IKEv2 from Android strongSwan to Cisco IOS with EAP and RSA Authentication

TAC **Document ID: 116837**

Contributed by Michal Garcarz and Salah Gherdaoui, Cisco TAC
Engineers.
Jan 21, 2016

Contents

Introduction

This document describes how to configure the mobile version of strongSwan in order to access a Cisco IOS® software VPN gateway via the Internet Key Exchange Version 2 (IKEv2) protocol.

Three examples are presented:

- Android phone with strongSwan that connects to the Cisco IOS software VPN gateway with Extensible Authentication Protocol - Message Digest 5 (EAP-MD5) authentication.
- Android phone with strongSwan that connects to the Cisco IOS software VPN gateway with certificate authentication (RSA).
- Android phone with strongSwan that connects to the Cisco IOS software VPN gateway behind Network Address Translation (NAT). There is a requirement to have two x509 extensions Subject Alternative Name in the VPN gateway certificate.

Cisco IOS software and strongSwan limitations are also included.

Prerequisites

Requirements

Cisco recommends that you have knowledge of these topics:

- Basic knowledge of OpenSSL configuration
- Basic knowledge of Cisco IOS software command-line interface (CLI) configuration
- Basic knowledge of IKEv2

Components Used

The information in this document is based on these software and hardware versions:

- Android 4.0 or later with strongSwan
- Cisco IOS Software Release 15.3T or later
- Cisco Identity Services Engine (ISE) Software, Version 1.1.4 and later

The information in this document was created from the devices in a specific lab environment. All of the devices used in this document started with a cleared (default) configuration. If your network is live, make sure that you understand the potential impact of any command.

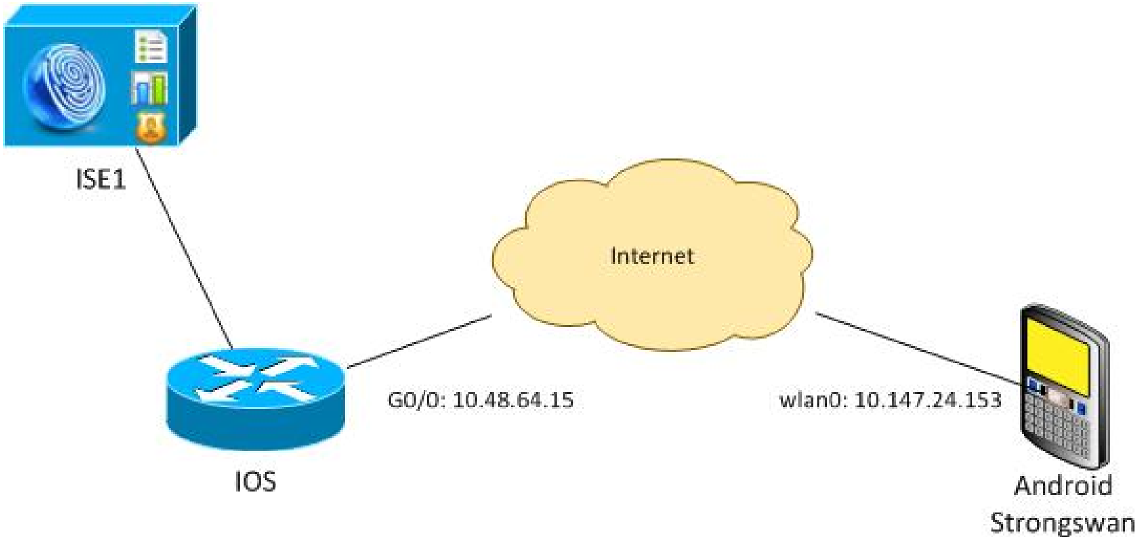
Configure

Notes:

The Output Interpreter Tool (registered customers only) supports certain **show** commands. Use the Output Interpreter Tool in order to view an analysis of **show** command output.

Refer to Important Information on Debug Commands before you use **debug** commands.

Network Diagram



Android strongSwan establishes an IKEv2 tunnel with a Cisco IOS software gateway in order to access internal networks securely.

Certificate Enrollment

Certificates are a prerequisite for both EAP-based and RSA-based authentication.

In the EAP authentication scenario, a certificate is needed only on the VPN gateway. The client connects to the Cisco IOS software only when the software presents a certificate signed by a Certificate Authority (CA) that is trusted on Android. An EAP session then starts for the client to authenticate to the Cisco IOS software.

For RSA-based authentication, both endpoints must have a correct certificate.

When an IP address is used as a peer-ID, there are additional requirements for the certificate. Android strongSwan verifies if the IP address of the VPN gateway is included in the x509 extension Subject Alternative Name. If not, Android drops the connection; this is a good practice as well as a recommendation of RFC 6125.

OpenSSL is used as a CA because the Cisco IOS software has a limitation: it cannot generate certificates with an extension that includes an IP address. All certificates are generated by OpenSSL and imported to Android and the Cisco IOS software.

In the Cisco IOS software, the **subject-alt-name** command can be used in order to create an extension that includes an IP address, but the command works only with self-signed certificates. Cisco Bug ID CSCui44783, "IOS ENH PKI ability to generate CSR with subject-alt-name extension," is an enhancement request to allow the Cisco IOS software to generate the extension for all types of enrollments.

This is an example of the commands that generate a CA:

```
#generate key
openssl genrsa -des3 -out ca.key 2048

#generate CSR
openssl req -new -key ca.key -out ca.csr

#remove protection
cp ca.key ca.key.org
openssl rsa -in ca.key.org -out ca.key

#self sign certificate
openssl x509 -req -days 365 -in ca.csr -signkey ca.key -out ca.crt
   -extensions v3_req -extfile conf_global.crt
```

conf_global.crt is a configuration file. The CA extension should be set to TRUE:

```
[ req ]
default_bits              = 1024             # Size of keys
default_md                = md5                      # message digest algorithm
string_mask               = nombstr          # permitted characters
#string_mask              = pkix     # permitted characters
distinguished_name        = req_distinguished_name
req_extensions            = v3_req

[ v3_req ]
basicConstraints          = CA:TRUE
subjectKeyIdentifier      = hash
```

The commands that generate a certificate are very similar for Cisco IOS software and Android. This example assumes that there is already a CA used to sign the certificate:

```
#generate key
openssl genrsa -des3 -out server.key 2048

#generate CSR
openssl req -new -key server.key -out server.csr

#remove protection
cp server.key server.key.org
openssl rsa -in server.key.org -out server.key

#sign the cert and add Alternate Subject Name extension from
   conf_global_cert.crt file with configuration
openssl x509 -req -in server.csr -CA ca.crt -CAkey ca.key -CAcreateserial
   -out server.crt -days 365 -extensions v3_req -extfile conf_global_cert.crt

#create pfx file containig CA cert and server cert
openssl pkcs12 -export -out server.pfx -inkey server.key -in server.crt
   -certfile ca.crt
```

conf_global_cert.crt is a configuration file. The Alternate Subject Name extension is a key setting. In this example, the CA extension is set to FALSE:

```
[ req ]
default_bits              = 1024             # Size of keys
default_md                = md5                      # message digest algorithm
string_mask               = nombstr          # permitted characters
#string_mask              = pkix     # permitted characters
distinguished_name        = req_distinguished_name
req_extensions            = v3_req

[ v3_req ]
basicConstraints          = CA:FALSE
subjectKeyIdentifier      = hash
subjectAltName            = @alt_names

[alt_names]
IP.1                      = 10.48.64.15
```

A certificate should be generated for both the Cisco IOS software and Android.

The IP address 10.48.64.15 belongs to the Cisco IOS software gateway. When you generate a certificate for the Cisco IOS software, make sure the subjectAltName is set to 10.48.64.15. Android validates the certificate received from Cisco IOS software and tries to find its IP address in the subjectAltName.

Cisco IOS Software

The Cisco IOS software needs to have a correct certificate installed for both RSA-based and EAP-based authentication.

