Integrazione CUAC con Microsoft AD

Sommario

Introduzione

In questo documento viene descritto il funzionamento del protocollo LDAP (Lightweight Directory Access Protocol) tra Cisco Unified Attendant Console (CUAC) e Microsoft Active Directory (AD) e le procedure utilizzate per integrare i due sistemi.

Prerequisiti

Requisiti

Cisco raccomanda la conoscenza dei seguenti argomenti:

- CUCM
- CUAC
- LDAP
- AD

Componenti usati

Le informazioni fornite in questo documento si basano sulla versione 10.x di CUAC.

Le informazioni discusse in questo documento fanno riferimento a dispositivi usati in uno specifico ambiente di emulazione. Su tutti i dispositivi menzionati nel documento la configurazione è stata ripristinata ai valori predefiniti. Se la rete è operativa, valutare attentamente eventuali conseguenze derivanti dall'uso dei comandi.

Premesse

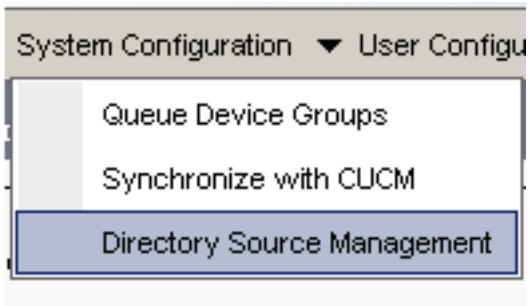
Nelle versioni precedenti di CUAC, il server ottiene gli utenti direttamente da Cisco Unified Communications Manager (CUCM) tramite query e filtri predefiniti. Con CUAC Premium Edition (CUACPE), gli amministratori possono integrare e importare gli utenti direttamente da AD. Ciò garantisce agli amministratori flessibilità per l'implementazione di attributi e filtri di propria scelta e requisiti.

> **Nota:** CUACPE è stato sostituito da CUAC Advanced Edition per le versioni 10 e successive.

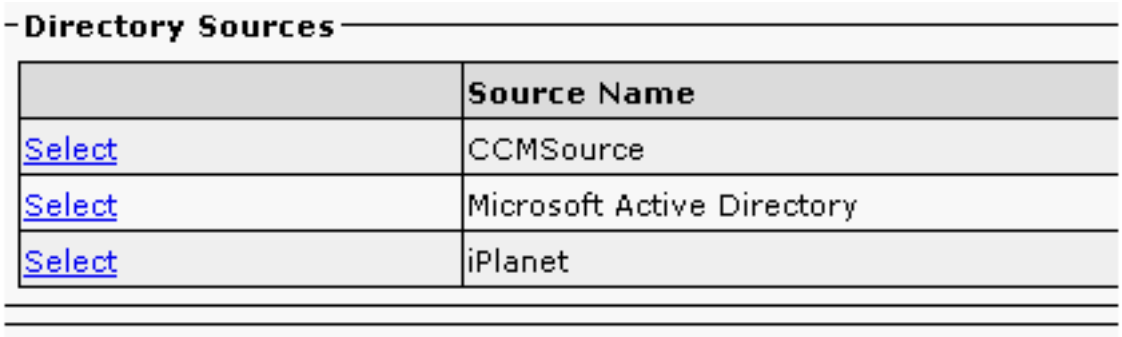
Integrare AD con CUAC e importare utenti da AD

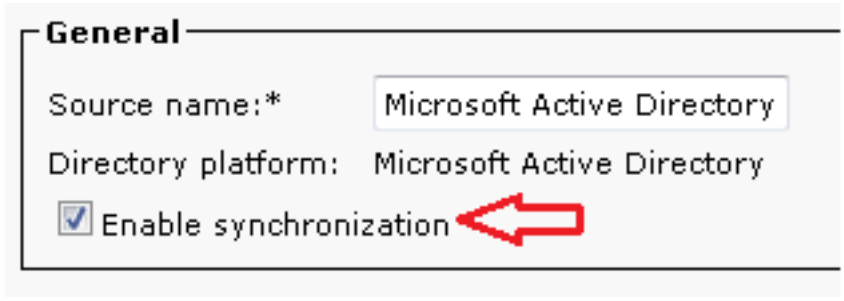
Completare questi passaggi per integrare CUAC con AD e importare gli utenti da AD:

1. Abilitare la sincronizzazione della directory per AD in CUAC.

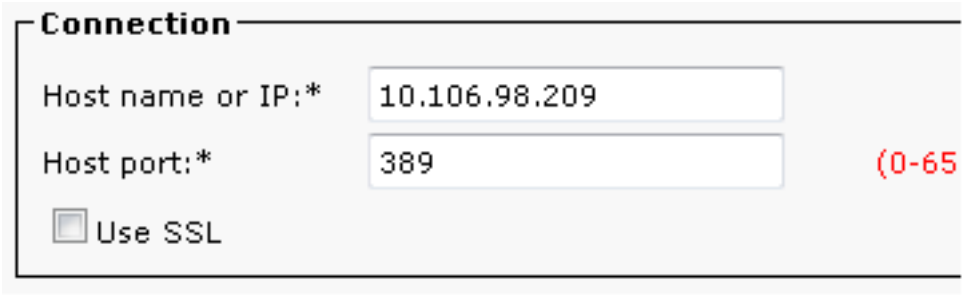


2. Selezionare **Microsoft Active Directory** e selezionare la casella di controllo **Abilita sincronizzazione**:

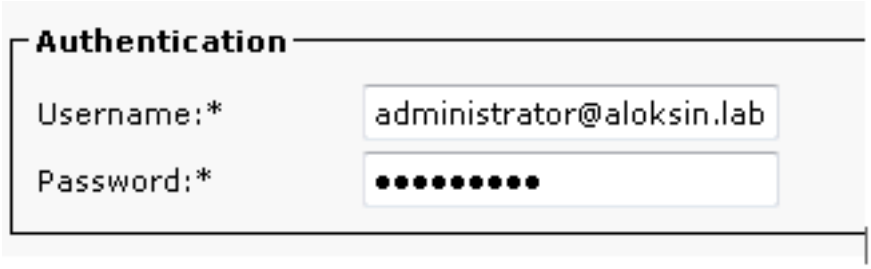




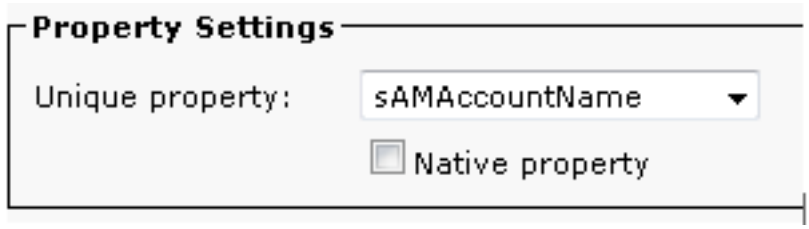
3. Immettere i dettagli di configurazione per il server Active Directory:



Nell'esempio, **administrator@aloksin.lab** viene utilizzato per l'autenticazione:



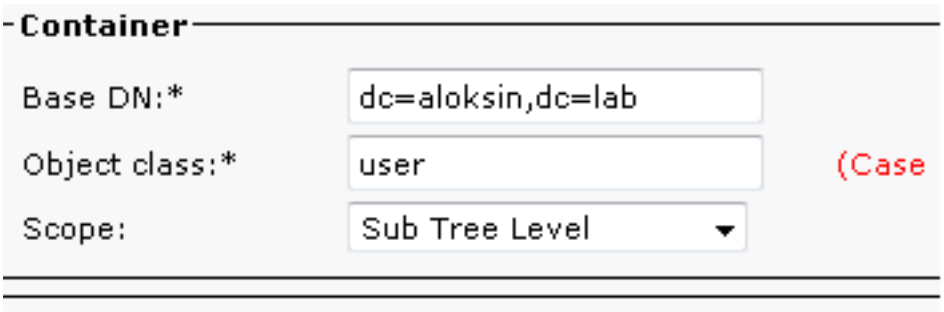
4. Nella sezione Impostazioni proprietà immettere i dettagli di configurazione per la proprietà Unique, visualizzata dopo aver immesso gli altri dettagli e aver fatto clic su **Salva**.



Nota: Si tratta di un valore univoco per ogni voce in AD. Se sono presenti valori duplicati, CUAC estrae una sola voce.

5. Nella sezione Contenitore immettere i dettagli di configurazione per il DN di base, ovvero l'ambito di ricerca utente in Active Directory.

