ESAV: An End-to-End Data-Plane Approach for Source Address Validation

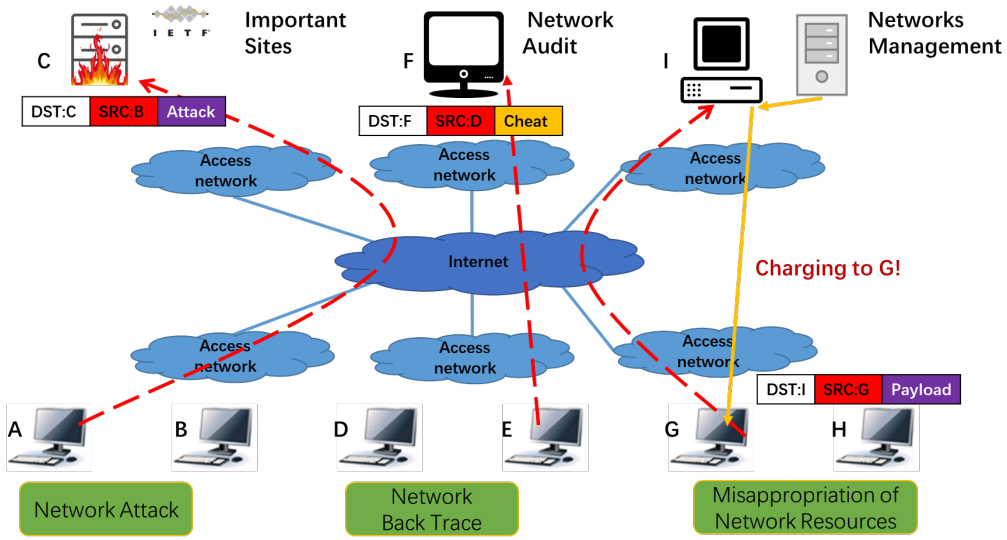
Tsinghua University

Ke Xu
Jianping Wu
Xiaoliang Wang
Yangfei Guo

2022.03

Motivation

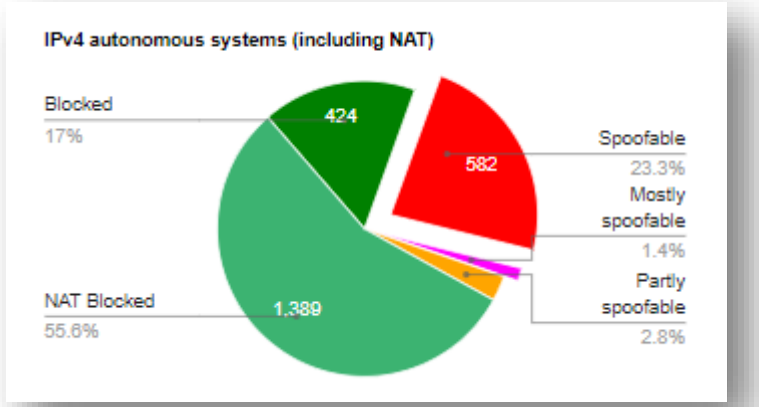
The Internet architecture includes no explicit notion of packet-level authenticity. A long-recognized consequence of this weakness is the ability to forge or “spoof” IP packet headers.

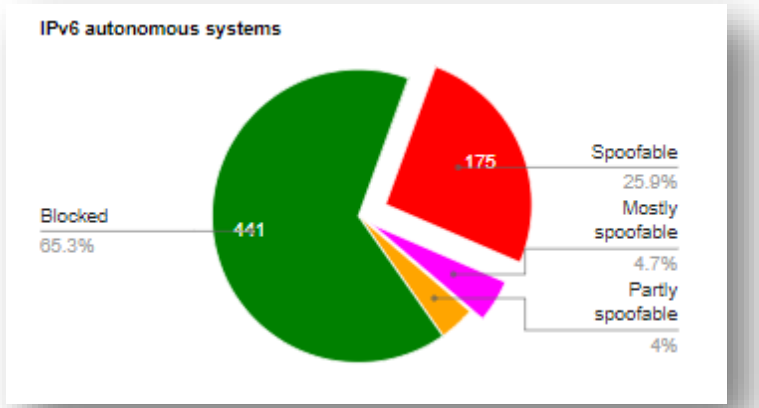


Vulnerability: It is difficult to resist attacks by disabling the IP source address.

Traceability: Attackers could conceal location and identity.

Management: It is difficult to realize billing and other management through the IP source address.





The root of IP forwarding lies in **spoofing** and difficult to **back trace.**

source https://spoofer.caida.org/summary.php

Goals

Clear security benefits

Having clear and consistent security benefits in various scenarios is the biggest deployment incentive for SAV.

