LEVO

Code to production API Security platform

eBPF - A game changer for discovering data flows

FOUNDER
& CEO

BUCHI REDDY B

APPDYNAMICS • CISCO • TRACEABLE. • D E Shaw & Co

Agenda

The need for realtime API & Data Observability

Why current approaches fail

How our innovation bridges the gap
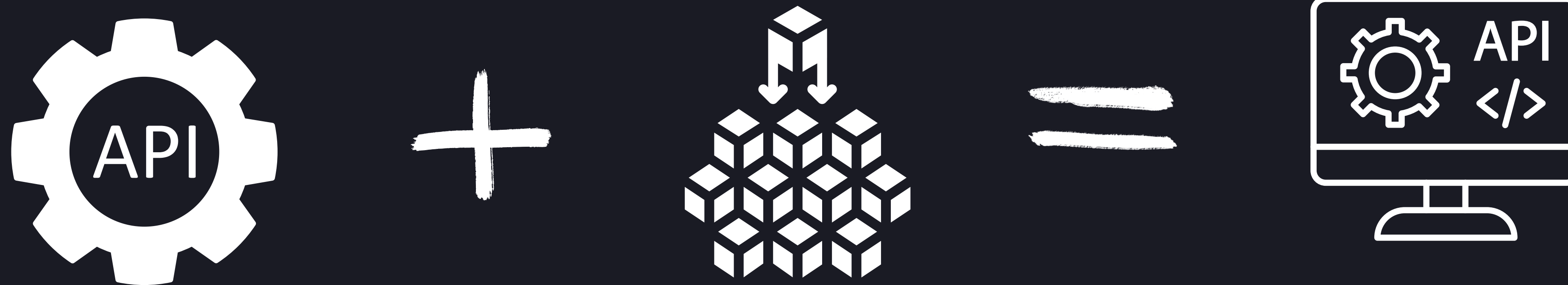
How we leverage eBPF

User Data Mapping with eBPF

Automatic Discovery, Documentation & Testing

Q&A

APIs & microservices have exploded

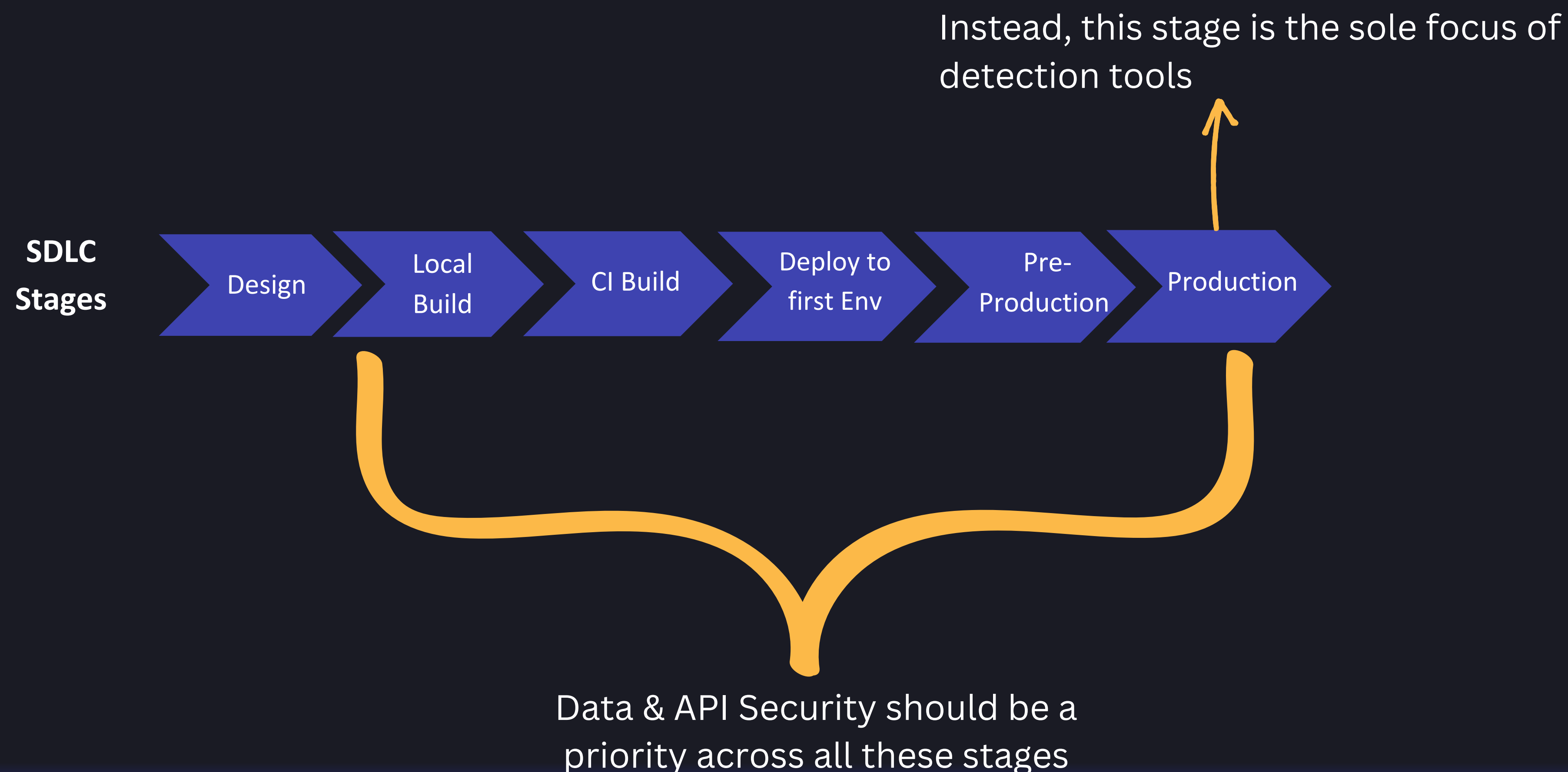Great for business innovation but threaten data security

