Implementation

Aryan Gupta

June 2020

1 Implementing IPSec (using PISKES) in SCION Architecture

There are two ways in which IPSec can be used in SCION.

1. Between an end host and the AS' Path Server(PS), Beacon Server(BS) and Certificate Server(CS)
2. Between border routers of different ASes

In the current SCION implementation, the various components of the AS, i.e., path server, beacon server and certificate servers are setup on a single system, an IPSec tunnel set up between the system on which AS is running and the end host will suffice if we are to use the first method.

The host in a SCION implementation has-

1. **SCION Dispatcher**- This packages the information received from the applications into a SCION Packet, attaching all the necessary headers. Since currently SCION is used as an overlay network on the current Internet implementation, this packet is then packaged into an IP packet and sent to the destination.
2. **SCION Daemon**- This runs as a background program and supplies the necessary information, for e.g., the paths to the destination, PISKES key, the certificates etc.

We want to implement an IPSec tunnel using PISKES key as the preshared key. This would mean deriving the key, using SCION Daemon to fetch it, and use this key as a preshared key.

2 How?

strongSwan is an OpenSource IPsec implementation. It enables easy setting up of an IPSec tunnel between two endpoints. strongSwan provides a suitable range of options for various configurations. It also provides an option to select between IKEv1 and IKEv2. After installing strongSwan, whenever we want to establish a new connection, we basically have to edit two files-

1. **ipsec.conf** - This file contains all the information about the IPSec connection. It involves the name of the connection, the source and destionation IP address, preffered mode of authentication, which cryptographic protocols to use etc.
2. **ipsec.secrets** - It contains the necessary information for the mode of authentication chosen in ipsec.conf file, for e.g., the shared secrets, or the credentials or in our case, the PISKES key.

We cam use strongSwan for fulfilling our purpose.

Development in scionproto In scionproto implementation, there is an option to run each AS in a separate docker container. If we can install strongSwan in each of the containers, we will be able to configure each of

the Ases separately as required.
If we want to make a connection between host A and host B, then we can use the sciond of host A and host B to fetch the PISKES key, then by running a script on both the hosts, we can edit the ipsec.conf and ipsec.secrets file, and automatically establish an IPSec connection between them.