The pfx file (which is a pkcs12 container) for the Cisco IOS software can be imported:

```
BSAN-2900-1(config)# crypto pki import TP pkcs12
   http://10.10.10.1/server.pfx password 123456
% Importing pkcs12...
Source filename [server.pfx]?
CRYPTO_PKI: Imported PKCS12 file successfully.
```

Use the **show crypto pki certificates verbose** command in order to verify that the import succeeded:

```
BSAN-2900-1# show crypto pki certificates verbose
Certificate
  Status: Available
  Version: 3
  Certificate Serial Number (hex): 00A003C5DCDEFA146C
  Certificate Usage: General Purpose
  Issuer:
    cn=Cisco
    ou=Cisco TAC
    o=Cisco
    l=Krakow
    st=Malopolskie
    c=PL
  Subject:
    Name: IOS
    IP Address: 10.48.64.15
    cn=IOS
    ou=TAC
    o=Cisco
    l=Krakow
    st=Malopolska
    c=PL
  Validity Date:
    start date: 18:04:09 UTC Aug 1 2013
    end   date: 18:04:09 UTC Aug 1 2014
  Subject Key Info:
    Public Key Algorithm: rsaEncryption
    RSA Public Key: (2048 bit)
  Signature Algorithm: SHA1 with RSA Encryption
  Fingerprint MD5: 2C45BF10 0BACB98D 444F5804 1DC27ECF
  Fingerprint SHA1: 26B66A66 DF5E7D6F 498DD653 A2C164D7 4C7A7F8F
  X509v3 extensions:
    X509v3 Subject Key ID: AD598A9B 8AB6893B AB3CB8B9 28B2039C 78441E72
    X509v3 Basic Constraints:
        CA: FALSE
    X509v3 Subject Alternative Name:

        10.48.64.15
    Authority Info Access:
  Associated Trustpoints: TP
  Storage: nvram:Cisco#146C.cer
  Key Label: TP
  Key storage device: private config

CA Certificate
  Status: Available
  Version: 3
  Certificate Serial Number (hex): 00DC8EAD98723DF56A
  Certificate Usage: General Purpose
  Issuer:
    cn=Cisco
    ou=Cisco TAC
    o=Cisco
    l=Krakow
    st=Malopolskie
    c=PL
  Subject:
    cn=Cisco
    ou=Cisco TAC
    o=Cisco
    l=Krakow
    st=Malopolskie
    c=PL
  Validity Date:
    start date: 16:39:55 UTC Jul 23 2013
    end   date: 16:39:55 UTC Jul 23 2014
  Subject Key Info:
    Public Key Algorithm: rsaEncryption
    RSA Public Key: (2048 bit)
  Signature Algorithm: SHA1 with RSA Encryption
  Fingerprint MD5: 0A2432DC 33F0DC46 AAB23E26 ED474B7E
  Fingerprint SHA1: A50E3892 ED5C4542 FA7FF584 DE07B6E0 654A62D0
  X509v3 extensions:
    X509v3 Subject Key ID: 786F263C 0F5A1963 D6AD18F8 86DCE7C9 0185911E
    X509v3 Basic Constraints:
        CA: TRUE
    Authority Info Access:
  Associated Trustpoints: TP
  Storage: nvram:Cisco#F56ACA.cer

BSAN-2900-1#show ip int brief
Interface                  IP-Address      OK? Method Status                Protocol
GigabitEthernet0/0         10.48.64.15     YES NVRAM  up                    up
```

Android

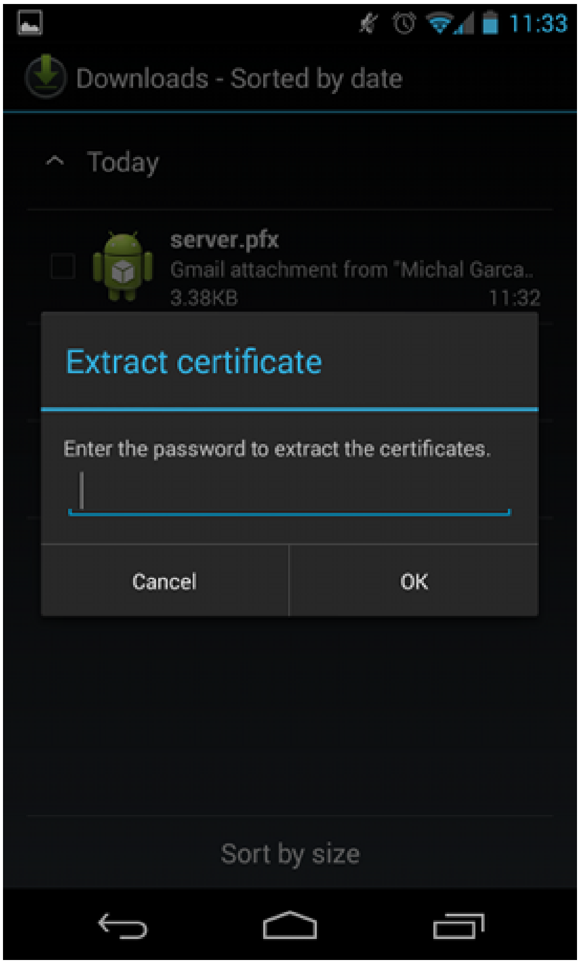
For EAP-based authentication, Andorid needs to have just the correct CA certificate installed.

For RSA-based authentication, Andorid needs to have both the CA certificate and its own certificate installed.

This procedure describes how to install both certificates:

1. Send the pfx file by email, and open it.

2. Provide the password that was used when the pfx file was generated.

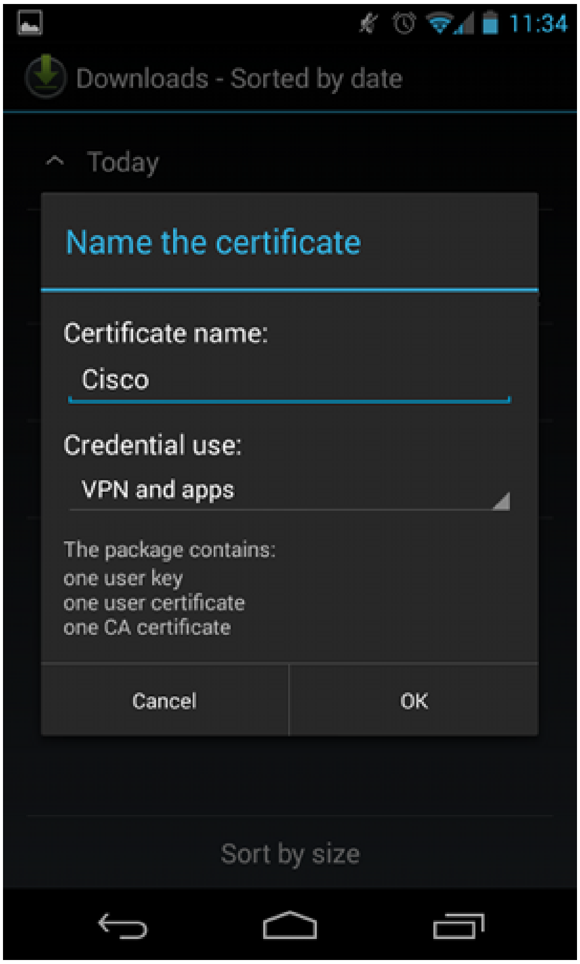


server.pfx
Gmail attachment from "Michal Garca..
3.38KB 11:32

Extract certificate

Enter the password to extract the certificates.

