strongSwan - Issue #3671

Windows client failed with 13843 against Strongswan via SQL backend

11.01.2021 15:36 - Cuong Truong

Status:	Feedback		
Priority:	Normal		
Assignee:			
Category:	configuration		
Affected version:	5.7.1	**Resolution:**	

Description

Hi, I have Strongswan ver 5.7.1 running in Ubuntu 18.04 with SQL backend enabled.

When I connect using Windows VPN client, the charon log shows:

```
Jan 11 14:18:17 strongswan charon: 00[DMN] Starting IKE charon daemon (strongSwan 5.7.1, Linux 4.1
5.0-122-generic, x86_64)
Jan 11 14:18:17 strongswan charon: 00[CFG] PKCS11 module '<name>' lacks library path
Jan 11 14:18:17 strongswan charon: 00[CFG] dnscert plugin is disabled
Jan 11 14:18:17 strongswan charon: 00[CFG] loading ca certificates from '/nix/store/y2xfv5ijynvnxp
jzj4ydrmgs5nbi10lg-strongswan/etc/ipsec.d/cacerts'
Jan 11 14:18:17 strongswan charon: 00[CFG] loading aa certificates from '/nix/store/y2xfv5ijynvnxp
jzj4ydrmgs5nbi10lg-strongswan/etc/ipsec.d/aacerts'
Jan 11 14:18:17 strongswan charon: 00[CFG] loading ocsp signer certificates from '/nix/store/y2xfv
5ijynvnxpjzj4ydrmgs5nbi10lg-strongswan/etc/ipsec.d/ocspcerts'
Jan 11 14:18:17 strongswan charon: 00[CFG] loading attribute certificates from '/nix/store/y2xfv5i
jynvnxpjzj4ydrmgs5nbi10lg-strongswan/etc/ipsec.d/acerts'
Jan 11 14:18:17 strongswan charon: 00[CFG] loading crls from '/nix/store/y2xfv5ijynvnxpjzj4ydrmgs5
nbi10lg-strongswan/etc/ipsec.d/crls'
Jan 11 14:18:17 strongswan charon: 00[CFG] loading secrets from '/nix/store/y2xfv5ijynvnxpjzj4ydrm
gs5nbi10lg-strongswan/etc/ipsec.secrets'
Jan 11 14:18:17 strongswan charon: 00[CFG] loading secrets from '/etc/ipsec.secrets'
Jan 11 14:18:17 strongswan charon: 00[CFG] read 0 triplets from /nix/store/y2xfv5ijynvnxpjzj4ydrmg
s5nbi10lg-strongswan/etc/ipsec.d/triplets.dat
Jan 11 14:18:17 strongswan charon: 00[CFG] loaded 0 RADIUS server configurations
Jan 11 14:18:17 strongswan charon: 00[CFG] no script for ext-auth script defined, disabled
Jan 11 14:18:17 strongswan charon: 00[LIB] loaded plugins: charon unbound pkcs11 aesni aes des rc2
 sha2 sha1 md5 mgf1 rdrand random nonce x509 revocation constraints acert pubkey pkcs1 pkcs7 pkcs8
 pkcs12 pgp dnskey sshkey dnscert pem openssl af-alg fips-prf gmp curve25519 chapoly xcbc cmac hma
c curl mysql sqlite attr kernel-netlink resolve socket-default connmark farp stroke vici sql updow
n eap-identity eap-sim eap-sim-file eap-sim-pcsc eap-aka eap-aka-3gpp2 eap-simaka-pseudonym eap-si
maka-reauth eap-md5 eap-gtc eap-mschapv2 eap-radius xauth-generic xauth-eap xauth-pam dhcp counter
s
Jan 11 14:18:17 strongswan charon: 00[JOB] spawning 16 worker threads
Jan 11 14:18:17 strongswan ipsec[7794]: charon (7822) started after 60 ms
Jan 11 14:18:19 strongswan charon: 14[CFG] loaded certificate '<certificate info hidden>'
Jan 11 14:18:19 strongswan charon: 10[CFG] loaded RSA private key
Jan 11 14:18:19 strongswan charon: 11[CFG] added vici pool 666: 10.4.0.0, 14 entries
Jan 11 14:18:41 strongswan charon: 09[NET] received packet: from 80.242.165.195[500] to 139.59.131
.79[500] (624 bytes)
Jan 11 14:18:41 strongswan charon: 09[ENC] parsed IKE_SA_INIT request 0 [ SA KE No N(FRAG_SUP) N(N
ATD_S_IP) N(NATD_D_IP) V V V V ]
Jan 11 14:18:41 strongswan charon: 09[IKE] received MS NT5 ISAKMPOAKLEY v9 vendor ID
Jan 11 14:18:41 strongswan charon: 09[IKE] received MS-Negotiation Discovery Capable vendor ID
Jan 11 14:18:41 strongswan charon: 09[IKE] received Vid-Initial-Contact vendor ID
Jan 11 14:18:41 strongswan charon: 09[ENC] received unknown vendor ID: 01:52:8b:bb:c0:06:96:12:18:
49:ab:9a:1c:5b:2a:51:00:00:00:02
Jan 11 14:18:41 strongswan charon: 09[IKE] 80.242.165.195 is initiating an IKE_SA
Jan 11 14:18:41 strongswan charon: 09[CFG] selected proposal: IKE:AES_CBC_256/HMAC_SHA2_256_128/PR
F_HMAC_SHA2_256/MODP_1024
Jan 11 14:18:41 strongswan charon: 09[IKE] remote host is behind NAT
Jan 11 14:18:41 strongswan charon: 09[ENC] generating IKE_SA_INIT response 0 [ SA KE No N(NATD_S_I
```

```
P) N(NATD_D_IP) N(MULT_AUTH) ]
Jan 11 14:18:41 strongswan charon: 09[NET] sending packet: from 139.59.131.79[500] to 80.242.165.1
95[500] (312 bytes)
Jan 11 14:18:41 strongswan charon: 08[NET] received packet: from 80.242.165.195[4500] to 139.59.13
1.79[4500] (1600 bytes)
Jan 11 14:18:41 strongswan charon: 08[ENC] parsed IKE_AUTH request 1 [ IDi CERTREQ N(MOBIKE_SUP) C
PRQ(ADDR DNS NBNS SRV ADDR6 DNS6 SRV6) SA TSi TSr ]
Jan 11 14:18:41 strongswan charon: 08[IKE] received cert request for "<certificate info hidden>"
Jan 11 14:18:41 strongswan charon: 08[IKE] received 62 cert requests for an unknown ca
Jan 11 14:18:41 strongswan charon: 08[CFG] looking for peer configs matching 139.59.131.79[%any]..
.80.242.165.195[192.168.22.102]
Jan 11 14:18:41 strongswan charon: 08[CFG] selected peer config 'WINDOWS-EAP'
Jan 11 14:18:41 strongswan charon: 08[IKE] using configured EAP-Identity inga
Jan 11 14:18:41 strongswan charon: 08[IKE] initiating EAP_MSCHAPV2 method (id 0x96)
Jan 11 14:18:41 strongswan charon: 08[IKE] peer supports MOBIKE
Jan 11 14:18:41 strongswan charon: 08[IKE] authentication of '<Ident hidden>' (myself) with RSA si
gnature successful
Jan 11 14:18:41 strongswan charon: 08[ENC] generating IKE_AUTH response 1 [ IDr AUTH EAP/REQ/MSCHA
PV2 ]
Jan 11 14:18:41 strongswan charon: 08[NET] sending packet: from 139.59.131.79[4500] to 80.242.165.
195[4500] (400 bytes)
Jan 11 14:19:11 strongswan charon: 12[JOB] deleting half open IKE_SA with 80.242.165.195 after tim
eout
```

