UniSAV: A Unified Framework for Internet-Scale Source Address Validation

Lancheng Qin, Libin Liu, Li Chen, Dan Li, Yuqian Shi, Hongbing Yang

Tsinghua University, Zhongguancun Laboratory

Necessity Source Address Validation (SAV)

- Source address spoofing leads to various malicious attacks [RFC 6959], represented by reflective DDoS attack
- Network operators deploy source address validation (SAV) to block the traffic with spoofing source IP addresses
- Since 2014, the MANRS initiatively has been calling on network operators to implement SAV as close to the source as possible

The Development of SAV in IETF

- The Best Current Practice (BCP) for SAV (i.e., BCP38 [RFC 2827]) was first published in 2000
  - BCP84 [RFC 3704 in 2004, RFC 8704 in 2020]
- The Source Address Validation Architecture (SAVA) [RFC 5210] was published in 2008
- IETF SAVI Working Group was formed in 2008 (concluded in 2018)
  - Standardize access-network SAV mechanisms that prevent nodes attached to the same IP link from spoofing each other's IP addresses
- IETF SAVNET Working Group was formed in 2022
  - Guide the development of new SAV mechanisms, including distributed protocols such as DSAV[1]

[1]https://datatracker.ietf.org/meeting/113/materials/slides-113-savnet-dsav-framework-01

The Development of SAV in IETF

- The Best Current Practice (BCP) for SAV (i.e., BCP38 [RFC 2827]) was first published in 2000
  - BCP84 [RFC 3704 in 2004, RFC 8704 in 2020]
- The Source Address Validation Architecture (SAVA) [RFC 5210] was published
- **However, recent SAV measurement studies show that the adoption of SAV is worrying low on the Internet**
  - Standardize access-network SAV mechanisms that prevent nodes attached to the same IP link from spoofing each other's IP addresses
- IETF SAVNET Working Group was formed in 2022
  - Guide the development of new SAV mechanisms, including distributed protocols such as DSAV[1]

[1]https://datatracker.ietf.org/meeting/113/materials/slides-113-savnet-dsav-framework-01

Problems of SAV Deployment

It remains a significant challenge to promote the wide deployment of SAV

☐ Lack of understanding

- ◆ Many network operators lack the technical knowledge, understanding, and practical experience. They do not know how SAV works or how to deploy or operate a specific SAV mechanism

☐ Lack of open source implementation

- ◆ There is very limited open source effort on SAV, it is difficult to form an acknowledged baseline standard, leading to differences in understanding and implementation of the same SAV mechanism

☐ Performance concerns

- ◆ Network operators cannot test and evaluate the performance of different SAV mechanisms, due to the lack of a publicly available testbed. Without sufficient tests, network operators hesitate to deploy SAV mechanisms in their networks

Problems of SAV Deployment

It remains a significant challenge to promote the wide deployment of SAV

Lack of understanding

- Many people lack the technical knowledge, understanding, and practical experience. They do not know how SAV works or how to deploy or operate a specific SAV mechanism

UniSAV provides an open platform to implement and emulate different SAV mechanisms

... d baseline standard, leading to differences in understanding and implementation of the same SAV mechanism

Performance concerns

- People cannot test and evaluate the performance of different SAV mechanisms, due to the lack of a publicly available testbed. Without sufficient tests, network operators hesitate to deploy SAV mechanisms in their networks

