ASA Anyconnect VPN和OpenLDAP授权与自定义架构和证书配置示例

目录

简介

本文档介绍如何使用自定义架构配置OpenLDAP，以支持连接到思科自适应安全设备(ASA)的Cisco Anyconnect安全移动客户端的每用户属性。 ASA配置非常基本，因为所有用户属性都从OpenLDAP服务器检索。本文档中还介绍了LDAP身份验证和授权与证书一起使用时的差异。

先决条件

要求

Cisco 建议您了解以下主题：

- 有关Linux配置的基本知识
- 有关ASA CLI配置的基本知识

使用的组件

本文档中的信息基于以下软件版本：

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

配置

基本OpenLDAP配置

步骤1.配置服务器。

本示例使用test-cisco.com ldap树。

ldap.conf文件用于设置本地ldap客户端可使用的系统级默认值。

注意：虽然您不需要设置系统级默认值，但在运行本地LDAP客户端时，这些默认值有助于测试和排除服务器故障。

/etc/openldap/ldap.conf:

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

slapd.conf文件用于OpenLDAP服务器配置。默认架构文件包括广泛使用的LDAP定义。例如，对象类名*person*在core.schema文件中定义。此配置使用该通用模式并为思科特定属性定义其自己的模式。

/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
```

步骤2.检验LDAP配置。

要验证基本OpenLDAP是否工作，请运行以下配置：

```
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
```

步骤3.将记录添加到数据库。

正确测试和配置所有内容后，将记录添加到数据库。要为用户和组添加基本容器，请运行以下配置：

```
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
```

自定义OpenLDAP架构

现在基本配置工作，您可以添加自定义架构。在此配置示例中，创建了名为CiscoPerson的新类型的对象类，并在此对象类中创建和使用这些属性：

- 思科横幅
- CiscoACLin
- CiscoDomain
- CiscoDNS
- CiscoIP地址
- CiscoIPNetmask
- CiscoSplitACL
- CiscoSplitTunnelPolicy
- 思科组策略

步骤1.在cisco.schema中创建新架构。

```
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 ) )
```

重要说明

- 为您的公司使用私有企业OID。任何OID都有效，但最佳做法是使用IANA分配的OID。本示例中配置的从1.3.6.1.4.1.9开始(由思科保留：http://www.iana.org/assignments/enterprise-numbers)。
- OID的以下部分(500.1.1-500.1.9)已用于不直接干扰Cisco OID的主树(“1.3.6.1.4.1.9”)。
- 此数据库使用*schema/core.ldif*中定义的Person对象类。该对象为TOP类型，记录只能包含一个此类属性(这就是CiscoPerson对象类为Auxiliary类型的原因)。
- 名为CiscoPerson的对象类必须包含SN或CN，并且可以包含之前定义的任何自定义思科属性。请注意，它还可以包括在其他方案中定义的任何其他属性(如*userPassword*或*telephoneNumber*)。
- 请记住，每个对象应具有不同的OID编号。
- 自定义属性不区分大小写，且*字符串*类型采用UTF-8编码，最多128个字符（由SYNTAX定义）。

步骤2.在sldap.conf中包含架构。

```
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
```

步骤3.重新启动服务。

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

步骤4.添加具有所有自定义属性的新用户。

在本例中，用户属于多个objectClass对象，并且它继承了所有对象的属性。通过此过程，可以轻松地添加其他架构或属性，而无需更改现有数据库记录。

```
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"
```

步骤5.为用户设置密码。

```
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
```

步骤6.检验配置。

```
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:: e1NTSEF9NXM4MUZtaS85YUcvV2ZQU3kzbEdtdzFPUkk0bHl3V0M=

# search result
search: 2
result: 0 Success

# numResponses: 2
# numEntries: 1
```

ASA 配置

步骤1.配置接口和证书。

```
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
```

步骤2.生成自签名证书。

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

步骤3.在外部接口上启用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
```

步骤4.拆分ACL配置。

ACL名称由OpenLDAP返回：

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

步骤5.创建使用默认组策略(DfltAccessPolicy)的隧道组名称。

具有特定LDAP属性(CiscoGroupPolicy*)的*用户将映射到另一个策略：策略1

```
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服务器配置使用ldap属性映射从OpenLDAP返回的属性映射到ASA可解释的属性（对于Anyconnect用户）。

