CentraSite

Authentication Topics and LDAP

Version 9.6

April 2014

This document applies to CentraSite Version 9.6.

Specifications contained herein are subject to change and these changes will be reported in subsequent release notes or new editions.

Document ID: IINM-AG-AUTHENT-96-20140318

Table of Contents

Preface

Authentication is the process of validating that a user's login credentials (for example the user's certificate, or user ID and password) match the credentials known to the system. CentraSite can use a number of different data sources, known as domains, to validate a user's credentials; these currently include the following:

- an *internal* text file;
- Microsoft Active Directory (AD), when used via LDAP;
- LDAP.

This document is intended for customers who wish to configure CentraSite's user authentication features.

Assumptions

If an external repository, for example LDAP, will be used, this document assumes that it has already been set up and that you have the necessary expertise and privileges to perform administrative tasks. Usually, the use of CentraSite does not influence any design decisions that were made in setting up an external user repository; CentraSite just needs to know how to access the users and groups of the user repository.

The content is organized under the following sections:

Overview of User Repositories	This section gives a general overview of how CentraSite makes use of external user repositories for performing authentication tasks.
Creating Authentication Configurations	This section describes how to create authentication configurations.
Configuring the "Internal" Authentication Type	This section describes how to set up a user repository of type "Internal" for CentraSite.
Configuring LDAP	This section describes how to set up your LDAP directory as the user repository for CentraSite.
Performing Maintenance on Authentication Configurations	This section describes how to maintain authentication configurations.
Creating an Administration User from the Command Line	This section describes how to create a CentraSite user who has the "CentraSite Administrator" role.
Logging of Login Authentication Messages	This section describes the authentication logging features.
Notes on Authentication in CentraSite	This section contains general information related to authentication in CentraSite.

1 Overview of User Repositories

A user repository is in general terms a set of user credentials (optionally including user certificates etc.), with the possible addition of information such as the groups to which a user belongs, the user's address, telephone number and the email address. Often, an enterprise implements a central user repository that can be used by applications throughout a network to authenticate users; when a user tries to log in to an application, the application issues a request to the user repository to check whether the user credentials that she supplied are valid. Usually the user repository is created and maintained separately from the applications that use it.

A newly-installed CentraSite system is configured to authenticate users against an internal text file. This is intended to enable an administrator to log in and modify the configuration as required to meet enterprise requirements; typically, and in particular if you are working in a distributed environment, where one or more Application Server Tiers and a separate Registry/Repository are involved, an external repository such as Active Directory or LDAP will form the core of the authentication process.

The following sections describe how you can configure CentraSite for use with user repositories.

Selecting a User Repository for Authentication

Access to information stored in CentraSite generally requires a user name and password, to ensure that data can only be stored, modified or retrieved by authorized users. CentraSite supports the following types of user repository:

- an internal text file;
- LDAP (for example Sun, OpenLDAP, ADS).

CentraSite maintains information about each kind of user repository in so-called *authentication configurations*. An authentication configuration specifies the type of user repository to be used and any parameters that are required to configure the user repository. CentraSite is delivered with one predefined authentication configuration, namely the configuration to use an internal text file, and this configuration is the default configuration. You can define additional authentication configurations; also, you can set any one of the defined configurations to be the default configuration.

See the section **Creating Authentication Configurations** for information about defining authentication configurations and setting the default authentication configuration.

In general, user authentication information is stored in the user repository, not in CentraSite. CentraSite can contain a copy of selected data fields from the user repository for each registered CentraSite user. The user information in the CentraSite user registry is stored in objects of the type "User". You can associate a CentraSite user object with a user in a user repository (see the “Associated with” feature in the user administration of CentraSite Control). In this case you can map data fields from the user repository into the user object in the CentraSite registry. The data in the mapped data fields is visible when you display the user object in CentraSite.

Domain Names of User Repositories

Each user repository is uniquely identified by a domain name. A user in a user repository is uniquely identified by the combination of domain name and user name.

When you log in to CentraSite Control, you must supply the name of a domain in which you are registered and your user name, in the format "*<DomainName>**<UserName>*", for example, "Headquarters\JSmith".

The domain name for an authentication configuration of type Internal is always "INTERNAL". Since this name is fixed, there can be only one such configuration defined per instance of the CentraSite registry.

The Default User Repository

While CentraSite is running, there is always exactly one default user repository. When you install CentraSite, the default user repository is set to the internal text file. You can change the default to any other user repository for which an authentication configuration exists.

Users who are registered in the default user repository can omit the domain name when they log in. For example, if the domain "Headquarters" is the default domain and it contains a user whose user name is "JSmith", then this user can log in as "JSmith" instead of "Headquarters\JSmith". Users who are not registered in the default user repository must always use the format "*<DomainName>**<UserName>*" to log in.

