds to solve this task are not discussed. Some of these could be additional accounts for users in a central domain, local user accounts in SAS (@saspw), etc.

SAS ENVIRONMENT

The environment that initiates this paper is a shared SAS environment. All servers belong to one organization with an active directory as authentication provider. Subsidiary companies have their own active directory. The active directories of both companies are connected via a domain forest trust. Users from these companies should also use the shared SAS environment.

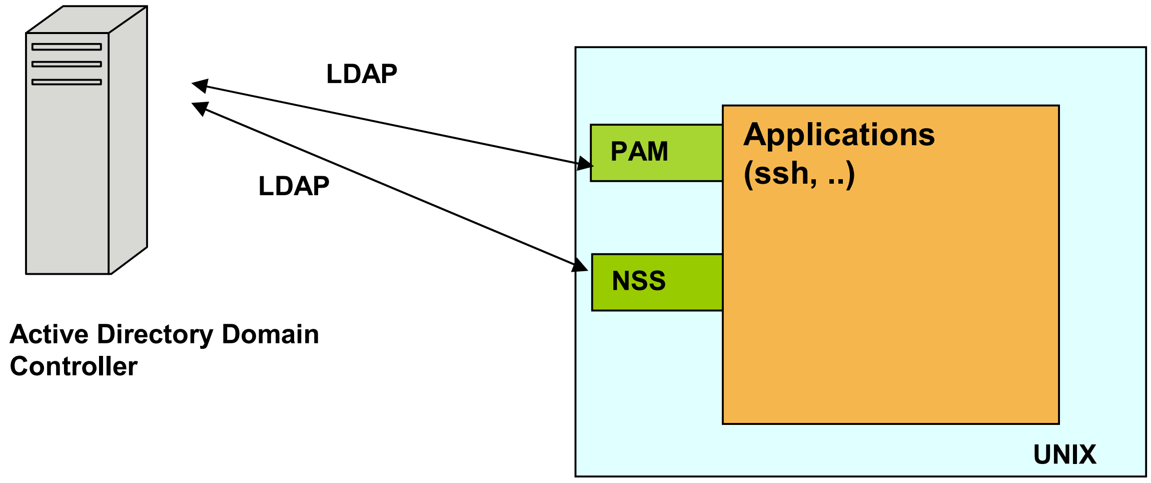
The default approach to authenticate users in a trusted domain environment is Kerberos, because Kerberos provides unique user principals over domain boarders. The user principal name (not the samAccountName) must be registered in the metadata server for the authorization process. Kerberos makes also a single sign on possible that means users don’t have to provide user credentials.

Only the web components of SAS 9.2 support Kerberos authentication on UNIX operation systems. SAS clients request a different solution. The following paragraphs will show the limitations of the default approach and offer a solution which allows users from different active directories to use SAS.

AUTHENTICATION AGAINST ACTIVE DIRECTORY

UNIX operating systems provide an abstraction layer for authentication. This layer is implemented using Pluggable Authentication Modules (PAM). These modules provide a high level application programming interface allowing applications to be written independent of the underlying authentication scheme. PAM handles the connection to the active directory for authentication and the lookup of user information is implemented by the Name Server Switch daemon (NSS).

