strongSwan - Bug #2238

eap-dynamic with ca certificate constraint for initiator broken

31.01.2017 17:07 - Noel Kuntze

Status:	Closed	**Start date:**	31.01.2017
Priority:	Normal	**Due date:**	
Assignee:	Tobias Brunner	**Estimated time:**	0.00 hour
Category:	libcharon		
Target version:	5.5.2		
Affected version:	5.5.1	**Resolution:**	Fixed

Description

Hello,

When eap-dynamic is used with eap-tls authentication and a ca constraint is set, the latter seems to make authentication fail, even when the intiator's certificate is signed by it.
charon.log is attached.

The important section in the log is here:

```
Tue, 2017-01-31 17:00 12[TLS] <mobile-eap-tls|2> received TLS CertificateVerify handshake (516 byt
es)
Tue, 2017-01-31 17:00 12[CFG] <mobile-eap-tls|2>   using certificate "C=DE, O=ThermiCorp, CN=Therm
is Style"
Tue, 2017-01-31 17:00 12[CFG] <mobile-eap-tls|2>   certificate "C=DE, O=ThermiCorp, CN=Thermis Sty
le" key: 4096 bit RSA
Tue, 2017-01-31 17:00 12[CFG] <mobile-eap-tls|2>   using trusted intermediate ca certificate "C=DE
, ST=Baden-W??rttemberg, O=ThermiCorp, OU=UserCA, CN=ThermiCorp UserCA Level 2"
Tue, 2017-01-31 17:00 12[CFG] <mobile-eap-tls|2> checking certificate status of "C=DE, O=ThermiCor
p, CN=Thermis Style"
Tue, 2017-01-31 17:00 12[CFG] <mobile-eap-tls|2> ocsp check skipped, no ocsp found
Tue, 2017-01-31 17:00 12[CFG] <mobile-eap-tls|2> certificate status is not available
Tue, 2017-01-31 17:00 12[CFG] <mobile-eap-tls|2>   certificate "C=DE, ST=Baden-W??rttemberg, O=The
rmiCorp, OU=UserCA, CN=ThermiCorp UserCA Level 2" key: 4096 bit RSA
Tue, 2017-01-31 17:00 12[CFG] <mobile-eap-tls|2>   using trusted ca certificate "C=DE, ST=Baden-W?
?rttemberg, L=Haslach, O=ThermiCorp, OU=Root CA, CN=ThermiCorp Root CA, E=noel.kuntze@google
mail.com"
Tue, 2017-01-31 17:00 12[CFG] <mobile-eap-tls|2> checking certificate status of "C=DE, ST=Baden-W?
?rttemberg, O=ThermiCorp, OU=UserCA, CN=ThermiCorp UserCA Level 2"
Tue, 2017-01-31 17:00 12[CFG] <mobile-eap-tls|2> ocsp check skipped, no ocsp found
Tue, 2017-01-31 17:00 12[CFG] <mobile-eap-tls|2> certificate status is not available
Tue, 2017-01-31 17:00 12[CFG] <mobile-eap-tls|2>   certificate "C=DE, ST=Baden-W??rttemberg, L=Has
lach, O=ThermiCorp, OU=Root CA, CN=ThermiCorp Root CA, E=noel.kuntze@googlemail.com" key: 40
96 bit RSA
Tue, 2017-01-31 17:00 12[CFG] <mobile-eap-tls|2>   reached self-signed root ca with a path length
of 1
Tue, 2017-01-31 17:00 12[TLS] <mobile-eap-tls|2> verified signature with SHA256/RSA
Tue, 2017-01-31 17:00 12[TLS] <mobile-eap-tls|2> processing TLS ChangeCipherSpec record (1 bytes)
Tue, 2017-01-31 17:00 12[TLS] <mobile-eap-tls|2> buffering 54 bytes, 54 bytes of 85 byte TLS recor
d received
Tue, 2017-01-31 17:00 12[TLS] <mobile-eap-tls|2> sending EAP_TLS acknowledgement packet
Tue, 2017-01-31 17:00 12[ENC] <mobile-eap-tls|2> generating IKE_AUTH response 8 [ EAP/REQ/TLS ]
Tue, 2017-01-31 17:00 12[NET] <mobile-eap-tls|2> sending packet: from 188.68.37.10[4500] to 78.43.
42.57[47525] (67 bytes)
Tue, 2017-01-31 17:00 23[NET] <mobile-eap-tls|2> received packet: from 78.43.42.57[47525] to 188.6
8.37.10[4500] (98 bytes)
Tue, 2017-01-31 17:00 23[ENC] <mobile-eap-tls|2> parsed IKE_AUTH request 9 [ EAP/RES/TLS ]
Tue, 2017-01-31 17:00 23[TLS] <mobile-eap-tls|2> buffering 31 bytes, 85 bytes of 85 byte TLS recor
d received
Tue, 2017-01-31 17:00 23[TLS] <mobile-eap-tls|2> processing buffered TLS Handshake record (80 byte
s)
Tue, 2017-01-31 17:00 23[TLS] <mobile-eap-tls|2> received TLS Finished handshake (12 bytes)
```

```
Tue, 2017-01-31 17:00 23[TLS] <mobile-eap-tls|2> sending TLS ChangeCipherSpec record (1 bytes)
Tue, 2017-01-31 17:00 23[TLS] <mobile-eap-tls|2> sending TLS Finished handshake (12 bytes)
Tue, 2017-01-31 17:00 23[TLS] <mobile-eap-tls|2> sending TLS Handshake record (80 bytes)
Tue, 2017-01-31 17:00 23[TLS] <mobile-eap-tls|2> sending EAP_TLS packet (101 bytes)
Tue, 2017-01-31 17:00 23[ENC] <mobile-eap-tls|2> generating IKE_AUTH response 9 [ EAP/REQ/TLS ]
Tue, 2017-01-31 17:00 23[NET] <mobile-eap-tls|2> sending packet: from 188.68.37.10[4500] to 78.43.
42.57[47525] (162 bytes)
Tue, 2017-01-31 17:00 18[NET] <mobile-eap-tls|2> received packet: from 78.43.42.57[47525] to 188.6
8.37.10[4500] (67 bytes)
Tue, 2017-01-31 17:00 18[ENC] <mobile-eap-tls|2> parsed IKE_AUTH request 10 [ EAP/RES/TLS ]
Tue, 2017-01-31 17:00 18[TLS] <mobile-eap-tls|2> received EAP_TLS acknowledgement packet
Tue, 2017-01-31 17:00 18[IKE] <mobile-eap-tls|2> EAP method EAP_TLS succeeded, MSK established
Tue, 2017-01-31 17:00 18[ENC] <mobile-eap-tls|2> generating IKE_AUTH response 10 [ EAP/SUCC ]
Tue, 2017-01-31 17:00 18[NET] <mobile-eap-tls|2> sending packet: from 188.68.37.10[4500] to 78.43.
42.57[47525] (65 bytes)
Tue, 2017-01-31 17:00 26[NET] <mobile-eap-tls|2> received packet: from 78.43.42.57[47525] to 188.6
8.37.10[4500] (97 bytes)
Tue, 2017-01-31 17:00 26[ENC] <mobile-eap-tls|2> parsed IKE_AUTH request 11 [ AUTH ]
Tue, 2017-01-31 17:00 26[IKE] <mobile-eap-tls|2> authentication of 'C=DE, O=ThermiCorp, CN=Thermis
 Style' with EAP successful
Tue, 2017-01-31 17:00 26[CFG] <mobile-eap-tls|2> constraint check failed: peer not authenticated b
y CA 'C=DE, ST=Baden-W??rttemberg, O=ThermiCorp, OU=UserCA, CN=ThermiCorp UserCA Level 2'
Tue, 2017-01-31 17:00 26[CFG] <mobile-eap-tls|2> selected peer config 'mobile-eap-tls' inacceptabl
e: non-matching authentication done
Tue, 2017-01-31 17:00 26[CFG] <mobile-eap-tls|2> switching to peer config 'mobile-eap-mschapv2'
```

