

# Exemple de configuration de ASA Anyconnect VPN et OpenLDAP Authorization avec schéma et certificats personnalisés

## Contenu

[Introduction](#)

[Conditions préalables](#)

[Conditions requises](#)

[Components Used](#)

[Configuration](#)

[Configuration OpenLDAP de base](#)

[Schéma Openldap personnalisé](#)

[Configuration ASA](#)

[Vérification](#)

[Tester l'accès VPN](#)

[Débogages](#)

[Authentification et autorisation séparées ASA](#)

[Attributs ASA de LDAP et de groupe local](#)

[ASA et LDAP avec authentification de certificat](#)

[Débogages](#)

[Authentification secondaire](#)

[Informations connexes](#)

## Introduction

Ce document décrit comment configurer OpenLDAP avec un schéma personnalisé pour prendre en charge les attributs par utilisateur pour Cisco Anyconnect Secure Mobility Client qui se connecte à un appareil de sécurité adaptatif Cisco (ASA). La configuration ASA est assez basique car tous les attributs utilisateur sont récupérés à partir du serveur OpenLDAP. Ce document décrit également les différences dans l'authentification et l'autorisation LDAP lorsqu'elles sont utilisées avec les certificats.

## Conditions préalables

### Conditions requises

Cisco vous recommande de prendre connaissance des rubriques suivantes :

- Connaissances de base sur la configuration Linux
- Connaissances de base sur la configuration de l'interface de ligne de commande ASA

### Components Used

Les informations contenues dans ce document sont basées sur les versions de logiciel suivantes :

- Cisco ASA version 8.4 et ultérieure
- OpenLDAP version 2.4.30

## Configuration

### Configuration OpenLDAP de base

#### Étape 1. Configurez le serveur.

Cet exemple utilise l'arborescence ldap test-cisco.com.

ldap.conf est utilisé pour définir les valeurs par défaut au niveau du système qui peuvent être utilisées par le client ldap local.

**Note:** Bien que vous ne soyez pas tenu de configurer les paramètres par défaut au niveau du système, ils peuvent vous aider à tester et dépanner le serveur lorsque vous exécutez un client ldap local.

/etc/openldap/ldap.conf:

```
BASE dc=test-cisco,dc=com
```

le fichier slapd.conf est utilisé pour la configuration du serveur OpenLDAP. Les fichiers de schéma par défaut incluent des définitions LDAP largement utilisées. Par exemple, la *personne* de nom de classe d'objet est définie dans le fichier core.schema. Cette configuration utilise ce schéma commun et définit son propre schéma pour les attributs propres à Cisco.

/etc/openldap/slapd.conf:

```
include /etc/openldap/schema/core.schema
include /etc/openldap/schema/cosine.schema
include /etc/openldap/schema/inetorgperson.schema
include /etc/openldap/schema/openldap.schema
include /etc/openldap/schema/nis.schema

# Defines backend database type and redirects all # queries with specified suffix to that
database
database hdb
suffix "dc=test-cisco,dc=com"
checkpoint 32 30

# Rootdn will be used to perform all administrative tasks.
rootdn "cn=Manager,dc=test-cisco,dc=com"

# Cleartext passwords, especially for the rootdn, should be avoid.
rootpw secret

directory /var/lib/openldap-data
index objectClass eq
```

#### Étape 2. Vérifiez la configuration LDAP.

Afin de vérifier que OpenLDAP de base fonctionne, exécutez cette configuration :

```
pluton openldap # /etc/init.d/slapd start
* Starting ldap-server [ ok ]
pluton openldap # ps ax | grep openldap
27562 ? Ssl 0:00 /usr/lib64/openldap/slapd -u ldap -g ldap -f
/etc/openldap/slapd.conf -h ldaps:// ldap:// ldapi://var/run/openldap/slapd.sock

pluton openldap # netstat -atcpn | grep slapd
Active Internet connections (servers and established)
Proto Recv-Q Send-Q Local Address Foreign Address State PID/Program name
tcp 0 0 0.0.0.0:636 0.0.0.0:* LISTEN 27562/slapd
tcp 0 0 0.0.0.0:389 0.0.0.0:* LISTEN 27562/slapd

pluton # ldapsearch -h 192.168.10.1 -D "CN=Manager,DC=test-cisco,DC=com" -w secret
# extended LDIF
#
# LDAPv3
# base <dc=test-cisco,dc=com> (default) with scope subtree
# filter: (objectclass=*)
# requesting: ALL
#
# search result
search: 2
result: 32 No such object

# numResponses: 1
```

### Étape 3. Ajouter des enregistrements à la base de données.

Une fois que vous avez testé et configuré correctement tout, ajoutez des enregistrements à la base de données. Afin d'ajouter des conteneurs de base pour les utilisateurs et les groupes, exécutez cette configuration :

```
pluton # cat root.ldiff
dn: dc=test-cisco,dc=com
objectclass: dcObject
objectclass: organization
o: test-cisco.com
dc: test-cisco

dn: ou=People,dc=test-cisco,dc=com
objectClass: organizationalUnit
objectClass: top
ou: People

dn: ou=Groups,dc=test-cisco,dc=com
objectClass: organizationalUnit
objectClass: top
ou: Groups

pluton # ldapadd -h 192.168.10.1 -D "CN=Manager,DC=test-cisco,DC=com"
-w secret -x -f root.ldiff
adding new entry "dc=test-cisco,dc=com"
adding new entry "ou=People,dc=test-cisco,dc=com"
adding new entry "ou=Groups,dc=test-cisco,dc=com"

pluton # ldapsearch -h 192.168.10.1 -D "CN=Manager,DC=test-cisco,DC=com" -w secret
# extended LDIF
#
```

```

# LDAPv3
# base <dc=test-cisco,dc=com> (default) with scope subtree
# filter: (objectclass=*)
# requesting: ALL
#
# test-cisco.com
dn: dc=test-cisco,dc=com
objectClass: dcObject
objectClass: organization
o: test-cisco.com
dc: test-cisco

# People, test-cisco.com
dn: ou=People,dc=test-cisco,dc=com
objectClass: organizationalUnit
objectClass: top
ou: People

# Groups, test-cisco.com
dn: ou=Groups,dc=test-cisco,dc=com
objectClass: organizationalUnit
objectClass: top
ou: Groups

# search result
search: 2
result: 0 Success

# numResponses: 4
# numEntries: 3

```

## Schéma Openldap personnalisé

Maintenant que la configuration de base fonctionne, vous pouvez ajouter un schéma personnalisé. Dans cet exemple de configuration, un nouveau type de classe d'objet nommé *CiscoPerson* est créé et ces attributs sont créés et utilisés dans cette classe d'objet :

- Bannière Cisco
- CiscoACLin
- DomaineCisco
- CiscoDNS
- AdresselPisco
- MasqueRéseauCisco
- CiscoSplitACL
- PolitiqueTunnelPartagéCisco
- StratégieGroupeCisco

### Étape 1. Créez le nouveau schéma dans cisco.schema.

```

pluton openldap # pwd
/etc/openldap
pluton openldap # cat schema/cisco.schema

attributetype ( 1.3.6.1.4.1.9.500.1.1
  NAME 'CiscoBanner'
  DESC 'Banner Name for VPN users'
  EQUALITY caseIgnoreMatch
  SUBSTR caseIgnoreSubstringsMatch