```
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
```

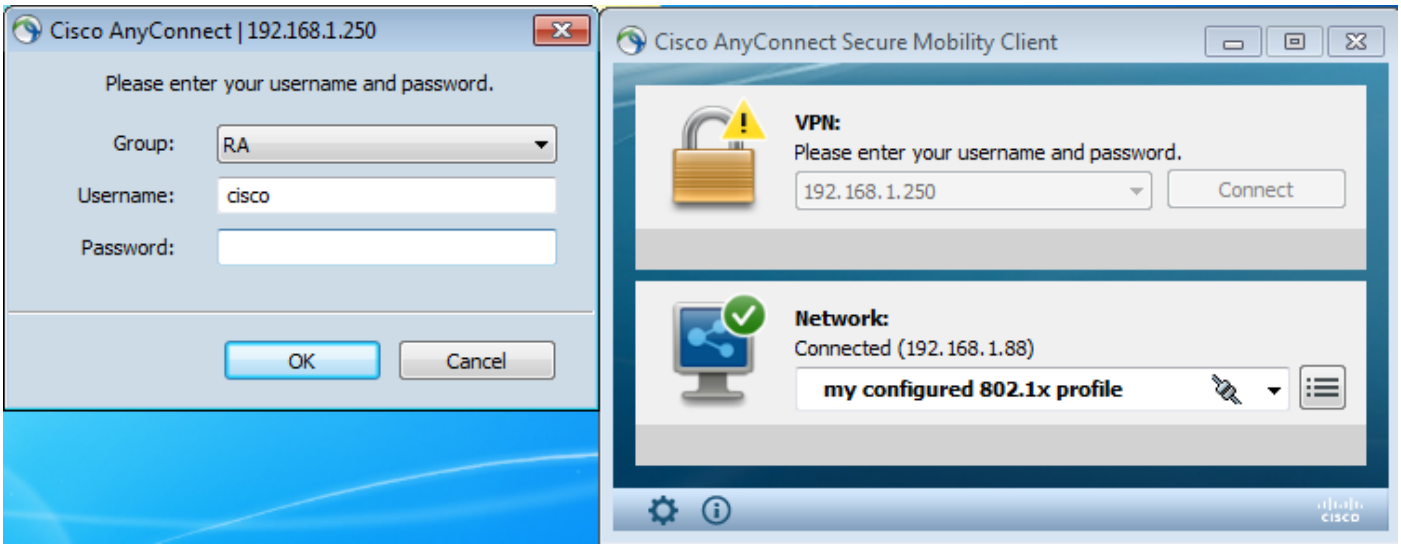
步骤6.启用LDAP服务器以对指定隧道组进行身份验证。

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

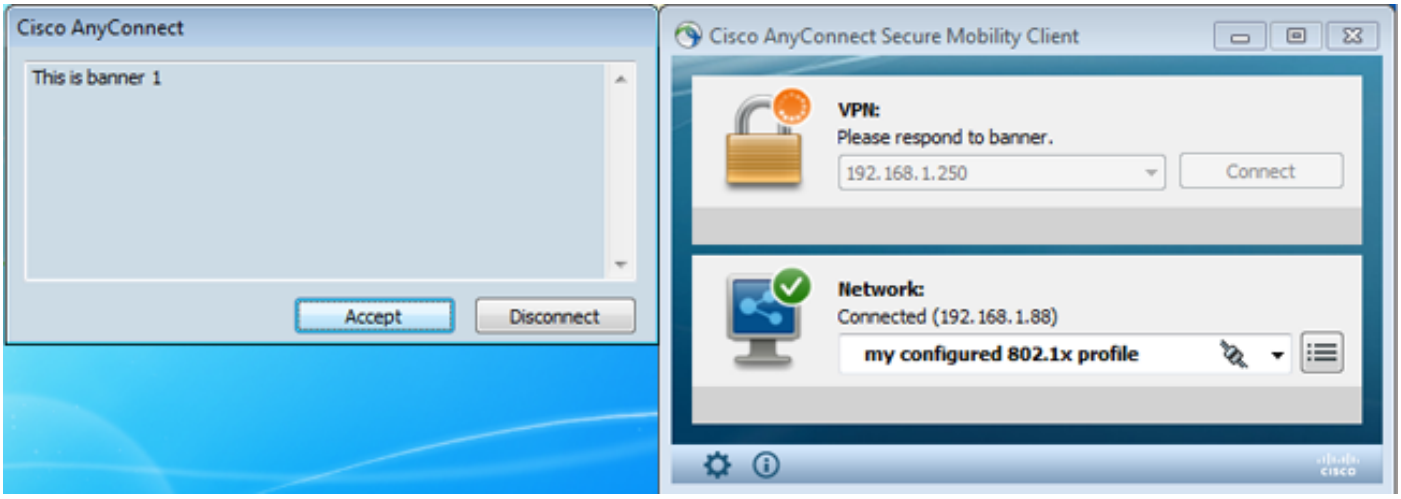
验证