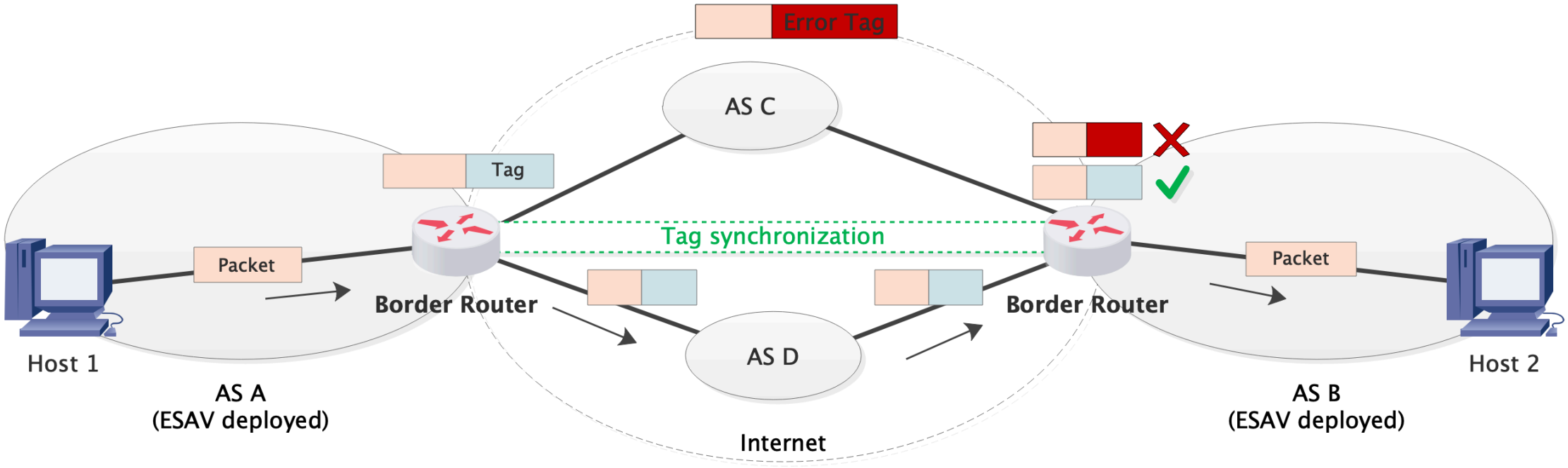
Scalability of deployment

Scalability from partial to large-scale deployments needs to be supported.

Flexibility of validation granularity

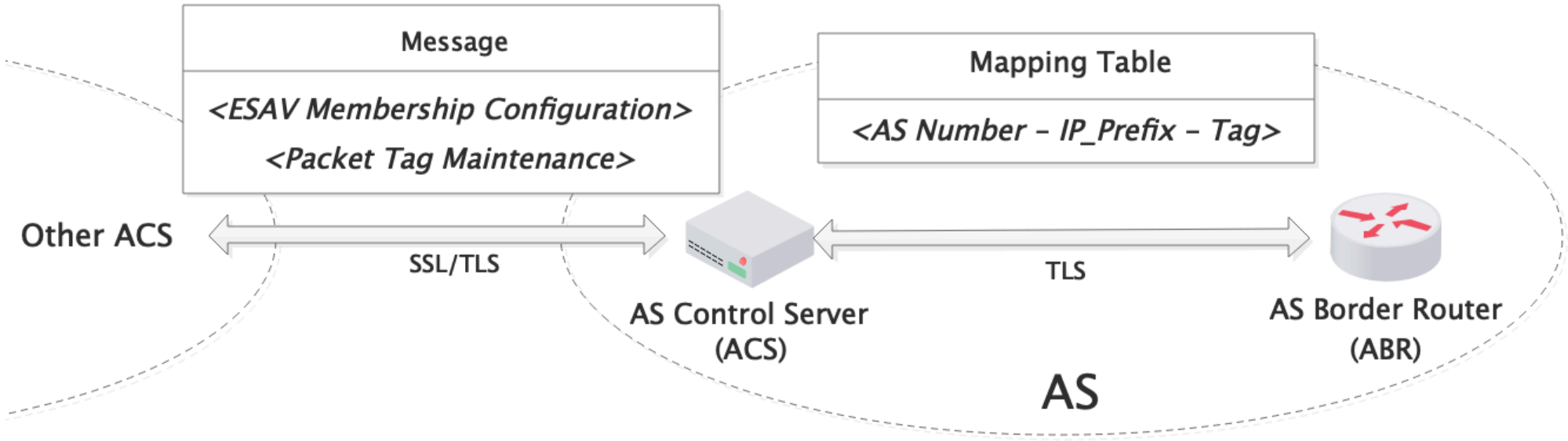
Flexible requirements for SAV granularity for different application scenarios need to be achieved.

Overview



1. End-to-end packet tag maintenance between ASes, using the existing One-Time Password.
2. Legitimate packets add the tag at the AS border and validate at the destination AS border.
3. Packets missing tags or carrying error tags will be filtered.

Overview



1. An AS with ESAV include one AS Control Server (ACS) and AS Border Routers (ABR).

2. ABR is responsible for various operations of packet tagging.

3. ACS is responsible for providing the ABR with the Tag corresponding to the different IP prefix pairs.

4. The interaction between ACSs completes membership configuration and tag negotiation, which requires the support of consensus algorithm and existing infrastructure(such as RPKI) .