```

```
ORDERING caseIgnoreOrderingMatch
SYNTAX 1.3.6.1.4.1.1466.115.121.1.15{128}
SINGLE-VALUE )
```

```
attributetype ( 1.3.6.1.4.1.9.500.1.2
NAME 'CiscoACLin'
DESC 'ACL in for VPN users'
EQUALITY caseIgnoreMatch
SUBSTR caseIgnoreSubstringsMatch
ORDERING caseIgnoreOrderingMatch
SYNTAX 1.3.6.1.4.1.1466.115.121.1.15{128}
SINGLE-VALUE )
```

```
attributetype ( 1.3.6.1.4.1.9.500.1.3
NAME 'CiscoDomain'
DESC 'Domain for VPN users'
EQUALITY caseIgnoreMatch
SUBSTR caseIgnoreSubstringsMatch
ORDERING caseIgnoreOrderingMatch
SYNTAX 1.3.6.1.4.1.1466.115.121.1.15{128}
SINGLE-VALUE )
```

```
attributetype ( 1.3.6.1.4.1.9.500.1.4
NAME 'CiscoDNS'
DESC 'DNS server for VPN users'
EQUALITY caseIgnoreMatch
SUBSTR caseIgnoreSubstringsMatch
ORDERING caseIgnoreOrderingMatch
SYNTAX 1.3.6.1.4.1.1466.115.121.1.15{128}
SINGLE-VALUE )
```

```
attributetype ( 1.3.6.1.4.1.9.500.1.5
NAME 'CiscoIPAddress'
DESC 'Address for VPN user'
EQUALITY caseIgnoreMatch
SUBSTR caseIgnoreSubstringsMatch
ORDERING caseIgnoreOrderingMatch
SYNTAX 1.3.6.1.4.1.1466.115.121.1.15{128}
SINGLE-VALUE )
```

```
attributetype ( 1.3.6.1.4.1.9.500.1.6
NAME 'CiscoIPNetmask'
DESC 'Address for VPN user'
EQUALITY caseIgnoreMatch
SUBSTR caseIgnoreSubstringsMatch
ORDERING caseIgnoreOrderingMatch
SYNTAX 1.3.6.1.4.1.1466.115.121.1.15{128}
SINGLE-VALUE )
```

```
attributetype ( 1.3.6.1.4.1.9.500.1.7
NAME 'CiscoSplitACL'
DESC 'Split tunnel list for VPN users'
EQUALITY caseIgnoreMatch
SUBSTR caseIgnoreSubstringsMatch
ORDERING caseIgnoreOrderingMatch
SYNTAX 1.3.6.1.4.1.1466.115.121.1.15{128}
SINGLE-VALUE )
```

```
attributetype ( 1.3.6.1.4.1.9.500.1.8
NAME 'CiscoSplitTunnelPolicy'
DESC 'Split tunnel policy for VPN users'
EQUALITY caseIgnoreMatch
SUBSTR caseIgnoreSubstringsMatch
ORDERING caseIgnoreOrderingMatch
```

```
SYNTAX 1.3.6.1.4.1.1466.115.121.1.15{128}
SINGLE-VALUE )
```

```
attributetype ( 1.3.6.1.4.1.9.500.1.9
NAME 'CiscoGroupPolicy'
DESC 'Group policy for VPN users'
EQUALITY caseIgnoreMatch
SUBSTR caseIgnoreSubstringsMatch
ORDERING caseIgnoreOrderingMatch
SYNTAX 1.3.6.1.4.1.1466.115.121.1.15{128}
SINGLE-VALUE )
```

```
objectclass ( 1.3.6.1.4.1.9.500.2.1 NAME 'CiscoPerson'
DESC 'My cisco person'
AUXILIARY
MUST ( sn $ cn )
MAY ( userPassword $ telephoneNumber $ seeAlso
$ description $ CiscoBanner $ CiscoACLin $ CiscoDomain
$ CiscoDNS $ CiscoIPAddress $ CiscoIPNetmask $ CiscoSplitACL
$ CiscoSplitTunnelPolicy $ CiscoGroupPolicy ) )
```

## Remarques importantes

- Utilisez des OID d'entreprise privée pour votre entreprise. Tous les OID fonctionneront, mais la meilleure pratique consiste à utiliser les OID attribués par l'IANA. Celui configuré dans ces exemples commence à 1.3.6.1.4.1.9 (réservé par Cisco : <http://www.iana.org/assignments/enterprise-numbers>).
- La partie suivante de l'OID (500.1.1-500.1.9) a été utilisée pour ne pas interférer directement dans l'arborescence principale de l'OID Cisco (« 1.3.6.1.4.1.9 »).
- Cette base de données utilise la classe d'objet *Personne* définie dans `schema/core.ldif`. Cet objet est de type TOP et les enregistrements ne peuvent inclure qu'un seul attribut de ce type (c'est pourquoi la classe d'objet *CiscoPerson* est de type Auxiliaire).
- La classe d'objet nommée *CiscoPerson* doit inclure SN ou CN et peut inclure n'importe lequel des attributs Cisco personnalisés définis précédemment. Notez qu'il peut également inclure tous les autres attributs définis dans d'autres schémas (tels que *userPassword* ou *phoneNumber*).
- N'oubliez pas que chaque objet doit avoir un numéro OID différent.
- Les attributs personnalisés ne respectent pas la casse et sont de type *chaîne* avec le codage UTF-8 et un maximum de 128 caractères (défini par SYNTAX).

## Étape 2. Inclure le schéma dans `slapd.conf`.

```
pluton openldap # cat slapd.conf | grep include
include         /etc/openldap/schema/core.schema
include         /etc/openldap/schema/cosine.schema
include         /etc/openldap/schema/inetorgperson.schema
include         /etc/openldap/schema/openldap.schema
include         /etc/openldap/schema/nis.schema
include        /etc/openldap/schema/cisco.schema
```

## Étape 3. Redémarrer les services.

```
puton openldap # /etc/init.d/slapd restart
* Stopping ldap-server [ ok ]
```

```
* Starting ldap-server
```

```
[ ok ]
```

## Étape 4. Ajoutez un nouvel utilisateur avec tous les attributs personnalisés.

Dans cet exemple, l'utilisateur appartient à plusieurs objets objectClass et hérite de tous leurs attributs. Avec ce processus, il est facile d'ajouter un schéma ou des attributs supplémentaires sans modifier les enregistrements de base de données existants.

```
pluton # cat users.ldiff
# User account
dn: uid=cisco,ou=people,dc=test-cisco,dc=com
cn: John Smith
givenName: John
sn: cisco
uid: cisco
uidNumber: 10000
gidNumber: 10000
homeDirectory: /home/cisco
mail: jsmith@dev.local
objectClass: top
objectClass: posixAccount
objectClass: shadowAccount
objectClass: inetOrgPerson
objectClass: organizationalPerson
objectClass: person
objectClass: CiscoPerson
loginShell: /bin/bash
userPassword: {CRYPT}*
CiscoBanner: This is banner 1
CiscoIPAddress: 10.1.1.1
CiscoIPNetmask: 255.255.255.128
CiscoDomain: domain1.com
CiscoDNS: 10.6.6.6
CiscoACLIn: ip:inacl#1=permit ip 10.1.1.0 255.255.255.128 10.11.11.0 255.255.255.0
CiscoSplitACL: ACL1
CiscoSplitTunnelPolicy: 1
CiscoGroupPolicy: POLICY1
```

```
pluton # ldapadd -h 192.168.10.1 -D "CN=Manager,DC=test-cisco,DC=com"
-w secret -x -f users.ldiff
adding new entry "uid=cisco,ou=people,dc=test-cisco,dc=com"
```

## Étape 5. Définissez le mot de passe de l'utilisateur.

```
pluton moje # ldappasswd -h 192.168.10.1 -D "CN=Manager,DC=test-cisco,DC=com"
-w secret -x uid=cisco,ou=people,dc=test-cisco,dc=com -s pass1
```

