



whoami

Worked as web3 consultant

Worked in DevOps and IoT

Blog: odyslam.com/blog

Twitter: [@odysseas_eth](https://twitter.com/odysseas_eth)



Section 1

Nomad is **Not** a Bridge



Section 1

Nomad **has** a Bridge



Section 2

Nomad is an **optimistic protocol**



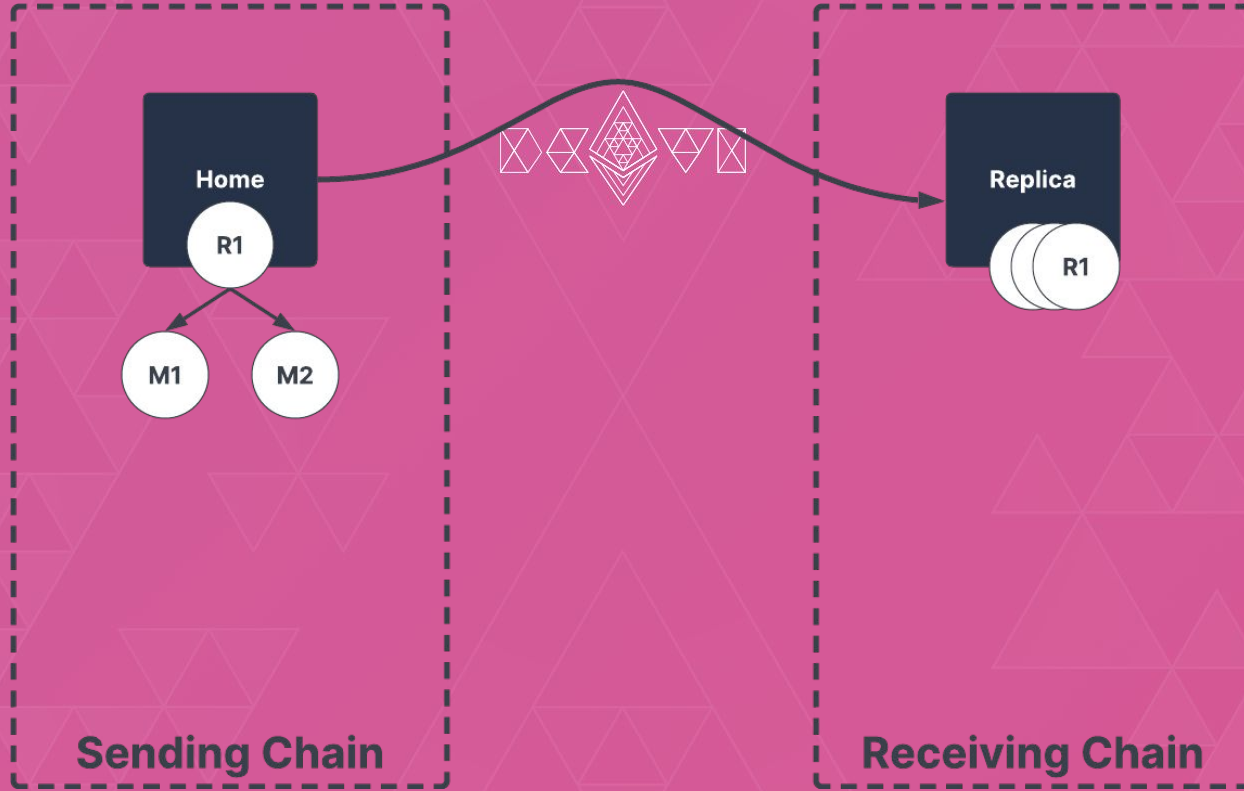
Section 2

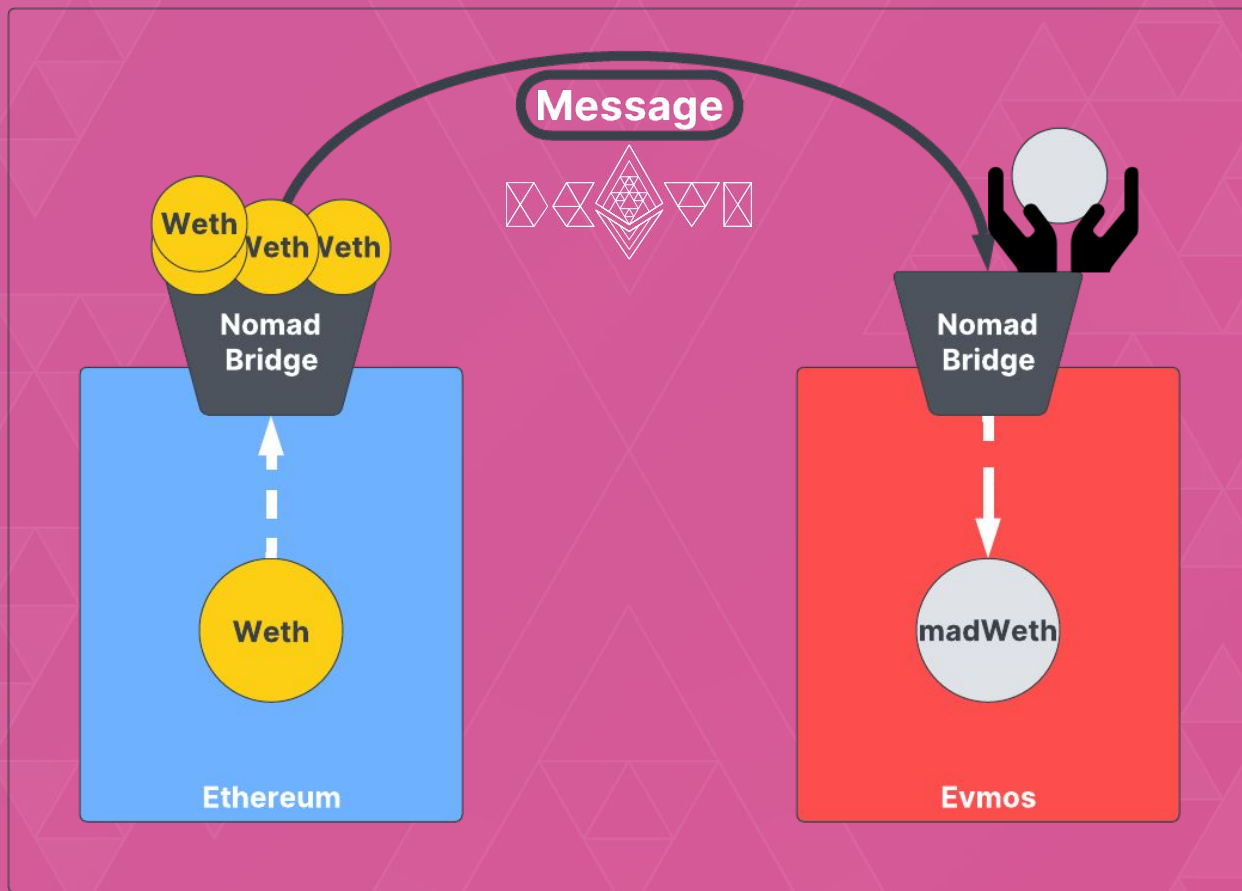
Nomad is an optimistic protocol for **interoperability**