2 Creating Authentication Configurations

When CentraSite utilizes a user repository, certain connection parameters are required. The connection parameters are stored in an authentication configuration. If you need to work with more than one user repository (for example, a user repository for test purposes and a user repository for a production environment), you can define several authentication configurations.

At any given time, only one authentication configuration can be the default authentication configuration.

You can create the authentication configurations as described in the following sections.

After creating or modifying the authentication settings, the new settings apply immediately to the CentraSite Registry/Repository.

Commands for Creating and Maintaining Authentication Configurations

Commands for creating and maintaining authentication configurations are available with the command line tool CentraSiteCommand. You can use this tool to perform the following tasks:

- Create an authentication configuration
- Modify an authentication configuration
- Delete an authentication configuration
- Set a default authentication configuration
- List the names of all defined authentication configurations
- List details of a specific authentication configurations
- Validate that an authentication configuration is correctly specified

For information on how to perform these tasks, see the section *Configuring the Authentication Settings* in the document *Basic Operations*.

Specifying the Domain Name

Unless the type of authentication that you wish to specify is "INTERNAL", the authentication configuration requires a domain name. This is the domain name that will be used to address the users who are authenticated against the specified user repository.

⚠ **Important:** When working with LDAP, the domain name should be the name of a specific domain controller (DC) node in the LDAP tree structure. There can be many DC nodes in an LDAP tree structure, and you must choose the DC node that is the deepest ancestor node (parent, grandparent etc.) of all of the user nodes. Here, “deepest” means furthest away from the LDAP tree's root node. For example, if the usernames in an LDAP tree structure are located in the LDAP path `uid=Username,ou=People,dc=mydomain,dc=com`, then both `dc=mydomain` and `dc=com` are ancestor DC nodes of the user nodes, but since `dc=mydomain` is deeper than `dc=com`, you should specify the domain name as "MYDOMAIN" and not "COM". If the path to the user nodes does not include any DC nodes, specify the root node. For example, if a user's full path is `cn=Username,ou=People,ou=RnD,o=Company`, set the domain name to "Company".

The domain name for an authentication configuration of type Internal is always "INTERNAL". Since this name is fixed, there can be only one such configuration defined per instance of the CentraSite registry.

Mapping User and Group Fields

When you specify an authentication configuration, you specify the correlation between properties stored in the CentraSite JAXR-based model for the object type *User* and properties stored in the external user repository.

The JAXR-based properties stored in CentraSite for the object type *User* are organized according to the following structure:

```
description
organization
personName
    firstName
    middleName
    lastName
    fullName
postalAddresses
    postalAddress
        street
        streetNumber
        postalCode
        city
        stateOrProvince
        country
        postalScheme
emailAddresses
    emailAddress
        address
telephoneNumbers
    telephoneNumber
        countryCode
        areaCode
        number
        extension
        url
URL
```

The mappings are used in CentraSite Control when you create a new CentraSite user and wish to associate the user with a user in the external user repository and also when you click on **Synchronize** for a user in the CentraSite Control. The corresponding dialog in CentraSite Control for locating the external user definition includes a search capability in which you can specify the JAXR-based mapping properties mentioned above to locate a particular user. The search mechanism translates the JAXR-based property searches into corresponding searches of the properties of the external user repository, using the mappings you define here. See the section *Adding a User* in the document *Users, Groups, Roles and Permissions* for further information.

Specify the mappings as required. Typically, you only specify mappings for properties that you wish to make available for searches of the external user repository. If you do not require the capability of searching the external user repository, you can leave all of the fields empty.

Note: User property mappings are not available for users who are stored in the internal repository.

3 Configuring the Internal Authentication Type

General

The Internal authentication type allows you to authenticate a user against a set of user names and passwords that are maintained in a text file on the CentraSite Registry/Repository. Passwords are stored in SHA-512 hashed format; they cannot be decrypted. All user names and passwords are case-sensitive.

A typical use of such an authentication type would be during the initial set-up and testing of all required CentraSite components. In a production environment, one would typically use a central repository, e.g. Microsoft Active Directory or LDAP, instead of Internal authentication.

The domain name for the Internal authentication type is always "INTERNAL"; this cannot be changed. A user who is registered in the text file can log in using the domain and name "INTERNAL\<UserName>", where <UserName> is the registered user name.

The Internal user repository initially contains one predefined user named "Administrator" with the password "manage". This user logs in using the domain and user name "INTERNAL\Administrator". If your default authentication configuration is the Internal configuration, this user can log in using just the user name "Administrator", without specifying the domain name explicitly.

Caution: As soon as possible after completing installation, you should change the password that is associated with the user "Administrator"!

The dialog for creating a configuration for Internal authentication (see the section **Creating Authentication Configurations**) asks for the following values:

Parameter	Description
Domain ID	The domain ID is always "INTERNAL". This cannot be changed.
Expiration	The number of seconds that the user is cached in the server after successful authentication. Changes made to the user, e.g. deletion or password changes, do not take effect until this time has elapsed.