Complete configuration:

swanctl.confswanctl.conf

```
connections {
        mobile-eap-tls {
                version = 2
                dpd_delay = 10
                dpd_timeout = 60
                proposals = chacha20poly1305-prfsha256-newhope128, aes256gcm16-prfsha256-modp4096,
 aes192gcm12-prfsha256-newhope128
                pools = app,app-v6
                send_cert = always
                fragmentation = yes
                local {
                        auth = pubkey
                        id = thermi.strangled.net
                }
                remote {
                        auth = eap-dynamic
                        cacerts = userca.pem
                        id = %any
                }
                children {
                        mobile-eap-tls-all {
                                updown = /usr/lib/strongswan/sudo_updown
                                dpd_action = clear
                                start_action = none
                                close_action = none
                                ipcomp = yes
                                esp_proposals = chacha20poly1305-newhope128, aes192gcm12-ecp521,ae
s128gcm8-aes256gcm16-ecp521,aes256-sha256-ecp521, chacha20poly1305-ecp521
                                local_ts = 0.0.0.0/0,2000::/3
                        }
                }
        }
        mobile-eap-mschapv2 {
                version = 2
                dpd_delay = 10
                dpd_timeout = 60
                proposals = chacha20poly1305-prfsha256-newhope128, aes256gcm16-prfsha256-modp4096,
```

```
 aes192gcm12-prfsha256-newhope128
                pools = app,app-v6
                send_cert = always
                fragmentation = yes
                local {
                        auth = pubkey
                        id = thermi.strangled.net
                }
                remote {
                        auth = eap-mschapv2
                        id = %any
                }
                children {
                        mobile-eap-mschapv2-all {
                                updown = /usr/lib/strongswan/sudo_updown
                                dpd_action = clear
                                start_action = none
                                close_action = none
                                ipcomp = yes
                                esp_proposals = chacha20poly1305-newhope128, aes192gcm12-ecp521,ae
s128gcm8-aes256gcm16-ecp521,aes256-sha256-ecp521, chacha20poly1305-ecp521
                                local_ts = 0.0.0.0/0,2000::/3
                        }
                }
        }
        any-device {
                version = 2
                dpd_delay = 10
                dpd_timeout = 90
                proposals = chacha20poly1305-prfsha256-newhope128, aes256gcm16-prfsha256-modp4096,
 aes192gcm12-prfsha256-newhope128
                send_cert = always
                fragmentation = yes
                unique = replace
                local {
                        auth = pubkey
                        id = thermi.strangled.net
                }
                remote {
                        auth = pubkey
                        cacerts = userca.pem
                        id = %any
                }
                children {
                        transport-mode {
                                mode = transport
                                esp_proposals = chacha20poly1305-newhope128, aes192gcm12-newhope12
8
                                start_action = none
                                updown = /usr/lib/strongswan/transport-updown
                        }
                        tunnel-mode {
                                mode = tunnel
                                esp_proposals = chacha20poly1305-newhope128, aes192gcm12-newhope12
8
                                start_action = none
                                updown = /usr/lib/strongswan/transport-updown
                        }
                }
        }
pools {
        app {
                addrs = 172.16.20.1/24
                dns = 172.16.25.1
        }
        app-v6 {
                addrs = 2a03:4000:13:10a::3-2a03:4000:13:10a::128
```

```
                dns = fdd2:54c4:4c90::1
        }
}
secrets {
        eap-test {
                id = test
                secret = test
        }
```

