

# 使用自定義架構和證書的ASA Anyconnect VPN和OpenLDAP授權配置示例

## 目錄

[簡介](#)

[必要條件](#)

[需求](#)

[採用元件](#)

[設定](#)

[基本OpenLDAP配置](#)

[自定義Openldap架構](#)

[ASA配置](#)

[驗證](#)

[測試VPN訪問](#)

[調試](#)

[ASA獨立的身份驗證和授權](#)

[來自LDAP和本地組的ASA屬性](#)

[帶證書身份驗證的ASA和LDAP](#)

[調試](#)

[輔助身份驗證](#)

[相關資訊](#)

## 簡介

本文檔介紹如何使用自定義架構配置OpenLDAP，以支援連線到思科自適應安全裝置(ASA)的Cisco Anyconnect安全移動客戶端的每個使用者屬性。ASA配置非常基本，因為所有使用者屬性都是從OpenLDAP伺服器檢索的。本文檔還介紹了與證書一起使用時LDAP身份驗證和授權方面的差異。

## 必要條件

### 需求

思科建議您瞭解以下主題：

- 有關Linux配置的基本知識
- 有關ASA CLI配置的基本知識

### 採用元件

本檔案中的資訊是根據以下軟體版本：

- Cisco ASA 8.4及更高版本
- OpenLDAP版本2.4.30

# 設定

## 基本OpenLDAP配置

### 步驟1.配置伺服器。

此示例使用test-cisco.com ldap樹。

ldap.conf檔案用於設定本地ldap客戶端可以使用的系統級別預設值。

**附註：**儘管您不需要設定系統級預設值，但可以在運行本地ldap客戶端時幫助測試和排除伺服器故障。

/etc/openldap/ldap.conf:

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

slapd.conf檔案用於OpenLDAP伺服器配置。預設架構檔案包括廣泛使用的LDAP定義。例如，對象類名稱*person*在core.schema檔案中定義。此配置使用通用模式，並為思科特定屬性定義自己的模式。

/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
```

### 步驟2.驗證LDAP配置。

若要驗證基本OpenLDAP是否正常工作，請運行以下配置：

```
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

```

### 步驟3.向資料庫新增記錄。

在正確測試和配置所有內容後，向資料庫新增記錄。若要為使用者和組新增基本容器，請運行以下配置：

```

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
```

## 自定義Openldap架構

現在，基本配置工作正常，您可以新增自定義架構。在此配置示例中，將建立一個名為 *CiscoPerson* 的新型對象類，並在此對象類中建立和使用這些屬性：

- 思科橫幅
- CiscoACLin
- CiscoDomain
- CiscoDNS
- CiscoIPAddress
- CiscoIP網路掩碼
- CiscoSplitACL
- CiscoSplitTunnelPolicy
- CiscoGroupPolicy

### 步驟1.在cisco.schema中建立新架構。

```
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 ) )

```

## 重要附註

- 為您的公司使用專用企業OID。任何OID都會起作用，但最佳實踐是使用IANA分配的OID。在此範例中設定的範例從1.3.6.1.4.1.9(由思科保留：<http://www.iana.org/assignments/enterprise-numbers>)。
- 以下OID部分(500.1.1-500.1.9)已用於不直接幹擾Cisco OID的主樹(「1.3.6.1.4.1.9」)。
- 此資料庫使用schema/core.ldif中定義的*Person*對象類。該對象為TOP型別，記錄只能包含一個此類屬性(因此*CiscoPerson*對象類為Auxiliary型別)。
- 名為*CiscoPerson*的對象類必須包括SN或CN，並且可包括之前定義的任何自定義Cisco屬性。請注意，它還可以包括在其他方案中定義的任何其他屬性(如*userPassword*或*telephoneNumber*)。
- 請記住，每個對象都應具有不同的OID編號。
- 自定義屬性不區分大小寫，並且字串型別具有UTF-8編碼，最多128個字元(由SYNTAX定義)。

## 步驟2.在slapd.conf中包含架構。