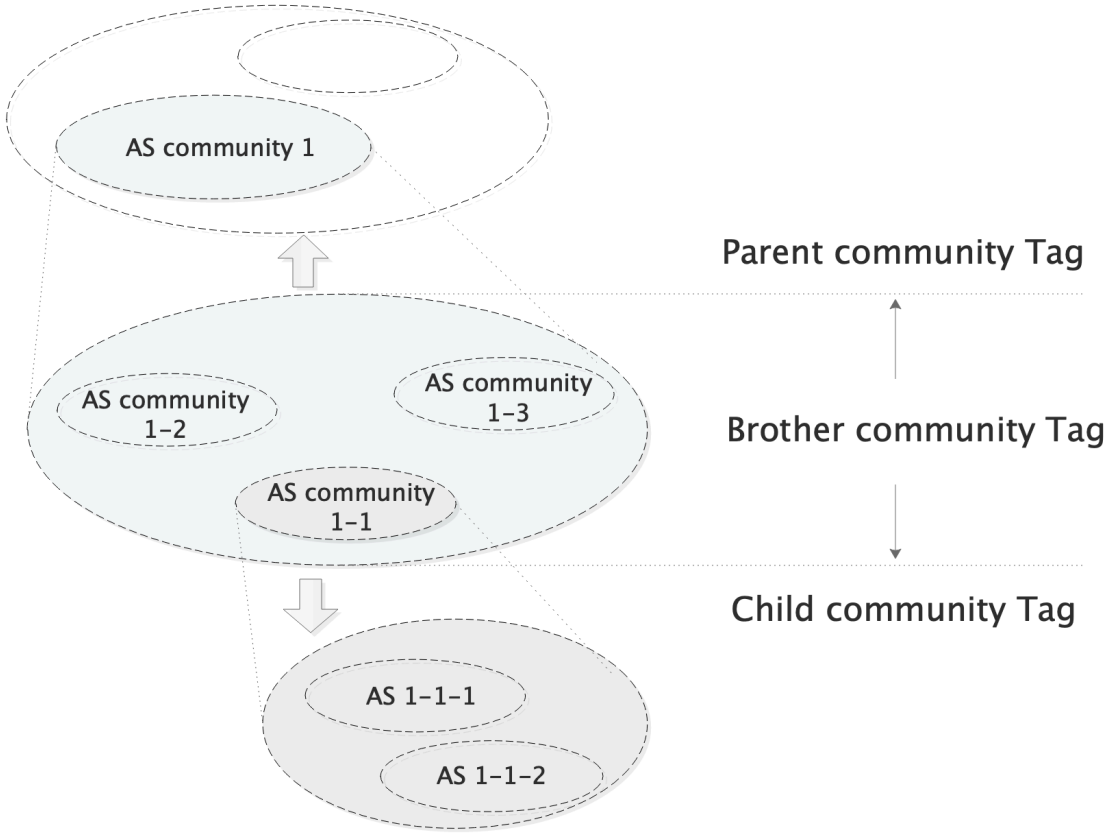
Hierarchical Structure

1. ASes can form the logical hierarchical AS communities.

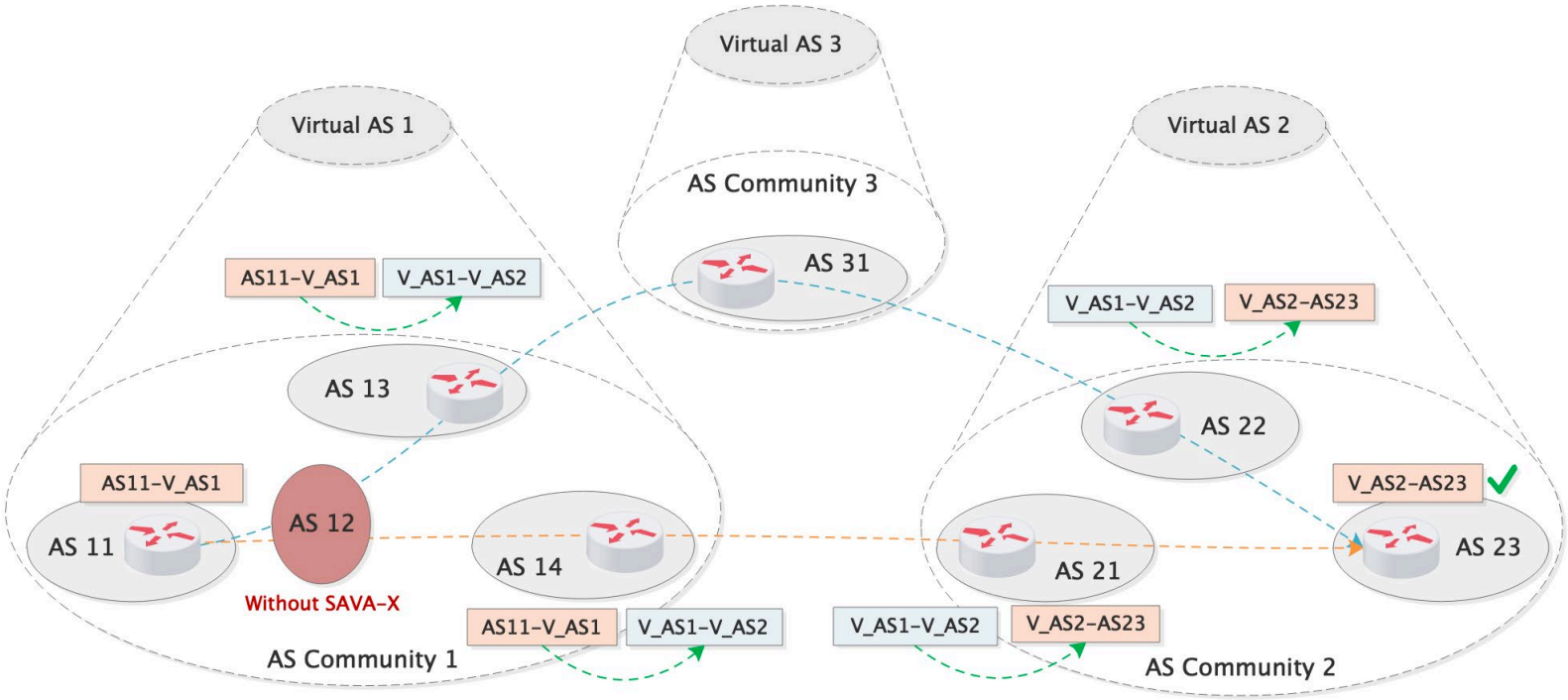
2. End-to-end Tag is only maintained between ASes within the same community.

3. Traffic entering or leaving the AS community is operated by the border ASes for Tag replacement.

- ✓ Reduce the size of the tags maintained between ADs.
- ✓ Hierarchy effectively blocks external changes and provides scalability in large scale deployments.
- ✓ Cross-layer verification will filter mali- cious traffic as early as possible to avoid wasting resources.