3. Provide the name for the imported certificate.



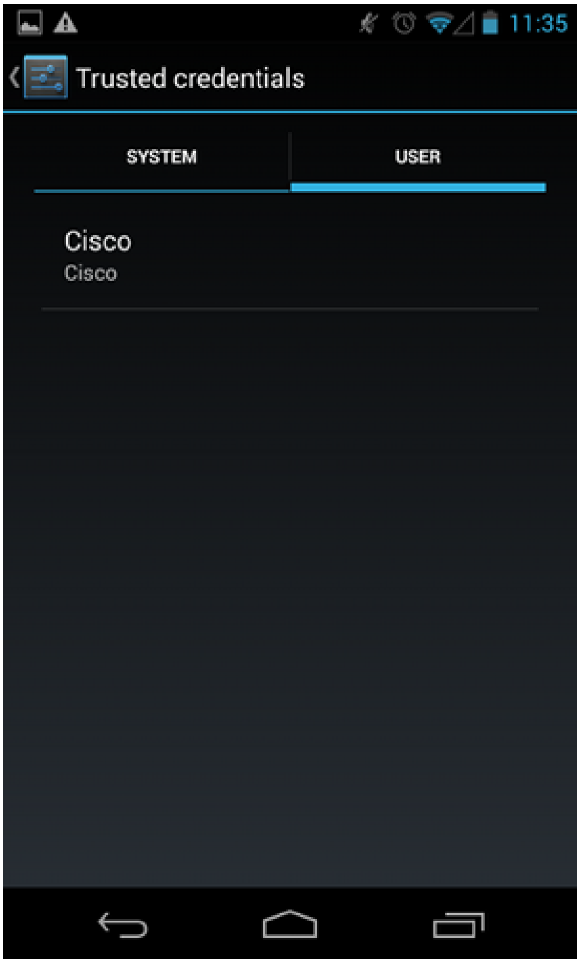
Name the certificate

Certificate name:
Cisco

Credential use:
VPN and apps

The package contains:
one user key
one user certificate
one CA certificate

4. Navigate to **Settings** > **Security** > **Trusted Credentials** in order to verify the certificate installation. The new certificate should appear in the user store:



At this point, a user certificate as well as a CA certificate are installed. The pfx file is a pkcs12 container with both the user certificate and the CA certificate.

Android has precise requirements when certificates are imported. For example, for a CA certificate to be imported successfully, Android requires that the x509v3 extension Basic Constraint CA be set to TRUE. Thus, when you generate a CA or use your own CA, it is important to verify that it has the correct extension:

```
pluton custom_ca # openssl x509 -in ca.crt -text
Certificate:
    Data&colon;
        Version: 3 (0x2)
        Serial Number:
            dc:8e:ad:98:72:3d:f5:6a
        Signature Algorithm: sha1WithRSAEncryption
        Issuer: C=PL, ST=Malopolskie, L=Krakow, O=Cisco, OU=Cisco TAC, CN=Cisco
<.....output omitted>

 X509v3 Basic Constraints:
               CA:TRUE

<.....output omitted>
```

EAP Authentication

Cisco IOS Software Configuration for EAP Authentication

IKEv2 allows the use of an EAP protocol stack in order to perform user authentication. The VPN gateway presents itself with the certificate. Once the client trusts that certificate, the client responds to the EAP request identity from the gateway. The Cisco IOS software uses that identity and sends a Radius-Request message to the authentication, authorization, and accounting (AAA) server, and an EAP-MD5 session is established between the supplicant (Android) and the authentication server (Access Control Server [ACS] or ISE).

After successful EAP-MD5 authentication, as indicated by a Radius-Accept message, the Cisco IOS software uses the configuration mode in order to push the IP address to the client and continue with traffic selector negotiation.

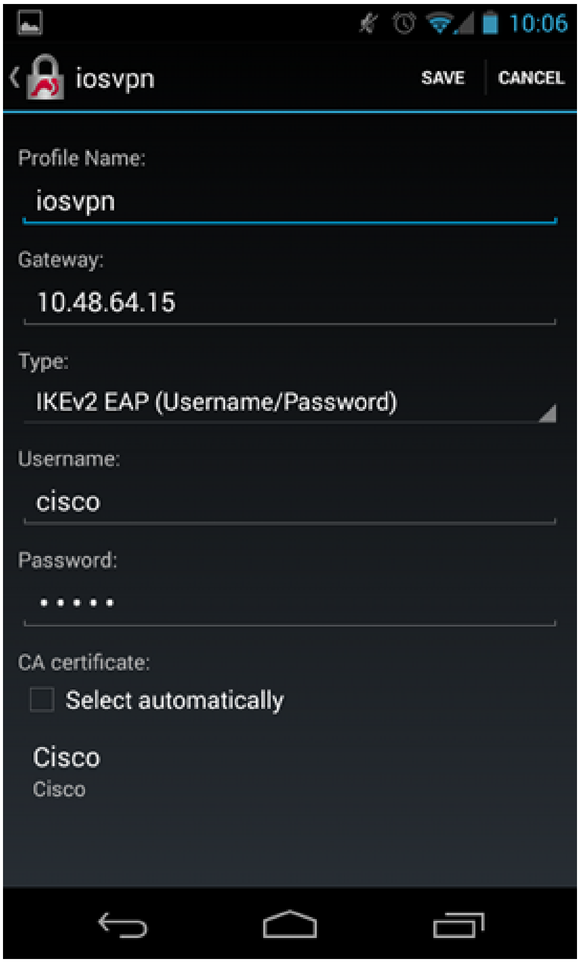
Notice that Android has sent IKEID=cisco (as configured). This IKEID received on the Cisco IOS software matches 'ikev2 profile PROF'.

```
aaa new-model
aaa authentication login eap-list-radius group radius
aaa authorization network IKE2_AUTHOR_LOCAL local

crypto pki trustpoint TP
 revocation-check none

crypto ikev2 authorization policy IKE2_AUTHOR_POLICY
 pool POOL
!
crypto ikev2 proposal ikev2-proposal
 encryption aes-cbc-128
 integrity sha1
 group 14
!
crypto ikev2 policy ikev2-policy
 proposal ikev2-proposal
!
!
crypto ikev2 profile PROF
 match identity remote key-id cisco
 authentication remote eap query-identity
 authentication local rsa-sig
 pki trustpoint TP
 aaa authentication eap eap-list-radius
 aaa authorization group eap list IKE2_AUTHOR_LOCAL IKE2_AUTHOR_POLICY
 aaa authorization user eap cached
 virtual-template 1

crypto ipsec transform-set 3DES-MD5 esp-aes esp-sha-hmac
 mode tunnel
!
crypto ipsec profile PROF
 set transform-set 3DES-MD5
 set ikev2-profile PROF

interface GigabitEthernet0/0
 ip address 10.48.64.15 255.255.255.128

interface Virtual-Template1 type tunnel
 ip unnumbered GigabitEthernet0/0
 tunnel mode ipsec ipv4
 tunnel protection ipsec profile PROF

ip local pool POOL 192.168.0.1 192.168.0.10

radius-server host 10.48.66.185 key cisco
```

Android Configuration for EAP Authentication

Android strongSwan must have EAP configured:

1. Disable automatic certificate selection; otherwise, 100 or more CERT_REQs are sent in the third packet.
2. Choose a specific certificate (CA) that was imported in the previous step; the username and password should be the same as on the AAA server.



Profile Name: iosvpn

Gateway: 10.48.64.15

Type: IKEv2 EAP (Username/Password)

Username: cisco

Password: •••••

CA certificate:

□ Select automatically

Cisco
Cisco

EAP Authentication Test

In the Cisco IOS software, these are the most important debugs for EAP authentication. Most output has been omitted for clarity:

```
debug crypto ikev2 error
debug crypto ikev2 internal
debug radius authentication
debug radius verbose

IKEv2:New ikev2 sa request admitted
IKEv2:(SA ID = 1):Searching policy based on peer's identity 'cisco' of type
   'FQDN'
IKEv2:(1): Choosing IKE profile PROF
IKEv2:Sending certificates as X509 certificates

RADIUS(00000025): Send Access-Request to 10.48.66.185:1645 id 1645/4,len 110
RADIUS: Received from id 1645/4 10.48.66.185:1645, Access-Challenge, len 79
RADIUS(00000025): Send Access-Request to 10.48.66.185:1645 id 1645/5,len 141
```

```
RADIUS: Received from id 1645/5 10.48.66.185:1645, Access-Challenge, len 100
RADIUS(00000025): Send Access-Request to 10.48.66.185:1645 id 1645/6,len 155
RADIUS: Received from id 1645/6 10.48.66.185:1645, Access-Accept, len 76

IKEv2:(SA ID = 1):SM Trace-> SA: I_SPI=AABAB198FACAAEDE R_SPI=D61F37C4DC875001
   (R) MsgID = 00000004 CurState: R_PROC_EAP_RESP Event: EV_RECV_EAP_SUCCESS

IKEv2:IKEv2 local AAA author request for 'IKE2_AUTHOR_POLICY'
IKEv2:Received group author attributes:  ipv4-pool: POOL, route-accept any tag:1
   distance:1
IKEv2:Allocated addr 192.168.0.2 from local pool POOL
IKEv2:(SA ID = 1):SM Trace-> SA: I_SPI=AABAB198FACAAEDE R_SPI=D61F37C4DC875001
   (R) MsgID = 00000005 CurState: R_VERIFY_AUTH Event:
   EV_OK_RECD_VERIFY_IPSEC_POLICY
%LINEPROTO-5-UPDOWN: Line protocol on Interface Virtual-Access1, changed state
   to up
```