```

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

```

## 步驟3.重新啟動服務。

```

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

```

## 步驟4.新增具有所有自定義屬性的新使用者。

在此示例中，使用者屬於多個objectClass對象，並且從所有這些對象繼承屬性。在此過程中，可以輕鬆地新增其他模式或屬性，而無需對現有資料庫記錄進行更改。

```

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"
```

### 步驟5.設定使用者的密碼。

```
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
```

### 步驟6.檢驗配置。

```
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
```

## ASA配置

### 步驟1.配置介面和證書。

```
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 blaebcc4 fdc549a2 010b83f3 aa0e12b3 f03a4f49
d8e6fdea 61776ae5 17daf7e4 6baf810d 37c24784 bd71429b dc0494c0 84a020ff
1be0c903 a055f634 1e29b6ea 7d7f3280 f161a86c 50d40b6c c24bc8b0 493c0918
8a185e05 1b52d8b0 0e
quit
```

## 步驟2.生成自簽名證書。

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

## 步驟3.在外部介面上啟用WebVPN。

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

## 步驟4.拆分ACL配置。

OpenLDAP返回ACL名稱：

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

## 步驟5.建立使用預設組策略(DfltAccessPolicy)的隧道組名稱。

具有特定LDAP屬性(*CiscoGroupPolicy*)的使用者將對映到其他策略：POLICY1

```
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
```

ASA aaa伺服器配置使用ldap屬性對映從OpenLDAP返回的屬性對映到ASA可以為Anyconnect使用者解釋的屬性。