   Il campo *Classe oggetto* viene utilizzato da Active Directory per determinare l'ambito di ricerca richiesto. Per impostazione predefinita, è impostato su *contatto*, il che significa che Active Directory cerca *contatti* (non utenti) nella base di ricerca richiesta. Per importare *gli utenti* in CUAC, modificare l'impostazione della classe Object in **user**:

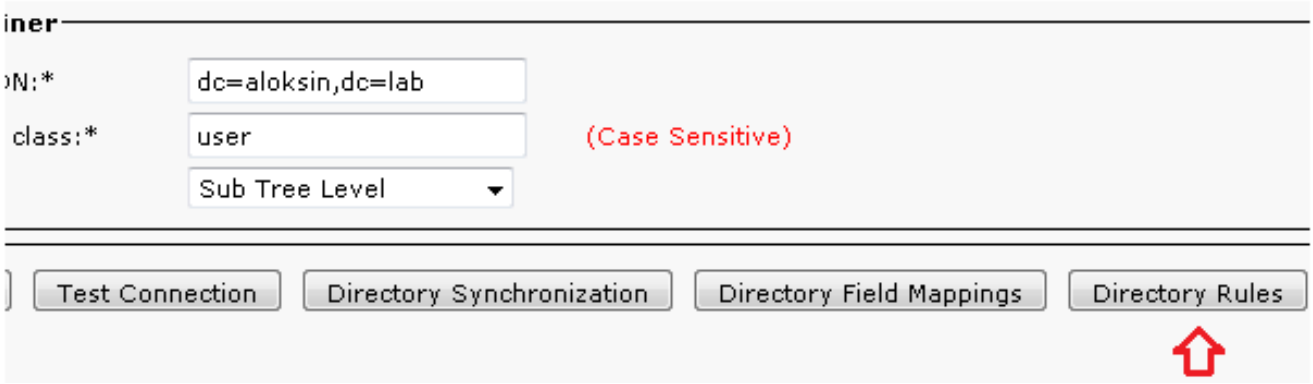


6. Salvare le impostazioni, fare clic su **Mapping campi directory** e configurare tutti gli attributi che si desidera importare per qualsiasi utente. Di seguito è riportata la configurazione

utilizzata in questo esempio:

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

7. Passare alla pagina Origine directory e fare clic su **Regole directory**:



8. Fare clic su **Aggiungi nuovo** e creare una regola. Quando si aggiunge una regola di directory, per impostazione predefinita viene visualizzato un filtro delle regole.

	Field	Operator	Value
	telephoneNumber	=	*

Nota: Non è necessario modificare il filtro delle regole. Importa tutti gli utenti con un numero di telefono configurato.

9. Per configurare la sincronizzazione automatica con Active Directory, fare clic sulla scheda **Sincronizzazione directory**.

10. Configurazione completata. Passare a **Engineering > Gestione servizi** e riavviare il plugin LDAP per avviare manualmente la sincronizzazione.

Funzionalità LDAP tra CUAC e AD

Riepilogo processo LDAP

Di seguito è riportato un riepilogo del processo LDAP tra CUAC e AD:

1. Viene stabilita una sessione TCP tra i due server (CUAC e AD).

2. CUAC invia una richiesta BIND ad AD e esegue l'autenticazione tramite l'utente configurato nelle impostazioni di autenticazione.

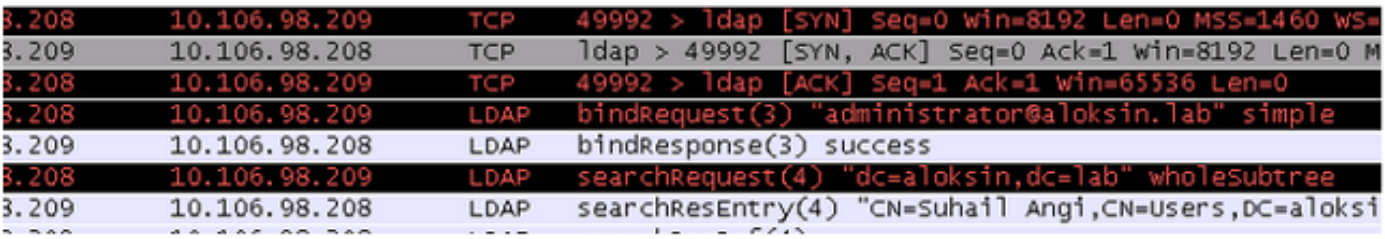
3. Una volta che l'AD ha autenticato l'utente, invia una notifica BIND riuscita a CUACPE.

4. CUAC invia una richiesta SEARCH ad AD, che dispone delle informazioni sull'ambito di ricerca, dei filtri per la ricerca e degli attributi per gli utenti filtrati.