Administration of Users and Passwords for Internal Authentication

CentraSite provides a command line tool *ssxtxtpasswd* to allow you to perform administration tasks on the internal authentication file, such as adding users, deleting users and changing passwords.

The command line tool is located at <SuiteInstallDir>*\common\security\ssx_32\bin\ssxtxtpasswd.exe* (Microsoft Windows 32-bit systems) or <SuiteInstallDir>*\common\security\ssx_64\bin\ssxtxtpasswd.exe* (Microsoft Windows 64-bit systems) or <SuiteInstallDir>*/common/security/ssx/bin/ssxtxtpasswd* (UNIX).

The text file is located at <*SuiteInstallDir*>*/common/conf/users.txt* (Microsoft Windows and UNIX).

The usage of the command line tool is as follows:

```
ssxtxtpasswd [-f filename] [-c] [-p password] [-d] UserId
```

The following example prompts the user to enter the password. The password is not echoed as is it typed in.

```
ssxtxtpasswd -f <SuiteInstallDir>/common/conf/users.txt <UserID>
```

If the user name already exists, the password will be replaced. If the user name does not already exist, a new entry with the given user name and password will be added.

The parameter `-c` creates the file for the repository, if it does not already exist. If `-c` is omitted and the file does not exist, an error is returned. The `-c` and `-d` parameters are mutually exclusive.

With "-p password", a password can be provided directly in the command line. This is mainly for scripting purposes, and prevents the tool from prompting for the password.

Caution: Be careful when writing scripts that use the ssxtxtpasswd `-p` parameter. Any script that contains a password in plaintext is a potential security hazard.

When "-d" is specified, the user name is removed from the repository. The `-c` and `-d` parameters are mutually exclusive.

4 Configuring Active Directory Server

Note: Using an Active Directory Server as the user repository is only supported via LDAP.

The dialog for Active Directory Server (see the section **Creating Authentication Configurations**) asks for the following values:

Parameter	Description
Domain ID	(As described in the section **Specifying the Domain ID**)
Host	The name of the machine on which the Active Directory server is running.
Port	The port on which the Active Directory server is running. Only needed if it is not the default value, which is 389.
Forest DN	The base bind distinguished name for the node under which all the domains reside. Example: if there are domains like: dc=HR,dc=abc,dc=com and dc=USA,dc=abc,dc=com and dc=EUR,dc=abc,dc=com, then this parameter would be "dc=abc,dc=com".
Expiration	The number of seconds that the user is cached in the server after successful authentication. Changes made to the user, e.g. deletion or password changes, do not take effect until this time has elapsed.

5 Configuring LDAP

Principles of Configuring against LDAP

CentraSite supports various LDAP configurations and provides standard settings that allow you to set up your authentication quickly against these standard systems.

There are many questions that are involved when you configure against an LDAP system:

- What kind of LDAP server is it?
- What is the hierarchical node structure of the LDAP server?
- In which kinds of objects are the user and group definitions contained?
- Which node properties contain the user names or group IDs?
- What other property mappings are required?

In general, before you begin to specify the configuration, we recommend you to study the LDAP structure and contents using an LDAP browser. There are various freeware tools such as JXplorer (**http://jxplorer.org/**) that allow you to do this. Using the LDAP browser, you can bind to an LDAP server, then navigate through the hierarchy to see the structures that contains the users and groups. Also, you can open the nodes that contain the definitions of individual users or groups, and view the properties that are stored for each user or group. An example of a node for a user `testuser01` might show the following properties:

Property name	Value
cn	testuser01
objectClass	OpenLDAPperson
Mail	JohnSmith@MyCompany.com
Phone	+1 234 555 678

The path to the node for this user might be *com/People/Location3/testuser01*, where *com* is the root node. The setup on this LDAP server might be that all users are stored under the *People* node (*com/People/…*) and all groups are stored under the *Groups* node (*com/Groups/…*). Since every CentraSite customer can define their LDAP user and group structures differently, the details of the LDAP configuration that you will perform in CentraSite vary accordingly, since you must map explicitly to the customer LDAP structures.

Performing the LDAP Configuration

The general values that you can specify for an LDAP configuration (see the section **Creating Authentication Configurations** above) are described in the following table.

Value	Description
Domain ID	(As described in the section **Specifying the Domain ID**)
LDAP server (host:port)	This is the host name (server and domain) of the machine where the LDAP server is located. You can specify a `Host:Port` combination in this field, where `Port` is the port number of the LDAP server on the host machine. You can specify multiple hosts in this field, using the blank character as a separator, for example `Host:Port Host:Port ...` If you specify multiple hosts, they are tried in the given order until a connection can be established. Each host can also be specified with a scheme such as "ldap" or "ldaps", using the syntax `ldap://Host:Port` or `ldaps://Host:Port`.
Server Type	This field allows you to specify the type of LDAP server that will be used. You can specify Active Directory as the server type if the Active Directory server is accessed via LDAP (e.g. from a UNIX system).
Caching time for user credentials	The number of seconds that the user is cached in the server after successful authentication. Changes made to the user, e.g. deletion or password changes, do not take effect until this time has elapsed. This setting is provided for performance reasons. The default value is 120 seconds. If the connection to the LDAP server is slow, you can increase this figure.

