

# Intra-domain Source Address Validation (SAVNET) Architecture

[draft-li-savnet-intra-domain-architecture-01](#)

D. Li, J. Wu, M. Huang, L. Chen, **N. Geng**, L. Qin, F. Gao

March 2023

# A Big Update

solution

~~draft-li-savnet-intra-domain-architecture-00~~

A preliminary **solution** to automatically generating accurate SAV rules through message interactions between routers



draft-li-savnet-intra-domain-architecture-01

An **architecture** collecting SAV information from multiple information sources for automated and accurate SAV rule generation

# Required Information for SAV Rule Generation

## ❑ Required SAV information

- ◆ **All source addresses/prefixes** even not advertised by route
- ◆ **All real incoming directions** and reduce non-real incoming directions

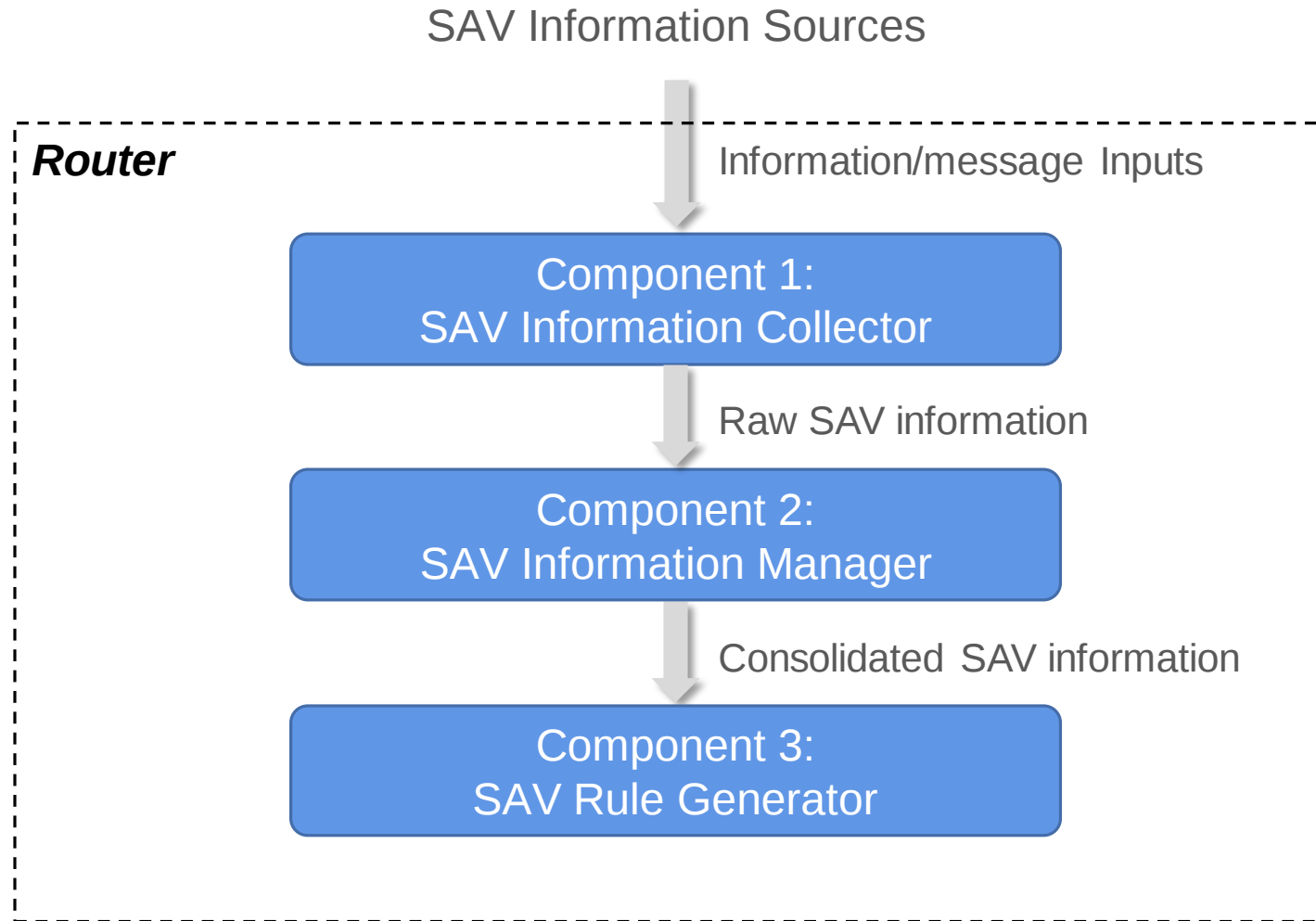
## ❑ Information sources of SAV mechanisms

- ◆ **Configuration** (e.g., ACL-based filtering)
- ◆ **Routing protocols** (e.g., strict uRPF and loose uRPF)
- ◆ **SAV protocols** (e.g., IGP-extended SAV, BGP-extended SAV)
- ◆ etc.

