

# Analysis of ROA-Only and TOA+ROA Methods for Prefixes Not Announced in BGP but Used for Source Addresses

.

**Kotikalapudi Sriram**

[Email: ksriram@nist.gov](mailto:ksriram@nist.gov)

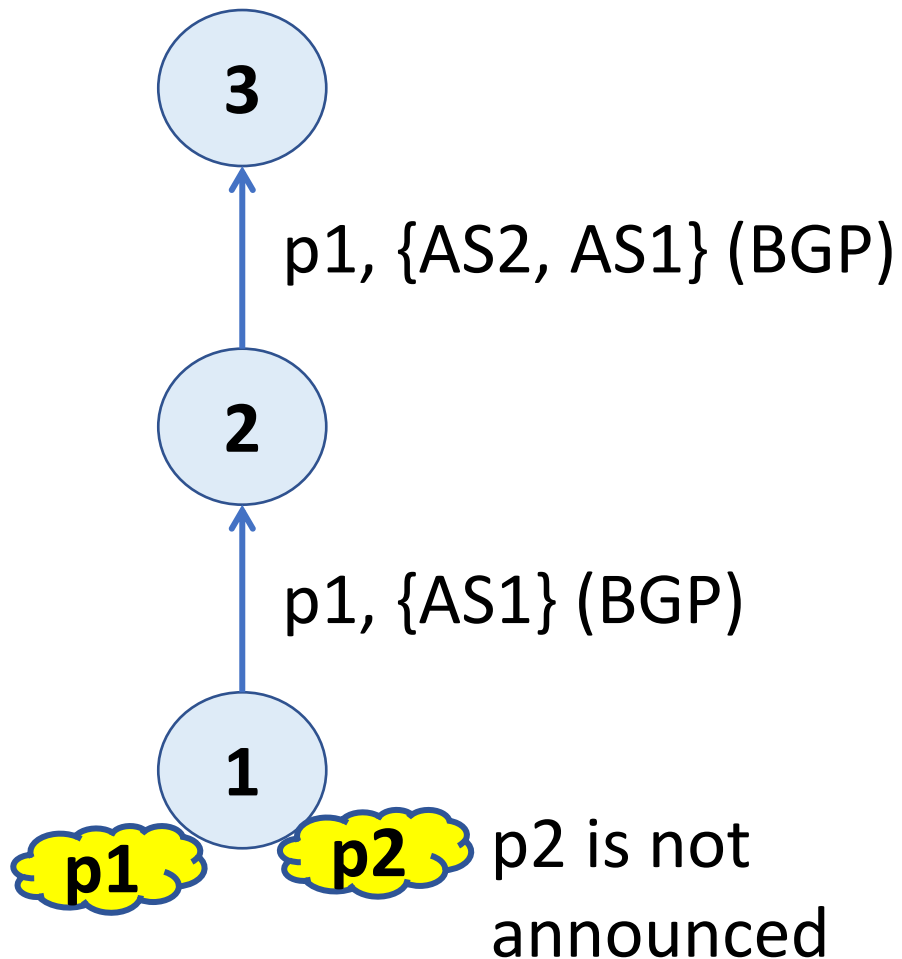
IETF SAVNET Meeting, IETF 125, March 2026

# Types of Prefixes Not-Routed But Sourced

Type of Prefix	Estimated Number
<b>Type 1:</b> Allocated; used internally in an AS; used for sourcing externally but not advertised	50 to 500 ?
<b>Type 2:</b> Some IANA allocated special-purpose prefixes with Source = True, Destination = False (prefix might be used from any AS)	10 to 20 ?
<b>Type 3:</b> CDN/DSR application – anycast prefixes (not routed from one AS but routed from other AS(es))	100 to 300 ?
<b>Total*</b>	<b>160 to 820 *</b>

\* 0.01 % to 0.06 % of all IPv4/IPv6 prefixes (1.3M) on Internet – very few

# How to Discover These Prefixes for SAV Purposes?



- p1 is a normal prefix; routed
- p2 is used for traffic sourcing only; not routed
- A way for the SAV method at AS 3 to discover p2 is to require p2 to register a **ROA: (p2, AS 1)**
- Another proposal: **TOA: (p2, AS 1)**