5. AD esegue la ricerca dell'oggetto richiesto (configurato nelle impostazioni della classe oggetto) nella base di ricerca. Vengono esclusi gli oggetti che soddisfano i criteri (filtro) descritti nel messaggio di richiesta SEARCH.

6. L'AD risponde al CUAC con i risultati della ricerca.

Di seguito è riportata un'acquisizione dello sniffer che illustra i seguenti passaggi:

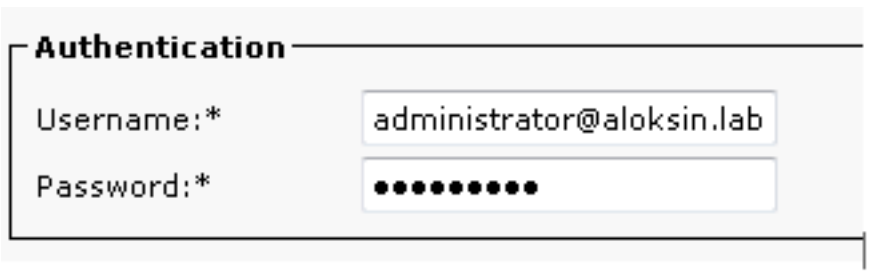


Dettagli processo LDAP

Una volta completata la configurazione di CUAC e riavviato il plugin LDAP, il server CUAC imposta una sessione TCP con AD.

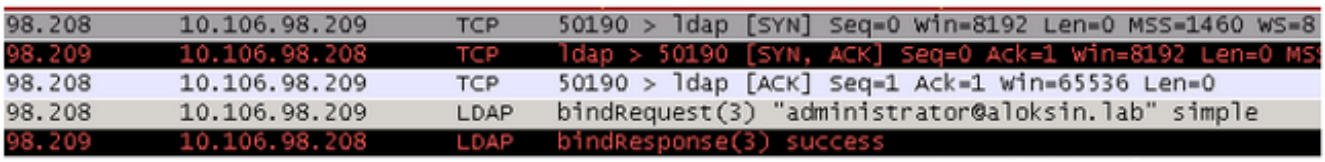
Il CUAC invia quindi una richiesta BIND per eseguire l'autenticazione con il server AD. Se l'autenticazione ha esito positivo, AD invia una risposta BIND riuscita al CUAC. Con questo, entrambi i server tentano di impostare una sessione sulla porta 389 per sincronizzare gli utenti e le loro informazioni.

Di seguito è riportata la configurazione sul server che definisce il nome distinto, utilizzato per l'autenticazione nella transazione BIND:

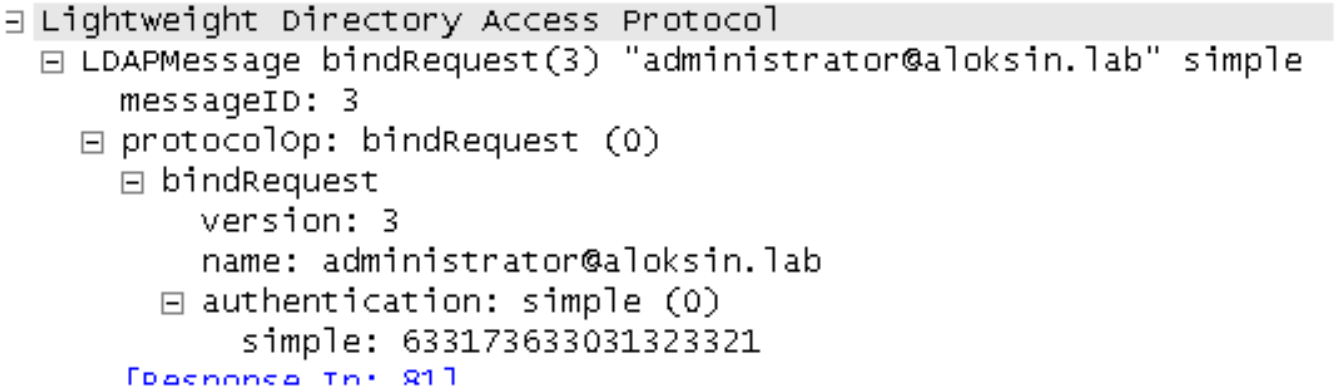


Questi messaggi vengono visualizzati nelle acquisizioni del pacchetto:

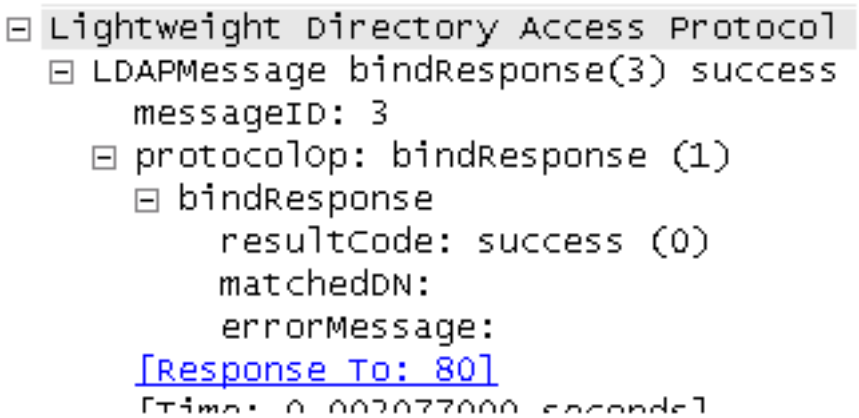
- Di seguito viene riportato l'handshake TCP seguito dalla richiesta BIND:



- Di seguito è riportata l'espansione della richiesta BIND:



- Di seguito è riportata l'espansione della risposta BIND, che indica la riuscita dell'autenticazione dell'utente (**amministratore** in questo esempio):



Se il binding ha esito positivo, il server invia una richiesta SEARCH ad Active Directory per importare gli utenti. Questa richiesta SEARCH contiene il filtro e gli attributi utilizzati da AD. L'AD cerca quindi gli utenti all'interno della base di ricerca definita (come indicato nel messaggio di richiesta SEARCH), che soddisfa i criteri del filtro e la verifica degli attributi.

Di seguito è riportato un esempio della richiesta SEARCH inviata dal CUCM:

```
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>
```

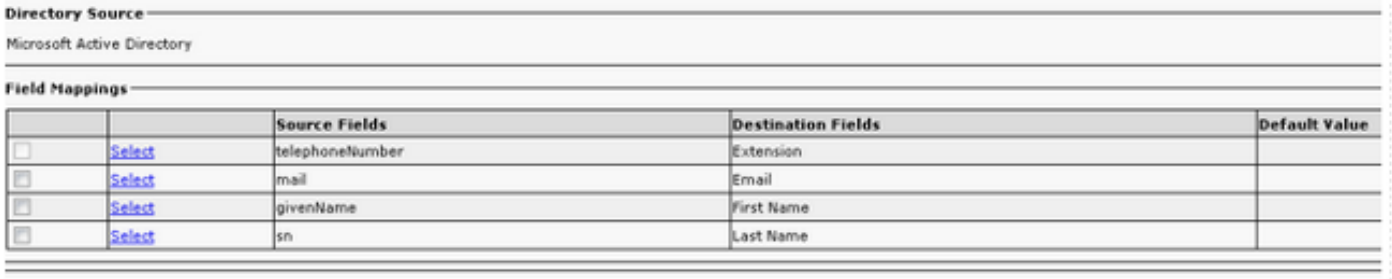
Quando l'AD riceve questa richiesta dal CUCM, cerca gli utenti nell'**oggetto base: dc=aloksin,dc=lab**, che soddisfa il filtro. Gli utenti che non soddisfano i requisiti specificati dal filtro vengono esclusi. L'AD risponde al CUCM con tutti gli utenti filtrati e invia i valori per gli attributi richiesti.

> **Nota:** Impossibile importare gli oggetti. Vengono importati solo *gli utenti*. Ciò si verifica perché il filtro inviato nel messaggio di richiesta SEARCH include **objectclass=user**.

Pertanto, la ricerca verrà eseguita solo per gli utenti e non per i contatti. CUCM dispone di tutte queste mappature e di un filtro per impostazione predefinita.

CUAC non è configurato per impostazione predefinita; non sono stati configurati dettagli di mapping per importare gli attributi per gli utenti, pertanto è necessario immettere tali dettagli manualmente. Per creare questi mapping, passare a **Configurazione di sistema > Gestione origine directory > Active Directory > Mapping campi directory**.