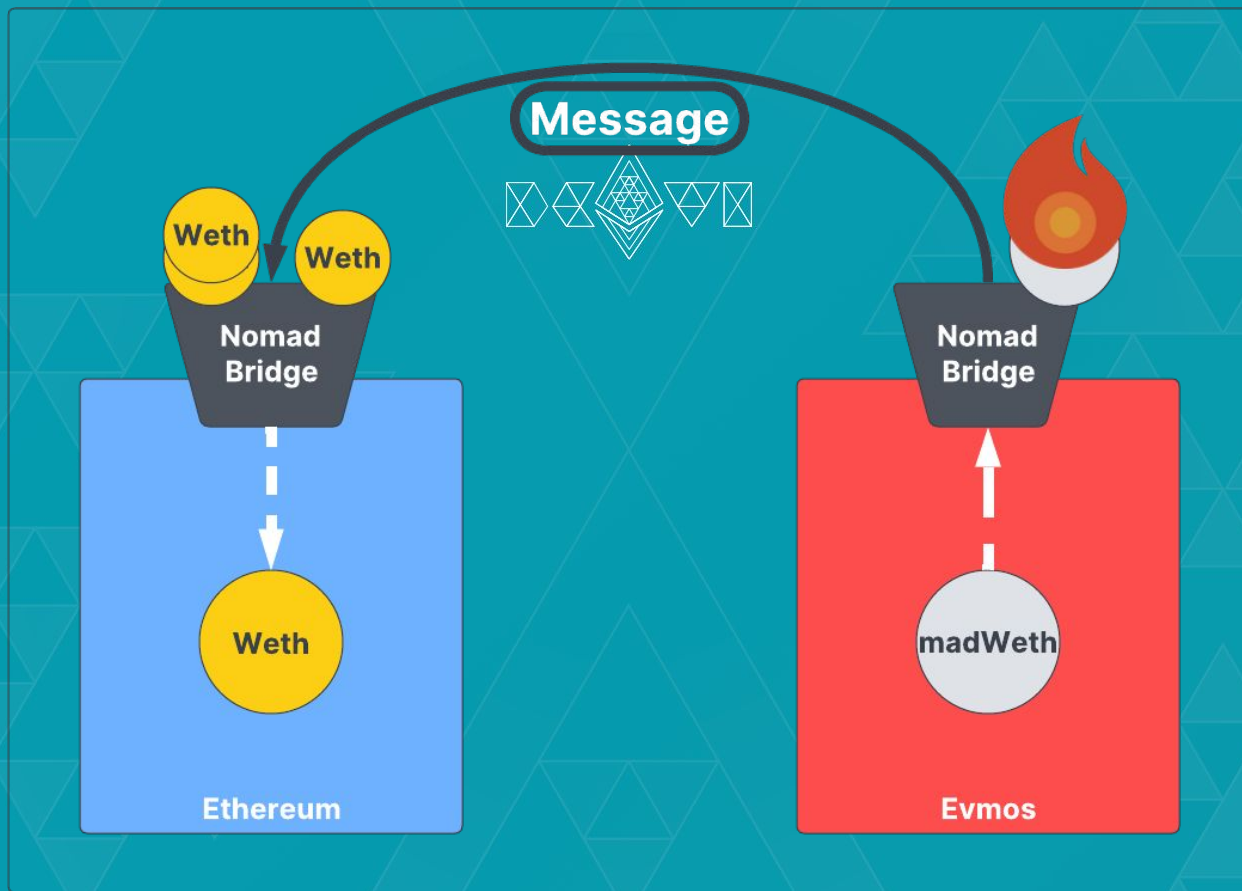


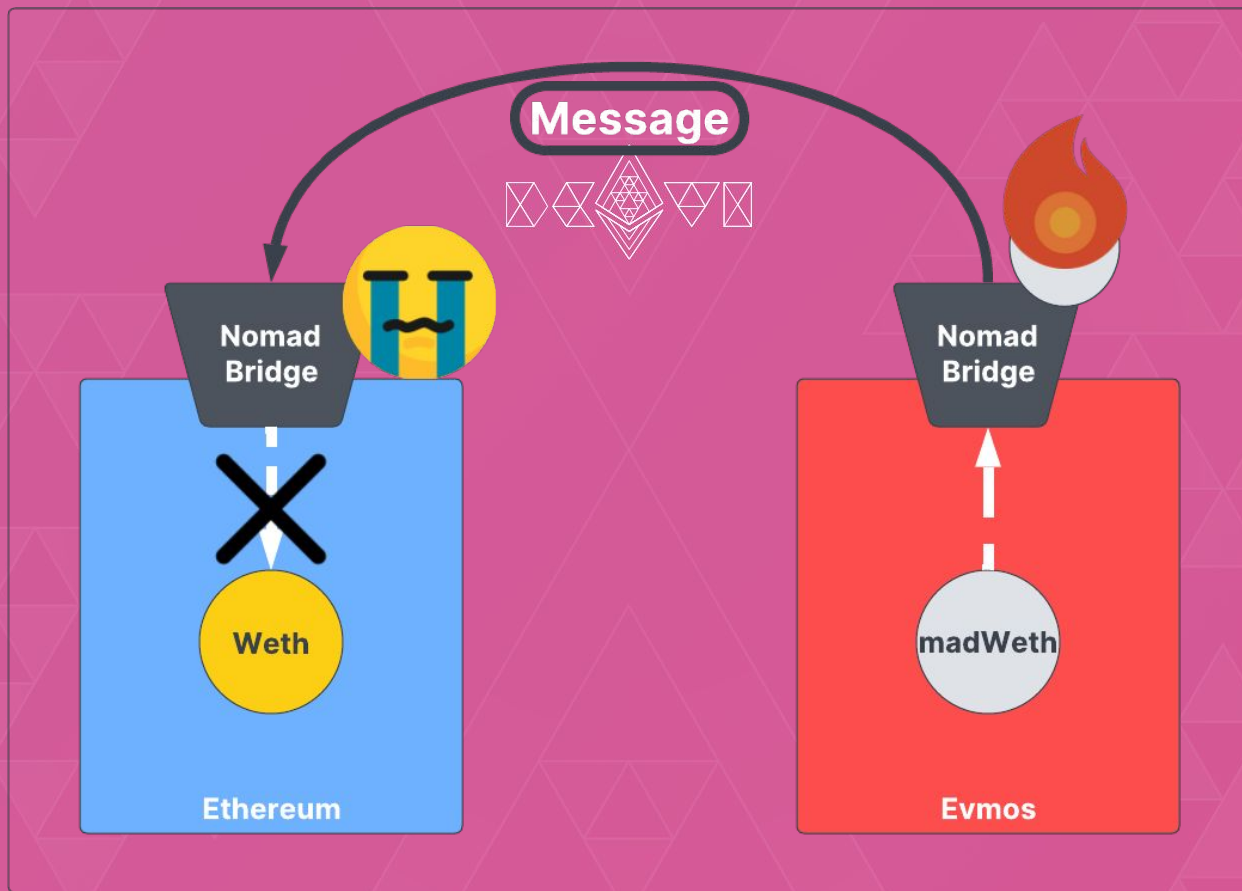
Section 2

Nomad is an optimistic protocol for **interoperability** that supports **arbitrary messages between domains**