The user-specific settings that you can specify are the standard LDAP settings. Refer to the documentation of your LDAP system supplier for details. Here are some examples.

User-specific Value	Description	Example
DN	The directory tree part of the distinguished name (standard LDAP terminology) of the entry. The method of specifying the path uses the standard LDAP path convention: first, a unique property of the DN node is specified, along with the property's value. Usually the property "ou" (organizational unit) is the property chosen	ou=people,dc=MyServer,dc=com This example identifies the node whose "ou" property has the value "people" and is located under the node whose "dc" property is "MyServer", which in

User-specific Value	Description	Example
	for this purpose. Then the next higher "dc" node (i.e. a node with a "dc" property), then the next higher "dc" node and so on, until finally the root node.	turn is located under the node whose "dc" property has the value "com".
Object	This identifies a property value that is used to categorize nodes as user nodes. For example, if you specify "OpenLDAPperson", this means that user nodes can be recognized by being of object class "OpenLDAPperson".	inetOrgPerson
Group Attribute	If the user repository specifies a property linking users to the groups to which they belong, specify the name of the property here. If there is no such property, leave this field blank.	memberOf
Field	This is the name of the property in the user node that uniquely identifies the user. (The attribute name of he RDN of users.)	cn

The group-specific settings that you can specify are the standard LDAP settings. Refer to the documentation of your LDAP system supplier for details. Here are some examples.

Group-specific Value	Description	Example
DN	This is similar to the DN property for users, as described above, but identifies a DN node for groups rather than for users.	ou=Groups,dc=abc,dc=de
Object	This identifies a property value that is used to categorize nodes as group nodes. For example, if you specify "groupOfNames", this means that group nodes can be recognized by being of object class "groupOfNames".	groupOfUniqueNames
User Attribute	If the user repository specifies a property linking a group to the users who are members of the group, specify the name of the property here. If there is no such property, leave this field blank.	member
Resolution	This specifies whether group nodes contain links to the users who are members of the group, or whether user nodes contain links to the groups they belong to. The option "recurse down" means that group nodes contain links to users. The option "recurse up" means that user nodes contain links to groups.	Recurse Up

Note: If you are using LDAP, note that only the "recurse up" option is supported for group resolution.

Technical Principal for LDAP

Background

CentraSite can only find and authenticate a user name via the LDAP mechanism if either:

- the user name is located directly beneath the LDAP node that represents all users (specified via the User DN configuration value – for example, if user names are in the form `uid=Username,ou=People,dc=mydomain,dc=com` then the user name must be beneath the node `ou=People,dc=mydomain,dc=com`), or:
- the LDAP server allows "anonymous bind".

The technical principal is a user name or user account that preferably should not belong to a real user; in other words, the technical principal is normally the ID of a fictitious user. It is intended for organizations that store their user entries in branched LDAP directory structures, for example `uid=Username,loc=Germany,ou=People,dc=mydomain,dc=com` but do not allow anonymous bind. The technical principal must be defined in LDAP as having (at least) read access to all users and groups that are to be used by CentraSite.

When CentraSite is configured to use this feature, *all* LDAP accesses take place using the technical principal. For example, if a user with user name "user1" and password "pwd1" wants to log in to CentraSite Control, LDAP is accessed using the technical principal and the record for the user "user1" is checked.

Creating a Credentials File

CentraSite provides a simple command-line tool to generate the credentials file for a given technical principal. You will need this credentials file in the next step, when you set up CentraSite to use the technical principal for authenticating user name.

Note: Before using this tool on UNIX systems, please ensure that the binary file *createTechUserCreds* has execute permission set; also, please set up the CentraSite environment by sourcing the file *centrasite_setenv.sh* (for example, under *bsh* or *bash*: `. ./centrasite_setenv.sh`).

The tool is invoked as follows (note that the UNIX version of the program is named *createTechUserCreds*):

```
  <Suite-Install-Dir>\common\security\<architecture>\bin\createTechUserCreds.exe ↩
[-f outputfile] [-k keyfile] [-p password] userId
```

where:

-f outputfile

specifies the name of the output file, i.e. the file into which the tool will write the generated credentials.

-k keyfile

specifies the name of the file that contains the key used for encryption and decryption of the password.

