strongSwan - Feature #3602

No event when other party IP address changes

19.10.2020 22:07 - Sergey Merzlikin

Status:	Closed	**Start date:**	
Priority:	Normal	**Due date:**	
Assignee:	Tobias Brunner	**Estimated time:**	0.00 hour
Category:	vici		
Target version:	5.9.2		
Resolution:	Fixed		

Description

Hello.

I use strongswan-systemd (Ubuntu 20.04) in transport mode (IKEv2) with IPIP tunnel. This works excellent until ISP provider of other party of connection changes its public IP address (other party is on cellular network with dynamic IP address and behind NAT).
Strongswan handles this situation too transparently: transport mode connection doesn't break, and no event raised.
But obviously after IP address change I need to modify IP address of tunnel connection (ip tunnel change ...), otherwise tunnel will not work.
I increased log level to 2 and found that when other party IP address changes two records are written to log:

```
Sep 28 14:38:25 srv-63 charon-systemd[2451761]: CHILD_SA 101{308} state change: INSTALLED => UPDATING
Sep 28 14:38:25 srv-63 charon-systemd[2451761]: CHILD_SA 101{308} state change: UPDATING => INSTALLED
```

Then I decided to catch these records via vici log event. Unfortunately vici log event is able to catch events with loglevel 0 and 1 only, and I didn't found means to increase log verbosity.

Filally I added separate file logger, directed it to named pipe, which was created and read by separate shell daemon, which in turn catches required event and modifies tunnel settings. It works, but I can't say that this solution is good...

So, my questions:
- Firstly, maybe I missed something, and there is already simple solution for my task.
- Is it possible to add event to vici (and/or updown) which fires when connection state changes (or at least when one of IP addresses changes)?
- Is it possible to add setting or API function to change vici log verbosity?

BTW, vici doesn't fire any events when strongswan shuts down. I expect to see full set of IKE and SA down events in this condition. I can't move from updown to vici events becouse of this.

Associated revisions

Revision ef636316 - 30.10.2020 09:58 - Tobias Brunner

vici: Send all queued messages during shutdown

This ensures that e.g. ike/child-updown messages are sent that were
queued but couldn't be sent (even the job to enable to on_write() callback
requires a worker thread that's not around anymore during shutdown).

References #3602.

Revision f97875b7 - 18.01.2021 13:33 - Tobias Brunner

Merge branch 'ike-update-event'

This modifies the signature of the listener_t::ike_update() callback so
that both addresses are passed and it's only called once if both
addresses change (e.g. for an address family switch).

The callback is now also triggered for MOBIKE updates and the event is
exposed via vici.

Fixes #3602.

History

#1 - 20.10.2020 20:25 - Tobias Brunner

- Status changed from New to Feedback

> I use strongswan-systemd (Ubuntu 20.04) in transport mode (IKEv2) with IPIP tunnel.

Why not just use tunnel mode? And if route-based, did you read RouteBasedVPN?

> Strongswan handles this situation too transparently: transport mode connection doesn't break, and no event raised.

Such a change is completely undetectable until one of the peers sends a message. And only a message from the host behind the NAT could fix it without breaking the connection, e.g. a DPD from it with NAT-D payloads, which then triggers a MOBIKE update, or simply an ESP packet that causes a kernel event that updates the address.

> Then I decided to catch these records via vici log event. Unfortunately vici log event is able to catch events with loglevel 0 and 1 only, and I didn't found means to increase log verbosity.

Besides that the vici logger performs badly, higher log levels could cause loops.

> - Firstly, maybe I missed something, and there is already simple solution for my task.

See my initial questions.

> - Is it possible to add event to vici (and/or updown) which fires when connection state changes (or at least when one of IP addresses changes)?

There is an API event, ike_update() (can be caught via custom plugin), that's currently not exposed via vici. We could theoretically change that but I'm unsure about the API for this event.

> - Is it possible to add setting or API function to change vici log verbosity?

No, see above.