## Étape 6. Vérifier la configuration

```
pluton # ldapsearch -h 192.168.10.1 -D "CN=Manager,DC=test-cisco,DC=com"
-w secret -b uid=cisco,ou=people,dc=test-cisco,dc=com
# extended LDIF
#
# LDAPv3
# base <uid=cisco,ou=people,dc=test-cisco,dc=com> with scope subtree
# filter: (objectclass=*)
# requesting: ALL
#
# cisco, People, test-cisco.com
dn: uid=cisco,ou=People,dc=test-cisco,dc=com
```

```
cn: John Smith
givenName: John
sn: cisco
uid: cisco
uidNumber: 10000
gidNumber: 10000
homeDirectory: /home/cisco
mail: jsmith@dev.local
objectClass: top
objectClass: posixAccount
objectClass: shadowAccount
objectClass: inetOrgPerson
objectClass: organizationalPerson
objectClass: person
objectClass: CiscoPerson
loginShell: /bin/bash
userPassword:: e0NSWVBuFSo=
CiscoBanner: This is banner 1
CiscoIPAddress: 10.1.1.1
CiscoIPNetmask: 255.255.255.128
CiscoDomain: domain1.com
CiscoDNS: 10.6.6.6
CiscoACLin: ip:inacl#1=permit ip 10.1.1.0 255.255.255.128 10.11.11.0 255.255.255.
0
CiscoSplitACL: ACL1
CiscoSplitTunnelPolicy: 1
CiscoGroupPolicy: POLICY1
userPassword:: e1NTSEF9NXM4MUZtaS85YUcvV2ZQU3kzbEdtdzFPUkk0bH13V0M=

# search result
search: 2
result: 0 Success

# numResponses: 2
# numEntries: 1
```

## Configuration ASA

### Étape 1. Configurez l'interface et le certificat.

```
interface GigabitEthernet0
 nameif inside
 security-level 100
 ip address 192.168.11.250 255.255.255.0
!
interface GigabitEthernet1
 nameif outside
 security-level 0
 ip address 192.168.1.250 255.255.255.0

crypto ca trustpoint CA
 keypair CA
 crl configure
crypto ca certificate chain CA
 certificate ca 00cf946de20d0ce6d9
 30820223 3082018c 020900cf 946de20d 0ce6d930 0d06092a 864886f7 0d010105
05003056 310b3009 06035504 06130250 4c310c30 0a060355 04080c03 4d617a31
0f300d06 03550407 0c065761 72736177 310c300a 06035504 0a0c0354 4143310c
300a0603 55040b0c 03524143 310c300a 06035504 030c0354 4143301e 170d3132
31313136 30383131 32365a17 0d313331 31313630 38313132 365a3056 310b3009
06035504 06130250 4c310c30 0a060355 04080c03 4d617a31 0f300d06 03550407
0c065761 72736177 310c300a 06035504 0a0c0354 4143310c 300a0603 55040b0c
```



```
03524143 310c300a 06035504 030c0354 41433081 9f300d06 092a8648 86f70d01
01010500 03818d00 30818902 818100d0 68af1ef6 9b256071 d39c8d25 4fb9f391
5a96e8e0 1ac424d5 fc9cf460 f09e181e f1487525 d982f3ae 29384ca8 13d5290d
a360e796 0224dce5 ffc0767e 6f54b991 967b54a4 4b3aa59e c2a69310 550029fb
cb1c3f45 3fb15d15 0d507b09 52b02a17 6189d591 87d42617 1d93b683 4d685005
34788fd0 2a899ca4 926e7318 1f914102 03010001 300d0609 2a864886 f70d0101
05050003 81810046 8c58cddb dfd6932b 9260af40 ebc63465 1f18a374 f5b7865c
a21b22f3 a07ebf57 d64312b7 57543c91 edc4088d 3c7b3c75 e3f29b8d b7e04e01
4dc2cb89 6935e07c 3518ad97 96e50aae 52e89265 92bb1aad a85656dc 931e2006
af4042a0 09826d29 88ca972e 5442e0c3 8c957978 4a15e5d9 cac5a12c b0604df4
97438706 c973a5
```

quit

```
certificate 00fe9c3d61e131cd9e
```

```
30820225 3082018e 020900fe 9c3d61e1 31cd9e30 0d06092a 864886f7 0d010105
05003056 310b3009 06035504 06130250 4c310c30 0a060355 04080c03 4d617a31
0f300d06 03550407 0c065761 72736177 310c300a 06035504 0a0c0354 4143310c
300a0603 55040b0c 03524143 310c300a 06035504 030c0354 4143301e 170d3132
31313136 31303336 31325a17 0d313331 31313631 30333631 325a3058 310b3009
06035504 06130250 4c310c30 0a060355 04080c03 4d617a31 11300f06 03550407
0c085761 72737a61 7761310c 300a0603 55040a0c 03414353 310c300a 06035504
0b0c0341 4353310c 300a0603 5504030c 03414353 30819f30 0d06092a 864886f7
0d010101 05000381 8d003081 89028181 00d15ee2 0f14597a 0703204b 22a2c5cc
34c0967e 74bb087c b16bc462 d1e4f99d 3d40bd19 5b80845e 08f2cccb e2ca0d01
aa6fe4f4 df287598 45956110 d3c66465 668ae4d2 8a9583e8 7a652685 19b25dfa
fce7b84e e1780dd0 1cd3d71e 0926db1a 74354b11 c5b976e0 07e7dd01 0b4115f0
662874c3 2ed5f87e 170b3baa f266f650 2f020301 0001300d 06092a86 4886f70d
01010505 00038181 00987d8e acfa9cac ab9dbb52 5bb61992 975e4bbe e9c28426
1dc3dd1e 87abd839 fa3a937d b1aebcc4 fdc549a2 010b83f3 aa0e12b3 f03a4f49
d8e6fdea 61776ae5 17daf7e4 6baf810d 37c24784 bd71429b dc0494c0 84a020ff
1be0c903 a055f634 1e29b6ea 7d7f3280 f161a86c 50d40b6c c24bc8b0 493c0918
8a185e05 1b52d8b0 0e
```

quit

## Étape 2. Générez un certificat auto-signé.

```
crypto ca trustpoint CA
enrollment self
crypto ca enroll CA
```

## Étape 3. Activez WebVPN sur l'interface externe.

```
ssl trust-point CA
webvpn
enable outside
anyconnect image disk0:/anyconnect-win-3.1.01065-k9.pkg 1
anyconnect enable
tunnel-group-list enable
```

## Étape 4. Fractionner la configuration de la liste de contrôle d'accès.

Le nom de la liste de contrôle d'accès est retourné par OpenLDAP :

```
access-list ACL1 standard permit 10.7.7.0 255.255.255.0
```

## Étape 5. Créez un nom de groupe de tunnels qui utilise la stratégie de groupe par défaut (DfltAccessPolicy).

Les utilisateurs possédant l'attribut LDAP spécifique (*CiscoGroupPolicy*) sont mappés à une autre stratégie : POLITIQUE1

```
group-policy DfltAccessPolicy internal
group-policy DfltAccessPolicy attributes
  vpn-tunnel-protocol ikev1 ikev2 l2tp-ipsec ssl-client ssl-clientless
```

```
group-policy POLICY1 internal
group-policy POLICY1 attributes
  vpn-tunnel-protocol ikev1 ikev2 l2tp-ipsec ssl-client ssl-clientless
```

```
tunnel-group RA type remote-access
tunnel-group RA general-attributes
tunnel-group RA webvpn-attributes
  group-alias RA enable
  without-csd
```

La configuration ASA aaa-server utilise ldap attribute-map pour le mappage à partir d'attributs retournés par OpenLDAP à des attributs qui peuvent être interprétés par ASA pour les utilisateurs Anyconnect.

```
ldap attribute-map LDAP-MAP
map-name CiscoACLin Cisco-AV-Pair
map-name CiscoBanner Banner1
map-name CiscoDNS Primary-DNS
map-name CiscoDomain IPsec-Default-Domain
map-name CiscoGroupPolicy IETF-Radius-Class
map-name CiscoIPAddress IETF-Radius-Framed-IP-Address
map-name CiscoIPNetmask IETF-Radius-Framed-IP-Netmask
map-name CiscoSplitACL IPsec-Split-Tunnel-List
map-name CiscoSplitTunnelPolicy IPsec-Split-Tunneling-Policy
```

```
aaa-server LDAP protocol ldap
aaa-server LDAP (inside) host 192.168.11.10
  ldap-base-dn DC=test-cisco,DC=com
  ldap-scope subtree
  ldap-naming-attribute uid
  ldap-login-password secret
  ldap-login-dn CN=Manager,DC=test-cisco,DC=com
  server-type openldap
  ldap-attribute-map LDAP-MA
```