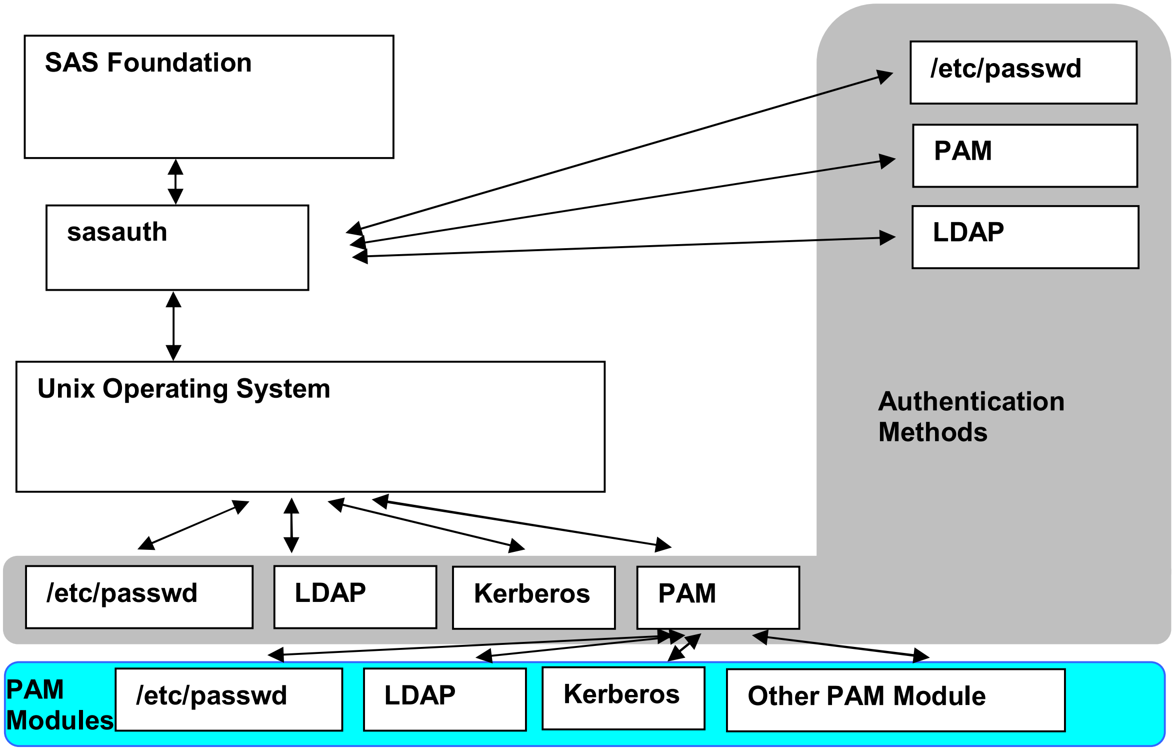
[1] http://tools.ietf.org/html/rfc2307



This requires that the SAS System has to be configured to use the PAM authentication method[2].

AUTHENTICATION IN A SAS UNIX ENVIRONMENT

In a SAS environment you can choose from several authentication methods. The default method is host based authentication. In this case the SAS System delegates the authentication to the local operating system. This can be the Password method that directly uses /etc/passwd file or the PAM method. PAM is required if the operating system also uses PAM e.g. for authentication against the active directory. Another method is a direct connection to an LDAP authentication provider.
SAS supports direct LDAP authentication. The handicap of the direct LDAP approach is that credentials will be sent in clear text. Therefore in this case it is recommended to encrypt the communication to protect these credentials.



[2] http://support.sas.com/documentation/installcenter/en/ikfdtnunxcg/61994/PDF/default/config.pdf

PAM authentication programming libraries provides only username/password combinations and do not return the UID, which is needed by SAS. To address this issue sasauth uses UNIX authentication calls to obtain the UID. As a consequence the SAS System has to be configured to use the same user information as the operating system. The configuration is done in the configuration file `!SASROOT/utilities/bin/sasauth.conf`.

In some cases it is required to configure a different authentication method for web clients to access a SAS server; especially when access is granted for external users.
In this case the authentication is done at the web application server and an authenticated user is delegated to the SAS system. Documentation describing this method for supported web application servers is available on support.sas.com[3].

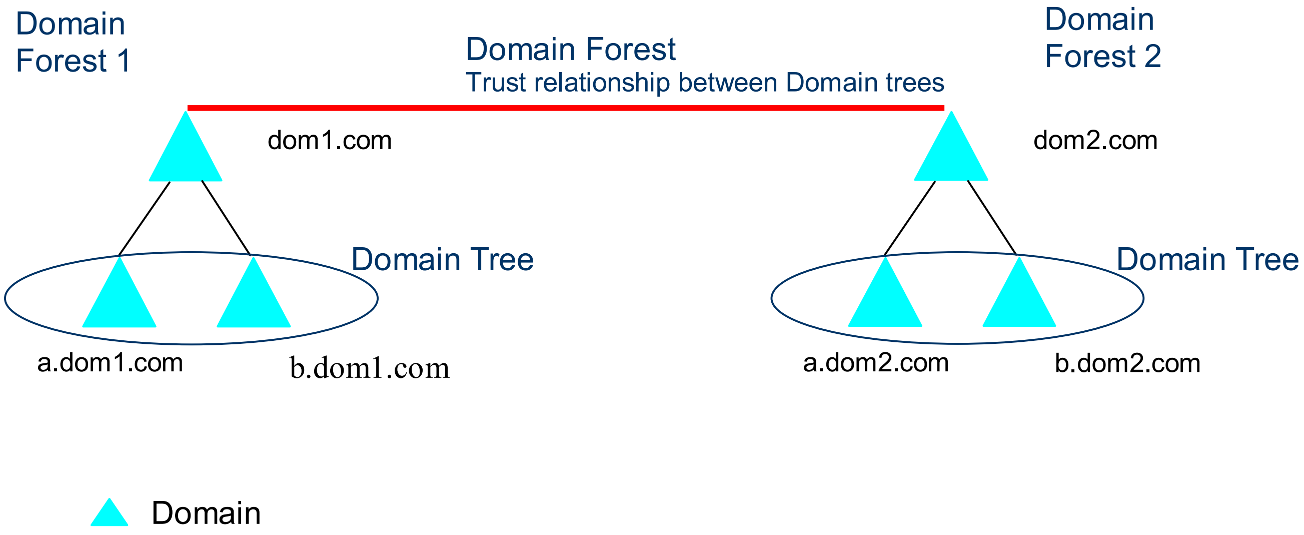
In this case the web application server is configured to access a user registry such as an LDAP Server.