```
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
```

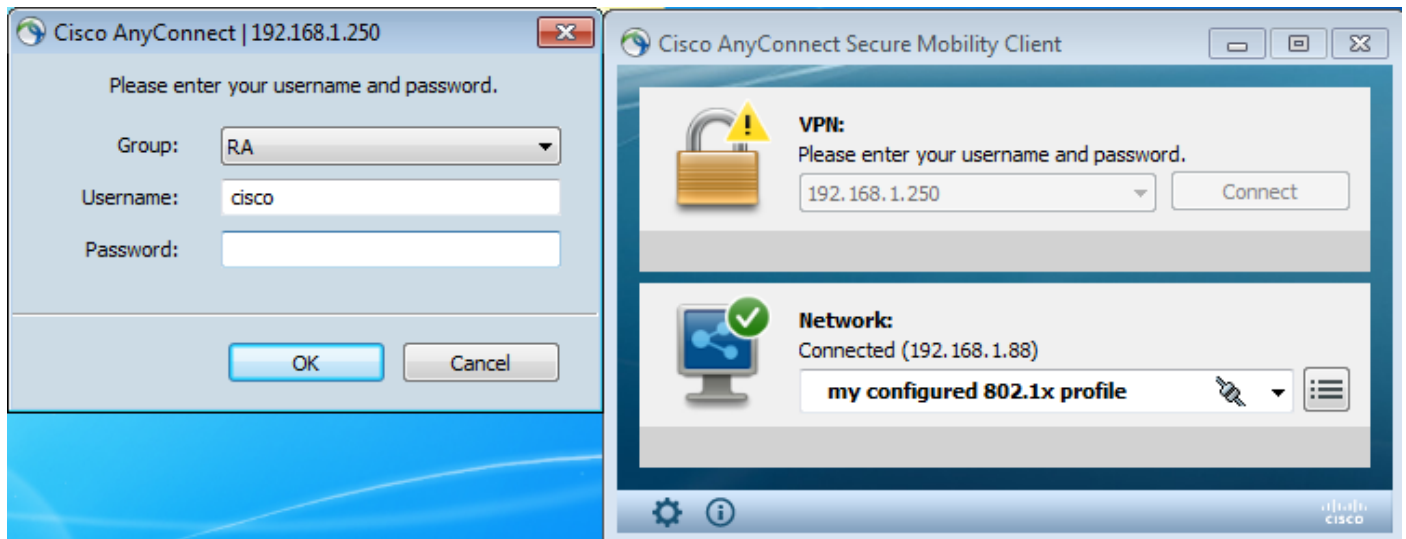
**步驟6. 啟用LDAP伺服器以進行指定隧道組的身份驗證。**

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

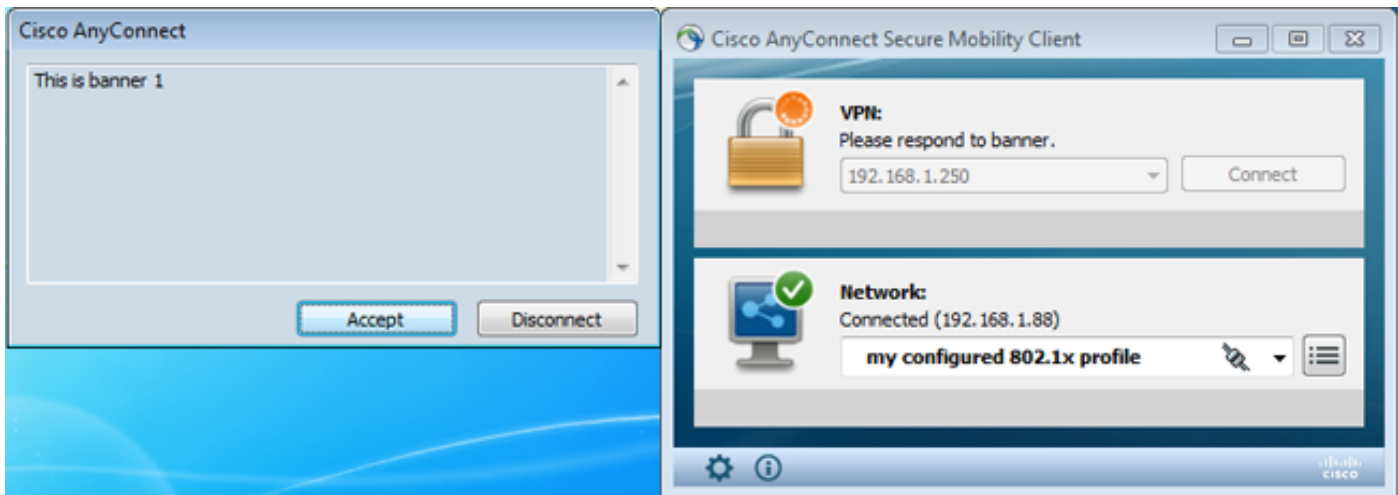