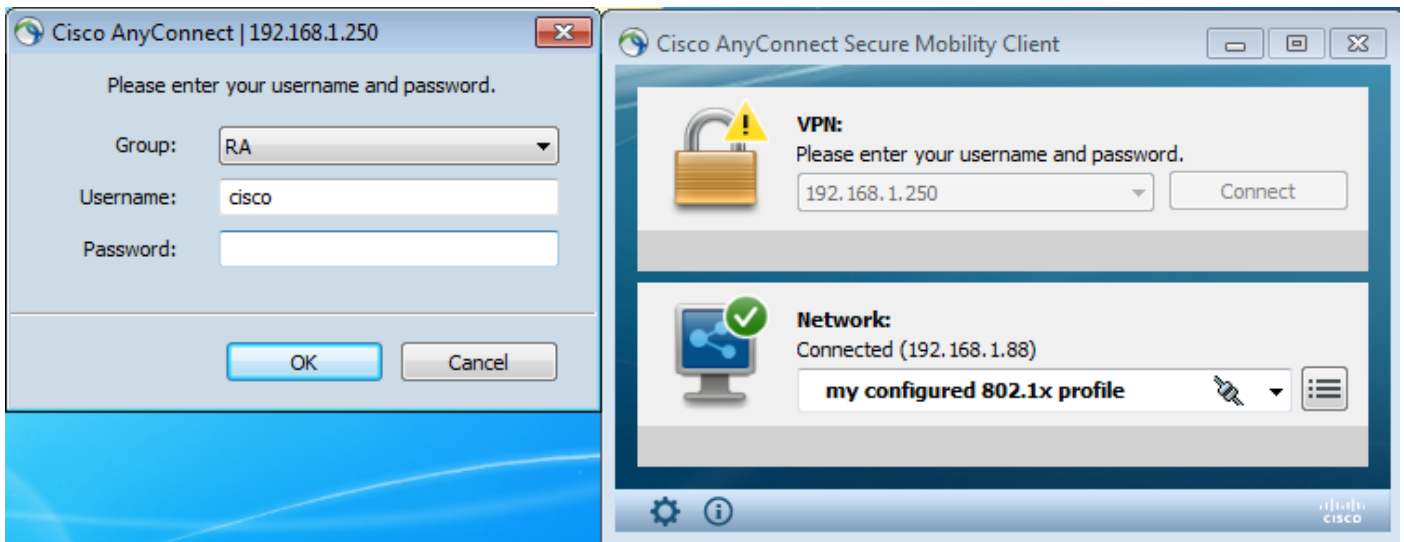
**Étape 6. Activez le serveur LDAP pour l'authentification pour le groupe de tunnels spécifié.**

```
tunnel-group RA general-attributes
  authentication-server-group LDAP
```

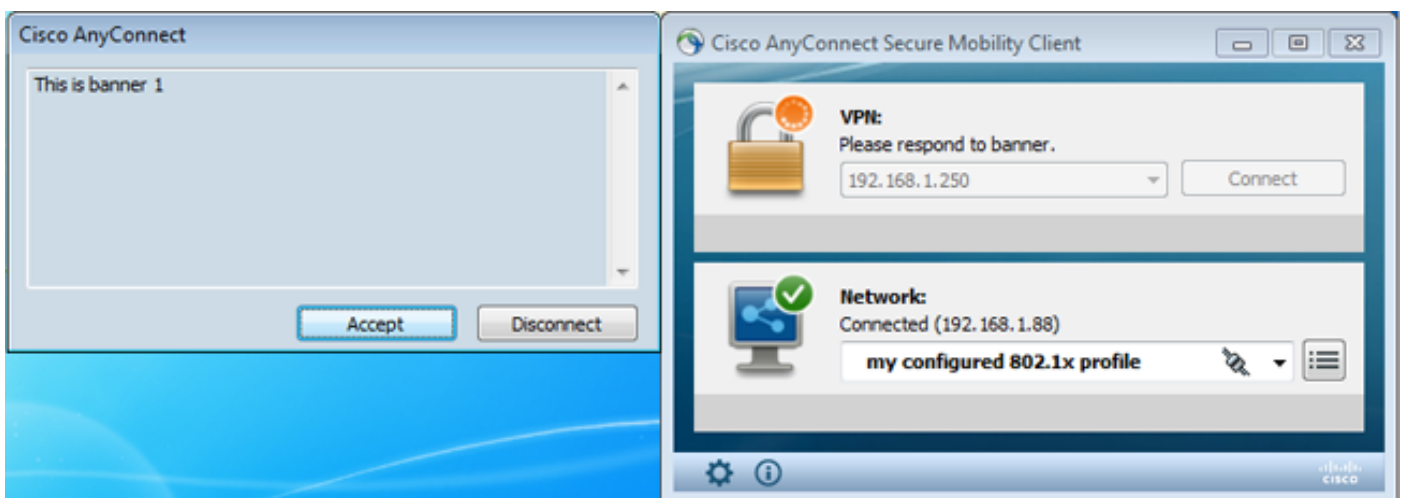
## Vérification

### Tester l'accès VPN

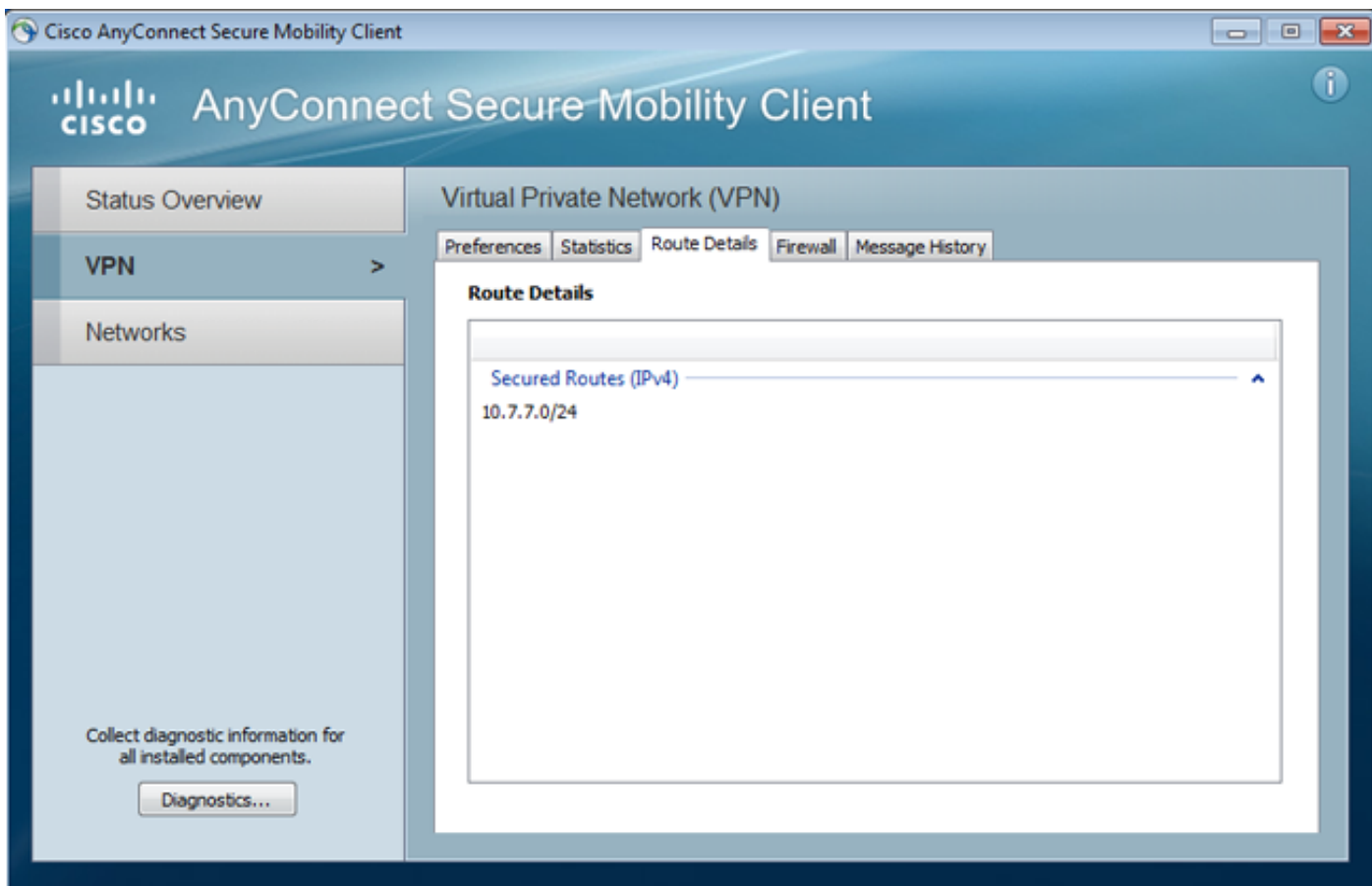
Anyconnect est configuré pour se connecter à 192.168.1.250. La connexion est le nom d'utilisateur *cisco* et le mot de passe *pass1*.