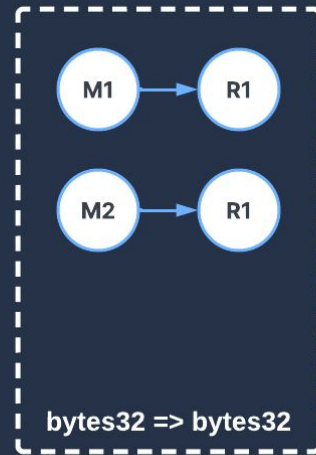
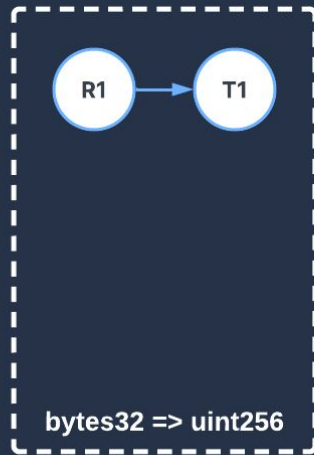




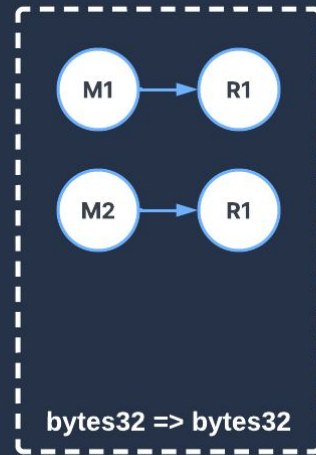
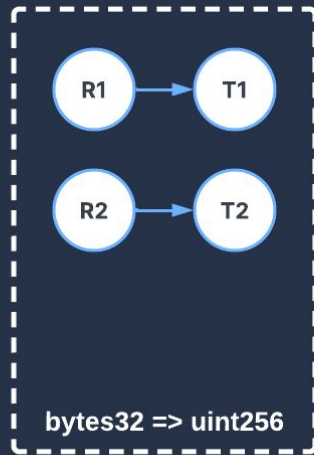




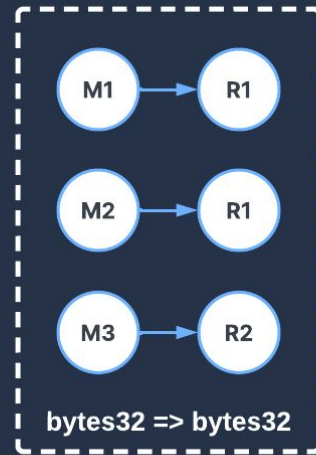
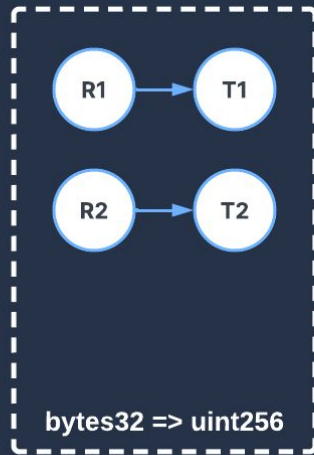
Replica



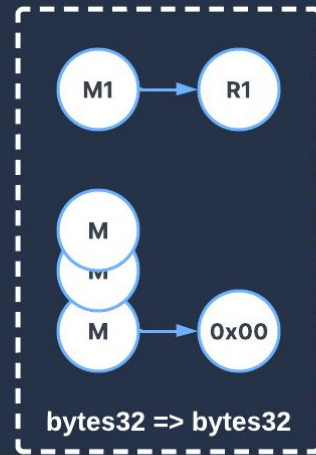
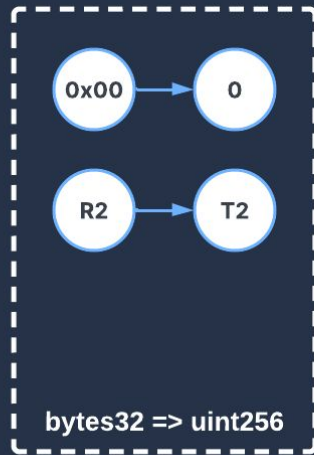
Replica



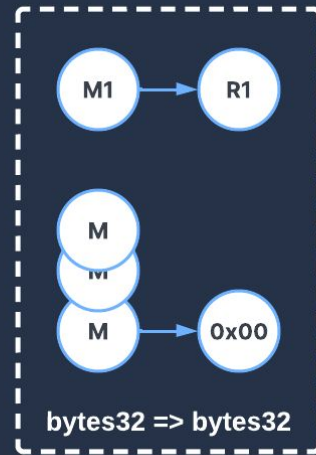
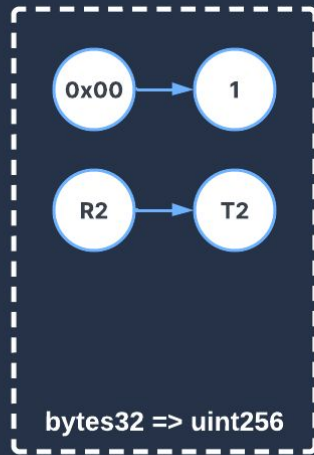
Replica