## ❑ Requirements for SAV mechanisms

- ◆ Accurate
- ◆ Automated
- ◆ Easy deployment
- ◆ Low communication overhead
- ◆ etc.

# Intra-domain SAVNET Architecture (Brief)



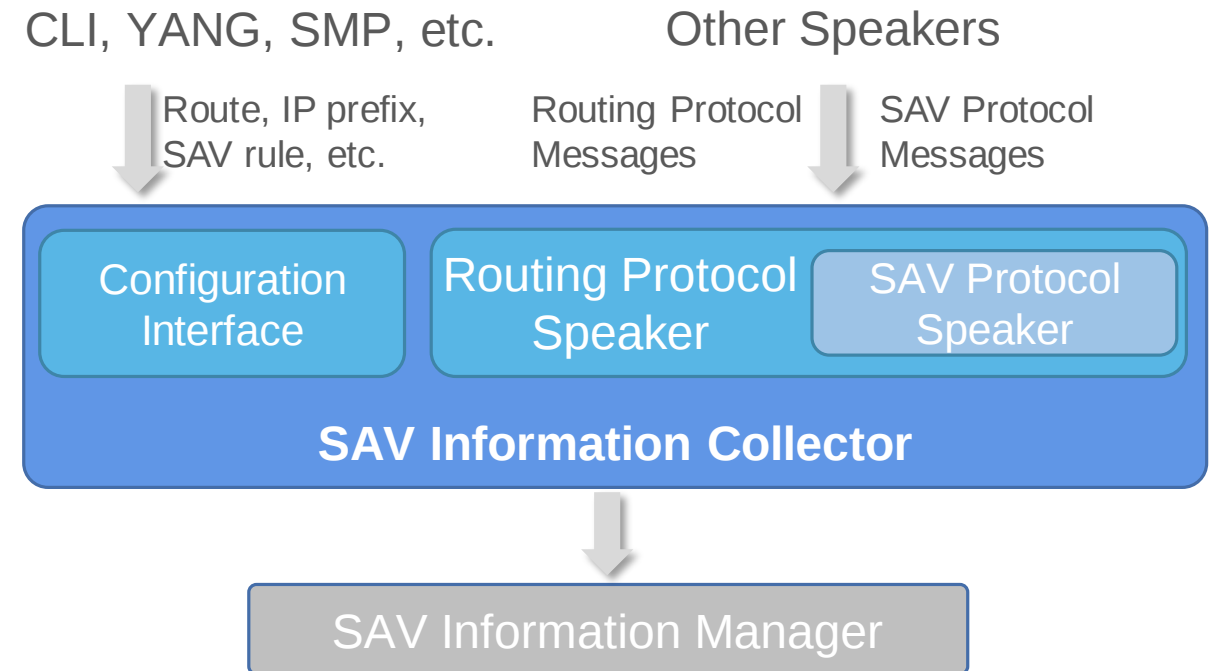
# Component 1: SAV Information Collector

## □ Main functions

- ◆ Collect SAV information from multiple information sources outside the router
- ◆ Disseminate the information to SAV Information Manager

## □ Information Sources

- ◆ Configuration (Basic)
- ◆ Routing protocols (Existing)
- ◆ SAV protocols (Future)

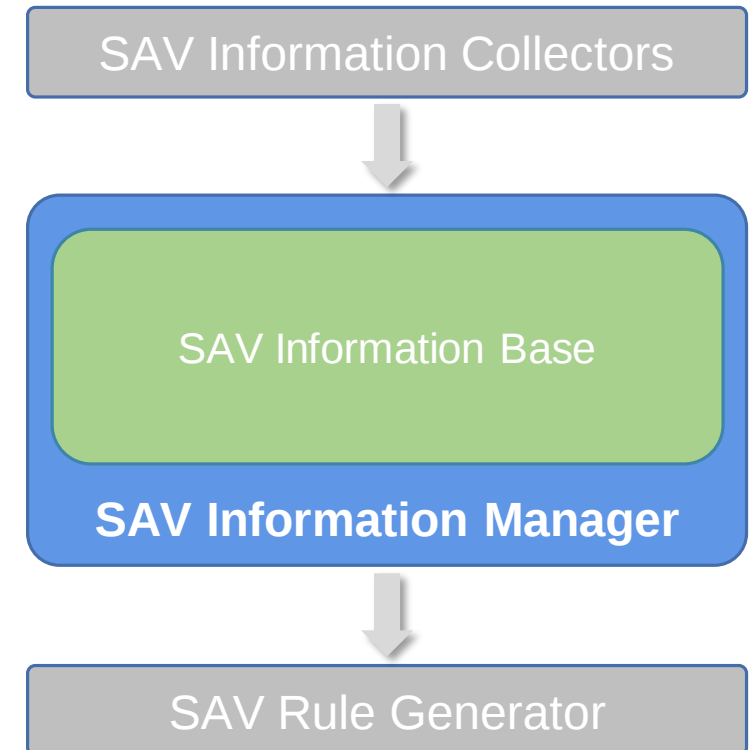


Note: SAV Management Protocol (SMP) represents any management protocols for SAV-related configurations