BASIC INFORMATION ABOUT ACTIVE DIRECTORY RELATIONSHIPS

Microsoft Active Directory administers objects. An object could be any user, system, computer, resource, etc within the active directory. The active directory organizes the objects in a hierarchy. There are two main categories: resources (computers,..) and security principals (users, groups,..).

A domain represents objects that are in a common database. The domain contains only objects that belong to this domain. A single domain or multiple domains in a contiguous namespace are named a “tree”. A collection of trees is named a “forest”. The forest is the outside boundary for objects (user, computers, groups).

The illustration below shows the domain forest 1 and the domain forest 2. Each forest contains a domain tree, containing two domains. In this example domains in the southern region could be grouped in a domain tree. The domain tree is integrated in the forest. The forest is the outside boundary for the objects existing in the respective domains. To overcome this border between forests trust relationships between forests are required.



Trusts inside a forest are automatically created during the process of creating a new domain in this forest. Trust relationships between forests have to be created manually.

These trusts are necessary to allow users from a domain in forest 1 to use resources in a domain of forest 2.

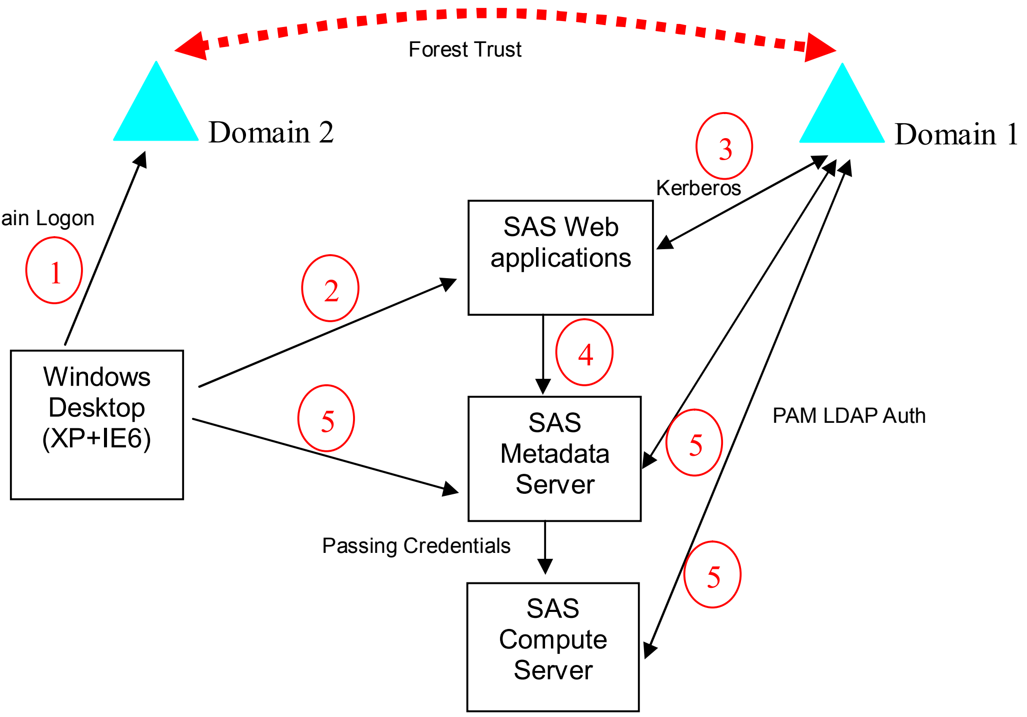
To explain the business case lets make an example. We have two companies with active directories merging into a new company . As you know joining different active directories into one big directory is a time consuming process. During the merger there is a need for resource sharing between the users from domain dom1.com and dom2.com. This can be accomplished with a trust joining the two forests into a domain forest.

