CUAC-Integration mit Microsoft AD

Inhalt

Einführung

In diesem Dokument wird die Funktionsweise des Lightweight Directory Access Protocol (LDAP) zwischen der Cisco Unified Attendant Console (CUAC) und dem Microsoft Active Directory (AD) sowie die Verfahren beschrieben, die zur Integration der beiden Systeme verwendet werden.

Voraussetzungen

Anforderungen

Cisco empfiehlt, dass Sie über Kenntnisse in folgenden Bereichen verfügen:

- CUCM
- CUAC
- LDAP
- AD

Verwendete Komponenten

Die Informationen in diesem Dokument basieren auf der CUAC-Version 10.x.

Die Informationen in diesem Dokument beziehen sich auf Geräte in einer speziell eingerichteten Testumgebung. Alle Geräte, die in diesem Dokument benutzt wurden, begannen mit einer gelöschten (Nichterfüllungs) Konfiguration. Wenn Ihr Netz Live ist, überprüfen Sie, ob Sie die mögliche Auswirkung jedes möglichen Befehls verstehen.

Hintergrundinformationen

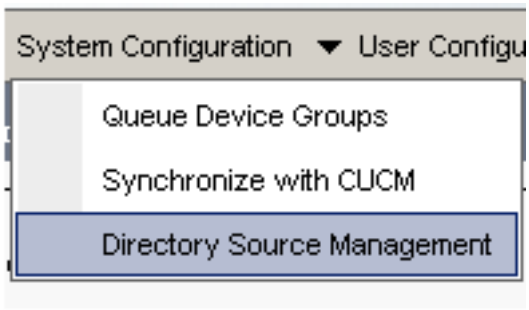
In früheren CUAC-Versionen bezieht der Server Benutzer direkt vom Cisco Unified Communications Manager (CUCM) über vordefinierte Abfragen und Filter. Mit der CUAC Premium Edition (CUACPE) können Administratoren Benutzer direkt aus dem AD integrieren und importieren. Dadurch können Administratoren Attribute und Filter ihrer Wahl und Anforderungen flexibel implementieren.

> **Hinweis:** CUACPE wurde jetzt durch die CUAC Advanced Edition für Version 10 und höher ersetzt.

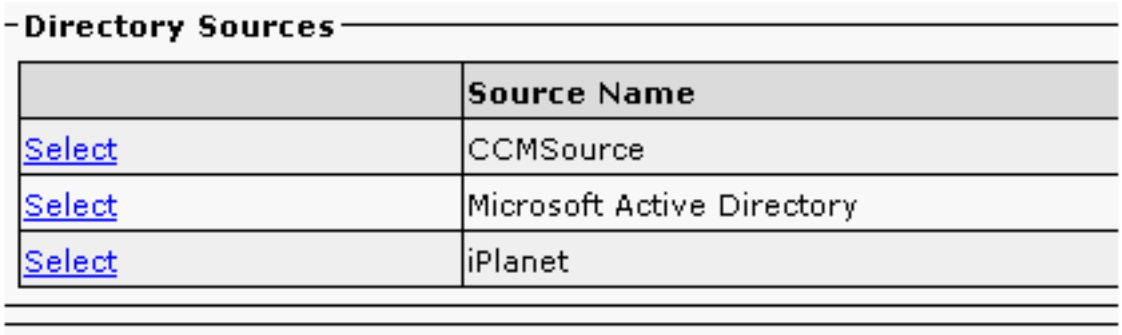
Integration von AD in CUAC und Import von Benutzern aus AD

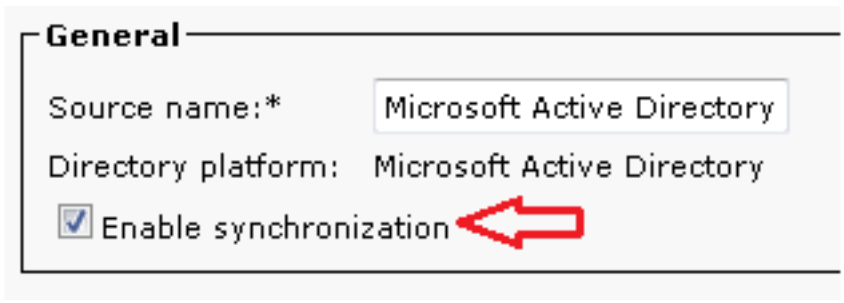
Gehen Sie wie folgt vor, um die CUAC in das AD zu integrieren und Benutzer aus dem AD zu importieren:

1. Aktivieren Sie die Verzeichnissynchronisierung für AD auf dem CUAC.

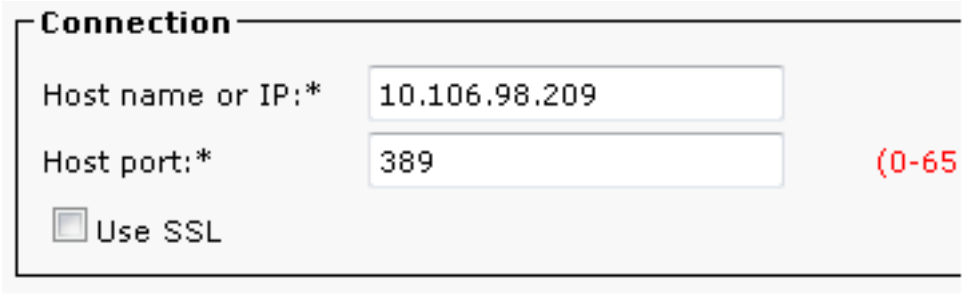


2. Wählen Sie **Microsoft Active Directory aus**, und aktivieren Sie das Kontrollkästchen **Synchronisierung aktivieren**:

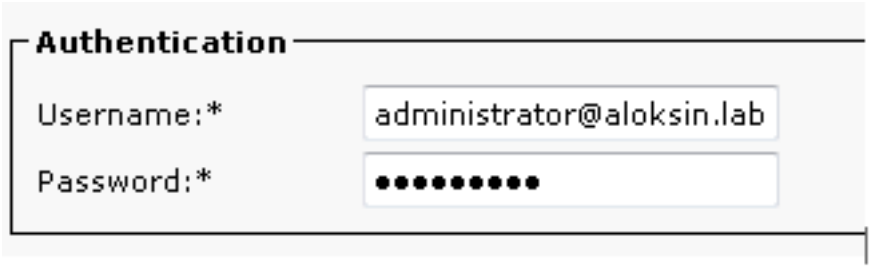




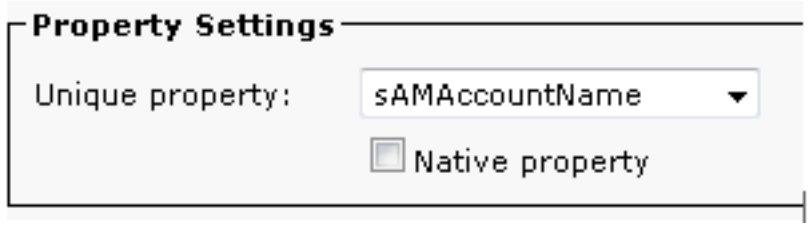
3. Geben Sie die Konfigurationsdetails für den Active Directory-Server ein:



In diesem Beispiel wird **administrator@aloksin.lab** für die Authentifizierung verwendet:



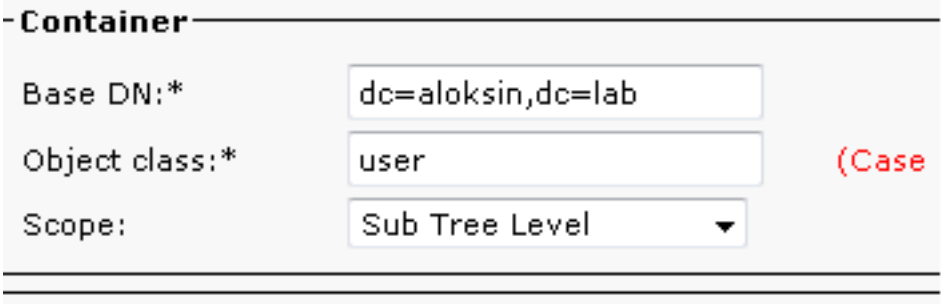
4. Geben Sie im Abschnitt Eigenschafteneinstellungen die Konfigurationsdetails für die Unique-Eigenschaft ein, die angezeigt wird, sobald Sie die anderen Details eingegeben haben, und klicken Sie auf **Speichern**.



Hinweis: Dies ist ein eindeutiger Wert für jeden Eintrag im AD. Wenn doppelte Werte vorliegen, wird vom CUAC nur ein Eintrag abgerufen.

5. Geben Sie im Bereich Container die Konfigurationsdetails für die Basis-DN ein, die der Suchbereich für Benutzer im AD ist.

