Connecting beSECURE to an Active Directory server

To test out the Active Directory integration with beSECURE your first step is to setup a Windows 2012 Datacenter (newer and older versions of Windows are supported, Windows 2012 is used as an example as an evaluation copy can be obtained from Microsoft for 180 days)

1. Roll out a “Windows Server 2012 R2 Datacenter” server
2. After the server has been installed and you are greeted with the login screen of Windows Server, login to the server.
3. Start a PowerShell prompt, type Install-windowsfeature -name AD-Domain-Services –IncludeManagementTools and hit Enter.
4. Create two demo Groups by typing:
   New-ADGroup -Name "Accounting" -GroupScope "DomainLocal" hit Enter.
   New-ADGroup -Name "IT" -GroupScope "DomainLocal"
5. Create two demo Users by typing:
   Add-ADGroupMember -Identity Accounting -Members "bob.johnson", "mary.smith" hit Enter.
   Add-ADGroupMember -Identity IT -Members "mary.smith" hit Enter.

At this point you have an Active Directory server which you can then integrate with beSECURE.

When a beSECURE user attempts to logon to the system he will need to provide the password, he was given at the Windows Server (in this case “Pass@word1!”) and click on the “with Active Directory” button rather than the regular “Log In” button.

The username with which he logs on to beSECURE will need to match the one on the Active Directory, in our setup either “bob.johnson” or “mary.smith”.

For security reasons, beSECURE will not first look up the requested account in its own database before attempting to login to the AD – rather it will first attempt to verify the credentials provided with the AD and only then will it try to access the account inside beSECURE.

The beSECURE account needs to be created before the user attempts to login to the beSECURE server with his AD credentials.

The Username created in beSECURE needs to match that of the “sAMAccountName” in the ActiveDirectory.

To find out what is the “sAMAccountName” a user has, you can use this PowerShell script:

```
# Function Find Distinguished Name
function find-dn { param([string]$adfindtype, [string]$cName)
 # Create A New ADSI Call
 $root = [ADSI]''
 # Create a New DirectorySearcher Object
 $searcher = new-object System.DirectoryServices.DirectorySearcher($root)
 # Set the filter to search for a specific CNAME
 $searcher.filter = "(&(objectClass=$adfindtype) (CN=$cName))"
 # Set results in $adfind variable
 $adfind = $searcher.findall()

 # If Search has Multiple Answers
 if ($adfind.count -gt 1) {
 $count = 0
 foreach($i in $adfind)
 {
 # Write Answers On Screen
 write-host $count ": " $i.path
 $count += 1
 }
 # Prompt User for Selection
 $selection = Read-Host "Please select item: "
 # Return the Selection
 write-host "sAMAccountName: " $adfind[$selection].Properties["sAMAccountName"]
 return "Path: " + $adfind[$selection].path
 }
 # Return the Answer
 write-host "sAMAccountName: " $adfind[0].Properties["sAMAccountName"]
 return "Path: " + $adfind[0].path
}
find-dn "user" "Bob Johnson"
```

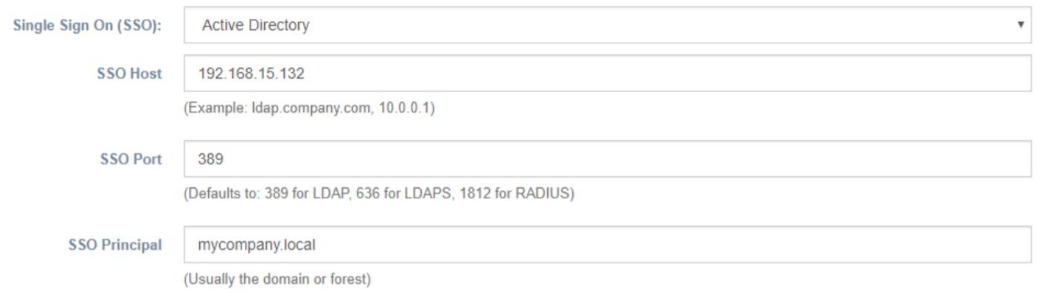
This will return when run:

```
sAMAccountName: bob.johnson
Path: LDAP://CN=Bob Johnson,CN=Users,DC=mycompany,DC=local
```

and the “bob.johnson” is the Username that you will need to put in beSECURE and use to login to beSECURE.