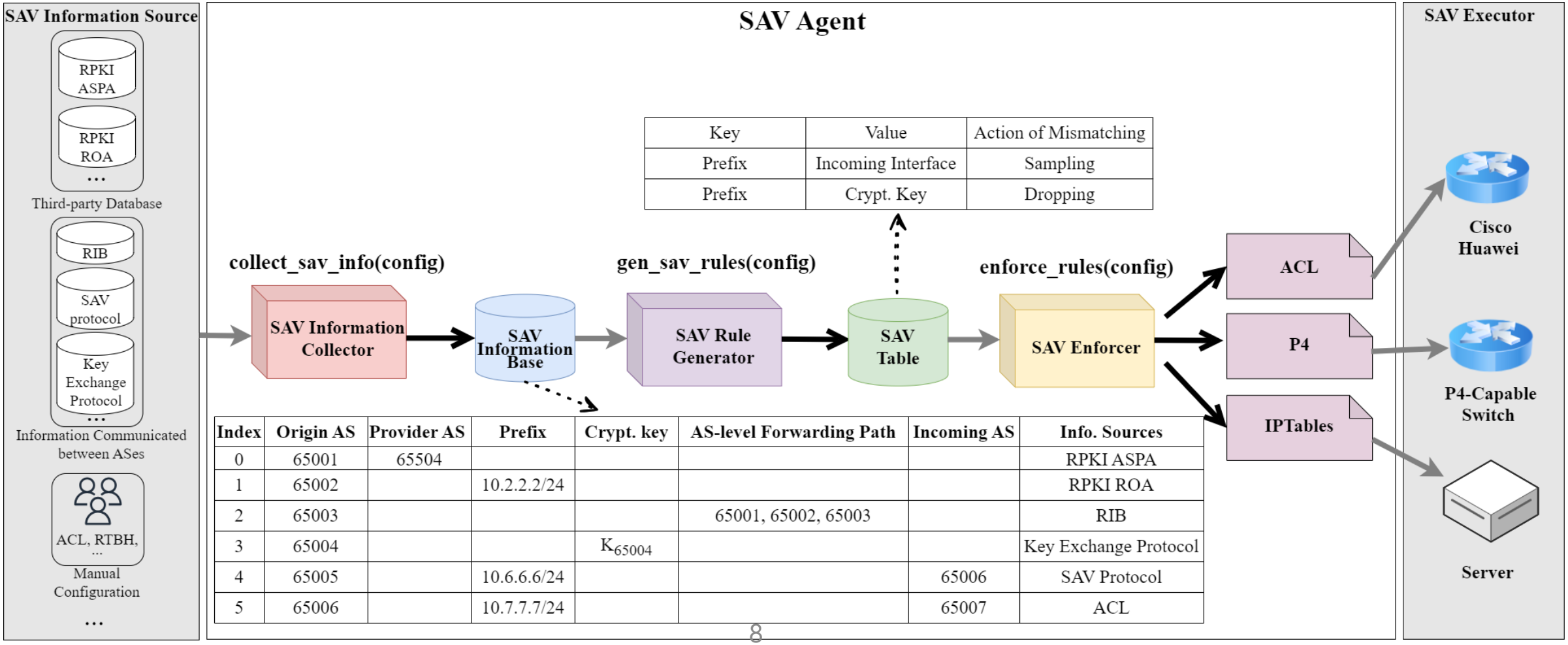
Summary of Existing SAV Mechanisms

Existing SAV Mechanisms	SAV Information Sources			SAV Rule Format	
	Manual Configuration	Third-party Public Database	Communication between ASes	(Prefix, Incoming Interface)	(Prefix, Crypt. Key)
uRPF, SAVE, DSAV			√	√	
Passport, EPIC, PISKES			√		√
BAR-SAV		√	√	√	
Ingress Filtering	√			√	

- This work first revisits existing SAV mechanisms from a high-level perspective and summarize a unified architecture that can cover all existing SAV mechanisms and possible future ones
- Existing SAV mechanisms encompass the same basic functions for collecting SAV information, generating SAV rules, and executing SAV filtering
  - ◆They can be different in SAV information source or SAV rule format

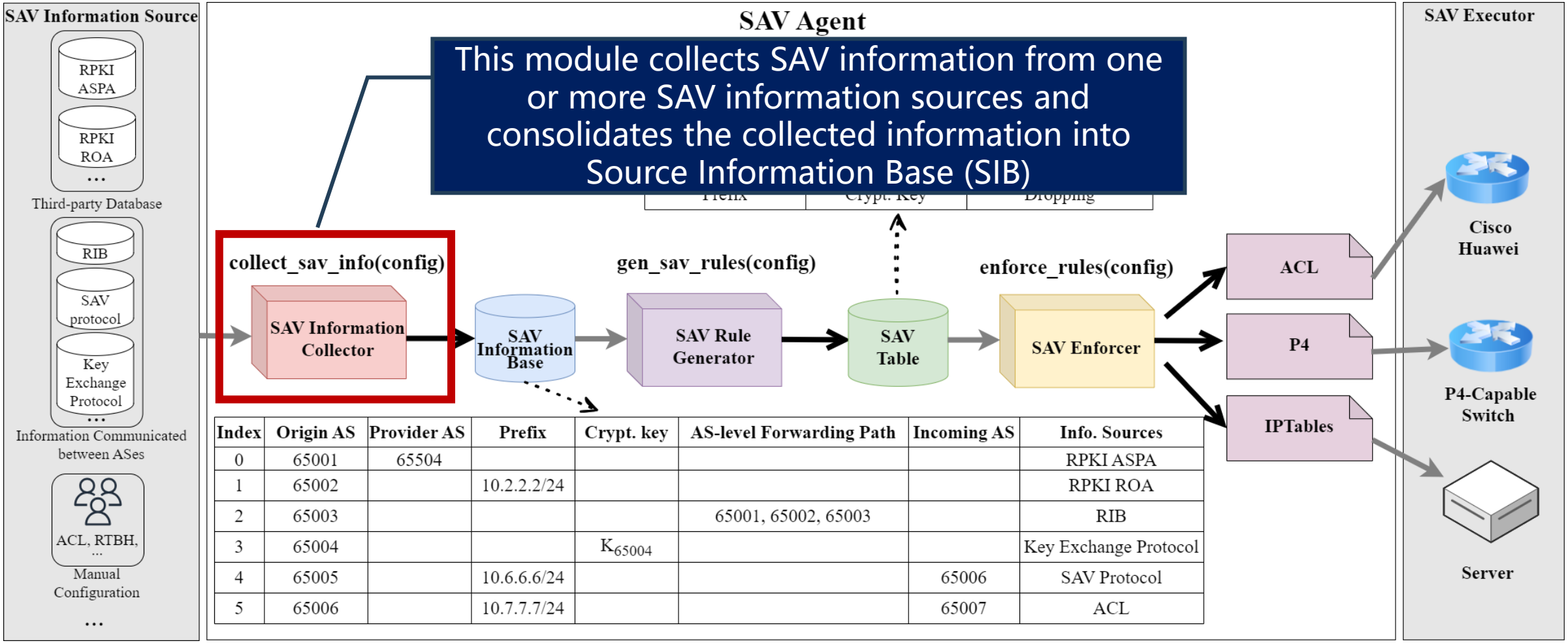
UniSAV Architecture

SAV Agent is the core concept of UniSAV, which has three modules and two data structures