The SQL tables are populated exactly with information taken from the config in the below, which got the same Windows VPN client to connect successfully to the same server above. All the configuration at Windows side is the same too.

```
connections {

   windows-eap {
      local_addrs  = 139.59.131.79
      pools = windows-eap-pool

      local {
         auth = pubkey
         certs = <cert-file>
         id = <FQDN ident>
      }
      remote {
         auth = eap-mschapv2
         eap_id = inga
      }
      children {
         windows-eap {
            local_ts  = 0.0.0.0/0
            esp_proposals = aes256-sha1
         }
      }
      version = 2
      proposals = aes256-sha2_256-prfsha256-modp1024
      send_certreq = no
   }
}

pools {
   windows-eap-pool {
      addrs = 10.4.0.0/28
      dns =  8.8.8.8,8.8.4.4
   }
}
secrets {
   eap-1 {
      id = inga
      secret = <secret>
   }
}
```

In my reasoning, if the SQL tables are populated with same info as in the config file, the resulting peer_cfg produced by SQl backend should be exactly the same as the one produced via the config file, which should not confuse the Windows VPN client...

I am using a forked from 5.7.1 to debug and do experiment. Please advice what should I do:
- to identify and debug this problem in general
- view the resulting peer_cfg objects if possible, so that I can compare them in two cases

Thank you very much for your help!

History

#1 - 11.01.2021 16:04 - Tobias Brunner

- Category set to configuration

- Status changed from New to Feedback

- Priority changed from High to Normal

What's clearly missing in the IKE_AUTH response is the server certificate. So you may have the wrong value in cert_policy (should either be 1 or 0).

Also, Windows usually wants an EAP-Identity exchange, i.e. you'd configure *eap_id=%any* (note that it's currently not possible to match a specific EAP-Identity without workaround, see #1057).

Also, the SQL plugin does not actually support EAP-Identities (see this pull request). Did you apply a patch for that?

> I am using a forked from 5.7.1 to debug and do experiment.

Why such an old version?

#2 - 11.01.2021 18:20 - Cuong Truong

Thanks for the quick response!

By the time we developed our first version of our software, the latest swan ver was 5.7.1. Since then we made several patches to a fork of that version due to our special needs. So we are now kind of stuck with ver 5.7.1 (we plan to do the upgrade but not for now...)