测试VPN访问

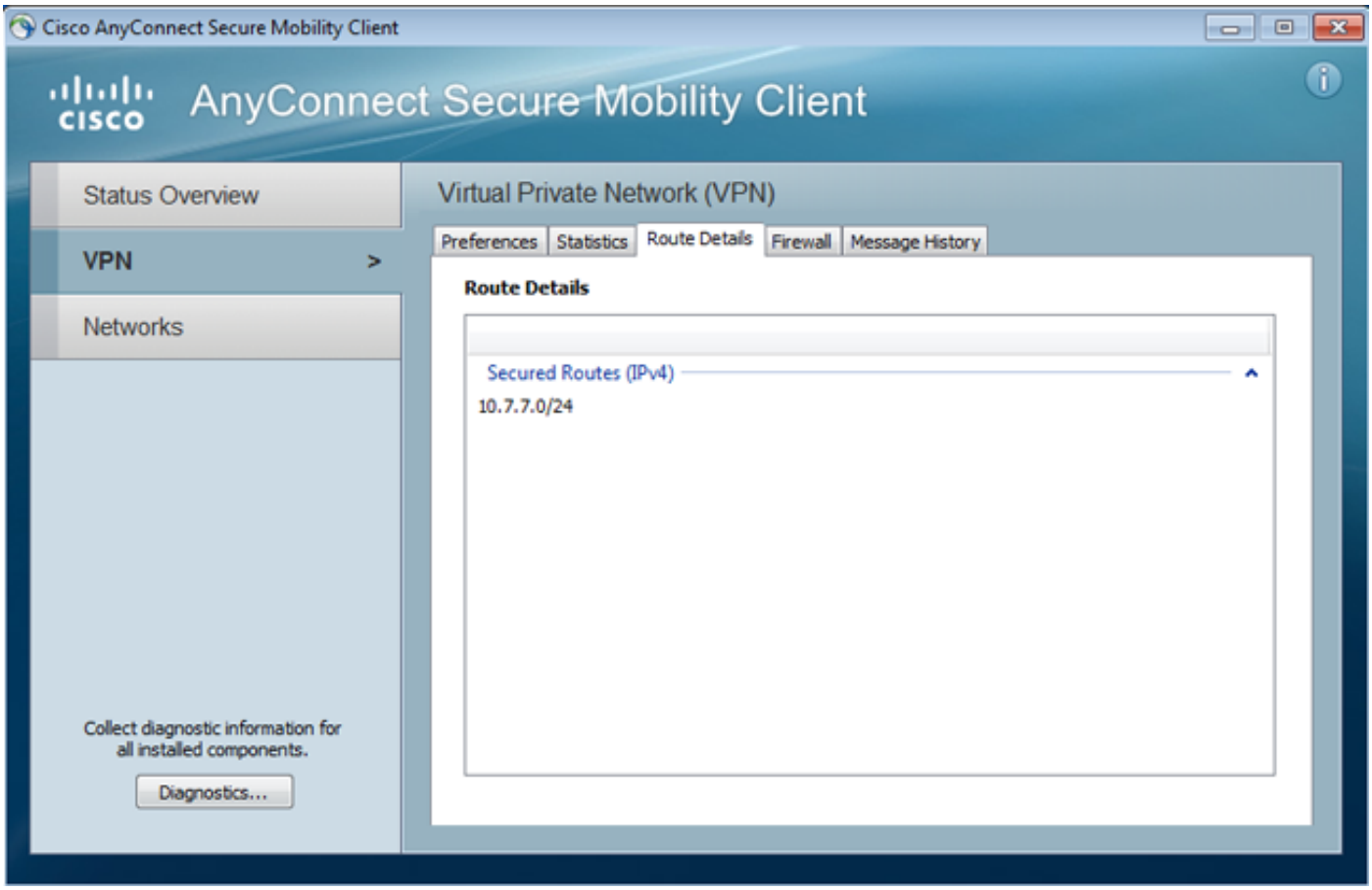
Anyconnect配置为连接到192.168.1.250。登录为用户名*cisco*和密*码pass1*。



身份验证后使用正确的标语。



发送正确的拆分ACL（ASA上定义的ACL1）。



Anyconnect接口配置了IP:10.1.1.1和网络掩码255.255.255.128。域为domain1.com，DNS服务器为10.6.6.6。