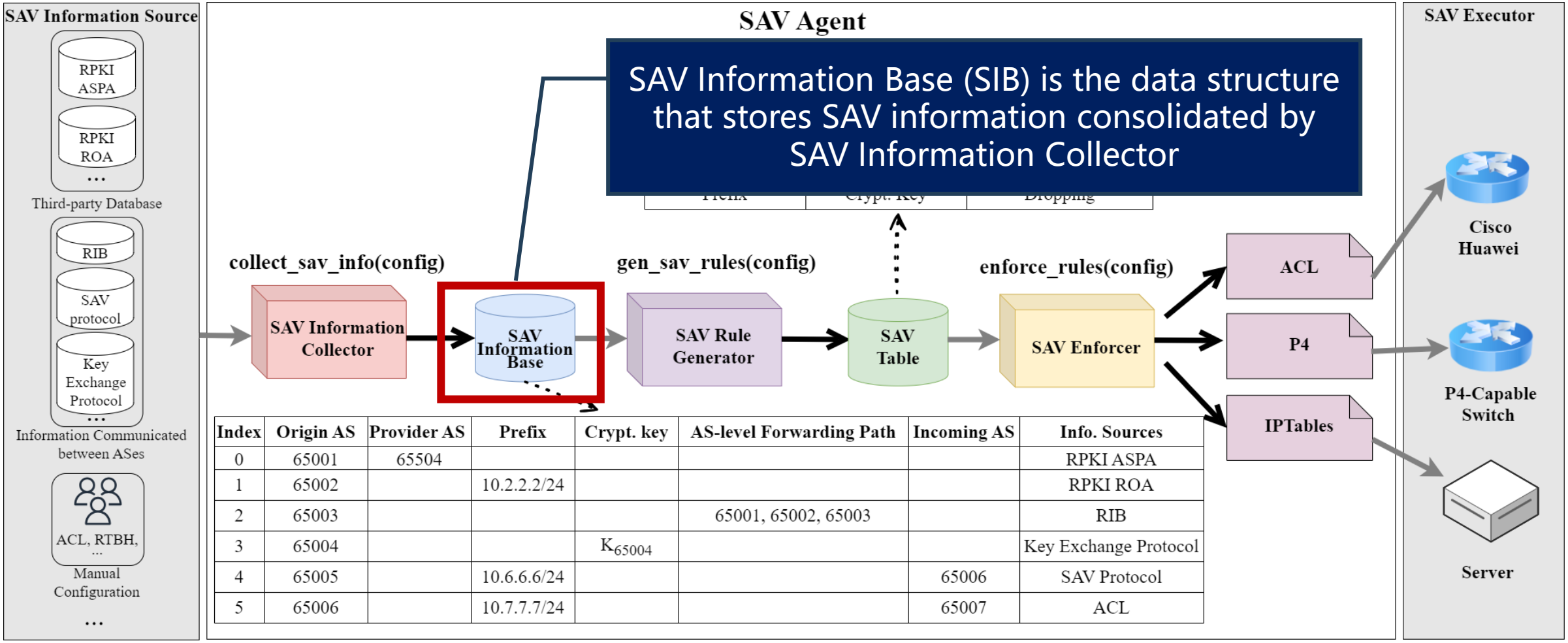
Index	Origin AS	Provider AS	Prefix	Crypt. key	AS-level Forwarding Path	Incoming AS	Info. Sources
0	65001	65504					RPKI ASPA
1	65002		10.2.2.2/24				RPKI ROA
2	65003				65001, 65002, 65003		RIB
3	65004			K_{65004}			Key Exchange Protocol
4	65005		10.6.6.6/24			65006	SAV Protocol
5	65006		10.7.7.7/24			65007	ACL

UniSAV Architecture



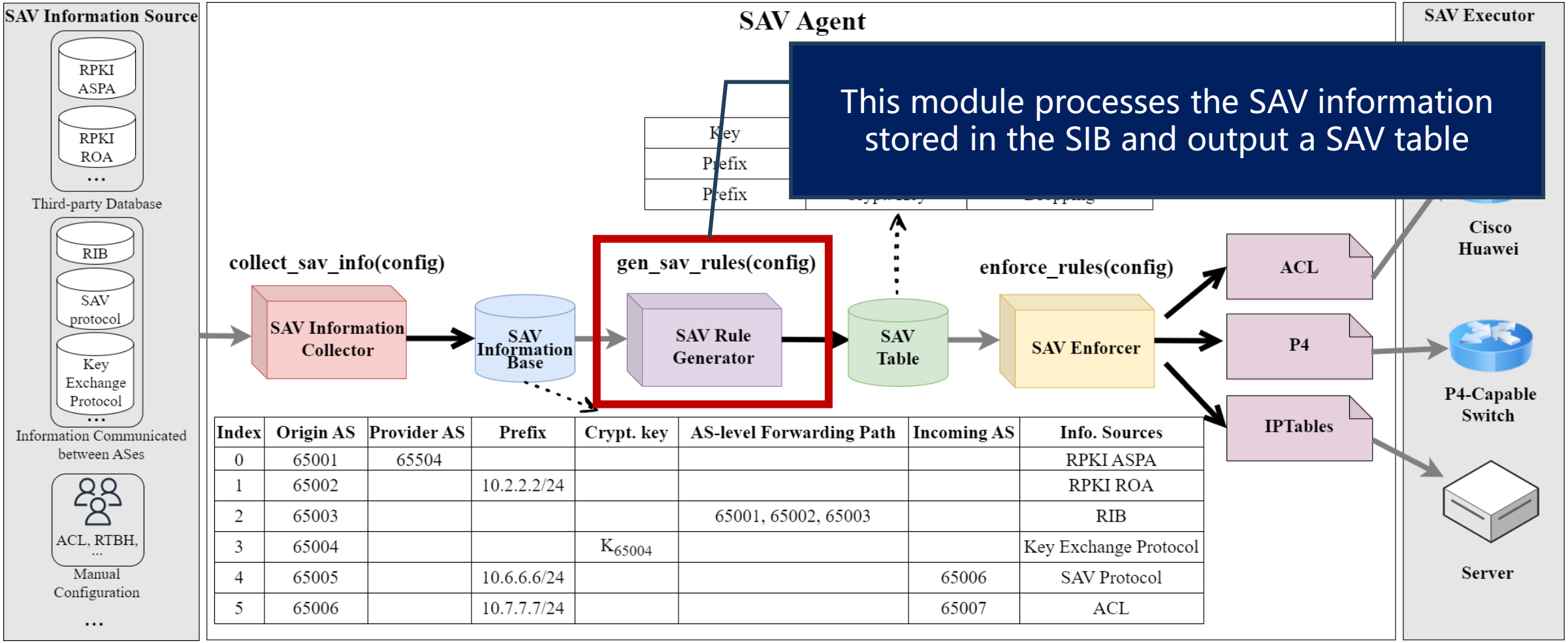
Index	Origin AS	Provider AS	Prefix	Crypt. key	AS-level Forwarding Path	Incoming AS	Info. Sources
0	65001	65504					RPKI ASPA
1	65002		10.2.2.2/24				RPKI ROA
2	65003				65001, 65002, 65003		RIB
3	65004			K_{65004}			Key Exchange Protocol
4	65005		10.6.6.6/24			65006	SAV Protocol
5	65006		10.7.7.7/24			65007	ACL