Tag Replacement

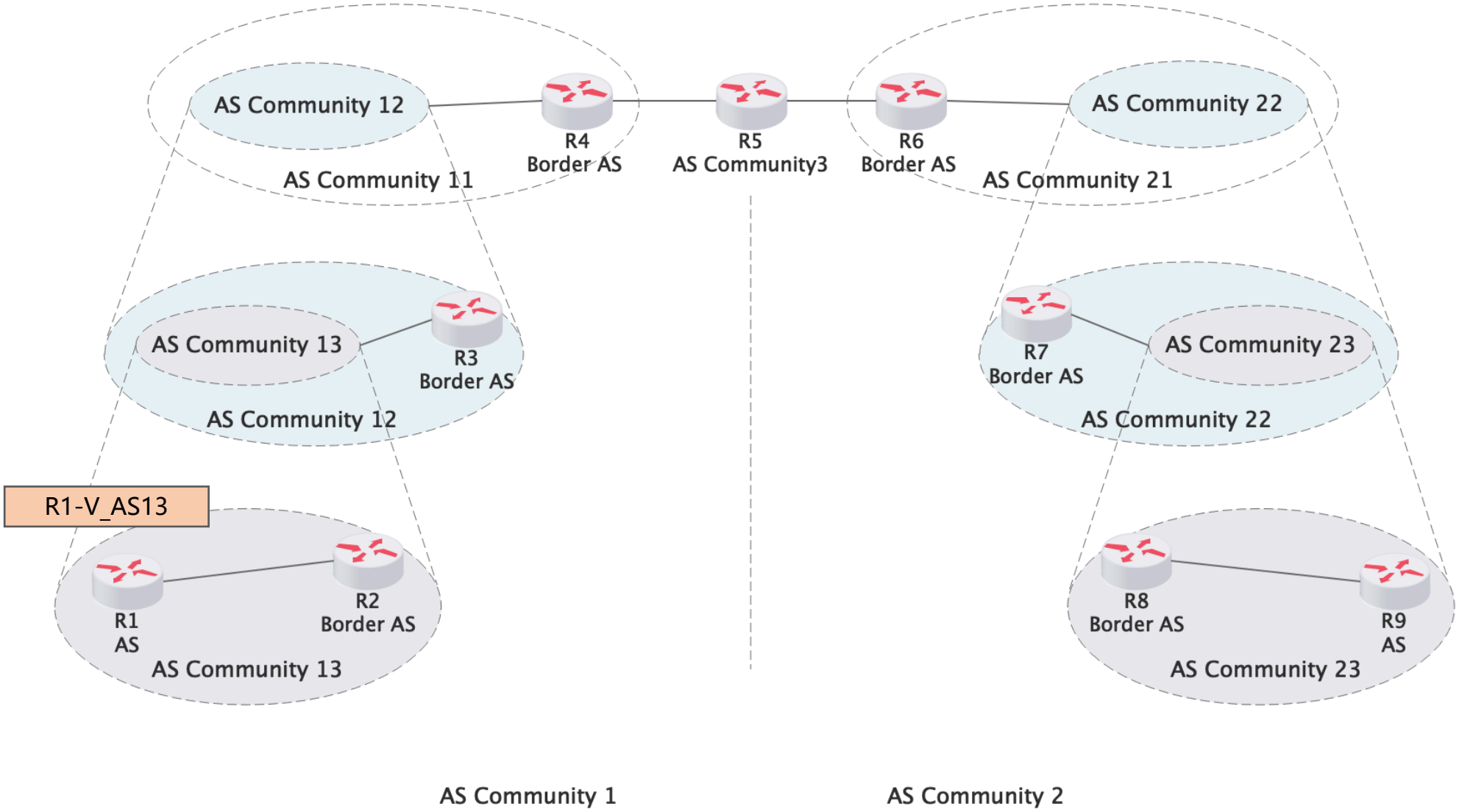


1. Each community contains a virtual AS, using the virtual AS to achieve inter-community Tag maintenance.

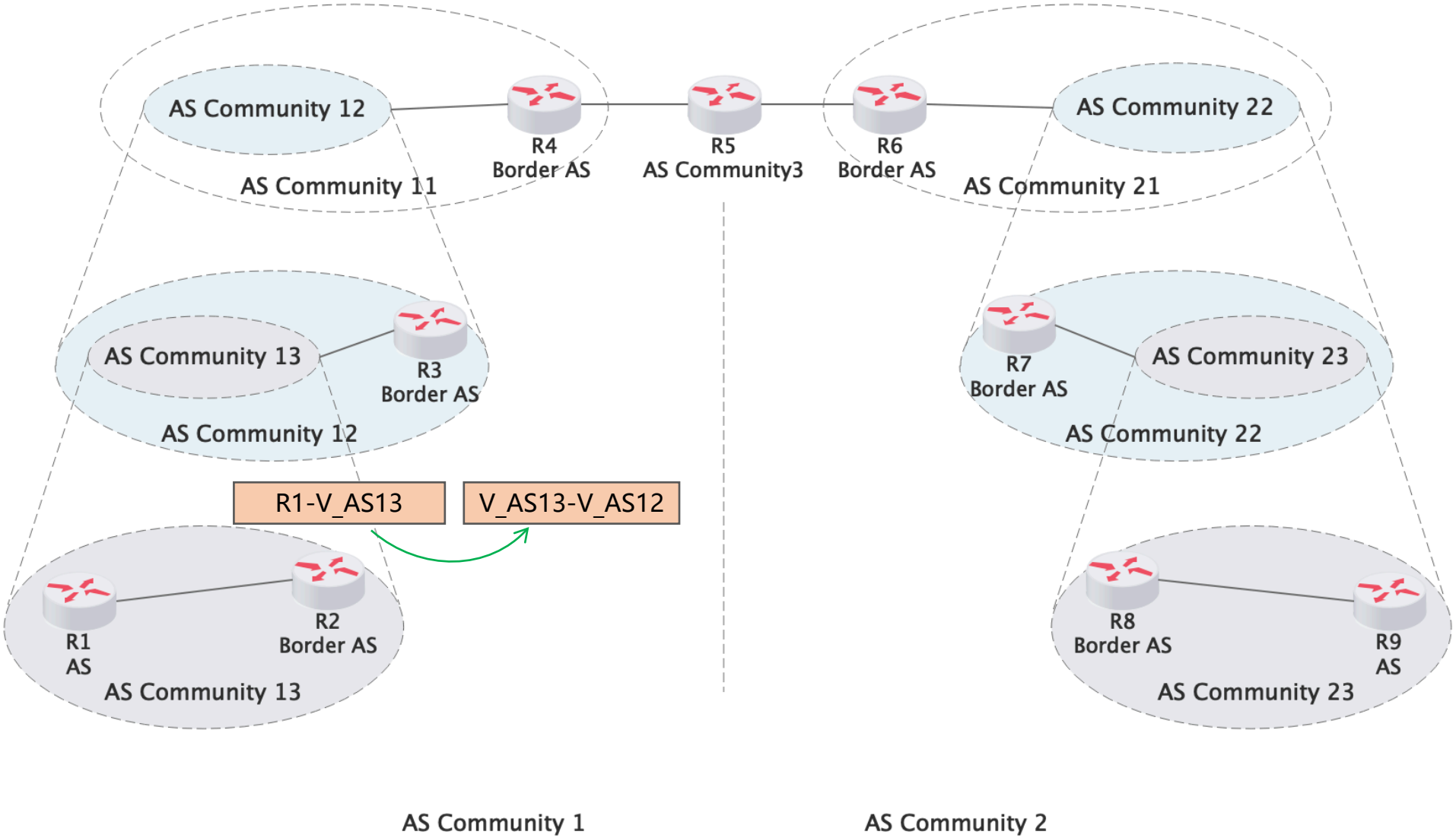
2. The community's border AS is responsible for Tag validation and replacement of packets entering and leaving the community.