Also, I am not aware of that pull request, so I made this patch to enable SQL to work with eap_identity:

```
diff --git a/src/libcharon/plugins/sql/sql_config.c b/src/libcharon/plugins/sql/sql_config.c
index d24bfeea6..4e65079e0 100644
--- a/src/libcharon/plugins/sql/sql_config.c
+++ b/src/libcharon/plugins/sql/sql_config.c
@@ -327,7 +327,7 @@ static peer_cfg_t *get_peer_cfg_by_id(private_sql_config_t *this, int id)

     e = this->db->query(this->db,
             "SELECT c.id, c.name, c.ike_cfg, l.type, l.data, r.type, r.data, "
-            "c.cert_policy, c.uniqueid, c.auth_method, c.eap_type, "
+            "c.cert_policy, c.uniqueid, c.auth_method, c.eap_identity, c.eap_type, "
             "c.eap_vendor, c.keyingtries, c.rekeytime, c.reauthtime, c.jitter, "
             "c.overtime, c.mobike, c.dpd_delay, c.virtual, c.pool, "
             "c.mediation, c.mediated_by, COALESCE(p.type, 0), p.data "
@@ -338,7 +338,7 @@ static peer_cfg_t *get_peer_cfg_by_id(private_sql_config_t *this, int id)
             "WHERE c.id = ?",
             DB_INT, id,
             DB_INT, DB_TEXT, DB_INT, DB_INT, DB_BLOB, DB_INT, DB_BLOB,
-            DB_INT, DB_INT, DB_INT, DB_INT, DB_INT,
+            DB_INT, DB_INT, DB_INT, DB_INT, DB_INT, DB_INT,
             DB_INT, DB_INT, DB_INT, DB_INT, DB_INT, DB_INT,
             DB_INT, DB_TEXT, DB_TEXT,
             DB_INT, DB_INT, DB_INT, DB_BLOB);
@@ -368,7 +368,7 @@ static peer_cfg_t *build_peer_cfg(private_sql_config_t *this, enumerator_t *e,
     int id, ike_cfg, l_type, r_type,
-        cert_policy, uniqueid, auth_method, eap_type, eap_vendor, keyingtries,
+        cert_policy, uniqueid, auth_method, eap_identity, eap_type, eap_vendor, keyingtries,
         rekeytime, reauthtime, jitter, overtime, mobike, dpd_delay,
         mediation, mediated_by, p_type;
     chunk_t l_data, r_data, p_data;
@@ -379,7 +379,7 @@ static peer_cfg_t *build_peer_cfg(private_sql_config_t *this, enumerator_t *e,

     while (e->enumerate(e,
```

```
             &id, &name, &ike_cfg, &l_type, &l_data, &r_type, &r_data,
-            &cert_policy, &uniqueid, &auth_method, &eap_type, &eap_vendor,
+            &cert_policy, &uniqueid, &auth_method, &eap_identity, &eap_type, &eap_vendor,
             &keyingtries, &rekeytime, &reauthtime, &jitter, &overtime, &mobike,
             &dpd_delay,    &virtual, &pool,
             &mediation, &mediated_by, &p_type, &p_data))
@@ -400,7 +400,7 @@ static peer_cfg_t *build_peer_cfg(private_sql_config_t *this, enumerator_t *e,
             local_id->destroy(local_id);
             remote_id->destroy(remote_id);
             continue;
-        } else if (!id_matches(other, remote_id)) {
+        } else if (!eap_identity && !id_matches(other, remote_id)) {
             DBG2(DBG_CFG, "peer cfg ..%Y doesn't match %Y", remote_id, other);
             local_id->destroy(local_id);
             remote_id->destroy(remote_id);
@@ -463,11 +463,18 @@ static peer_cfg_t *build_peer_cfg(private_sql_config_t *this, enumerator_t *e,
             auth->add(auth, AUTH_RULE_IDENTITY, local_id);
             peer_cfg->add_auth_cfg(peer_cfg, auth, TRUE);
             auth = auth_cfg_create();
-            auth->add(auth, AUTH_RULE_IDENTITY, remote_id);
+            if (!eap_identity)
+            {
+                auth->add(auth, AUTH_RULE_IDENTITY, remote_id);
+            }
             if (eap_type)
             {
                 auth->add(auth, AUTH_RULE_AUTH_CLASS, AUTH_CLASS_EAP);
                 auth->add(auth, AUTH_RULE_EAP_TYPE, eap_type);
+                if (eap_identity)