API Sprawl is a top concern
for 48% of enterprises

Yet there is no tool to gain visibility into APIs and data flows at scale

Thus vulnerabilities leak into production

Legacy tools aren't designed to prevent API- induced data breaches

Instead, this stage is the sole focus of detection tools

SDLC Stages

| Design | Local Build | CI Build | Deploy to first Env | Pre-Production | Production |

Data & API Security should be a priority across all these stages

Observability & Discovery - Foundational for API & Data Security

Difficult to achieve at scale as it requires:

Real-time API Discovery

AuthN Automation

End User Context

Sensitive Data Discovery

An enterprise may have 5K-10K APIs & PBs of data, each with several parameters, making manual API spec writing and data securtiy impossible

APIs could be authenticating in different ways and all those schemes have to be automated from API Discovery to Documentation, which leads us to Security
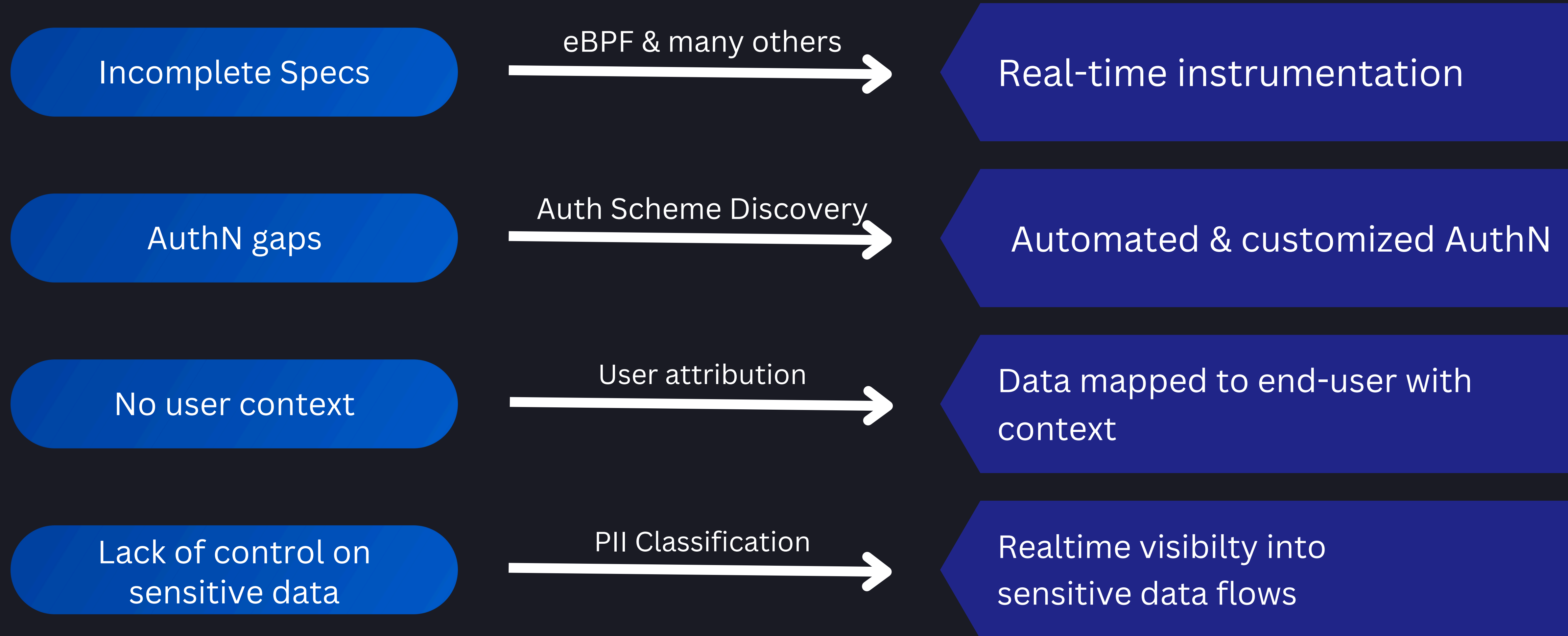
Identity and access details of the end-user using eBPF and this is the foundation we lay to effectively Securtiy test data & APIs at scale while also automating it