3. Guaranteed Tag replacement is independent of the actual forwarding path, simplifying border router verification logic.

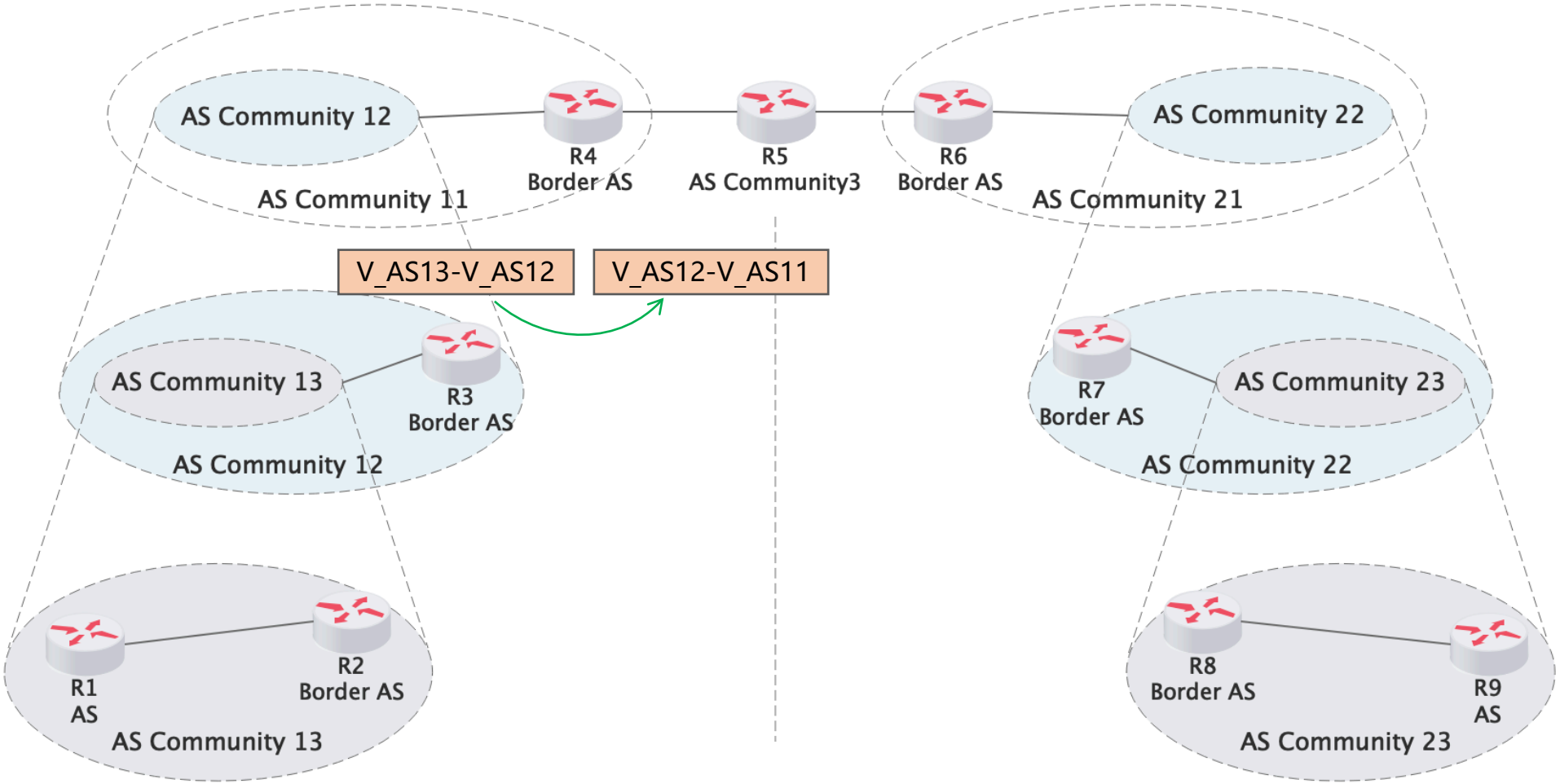
Example


AS Community 12
R4
Border AS
R5
AS Community3
R6
Border AS
AS Community 22
AS Community 11
AS Community 21
AS Community 13
R3
Border AS
AS Community 12
R7
Border AS
AS Community 23
AS Community 22
R1-V_AS13
R1
AS
R2
Border AS
AS Community 13
R8
Border AS
R9
AS
AS Community 23
AS Community 1
AS Community 2