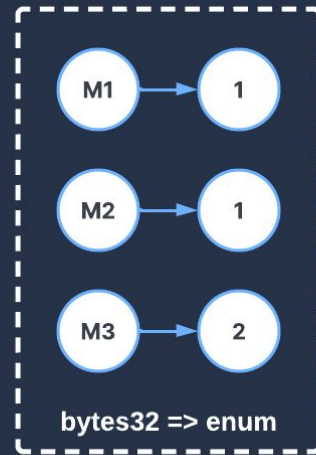
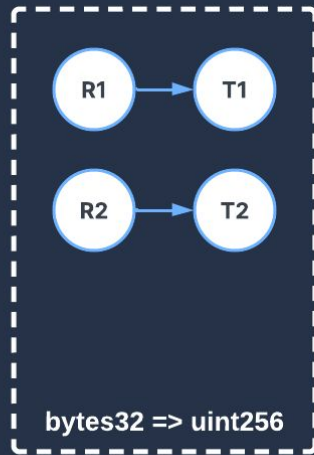
Replica



Replica



Replica



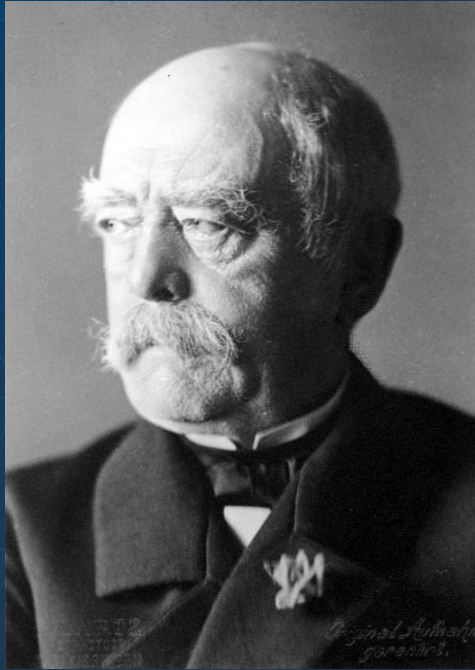
Restarting the Protocol is **Easy**

Restarting the Bridge is **Hard**

Principles:

1. Users should be able to collect funds as they are recovered
2. Recovered funds should be fairly distributed to users

Nomad Now



“Only a fool learns from his own mistakes. The wise man learns from the mistakes of others.”

— Otto von Bismarck



Test

Observe

Engage

Communicate



Test

Observe

Engage

Communicate

Test (🔧, 🔥)

Unit Tests

Property-based Tests

Integration Tests

Forking Tests

Invariant Tests

- + **Static analysis**
- + **Storage layout**

Unit-tests

Input = 5



Function



Output = 25

Property-based tests

Input = x



Function



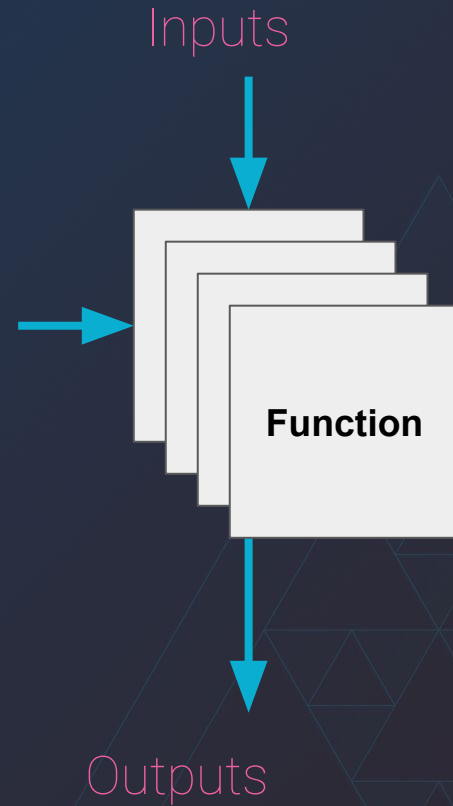
Output = $x * 5$

Integration tests



Forking tests

**Chain
state**



Invariant Tests

1. **Define Invariants**
2. **Test Invariants**

Example:

Every message that is received, must have been dispatched

Finally

Static Analysis Tools

Storage Layout Analysis

(Resource: [forge-inspect.sh](https://github.com/0x00sec/forge-inspect.sh))

Priority

Unit Tests

Property-based Tests

Forking Tests

Audits

Storage layout



Easy with Foundry

U up?