> BTW, vici doesn't fire any events when strongswan shuts down. I expect to see full set of IKE and SA down events in this condition. I can't move from updown to vici events becouse of this.

Interesting. The ike/child-updown events should theoretically be triggered in vici just like the updown script is called during shutdown. However, unlike updown, vici does send these events asynchronously and during shutdown the worker threads required for that are already gone. A possible workaround is to keep some state in the vici client and clean up based on that when the event connection with the daemon gets terminated. Anyway, I pushed a possible fix to the *3602-vici-events* branch.

#2 - 20.10.2020 22:21 - Sergey Merzlikin

> Why not just use tunnel mode? And if route-based, did you read RouteBasedVPN?

The other party is a router with limited functionality. It supports tunnel mode, but for one subnet only while I need to route 10+ subnets there. As alternative I can create 10+ CHILD_SAs, but I don't like this idea. Router supports IPIP, GRE and EoIP tunnels; VTI and XFRM are not supported.

> Such a change is completely undetectable until one of the peers sends a message. And only a message from the host behind the NAT could fix it without breaking the connection, e.g. a DPD from it with NAT-D payloads, which then triggers a MOBIKE update, or simply an ESP packet that causes a kernel event that updates the address.

BTW, I tried to disable MOBIKE on my party without success. Even with disabled MOBIKE IP address changes don't break connection. Maybe I need to disable MOBIKE on other party? It is not configurable there...

> Besides that the vici logger performs badly, higher log levels could cause loops.

Do you have plans to fix this? Or vici logger is already legacy?

> There is an API event, ike_update() (can be caught via custom plugin), that's currently not exposed via vici. We could theoretically change that but I'm unsure about the API for this event.

It will be nice if this event would be exposed in vici.

Currently I'm testing my solution with parsing log via named pipe. My intermediate results - it is unreliable. Some log records are missing at the parser side. I will investigate further.

#3 - 22.10.2020 19:32 - Tobias Brunner

> The other party is a router with limited functionality. It supports tunnel mode, but for one subnet only while I need to route 10+ subnets there. As alternative I can create 10+ CHILD_SAs, but I don't like this idea. Router supports IPIP, GRE and EoIP tunnels; VTI and XFRM are not supported.

The use of the latter are basically a local matter, no need for the peer to support them. But whether that works still depends on the other end (e.g. if it also creates some kind of interface) and what traffic selectors could be negotiated (e.g. 0.0.0.0/0 on both ends so only routing decides what to tunnel). I guess the peer doesn't support BEET mode? That could allow you to use "virtual" IPs for the tunnel interface which would not be affected by address changes (doing that with tunnel mode might also work but with more overhead).

And why not negotiate separate CHILD_SAs? There is not that much overhead if you don't use reauthentication that occasionally requires reestablishing them from scratch. With trap policies you could even let them get created on demand as they are needed.

> BTW, I tried to disable MOBIKE on my party without success. Even with disabled MOBIKE IP address changes don't break connection. Maybe I need to disable MOBIKE on other party? It is not configurable there...

If the change is triggered by an ESP packet from a new IP/port, the SAs are just updated implicitly also without MOBIKE.

> > Besides that the vici logger performs badly, higher log levels could cause loops.
>
> Do you have plans to fix this? Or vici logger is already legacy?

No, I don't plan to change anything in that regards. I don't see much reason for a logger via vici and parsing log messages is a bad idea in the first place.

> > There is an API event, ike_update() (can be caught via custom plugin), that's currently not exposed via vici. We could theoretically change that but I'm unsure about the API for this event.
>
> It will be nice if this event would be exposed in vici.

I wonder if it's necessary to pass information on the CHILD_SAs (I suppose it depends on what has to be done with the new IP address(es)). This might require a new child_update() event (doesn't exist yet, not even for plugins), however, that might cause a lot of additional events. So I guess we could also just send all CHILD_SAs with the *ike-update* vici event (similar to the message received from the *list-sa* event when streaming SAs via *list-sas* vici command). I pushed a commit that does so to the *3602-vici-events* branch. The event is currently triggered twice if both local and remote addresses change (e.g. when switching address families), I think it would be a good idea to change this so that the event is only triggered once.