## 驗證

### 測試VPN訪問

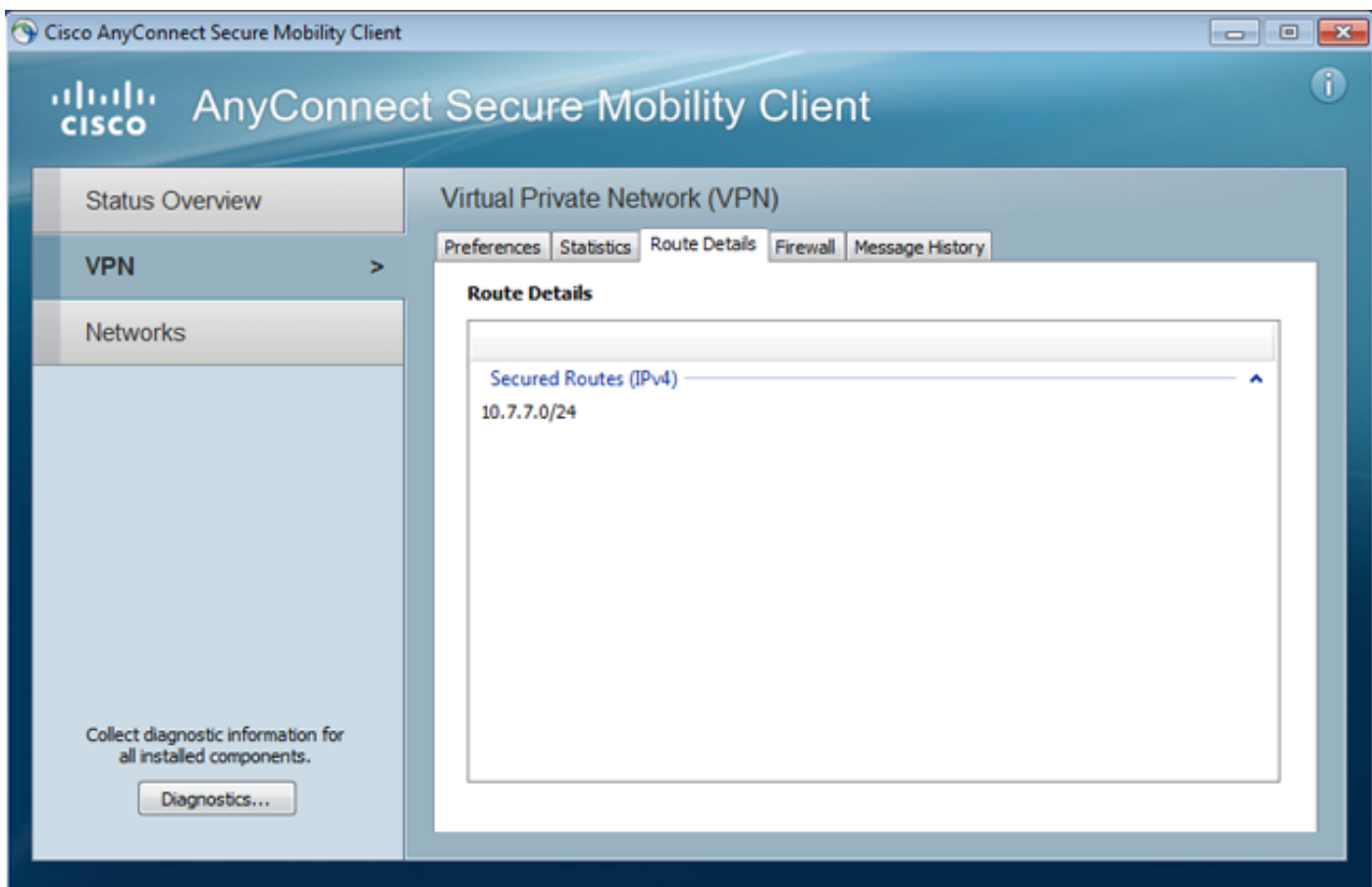
Anyconnect配置為連線到192.168.1.250。登入是使用者名稱cisco和密碼pass1。



身份驗證後，使用正確的標語。



傳送正確的拆分ACL ( ASA上定義的ACL1 )。



Anyconnect介面配置了IP:10.1.1.1和網路掩碼255.255.255.128。域為domain1.com,DNS伺服器為10.6.6.6。