The Android logs indicate:

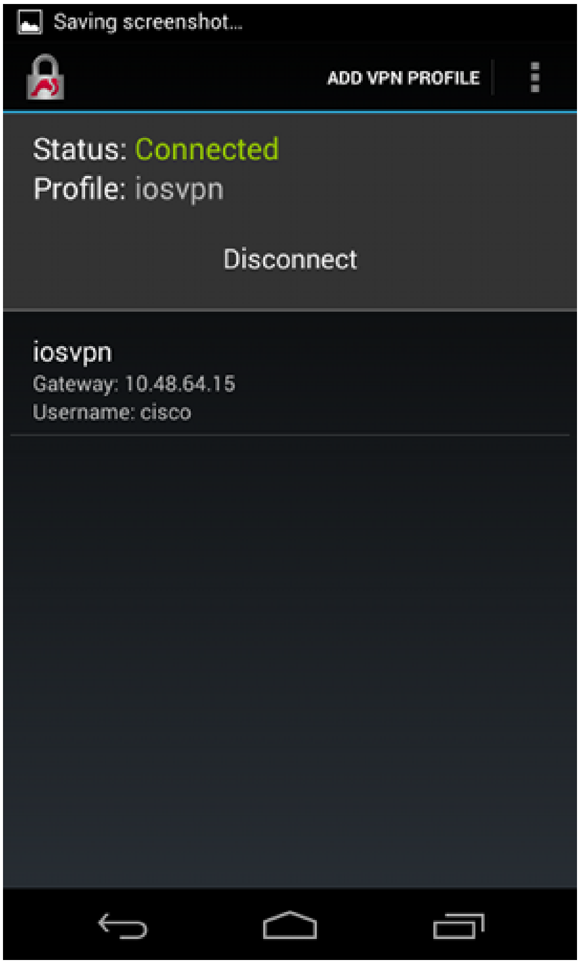
```
00[DMN] Starting IKE charon daemon (strongSwan 5.1.0dr2,
   Linux 3.4.0-perf-gf43c3d9, armv7l)
00[KNL] kernel-netlink plugin might require CAP_NET_ADMIN capability
00[LIB] loaded plugins: androidbridge charon android-log openssl fips-prf
   random nonce pubkey pkcs1 pkcs8 pem xcbc hmac socket-default kernel-netlink
00[LIB] unable to load 9 plugin features (9 due to unmet dependencies)
00[JOB] spawning 16 worker threads
13[IKE] initiating IKE_SA android[1] to 10.48.64.15
13[ENC] generating IKE_SA_INIT request 0 [ SA KE No N(NATD_S_IP) N(NATD_D_IP) ]
13[NET] sending packet: from 10.147.24.153[45581] to 10.48.64.15[500]
   (648 bytes)
11[NET] received packet: from 10.48.64.15[500] to 10.147.24.153[45581]
   (497 bytes)
11[ENC] parsed IKE_SA_INIT response 0 [ SA KE No V V N(NATD_S_IP) N(NATD_D_IP)
   CERTREQ N(HTTP_CERT_LOOK) ]
11[ENC] received unknown vendor ID:
   43:49:53:43:4f:2d:44:45:4c:45:54:45:2d:52:45:41:53:4f:4e
11[ENC] received unknown vendor ID:
   46:4c:45:58:56:50:4e:2d:53:55:50:50:4f:52:54:45:44
11[IKE] faking NAT situation to enforce UDP encapsulation
11[IKE] cert payload ANY not supported - ignored
11[IKE] sending cert request for "C=PL, ST=Malopolskie, L=Krakow, O=Cisco,
   OU=Cisco TAC, CN=Cisco"
11[IKE] establishing CHILD_SA android
11[ENC] generating IKE_AUTH request 1 [ IDi N(INIT_CONTACT) CERTREQ
   CP(ADDR ADDR6 DNS DNS6) N(ESP_TFC_PAD_N) SA TSi TSr N(MOBIKE_SUP)
11[NET] sending packet: from 10.147.24.153[35564] to 10.48.64.15[4500]
   (508 bytes)
10[NET] received packet: from 10.48.64.15[4500] to 10.147.24.153[35564]
   (1292 bytes)
10[ENC] parsed IKE_AUTH response 1 [ V IDr CERT AUTH EAP/REQ/ID ]
10[IKE] received end entity cert "C=PL, ST=Malopolska, L=Krakow, O=Cisco,
   OU=TAC, CN=IOS"
10[CFG] using certificate "C=PL, ST=Malopolska, L=Krakow, O=Cisco, OU=TAC,
   CN=IOS"
10[CFG] using trusted ca certificate "C=PL, ST=Malopolskie, L=Krakow, O=Cisco,
   OU=Cisco TAC, CN=Cisco"
10[CFG] reached self-signed root ca with a path length of 0
10[IKE] authentication of '10.48.64.15' with RSA signature successful
10[IKE] server requested EAP_IDENTITY (id 0x3B), sending 'cisco'
10[ENC] generating IKE_AUTH request 2 [ EAP/RES/ID ]
10[NET] sending packet: from 10.147.24.153[35564] to 10.48.64.15[4500]
   (76 bytes)
09[NET] received packet: from 10.48.64.15[4500] to 10.147.24.153[35564]
   (76 bytes)
09[ENC] parsed IKE_AUTH response 2 [ EAP/REQ/TLS ]
09[IKE] server requested EAP_TLS authentication (id 0x59)
```

```
09[IKE] EAP method not supported, sending EAP_NAK
09[ENC] generating IKE_AUTH request 3 [ EAP/RES/NAK ]
09[NET] sending packet: from 10.147.24.153[35564] to 10.48.64.15[4500]
   (76 bytes)
08[NET] received packet: from 10.48.64.15[4500] to 10.147.24.153[35564]
   (92 bytes)
08[ENC] parsed IKE_AUTH response 3 [ EAP/REQ/MD5 ]
08[IKE] server requested EAP_MD5 authentication (id 0x5A)
08[ENC] generating IKE_AUTH request 4 [ EAP/RES/MD5 ]
08[NET] sending packet: from 10.147.24.153[35564] to 10.48.64.15[4500]
   (92 bytes)
07[NET] received packet: from 10.48.64.15[4500] to 10.147.24.153[35564]
   (76 bytes)
07[ENC] parsed IKE_AUTH response 4 [ EAP/SUCC ]
07[IKE] EAP method EAP_MD5 succeeded, no MSK established
07[IKE] authentication of 'cisco' (myself) with EAP
07[ENC] generating IKE_AUTH request 5 [ AUTH ]
07[NET] sending packet: from 10.147.24.153[35564] to 10.48.64.15[4500]
   (92 bytes)
06[NET] received packet: from 10.48.64.15[4500] to 10.147.24.153[35564]
   (236 bytes)
06[ENC] parsed IKE_AUTH response 5 [ AUTH CP(ADDR) SA TSi TSr N(SET_WINSIZE)
   N(ESP_TFC_PAD_N) N(NON_FIRST_FRAG) ]
06[IKE] authentication of '10.48.64.15' with EAP successful
06[IKE] IKE_SA android[1] established between
   10.147.24.153[cisco]...10.48.64.15[10.48.64.15]
06[IKE] scheduling rekeying in 35421s
06[IKE] maximum IKE_SA lifetime 36021s
06[IKE] installing new virtual IP 192.168.0.1
06[IKE] received ESP_TFC_PADDING_NOT_SUPPORTED, not using ESPv3 TFC padding
06[IKE] CHILD_SA android{1} established with SPIs c776cb4f_i ea27f072_o and
   TS 192.168.0.1/32 === 0.0.0.0/0
06[DMN] setting up TUN device for CHILD_SA android{1}
06[DMN] successfully created TUN device
```

