

rov_enforcement_score_cp.py — ROV Enforcement & Hijacking Resistance Score

(Real-time control-plane inference of how easily a forged BGP announcement originating from an ASN would propagate)

1. Purpose & Role in the Platform

`rov_enforcement_score_cp.py` measures how safe or unsafe an ASN is as the origin of a hijacked/forged BGP announcement.

The metric answers the attacker-centric, operational-security question:

“If an attacker originates a forged/hijacked announcement from this ASN, how likely is that announcement to propagate across the Internet — and how often would it get filtered?”

The measurement is entirely empirical, based only on:

- **live BGP UPDATES** from RIPE RIS Live
- **real RPKI validation** from Routinator

No assumptions.

No policy declarations.

Only direct observation of how the ASN behaves when exposed to RPKI-Invalid announcements.

This matters because:

- **ASNs that *forward* RPKI-Invalid routes are extremely easy to abuse as attack-origins.**
- **ASNs that *filter* Invalids make attack attempts fail early, preventing propagation.**

- The more consistently an ASN filters Invalid announcements, the lower the chance that a forged origin inside that ASN succeeds in spreading globally.

The metric directly feeds into:

- **Vulnerability scoring (source-side security)**
 - **hijack propagation modeling**
 - **ML-based ASN risk classification**
 - **origin-trustworthiness dashboards**
-

2. High-Level Concept (Attacker-Origin View)

The script continuously observes how each ASN behaves **as a transit/origin environment** when confronted with:

- RPKI-Invalid announcements
- RPKI-Valid announcements for the same prefix/origin
- Differences in propagation across vantage points

Using these real-world contradictions, the script determines:

- **LEAKED** → ASN forwarded an Invalid announcement
→ *Meaning an attacker inside this ASN could easily inject a forged origin.*
- **BLOCKED** → ASN forwarded only the Valid version while others propagated the Invalid
→ *Meaning the ASN filters Invalids and is resistant to being abused as a attack-source.*

The result is a **probabilistic enforcement score** representing how safe the ASN is as a potential origin of a hijacked announcement.

3. Operational Flow (Simplified and Clear)

1. Connect to **RIPE RIS Live** (WebSocket).
 2. Parse every BGP UPDATE (prefix, origin, vantage, AS_PATH).
 3. Clean AS_PATH for correct inference.
 4. Validate (prefix, origin) using **Routinator**.
 5. Compare vantage points:
 - some see the origin as **Invalid**
 - others see the same origin as **Valid**
 6. Infer enforcement behavior from how each ASN handles Invalid vs Valid propagation.
 7. Compute:
 - ROV enforcement score (0..1)
 - 95% credible interval
 - confidence score
 - classification
 8. Store results in SQLite in real time.
-

4. How the Script Evaluates Attack-Source Security

The central idea:

If an ASN frequently forwards RPKI-Invalid routes, then a forged origin injected *from inside that ASN* is highly likely to propagate — making the ASN

unsafe.

If an ASN consistently refuses to forward Invalids, attack attempts originating inside it will die immediately.

The metric measures exactly that.

5. Detecting LEAKED Events ("Attack-Friendly" Behavior)

A **LEAKED** event is recorded when:

- At least one vantage point sees the announcement as **RPKI-Invalid**, and
- The cleaned AS_PATH includes ASN X.

Interpretation (origin-side perspective):

- ASN X forwarded an Invalid announcement.
- If an attacker hijacked an origin inside ASN X, the forged announcement would likely continue through ASN X and propagate.
- ASN X is therefore **permissive** and **high-risk** as a hijack-source environment.

This is **direct, undeniable evidence** that hijacked origins are unlikely to be stopped inside that ASN.

6. Detecting BLOCKED Events (Attack-Resistant Behavior)

A **BLOCKED** event is recorded when:

1. Some vantage points receive the announcement as **Invalid**.
2. Other vantage points receive the **Valid** version of the same (**prefix**, **origin**).