Active Directory SSO

The “Log in Security” settings on beSECURE will be configured as seen in the screenshot below:



Single Sign On (SSO):	Active Directory
SSO Host	192.168.15.132 (Example: ldap.company.com, 10.0.0.1)
SSO Port	389 (Defaults to: 389 for LDAP, 636 for LDAPS, 1812 for RADIUS)
SSO Principal	mycompany.local (Usually the domain or forest)

In the case, where the AD is on the IP address of 192.168.15.132 (a FQDN DNS hostname can also be used), the Port of the LDAP (389) is provided and the Principal of ‘mycompany.local’ is provided (this is based on the output of “LDAP://CN=Bob Johnson,CN=Users,DC=mycompany,DC=local” and the combining of the “DC” values such that it becomes ‘mycompany.local’.

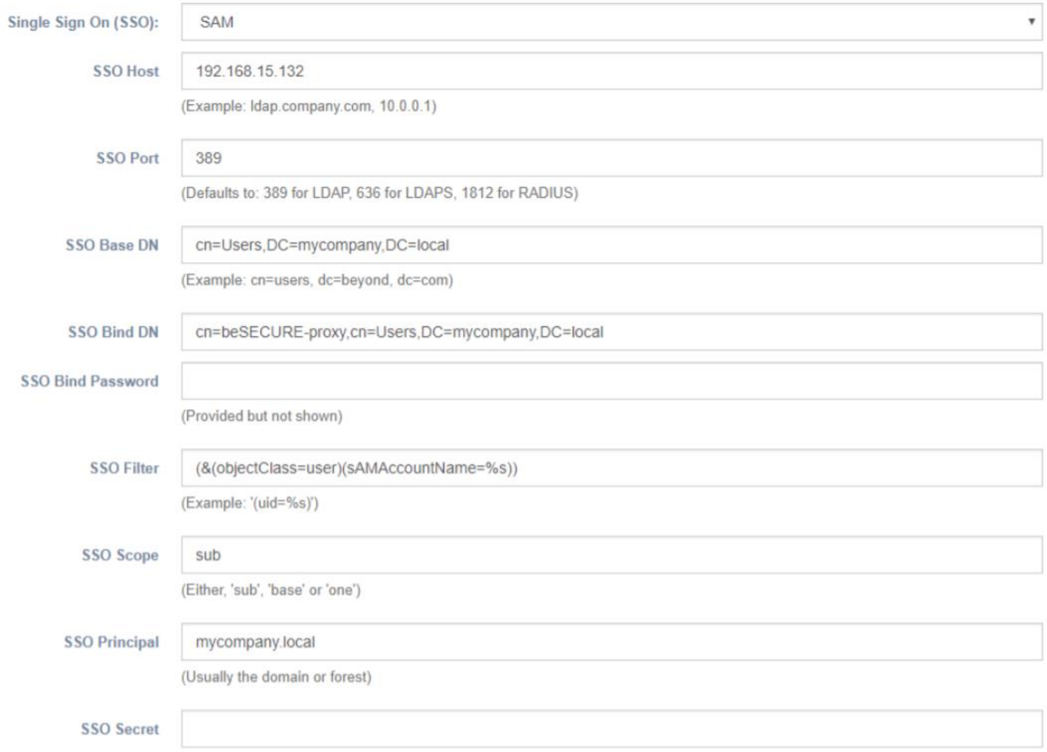
LDAP SSO

Active Directory SSO is the easiest to implement and requires no configuration beyond the 3 values (Host, Port and Principal), but gives very little flexibility to controlling the rights to who can or cannot login.

The more industry standard method of preforming SSO against an Active Directory server is using an LDAP Proxy account.