UniSAV Architecture



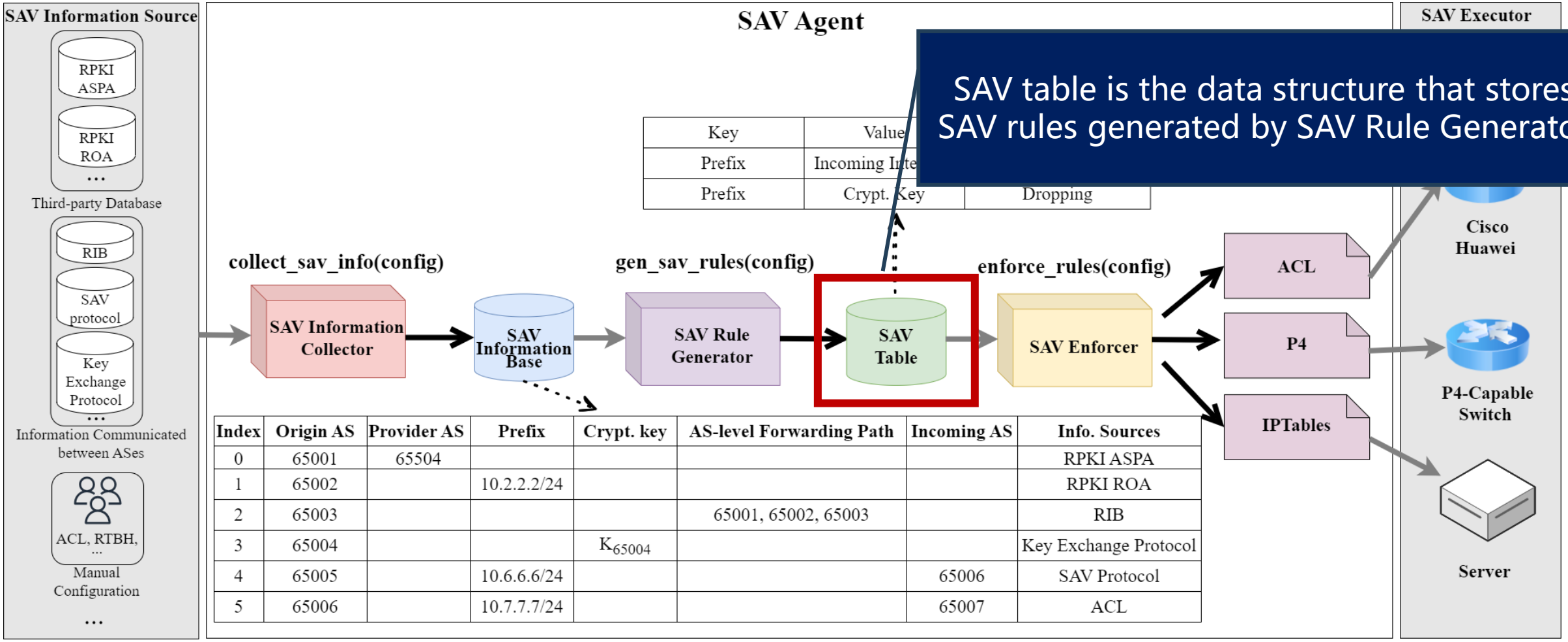
Index	Origin AS	Provider AS	Prefix	Crypt. key	AS-level Forwarding Path	Incoming AS	Info. Sources
0	65001	65504					RPKI ASPA
1	65002		10.2.2.2/24				RPKI ROA
2	65003				65001, 65002, 65003		RIB
3	65004			K_{65004}			Key Exchange Protocol
4	65005		10.6.6.6/24			65006	SAV Protocol
5	65006		10.7.7.7/24			65007	ACL