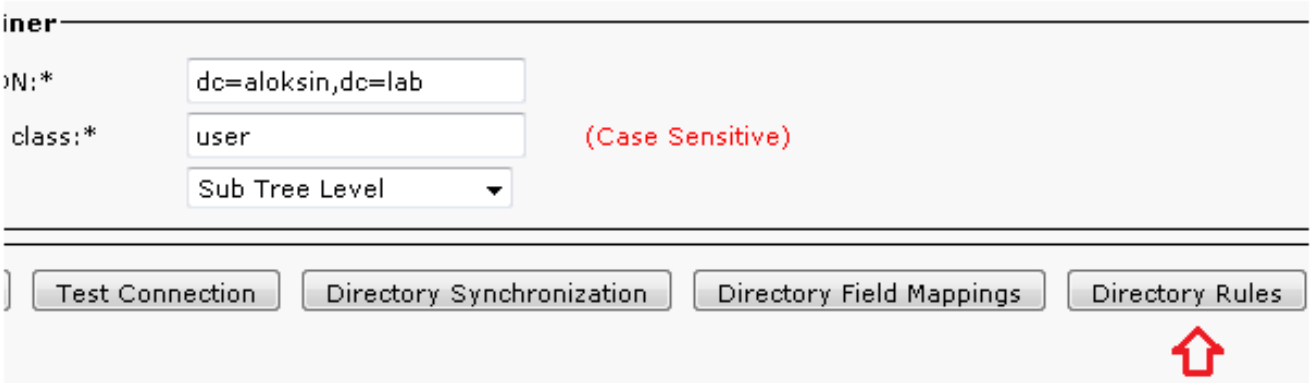
   Das Feld *Object-Klasse* wird vom AD verwendet, um den angeforderten Suchbereich zu bestimmen. Standardmäßig ist sie auf *Contact* festgelegt, was bedeutet, dass das AD nach *Kontakten* (nicht Benutzern) in der angeforderten Suchbasis sucht. Um *Benutzer* in CUAC zu importieren, ändern Sie die Einstellung Object-Klasse in **user**:



6. Speichern Sie die Einstellungen, klicken Sie auf **Verzeichnisfeldzuordnungen**, und konfigurieren Sie alle Attribute, die Sie für einen beliebigen Benutzer importieren möchten. Im folgenden Beispiel wird die Konfiguration verwendet:

Source Fields	Destination Fields	Defaul
telephoneNumber	Extension	
mail	Email	
givenName	First Name	
sn	Last Name	

7. Navigieren Sie zur Quellseite des Verzeichnisses, und klicken Sie auf **Verzeichnisregeln**:

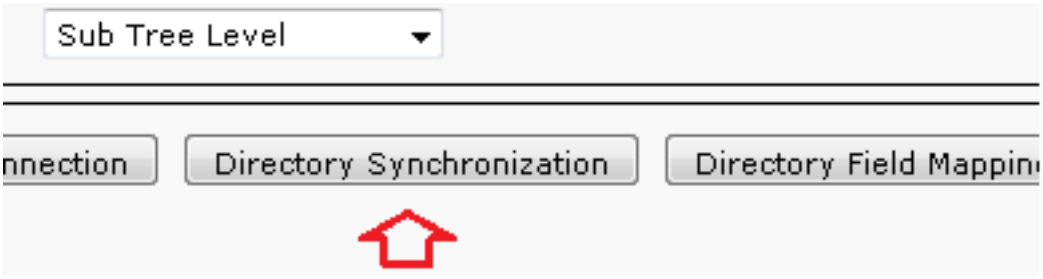


8. Klicken Sie auf **Neu hinzufügen**, und erstellen Sie eine Regel. Wenn Sie eine Verzeichnisregel hinzufügen, wird standardmäßig ein Regelfilter angezeigt.

Field	Operator	Value
telephoneNumber	=	*

Hinweis: Der Regelfilter muss nicht geändert werden. Es importiert alle Benutzer, für die eine Telefonnummer konfiguriert wurde.

9. Um die automatische Synchronisierung mit dem AD zu konfigurieren, klicken Sie auf die Registerkarte **Verzeichnissynchronisierung**.



10. Die Konfiguration ist nun abgeschlossen. Navigieren Sie zu **Engineering > Service Management** und starten Sie das LDAP-Plugin neu, um die Synchronisierung manuell zu starten.

LDAP-Funktionalität zwischen CUAC und AD

LDAP-Prozesszusammenfassung

Im Folgenden finden Sie eine Zusammenfassung des LDAP-Prozesses zwischen CUAC und AD:

1. Zwischen den beiden Servern (CUAC und AD) wird eine TCP-Sitzung eingerichtet.

2. Der CUAC sendet eine BIND-Anforderung an das AD und authentifiziert sich über den Benutzer, der in den Authentifizierungseinstellungen konfiguriert ist.