The file should consist of a single line of 64 hexadecimal characters, i.e. each character is in the range [0-9],[a-f] (if it is longer than 64 characters, the excess characters are ignored). As usual, each pair of hexadecimal characters denotes one byte. The first 16 bytes are used as the AES encryption key; the next 16 bytes are used as the AES initialization vector.

If this parameter is omitted, the system uses a default key.

-p password

specifies the password for the given technical principal.

> **Caution:** This parameter is provided for use in batch scripts. The password is specified “in clear text”, i.e. unencrypted, and the batch script therefore presents a potential security risk. Take care to control access to any batch script that uses this parameter. Wherever possible, you should omit this parameter and enter the password interactively instead.

userId

is the user name that will be associated with the generated credentials. Use the full path to the user, for example `cn=techprincipal,ou=services,o=bigcompany`.

Unless you specified the `-p` parameter, after entering the command you will be prompted to enter the password (it will not be echoed visibly). The user name and the encrypted password are then stored in the credentials file in the location that you specified.

Example of Configuring LDAP Authentication

This section contains an example of setting up LDAP Authentication using the command line tool *CentraSiteCommand*, which is located in *C:\SoftwareAG\CentraSite\utilities*.

The command to start the command line tool is as follows. The example assumes that there is a user "AdminUser" who has the CentraSite Administrator role, and this user has the password "AdminPass".

```
cd C:\SoftwareAG\CentraSite\utilities
CentraSiteCommand.cmd set Authentication -user AdminUser -password AdminPass
```