With the overflowing data sources within Enterprises, there is an uncontrolled exposure of Sensitive Data. Discovering Sensitive data and identifying their exposure across APIs is paramount

How we secure APIs and Data Access at scale

|		
Incomplete Specs	eBPF & many others →	Real-time instrumentation
AuthN gaps	Auth Scheme Discovery →	Automated & customized AuthN
No user context	User attribution →	Data mapped to end-user with context
Lack of control on sensitive data	PII Classification →	Realtime visibilty into sensitive data flows

Building upon eBPF

How we have maximized the impact of this powerful technology:

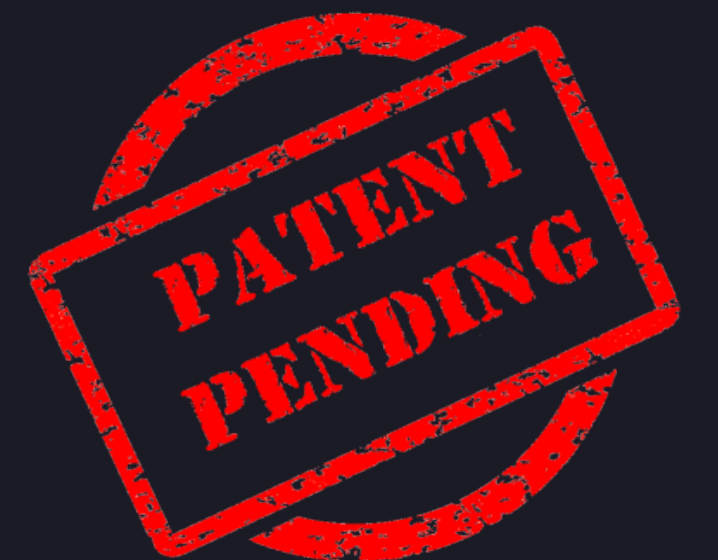API Usage insights without depending on application code changes
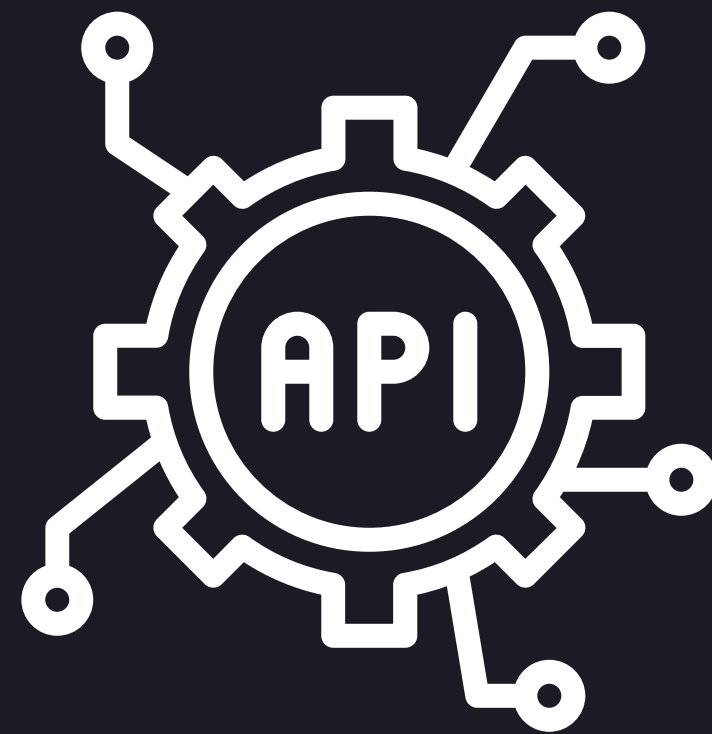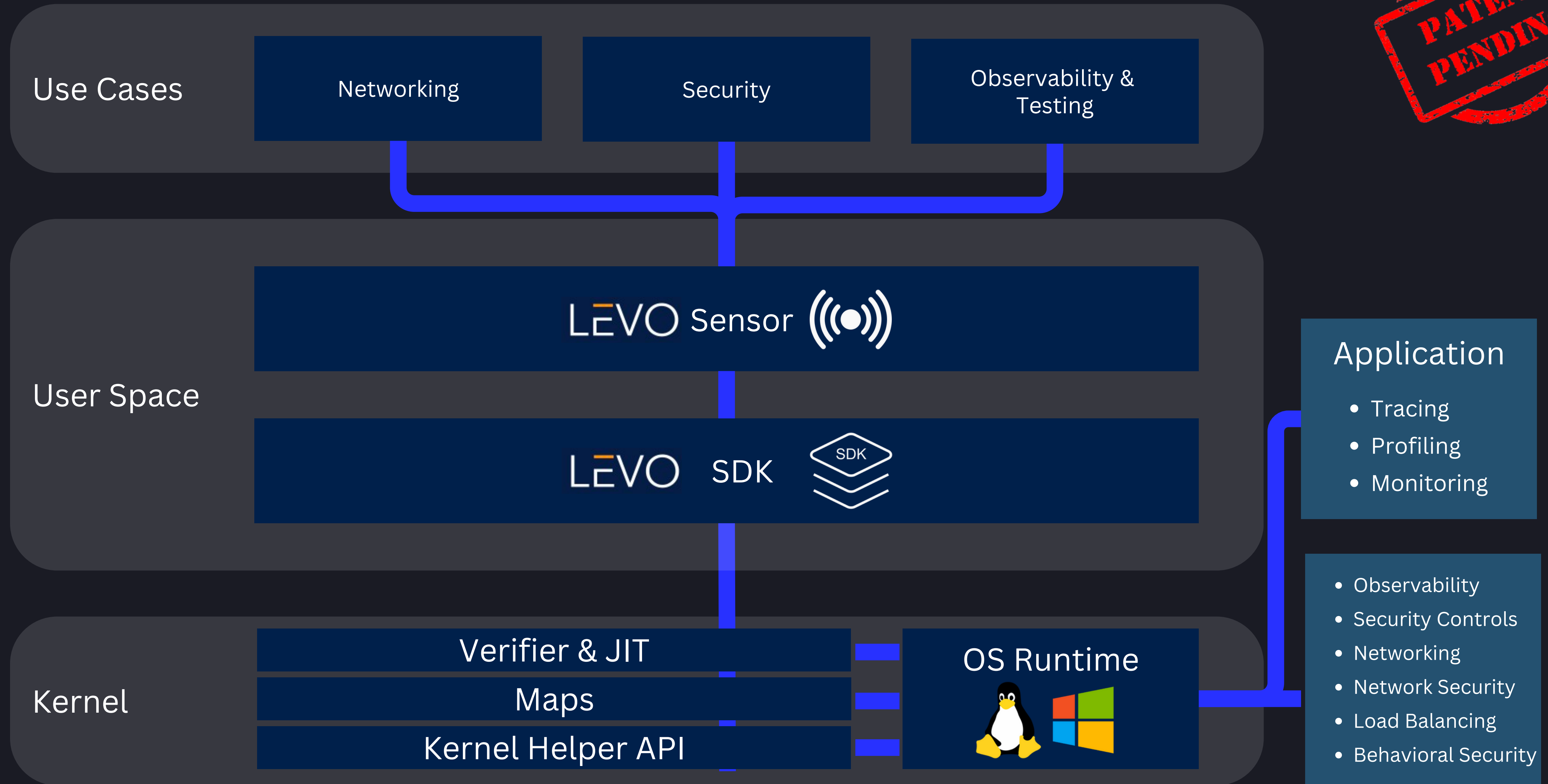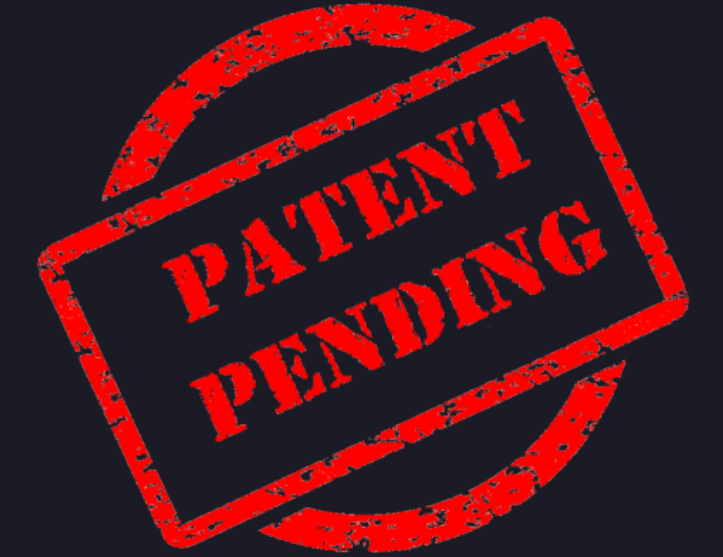
Automatic identification of API endpoints