Après authentification, la bannière correcte est utilisée.



La liste de contrôle d'accès fractionnée correcte est envoyée (ACL1 définie sur ASA).



L'interface Anyconnect est configurée avec IP : 10.1.1.1 et masque de réseau 255.255.255.128. Le domaine est domain1.com et le serveur DNS est 10.6.6.6.

```

Ethernet adapter Połączenie lokalne 2:
Connection-specific DNS Suffix . . : domain1.com
Description . . . . . : Cisco AnyConnect Secure Mobility Client U
Virtual Miniport Adapter for Windows x64
Physical Address. . . . . : 00-05-9A-3C-7A-00
DHCP Enabled. . . . . : No
Autoconfiguration Enabled . . . . : Yes
Link-local IPv6 Address . . . . . : fe80::2015:d34b:e3a8:1787%14(Preferred)
Link-local IPv6 Address . . . . . : fe80::3a02:5a4a:4b9b:ddf2%14(Preferred)
Link-local IPv6 Address . . . . . : fe80::4fd8:3523:c111:ad1d%14(Preferred)
IPv4 Address. . . . . : 10.1.1.1(Preferred)
Subnet Mask . . . . . : 255.255.255.128
Default Gateway . . . . . :
DNS Servers . . . . . : 10.6.6.6
NetBIOS over Tcpip. . . . . : Enabled

```

Sur l'ASA, l'utilisateur *cisco* a reçu l'adresse IP suivante : 10.1.1.1 et est affecté à la stratégie de groupe *POLICY1*.

```
ASA# show vpn-sessiondb detail anyconnect
```

```
Session Type: AnyConnect Detailed
```

```

Username       : cisco                Index       : 29
Assigned IP    : 10.1.1.1                Public IP   : 192.168.1.88
Protocol       : AnyConnect-Parent SSL-Tunnel
License        : AnyConnect Premium
Encryption     : RC4                    Hashing     : none SHA1
Bytes Tx       : 10212                  Bytes Rx    : 856
Pkts Tx        : 8                      Pkts Rx     : 2
Pkts Tx Drop   : 0                      Pkts Rx Drop : 0
Group Policy   : POLICY1                 Tunnel Group : RA
Login Time     : 10:18:25 UTC Thu Apr 4 2013
Duration       : 0h:00m:17s
Inactivity     : 0h:00m:00s

```

NAC Result : Unknown  
VLAN Mapping : N/A VLAN : none

AnyConnect-Parent Tunnels: 1  
SSL-Tunnel Tunnels: 1

AnyConnect-Parent:

Tunnel ID : 29.1  
Public IP : 192.168.1.88  
Encryption : none TCP Src Port : 49262  
TCP Dst Port : 443 Auth Mode : userPassword  
Idle Time Out: 30 Minutes Idle TO Left : 29 Minutes  
Client Type : AnyConnect  
Client Ver : 3.1.01065  
Bytes Tx : 5106 Bytes Rx : 788  
Pkts Tx : 4 Pkts Rx : 1  
Pkts Tx Drop : 0 Pkts Rx Drop : 0

SSL-Tunnel:

Tunnel ID : 29.2  
Assigned IP : 10.1.1.1 Public IP : 192.168.1.88  
Encryption : RC4 Hashing : SHA1  
Encapsulation: TLSv1.0 TCP Src Port : 49265  
TCP Dst Port : 443 Auth Mode : userPassword  
Idle Time Out: 30 Minutes Idle TO Left : 29 Minutes  
Client Type : SSL VPN Client  
Client Ver : Cisco AnyConnect VPN Agent for Windows 3.1.01065  
Bytes Tx : 5106 Bytes Rx : 68  
Pkts Tx : 4 Pkts Rx : 1  
Pkts Tx Drop : 0 Pkts Rx Drop : 0  
**Filter Name : AAA-user-cisco-E0CF3C05**

NAC:

Reval Int (T): 0 Seconds Reval Left(T): 0 Seconds  
SQ Int (T) : 0 Seconds EoU Age(T) : 17 Seconds  
Hold Left (T): 0 Seconds Posture Token:

En outre, la liste d'accès dynamique est installée pour cet utilisateur :

ASA# **show access-list AAA-user-cisco-E0CF3C05**

```
access-list AAA-user-cisco-E0CF3C05; 1 elements; name hash: 0xf9b6b75c (dynamic)
access-list AAA-user-cisco-E0CF3C05 line 1 extended permit
ip 10.1.1.0 255.255.255.128 10.11.11.0 255.255.255.0
(hitcnt=0) 0xf8010475
```

## Déboguages

Après avoir activé les débogages, vous pouvez suivre chaque étape de la session WebVPN.

Cet exemple montre l'authentification LDAP avec la récupération d'attribut :

ASA# **show debug**

```
debug ldap enabled at level 255
debug webvpn anyconnect enabled at level 254
ASA#
[63] Session Start
[63] New request Session, context 0xbbe10120, reqType = Authentication
[63] Fiber started
[63] Creating LDAP context with uri=ldap://192.168.11.10:389
[63] Connect to LDAP server: ldap://192.168.11.10:389, status = Successful
[63] supportedLDAPVersion: value = 3
```

```

[63] Binding as Manager
[63] Performing Simple authentication for Manager to 192.168.11.10
[63] LDAP Search:
      Base DN = [DC=test-cisco,DC=com]
      Filter  = [uid=cisco]
      Scope   = [SUBTREE]
[63] User DN = [uid=cisco,ou=People,dc=test-cisco,dc=com]
[63] Server type for 192.168.11.10 unknown - no password policy
[63] Binding as cisco
[63] Performing Simple authentication for cisco to 192.168.11.10
[63] Processing LDAP response for user cisco
[63] Authentication successful for cisco to 192.168.11.10
[63] Retrieved User Attributes:
[63]   cn: value = John Smith
[63]   givenName: value = John
[63]   sn: value = cisco
[63]   uid: value = cisco
[63]   uidNumber: value = 10000
[63]   gidNumber: value = 10000
[63]   homeDirectory: value = /home/cisco
[63]   mail: value = jsmith@dev.local
[63]   objectClass: value = top
[63]   objectClass: value = posixAccount
[63]   objectClass: value = shadowAccount
[63]   objectClass: value = inetOrgPerson
[63]   objectClass: value = organizationalPerson
[63]   objectClass: value = person
[63]   objectClass: value = CiscoPerson
[63]   loginShell: value = /bin/bash