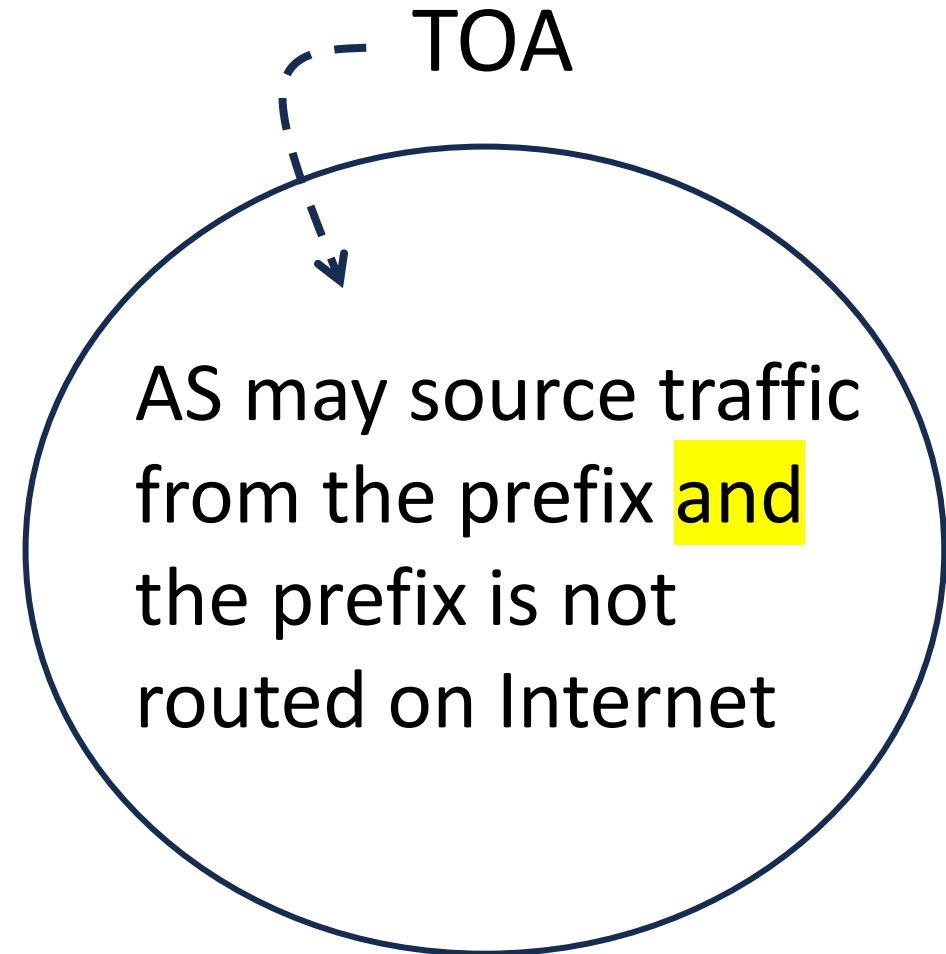
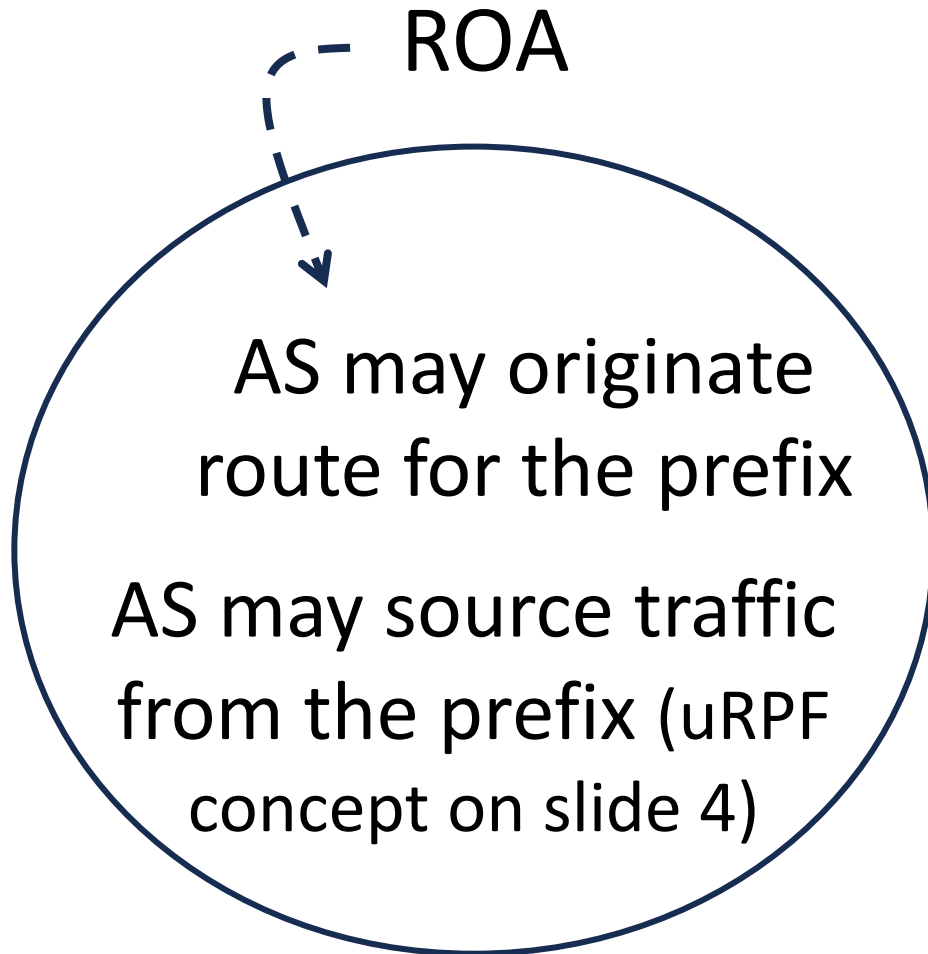
TOA = Traffic Origin Authorization