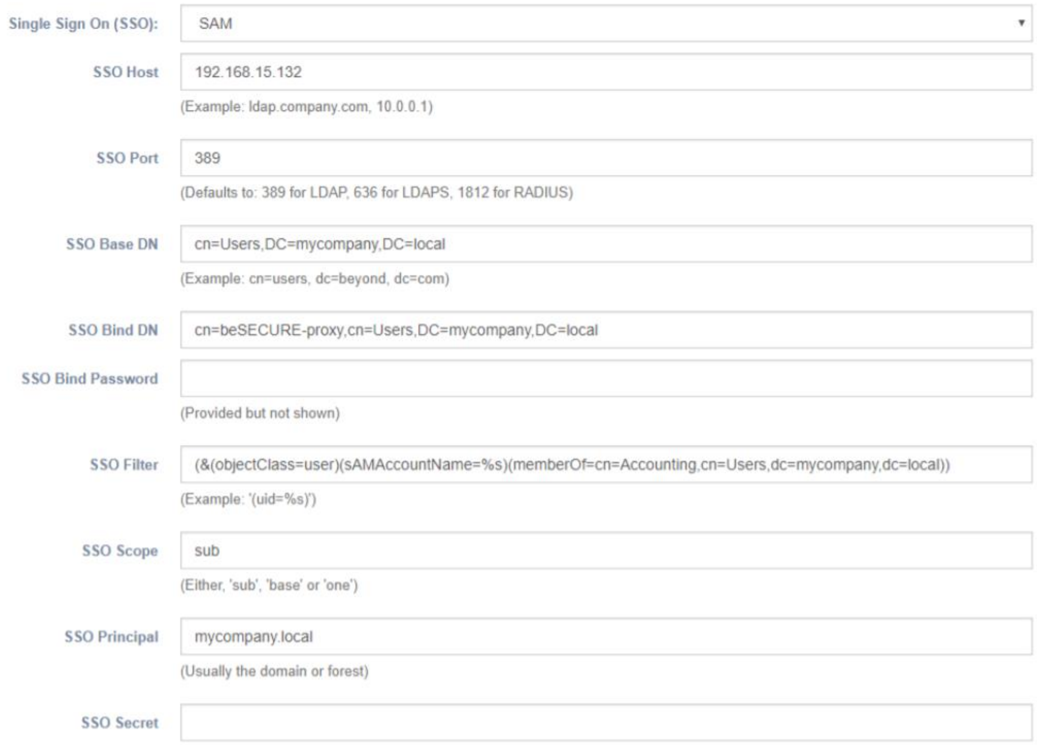
As most modern AD LDAP connections stopped supporting anonymous binding (for security reasons), you will need to create a LDAP Binding account. This account should not be the administrator of your domain – again for security reasons – rather a user that is part of the Domain Administrator group and is dedicated to beSECURE binding account.

For example, a username called ‘beSECURE-proxy’ has been created in our setup.



Single Sign On (SSO):	SAM ▾
SSO Host	192.168.15.132 (Example: ldap.company.com, 10.0.0.1)
SSO Port	389 (Defaults to: 389 for LDAP, 636 for LDAPS, 1812 for RADIUS)
SSO Base DN	cn=Users,DC=mycompany,DC=local (Example: cn=users, dc=beyond, dc=com)
SSO Bind DN	cn=beSECURE-proxy,cn=Users,DC=mycompany,DC=local
SSO Bind Password	 (Provided but not shown)
SSO Filter	(&(objectClass=user)(sAMAccountName=%s)) (Example: '(uid=%s)')
SSO Scope	sub (Either, 'sub', 'base' or 'one')
SSO Principal	mycompany.local (Usually the domain or forest)
SSO Secret	

The SSO Filter, can be further restricted such that only certain users from a certain group can login to the system for example, here limited so that only users who are members of the 'Accounting' group can login to the beSECURE system:



Single Sign On (SSO):	SAM
SSO Host	192.168.15.132 (Example: ldap.company.com, 10.0.0.1)
SSO Port	389 (Defaults to: 389 for LDAP, 636 for LDAPS, 1812 for RADIUS)
SSO Base DN	cn=Users,DC=mycompany,DC=local (Example: cn=users, dc=beyond, dc=com)
SSO Bind DN	cn=beSECURE-proxy,cn=Users,DC=mycompany,DC=local
SSO Bind Password	 (Provided but not shown)
SSO Filter	(&(objectClass=user)(sAMAccountName=%s)(memberOf=cn=Accounting,cn=Users,dc=mycompany,dc=local)) (Example: '(uid=%s)')
SSO Scope	sub (Either, 'sub', 'base' or 'one')
SSO Principal	mycompany.local (Usually the domain or forest)
SSO Secret	

When a customer receives the beSECURE setup it would normally include a VM machine and there are a few set up options:

- IS and LSS (beSECUREII)
- IS (if customer already owns LSS)
- LSS if the customer would like to deploy a new Scanning Server on his network.