# Component 2: SAV Information Manager

## □ Main functions

- ◆ Consolidate SAV information from different information sources
- ◆ Store the information (prefix, direction, info. source, etc. ) in SAV Information Base
- ◆ Provide information to SAV Rule Generator



# Component 3: SAV Rule Generator

## □ Main functions

- ◆ Generate SAV rules (<prefix, interface>) based on SAV information
- ◆ Store the rules in SAV Table<sup>[1]</sup>
- ◆ Update the rules to the data plane

## □ SAV information conflicts for a prefix

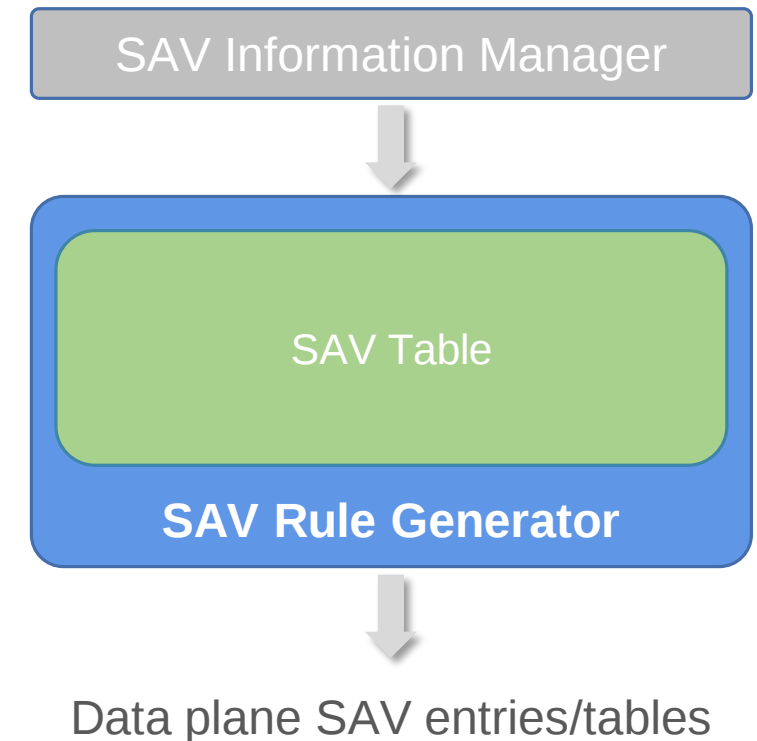
- ◆ Priorities can be set to these sources

### Example 1:

High priority source X: <P1, intf1>  
Low priority source Y: <P1, intf2>  
Output SAV rule: <P1, intf1>