Note: Type 2 prefixes in slide 2 need a different solution; see slide 10

# ROA Semantic

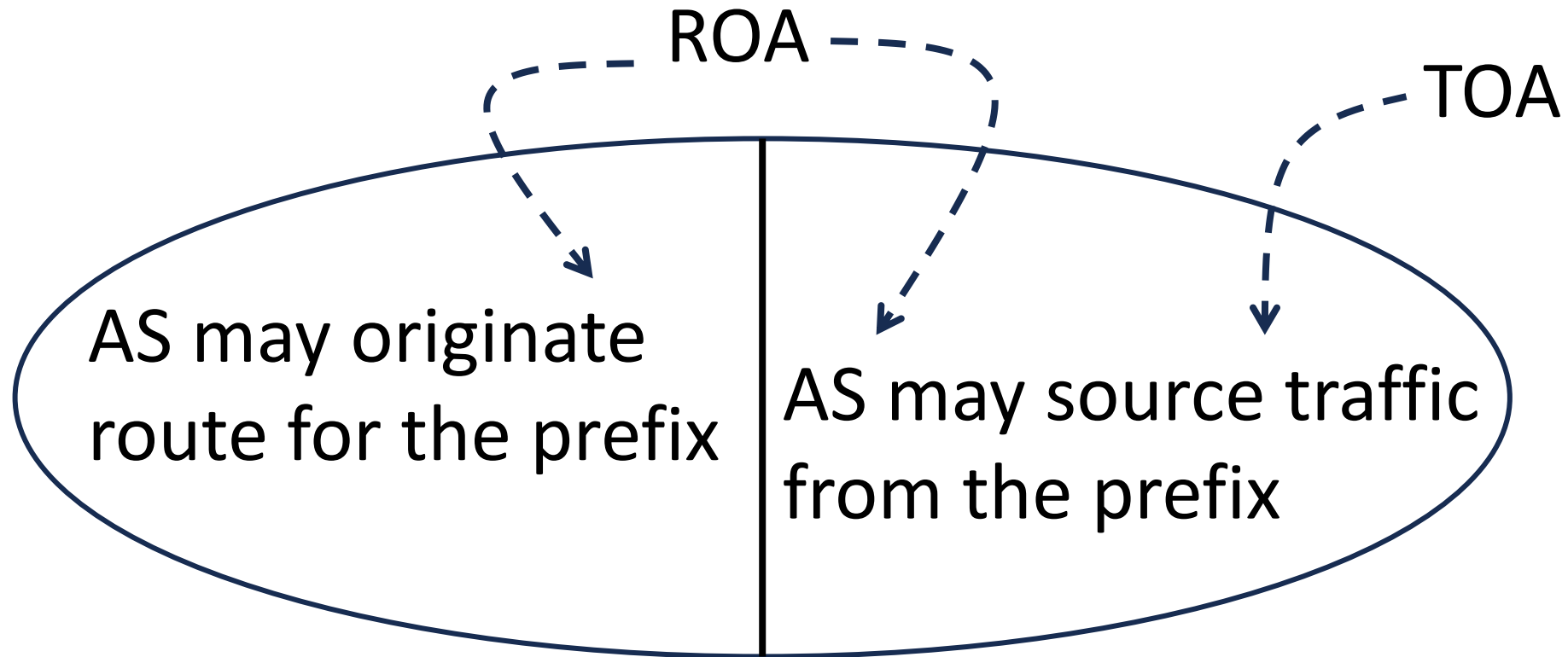
- Prefix in the ROA is authorized to be originated by the AS listed
- Use case: If ROA payload is {(P1 P2 P3 ...), AS 1}, then routes for P1, P2, P3, ... with origin AS 1 received on any interface at any AS are Valid (per ROV)
- uRPF principle is foundational to SAV [BCP 38] [BCP 84]
  - If a ROA allows a route for prefix P to be accepted on router interface X, then accept – on interface X – any incoming data traffic with source addresses in P provided that the AS in the ROA is in the CC associated with interface X.  
(extension of feasible path uRPF concept)