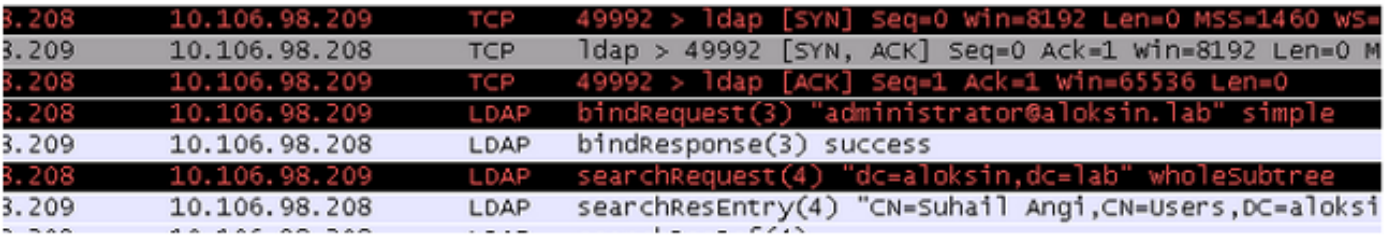
3. Sobald das AD den Benutzer erfolgreich authentifiziert hat, sendet es eine BIND Success-Benachrichtigung an das CUACPE.

4. Der CUAC sendet eine SUCHanforderung an das AD, das über die Suchbereichsinformationen, Filter für die Suche und Attribute für jeden gefilterten Benutzer verfügt.

5. Das AD sucht in der Suchbasis nach dem angeforderten Objekt (das in den Objektklasseneinstellungen konfiguriert ist). Es filtert Objekte, die den in der SUCH-Anforderungsnachricht angegebenen Kriterien (Filter) entsprechen.

6. Das AD antwortet mit den Suchergebnissen auf das CUAC.

Hier eine Sniffer-Erfassung, die die folgenden Schritte veranschaulicht:

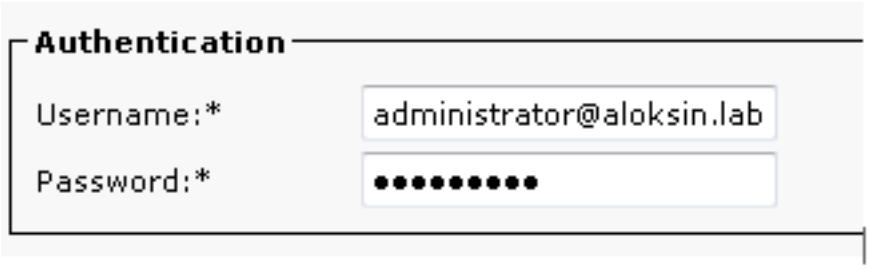


LDAP-Prozessdetails

Sobald die Konfiguration auf dem CUAC abgeschlossen und das LDAP-Plug-In neu gestartet wurde, richtet der CUAC-Server eine TCP-Sitzung mit dem AD ein.

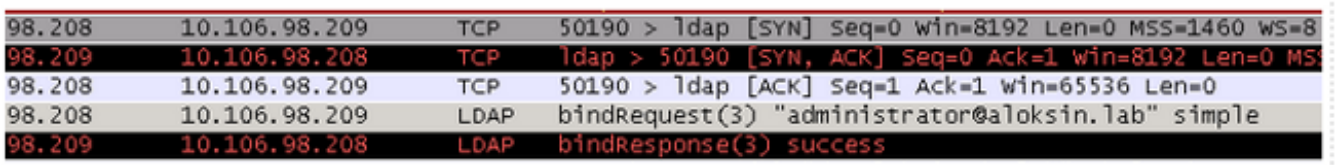
Der CUAC sendet dann eine BIND-Anfrage, um sich beim AD-Server zu authentifizieren. Wenn die Authentifizierung erfolgreich ist, sendet das AD eine BIND Success-Antwort an das CUAC. Dadurch versuchen beide Server, eine Sitzung auf Port 389 einzurichten, um Benutzer und deren Informationen zu synchronisieren.

Die folgende Konfiguration auf dem Server definiert den Distinguished Name, der für die Authentifizierung in der BIND-Transaktion verwendet wird:

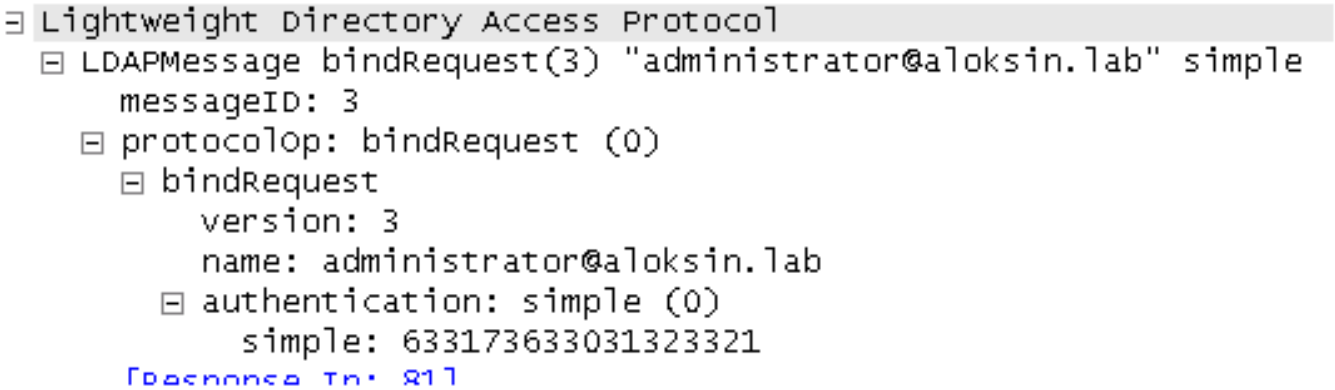


Diese Meldungen werden in der Paketerfassung angezeigt:

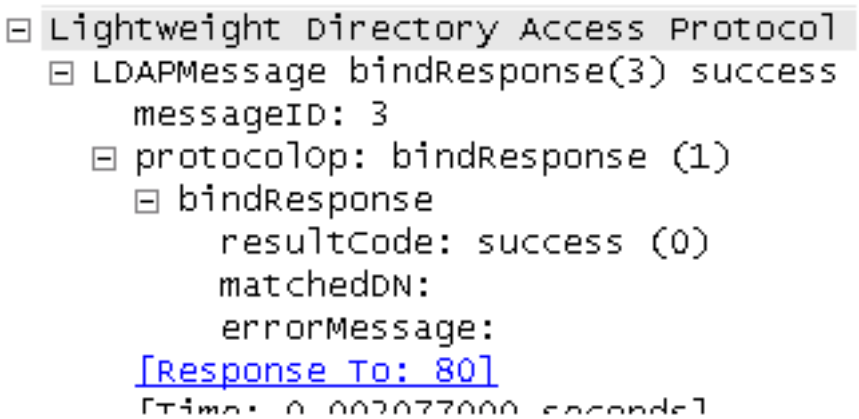
- Hier ist der TCP-Handshake gefolgt von der BIND-Anforderung:



- Hier die Erweiterung der BIND-Anforderung:



- Hier ist die Erweiterung der BIND-Antwort, die auf eine erfolgreiche Authentifizierung des Benutzers hinweist (**Administrator** in diesem Beispiel):



Nach erfolgreicher Bindung sendet der Server eine SUCHanforderung an das AD, um Benutzer zu importieren. Diese Suchanfrage enthält die Filter und Attribute, die vom AD verwendet werden. Das AD sucht dann nach Benutzern in der definierten Suchbasis (wie in der SUCHnachricht beschrieben), die die Kriterien des Filters und der Attributüberprüfung erfüllt.

Im Folgenden finden Sie ein Beispiel für die vom CUCM gesendete SUCHanforderung:

```
Lightweight Directory Access Protocol
    LDAPMessage searchRequest(2) "dc=aloksin,dc=lab" wholeSubtree
        messageID: 2
        protocolOp: searchRequest (3)
            searchRequest
                baseObject: dc=aloksin,dc=lab
                scope: wholeSubtree (2)
                derefAliases: derefAlways (3)
                sizeLimit: 0
                timeLimit: 0
                typesOnly: False
                Filter: (&(&(objectclass=user)(!(objectclass=Computer)))
(!(UserAccountControl:1.2.840.113556.1.4.803:=2)))
                    filter: and (0)
                        and: (&(&(objectclass=user)(!(objectclass=Computer)))
(!(UserAccountControl:1.2.840.113556.1.4.803:=2)))
                            and: 3 items
                                Filter: (objectclass=user)
                                    and item: equalityMatch (3)
                                        equalityMatch
                                            attributeDesc: objectclass
                                            assertionValue: user
                                Filter: (!(objectclass=Computer))
                                    and item: not (2)
                                        Filter: (objectclass=Computer)
                                            not: equalityMatch (3)
                                                equalityMatch
                                                    attributeDesc: objectclass
                                                    assertionValue: Computer
                                Filter: (!(UserAccountControl:1.2.840.113556.1.4.
803:=2))
                                    and item: not (2)
                                        Filter: (UserAccountControl:1.2.840.113556
.1.4.803:=2)
                                            not: extensibleMatch (9)
                                                extensibleMatch UserAccountControl
                                                    matchingRule: 1.2.840.113556.
1.4.803
                                                    type: UserAccountControl
                                                    matchValue: 2
                                                    dnAttributes: False
                attributes: 15 items
                    AttributeDescription: objectguid
                    AttributeDescription: samaccountname
                    AttributeDescription: givenname
                    AttributeDescription: middlename
                    AttributeDescription: sn
                    AttributeDescription: manager
                    AttributeDescription: department
                    AttributeDescription: telephonenumber
                    AttributeDescription: mail
                    AttributeDescription: title
                    AttributeDescription: homephone
                    AttributeDescription: mobile
                    AttributeDescription: pager
                    AttributeDescription: msrtcsip-primaryuseraddress
                    AttributeDescription: msrtcsip-primaryuseraddress
        [Response In: 103]
        controls: 1 item
            Control
                controlType: 1.2.840.113556.1.4.319 (pagedResultsControl)
                criticality: True
```

