

# Integração CUAC com Microsoft AD

## Índice

[Introdução](#)

[Pré-requisitos](#)

[Requisitos](#)

[Componentes Utilizados](#)

[Informações de Apoio](#)

[Integre o AD com CUAC e importe usuários do AD](#)

[Funcionalidade LDAP entre CUAC e AD](#)

[Sumário do processo LDAP](#)

[Detalhes de processo LDAP](#)

## Introdução

Este documento descreve a maneira em que o Lightweight Directory Access Protocol (LDAP) trabalha entre o console de atendimento unificado Cisco (CUAC) e o microsoft active directory (AD) e os procedimentos que são usadas a fim integrar os dois sistemas.

## Pré-requisitos

### Requisitos

A Cisco recomenda que você tenha conhecimento destes tópicos:

- CUCM
- CUAC
- LDAP
- AD

### [Componentes Utilizados](#)

A informação neste documento é baseada na versão 10.x CUAC.

As informações neste documento foram criadas a partir de dispositivos em um ambiente de laboratório específico. Todos os dispositivos utilizados neste documento foram iniciados com uma configuração (padrão) inicial. Se a sua rede estiver ativa, certifique-se de que entende o impacto potencial de qualquer comando.

## Informações de Apoio

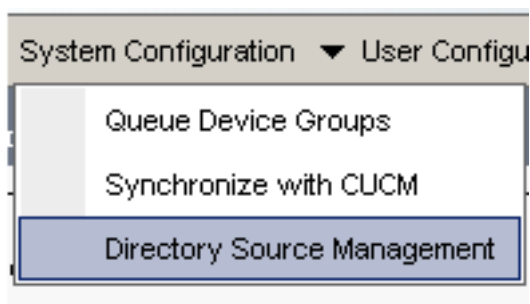
Em umas versões mais adiantadas CUAC, o server obtém usuários diretamente do gerente das comunicações unificadas de Cisco (CUCM) através das perguntas predefinidas e dos filtros. Com a edição superior CUAC (CUACPE), são permitidos aos administradores integrar e importar usuários diretamente do AD. Isto concede a flexibilidade aos administradores para a aplicação dos atributos e filtros de suas próprias escolha e exigências.

Nota: O CUACPE tem sido substituído agora com a edição avançada CUAC para versões 10 e mais recente.

## Integre o AD com CUAC e importe usuários do AD

Termine estas etapas a fim integrar o CUAC com o AD e importar usuários do AD:

1. Permita a sincronização do diretório para o AD no CUAC.



2. Selecione o **microsoft active directory** e verifique a caixa de verificação da **sincronização da possibilidade**:


**- Directory Sources**

	Source Name
<a href="#">Select</a>	CCMSource
<a href="#">Select</a>	Microsoft Active Directory
<a href="#">Select</a>	iPlanet

**General**

Source name:\*

Directory platform: Microsoft Active Directory

Enable synchronization 

3. Entre os detalhes de configuração para o servidor ativo directory:

**Connection**

Host name or IP:\*

Host port:\*  (0-65)

Use SSL

Para este exemplo, **administrator@aloksin.lab** é usado para a autenticação:

**Authentication**

Username:\*

Password:\*

4. Na propriedade os ajustes seccionam, incorporam os detalhes de configuração para a propriedade original, que aparece uma vez você incorpora os outros detalhes e clica a **salv guarda**.

**Property Settings**

Unique property:  ▼

Native property

Nota: Este é um valor exclusivo para cada entrada no AD. Se há uns valores duplicados, o CUAC puxa somente uma entrada.

5. Na seção do recipiente, incorpore os detalhes de configuração para a base DN, que é o espaço da pesquisa de usuário no AD.

O campo de *classe do objeto* é usado pelo AD a fim determinar o espaço pedido da busca. À revelia, é ajustado *para contactar*, assim que significa que o AD procura *contatos* (não usuários) na base pedida da busca. A fim importar *usuários no CUAC*, mude a classe de objeto que ajusta-se ao **usuário**:

**- Container**

Base DN:\*

Object class:\*  (Case

Scope:  ▼

6. Salvar os ajustes, clique **mapeamentos do campo do diretório**, e configurar todos os atributos que você gostaria de importar para todo o usuário. Está aqui a configuração que é usada neste exemplo:

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

7. Navegue à página da fonte do diretório e clique **regras do diretório**:


iner

DN:\*

class:\*  (Case Sensitive)

▼

---



8. O clique **adiciona novo** e cria uma regra. Quando você adiciona uma regra do diretório, um filtro da regra aparece à revelia.


Field	Operator	Value
telephoneNumber	=	*

Nota: Não há nenhuma necessidade de mudar o filtro da regra. Importa todos os usuários que têm um número de telefone configurado.

9. A fim configurar a auto-sincronização com o AD, clique a aba da **sincronização do diretório**.

▼

---



10. A configuração está agora completa. Navegue ao **planejamento > Gerenciamento do serviço** e reinicie o LDAP de encaixe a fim começar manualmente a sincronização.