This example shows how to verify status on the Cisco IOS software:

```
BSAN-2900-1#show crypto session detail
Crypto session current status

Code: C - IKE Configuration mode, D - Dead Peer Detection
K - Keepalives, N - NAT-traversal, T - cTCP encapsulation
X - IKE Extended Authentication, F - IKE Fragmentation

Interface: Virtual-Access1
Uptime: 00:02:12
Session status: UP-ACTIVE
Peer: 10.147.24.153 port 60511 fvrf: (none) ivrf: (none)
      Phase1_id: cisco
      Desc: (none)
  IKEv2 SA: local 10.48.64.15/4500 remote 10.147.24.153/60511 Active
          Capabilities:NX connid:1 lifetime:23:57:48
  IPSEC FLOW: permit ip 0.0.0.0/0.0.0.0 host 192.168.0.2
        Active SAs: 2, origin: crypto map
        Inbound:  #pkts dec'ed 40 drop 0 life (KB/Sec) 4351537/3468
        Outbound: #pkts enc'ed 5 drop 0 life (KB/Sec) 4351542/3468

BSAN-2900-1#show crypto ikev2 sa detailed
 IPv4 Crypto IKEv2  SA

Tunnel-id Local                 Remote                fvrf/ivrf         Status
1         10.48.64.15/4500      10.147.24.153/60511   none/none         READY
      Encr: AES-CBC, keysize: 128, Hash: SHA96, DH Grp:14, Auth sign: RSA,
         Auth verify: EAP
      Life/Active Time: 86400/137 sec
      CE id: 1002, Session-id: 2
      Status Description: Negotiation done
      Local spi: D61F37C4DC875001       Remote spi: AABAB198FACAAEDE
      Local id: 10.48.64.15
      Remote id: cisco
      Remote EAP id: cisco
      Local req msg id:  0              Remote req msg id:  6
      Local next msg id: 0              Remote next msg id: 6
      Local req queued:  0              Remote req queued:  6
      Local window:      5              Remote window:      1
      DPD configured for 0 seconds, retry 0
      Fragmentation not configured.
      Extended Authentication configured.
      NAT-T is detected  outside
      Cisco Trust Security SGT is disabled
      Assigned host addr: 192.168.0.2
      Initiator of SA : No
```

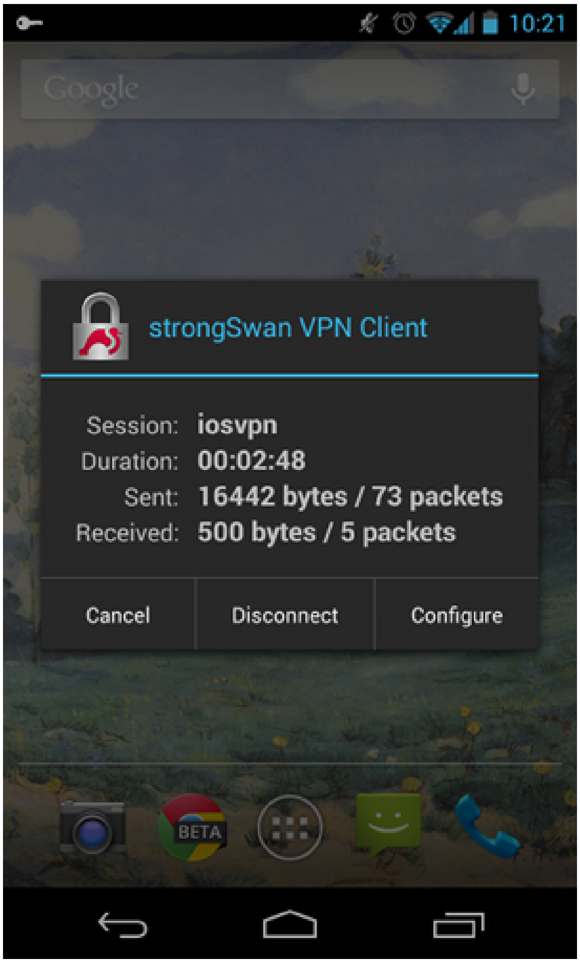
These figures show how to verify status on Android:



Saving screenshot...

Status: Connected
Profile: iosvpn

iosvpn
Gateway: 10.48.64.15
Username: cisco



strongSwan VPN Client

Session: iosvpn
Duration: 00:02:48
Sent: 16442 bytes / 73 packets
Received: 500 bytes / 5 packets

RSA Authentication

Cisco IOS Software Configuration for RSA Authentication

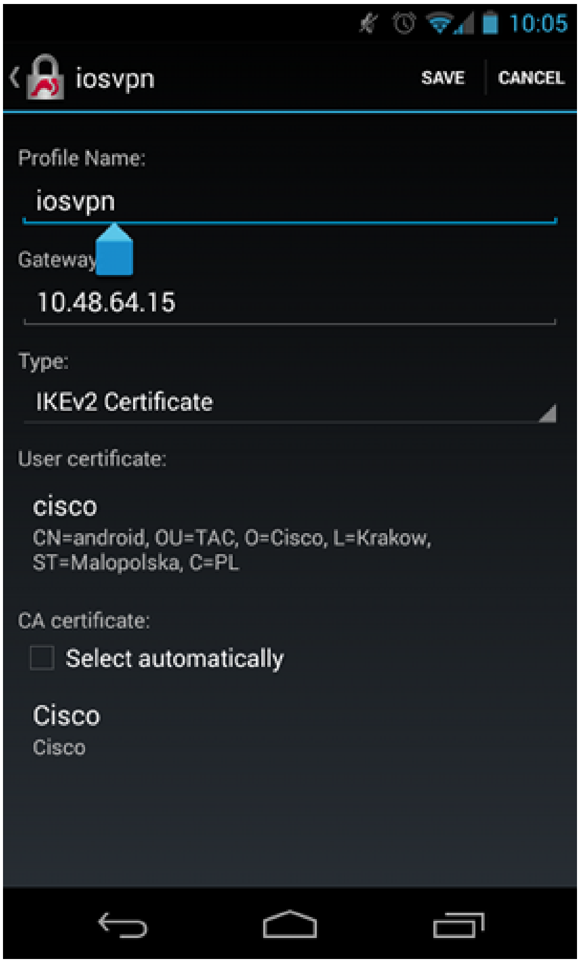
In Rivest-Shamir-Adleman (RSA) authentication, Android sends the certificate in order to authenticate to the Cisco IOS software. That is why the certificate map that binds that traffic to a specific IKEv2 profile is needed. User EAP authentication is not required.