strongswan.confstrongswan.conf

```
# strongswan.conf - strongSwan configuration file
#
# Refer to the strongswan.conf(5) manpage for details
#
# Configuration changes should be made in the included files

charon-systemd {
        load = random nonce test-vectors af-alg openssl x509 revocation constraints curl pubkey pk
cs1 pkcs7 pkcs8 pem gmp attr kernel-netlink socket-default vici eap-identity eap-gtc eap-mschapv2
eap-radius xauth-generic xauth-eap unity eap-tls eap-ttls chapoly sha3 mgf1 bliss ntru newhope eap
-dynamic
        make_before_break=yes
        retransmit_tries = 6
        retransmit_timeout = 3.0
        retransmit_base = 1.7
        cisco_unity=yes
        dos_protection = yes
        threads = 32
        replay_window = 32
        retry_initiate_interval = 3
        interface_use = ens3
        install_routes = no
        send_vendor_id = yes

        crypto_test {
                on_add = yes
        }

        journal {
                time_format = %a, %Y-%m-%d %R
                default = 0
                append=yes
                ike_name=yes
                flush_line=yes
        }
        filelog {
                /var/log/charon.log {
                        time_format = %a, %Y-%m-%d %R
                        default = 2
                        mgr = 0
                        net = 1
                        enc = 1
                        asn = 1
                        job = 1
                        knl = 1
                        ike_name = yes
                        append = no
                        flush_line = yes
                }
        }
        plugins {
                eap-dynamic {
                        prefer_user = yes
                        preferred = eap-mschapv2, eap-tls
                }
```

```
        }
        tls {
                ciphers = aes128, aes256
                key_exchange = ecdhe-ecdsa, ecdhe-rsa, dhe-rsa, rsa
                mac = sha256, sha384
        }
}

swanctl {
        load = test-vectors sqlite aes des rc2 sha1 sha2 md5 random nonce x509 revocation constrai
nts pubkey pkcs1 pkcs7 pkcs8 pkcs12 pgp dnskey sshkey pem openssl mysql af-alg fips-prf gmp xcbc c
mac hmac ctr ccm gcm ntru curl
}
```

Associated revisions

Revision 865fd804 - 07.02.2017 10:52 - Tobias Brunner

eap-dynamic: Publish the get_auth() method of the wrapped EAP method

Fixes #2238.

History

#1 - 01.02.2017 11:23 - Tobias Brunner

- Tracker changed from Issue to Bug

- Category set to libcharon

- Status changed from New to Feedback

The *eap-dynamic* plugin currently does not implement the get_auth() method. Could you please try if the patch in the *2238-eap-dynamic-auth* branch works for you?

But I wonder, what happens if the client selects EAP-MSCHAPv2? Since there is no client certificate then, wouldn't the constraint prevent that too?

#2 - 01.02.2017 16:20 - Noel Kuntze

Yes, the patch works.
Yes, if EAP-MSCHAPv2 is chosen, the constraint prevents continuing with that connection. However, if a second connection with EAP-MSCHAPv2 is defined, charon switches to using it and it works fine.

#3 - 01.02.2017 16:59 - Tobias Brunner

- Assignee set to Tobias Brunner

- Target version set to 5.5.2

- Resolution set to Fixed

OK, I'll line this up for the next release.

#4 - 16.02.2017 19:28 - Tobias Brunner

- Status changed from Feedback to Closed

Files

charon.log	105 KB	31.01.2017	Noel Kuntze