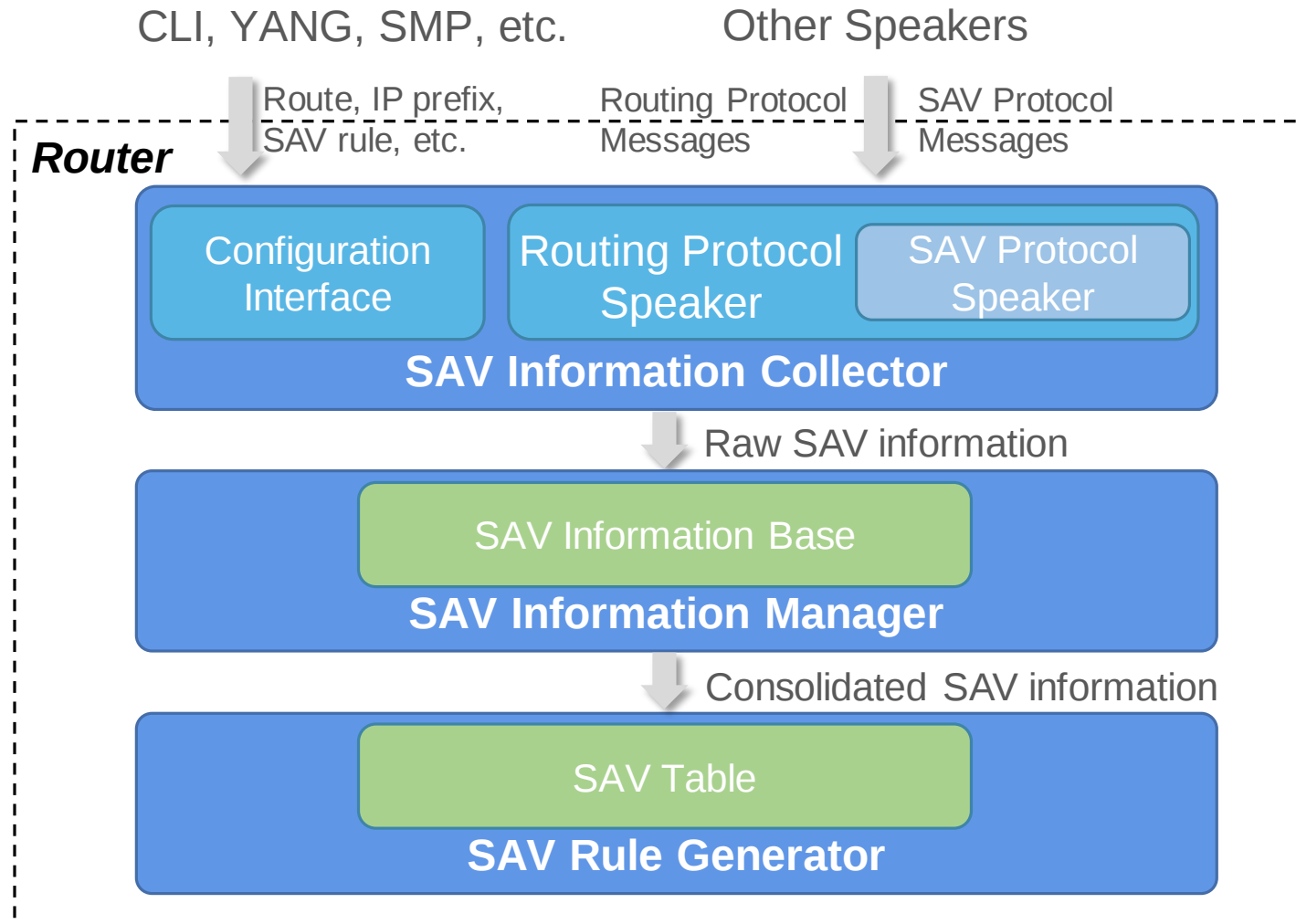
### Example 2:

Same priority source X: <P1, intf1>  
Same priority source Y: <P1, intf2>  
Output SAV rule: <P1, [intf1, intf2]>



[1] draft-huang-savnet-sav-table-01

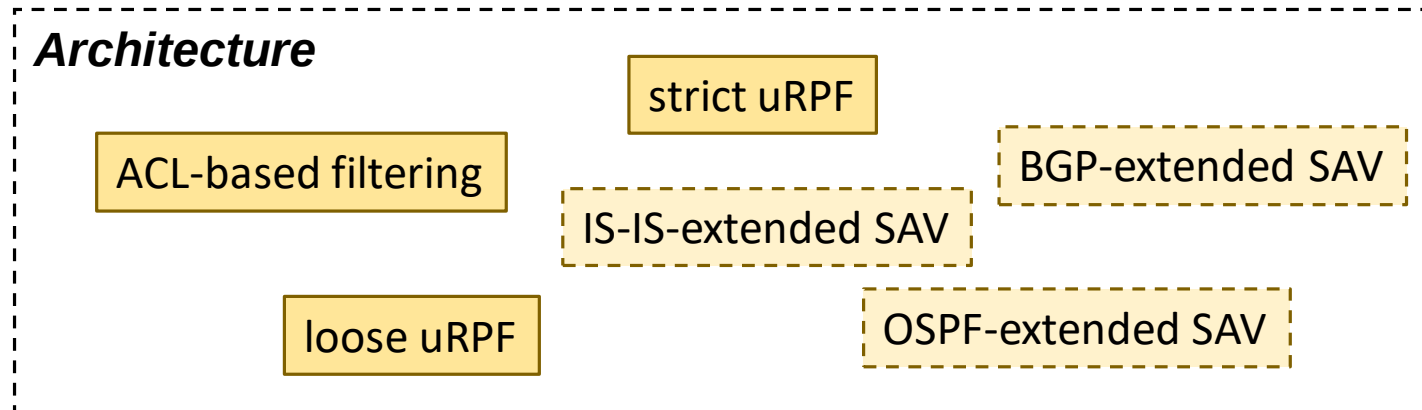
# Intra-domain SAVNET Architecture



# Conclusion

## □ Propose a high-level architecture design

- ◆ Three components: SAV Information Collector, SAV Information Manager, SAV Rule Generator
- ◆ Suitable to route-based SAV mechanisms



Thanks!