The sample interactive dialog is as follows. The input values are shown in **bold type**.

```
Executing the command : set Authentication

============================================================
Step 1: Basic LDAP Host Information
------------------------------------------------------------


LDAP Server (ldap(s)://host:port):ldap://MyServer01:10388


1 = Active Directory

2 = OpenLDAP

3 = Sun ONE Directory

4 = IBM Tivoli

5 = Novell eDirectory

6 = Apache Directory

0 = other

server type:6

caching time for user credentials (Default 120 seconds):120

Do you want to use the LDAP Technical User (Y/N) [N]:Y

Provide the LDAP Technical User credentials file.
techLdapUserCredFile: c:\credentials\admin.txt

Provide the LDAP Technical User Key file
techLdapUserKeyFile: c:\SoftwareAG\common\security\ssx_32\etc\alt_keyfile.txt

Trying to connect to LDAP at: ldap://MyServer01:10388
============================================================
Check 1: LDAP Host   >> PLEASE WAIT...
------------------------------------------------------------

[OK] LDAP server found.

Repeat configuration step 1, Continue, or End? (R/C/E) [C]:

============================================================
Step 2: Basic User Information
```

```
-----------------------------------------------------------

userid field [cn]:cn

object class [top,person,organizationalperson,inetorgperson]:inetOrgPerson

Specify the root node (DN) for all groups.

user base DN:ou=people,ou=abc,o=sag

Specify the domain ID

domain:MyDomain


To test the connection, please provide the credentials of a valid LDAP user.

userid:alice0

password:

===========================================================
Check 2: LDAP User   >> PLEASE WAIT...
-----------------------------------------------------------

[OK]  User authenticated successfully


Do you want to see the LDAP search trace? (Y/N) [N]:

Repeat configuration step 2, Continue, or End? (R/C/E) [C]:

===========================================================
Step 3: User Properties Mapping
-----------------------------------------------------------
Default Mapping:telephoneNumbers:telephoneNumber:number >> telephoneNumber
personName:firstName >> cn
description >> description
personName:lastName >> sn

Do you want to keep this default mapping? (Y/N) [N]:

Please provide your custom mapping (press Enter if unmapped)

organization:
description:
personName:firstName:givenName
personName:middleName:
personName:lastName:sn
personName:fullName:displayName
postalAddresses:postalAddress:city:
postalAddresses:postalAddress:stateOrProvince:
```

```
postalAddresses:postalAddress:country:
postalAddresses:postalAddress:postalScheme:
emailAddresses:emailAddress:address:mail
telephoneNumbers:telephoneNumber:countryCode:
postalAddresses:postalAddress:postalCode:postalCode
postalAddresses:postalAddress:streetNumber:postalAddress
telephoneNumbers:telephoneNumber:areaCode:
telephoneNumbers:telephoneNumber:number:telephoneNumber
telephoneNumbers:telephoneNumber:extension:
telephoneNumbers:telephoneNumber:url:
URL:

===========================================================
Check 3: LDAP User Properties   >> PLEASE WAIT...
-----------------------------------------------------------

The above user properties matched with ldap:

displayName = Alice 0
mail = mail123@test.de
givenName = Alice0
sn = surname



Do you want to see the LDAP search trace? (Y/N) [N]:

Repeat configuration step 4, Continue, or End? (R/C/E) [C]:

===========================================================
Step 4: User Search
-----------------------------------------------------------
The LDAP directory will now be searched based on a search filter that applies to
the user IDs.

search filter (e.g. user*):employee*

===========================================================
Check 4: LDAP User Search   >> PLEASE WAIT...
-----------------------------------------------------------

User elements found: 8. Do you want to see them?

Do you want to see all the users? (Y/N) [N]:y
User with filter: employee*
-> sag\employee1
-> sag\employee3
-> sag\employee4
-> sag\employee5
-> sag\employee6
-> sag\employee7
-> sag\employee8
-> sag\employee2


Do you want to see the LDAP search trace? (Y/N) [N]:

Repeat configuration step 5, Continue, or End? (R/C/E) [C]:

===========================================================
Step 5: Basic Group Information
-----------------------------------------------------------

groupid field [cn]: cn

object class [top,groupOfUniqueNames]: groupOfUniqueNames

Specify the root node (DN) for all groups.


group base DN:ou=groups,ou=abc,o=sag


To test the group retrieval, please provide the group ID of valid LDAP group.


groupid:group1

===========================================================
Check 5: LDAP Group   >> PLEASE WAIT...
-----------------------------------------------------------

[OK]  The following group was found:
cn = group1


Do you want to see the LDAP search trace? (Y/N) [N]:

Repeat configuration step 6, Continue, or End? (R/C/E) [C]:

===========================================================
Step 6: Basic Group Information Mapping
-----------------------------------------------------------
Please provide your group information mapping

description:description

===========================================================
Check 6: LDAP Group Mapping   >> PLEASE WAIT...
-----------------------------------------------------------

[OK]  The following group was found:
description = group1
```

```
Do you want to see the LDAP search trace? (Y/N) [N]:

Repeat configuration step 7, Continue, or End? (R/C/E) [C]:

===========================================================
Step 7: Group Search
-----------------------------------------------------------
The LDAP directory will now be searched based on a search filter that applies to
the group IDs

search filter (e.g. group*):*group*

===========================================================
Check 7: LDAP Group Search   >> PLEASE WAIT...
-----------------------------------------------------------

Group elements found:  6 . Do you want to see them?

Do you want to see all groups? (Y/N) [N] :Y
Groups with filter: *group*
-> sag\invalidgroup
-> sag\group1
-> sag\group2
-> sag\AdminGroup
-> sag\group3
-> sag\group4


Do you want to see the LDAP search trace? (Y/N) [N]:

Repeat configuration step 8, Continue, or End? (R/C/E) [C]:

===========================================================
Step 8: Group Resolution
-----------------------------------------------------------
There are 3 methods available to perform the group resolution
Select one of the following:
"\"Recurse Up "\": Read the (multi-valued) field from the user entry, which poin
ts to the groups.
"\"Recurse Down"\": Search all groups that specify the current/authenticated use
r as a member.
"\"Computed property"\": Use a "\"computed property"\" field from the user entry
, that specifies all groups where the user is a member.

group resolution type (ru, rd or cp) [rd]: rd

group attribute that points to the users [uniqueMember]: uniqueMember

===========================================================
Check 8: LDAP Group Resolution Host   >> PLEASE WAIT...
-----------------------------------------------------------
```

```
Groups for user found: 1 Do you want to see them?
Do you want to see all groups? (Y/N) [N] :
Y
Find all groups of user alice0
-> sag\invalidgroup


Do you want to see the LDAP search trace? (Y/N) [N]:

Repeat configuration step 9, Continue, or End? (R/C/E) [C]:

===========================================================
Check 9: Displaying and saving configuration in CentraSite >> PLEASE WAIT...
-----------------------------------------------------------



Repeat configuration step 10, Continue, or End? (R/C/E) [C]:

===========================================================
Current Configuration
-----------------------------------------------------------

Domain ID: MyDomain
Host: ldap://MyServer01:10388
Server Type: ApacheDS
Expiration : 120

Users:
DN: ou=people,ou=abc,o=sag
Object: inetOrgPerson
Group Attribute: ou
Field: cn

Groups:
DN: ou=groups,ou=abc,o=sag
Object: groupOfUniqueNames
User Attribute: uniqueMember
Resolution: rd


Do you want to save the configuration to CentraSite? (Y/N) [N]:Y
Successfully executed the command : set Authentication
```

6 Performing Maintenance on Authentication Configurations

This section describes how to perform various maintenance operations on your defined authentication configurations.

Testing an existing Authentication Configuration

You can test whether an authentication configuration contains the correct values for accessing the user repository.

Note: The feature is currently only available for LDAP authentication configurations.

To test an LDAP authentication configuration, use the command line tool CentraSiteCommand with the option `validate Authentication`. Details of the tool syntax are provided in the section *Configuring the Authentication Settings* of the document *Basic Operations*.

During the validation, CentraSite attempts to access the user repository and returns status messages indicating the following:

- Whether the user repository is currently accessible.
- Whether the user with the given password exists in the user repository.
- Whether the mappings for users are correct.
- Whether the references between groups and users are correct.