## Funcionalidade LDAP entre CUAC e AD

### Sumário do processo LDAP

Está aqui um sumário do processo LDAP entre o CUAC e o AD:

1. Uma sessão de TCP é estabelecida entre os dois server (CUAC e AD).
2. O CUAC envia um pedido do LIGAMENTO ao AD e autentica-o através do usuário que é configurado nos ajustes da autenticação.
3. Uma vez que o AD autentica com sucesso o usuário, envia uma notificação do sucesso do LIGAMENTO ao CUACPE.
4. O CUAC envia uma solicitação de pesquisa ao AD, que tem a informação de escopo da busca, filtros para a busca, e a atribui para todo o usuário filtrado.
5. O AD faz a varredura para o objeto solicitado (configurado nos ajustes da classe de objeto) na base da busca. Filtra para fora os objetos que combinam os critérios (filtro) detalhados na mensagem da solicitação de pesquisa.
6. O AD responde ao CUAC com os resultados da busca.

Está aqui uma captação do sniffer que ilustre estas etapas:

```
3.208 10.106.98.209 TCP 49992 > ldap [SYN] Seq=0 win=8192 Len=0 MSS=1460 WS=
3.209 10.106.98.208 TCP ldap > 49992 [SYN, ACK] Seq=0 Ack=1 win=8192 Len=0 M
3.208 10.106.98.209 TCP 49992 > ldap [ACK] Seq=1 Ack=1 win=65536 Len=0
3.208 10.106.98.209 LDAP bindRequest(3) "administrator@aloksin.lab" simple
3.209 10.106.98.208 LDAP bindResponse(3) success
3.208 10.106.98.209 LDAP searchRequest(4) "dc=aloksin,dc=lab" wholeSubtree
3.209 10.106.98.208 LDAP searchResEntry(4) "CN=Suhail Angi,CN=Users,DC=aloksi
```

## Detalhes de processo LDAP

A configuração no CUAC é terminada uma vez e o LDAP de encaixe é reiniciado, os conjuntos de servidor CUAC - acima de uma sessão de TCP com o AD.

O CUAC envia então um pedido do LIGAMENTO a fim autenticar com o server AD. Se a autenticação é bem sucedida, o AD envia uma resposta do sucesso do LIGAMENTO ao CUAC. Com isto, ambos os server tentam estabelecer uma sessão em usuários da sincronização da porta 389 e em sua informação.

Está aqui a configuração no server que define o nome destacado, que é usado para a autenticação na transação do LIGAMENTO:

**Authentication**  
Username:\* administrator@aloksin.lab  
Password:\* ●●●●●●●●

Estas mensagens aparecem nas capturas de pacote de informação:

- Está aqui o cumprimento de TCP, seguido pelo pedido do LIGAMENTO:

```

98.208 10.106.98.209 TCP 50190 > ldap [SYN] Seq=0 win=8192 Len=0 MSS=1460 WS=8
98.209 10.106.98.208 TCP ldap > 50190 [SYN, ACK] Seq=0 Ack=1 win=8192 Len=0 MS
98.208 10.106.98.209 TCP 50190 > ldap [ACK] Seq=1 Ack=1 win=65536 Len=0
98.208 10.106.98.209 LDAP bindRequest(3) "administrator@aloksin.lab" simple
98.209 10.106.98.208 LDAP bindResponse(3) success

```

- Está aqui a expansão do pedido do LIGAMENTO:

```

Lightweight Directory Access Protocol
  LDAPMessage bindRequest(3) "administrator@aloksin.lab" simple
    messageID: 3
    protocolOp: bindRequest (0)
      bindRequest
        version: 3
        name: administrator@aloksin.lab
      authentication: simple (0)
        simple: 633173633031323321
    [Response To: 81]

```

- Está aqui a expansão da resposta do LIGAMENTO, que indica a autenticação bem sucedida do usuário (**administrador** neste exemplo):

```

Lightweight Directory Access Protocol
  LDAPMessage bindResponse(3) success
    messageID: 3
    protocolOp: bindResponse (1)
      bindResponse
        resultCode: success (0)
        matchedDN:
        errorMessage:
    [Response To: 80]
    [Time: 0.002077000 seconds]

```

Em cima de um ligamento bem sucedido, o server envia uma solicitação de pesquisa ao AD a fim importar usuários. Esta solicitação de pesquisa contém o filtro e os atributos que são usados pelo AD. O AD procura então por usuários dentro da base definida da busca (como detalhado na mensagem da solicitação de pesquisa), que cumpre os critérios no filtro e na verificação dos atributos.