+                {
+                    auth->add(auth, AUTH_RULE_EAP_IDENTITY, remote_id);
+                }
                 if (eap_vendor)
                 {
                     auth->add(auth, AUTH_RULE_EAP_VENDOR, eap_vendor);
@@ -497,7 +504,7 @@ METHOD(backend_t, get_peer_cfg_by_name, peer_cfg_t*,

     e = this->db->query(this->db,
             "SELECT c.id, c.name, c.ike_cfg, l.type, l.data, r.type, r.data, "
-            "c.cert_policy, c.uniqueid, c.auth_method, c.eap_type, "
+            "c.cert_policy, c.uniqueid, c.auth_method, c.eap_identity, c.eap_type, "
             "c.eap_vendor, c.keyingtries, c.rekeytime, c.reauthtime, c.jitter, "
             "c.overtime, c.mobike, c.dpd_delay, c.virtual, c.pool, "
             "c.mediation, c.mediated_by, COALESCE(p.type, 0), p.data "
@@ -508,7 +515,7 @@ METHOD(backend_t, get_peer_cfg_by_name, peer_cfg_t*,
             "WHERE c.ike_version = ? AND c.name = ?",
             DB_INT, 2, DB_TEXT, name,
             DB_INT, DB_TEXT, DB_INT, DB_INT, DB_BLOB, DB_INT, DB_BLOB,
-            DB_INT, DB_INT, DB_INT, DB_INT, DB_INT,
+            DB_INT, DB_INT, DB_INT, DB_INT, DB_INT, DB_INT,
             DB_INT, DB_INT, DB_INT, DB_INT, DB_INT, DB_INT,
             DB_INT, DB_TEXT, DB_TEXT,
             DB_INT, DB_INT,    DB_INT, DB_BLOB);
@@ -650,7 +657,7 @@ METHOD(backend_t, create_peer_cfg_enumerator, enumerator_t*,
     /* TODO: only get configs whose IDs match exactly or contain wildcards */
     e->inner = this->db->query(this->db,
             "SELECT c.id, c.name, c.ike_cfg, l.type, l.data, r.type, r.data, "
-            "c.cert_policy, c.uniqueid, c.auth_method, c.eap_type, "
+            "c.cert_policy, c.uniqueid, c.auth_method, c.eap_identity, c.eap_type, "
             "c.eap_vendor, c.keyingtries, c.rekeytime, c.reauthtime, c.jitter, "
             "c.overtime, c.mobike, c.dpd_delay, c.virtual, c.pool, "
             "c.mediation, c.mediated_by, COALESCE(p.type, 0), p.data "
@@ -661,7 +668,7 @@ METHOD(backend_t, create_peer_cfg_enumerator, enumerator_t*,
             "WHERE c.ike_version = ?",
             DB_INT, 2,
             DB_INT, DB_TEXT, DB_INT, DB_INT, DB_BLOB, DB_INT, DB_BLOB,
-            DB_INT, DB_INT, DB_INT, DB_INT, DB_INT,
+            DB_INT, DB_INT, DB_INT, DB_INT, DB_INT, DB_INT,
             DB_INT, DB_INT, DB_INT, DB_INT, DB_INT, DB_INT,
             DB_INT,    DB_TEXT, DB_TEXT,
             DB_INT, DB_INT, DB_INT, DB_BLOB);
diff --git a/src/pool/mysql.sql b/src/pool/mysql.sql
index 1b437593d..3b4caa01e 100644
--- a/src/pool/mysql.sql
+++ b/src/pool/mysql.sql
```

```
@@ -84,6 +84,7 @@ CREATE TABLE `peer_configs` (
   `cert_policy` tinyint(3) unsigned NOT NULL default '1',
   `uniqueid` tinyint(3) unsigned NOT NULL default '0',
   `auth_method` tinyint(3) unsigned NOT NULL default '1',
+  `eap_identity` tinyint(1) unsigned NOT NULL default '0',
   `eap_type` tinyint(3) unsigned NOT NULL default '0',
   `eap_vendor` smallint(5) unsigned NOT NULL default '0',
   `keyingtries` tinyint(3) unsigned NOT NULL default '3',
diff --git a/src/pool/sqlite.sql b/src/pool/sqlite.sql
index a35094073..80059201a 100644
--- a/src/pool/sqlite.sql
+++ b/src/pool/sqlite.sql
@@ -86,6 +86,7 @@ CREATE TABLE peer_configs (
   cert_policy INTEGER NOT NULL DEFAULT '1',
   uniqueid INTEGER NOT NULL DEFAULT '0',
   auth_method INTEGER NOT NULL DEFAULT '1',
+  eap_identity INTEGER NOT NULL DEFAULT '0',
   eap_type INTEGER NOT NULL DEFAULT '0',
   eap_vendor INTEGER NOT NULL DEFAULT '0',
   keyingtries INTEGER NOT NULL DEFAULT '3',
```