[3] http://support.sas.com/resources/thirdpartysupport/v92/appservers/index.html
http://support.sas.com/resources/thirdpartysupport/v92/appservers/ConfiguringJBossWebAuth.pdf
http://support.sas.com/resources/thirdpartysupport/v92/appservers/IWAWebSphere.pdf

SAS IN A TRUSTED DOMAIN ENVIRONMENT

This chapter describes a SAS multiple tier installation in an environment where users of SAS are based in different active directories. The discussed environment uses SAS Enterprise BI Server. This package contains web components, managing and computing units. The SAS installation uses domain 1 from forest 1 as authentication provider. The users in domain 2 should also use this installation. Business users connect to the SAS system with desktop computers or laptops running a flavor of Windows..

The following picture illustrates the environment. First we have a look at the authentication needs in a SAS environment.



1. The client issues a domain logon on his workplace computer to domain 2
2. The client accesses the SAS web application (e.g. SAS Portal).
3. The web application refers the request to the Logon Manager. The Logon Manager connects to the active directory and authenticates the user using the appropriate JAAS module of the Java Web application Server.
4. The authenticated user is delegated to the metadata server.
5. The client tools like Enterprise Guide® and SAS® Data Integration Studio connect directly to SAS metadata server and subsequently start tasks on the SAS compute server after authentication has been successful.

Step one is done in every scenario. After being logged on to the windows desktop the user is authenticated to his domain. This is the starting point for all possible methods to access the SAS system in this environment.

The following chapter describes the different access methods to connect to SAS system in detail.

ACCESS FOR SAS CLIENTS IN A SCENARIO WITH MULTIPLE ACTIVE DIRECTORIES

During a login process using a SAS Client [4] with server components on UNIX, the SAS user is prompted for credentials. These credentials are used to authenticate against the authentication provider of the SAS Metadata Server. (Details described in Chapter 11 in SAS Security administration guide).

Different possible authentication methods on Unix systems have already been described. Limitations of pam_ldap standard and direct ldap allow only a single LDAP configuration. This is the problem that has to be solved to use SAS in a multi domain environment. The same requirements have to be met during the following authentication steps at the compute server. For example if the object spawner creates a SAS Workspace Session for the use of the EG or other

[4] http://support.sas.com/documentation/cdl/en/bisecag/61133/PDF/default/bisecag.pdf

SAS products using SAS sessions in the security context of the user currently logged in.

A routing like Kerberos is not possible, because the realm of the source domain of the user is not transported with the authentication process. On the SAS side there is only the possibility to implement a proprietary authentication module (described in `!SASROOT/utilities/src/auth/docs.pdf`). The other touch point is the operating system itself.

The authentication capabilities of the operating system can be enhanced with third party software. For example the Quest® Authentication Services[5] can bypass the limitations described above. The Quest approach allows the configuration of user and group search paths. To enable cross forest authentication in Quest using simple names it is necessary to list any domains in foreign forests using the cross-forest-domains option. (vas.conf man page) The details to this approach are described in Quest AuthenticationServices_4.0_AdminGuide. There might be some products on the market that have the same opportunities.

The installation of an additional software product doesn't solve all issues. There are some challenges left on the organizational side. A UNIX system account has a UID and groups on the system have GUIDs. These attributes are stored in the active directory, if used as authentication provider. In this case, the UID comes from the active directory where the user is based. That means in a trusted domain environment the UID and GUID has to be unique across the boundaries of the domains. If you can't meet this requirement there is a risk of users sharing the same UID and being able to access the data of another user with the same UID defined in a different active directory. This requires the definition of a process ensuring that UIDs are unique throughout enterprise.