# Originally TOA Sematic was Meant to be Distinct from ROA



- This semantic of TOA would have required changes in RFC 6811 and the widely deployed ROV

# Current TOA Semantic Copies (Overlaps) a Part of ROA Semantic



- For a prefix used for sourcing but not routed on Internet, the TOA is required to be accompanied by an AS0 ROA

# Requirements for Any Proposed Method for Unrouted but Sourced Prefixes

## Functionality requirement for these prefixes:

- Method must discover the prefixes for SAV inclusion

## Uniform SIDROPS/IDR goals for all prefixes on the Internet:

- Prefix is protected from route hijacks (use ROA)
- Prefix route is protected from route leaks and some forms of AS\_PATH manipulations (use ASPA, OTC)

# Compare ROA-Only vs. TOA+ROA Methods – for Type 1 Prefixes (1 of 2)

ROA-Only  
method

- Register a ROA for the prefix with the AS
- Do not announce the prefix in BGP



These two solutions are equivalent in terms of functionality\*\*

TOA+ROA  
method\*

- Register a TOA for the prefix with the AS
- Register an ASO ROA for the prefix
- Do not announce the prefix in BGP

\* TOA+ROA method always needs support from ROA

\*\* TOA+ROA method inferior in dynamic TE cases (see slides 16 and 17)

## Compare ROA-Only vs. TOA+ROA Methods – for Type 1 Prefixes (2 of 2)

- Both methods protect against route hijacks
- TOA+ROA method prevents forged-origin prefix hijacks (indirectly, due to ASO ROA that is present) for the very few prefixes – a slim advantage; **ASPA solution already is a broad-based solution for that**

Type of Solution	Max # IPv4/IPv6 prefixes protected from forged-origin prefix hijacks
TOA (with ASO ROAs)	500 (0.04%)
ASPA (with normal ROAs)	All 1,300,000 (100%)

## Compare ROA-Only vs. TOA+ROA Methods – for Type 3 (CDN/DSR) Prefixes

- Both methods meet prefix discovery requirement for SAV
- Neither method protects against forged-origin prefix hijacks
  - But ASPA solution/implementations exist for this
- ROA-Only method has more advantages (see slides 11, 15)

## Solution for Type 2 (IANA Special) Prefixes

- Carefully understand and list these prefixes (only a few)
- This may be a simple universal list that network operator can augment to SAV allowlist on any interface
- Operator can ignore this if not applicable

## Key Trade-Offs: ROA-Only vs. TOA+ROA Method

- Both methods discover the prefixes (Type 1, Type 3)
- TE flipping of multi-origin prefixes: ROA-only method provides stability and operational simplicity; don't use TOA+ROA (slides 16, 17)
- TOA+ROA method: Involves new RPKI object; it takes time and effort in the IETF standardization process
  - To what advantage? Forged-origin prefix hijack prevention can be provided to a handful of Type 1 prefixes; a slim advantage that TOA+ROA method may claim
  - Note that the ASPA solution for prevention of forged-origin prefix hijacks applies to all 1.3M Internet prefixes
- TOA+ROA method: Unnecessary TOA proliferation possible