```

**Important !** Les attributs LDAP personnalisés sont mappés aux attributs ASA tels que définis dans la carte d'attribut ldap :

```

[63]   CiscoBanner: value = This is banner 1
[63]     mapped to Banner1: value = This is banner 1
[63]   CiscoIPAddress: value = 10.1.1.1
[63]     mapped to IETF-Radius-Framed-IP-Address: value = 10.1.1.1
[63]   CiscoIPNetmask: value = 255.255.255.128
[63]     mapped to IETF-Radius-Framed-IP-Netmask: value = 255.255.255.128
[63]   CiscoDomain: value = domain1.com
[63]     mapped to IPSec-Default-Domain: value = domain1.com
[63]   CiscoDNS: value = 10.6.6.6
[63]     mapped to Primary-DNS: value = 10.6.6.6
[63]   CiscoACLin: value = ip:inacl#1=permit
ip 10.1.1.0 255.255.255.128 10.11.11.0 255.255.255.0
[63]     mapped to Cisco-AV-Pair: value = ip:inacl#1=permit
ip 10.1.1.0 255.255.255.128 10.11.11.0 255.255.255.0
[63]   CiscoSplitACL: value = ACL1
[63]     mapped to IPSec-Split-Tunnel-List: value = ACL1
[63]   CiscoSplitTunnelPolicy: value = 1
[63]     mapped to IPSec-Split-Tunneling-Policy: value = 1
[63]   CiscoGroupPolicy: value = POLICY1
[63]     mapped to IETF-Radius-Class: value = POLICY1
[63]     mapped to LDAP-Class: value = POLICY1
[63]   userPassword: value = {SSHA}5s81Fmi/9aG/WfPSy3lGmw1ORI4lywWC
[63] ATTR_CISCO_AV_PAIR attribute contains 68 bytes
[63] Fiber exit Tx=315 bytes Rx=907 bytes, status=1
[63] Session End

```

La session LDAP est terminée. Maintenant, ASA traite et applique ces attributs.

La liste de contrôle d'accès dynamique est créée (en fonction de l'entrée ACE dans la paire Cisco-AV) :

```
webvpn_svc_parse_acl: processing ACL: name: 'AAA-user-cisco-E0CF3C05',  
list: YES, id -1  
webvpn_svc_parse_acl: before add: acl_id: -1, acl_name: AAA-user-cisco-E0CF3C05  
webvpn_svc_parse_acl: after add: acl_id: 5, acl_name: AAA-user-cisco-E0CF3C05,  
refcnt: 1
```

## La session WebVPN se poursuit :

```
webvpn_rx_data_tunnel_connect  
CSTP state = HEADER_PROCESSING  
http_parse_cstp_method()  
...input: 'CONNECT /CSCOSSLC/tunnel HTTP/1.1'  
webvpn_cstp_parse_request_field()  
...input: 'Host: 192.168.1.250'  
Processing CSTP header line: 'Host: 192.168.1.250'  
webvpn_cstp_parse_request_field()  
...input: 'User-Agent: Cisco AnyConnect VPN Agent for Windows 3.1.01065'  
Processing CSTP header line: 'User-Agent: Cisco AnyConnect VPN Agent  
for Windows 3.1.01065'  
Setting user-agent to: 'Cisco AnyConnect VPN Agent for Windows 3.1.01065'  
webvpn_cstp_parse_request_field()  
...input: 'Cookie: webvpn=1476503744@122880@  
1365070898@908F356D1C1F4CDF1138088854AF0E480FDCB1BD'  
Processing CSTP header line: 'Cookie: webvpn=1476503744@122880@  
1365070898@908F356D1C1F4CDF1138088854AF0E480FDCB1BD'  
Found WebVPN cookie: 'webvpn=1476503744@122880@  
1365070898@908F356D1C1F4CDF1138088854AF0E480FDCB1BD'  
WebVPN Cookie: 'webvpn=1476503744@122880@1365070898@  
908F356D1C1F4CDF1138088854AF0E480FDCB1BD'  
IPADDR: '1476503744', INDEX: '122880', LOGIN: '1365070898'  
webvpn_cstp_parse_request_field()  
...input: 'X-CSTP-Version: 1'  
Processing CSTP header line: 'X-CSTP-Version: 1'  
Setting version to '1'  
webvpn_cstp_parse_request_field()  
...input: 'X-CSTP-Hostname: admin-Komputer'  
Processing CSTP header line: 'X-CSTP-Hostname: admin-Komputer'  
Setting hostname to: 'admin-Komputer'  
webvpn_cstp_parse_request_field()  
...input: 'X-CSTP-MTU: 1367'  
Processing CSTP header line: 'X-CSTP-MTU: 1367'  
webvpn_cstp_parse_request_field()  
...input: 'X-CSTP-Address-Type: IPv6,IPv4'  
Processing CSTP header line: 'X-CSTP-Address-Type: IPv6,IPv4'  
webvpn_cstp_parse_request_field()  
...input: 'X-CSTP-Local-Address-IP4: 192.168.1.88'  
webvpn_cstp_parse_request_field()  
...input: 'X-CSTP-Base-MTU: 1468'  
webvpn_cstp_parse_request_field()  
...input: 'X-CSTP-Remote-Address-IP4: 192.168.1.250'  
webvpn_cstp_parse_request_field()  
...input: 'X-CSTP-Full-IPv6-Capability: true'  
webvpn_cstp_parse_request_field()  
...input: 'X-DTLS-Master-Secret: F5ADDD0151261404504FC3B165C3B68A90E51  
A1C8EB7EA9B2FE70F1EB8E10929FFD79650B07E218EC8774678CDE1FB5E'  
Processing CSTP header line: 'X-DTLS-Master-Secret: F5ADDD015126140450  
4FC3B165C3B68A90E51A1C8EB7EA9B2FE70F1EB8E10929FFD79650B07E2  
18EC8774678CDE1FB5E'  
webvpn_cstp_parse_request_field()  
...input: 'X-DTLS-CipherSuite: AES256-SHA:AES128-SHA:DES-CBC3-SHA:DES-CBC-SHA'  
Processing CSTP header line: 'X-DTLS-CipherSuite: AES256-SHA:AES128-SHA:  
DES-CBC3-SHA:DES-CBC-SHA'
```

```
webvpn_cstp_parse_request_field()
...input: 'X-DTLS-Accept-Encoding: lzs'
Processing CSTL header line: 'X-DTLS-Accept-Encoding: lzs'
webvpn_cstp_parse_request_field()
...input: 'X-DTLS-Header-Pad-Length: 0'
webvpn_cstp_parse_request_field()
...input: 'X-CSTP-Accept-Encoding: lzs,deflate'
Processing CSTP header line: 'X-CSTP-Accept-Encoding: lzs,deflate'
webvpn_cstp_parse_request_field()
...input: 'X-CSTP-Protocol: Copyright (c) 2004 Cisco Systems, Inc.'
Processing CSTP header line: 'X-CSTP-Protocol:
Copyright (c) 2004 Cisco Systems, Inc.'
```

Ensuite, l'affectation d'adresse se produit. Notez qu'aucun pool d'adresses IP n'est défini sur l'ASA. Si LDAP ne renvoie pas l'attribut *CiscoIPAddress* (mappé à *IETF-Radius-Framed-IP-Address* et utilisé pour l'attribution d'adresse IP), la configuration échouera à ce stade.

```
Validating address: 10.1.1.1
CSTP state = WAIT_FOR_ADDRESS
webvpn_cstp_accept_address: 10.1.1.1/255.255.255.128
webvpn_cstp_accept_ipv6_address: No IPv6 Address
CSTP state = HAVE_ADDRESS
```