This is an example of how RSA authentication for a remote peer is set:

```
crypto pki certificate map CERT_MAP 10
 subject-name co android

crypto ikev2 profile PROF
 match certificate CERT_MAP
 authentication remote rsa-sig
 authentication local rsa-sig
 pki trustpoint TP
 aaa authorization group cert list IKE2_AUTHOR_LOCAL IKE2_AUTHOR_POLICY
 virtual-template 1
```

Android Configuration for RSA Authentication

User credentials have been replaced by the user certificate:



Profile Name:
iosvpn

Gateway:
10.48.64.15

Type:
IKEv2 Certificate

User certificate:
cisco
CN=android, OU=TAC, O=Cisco, L=Krakow, ST=Malopolska, C=PL

CA certificate:
□ Select automatically

Cisco
Cisco

RSA Authentication Test

In the Cisco IOS software, these are the most important debugs for RSA authentication. Most output has been omitted for clarity:

```
debug crypto ikev2 error
debug crypto ikev2 internal
debug crypto pki transactions
debug crypto pki validation
debug crypto pki messages

IKEv2:New ikev2 sa request admitted
IKEv2:(SA ID = 1):Searching policy based on peer's identity 'cn=android,ou=TAC,
   o=Cisco,l=Krakow,st=Malopolska,c=PL' of type 'DER ASN1 DN'
IKEv2:(1): Choosing IKE profile PROF
IKEv2:Sending certificates as X509 certificates
IKEv2:(SA ID = 1):Peer's authentication method is 'RSA'
IKEv2:Peer has sent X509 certificates
CRYPTO_PKI: Found a issuer match
CRYPTO_PKI: (9000B) Certificate is verified
CRYPTO_PKI: (9000B) Certificate validation succeeded
IKEv2:(SA ID = 1):[Crypto Engine -> IKEv2] Verification of signed
   authentication data PASSED

IKEv2:IKEv2 local AAA author request for 'IKE2_AUTHOR_POLICY'
IKEv2:Received group author attributes:  ipv4-pool: POOL, route-accept any tag:1
   distance:1
IKEv2:Allocated addr 192.168.0.3 from local pool POOL
IKEv2:(SA ID = 1):SM Trace-> SA: I_SPI=E53A57E359A8437C R_SPI=A03D273FC75EEBD9
(R) MsgID = 00000001 CurState: R_VERIFY_AUTH Event:
   EV_OK_RECD_VERIFY_IPSEC_POLICY
```

```
%LINEPROTO-5-UPDOWN: Line protocol on Interface Virtual-Access1, changed state
   to up
```

The Android logs indicate:

```
00[DMN] Starting IKE charon daemon (strongSwan 5.1.0dr2,
   Linux 3.4.0-perf-gf43c3d9, armv7l)
00[KNL] kernel-netlink plugin might require CAP_NET_ADMIN capability
00[LIB] loaded plugins: androidbridge charon android-log openssl fips-prf
   random nonce pubkey pkcs1 pkcs8 pem xcbc hmac socket-default
00[LIB] unable to load 9 plugin features (9 due to unmet dependencies)
00[JOB] spawning 16 worker threads
05[CFG] loaded user certificate 'C=PL, ST=Malopolska, L=Krakow, O=Cisco,
   OU=TAC, CN=android' and private key
05[CFG] loaded CA certificate 'C=PL, ST=Malopolskie, L=Krakow, O=Cisco,
OU=Cisco TAC, CN=Cisco'

05[IKE] initiating IKE_SA android[4] to 10.48.64.15
05[ENC] generating IKE_SA_INIT request 0 [ SA KE No N(NATD_S_IP) N(NATD_D_IP) ]
05[NET] sending packet: from 10.147.24.153[34697] to 10.48.64.15[500]
   (648 bytes)
10[NET] received packet: from 10.48.64.15[500] to 10.147.24.153[34697]
   (497 bytes)
10[ENC] parsed IKE_SA_INIT response 0 [ SA KE No V V N(NATD_S_IP) N(NATD_D_IP)
   CERTREQ N(HTTP_CERT_LOOK) ]
10[ENC] received unknown vendor ID:
   43:49:53:43:4f:2d:44:45:4c:45:54:45:2d:52:45:41:53:4f:4e
10[ENC] received unknown vendor ID:
   46:4c:45:58:56:50:4e:2d:53:55:50:50:4f:52:54:45:44
10[IKE] faking NAT situation to enforce UDP encapsulation
10[IKE] cert payload ANY not supported - ignored
10[IKE] sending cert request for "C=PL, ST=Malopolskie, L=Krakow, O=Cisco,
   OU=Cisco TAC, CN=Cisco"
10[IKE] authentication of 'C=PL, ST=Malopolska, L=Krakow, O=Cisco, OU=TAC,
   CN=android' (myself) with RSA signature successful
10[IKE] sending end entity cert "C=PL, ST=Malopolska, L=Krakow, O=Cisco,
   OU=TAC, CN=android"
10[IKE] establishing CHILD_SA android
10[ENC] generating IKE_AUTH request 1 [ IDi CERT N(INIT_CONTACT) CERTREQ
   AUTH CP(ADDR ADDR6 DNS DNS6) N(ESP_TFC_PAD_N) SA
10[NET] sending packet: from 10.147.24.153[44527] to 10.48.64.15[4500]
   (1788 bytes)
12[NET] received packet: from 10.48.64.15[4500] to 10.147.24.153[44527]
   (1420 bytes)
12[ENC] parsed IKE_AUTH response 1 [ V IDr CERT AUTH CP(ADDR) SA TSi TSr
   N(SET_WINSIZE) N(ESP_TFC_PAD_N) N(NON_FIRST_FRAG)
12[IKE] received end entity cert "C=PL, ST=Malopolska, L=Krakow, O=Cisco,
   OU=TAC, CN=IOS"
12[CFG] using certificate "C=PL, ST=Malopolska, L=Krakow, O=Cisco, OU=TAC,
   CN=IOS"
12[CFG] using trusted ca certificate "C=PL, ST=Malopolskie, L=Krakow, O=Cisco,
   OU=Cisco TAC, CN=Cisco"
12[CFG] reached self-signed root ca with a path length of 0
12[IKE] authentication of '10.48.64.15' with RSA signature successful
12[IKE] IKE_SA android[4] established between 10.147.24.153[C=PL,
   ST=Malopolska, L=Krakow, O=Cisco, OU=TAC,
   CN=android]...10.48.64.15[10.48.64.15]
12[IKE] scheduling rekeying in 35413s
12[IKE] maximum IKE_SA lifetime 36013s
12[IKE] installing new virtual IP 192.168.0.3
12[IKE] received ESP_TFC_PADDING_NOT_SUPPORTED, not using ESPv3 TFC padding
12[IKE] CHILD_SA android{4} established with SPIs ecb3af87_i b2279175_o and
   TS 192.168.0.3/32 === 0.0.0.0/0
12[DMN] setting up TUN device for CHILD_SA android{4}
12[DMN] successfully created TUN device
```

In the Cisco IOS software, RSA is used for both signing and verification; in the previous scenario, EAP was used for verification:

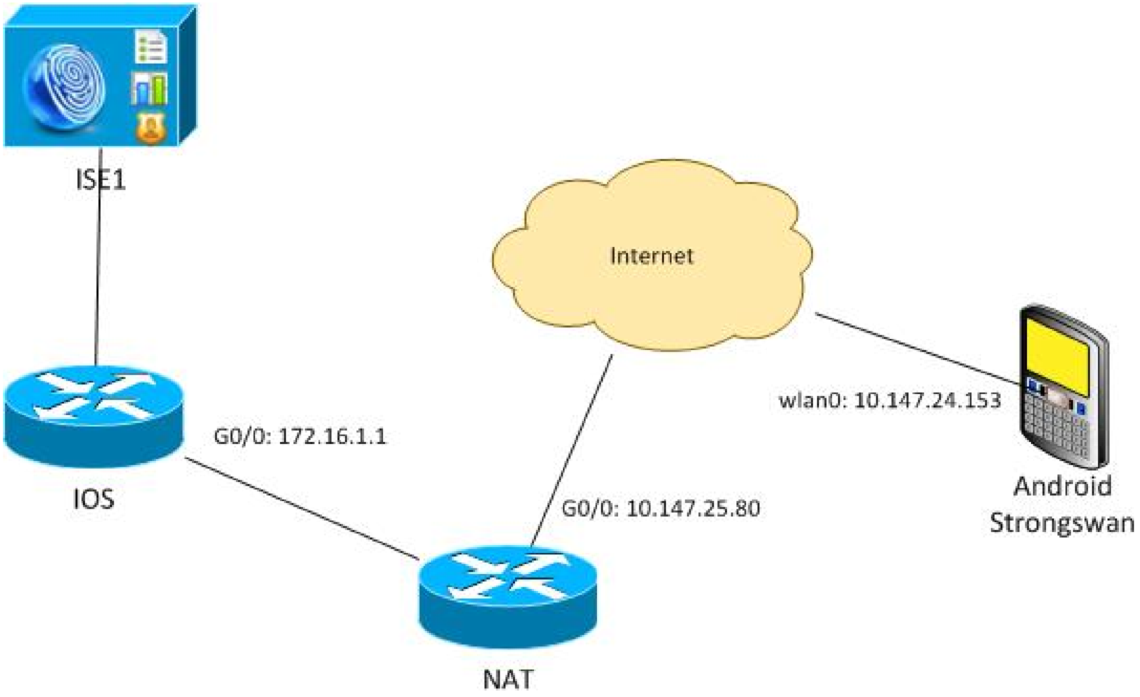
```
BSAN-2900-1#show crypto ikev2 sa detailed
 IPv4 Crypto IKEv2  SA

Tunnel-id Local                 Remote                fvrf/ivrf          Status
1         10.48.64.15/4500      10.147.24.153/44527   none/none          READY
      Encr: AES-CBC, keysize: 128, Hash: SHA96, DH Grp:14, Auth sign: RSA,
         Auth verify: RSA
      Life/Active Time: 86400/16 sec
      CE id: 1010, Session-id: 3
      Status Description: Negotiation done
      Local spi: A03D273FC75EEBD9       Remote spi: E53A57E359A8437C
      Local id: 10.48.64.15
      Remote id: cn=android,ou=TAC,o=Cisco,l=Krakow,st=Malopolska,c=PL
      Local req msg id:  0              Remote req msg id:  2
      Local next msg id: 0              Remote next msg id: 2
      Local req queued:  0              Remote req queued:  2
      Local window:      5              Remote window:      1
      DPD configured for 0 seconds, retry 0
      Fragmentation not configured.
      Extended Authentication not configured.
      NAT-T is detected  outside
      Cisco Trust Security SGT is disabled
      Assigned host addr: 192.168.0.3
      Initiator of SA : No
```

Status verification on Android is similar to that in the previous scenario.

VPN Gateway Behind NAT - strongSwan and Cisco IOS Software Limitations

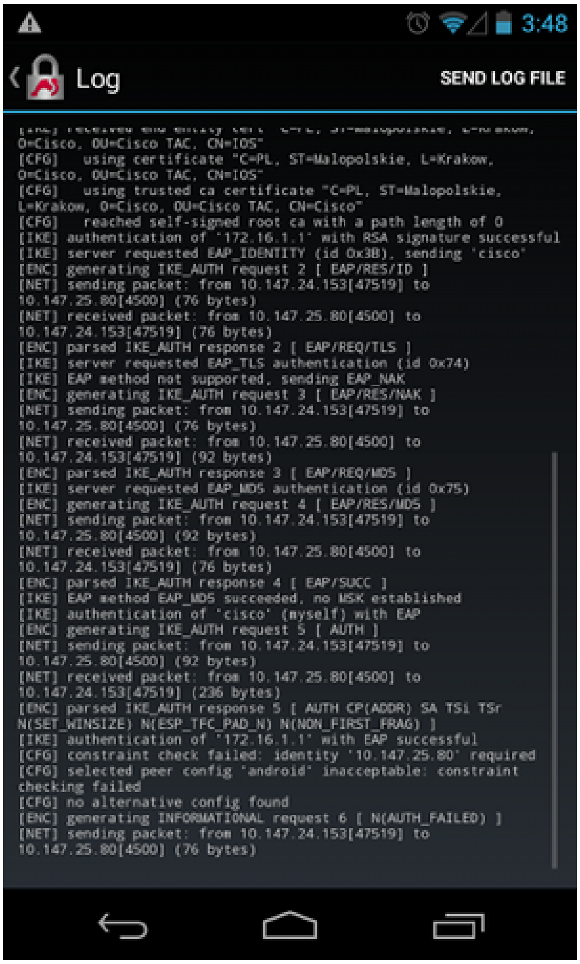
This example explains a limitation of strongSwan certificate verifications.

Assume that the Cisco IOS software VPN gateway IP address is statically translated from 172.16.1.1 to 10.147.25.80. EAP authentication is used.



Assume also that the Cisco IOS software certificate has a Subject Alternative Name for both 172.16.1.1 and 10.147.25.80.

After successful EAP authentication, Android performs verification and tries to find the IP address of the peer that was used in Android configuration (10.147.25.80) in the Subject Alternative Name extension. The verification fails:



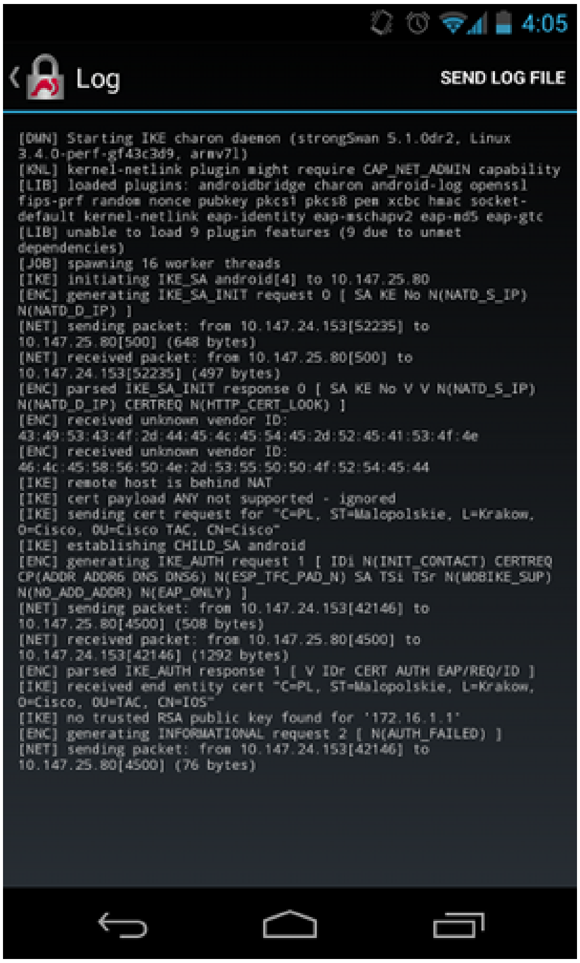
```
[IKE] received end entity cert "C=PL, ST=Malopolskie, L=Krakow,
O=Cisco, OU=Cisco TAC, CN=IOS"
[CFG]   using certificate "C=PL, ST=Malopolskie, L=Krakow,
O=Cisco, OU=Cisco TAC, CN=IOS"
[CFG]   using trusted ca certificate "C=PL, ST=Malopolskie,
L=Krakow, O=Cisco, OU=Cisco TAC, CN=Cisco"
[CFG]   reached self-signed root ca with a path length of 0
[IKE] authentication of '172.16.1.1' with RSA signature successful
[IKE] server requested EAP_IDENTITY (id 0x3B), sending 'cisco'
[ENC] generating IKE_AUTH request 2 [ EAP/RES/ID ]
[NET] sending packet: from 10.147.24.153[47519] to
10.147.25.80[4500] (76 bytes)
[NET] received packet: from 10.147.25.80[4500] to
10.147.24.153[47519] (76 bytes)
[ENC] parsed IKE_AUTH response 2 [ EAP/REQ/TLS ]
[IKE] server requested EAP_TLS authentication (id 0x74)
[IKE] EAP method not supported, sending EAP_NAK
[ENC] generating IKE_AUTH request 3 [ EAP/RES/NAK ]
[NET] sending packet: from 10.147.24.153[47519] to
10.147.25.80[4500] (76 bytes)
[NET] received packet: from 10.147.25.80[4500] to
10.147.24.153[47519] (92 bytes)
[ENC] parsed IKE_AUTH response 3 [ EAP/REQ/MD5 ]
[IKE] server requested EAP_MD5 authentication (id 0x75)
[ENC] generating IKE_AUTH request 4 [ EAP/RES/MD5 ]
[NET] sending packet: from 10.147.24.153[47519] to
10.147.25.80[4500] (92 bytes)
[NET] received packet: from 10.147.25.80[4500] to
10.147.24.153[47519] (76 bytes)
[ENC] parsed IKE_AUTH response 4 [ EAP/SUCC ]
[IKE] EAP method EAP_MD5 succeeded, no MSK established
[IKE] authentication of 'cisco' (myself) with EAP
[ENC] generating IKE_AUTH request 5 [ AUTH ]
[NET] sending packet: from 10.147.24.153[47519] to
10.147.25.80[4500] (92 bytes)
[NET] received packet: from 10.147.25.80[4500] to
10.147.24.153[47519] (236 bytes)
[ENC] parsed IKE_AUTH response 5 [ AUTH CP(ADDR) SA TSi TSr
N(SET_WINSIZE) N(ESP_TFC_PAD_N) N(NON_FIRST_FRAG) ]
[IKE] authentication of '172.16.1.1' with EAP successful
[CFG] constraint check failed: identity '10.147.25.80' required
[CFG] selected peer config 'android' inacceptable: constraint
checking failed
[CFG] no alternative config found
[ENC] generating INFORMATIONAL request 6 [ N(AUTH_FAILED) ]
[NET] sending packet: from 10.147.24.153[47519] to
10.147.25.80[4500] (76 bytes)
```