Some of the possible error messages and their causes are listed below:

Check for basic host info failed: Cannot contact the server. SSX LDAP Error: The LDAP search was aborted due to exceeding the limit of the client side timeout parameter (-130)

This message appears if you specify an incorrect port number, even if the host name is correct.

Check for basic user info failed: [ERROR] User authentication for "LDAPUSER-NAME" failed

This message appears if any of the following conditions is met:

- Incorrect DN specified in user information.
- Incorrect object class specified in user information.
- Incorrect user filed in user information.
- Invalid user-ID/password combination.

Editing an existing Authentication Configuration

To edit an existing authentication configuration, use the command line tool CentraSiteCommand with the option `set Authentication`. Details of the tool syntax are provided in the section *Configuring the Authentication Settings* of the document *Basic Operations*.

Deleting an existing Authentication Configuration

If you do not require a particular authentication configuration any more, you can delete it from the list of available configurations.

You cannot remove the pre-installed domain "INTERNAL".

If you remove a configuration that is the current default configuration, the configuration is removed and the default reverts to the INTERNAL configuration.

To delete an existing authentication configuration, use the command line tool CentraSiteCommand with the option `remove Authentication`. Details of the tool syntax are provided in the section *Configuring the Authentication Settings* of the document *Basic Operations*.

Note: When you delete an authentication configuration, CentraSite does not delete the user objects that are associated with this configuration. Thus, these users will still be displayed in the list of users in CentraSite Control, even though the domain to which they belong is no longer accessible to CentraSite.

Setting a new Default Authentication Configuration

If you have defined more than one authentication configuration, you can change the current default configuration to one of the other configurations.

The user domain of the new default configuration must include at least one user who is defined in CentraSite with the "CentraSite Administrator" role, otherwise you will be prompted to enter a user who will be defined as administrator in that configuration.

To set a new default authentication configuration, use the command line tool CentraSiteCommand with the option `set DefaultDomain`. Details of the tool syntax are provided in the section *Configuring the Authentication Settings* of the document *Basic Operations*.

If the user domain of the configuration that you wish to set to the default does not contain any user who is defined in CentraSite with the "CentraSite Administrator" role, a dialog will appear, asking you to provide the user name and password of a domain user who will be granted this role in CentraSite.

If the user already exists in CentraSite, but does not have the "CentraSite Administrator" role, the role will be granted to the user. If the user does not exist in CentraSite, a user with the given user name will be created in CentraSite and will be granted the "CentraSite Administrator" role.

The dialog also allows you to specify an organization for the user, in cases where the user did not already exist in CentraSite. The newly created CentraSite user will be assigned to this organization. If you do not specify an organization, the user is assigned to the default organization.

Users who are in the default domain can log in without having to specify the domain name, but they can specify the domain name if they wish. Users who are not in the current default domain always have to specify the domain name when logging in.

Notes:

1. If your default authentication configuration contains only one user who has the "CentraSite Administrator" role in CentraSite, it is not possible to delete this user from CentraSite, or to remove the "CentraSite Administrator" role from the user. This is because the default configuration must always contain at least one user who is defined in CentraSite with the "CentraSite Administrator" role.
2. If you try to log in to a CentraSite component (for example, CentraSite Control) by supplying a user name and password but no domain name, the authentication mechanism assumes that you belong to the domain of the default configuration and will authenticate you against this domain. If you change the default configuration as described above and subsequently try to log in to a CentraSite component, you must supply your domain name in addition to your user name, so that the authentication mechanism knows which domain to use to check your credentials.

When you set a new default authentication configuration, you might wish to change the association between CentraSite users (i.e. CentraSite registry objects representing users) and users in the external user repository. For information on how to do this, and in particular if you wish to do this for many users, refer to the topic *Re-Associating Users* in the document *Users, Groups, Roles and Permissions*.

7 Creating an Administration User from the Command Line

As stated above, the user domain of the default authentication configuration must contain at least one user who is defined in CentraSite with the "CentraSite Administrator" role. Under certain circumstances, it might happen that such users are no longer available in the user repository, for example:

- if the user repository is currently not available (e.g. LDAP Server currently unavailable);
- if all of the users who have the "CentraSite Administrator" role in CentraSite have been deleted from the user repository.

In such cases, there are no users any more who can log in to CentraSite as a user with the "CentraSite Administrator" role, so no CentraSite administration tasks can be performed.

To resolve this problem, a mechanism is available to create a user in the user repository, assigned to the "CentraSite Administrator" role in CentraSite.

You can create an administration user by executing the following command in the command line interface *CentraSiteCommand.cmd* (Windows) or *CentraSiteCommand.sh* (UNIX) of Command Central. The tool is located in `<CentraSiteInstallDir>`/utilities.

If you start this command line tool with no parameters, you receive a help text summarizing the required input parameters.