UniSAV Architecture



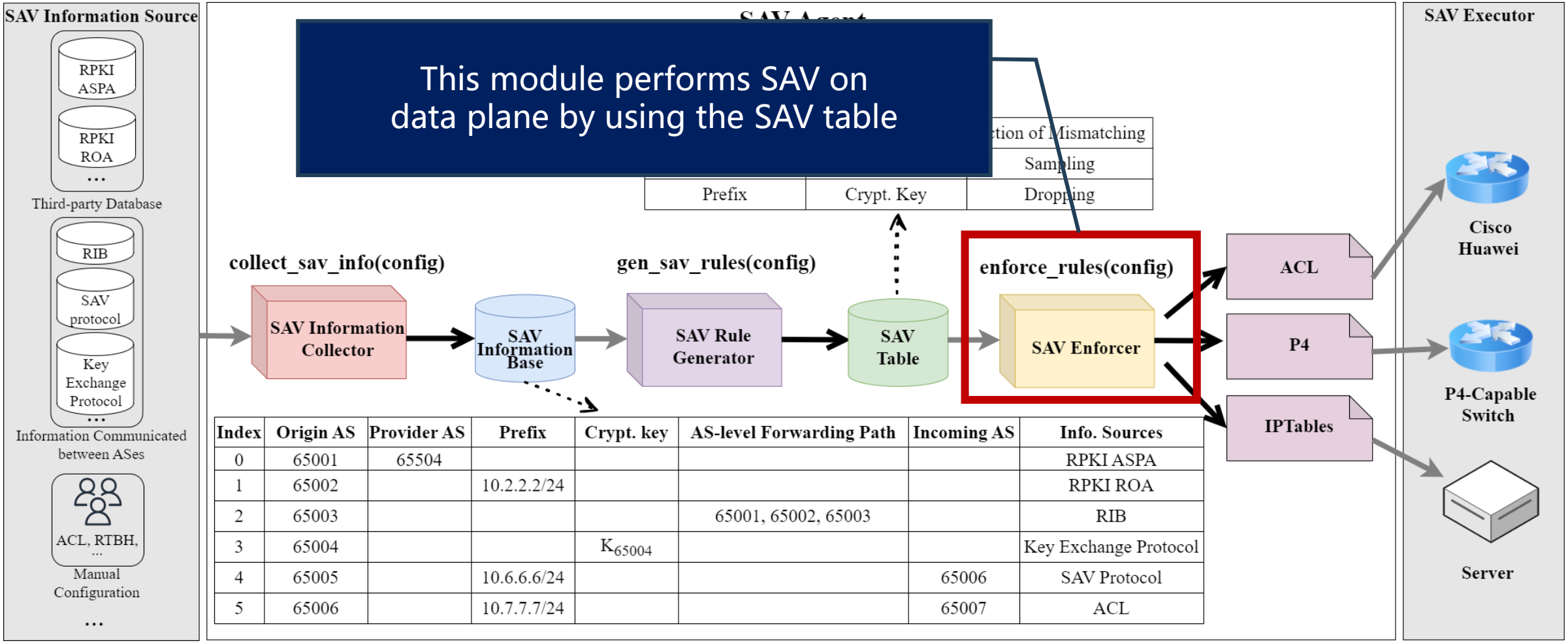
Index	Origin AS	Provider AS	Prefix	Crypt. key	AS-level Forwarding Path	Incoming AS	Info. Sources
0	65001	65504					RPKI ASPA
1	65002		10.2.2.2/24				RPKI ROA
2	65003				65001, 65002, 65003		RIB
3	65004			K_{65004}			Key Exchange Protocol
4	65005		10.6.6.6/24			65006	SAV Protocol
5	65006		10.7.7.7/24			65007	ACL

UniSAV Architecture



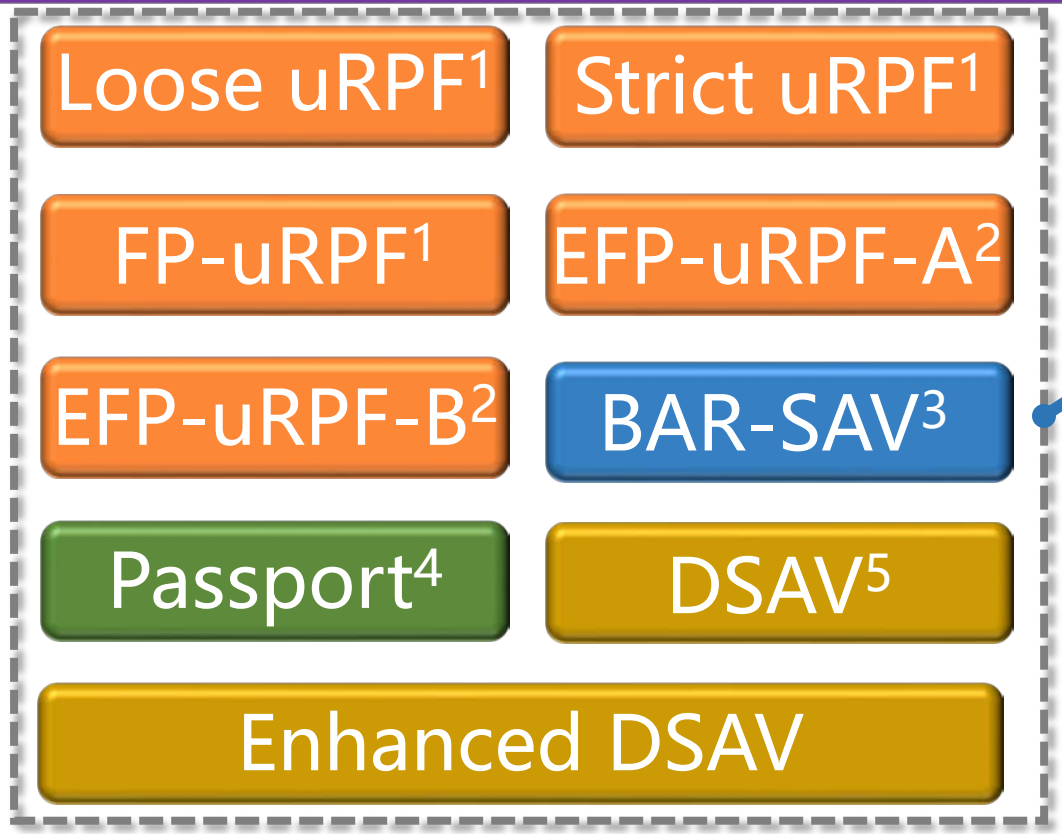
Index	Origin AS	Provider AS	Prefix	Crypt. key	AS-level Forwarding Path	Incoming AS	Info. Sources
0	65001	65504					RPKI ASPA
1	65002		10.2.2.2/24				RPKI ROA
2	65003				65001, 65002, 65003		RIB
3	65004			K_{65004}			Key Exchange Protocol
4	65005		10.6.6.6/24			65006	SAV Protocol
5	65006		10.7.7.7/24			65007	ACL