**Thank you!**

**Q & A**



# Backup slides

# Network Operator Feedback (NANOG Discussion)

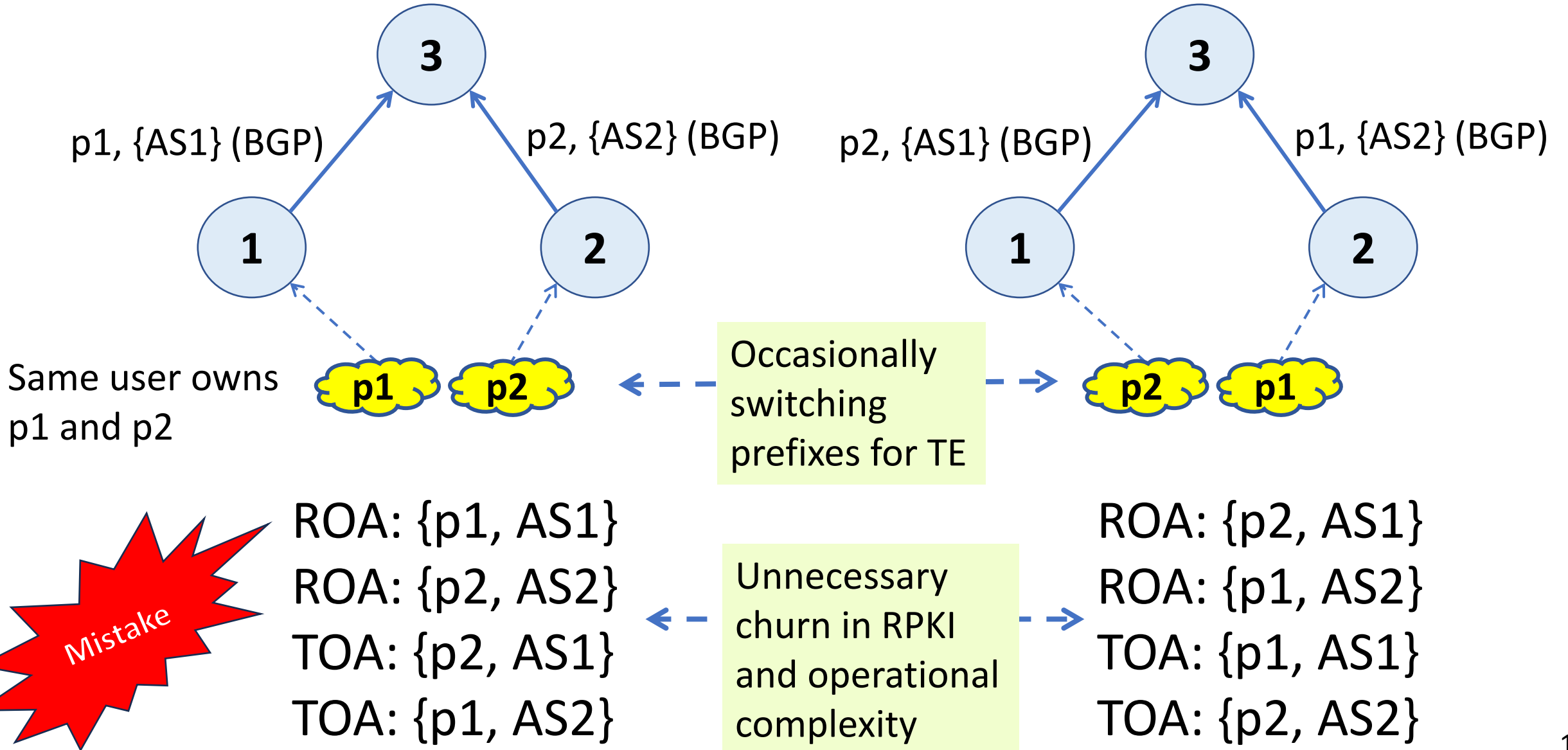
1. We don't mind advertising a prefix (normally not routed) to ensure passing SAV filtering
2. We don't mind creating a ROA so that the data traffic can pass SAV filtering

# TOA+ROA Method vs. ROA-Only Method: Pros-Cons Summary

	<b>TOA+ROA method</b>	<b>ROA-only method</b>
Needs new RPKI object	Yes	No
Complexity (user learning curve)	Increased	Not increased
User confusion (should I create ROA, TOA, or both (to be safe)?)	Unnecessary proliferation of TOAs	Not a concern
SAV/ROV functions in special scenario (multi-homed prefixes and TE flipping)	Operator must learn not to use TOA+ROA (slide 16)	Stable solution (slide 17)
Operations complexity	Added complexity	No new issues
RPKI abuse (malicious RPKI objects)	TOA adds another dimension for this	No new avenues added