The parameters of the command are case-sensitive, so for example the parameter "-url" must be specified as shown and not as "-URL".

The syntax of the command is as follows:

```
CentraSiteCommand add Admin [-url <CENTRASITE-URL>] -user <USER-ID> -password
<PASSWORD> -domain <DOMAIN> -domainUser <DOMAIN-USER-ID> -domainPassword
<DOMAIN-PASSWORD> -organization <ORGANIZATION>
```

The following table describes the complete set of input parameters that you can use with the `add Admin` utility:

Parameter	Description
`-url`	The fully qualified URL (http://localhost:53307/CentraSite/CentraSite) for the CentraSite registry/repository.
`-user`	The user name of a user in the user repository who has the required rights to create a new user in the user repository.
`-password`	The password of the user identified by the parameter "-user".
`-domain`	The domain name of the user repository associated with the configuration.
`-domainUser ↩`	The user name for a new user to be created in the user repository. A user with this name will be created in CentraSite and will have the role "CentraSite Administrator".
`-domainPassword ↩`	The user repository password for the user specified in `domainUser ↩` .
`-organization`	The organization to which the newly created CentraSite user will belong.

8 Logging of Login Authentication Messages

Purpose of Log Files

If you have configured your authentication settings but still experience problems when trying to log in, you can use CentraSite's log files to analyze the problem. Some log file entries contain information about authentication problems in general, whereas other log file entries contain information about authentication problems related to individual CentraSite components.

You can configure the authentication logging mechanisms by modifying parameters in the file *jaas.config*, which is located in `<SuiteInstallDir>`*/profiles/CTP/configuration*. The parameters in this file allow you to make the following changes:

- Switch authentication logging on or off for all CentraSite components.
- Specify the depth of logging required.

Activating the Authentication Logging

The logging of authentication messages is controlled by properties you can set in the *CentraSite* module in the file *jaas.config*.

The *CentraSite* module consists of one or more actions, and each action introduced by a specification such as:

```
com.softwareag.security.jaas.login.ssx.SSXLoginModule ...
```

The first of these sections relates to the SIN component of the authentication mechanism. This is the top-level authentication component, and any logging properties that you specify here apply to the logging for all SIN authentication components that the login module applies to.

The properties that you can specify for the top-level SIN component are:

- **useLog**
  Specify "true" to switch logging on, or "false" to switch logging off.
- **logLevel**
  Specify the level of logging information required. Possible values are: "error" (log only error messages), "info" (log error and information messages) and "debug" (log all messages with additional debug information).
- **logFile**
  Specify the path and file name of the log file.

The properties that you can specify for the individual authentication components are:

- **nativeLogLevel**
  Specify the level of logging information required. You can specify a number from 0 to 6, with 6 providing the most logging information and 0 the least.
- **nativeLogFile**
  Specify the path and file name of the log file.

Here is an example showing logging switched on for SIN and SSX:

```
CentraSite {
  com.softwareag.security.jaas.login.ssx.SSXLoginModule requisite
    useLog="true"
    logLevel="debug"
    logFile="C:/SoftwareAG/profiles/CTP/logs/sin.log"
    nativeLogLevel=6
    nativeLogFile="C:/SoftwareAG/profiles/CTP/logs/cs_internal.log"
    ↩
options_url="http://MyServer:53307/CentraSite/CentraSite/ino:noauth:GetSinConfiguration="
    CreateGroups="false"
    UseDomainForOptionsURL="true";
};
```

This configuration creates the following log files:

- *<SuiteInstallDir>/profiles/CTP/logs/sin.log*
- *<SuiteInstallDir>/profiles/CTP/logs/cs_internal.log*

The log shows whether login attempts are successful or not, and indicates the user domain where CentraSite attempted to find the login user credentials, for example:

```
...Authenticator (<domain>, ...) was created successfully

...login of user <username> (domain: <domain>) was successful.
```

If the authentication was not successful, a message such as the following is displayed:

```
Login of user <username> (host: <hostname>, port:<portnumber>) failed.
```

9 Notes on Authentication in CentraSite

This section contains general information related to authentication in CentraSite.

Case-Sensitivity

User names and domain names are treated as either case-sensitive or case-insensitive, according to the configured authentication mechanism.

INTERNAL authentication	case-sensitive
Active Directory authentication	case-insensitive
LDAP authentication	case-insensitive

Working in an Offline Environment

If you wish to work in an offline environment, for example on a laptop computer that is not connected to the network, you should be aware of certain restrictions that apply in the area of authentication.

Important: When CentraSite is installed in an environment where the users are authenticated against a central service, for example an LDAP server, authentication will not work if the machine is disconnected from the network. So if you intend to use CentraSite on a mobile device when it is not connected to the network, ensure that at least one user is available who can also be authenticated offline, for example from an internal or local operating system user repository.