```

Ethernet adapter Połączenie lokalne 2:

    Connection-specific DNS Suffix . . . : domain1.com
    Description . . . . . : Cisco AnyConnect Secure Mobility Client U
    Physical 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
  
```

在ASA上，使用者cisco已收到IP:10.1.1.1並分配給組策略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:

此外，還會為該使用者安裝動態訪問清單：

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

**調試**

啟用調試後，您可以跟蹤WebVPN會話的每個步驟。

此示例顯示了LDAP身份驗證以及屬性檢索：

```
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
```

**重要！** 自定義LDAP屬性對映到ASA屬性（如ldap attribute-map中所定義）：

```
[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
```

LDAP會話已完成。現在，ASA處理和應用這些屬性。

建立動態ACL ( 根據Cisco-AV-Pair中的專案ACE ) :

```
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
```

WebVPN會話繼續 :

```
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.'

```

接著，進行地址分配。請注意，ASA上未定義IP池。如果LDAP沒有返回CiscoIPAddress屬性(該屬性對映到IETF-Radius-Framed-IP-Address，並且用於IP地址分配)，則配置在此階段將失敗。

```

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

```

WebVPN會話完成：

```

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

```

## ASA獨立的身份驗證和授權

有時，最好將身份驗證和授權過程分開。例如，對本地定義的使用者使用密碼身份驗證；然後，在本地身份驗證成功後，從LDAP伺服器檢索所有使用者屬性：

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

不同之處在於LDAP會話。在上一個示例中，ASA:

- 已使用管理器憑據繫結到OpenLDAP，
- 已搜尋使用者 *cisco*，且
- 使用思科憑證繫結到OpenLDAP ( 簡單身份驗證 )。

目前，通過LDAP授權，第三步不再必要，因為使用者已通過本地資料庫進行身份驗證。

更常見的情況包括使用RSA令牌進行身份驗證過程，使用LDAP/AD屬性進行授權。

## 來自LDAP和本地組的ASA屬性

瞭解LDAP屬性和RADIUS屬性之間的區別非常重要。

使用LDAP時，ASA不允許對映到任何 *radius* 屬性。例如，使用RADIUS時，可能會返回 *cisco-av-pair* 屬性217(Address-Pools)。該屬性定義用於分配IP地址的本地配置的IP地址池。

使用LDAP對映時，無法使用該特定 *cisco-av-pair* 屬性。具有LDAP對映的 *cisco-av-pair* 屬性只能用於指定不同型別的ACL。

LDAP中的這些限制使其無法像Radius一樣靈活。要解決這個問題，可以在ASA上建立本地定義的組策略，該組策略具有無法從ldap對映的屬性 ( 如Address-Pools )。一旦LDAP使用者通過身份驗證，它們就會分配給該組策略 ( 在我們的示例POLICY1中 )，並且會從組策略中重新檢索非使用者特定屬性。

LDAP對映支援的完整屬性清單可以在本文檔中找到：[使用CLI 8.4和8.6的Cisco ASA 5500系列配置指南](#)

您可以與ASA支援的RADIUS VPN3000屬性的完整清單進行比較；請參閱以下檔案：[使用CLI 8.4和8.6的Cisco ASA 5500系列配置指南](#)

有關ASA支援的RADIUS IETF屬性的完整清單，請參閱以下文檔：[使用CLI 8.4和8.6的Cisco ASA 5500系列配置指南](#)

## 帶證書身份驗證的ASA和LDAP

ASA不支援LDAP證書屬性檢索以及與Anyconnect提供的證書的二進位制比較。此功能留給Cisco ACS或ISE ( 且僅留給802.1x請求方 )，因為VPN身份驗證在網路訪問裝置(NAD)上終止。

還有另一個解決辦法。當使用者身份驗證使用證書時，ASA執行證書驗證，並可根據證書中的特定欄位 ( 例如CN ) 檢索LDAP屬性：

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



authentication certificate

ASA驗證使用者證書後，將執行LDAP授權，並檢索和應用使用者屬性（來自CN欄位）。

## 調試

已使用使用者證書：cn=test1,ou=Security,o=Cisco,l=Krakow,st=PL,c=PL

證書對映配置為將該證書對映到RA隧道組：

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

證書驗證和對映：

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.

使用LDAP從證書和授權中提取使用者名稱：

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

從LDAP檢索屬性：

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**

思科對映屬性：

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**

## 輔助身份驗證

如果需要雙因素身份驗證，可以將令牌密碼與LDAP身份驗證和授權一起使用：

```
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
```

然後，使用者必須提供來自RSA的使用者名稱和密碼（使用者擁有的東西 — 令牌）以及LDAP使用者名稱/密碼（使用者知道的東西）。也可以使用證書中的使用者名稱進行輔助身份驗證。有關雙重身份驗證的詳細資訊，請參閱[使用CLI 8.4和8.6的Cisco ASA 5500系列配置指南](#)。

## 相關資訊

- [使用CLI 8.4和8.6的Cisco ASA 5500系列配置指南](#)
- [OpenLDAP軟體2.4管理員指南](#)
- [私營企業號碼](#)
- [技術支援與文件 - Cisco Systems](#)