ACCESS TO SAS WEB COMPONENTS USING WEB AUTHENTICATION

There are two possible configurations for SAS web components[6]. The default method is "SAS authentication". In this case the authentication is done using the authentication provider defined for the SAS metadata server. The alternative method is "web authentication". In this case users are authenticated by the J2EE web application server via JAAS (Java Authentication and Authorization Service[7]) using an alternate authentication provider.

The SAS web application delegates SAS users to the SAS Logon Manager if this user has not already been authenticated. SAS needs unique credentials to authenticate the user from the Login Manager. In an active directory environment it's the user principal. Some organizations use the SAM-Account-Name as login name. This attribute is not unique in a domain forest; therefore SAS may in some cases not be able to authenticate the user.

For web access "Integrated Windows Authentication" is possible using SAS 9.2 for all operating systems[8]. This approach uses Kerberos for authentication. The Kerberos protocol is designed to operate across organization boundaries. The different domains in our scenario describe different "realms" in the Kerberos namespace. The name of the realm in which the client is registered is part of the client's name and can be used for the authentication process[9]. If active directories are connected with trusts Kerberos allows authentication across the boundaries of the domain. The process is described in detail in a document[10] from Microsoft (see paragraph **Kerberos Authentication Process Over Forest Trusts**).

The Kerberos authentication allows a single-sign-on and is additionally able to cross the domain boundary. In SAS 9.2 this is supported for the web tier only if you are using non-Windows SAS servers.

For Windows-only SAS environments, Kerberos(IWA) is also supported for authentication SAS clients.

CONCLUSION

Using SAS on UNIX with multiple domains as authentication providers is not an easy project. For a working configuration of SAS 9.2 in such environments a basic understanding of several technologies is required. First the authentication of UNIX against active directories with PAM. Second the possibilities of the SAS configuration for authentication. For multiple domain environments a basic understanding of Windows domains and trusts is also necessary. For the web configuration with Kerberos you need the knowledge and skills described in the paper from Heesun Park of last years SAS Global forum (Integrated Windows Authentication Support for SAS® 9.2 Enterprise BI Web Applications)[11].
For the other access requirements you need to enhance the capabilities of the operating system. The paper shows the Quest approach. This approach allows sharing scalable SAS infrastructures on UNIX among users based in different active directories throughout international organizations or enterprises

[5] http://www.quest.com/authentication-services/
[6] http://support.sas.com/documentation/cdl/en/bisecag/61133/PDF/default/bisecag.pdf (page 157)
[7] http://www.ibm.com/developerworks/java/library/j-pj2ee9.html
[8] http://support.sas.com/resources/papers/proceedings10/312-2010.pdf
[9] http://www.ietf.org/rfc/rfc4120.txt
[10] http://technet.microsoft.com/en-us/library/cc773178(WS.10).aspx
[11] http://support.sas.com/resources/papers/proceedings10/312-2010.pdf

REFERENCES

Configuration Guide for SAS® 9.2 Foundation for UNIX® Environments
http://support.sas.com/documentation/installcenter/en/ikfdtnunxcg/61994/PDF/default/config.pdf

SAS 9.2 Support for Web Application Servers and HTTP Servers
http://support.sas.com/resources/thirdpartysupport/v92/appservers/index.html

Configuring Application Server for Web Authentication with SAS 9.2 Web Applications
http://support.sas.com/resources/thirdpartysupport/v92/appservers/ConfiguringJBossWebAuth.pdf
http://support.sas.com/resources/thirdpartysupport/v92/appservers/IWAWebSphere.pdf

SAS® 9.2 Intelligence Platform Security Administration Guide
http://support.sas.com/documentation/cdl/en/bisecag/61133/PDF/default/bisecag.pdf

Quest Authentication Services
http://www.quest.com/authentication-services/

Java security with JAAS
http://www.ibm.com/developerworks/java/library/j-pj2ee9.html

Integrated Windows Authentication Support for SAS® 9.2 Enterprise BI Web Applications
http://support.sas.com/resources/papers/proceedings10/312-2010.pdf

The Kerberos Network Authentication Service
http://www.ietf.org/rfc/rfc4120.txt
http://technet.microsoft.com/en-us/library/cc773178(WS.10).aspx

CONTACT INFORMATION

Your comments and questions are valued and encouraged. Contact the author at:

Name: Jan Bigalke
Allianz Managed Operations & Services SE, Business Unit ASIC
jan.bigalke@allianz.com