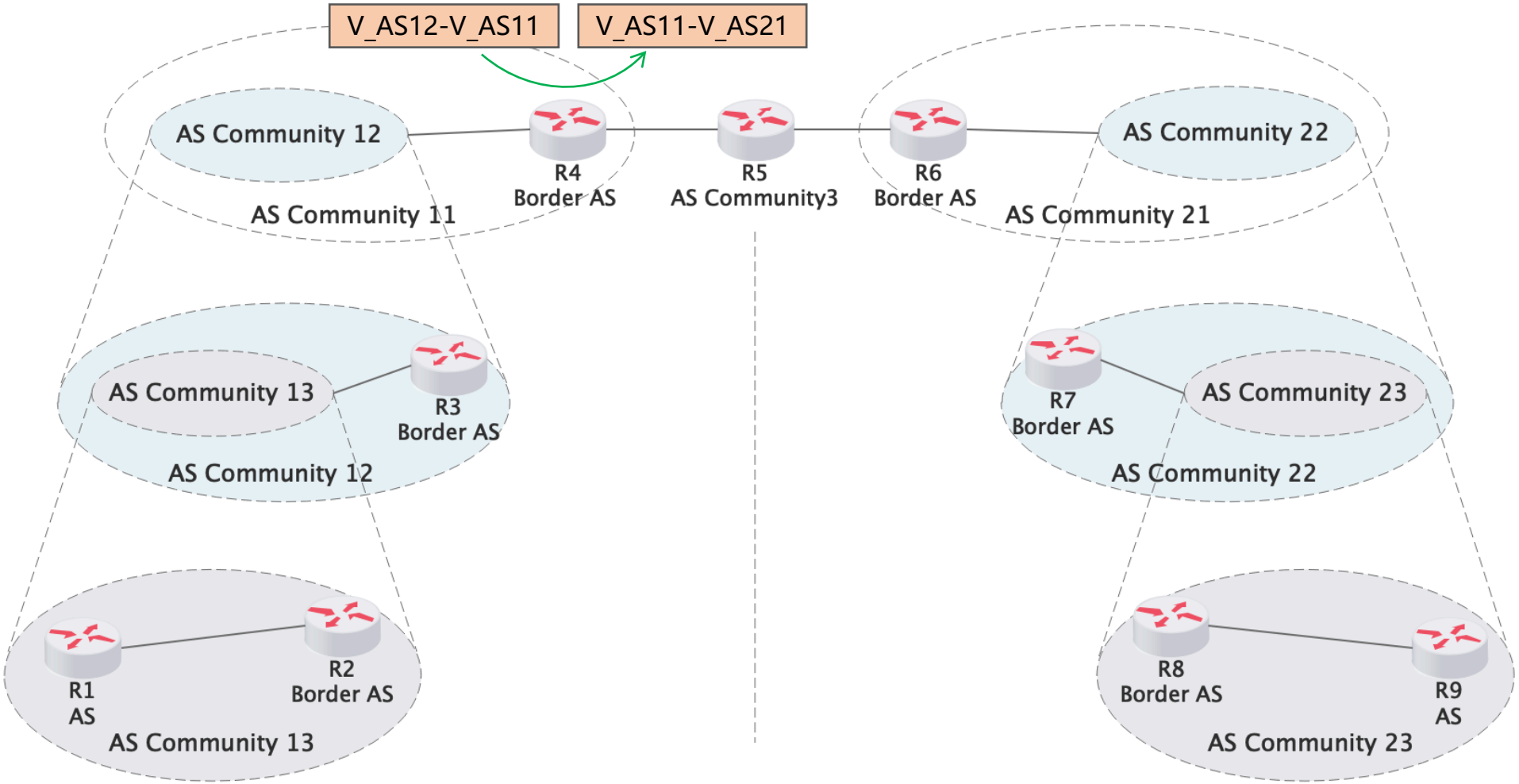
Example


AS Community 12
R4
Border AS
R5
AS Community3
R6
Border AS
AS Community 22
AS Community 11
AS Community 21
AS Community 13
R3
Border AS
AS Community 12
R7
Border AS
AS Community 23
AS Community 22
R1-V_AS13
V_AS13-V_AS12
R1
AS
R2
Border AS
AS Community 13
R8
Border AS
R9
AS
AS Community 23
AS Community 1
AS Community 2