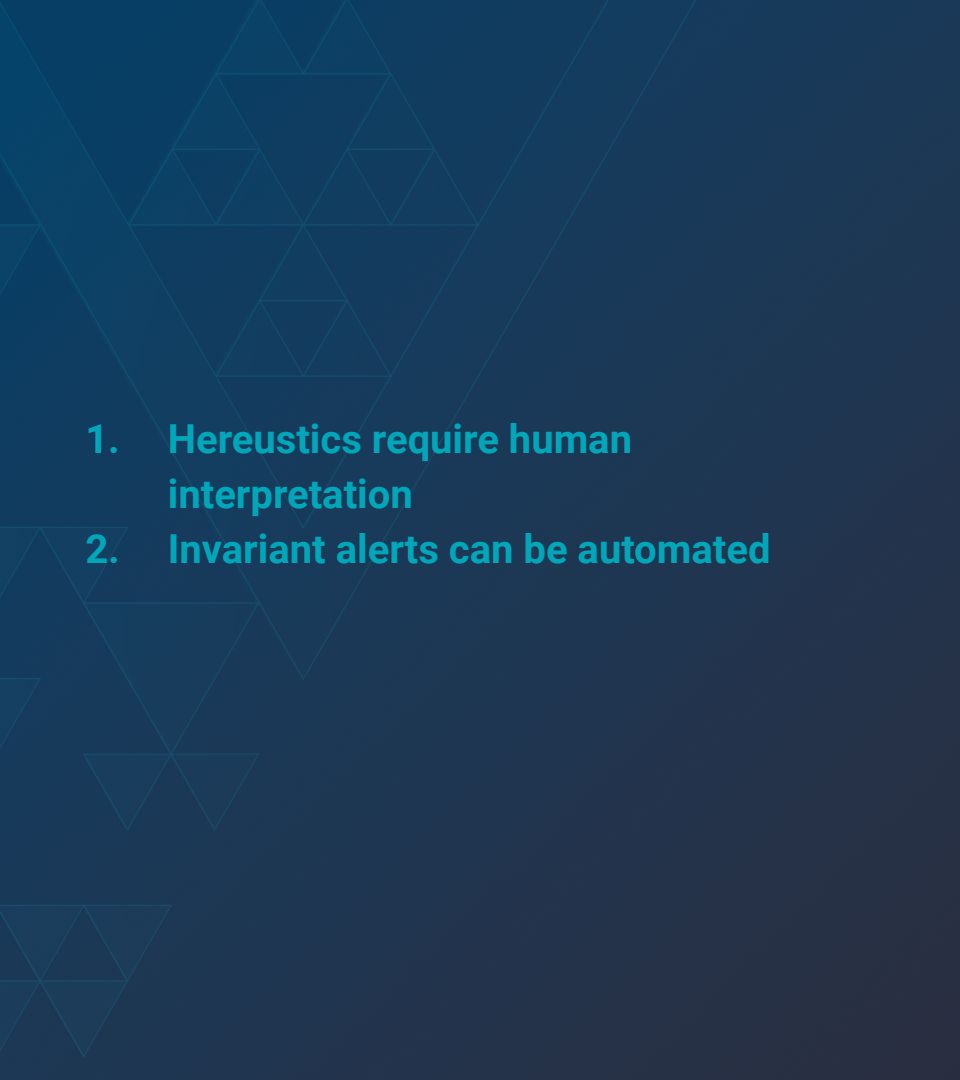
It's already too late

Observe 

Web 2 is mature, like Google SRE handbook

1. Start with Business Objectives
2. Define **Actionable Alerts**
3. Define Playbook for **every Alert**
4. Beware of Alert Fatigue 🤖

Web2 tactics

- 
1. **Hereustics require human interpretation**
 2. **Invariant alerts can be automated**

Hereustic and Invariant Alerts

Engage 

We do not rise to the level of our expectations. We fall to the level of our training

- Archilochus



Focus

Explicit ownership

Explicit outcomes

Gameday. Gameday. Gameday

**Read [The Anatomy of a Good
Emergency Procedure](#)**

Communicate 

The background of the slide features a dark blue gradient with a subtle, repeating pattern of light blue triangles of various sizes, some pointing up and some pointing down, creating a geometric texture.