3. ASN X appears on valid-only paths.
4. ASN X never appears on invalid paths.

Interpretation:

- ASN X filters Invalid updates.
- A forged/or hijacked origin inside ASN X would likely be dropped immediately.
- ASN X is therefore **safe**, **strict**, and **hard to abuse** as a hijack-source.

BLOCKED events provide **strong empirical evidence** of internal ROV enforcement.

7. Defining Vulnerability Opportunities

For each ASN:

`opportunities = leaked_events + blocked_events`

An ASN needs at least one such opportunity to be scored.

Why?

Because scoring vulnerability requires actual empirical exposure to Invalid-vs-Valid contradictions.

No opportunities → no real data → ASN cannot be evaluated.

8. Computing the Enforcement / Vulnerability Score

8.1 Bayesian Probability of Enforcement

`score = (1 + blocked) / (2 + blocked + leaked)`

What this means:

- HIGH score → ASN filters Invalids → **hard to hijack from inside**
- LOW score → ASN forwards Invalids → **easy to hijack from inside**

Why this is the correct model:

- Uses a neutral prior (Beta 1,1)
 - Stabilizes early measurements
 - Converges rapidly with more evidence
 - Produces realistic probabilities without overfitting
-

8.2 95% Credible Interval

Shows how certain we are about the enforcement/vulnerability probability.

- Wide interval → not enough data
- Narrow interval → high certainty about hijack resistance

This is essential when comparing ASNs.

8.3 Confidence Score

```
confidence = min(1, opps/20) * min(1, vantage_count/30)
```

Confidence increases when:

- the ASN is exposed to many Invalid-vs-Valid situations
- multiple independent vantage points observe consistent behavior

High confidence means the vulnerable evaluation is reliable.

8.4 ASN Classification

Categories reflect vulnerability resistance:

- **strong_rov** → extremely hard to hijack from inside
- **probable_rov** → strong resistance
- **possible_rov** → partial resistance
- **no_validation / none** → easy to hijack from inside or unobserved

These are operational categories — no assumptions, only observed outcomes.

9. Why This Approach Accurately Measures Vulnerability

✓ Uses only real control-plane behavior

No modeling. No guesses.

✓ LEAKED = direct evidence of “attack-friendly” behavior

An Invalid route cannot reach a vantage unless every ASN on the path forwards it.

✓ BLOCKED = direct evidence of attack-resistance

A vantage seeing only Valid while others see Invalid implies filtering.

✓ Bayesian scoring smooths noise

Better stability than raw ratios.

✓ Confidence computation prevents premature conclusions

Great for operators.

✓ **RIPE RIS + Routinator = trusted, real-world sources**

Measurements match real routing infrastructure behavior.

✓ **The metric describes *origin security***

Precisely what matters for hijack propagation.

10. Interpretation (From a Hijacked-Origin Perspective)

High enforcement score

- Strong ROV
- Forged origins from this ASN almost certainly fail
- Vulnerability: **very low**
- Hijack propagation: **unlikely**

Low enforcement score

- ASN forwards Invalids
 - Forged announcements from this ASN will likely propagate
 - Vulnerability: **high**
 - Hijack propagation: **very likely**
-

11. Limitations (Honest and Accurate)

- Depends on RIS vantage diversity
- ASN must appear in Invalid-vs-Valid contradictions
- Cannot observe internal filters not exposed in control-plane
- Routinator availability affects data

- Short observation windows reduce evidence
-

12. Why These Sources Were Chosen

RIPE RIS Live

- Real control-plane behavior
- Wide global visibility
- Captures real propagation differences essential for vulnerability evaluation

Routinator

- Cryptographically authoritative RPKI validation
- Matches real-world ROV behavior inside ISPs
- Deterministic and reliable

Together they form the **most accurate possible foundation for measuring how safe or dangerous an ASN is as a attack-origin environment.**