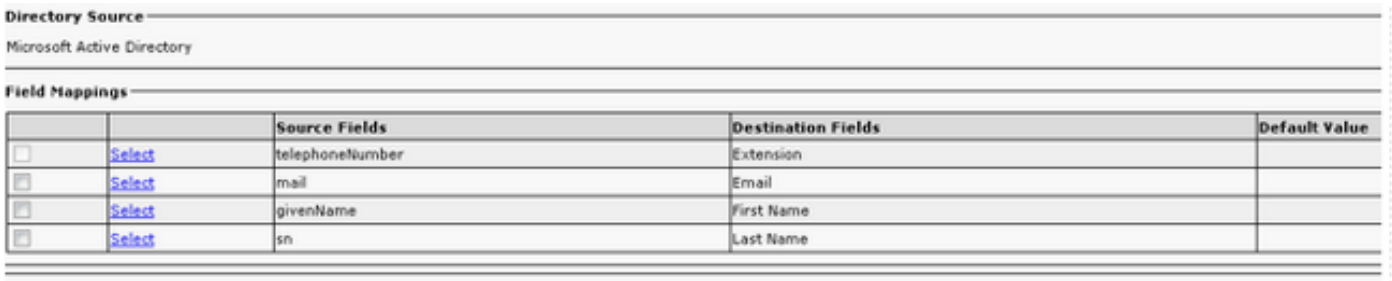
```
SearchControlValue
    size: 250
    cookie: <MISSING>
```

Wenn das AD diese Anforderung vom CUCM empfängt, sucht es nach Benutzern im **baseObject: dc=aloksin,dc=lab**, das den Filter erfüllt. Jeder Benutzer, der die vom Filter angegebenen Anforderungen nicht erfüllt, wird ausgeschlossen. Das AD antwortet mit allen gefilterten Benutzern auf den CUCM und sendet die Werte für die angeforderten Attribute.

> **Hinweis:** Objekte können nicht importiert werden. Nur *Benutzer* werden importiert. Der Grund hierfür ist, dass der Filter, der in der SUCHE-Anforderungsnachricht gesendet wird, **objectclass=user** enthält. Daher sucht das AD nur nach Benutzern, nicht nach Kontakten. Der CUCM verfügt standardmäßig über alle diese Zuordnungen und einen Filter.

Das CUAC ist nicht standardmäßig konfiguriert. Da keine Zuordnungsdetails konfiguriert sind, um Attribute für Benutzer zu importieren, müssen Sie diese Details manuell eingeben. Um diese Zuordnungen zu erstellen, navigieren Sie zu **Systemkonfiguration > Verzeichnisquellenverwaltung > Active Directory > Directory Field Mapping**.

Administratoren können Felder nach ihren eigenen Anforderungen zuordnen. Hier ein Beispiel:

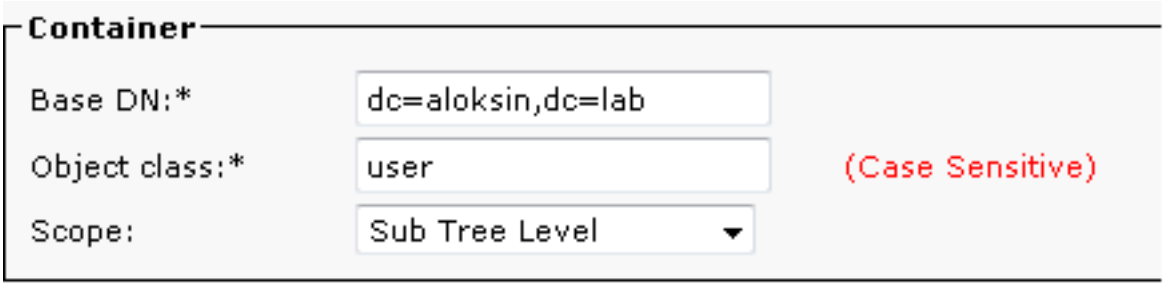


Die Informationen für das Ausgangsfeld werden in der Anforderungsnachricht SUCHEN an das AD gesendet. Wenn das AD die Antwortmeldung SUARCH sendet, werden diese Werte in den Zielfeldern des CUACPE gespeichert.