The logs indicate:

```
constraint check failed: identity '10.147.25.80' required
```

The failure occurred because Android can read only the first Subject Alternative Name extension (172.16.1.1).

Now, assume that the Cisco IOS software certificate has both addresses in Subject Alternative Name but in the reverse order: 10.147.25.80 and 172.16.1.1. Android performs validation when it receives the IKEID, which is the IP address of VPN gateway (172.16.1.1), in the third packet:



```
[DMN] Starting IKE charon daemon (strongSwan 5.1.0dr2, Linux
3.4.0-perf-gf43c3d9, armv7l)
[KNL] kernel-netlink plugin might require CAP_NET_ADMIN capability
[LIB] loaded plugins: androidbridge charon android-log openssl
fips-prf random nonce pubkey pkcs1 pkcs8 pem xcbc hmac socket-
default kernel-netlink eap-identity eap-mschapv2 eap-md5 eap-gtc
[LIB] unable to load 9 plugin features (9 due to unmet
dependencies)
[JOB] spawning 16 worker threads
[IKE] initiating IKE_SA android[4] to 10.147.25.80
[ENC] generating IKE_SA_INIT request 0 [ SA KE No N(NATD_S_IP)
N(NATD_D_IP) ]
[NET] sending packet: from 10.147.24.153[52235] to
10.147.25.80[500] (648 bytes)
[NET] received packet: from 10.147.25.80[500] to
10.147.24.153[52235] (497 bytes)
[ENC] parsed IKE_SA_INIT response 0 [ SA KE No V V N(NATD_S_IP)
N(NATD_D_IP) CERTREQ N(HTTP_CERT_LOOK) ]
[ENC] received unknown vendor ID:
43:49:53:43:4f:2d:44:45:4c:45:54:45:2d:52:45:41:53:4f:4e
[ENC] received unknown vendor ID:
46:4c:45:58:56:50:4e:2d:53:55:50:50:4f:52:54:45:44
[IKE] remote host is behind NAT
[IKE] cert payload ANY not supported - ignored
[IKE] sending cert request for "C=PL, ST=Malopolskie, L=Krakow,
O=Cisco, OU=Cisco TAC, CN=Cisco"
[IKE] establishing CHILD_SA android
[ENC] generating IKE_AUTH request 1 [ IDi N(INIT_CONTACT) CERTREQ
CP(ADDR ADDR6 DNS DNS6) N(ESP_TFC_PAD_N) SA TSi TSr N(MOBIKE_SUP)
N(NO_ADD_ADDR) N(EAP_ONLY) ]
[NET] sending packet: from 10.147.24.153[42146] to
10.147.25.80[4500] (508 bytes)
[NET] received packet: from 10.147.25.80[4500] to
10.147.24.153[42146] (1292 bytes)
[ENC] parsed IKE_AUTH response 1 [ V IDr CERT AUTH EAP/REQ/ID ]
[IKE] received end entity cert "C=PL, ST=Malopolskie, L=Krakow,
O=Cisco, OU=TAC, CN=IOS"
[IKE] no trusted RSA public key found for '172.16.1.1'
[ENC] generating INFORMATIONAL request 2 [ N(AUTH_FAILED) ]
[NET] sending packet: from 10.147.24.153[42146] to
10.147.25.80[4500] (76 bytes)
```

Now the log shows:

```
no trusted RSA public key found for '172.16.1.1'
```

Thus, when Android receives the IKEID, it needs to find the IKEID in the Subject Alternative Name and can use only the first IP address.

Note: In EAP authentication, the IKEID sent by the Cisco IOS software is the IP address by default. In RSA authentication, the IKEID is the certificate DN by default. Use the **identity** command under the ikev2 profile in order to change these values manually.

Verify

Verification and test procedures are available within the configuration examples.

Troubleshoot

This section provides information you can use to troubleshoot your configuration.

strongSwan CA Multiple CERT_REQ

When the certificate setting on strongSwan is Automatic Selection (the default), Android sends CERT_REQ for all trusted certificates in the local store in the third packet . The Cisco IOS software might drop the request because it recognizes a large number of certificate requests as a Denial of Service attack:

```
*Jul 15 07:54:13: IKEv2:number of cert req exceeds the reasonable limit (100)
```

Tunnel Source on DVTI

Although it is fairly common to set the tunnel source on a virtual tunnel interface (VTI), it is not necessary here. Assume the **tunnel source** command is under a dynamic VTI (DVTI):

```
interface Virtual-Template1 type tunnel
 ip unnumbered GigabitEthernet0/0
 tunnel source GigabitEthernet0/0
 tunnel mode ipsec ipv4
 tunnel protection ipsec profile PROF
```

After authentication, if the Cisco IOS software tries to create virtual access interface that is cloned from a virtual template, it returns an error:

```
*Aug  1 13:34:22 IKEv2:Allocated addr 192.168.0.9 from local pool POOL
*Aug  1 13:34:22 IKEv2:(SA ID = 1):Set received config mode data
*Aug  1 13:34:22 IKEv2:% DVTI create request sent for profile PROF with PSH
   index 1
*Aug  1 13:34:22 IKEv2:Failed to process KMI delete SA message with error 4
*Aug  1 13:34:24 IKEv2:Got a packet from dispatcher
*Aug  1 13:34:24 IKEv2:Processing an item off the pak queue
*Aug  1 13:34:24 IKEv2:Negotiation context locked currently in use
```

Two seconds after the failure, the Cisco IOS software receives a retransmitted IKE_AUTH from Android. That packet is dropped.

Cisco IOS Software Bugs and Enhancement Requests

- Cisco Bug ID CSCui46418, "IOS Ikev2 ip address sent as identity for RSA authentication."
  This bug is not a problem, as long as strongSwan can see a correct Subject Alternative Name (the IP address) when it looks for the IKEID in the certificate in order to perform verification.
- Cisco Bug ID CSCui44976, "IOS PKI incorrectly displayed X509v3 extension Subject Alternative Name."
  This bug occurs only when there are multiple IP addresses in the Subject Alternative Name. Only the last IP address is displayed, but that does not impact certificate usage. The whole certificate is sent and processed correctly.
- Cisco Bug ID CSCui44783, "IOS ENH PKI ability to generate CSR with subject-alt-name extension."
- Cisco Bug ID CSCui44335, "ASA ENH Certificate x509 extensions displayed."

Related Information

- **Cisco IOS 15.3 VPN Configuration Guide**
- **Cisco IOS 15.3 Command Reference**
- **Cisco IOS Flex VPN Configuration Guide**
- **Technical Support & Documentation - Cisco Systems**

Updated: Jan 21, 2016 Document ID: 116837