Enhanced visibility & control over data, PII
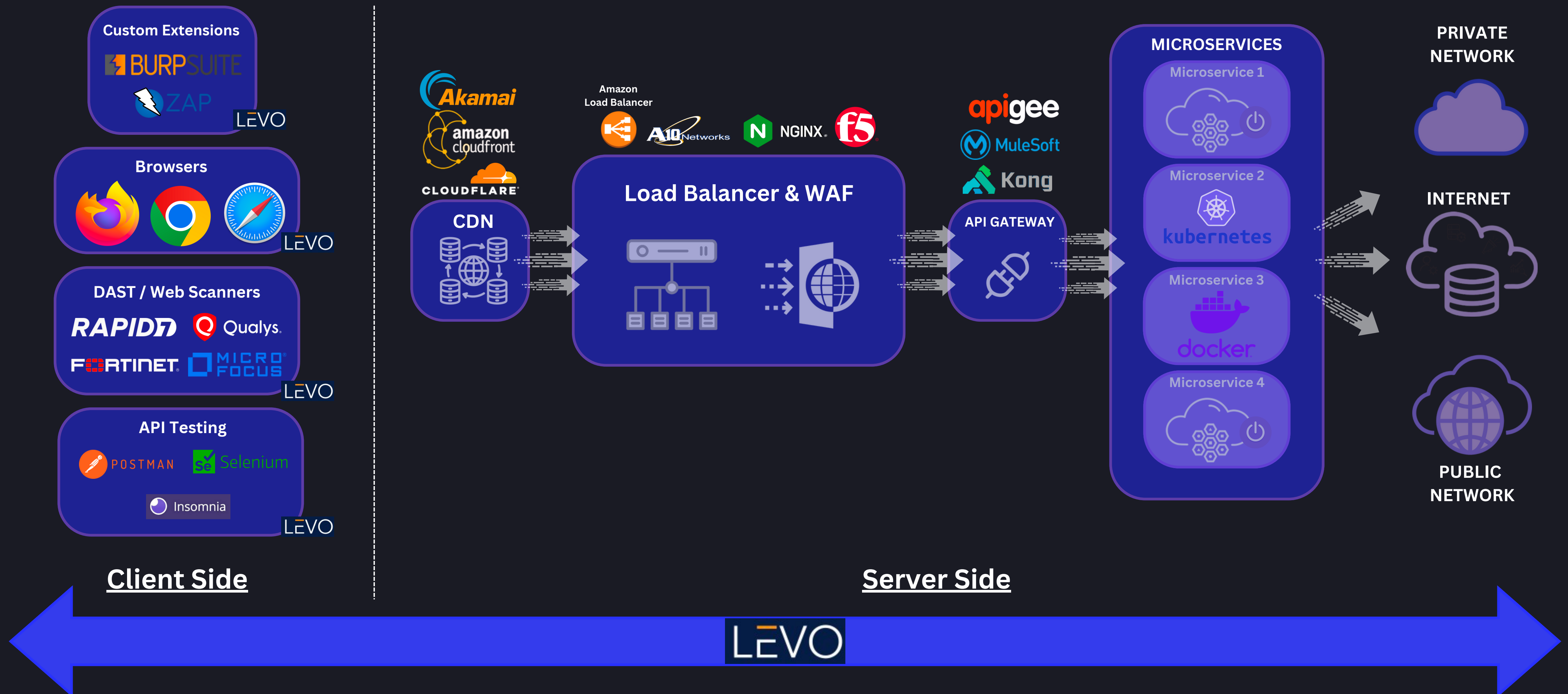
