

A decorative graphic on the left side of the slide consisting of two overlapping parallelograms. The front one is blue and the back one is a light greenish-blue. They are positioned diagonally, with the blue one partially covering the green one.

Seeing Beyond the Signatures:

Federated Multisig Observability



What are Multisigs?

- A popular saying by Andreas Antonopolous went by “Your keys, your bitcoin. Not your keys, not your bitcoin.”
- The purpose of multi-sig is to spread out the attack surface by requiring things like visiting multiple locations to sign a transaction, possibly multiple states, and using multiple hardware vendors from different countries.



The Problem: Invisible Complexity

- When 7 signatures are needed from 12 validators across 4 continents and only 3 show up...
- How do you even begin to debug that?

What You Actually Need to Know:

- Which validators are online?
- Are they receiving the transaction?
- Is there a network problem?
- Should we be worried about security?



The Multisig Mirage

Alice + Bob + Charlie =  Transaction Signed




What's Actually Happening:

- Transaction Initiation
- Hash generation
- Signature Collection Process
- Signature Aggregation
- Verification Process
- Execution Phase



The Invisible Infrastructure

What Teams Currently Monitor (10%):

-  Transaction success/failure
-  Basic signature counts
-  Wallet balances

What Remains Hidden (90%):

- **Signature Collection Patterns:** Who signs when?
- **Network Topology Health:** Are validators reachable?
- **Cryptographic Performance:** Proof generation bottlenecks
- **Byzantine Behavior Detection:** Malicious participants
- **Threshold Dynamics:** Why did 5/7 become 3/7?



The Anatomy of a Federated Signature

The 12 Step Dance Nobody Sees:

1. **Proposal Broadcast** → All validators notified
2. **Eligibility Check**: Validate participation rights
3. **Nonce Generation** → Prevent replay attacks
4. **Partial Signature**: Each validator contributes
5. **Aggregation Protocol** → Combine partial signatures
6. **Verification Round** → Cross-validate results
7. **Threshold Assessment**: Sufficient signatures?
8. **Byzantine Detection** → Identify bad actors
9. **Commitment Phase** → Final signature assembly
10. **Network Consensus** → Agreement on final state
11. **State Transition** → Update global state
12. **Finalization** → Transaction executed

Each step has 3-5 potential failure modes



Why Observability matters in federated networks

- We have the issue of fragmented systems across operators
- Operational silence, which equals slow recovery time
- Invincible failures caused by timeouts or slow signers, which result in a stop in block production due to chain halts



The Vision: True Federated Observability

In federated multisig systems, true visibility is elusive. Each operator maintains independence while contributing to a shared observability layer.

The goal is to ensure real-time insight, identify bottlenecks, and increase network stability without sacrificing decentralization.



Meet Grafana Alloy

As the documentation suggests, “Grafana Alloy combines the strengths of the leading collectors into one place. Whether observing applications, infrastructure, or both, Grafana Alloy can collect, process, and export telemetry signals to scale and future-proof your observability approach.”



Utilising Grafana Alloy

-> **Centralized Logs with Loki**

Alloy scrapes logs locally and sends them to Loki, enabling a real-time view into node behaviours, errors, and events without compromising local setup.

-> **Metrics via Remote Write**

Performance and resource usage are collected and sent to centralised Prometheus using Alloy's native support for `remote_write`.

-> **Traces with OpenTelemetry**

Alloy integrates tracing data across operator nodes using OpenTelemetry, enabling end-to-end insight into signing events and delays.



Deployment Architecture

-> **Per Operator Configs**

Each operator runs Alloy with its own configuration tuned to its infrastructure—whether Kubernetes, bare metal, or cloud-native.

-> **Centralized Observability**

Despite localised configs, all data flows to shared endpoints—Loki for logs, Prometheus for metrics, and Tempo for tracing.

-> **Security & Autonomy**

Operators retain full control over their environments. Secure endpoints and tokenised access ensure data protection without central control.



What can you see and do?

-> **Live Operational Insight**

From knowing who the slowest signer is to detecting error bursts in logs, Alloy exposes the internal life of each participant.

End-to-end traces show how requests propagate across signers, highlighting performance, delay, and failure points in real time.



The Impact: From Darkness to Light

-> Before: The 90% Problem

- Blind spots in signature collection
- Hours of debugging failed transactions
- Network issues discovered too late
- Byzantine behavior goes undetected

-> After: Full Visibility

- Real-time insights across all validators
- Proactive issue detection and resolution
- Complete transaction lifecycle tracing
- Enhanced security through behavioral monitoring



Key Benefits Delivered

Operational Excellence

- **Faster Recovery:** Reduce MTTR from hours to minutes
- **Proactive Monitoring:** Catch issues before they impact users
- **Performance Optimization:** Identify and resolve bottlenecks

Security & Trust

- **Byzantine Detection:** Spot malicious behavior immediately
- **Audit Trail:** Complete visibility for compliance
- **Threshold Monitoring:** Understand signature dynamics

Federated Approach

- **Operator Autonomy:** Each validator maintains independence
- **Shared Insights:** Collective visibility benefits everyone
- **Scalable Architecture:** Grows with your network



Questions & Discussion

Let's Discuss:

- What are your biggest external validators' operational pain points?
- Which monitoring gaps impact your network most?
- How can you customize this approach for your infrastructure?



Thank You

Seeing Beyond the Signatures: Federated Multisig Observability

Making the invisible visible. Making the complex manageable. Making federated networks truly observable.