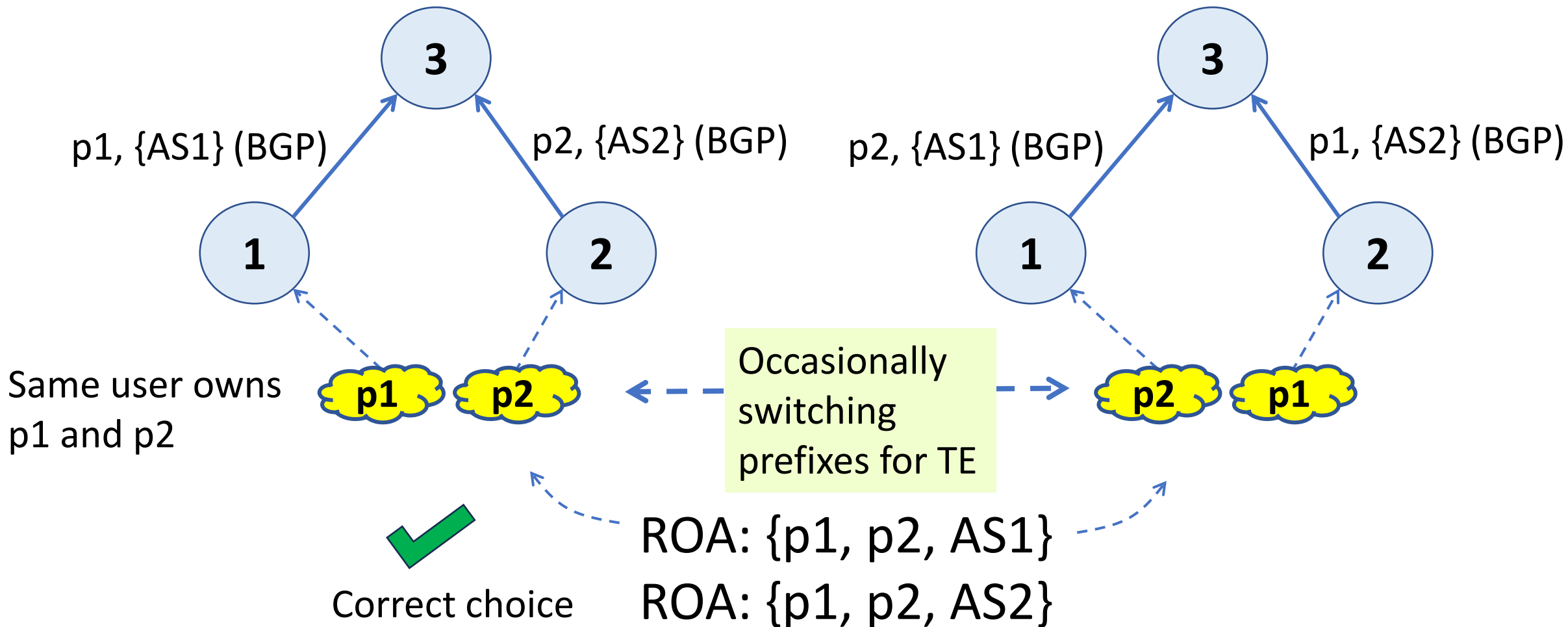
# Scenario when Prefixes May Flip for TE or Other Purposes

## TOA+ROA Method Causes Operational Complexity



# Scenario when Prefixes May Flip for TE or Other Purposes

## ROA-Only Method Facilitates Operational Simplicity



No change in the ROAs required;  
operational simplicity and no churn in the RPKI

# Potential Confusion: Unnecessary TOA Proliferation

- With the TOA+ROA method, many prefix owners who have a ROA, may unnecessarily create a TOA also to play totally safe
- **This leads to Unnecessary TOA Proliferation**

## Issue for ROA Aware but TOA Agnostic ASes

- Stub ASes are 84% of all ASes
- They create ROAs for prefixes originated
- With TOA+ROA method, TOA does not apply to them (for non-BGP customer interfaces)
- They monitor ROAs to detect foul ROAs with their AS
- They are TOA agnostic
- Attacker creates many TOAs including their AS
- The foul TOAs go undetected; used to evade SAV and conduct DoS attacks from elsewhere in the CC