Talk with users



no



Talk with legal



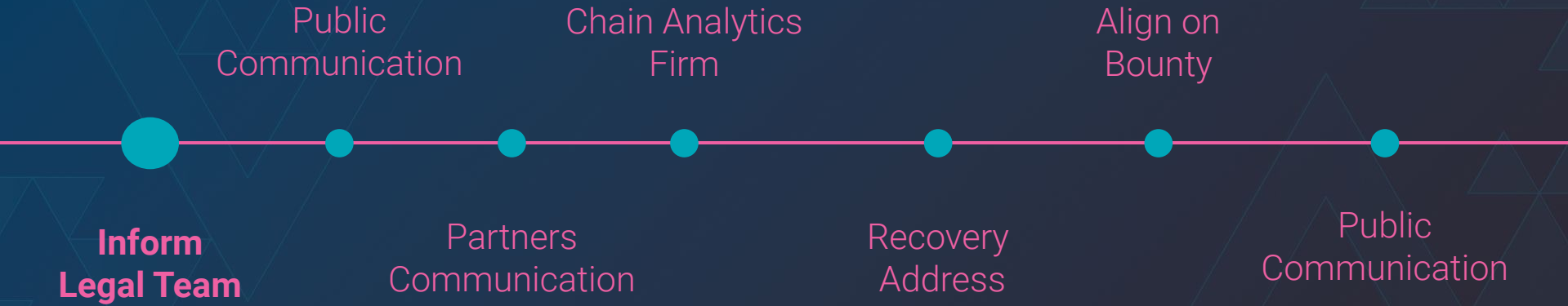
Honesty. No sugar coating

Pre-approved messaging

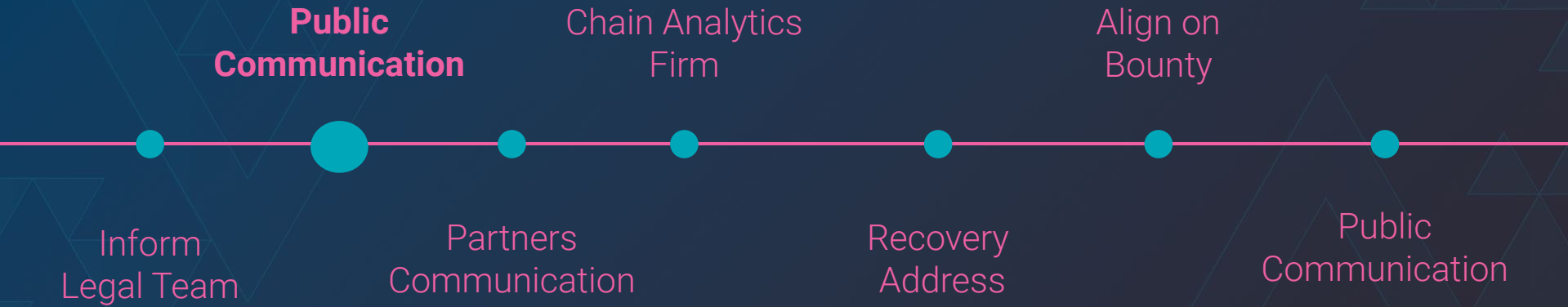
- **What are you doing now**
- **What do you plan doing next**