API Monitoring within complex infrastructures
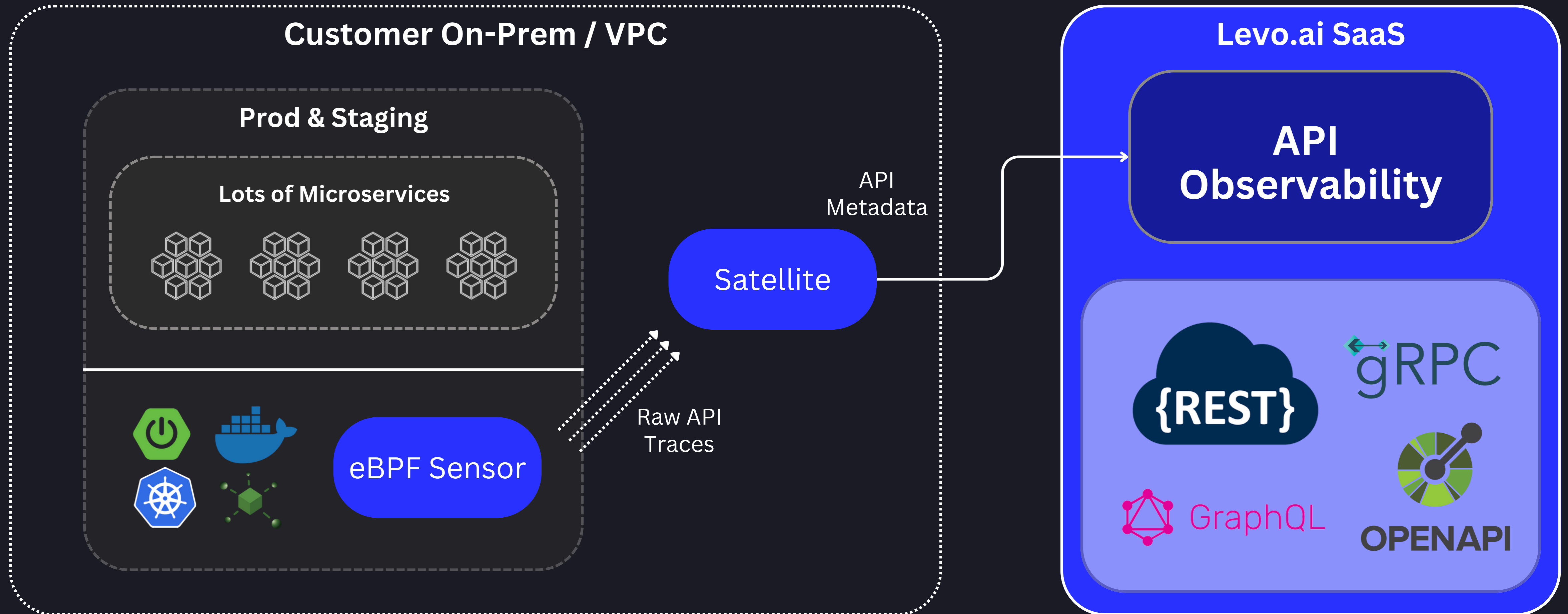
PATENT PENDING