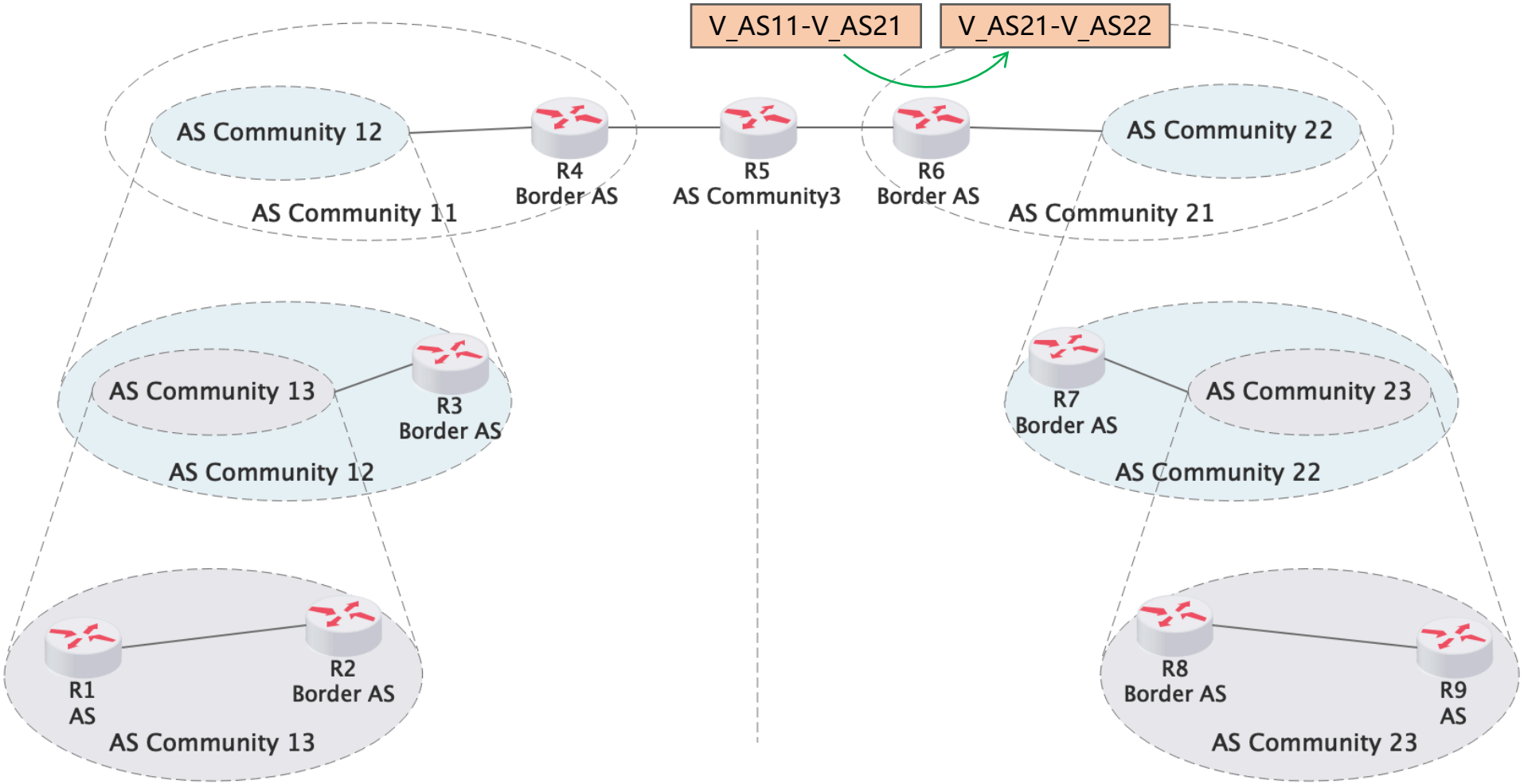
Example


AS Community 12
R4
Border AS
R5
AS Community3
R6
Border AS
AS Community 22
AS Community 11
AS Community 21
V_AS13-V_AS12
V_AS12-V_AS11
AS Community 13
R3
Border AS
AS Community 12
R7
Border AS
AS Community 23
AS Community 22
R1
AS
R2
Border AS
AS Community 13
R8
Border AS
R9
AS
AS Community 23
AS Community 1
AS Community 2

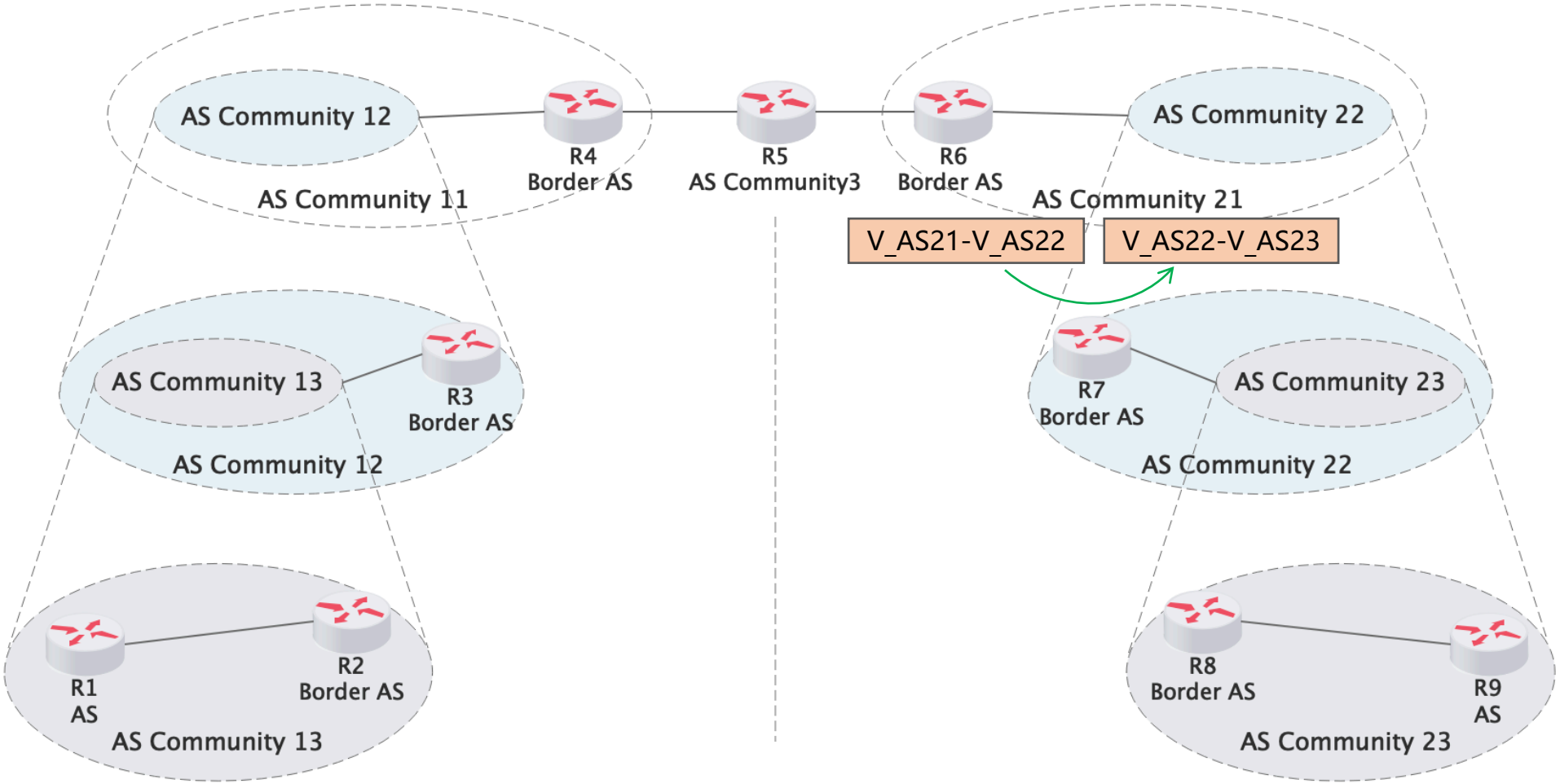
Example


V_AS12-V_AS11
V_AS11-V_AS21
AS Community 12
R4
Border AS
R5
AS Community3
R6
Border AS
AS Community 22
AS Community 11
AS Community 21
AS Community 13
R3
Border AS
AS Community 12
R7
Border AS
AS Community 23
AS Community 22
R1
AS
R2
Border AS
AS Community 13
R8
Border AS
R9
AS
AS Community 23
AS Community 1
AS Community 2

