ACL Configuration for SONiC with ConfigDB

# Overview

This document outline the high priority scenarios where ACL configuration needs be supported in SONiC, and how we propose to support them in SONiC adapting ConfigDB.

# Scenarios

1. [P0] ACL full update: an ACL rule definition file will be provided in openconfig json format. All existing ACLs will be removed, and new ACLs will be configured according to the file. ACL tables and/or mirror sessions are configured beforehand based on minigraph or config DB dump.
2. [P1] ACL incremental update: an ACL definition file will be provided in openconfig json format. The difference between ACL in the file and current ACL will be extracted, and ACLs will be added, deleted, or modified based on the diff. ACL table binding and mirror session information can be assumed unchanged.
3. [P2] ACL rule and binding dynamic configuration through CLI. Detail grammar / use cases has not been yet defined.

# Components

Here’s the proposed data path for ACL configuration.



Current ACL\_TABLE, ACL\_RULE\_TABLE, and MIRROR\_SESSION\_TABLE in AppDB shall be moved into **ConfigDB**. Orchagent will subscribe to ConfigDB instead. Table schemas shall mostly remain same, excepts that 1. Binding ports field in ACL\_TABLE will support port channels and vlans; 2. several fields such as gre\_type in MIRROR\_SESSION\_TABLE should be considered optional.

**ACL Definition File** contains the ACL rule information. It follows an openconfig-compatible json schema which was defined in a previous document. A parser for this file is already available at <https://github.com/Azure/sonic-buildimage/blob/master/src/sonic-config-engine/translate_acl>.

**ACL Loader** is a tool that loads ACL rules from ACL Definition File, and translate them into ConfigDB schema. It will then write them into config DB after clear corresponding ACL tables (in scenario A), or compared them to current value in config DB and do addition, deletion and modification correspondingly (in scenario B). In either scenario, it can be assumed that ACL table and mirror session has already been set up in config DB, and rules that don’t match any exiting table could be ignored.

ACL table binding information currently lies in MINIGRAPH. There needs to be a **Minigraph Parser** that is able to read that information from MINIGRAPH, generate ACL\_TABLE and MIRROR\_SESSION table entries accordingly, and write them in configDB.

# Work Items

* [P0] Move ACL\_TABLE, ACL\_RULE\_TABLE, and PORT\_MIRROR\_TABLE to config DB, rename to ACL\_TABLE, ACL\_RULE and MIRROR\_SESSION correspondingly.
* [P0] “src\_ip”, “gre\_type”, “dscp”, and “queue” fields in MIRROR\_SESSION\_TABLE will become optional. Orchagent will fill them with [platform-dependent default values](https://github.com/Azure/sonic-buildimage/blob/master/dockers/docker-orchagent/mirror.json.j2) if they are not specified in config DB.
* [P0] “ports” field in ACL\_TABLE will need to support port channels and vlans.
* [P0] Modify existing [Minigraph Parser](https://github.com/Azure/sonic-buildimage/blob/master/src/sonic-config-engine/minigraph.py), so that it outputs in ACL\_TABLE and MIRROR\_SESSION schema instead of current “minigraph\_acl” and “erspan\_destination” dictionaries. The data pipeline to load parser output into config DB is already ready.
* [P0] Modify existing [translate\_acl](https://github.com/Azure/sonic-buildimage/blob/master/src/sonic-config-engine/translate_acl) tool to form an ACL Loader that will manipulate ACL\_RULE table in config DB instead of generating swssconfig json files.
* [P1] ACL Loader should support an option to compare between target rules and current rules and to only apply diff instead of to erase and reconfigure everything.
* [P0] Orchagent needs to subscribe on ConfigDB instead of AppDB. As currently only a python version of configDB SDK is available (py-swsssdk), an effort will be needed to migrate the SDK to C similar to the AppDB SDK (swss-common).
* [P2] Design CLI grammar for ACL configuration, and implement in sonic-utilities.
* [?] Add a type in ACL\_TABLE to indicate control-plane ACL.