Gli amministratori possono mappare i campi in base ai propri requisiti. Di seguito è riportato un esempio:

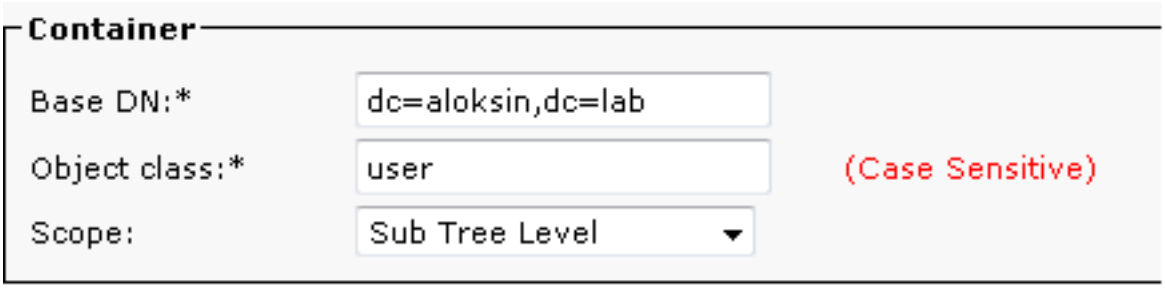


Le informazioni del campo Origine vengono inviate ad AD nel messaggio di richiesta SEARCH. Quando AD invia il messaggio di risposta SEARCH, questi valori vengono memorizzati nei campi di destinazione in CUACPE.

Per impostazione predefinita, in CUAC la classe oggetto è impostata su *contatti*. Se si utilizza questa impostazione predefinita, il filtro inviato all'AD viene visualizzato come illustrato di seguito:

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

Con questo filtro, l'AD non restituisce mai alcun utente a CUACPE, poiché cerca *contatti* nella base di ricerca, non *utenti*. Per questo motivo, è necessario impostare Classe oggetto su **utente**:



Fino a questo punto, queste impostazioni sono state configurate su CUAC:

- Dettagli connessioni
- Autenticazione (utente distinto per l'associazione)
- Impostazioni contenitore
- Mapping directory

In questo esempio, la proprietà Unique è configurata come **sAMAccountName**. Se si riavvia il plugin LDAP in CUAC e si controlla il messaggio di richiesta SEARCH, non vi saranno attributi o filtri ad eccezione di **ObjectClass=user**:

```
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]
```

Si noti che la regola di directory non è presente in questo punto. Per sincronizzare i contatti con Active Directory, è necessario creare una regola. Per impostazione predefinita, non è configurata alcuna regola di directory. Non appena ne viene creato uno, è già presente un filtro. Non è necessario modificare il filtro in quanto è necessario importare tutti gli utenti con un numero di telefono.

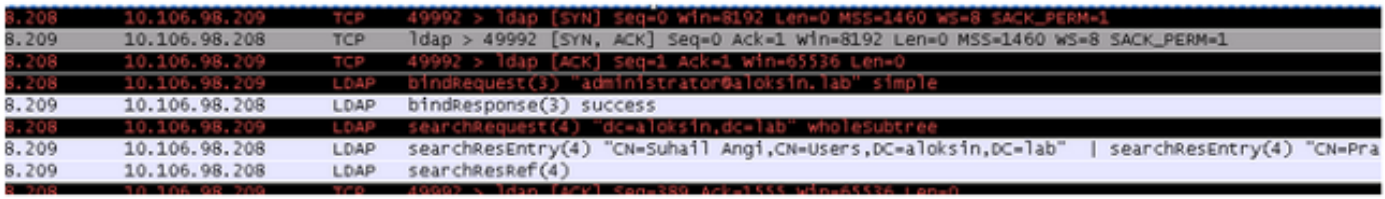
Field	Operator	Val
telephoneNumber	=	*

Riavviare il plug-in LDAP per avviare una sincronizzazione con AD e importare gli utenti. Ecco la richiesta SEARCH del CUAC:

```
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>
```

Se vengono individuati utenti che soddisfano i criteri specificati nel messaggio di richiesta SEARCH, AD invierà un messaggio *SearchResEntry* contenente le informazioni sull'utente.



Di seguito è riportato il messaggio SearchResEntry:

```
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}.........
                           .....
```

Nota: Nella risposta non è presente un indirizzo di posta elettronica, anche se questo attributo è richiesto. L'ID di posta elettronica non è stato configurato per gli utenti in Active Directory.

Una volta ricevuti, questi valori vengono memorizzati nella tabella SQL (Structured Query Language). È quindi possibile accedere alla console, che recupera l'elenco degli utenti dalla tabella SQL nel server CUACPE.