Harnessing eBPF's magic for discovery

PATENT PENDING

Use Cases

Networking	Security	Observability & Testing

User Space

LEVO Sensor

LEVO SDK

Application

- Tracing
- Profiling
- Monitoring

Kernel

Verifier & JIT

Maps

Kernel Helper API

OS Runtime

- Observability
- Security Controls
- Networking
- Network Security
- Load Balancing
- Behavioral Security

Comprehensive discovery to give 100% visibility

Custom Extensions
BURP SUITE
ZAP
LEVO

Browsers
LEVO

DAST / Web Scanners
RAPID7 Qualys
F⊃RTINET MICRO FOCUS
LEVO

API Testing
POSTMAN Selenium
Insomnia
LEVO

Akamai
amazon cloudfront
CLOUDFLARE

CDN

Amazon Load Balancer
A10 Networks NGINX f5

Load Balancer & WAF

apigee
MuleSoft
Kong

API GATEWAY

MICROSERVICES
Microservice 1
Microservice 2 kubernetes
Microservice 3 docker
Microservice 4

PRIVATE NETWORK

INTERNET

PUBLIC NETWORK
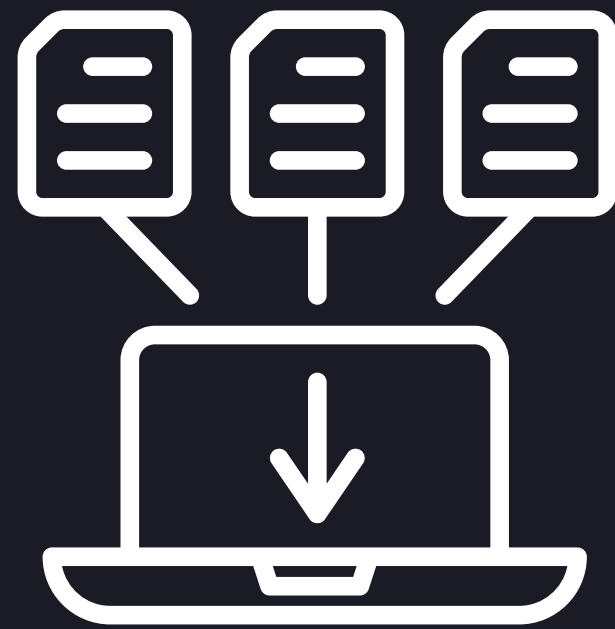
Client Side

Server Side

LEVO

Frictionless API & Data Observability

Levo.ai's Privacy Preserving Architecture

Automate API Discovery and Documentation

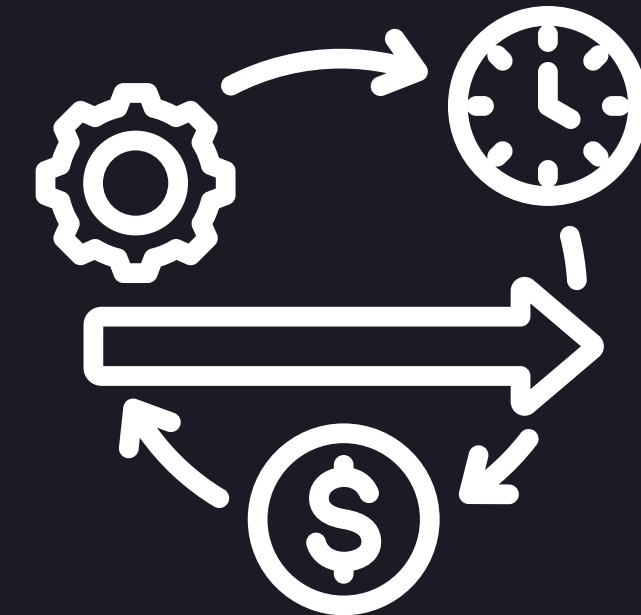## What is eBPF capable of doing :

Passive and real-time discovery of all internal, external and thirdparty endpoints at scale without any app changes