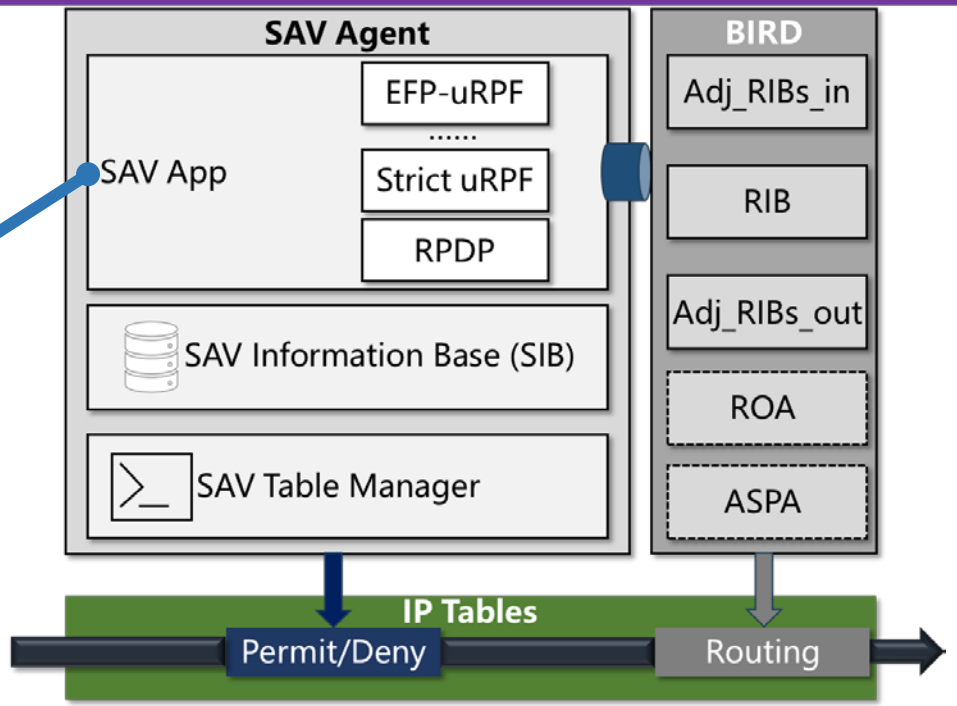
UniSAV Architecture



Index	Origin AS	Provider AS	Prefix	Crypt. key	AS-level Forwarding Path	Incoming AS	Info. Sources
0	65001	65504					RPKI ASPA
1	65002		10.2.2.2/24				RPKI ROA
2	65003				65001, 65002, 65003		RIB
3	65004			K_{65004}			Key Exchange Protocol
4	65005		10.6.6.6/24			65006	SAV Protocol
5	65006		10.7.7.7/24			65007	ACL

Implementations with UniSAV

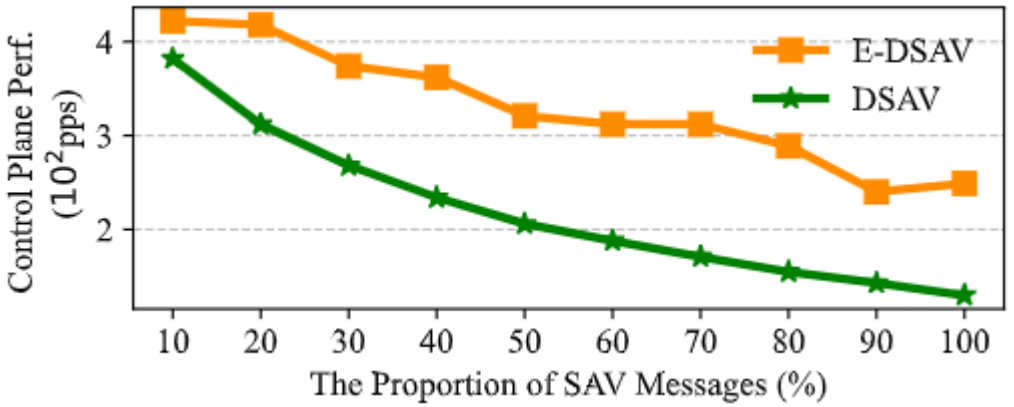




[1]RFC3704: https://datatracker.ietf.org/doc/html/rfc3704
[2]RFC8704: https://datatracker.ietf.org/doc/html/rfc8704
[3]https://datatracker.ietf.org/doc/draft-ietf-sidrops-bar-sav/
[4]Passport: Secure and Adoptable Source Authentication, NSDI 2008
[5]https://datatracker.ietf.org/meeting/113/materials/slides-113-savnet-dsav-framework-01

E-DSAV

- Enhanced DSAV (E-DSAV) makes the three improvements upon DSAV[1].
  - Reducing the size of the communicated information by **using *ASN* to replace source prefixes** of the corresponding AS within the communicated messages
  - Implementing a **neighbor discovery mechanism** for building neighbor relationships
  - **Decoupling control and data channels**
    - Only the control channel reuses the BGP connection of the underlying router.
    - For ASNs, the E-DSAV uses a separate data channel.