Which got me to the point where the peer_cfg was correctly selected.

As per your suggestion, I set cert_policy to 0 and 1 respectively, but still got the same logs as I attached above.

Any other suggestion? Or maybe the patch I made is wrong?

Thanks a lot!

#3 - 12.01.2021 13:23 - Tobias Brunner

> Also, I am not aware of that pull request, so I made this patch to enable SQL to work with eap_identity:

Not sure if that was your goal, but the code just configures the remote identity also as EAP identity in case eap_identity is non-zero. It does not actually add support for arbitrary EAP-Identities (although, as I mentioned in the PR, other than %any the behavior might not be as expected).

> As per your suggestion, I set cert_policy to 0 and 1 respectively, but still got the same logs as I attached above.

If no certificate is sent, the setting apparently was not set or read correctly.

#4 - 12.01.2021 16:45 - Cuong Truong

Thanks and I am certain that the cert_policy setting is read correctly. I added a log in the function:

```
static peer_cfg_t *build_peer_cfg(private_sql_config_t *this, enumerator_t *e,
```

in libcharon/plugins/sql/sql_config.c. It always correctly print out the current value set in DB for the cert_policy variable.

Where in the code base can I find this cert_policy is used? I can do some debugging from there...

Thanks a lot!

#5 - 12.01.2021 17:22 - Tobias Brunner

> Where in the code base can I find this cert_policy is used?

source:src/libcharon/sa/ikev2/tasks/ike_cert_post.c#L233

#6 - 13.01.2021 14:12 - Cuong Truong

you are right. It fails right here:

```
cert = auth->get(auth, AUTH_RULE_SUBJECT_CERT);
if (!cert)
{
    return FALSE;
```

```
}
```

Can you please advice, when and where should that cert get added? I didn't find a clue in the sql plugin code..

Thanks a lot!

#7 - 13.01.2021 14:43 - Tobias Brunner

> Can you please advice, when and where should that cert get added? I didn't find a clue in the sql plugin code..

You can't explicitly associate certificates with connections with the *sql* plugin. The certificate is added when creating the signature (actually during the private key lookup, which happens based on the configured local identity).