Comprehensive discovery regardless of languages, frameworks, SSL protocols and runtimes with rate-limiting and configuration at every stage
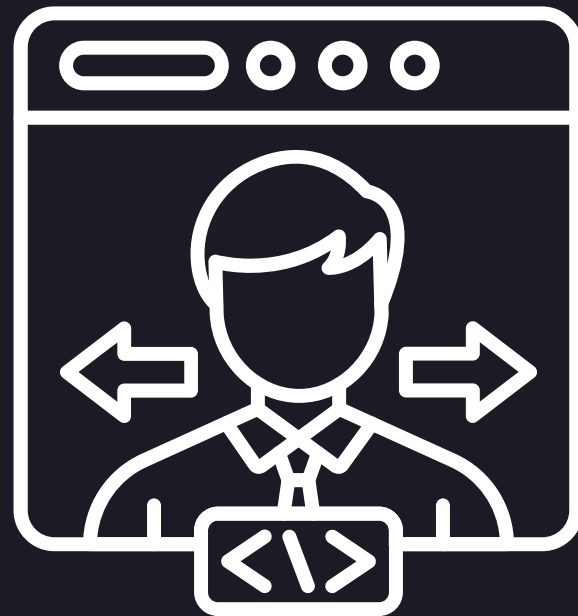
Automatic generation of accurate OpenApi/Swagger specs

Accelerated secure development cycle through application ownership

Test for API consumption, not outdated assumptions

How we harness end-user context with eBPF:

Accurately simulate real-world scenarios through enduser payload banks

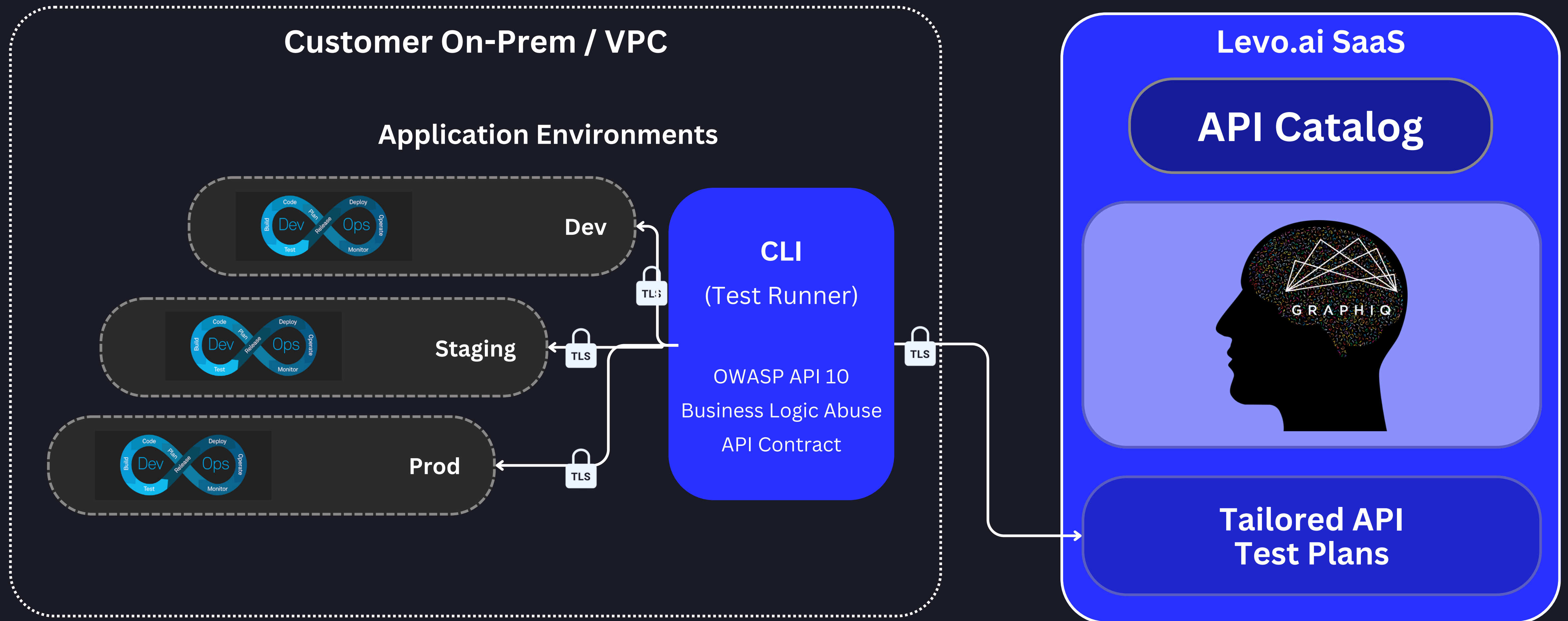Effective testing to achieve compliant API behavior with proper 2xx responses

Extrapolate and increase coverage by reusing across endpoints

Thank You

Appendix

Autogenerated Security Tests execute in CI/CD

Automate API Discovery and Documentation