La session WebVPN se termine comme suit :

```
SVC: NP setup
np_svc_create_session(0x1E000, 0xb5eafa80, TRUE)
webvpn_svc_np_setup
SVC ACL Name: AAA-user-cisco-E0CF3C05
SVC ACL ID: 5
SVC ACL ID: 5
vpn_put_uauth success!
SVC IPv6 ACL Name: NULL
SVC IPv6 ACL ID: -1
SVC: adding to sessmgmt
SVC: Sending response
Sending X-CSTP-FW-RULE msgs: Start
Sending X-CSTP-FW-RULE msgs: Done
Sending X-CSTP-Quarantine: false
Sending X-CSTP-Disable-Always-On-VPN: false
Unable to initiate NAC, NAC might not be enabled or invalid policy
CSTP state = CONNECTED
```

## Authentification et autorisation séparées ASA

Il est parfois préférable de séparer le processus d'authentification et d'autorisation. Par exemple, utilisez l'authentification par mot de passe pour les utilisateurs définis localement ; après une authentification locale réussie, récupérez tous les attributs utilisateur du serveur LDAP :

```
username cisco password cisco
tunnel-group RA general-attributes
authentication-server-group LOCAL
authorization-server-group LDAP
```

La différence se trouve dans la session LDAP. Dans l'exemple précédent, ASA :

- lié à OpenLDAP avec les informations d'identification de Manager,
- recherche de l'utilisateur *cisco*, et
- binded (authentification simple) à OpenLDAP avec des informations d'identification Cisco.



Actuellement, avec l'autorisation LDAP, la troisième étape n'est plus nécessaire, car l'utilisateur a déjà été authentifié via la base de données locale.

Des scénarios plus courants impliquent l'utilisation de jetons RSA pour le processus d'authentification et d'attributs LDAP/AD pour l'autorisation.

## Attributs ASA de LDAP et de groupe local

Il est important de comprendre la différence entre les attributs LDAP et RADIUS.

Lorsque vous utilisez LDAP, ASA n'autorise pas le mappage à un attribut *radius*. Par exemple, lorsque vous utilisez RADIUS, il est possible de renvoyer l'attribut *cisco-av-pair 217* (Address-Pools). Cet attribut définit un pool d'adresses IP configuré localement qui sont utilisées pour attribuer des adresses IP.

Avec le mappage LDAP, il est impossible d'utiliser cet attribut *cisco-av-pair* spécifique. L'attribut *cisco-av-pair* avec mappage LDAP ne peut être utilisé que pour spécifier différents types de listes de contrôle d'accès.

Ces limitations dans LDAP l'empêchent d'être aussi flexible que Radius. Pour contourner cette stratégie de groupe définie localement peut être créée sur l'ASA avec des attributs qui ne peuvent pas être mappés à partir de ldap (comme les pools d'adresses). Une fois l'utilisateur LDAP authentifié, il est affecté à cette stratégie de groupe (dans notre exemple POLICY1) et aux attributs non spécifiques à l'utilisateur récupérés à partir de la stratégie de groupe.

La liste complète des attributs prise en charge par le mappage LDAP se trouve dans ce document : [Guide de configuration de la gamme Cisco ASA 5500 à l'aide de la CLI, 8.4 et 8.6](#)

Vous pouvez comparer la liste complète des attributs RADIUS VPN3000 pris en charge par ASA ; reportez-vous à ce document : [Guide de configuration de la gamme Cisco ASA 5500 à l'aide de la CLI, 8.4 et 8.6](#)

Reportez-vous à ce document pour obtenir la liste complète des attributs RADIUS IETF pris en charge par ASA : [Guide de configuration de la gamme Cisco ASA 5500 à l'aide de la CLI, 8.4 et 8.6](#)

## ASA et LDAP avec authentification de certificat

ASA ne prend pas en charge la récupération d'attribut de certificat LDAP et la comparaison binaire avec le certificat fourni par Anyconnect. Cette fonctionnalité est réservée à Cisco ACS ou ISE (et uniquement aux supplicants 802.1x), car l'authentification VPN est interrompue sur un périphérique d'accès au réseau (NAD).

Il y a une autre solution. Lorsque l'authentification des utilisateurs utilise des certificats, ASA effectue la validation des certificats et peut récupérer des attributs LDAP en fonction de champs spécifiques du certificat (par exemple, CN) :

```
tunnel-group RA general-attributes
authorization-server-group LDAP
username-from-certificate CN
authorization-required
```

```
tunnel-group RA webvpn-attributes
 authentication certificate
```

Une fois le certificat utilisateur validé par ASA, l'autorisation LDAP est exécutée et les attributs utilisateur (du champ CN) sont récupérés et appliqués.

## Déboguages

Le certificat utilisateur a été utilisé : cn=test1, ou=Sécurité, o=Cisco, l=Cracovie, st=PL, c=PL

Le mappage de certificat est configuré pour mapper ce certificat au groupe de tunnels RA :

```
crypto ca certificate map MAP-RA 10
 issuer-name co tac
webvpn
certificate-group-map MAP-RA 10 RA
```

Validation et mappage des certificats :

ASA# **show debug**

```
debug ldap enabled at level 255
debug webvpn anyconnect enabled at level 254
debug crypto ca enabled at level 3
debug crypto ca messages enabled at level 3
debug crypto ca transactions enabled at level 3
```

Apr 09 2013 17:31:32: %ASA-7-717025: **Validating certificate chain** containing 1 certificate(s).

Apr 09 2013 17:31:32: %ASA-7-717029: **Identified client certificate** within certificate chain.  
serial number: 00FE9C3D61E131CDB1, subject name:  
cn=test1,ou=Security,o=Cisco,l=Krakow,st=PL,c=PL.

Apr 09 2013 17:31:32: %ASA-6-717022: **Certificate was successfully validated.** Certificate is resident and trusted, serial number: 00FE9C3D61E131CDB1, subject name:  
**cn=test1,ou=Security,o=Cisco,l=Krakow,st=PL,c=PL.**

Apr 09 2013 17:31:32: %ASA-6-717028: **Certificate chain was successfully validated** with revocation status check.

Apr 09 2013 17:31:32: %ASA-6-725002: Device completed SSL handshake with client outside:192.168.1.88/49179

Apr 09 2013 17:31:32: %ASA-7-717036: **Looking for a tunnel group match based on certificate maps** for peer certificate with serial number: 00FE9C3D61E131CDB1, subject name:  
cn=test1,ou=Security,o=Cisco,l=Krakow,st=PL,c=PL, issuer\_name:  
cn=TAC,ou=RAC,o=TAC,l=Warsaw,st=Maz,c=PL.

Apr 09 2013 17:31:32: %ASA-7-717038: **Tunnel group match found. Tunnel Group: RA, Peer certificate:** serial number: 00FE9C3D61E131CDB1, subject name:  
cn=test1,ou=Security,o=Cisco,l=Krakow,st=PL,c=PL, issuer\_name:  
cn=TAC,ou=RAC,o=TAC,l=Warsaw,st=Maz,c=PL.