What the users want

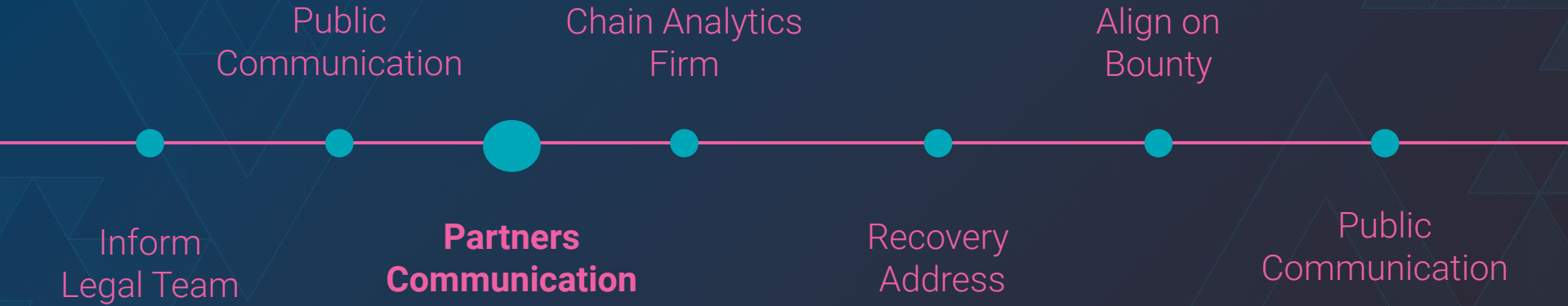
First days after the incident



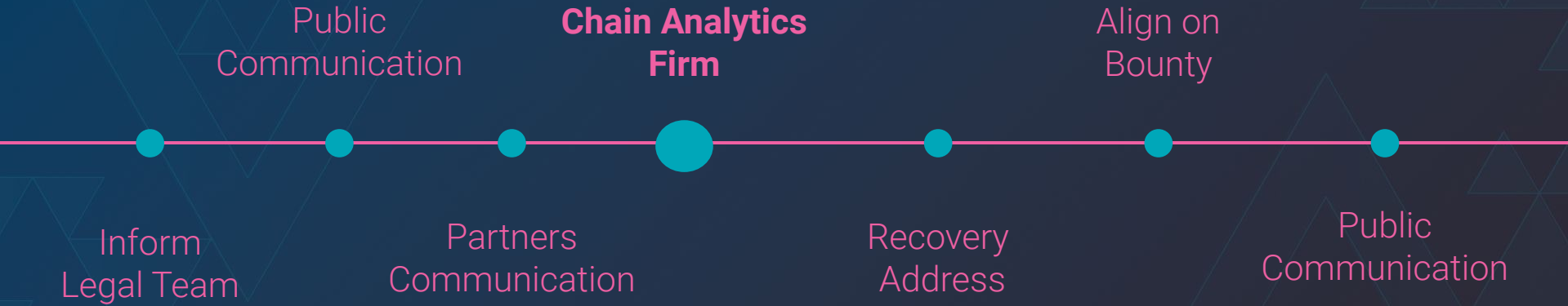
First days after the incident



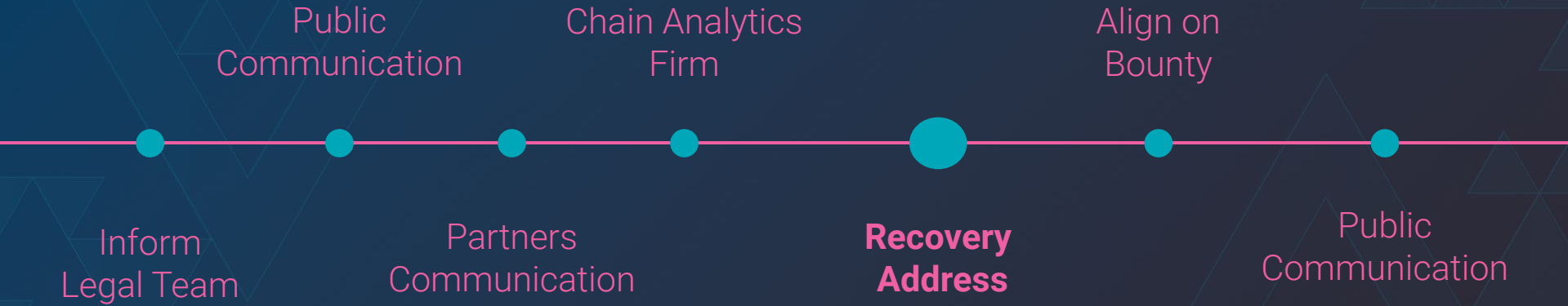
First days after the incident



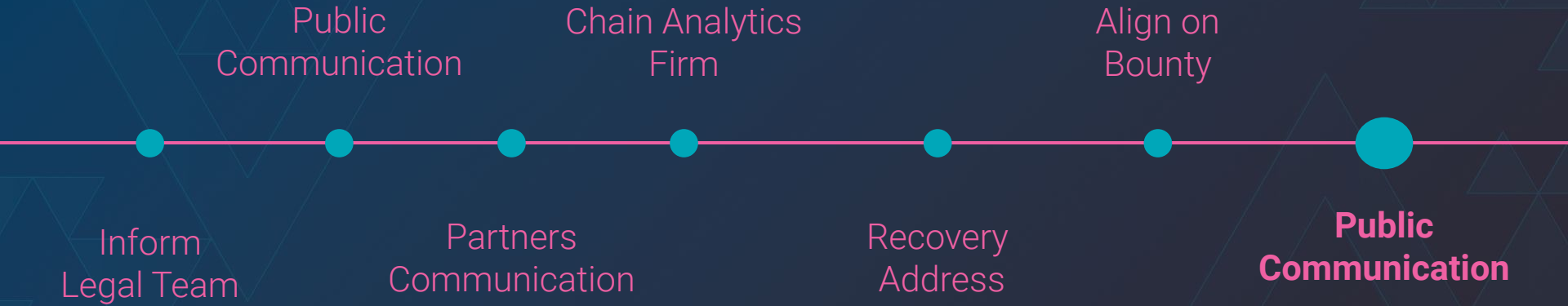
First days after the incident



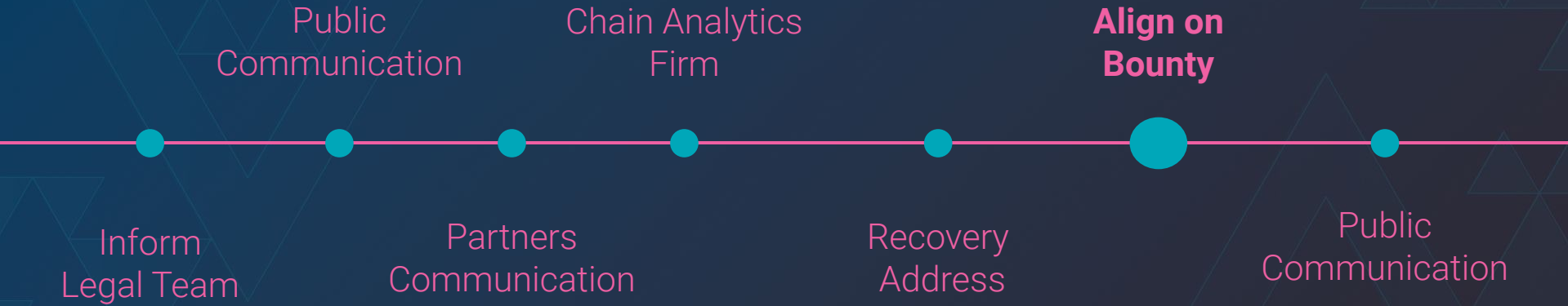
First days after the incident



First days after the incident



First days after the incident



Maintenance

Public
Communication

**Chain Analytics
Firm**

Align on
Bounty

**Inform
Legal Team**

**Partners
Communication**

Recovery
Address

**Public
Communication**



Thank you!

Odysseas.eth

Protocol Engineer, Nomad

odysseas@nomad.xyz



@odysseas_eth