```
Ethernet adapter Połączenie lokalne 2:

   Connection-specific DNS Suffix  . : domain1.com
   Description . . . . . . . . . . . : Cisco AnyConnect Secure Mobility Client V
irtual 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，用户*cisco*已收到IP:10.1.1.1，并分配给组策略*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，还为该用户安装了动态访问列表：

```
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
```

调试

启用调试后，可以跟踪WebVPN会话的每个步骤。

此示例显示LDAP身份验证和属性检索：

```
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！自定义LDAP属性映射到ASA属性，如ldap属性映射中所定义：

```
[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会话已完成。现在，ASA将处理并应用这些属性。

动态ACL创建（根据Cisco-AV-Pair中的条目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会话继续：

```
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.'
```

接下来，进行地址分配。请注意，ASA上未定义IP池。如果LDAP不返回*CiscoIPAddress*属性(映射到*IETF-Radius-Framed-IP-Address并用于IP地址分配*)，则此阶段的配置将失败。

```
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会话完成：

```
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独立身份验证和授权

有时，最好将身份验证和授权过程分开。例如，对本地定义的用户使用密码身份验证；然后，在本地身份验证成功后，从LDAP服务器检索所有用户属性：

```
username cisco password cisco
tunnel-group RA general-attributes
authentication-server-group LOCAL
 authorization-server-group LDAP
```
区别在于LDAP会话。在上一个示例中，ASA:

- 绑定到OpenLDAP，带Manager凭证，
- 已搜索用户*cisco*和
- 已绑定（简单身份验证）到具有思科凭证的OpenLDAP。

目前，使用LDAP授权时，不再需要第三步，因为用户已通过本地数据库进行身份验证。

更常见的情况包括使用RSA令牌进行身份验证过程和使用LDAP/AD属性进行授权。

来自LDAP和本地组的ASA属性

了解LDAP属性和RADIUS属性之间的区别非常重要。

使用LDAP时，ASA不允许映射到任何*RADIUS属*性。例如，使用RADIUS时，可以返回*cisco-av-pair属*性217（地址池）。 该属性定义本地配置的用于分配IP地址的IP地址池。

使用LDAP映射时，无法使用该特*定的cisco-av-pair属*性。带*LDAP映射的*cisco-av-pair属性只能用于指定不同类型的ACL。

LDAP中的这些限制使其无法像Radius一样灵活。要处理此本地定义的组策略，可以在ASA上使用无法从ldap映射的属性（如地址池）创建。 LDAP用户经过身份验证后，会将其分配到该组策略（在我们的示例POLICY1中）以及从组策略重新检索的非用户特定属性。

LDAP映射支持的完整属性列表可在本文档中找到： 使用CLI、8.4和8.6的Cisco ASA 5500系列配置指南

您可以与ASA支持的RADIUS VPN3000属性的完整列表进行比较；请参阅本文档：使用CLI、8.4和8.6的Cisco ASA 5500系列配置指南

有关ASA支持的RADIUS IETF属性的完整列表，请参阅本文档：使用CLI、8.4和8.6的Cisco ASA 5500系列配置指南

带证书身份验证的ASA和LDAP

ASA不支持LDAP证书属性检索和与Anyconnect提供的证书的二进制比较。该功能保留给Cisco ACS或ISE（且仅适用于802.1x请求方），因为VPN身份验证在网络接入设备(NAD)上终止。

还有一个解决方案。当用户身份验证使用证书时，ASA执行证书验证，并可以根据证书（例如CN）中的特定字段检索LDAP属性：

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

在用户证书由ASA验证后，执行LDAP授权并检索和应用用户属性（来自CN字段）。

调试

已使用用户证书：cn=test1,ou=安全，o=Cisco，l=Krakow，st=PL，c=PL

证书映射配置为将该证书映射到RA隧道组：

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

证书验证和映射：

```
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从证书和授权中提取用户名：

```
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

从LDAP检索属性：

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**

思科映射属性：

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
```

.

辅助身份验证

如果需要双因素身份验证，则可以使用令牌密码以及LDAP身份验证和授权：

```
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
```

然后，用户必须提供来自RSA的用户名和密码（用户拥有的信息 — 令牌）以及LDAP用户名/密码（用户知道的信息）。 也可以使用证书中的用户名进行辅助身份验证。有关双重身份验证的详细信息，请参阅使用CLI、8.4和8.6的Cisco ASA 5500系列配置指南。

相关信息

- 使用CLI、8.4和8.6的Cisco ASA 5500系列配置指南
- OpenLDAP软件2.4管理员指南
- 私有企业编号
- 技术支持和文档 - Cisco Systems