[1]https://datatracker.ietf.org/meeting/113/materials/slides-113-savnet-dsav-framework-01

SAV Benchmark

- Real-world AS-level network topology
  - Using real BGP data from public route collectors provided by RouteViews[1] and RIPE RIS[2]
  - Parsing and extracting *AS path* attribute from the BGP data and obtaining neighboring relation between ASes
  - Creating links for the neighboring ASes to build the AS-level Internet topology
  - Obtaining the business relationship between ASes according to the data from CAIDA[3]
- Sub-graphs generated based on the full topology
  - A connected component of the full topology
  - Assigning routing policies based on the business relationship and the valley-free principle
- Three classic scenarios
  - Symmetric routing, NO_EXPORT, direct server return (DSR)

[1]http://www.routeviews.org/routeviews/
[2]https://www.ripe.net/analyse/internet-measurements/routing-information-service-ris/ris-raw-data
[3]https://catalog.caida.org/dataset/as_relationships_serial_1

Emulation Setups

- Testbed
  - Using a x86 server machine with two 2.2GHz 26-core Intel Xeon Gold 5320 CPUs, 256GB DDR4 RAM, 2 1TB SSDs, and 1 12TB SAS HDDs
  - Running Ubuntu 22.04.2 LTS with kernel version 5.15.0
  - Using Docker 24.0.2 with the image ubuntu:22.04 for each container to emulate an AS
  - Running BIRD 2.0.12 as the AS border router and using iptables 1.8.7 to filter packets
- Methodology
  - Evaluating the performance of these mechanisms in terms of **validation accuracy, control plane performance, data plane performance, and scalability**
  - Using the network topology with 50 ASes
    - Except for the scalability experiments
  - Varying the deployment ratios of the SAV mechanisms from 10% to 100%

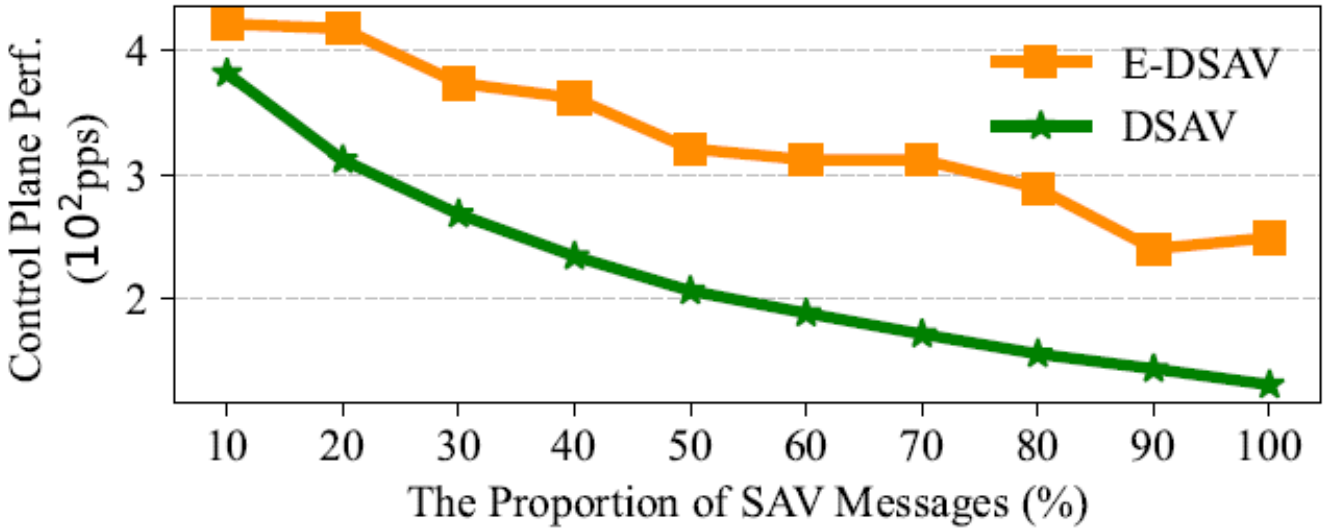
SAV Accuracy

Scenarios	Loose uRPF	Strict uRPF	FP-uRPF	EFP-uRPF-A	EFP-uRPF-B	BAR-SAV	Passport	DSAV	E-DSAV
Symmetric Routing	IP	√	√	√	IP	√	√	√	√
NO-EXPORT	IP	IB	IB	IB	IP	IB	√	√	√
DSR	IP	IB	IB	IB	IP & IB	IB	√	√	√

The SAV accuracy of different SAV mechanisms implemented on top of UniSAV in the scenarios including symmetric routing, NO-EXPORT, and Direct Server Return (DSR) (√: Accurate Validation, IP: Improper Permit, IB: Improper Block)

- The table shows SAV accuracy results of different SAV mechanisms in the three scenarios
  - In symmetric routing scenario, both Loose uRPF and EFP-uRPF with algorithm B may improperly permit spoofing traffic
  - In NO-EXPORT and DSR scenarios, both Loose uRPF and EFP-uRPF with algorithm B may improperly permit spoofing traffic; Strict uRPF, FP-uRPF, EFP-uRPF with algorithm A and B, and BAR-SAV may improperly block legitimate traffic
  - The results are the same as the theoretical analysis in [draft-ietf-savnet-inter-domain-problem-statement]