#4 - 23.10.2020 02:05 - Sergey Merzlikin

> The use of the latter are basically a local matter, no need for the peer to support them. But whether that works still depends on the other end (e.g. if it also creates some kind of interface) and what traffic selectors could be negotiated (e.g. 0.0.0.0/0 on both ends so only routing decides what to tunnel).

The other end is keenetic router. It has linux and strongswan inside, but developers decided to make cisco-like configuration and command-line interface for it. So, I'm configuring IKE via crypto map, crypto policy and so on. Developers promise add support for vti interfaces in the next version of firmware. It will be at least in new year, and only for relatively new devices...

> I guess the peer doesn't support BEET mode?

Yes, it doesn't support. But it is interesting mode, probably I overlooked it. I will try it in another place.

> And why not negotiate separate CHILD_SAs? There is not that much overhead if you don't use reauthentication that occasionally requires reestablishing them from scratch. With trap policies you could even let them get created on demand as they are needed.

Manageability. With many CHILD_SAs config file is larger, log is longer, and probability to catch a bug is many times bigger. I use this configuration with cisco router on the other end.

> No, I don't plan to change anything in that regards. I don't see much reason for a logger via vici and parsing log messages is a bad idea in the first place.

Agree, it is a bad idea, but if other ideas are exhausted...

> I wonder if it's necessary to pass information on the CHILD_SAs (I suppose it depends on what has to be done with the new IP address(es)). This might require a new child_update() event (doesn't exist yet, not even for plugins), however, that might cause a lot of additional events. So I guess we could also just send all CHILD_SAs with the *ike-update* vici event (similar to the message received from the *list-sa* event when streaming SAs via *list-sas* vici command). I pushed a commit that does so to the *3602-vici-events* branch. The event is currently triggered twice if both local and remote addresses change (e.g. when switching address families), I think it would be a good idea to change this so that the event is only triggered once.

Isn't it big overhead to send all CHILD_SAs there? Maybe it will be better just to send minimal information in event (IKE_SA id and new addresses), all other information may be queried separately if required in event processing.

#5 - 29.10.2020 09:31 - Tobias Brunner

> Isn't it big overhead to send all CHILD_SAs there?

It's probably not that much of a problem in general (in particular because the overhead of querying the SAs later might be higher if that has to happen every time). But it really depends on what the users want to do or how much state a vici client already keeps itself. There is currently no depth parameter (or something similar) when subscribing to events.

> Maybe it will be better just to send minimal information in event (IKE_SA id and new addresses), all other information may be queried separately if required in event processing.

I'd send the IKE_SA information for consistency with the other events (e.g. *ike-rekey* or *ike-updown*). But we can omit the CHILD_SA information as e.g. updating tunnel interfaces should not require information about the CHILD_SAs.

Unfortunately, I've noticed that the *ike-update* event is not actually triggered for MOBIKE updates, only address changes triggered via received ESP payloads (or some IKE messages if MOBIKE is disabled) used that path. So these changes won't make it into the next release.

#6 - 27.11.2020 12:25 - Tobias Brunner

> Unfortunately, I've noticed that the *ike-update* event is not actually triggered for MOBIKE updates, only address changes triggered via received ESP payloads (or some IKE messages if MOBIKE is disabled) used that path.

I've pushed changes related to this to the *3602-ike-update-event* branch.

#7 - 27.11.2020 12:26 - Tobias Brunner

- Tracker changed from Issue to Feature

- Target version set to 5.9.2

- Affected version deleted (5.8.2)

#8 - 18.01.2021 13:34 - Tobias Brunner

- Status changed from Feedback to Closed

- Assignee set to Tobias Brunner

- Resolution set to Fixed