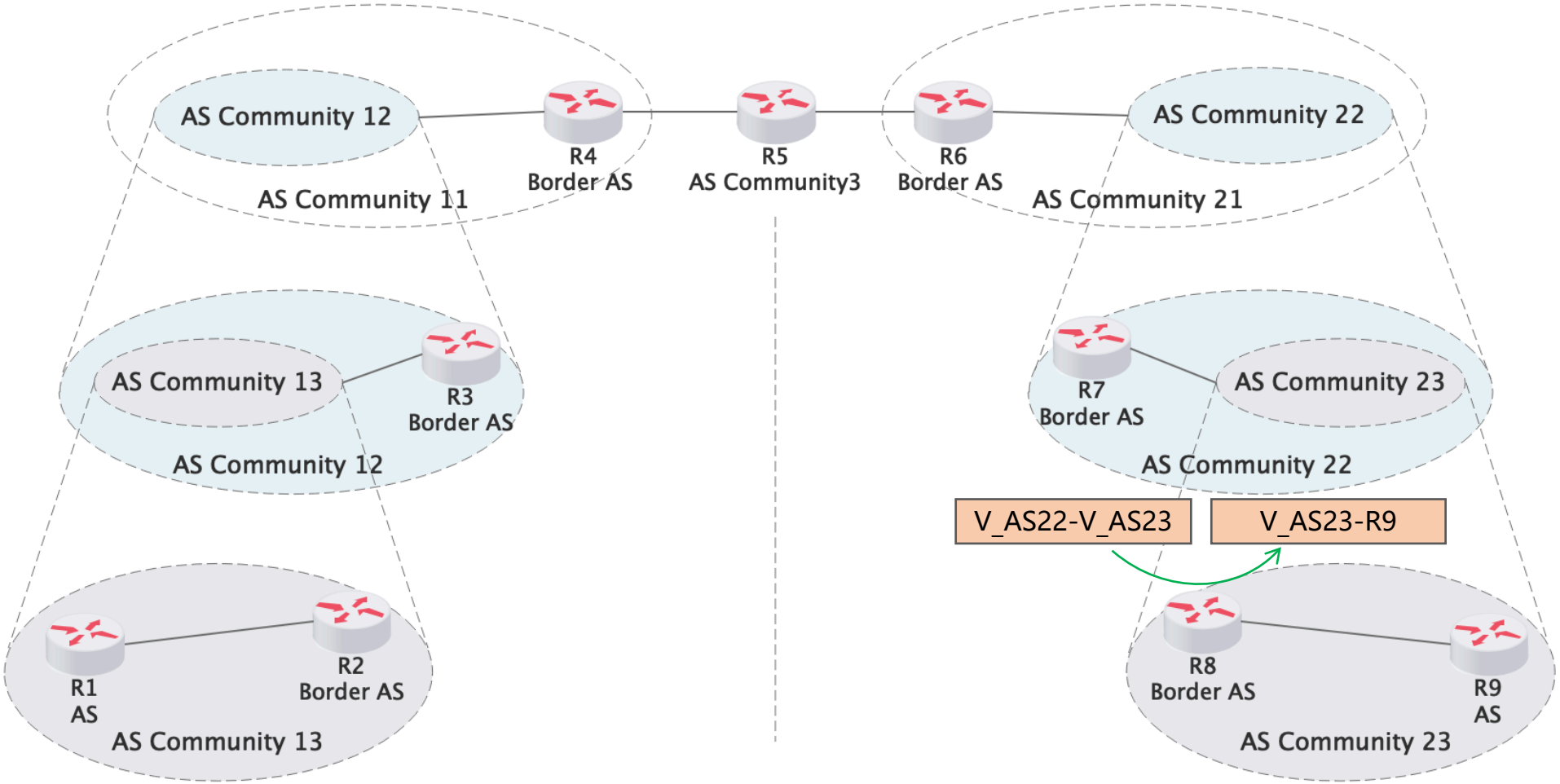
Example


V_AS11-V_AS21
V_AS21-V_AS22
AS Community 12
AS Community 11
R4
Border AS
R5
AS Community3
R6
Border AS
AS Community 22
AS Community 21
AS Community 13
R3
Border AS
AS Community 12
R7
Border AS
AS Community 23
AS Community 22
R1
AS
R2
Border AS
AS Community 13
R8
Border AS
R9
AS
AS Community 23
AS Community 1
AS Community 2

Example


AS Community 12
R4
Border AS
R5
AS Community3
R6
Border AS
AS Community 22
AS Community 11
AS Community 21
V_AS21-V_AS22
V_AS22-V_AS23
AS Community 13
R3
Border AS
AS Community 12
R7
Border AS
AS Community 23
AS Community 22
R1
AS
R2
Border AS
AS Community 13
R8
Border AS
R9
AS
AS Community 23
AS Community 1
AS Community 2

Example


AS Community 12
R4
Border AS
R5
AS Community3
R6
Border AS
AS Community 22
AS Community 11
AS Community 21
AS Community 13
R3
Border AS
AS Community 12
R7
Border AS
AS Community 23
AS Community 22
V_AS22-V_AS23
V_AS23-R9
R1
AS
R2
Border AS
AS Community 13
R8
Border AS
R9
AS
AS Community 23
AS Community 1
AS Community 2

Summary

1. ESAV is a cryptography-based SAV to guarantee consistent security benefits and provide scalability for different deployment scales and validation granularity.

2. ESAV uses hierarchy and tag replacement to reduce overhead and improve scalability compared to traditional cryptography-based schemes (such as SPM, IPSec).

Open Questions

1. For the limitation of IPv4 options header, how does ESAV provide validation and protection for both IPv4 and IPv6?

2. How to maintain and manage a hierarchical architecture for ESAV? Distributed or centralized?

3. What IPv6 option header should ESAV use, destination option header, hop-by-hop option header or routing option header?

Thanks!