Extraction du nom d'utilisateur du certificat et de l'autorisation à l'aide de LDAP :

Apr 09 2013 17:31:32: %ASA-7-113028: **Extraction of username from VPN client certificate has been requested.** [Request 53]

Apr 09 2013 17:31:32: %ASA-7-113028: Extraction of username from VPN client certificate has

started. [Request 53]

Apr 09 2013 17:31:32: %ASA-7-113028: Extraction of username from VPN client certificate has finished successfully. [Request 53]

Apr 09 2013 17:31:32: %ASA-7-113028: Extraction of username from VPN client certificate has completed. [Request 53]

Apr 09 2013 17:31:32: %ASA-6-302013: Built outbound TCP connection 286 for inside:192.168.11.10/389 (192.168.11.10/389) to identity:192.168.11.250/33383 (192.168.11.250/33383)

Apr 09 2013 17:31:32: %ASA-6-113004: **AAA user authorization Successful : server = 192.168.11.10 : user = test1**

Apr 09 2013 17:31:32: %ASA-6-113003: AAA group policy for user test1 is being set to POLICY1

Apr 09 2013 17:31:32: %ASA-6-113011: AAA retrieved user specific group policy (POLICY1) for user = test1

Apr 09 2013 17:31:32: %ASA-6-113009: AAA retrieved default group policy (MY) for user = test1

Apr 09 2013 17:31:32: %ASA-6-113008: AAA transaction status ACCEPT : user = test1

### Extraction d'attributs à partir de LDAP :

Apr 09 2013 17:31:32: %ASA-7-734003: DAP: User test1, Addr 192.168.1.88: Session Attribute aaa.ldap.cn = **John Smith**

Apr 09 2013 17:31:32: %ASA-7-734003: DAP: User test1, Addr 192.168.1.88: Session Attribute aaa.ldap.givenName = **John**

Apr 09 2013 17:31:32: %ASA-7-734003: DAP: User test1, Addr 192.168.1.88: Session Attribute aaa.ldap.sn = **test1**

Apr 09 2013 17:31:32: %ASA-7-734003: DAP: User test1, Addr 192.168.1.88: Session Attribute aaa.ldap.uid = **test1**

Apr 09 2013 17:31:32: %ASA-7-734003: DAP: User test1, Addr 192.168.1.88: Session Attribute aaa.ldap.uidNumber = **10000**

Apr 09 2013 17:31:32: %ASA-7-734003: DAP: User test1, Addr 192.168.1.88: Session Attribute aaa.ldap.gidNumber = **10000**

Apr 09 2013 17:31:32: %ASA-7-734003: DAP: User test1, Addr 192.168.1.88: Session Attribute aaa.ldap.homeDirectory = **/home/cisco**

Apr 09 2013 17:31:32: %ASA-7-734003: DAP: User test1, Addr 192.168.1.88: Session Attribute aaa.ldap.mail = **jsmith@dev.local**

Apr 09 2013 17:31:32: %ASA-7-734003: DAP: User test1, Addr 192.168.1.88: Session Attribute aaa.ldap.objectClass.1 = **top**

Apr 09 2013 17:31:32: %ASA-7-734003: DAP: User test1, Addr 192.168.1.88: Session Attribute aaa.ldap.objectClass.2 = **posixAccount**

Apr 09 2013 17:31:32: %ASA-7-734003: DAP: User test1, Addr 192.168.1.88: Session Attribute aaa.ldap.objectClass.3 = **shadowAccount**

Apr 09 2013 17:31:32: %ASA-7-734003: DAP: User test1, Addr 192.168.1.88: Session Attribute  
aaa.ldap.**objectClass.4 = inetOrgPerson**

Apr 09 2013 17:31:32: %ASA-7-734003: DAP: User test1, Addr 192.168.1.88: Session Attribute  
aaa.ldap.**objectClass.5 = organizationalPerson**

Apr 09 2013 17:31:32: %ASA-7-734003: DAP: User test1, Addr 192.168.1.88: Session Attribute  
aaa.ldap.**objectClass.6 = person**

Apr 09 2013 17:31:32: %ASA-7-734003: DAP: User test1, Addr 192.168.1.88: Session Attribute  
aaa.ldap.**objectClass.7 = CiscoPerson**

Apr 09 2013 17:31:32: %ASA-7-734003: DAP: User test1, Addr 192.168.1.88: Session Attribute  
aaa.ldap.**loginShell = /bin/bash**

Apr 09 2013 17:31:32: %ASA-7-734003: DAP: User test1, Addr 192.168.1.88: Session Attribute  
aaa.ldap.**userPassword = {CRYPT}\***

Apr 09 2013 17:31:32: %ASA-7-734003: DAP: User test1, Addr 192.168.1.88: Session Attribute  
aaa.ldap.**CiscoBanner = This is banner 1**

Apr 09 2013 17:31:32: %ASA-7-734003: DAP: User test1, Addr 192.168.1.88: Session Attribute  
aaa.ldap.**CiscoIPAddress = 10.1.1.1**

Apr 09 2013 17:31:32: %ASA-7-734003: DAP: User test1, Addr 192.168.1.88: Session Attribute  
aaa.ldap.**CiscoIPNetmask = 255.255.255.128**

Apr 09 2013 17:31:32: %ASA-7-734003: DAP: User test1, Addr 192.168.1.88: Session Attribute  
aaa.ldap.**CiscoDomain = domain1.com**

Apr 09 2013 17:31:32: %ASA-7-734003: DAP: User test1, Addr 192.168.1.88: Session Attribute  
aaa.ldap.**CiscoDNS = 10.6.6.6**

Apr 09 2013 17:31:32: %ASA-7-734003: DAP: User test1, Addr 192.168.1.88: Session Attribute  
aaa.ldap.**CiscoACLIn = ip:inacl#1=permit ip 10.1.1.0 255.255.255.128 10.11.11.0 255.255.255.0**

Apr 09 2013 17:31:32: %ASA-7-734003: DAP: User test1, Addr 192.168.1.88: Session Attribute  
aaa.ldap.**CiscoSplitACL = ACL1**

Apr 09 2013 17:31:32: %ASA-7-734003: DAP: User test1, Addr 192.168.1.88: Session Attribute  
aaa.ldap.**CiscoSplitTunnelPolicy = 1**

Apr 09 2013 17:31:32: %ASA-7-734003: DAP: User test1, Addr 192.168.1.88: Session Attribute  
aaa.ldap.**CiscoGroupPolicy = POLICY1**

### Attributs mappés Cisco :

Apr 09 2013 17:31:32: %ASA-7-734003: DAP: User test1, Addr 192.168.1.88: Session Attribute  
aaa.**cisco.grouppolicy = POLICY1**

Apr 09 2013 17:31:32: %ASA-7-734003: DAP: User test1, Addr 192.168.1.88: Session Attribute  
aaa.**cisco.ipaddress = 10.1.1.1**

Apr 09 2013 17:31:32: %ASA-7-734003: DAP: User test1, Addr 192.168.1.88: Session Attribute  
aaa.**cisco.username = test1**

Apr 09 2013 17:31:32: %ASA-7-734003: DAP: User test1, Addr 192.168.1.88: Session Attribute  
aaa.**cisco.username1 = test1**

Apr 09 2013 17:31:32: %ASA-7-734003: DAP: User test1, Addr 192.168.1.88: Session Attribute  
aaa.cisco.username2 =

Apr 09 2013 17:31:32: %ASA-7-734003: DAP: User test1, Addr 192.168.1.88: Session Attribute  
aaa.cisco.tunnelgroup = RA

Apr 09 2013 17:31:32: %ASA-6-734001: DAP: User test1, Addr 192.168.1.88, Connection AnyConnect:  
The following **DAP records** were selected for this connection: **DfltAccessPolicy**

Apr 09 2013 17:31:32: %ASA-6-113039: **Group**

## Authentification secondaire

Si une authentification à deux facteurs est requise, il est possible d'utiliser un mot de passe de jeton avec l'authentification et l'autorisation LDAP :

```
tunnel-group RA general-attributes
 authentication-server-group RSA
 secondary-authentication-server-group LDAP
 authorization-server-group LDAP
tunnel-group RA webvpn-attributes
 authentication aaa
```

Ensuite, l'utilisateur doit fournir un nom d'utilisateur et un mot de passe de RSA (ce que l'utilisateur a : un jeton), ainsi qu'un nom d'utilisateur/mot de passe LDAP (ce que l'utilisateur connaît). Il est également possible d'utiliser un nom d'utilisateur du certificat pour l'authentification secondaire. Pour plus d'informations sur la double authentification, reportez-vous au [Guide de configuration de la gamme Cisco ASA 5500 à l'aide de l'interface de ligne de commande, 8.4 et 8.6](#).

## Informations connexes

- [Guide de configuration de la gamme Cisco ASA 5500 à l'aide de la CLI, 8.4 et 8.6](#)
- [Guide de l'administrateur du logiciel OpenLDAP 2.4](#)
- [Numéros d'entreprise privée](#)
- [Support et documentation techniques - Cisco Systems](#)