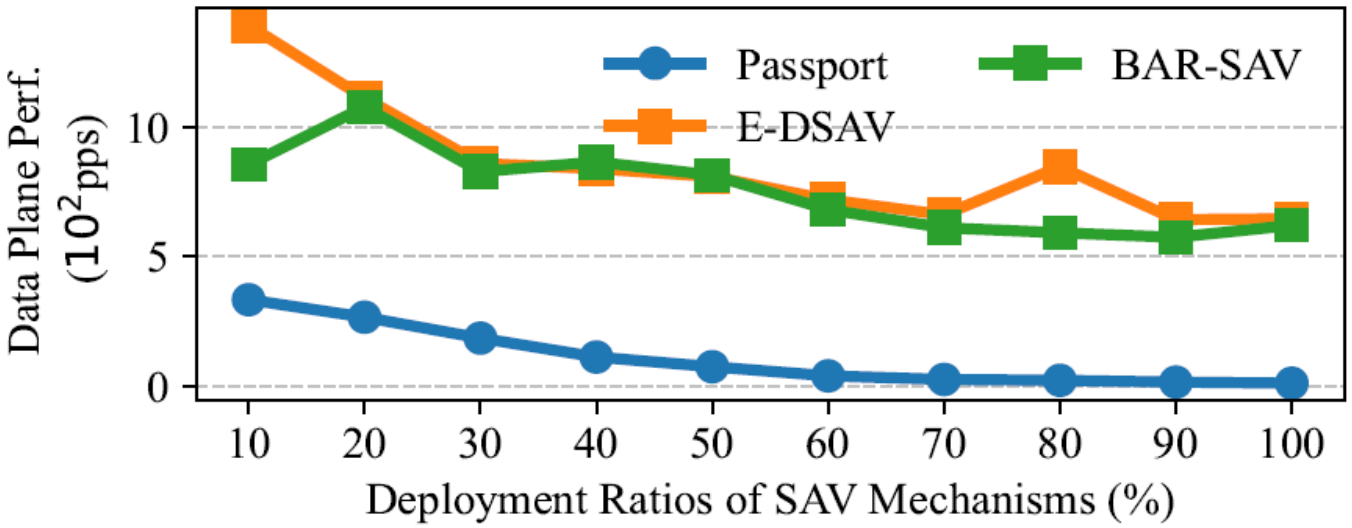
Control Plane Performance



The control plane performance for processing pure BGP messages in terms of packets per second with varying proportions of SAV messages. The proportions of SAV messages are calculated by the number of SAV messages over the total number of messages for SAV and pure BGP

- The control plane performance when processing pure BGP messages under varying proportions of SAV messages
  - Both DSAV and E-DSAV impact the efficiency of the control plane in dealing with pure BGP messages, but the underlying reasons are different
    - For E-DSAV, the limitations arise from resource constraints within each container.
    - Instead, DSAV not only needs to communicates more messages but also necessitates additional resources for parsing the delivered SAV messages

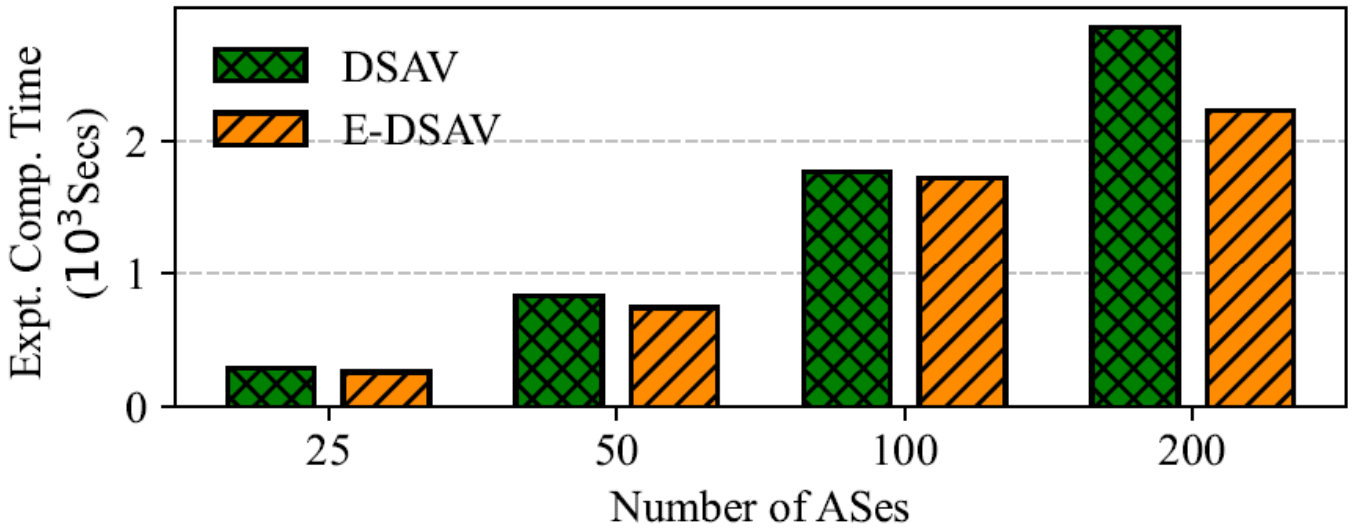
Data Plane Performance



The data plane forwarding performance of the SAV mechanisms with varying deployment ratios. Here, we employ iptables to execute SAV within the data plane. We implement a traffic generation tool to generate packets with fixed 1.5KB to evaluate the data plane forwarding performance in terms of packets per second.

- The data plane performance results of BAR-SAV, Passport, and E-DSAV
  - Passport always has a more significant impact on data plane forwarding performance than other SAV mechanisms
  - The data plane forwarding performance of each SAV mechanism decreases as the deployment ratio increases, because the size of the SAV table within each AS increases with the increase of deployment ratio, larger SAV table results in longer query time for each incoming packet

Scalability of UniSAV



The experiment completion time of UniSAV across different network scales. We vary the network scales by increasing the number of ASes for the testbed experiments, and then calculate the experiment completion time. The experiment completion time is the longest time elapsed from launching the Docker environment to generating complete SAV Table among all ASes.

- The total experiment time of UniSAV with AS numbers from 25 to 200, by taking DSAV and E-DSAV as examples.
  - Both the experiment completion times for DSAV and E-DSAV increase along with the increase of AS numbers.
  - Compared with DSAV, E-DSAV shows a slower growth trend with the increase of network size. This is because E-DSAV converges faster than DSAV.

Our Related Works in IETF

- draft-ietf-savnet-intra-domain-problem-statement
- draft-ietf-savnet-inter-domain-problem-statement
- draft-ietf-savnet-intra-domain-architecture
- draft-wu-savnet-inter-domain-architecture
- draft-li-savnet-source-prefix-advertisement
- draft-li-sidrops-bicone-sav
- draft-chen-bmwg-savnet-sav-benchmarking

Thank you!

Making the Internet work better!