Beachten Sie, dass für CUAC standardmäßig die Object-Klasse auf *Kontakte* festgelegt ist. Wenn diese Standardeinstellung verwendet wird, wird der an das AD gesendete Filter wie folgt angezeigt:

```
Filter: (&(&(objectclass=contact)(  ............
```

Mit diesem Filter gibt das AD niemals Benutzer an CUACPE zurück, da es nach *Kontakten* in der Suchbasis sucht, nicht nach *Benutzern*. Aus diesem Grund müssen Sie Object Class in **user** ändern:



Bis zu diesem Zeitpunkt wurden diese Einstellungen für CUAC konfiguriert:

- Verbindungsdetails
- Authentifizierung (Distinguished User for Binding)
- Containereinstellungen
- Verzeichniszuordnung

In diesem Beispiel wird die Unique-Eigenschaft als **sAMAccountName** konfiguriert. Wenn Sie das LDAP-Plug-In auf dem CUAC neu starten und die SUCHE-Anforderungsmeldung überprüfen, enthält es außer dem **ObjectClass=user** keine Attribute oder Filter:

```
Lightweight Directory Access Protocol
   LDAPMessage searchRequest(224) "dc=aloksin,dc=lab" wholeSubtree
       messageID: 224
       protocolOp: searchRequest (3)
           searchRequest
               baseObject: dc=aloksin,dc=lab
               scope: wholeSubtree (2)
               derefAliases: neverDerefAliases (0)
               sizeLimit: 1
               timeLimit: 0
               typesOnly: True
               Filter: (ObjectClass=user)
                   filter: equalityMatch (3)
                       equalityMatch
                           attributeDesc: ObjectClass
                           assertionValue: user
               attributes: 0 items
       [Response In: 43]
```

Beachten Sie, dass hier die Verzeichnisregel fehlt. Um die Kontakte mit dem AD zu synchronisieren, müssen Sie eine Regel erstellen. Standardmäßig ist keine Verzeichnisregel konfiguriert. Sobald ein Filter erstellt wurde, ist dieser bereits vorhanden. Sie müssen den Filter nicht ändern, da Sie alle Benutzer mit einer Telefonnummer importieren müssen.

