strongSwan - Issue #926

HA: resync errors when a node is joining a cluster

10.04.2015 12:59 - Emeric Poupon

Status:	New		
Priority:	Normal		
Assignee:			
Category:			
Affected version:	5.3.0	**Resolution:**	

Description

Hello,

When a node is joining a cluster, a RESYNC message is sent to the other node.
However, with large connection setups (hundreads of connections), sending all the connections from starter to charon may be quite long.
Unfortunately the joining node may receive IKE SA and even CHILD SA from the other node while the configuration is not yet fully loaded.
This leads to bad synchronization of the IKE SA/CHILD SA states.

Regards,

History

#1 - 10.04.2015 14:27 - Martin Willi

Hi Emeric,

Yes, I assume you are referring to the mailing list discussion, and the patch you posted.

I've slightly adapted your patch, splitted it up and pushed it to the ha-sync-delay branch.

However, there is one major issue with the patch: It only works with an *ipsec.conf* backend, but breaks resynchronization if starter is not used, but any other/custom configuration backend.

We could extend vici/swanctl to send these signals. Not sure if somebody uses a custom backend along with HA, propably not.

Another solution could be if stroke tells the HA plugin that it supports such a signal. For example, HA could use the existing resync behavior, but if it receives RELOAD_START then defers execution until it receives (the same amount of) RELOAD_STOP.

Regards
Martin

#2 - 10.04.2015 15:05 - Emeric Poupon

Hello,

First, thanks for your support.

Yes, your solution may be better since it would not break a non "ipsec.conf" backend.

However, I just discovered another annoying related problem:
The node starts loading the connections received from starter. While the connections are loading, IKE SA and CHILD SA are set up and sent synchronously to the other node. Once the connection load is complete a RESYNC is performed and the node is then receiving the freshly negotiated IKE SA/CHILD SA from the other node. This results of course with a lot of errors (duplicated entries, duplicated SPI entries in kernel, etc.)

Actually, it looks as if we would need to prevent charon from doing anything on the network while a reloading is in progress.

What do you think?

Regards,
Emeric