Está aqui um exemplo da solicitação de pesquisa que é enviada pelo 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 o AD recebe este pedido do CUCM, procura por usuários no **baseObject: dc=aloksin, dc=lab**, que satisfaz o filtro. Todo o usuário que não cumprir as exigências que são detalhadas pelo filtro é deixado para fora. O AD responde ao CUCM com todos os usuários filtrados e envia os valores para os atributos pedidos.

Nota: Os objetos não podem ser importados. Somente os *usuários* são importados. Isto é porque o filtro que é enviado na mensagem da solicitação de pesquisa inclui o

**objectclass=user**. Daqui, o AD procura somente por usuários, não contatos. O CUCM tem todo estes mapeamentos e filtro à revelia.

O CUAC não é configurado à revelia; não há nenhum mapeamento detalha configurado a fim importar atributos para usuários, assim que você deve entrar estes detalhes manualmente. A fim criar estes mapeamentos, navegue à **configuração de sistema > ao gerenciamento de origem do diretório > ao mapeamento do campo do diretório ativo > do diretório**.

São permitidos aos administradores traçar campos por suas próprias exigências. Aqui está um exemplo:

Directory Source				
Microsoft Active Directory				
Field Mappings				
		Source Fields	Destination Fields	Default Value
<input type="checkbox"/>	Select	telephonenumber	Extension	
<input type="checkbox"/>	Select	mail	Email	
<input type="checkbox"/>	Select	givenName	First Name	
<input type="checkbox"/>	Select	sn	Last Name	

A informação do campo de fonte é enviada ao AD na mensagem da solicitação de pesquisa. Quando o AD envia o mensagem de resposta da BUSCA, estes valores estão armazenados nos campos de destino no CUACPE.

Note que o CUAC tem à revelia a classe de objeto ajustada aos *contatos*. Se esta configuração padrão é usada, o filtro que é enviado ao AD aparece como mostrado aqui:

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

Com este filtro, o AD nunca retorna todos os usuários ao CUACPE, desde que procura por *contatos na base da busca, não usuários*. Por este motivo, você deve mudar a classe de objeto ao **usuário**:

Container	
Base DN:*	<input type="text" value="dc=aloksin,dc=lab"/>
Object class:*	<input type="text" value="user"/> (Case Sensitive)
Scope:	<input type="text" value="Sub Tree Level"/> ▼

Até este ponto, estes ajustes foram configurados no CUAC:

- Detalhes das conexões
- Autenticação (distinto usuário para ligar)
- Ajustes do recipiente
- Mapeamento do diretório

Neste exemplo, a propriedade original é configurada como o **sAMAccountName**. Se você reinicia o LDAP de encaixe no CUAC e verifica a mensagem da solicitação de pesquisa, não contém nenhum atributo ou filtro exceto o **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]

```

Note que a regra do diretório falta aqui. Sincronização os contatos com o AD, você deve criar uma regra. À revelia, não há nenhuma regra do diretório configurada. Assim que um for criado, um filtro está já atual. Não há nenhuma necessidade de mudar o filtro, como você deve importar todos os usuários que têm um número de telefone.

Field	Operator	Value
telephoneNumber	=	*

Reinicie o LDAP de encaixe a fim iniciar uma sincronização com o AD e importar os usuários. Está aqui a solicitação de pesquisa do 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 o AD encontra os usuários que combinam os critérios detalhados na mensagem da solicitação de pesquisa, a seguir envia uma mensagem de *SearchResEntry* que contenha a informação sobre o usuário.

```
8.208 10.106.98.209 TCP 49992 > 1dap [SYN] Seq=0 Win=8192 Len=0 MSS=1460 WS=8 SACK_PERM=1
8.209 10.106.98.208 TCP 1dap > 49992 [SYN, ACK] Seq=0 Ack=1 Win=8192 Len=0 MSS=1460 WS=8 SACK_PERM=1
8.208 10.106.98.209 TCP 49992 > 1dap [ACK] Seq=1 Ack=1 Win=65536 Len=0
8.208 10.106.98.209 LDAP bindRequest(3) "administrator@aloksin.lab" simple
8.209 10.106.98.208 LDAP bindResponse(3) success
8.208 10.106.98.209 LDAP searchRequest(4) "dc=aloksin,dc=lab" wholeSubtree
8.209 10.106.98.208 LDAP searchResEntry(4) "CN=Suhail Angi,CN=Users,DC=aloksin,DC=lab" | searchResEntry(4) "CN=Pra
8.209 10.106.98.208 LDAP searchResRef(4)
8.208 10.106.98.209 TCP 49992 > 1dap [ACK] Seq=389 Ack=1555 Win=65536 Len=0
```

Está aqui a mensagem de 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: Não há nenhum CORREIO na resposta, mesmo que este atributo seja pedido. Isto é porque o CORREIO ID não foi configurado para usuários no AD.

Uma vez que estes valores são recebidos pelo CUAC, armazena-o na tabela da língua de consulta estruturada (SQL). Você pode então registrar no console, e o console busca a lista de usuários desta tabela SQL no server CUACPE.