DevSecOps and Zero Trust Architecture (ZTA) For Multi-Cloud Environments

Welcome to the second NIST-Tetrate Conference!
By Varun Talwar

Co-founder of Tetrate, Co-creator of gRPC and Istio projects
90% of web-enabled applications will be more exposed to attack by API weaknesses than via the user interface. According to Gartner, by 2022, API abuses will be the most-frequent attack vector for enterprise web applications data breaches.

DevSecOps ≠ More burden on Developers
When developers are writing a service, they worry a lot about the API, what are the methods, how does it work?
When developers are writing a service, they worry a lot about the API, what are the methods, how does it work?

But when you’re deploying microservices, then you start need to think about other questions:

What are the policies that are calling this service?

Does it have a quota?
Does it have a denial of service?
How does it get authenticated?
How is it secured?
When developers are writing a service, they worry a lot about the API, what are the methods, how does it work?

But when you’re deploying microservices, then you start need to think about other questions:

What are the policies that are calling this service?
Does it have a quota?
Does it have a denial of service?
How does it get authenticated?
How is it secured?

All of these questions are not about what the API does, but are operations pieces.
Why should the devs be burdened with implementing security?

Can the app runtime provide guarantee that a developer will never get security wrong?

Can we make “secured by default” – the norm?

or be concerned with defining security policies?
Why should the devs be burdened with implementing security?

Can the app runtime provide guarantee that a developer can never get security wrong?

Can we make “secured by default” - the norm?

Istio at its core decouples developers from operations.
Can the app runtime provide guarantee that a developer can never get security wrong?

Can we make "secured by default" the norm? or be concerned with defining security policies?

Why should devs be burdened with implementing security?

Istio at its core decouples developers from operations

Istio’s core security features

- Authentication & Authorization
- Encryption of service communication at scale
- Service communications are secured by default
- Enforce policies consistently across diverse protocols

NIST IR 8313 – Attribute-based Access Control for Microservices-based Applications using Service Mesh

- an authenticatable runtime identity for services
- the ability to authenticate application (user) credentials
- encryption in transit of communication between services
- A Policy Enforcement Point (PEP) separately deployable and controllable from the application – the service mesh’s sidecar proxies
- And logs and metrics for monitoring policy enforcement
Application level Security OR Zero Trust
Thank you for tuning in!

Enjoy the conference