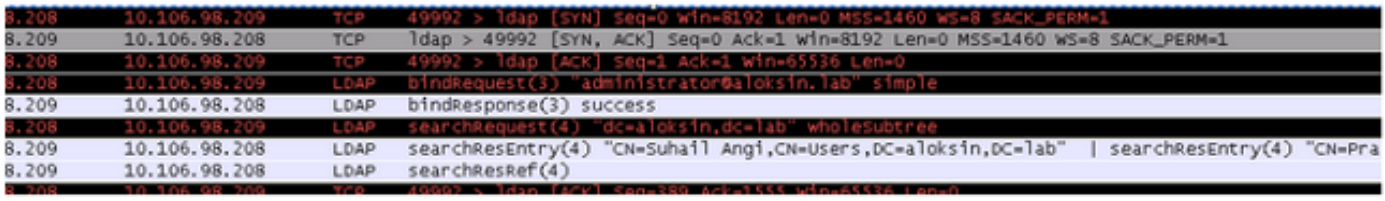
Field	Operator	Val
telephoneNumber	=	*

Starten Sie das LDAP-Plug-In neu, um eine Synchronisierung mit dem AD zu initiieren und die Benutzer zu importieren. Die SUCHanforderung von CUAC sieht Folgendes vor:

```
Lightweight Directory Access Protocol
   LDAPMessage searchRequest(4) "dc=aloksin,dc=lab" wholeSubtree
       messageID: 4
       protocolOp: searchRequest (3)
           searchRequest
               baseObject: dc=aloksin,dc=lab
               scope: wholeSubtree (2)
               derefAliases: neverDerefAliases (0)
               sizeLimit: 0
               timeLimit: 15
               typesOnly: False
               Filter: (&(&(objectclass=user)(telephoneNumber=*))
(!(UserAccountControl:1.2.840.113556.1.4.803:=2)))
                   filter: and (0)
                       and: (&(&(objectclass=user)(telephoneNumber=*))
(!(UserAccountControl:1.2.840.113556.1.4.803:=2)))
                           and: 3 items
                               Filter: (objectclass=user)
                                   and item: equalityMatch (3)
```

```
                                        equalityMatch
                                            attributeDesc: objectclass
                                            assertionValue: user
                                    Filter: (telephoneNumber=*)
                                        and item: present (7)
                                            present: telephoneNumber
                                    Filter: (!(UserAccountControl:1.2.840.113556.
1.4.803:=2))
                                        and item: not (2)
                                            Filter: (UserAccountControl:1.2.840.113556.
1.4.803:=2)
                                                not: extensibleMatch (9)
                                                    extensibleMatch UserAccountControl
                                                        matchingRule: 1.2.840.113556.1.
4.803
                                                        type: UserAccountControl
                                                        matchValue: 2
                                                        dnAttributes: False
                attributes: 10 items
                    AttributeDescription: TELEPHONENUMBER
                    AttributeDescription: MAIL
                    AttributeDescription: GIVENNAME
                    AttributeDescription: SN
                    AttributeDescription: sAMAccountName
                    AttributeDescription: ObjectClass
                    AttributeDescription: whenCreated
                    AttributeDescription: whenChanged
                    AttributeDescription: uSNCreated
                    AttributeDescription: uSNChanged
        [Response In: 11405]
        controls: 1 item
            Control
                controlType: 1.2.840.113556.1.4.319 (pagedResultsControl)
                SearchControlValue
                    size: 500
                    cookie: <MISSING>
```

Wenn das AD Benutzer findet, die den in der SUCHanforderungsnachricht angegebenen Kriterien entsprechen, sendet es eine *SearchResEntry*-Nachricht, die die Benutzerinformationen enthält.



Die Nachricht SearchResEntry lautet wie folgt:

```
Lightweight Directory Access Protocol
   LDAPMessage searchResEntry(4) "CN=Suhail Angi,CN=Users,DC=aloksin,DC=lab" [4 results]
       messageID: 4
       protocolOp: searchResEntry (4)
           searchResEntry
               objectName: CN=Suhail Angi,CN=Users,DC=aloksin,DC=lab
               attributes: 9 items
                   PartialAttributeList item objectClass
                       type: objectClass
                       vals: 4 items
                           top
                           person
                           organizationalPerson
```

```
                    user
                PartialAttributeList item sn
                    type: sn
                    vals: 1 item
                        Angi
                PartialAttributeList item telephoneNumber
                    type: telephoneNumber
                    vals: 1 item
                        1002
                PartialAttributeList item givenName
                    type: givenName
                    vals: 1 item
                        Suhail
                PartialAttributeList item whenCreated
                    type: whenCreated
                    vals: 1 item
                        20131222000850.0Z
                PartialAttributeList item whenChanged
                    type: whenChanged
                    vals: 1 item
                        20131222023413.0Z
                PartialAttributeList item uSNCreated
                    type: uSNCreated
                    vals: 1 item
                        12802
                PartialAttributeList item uSNChanged
                    type: uSNChanged
                    vals: 1 item
                        12843
                PartialAttributeList item sAMAccountName
                    type: sAMAccountName
                    vals: 1 item
                        sangi
        [Response To: 11404]
        [Time: 0.001565000 seconds]
Lightweight Directory Access Protocol
    LDAPMessage searchResEntry(4) "CN=Pragathi NS,CN=Users,DC=aloksin,DC=lab" [5 results]
        messageID: 4
        protocolOp: searchResEntry (4)
            searchResEntry
                objectName: CN=Pragathi NS,CN=Users,DC=aloksin,DC=lab
                attributes: 9 items
                    PartialAttributeList item objectClass
                        type: objectClass
                        vals: 4 items
                            top
                            person
                            organizationalPerson
                            user
                    PartialAttributeList item sn
                        type: sn
                        vals: 1 item
                            NS
                    PartialAttributeList item telephoneNumber
                        type: telephoneNumber
                        vals: 1 item
                            1000
                           .......
                           ....{message truncated}.........
                           .....
```

Hinweis: Die Antwort enthält keine MAIL, obwohl dieses Attribut angefordert wird. Dies liegt daran, dass die MAIL-ID nicht für Benutzer im AD konfiguriert wurde.

Wenn diese Werte vom CUAC empfangen wurden, werden sie in der SQL-Tabelle (Structured Query Language) gespeichert. Sie können sich dann bei der Konsole anmelden, und die Konsole ruft die Benutzerliste aus dieser SQL-Tabelle auf dem CUACPE-Server ab.