> 4,100

Publicly disclosed data breaches occurred in 2021

22 billion

records being exposed

bit.ly/vilojona-zerotrust

Zero Trust

Don’t trust anyone….

bit.ly/vilojona-zerotrust

What can you expect ?

- ✅ Simple introduction to ZTA
- ❌ Detailed description about ZTA
- ❌ THE way
- ✅ Live demo with Istio, Quarkus service and Minikube

bit.ly/vilojona-zerotrust

My goal for this presentation

Light
Sparkle of Curiosity
about ZTA

bit.ly/vilojona-zerotrust

Jonathan Vila

@vilojona

Contact

Java impassioned

Java Champions

Community driven

BCN } JUG

JBCN Conf

DevBcn

Speaker experience

DEVNEXUS

DEVOXX United Kingdom

JNation

JConf Méx.

JConf Centroamérica 2020

Jfokus

Developer from the root

PASCAL

Visual Basic

JS

php

Delphi

GO

{ // I // Code // Java }

Professional work

Red Hat

ocado TECHNOLOGY

netcentric A Cognizant Digital Business

Dev Advocate

sonar

sonar

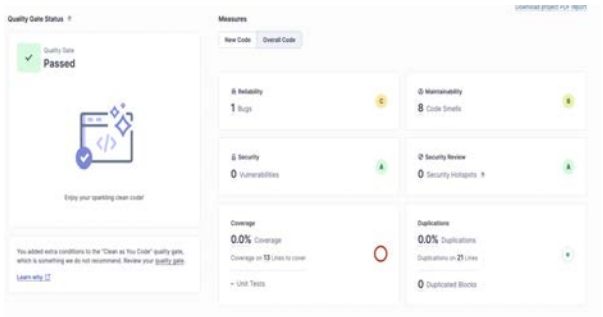
home of {clean code}

IDE

sonarlint Free

On Premise

sonarqube Free



Hosted

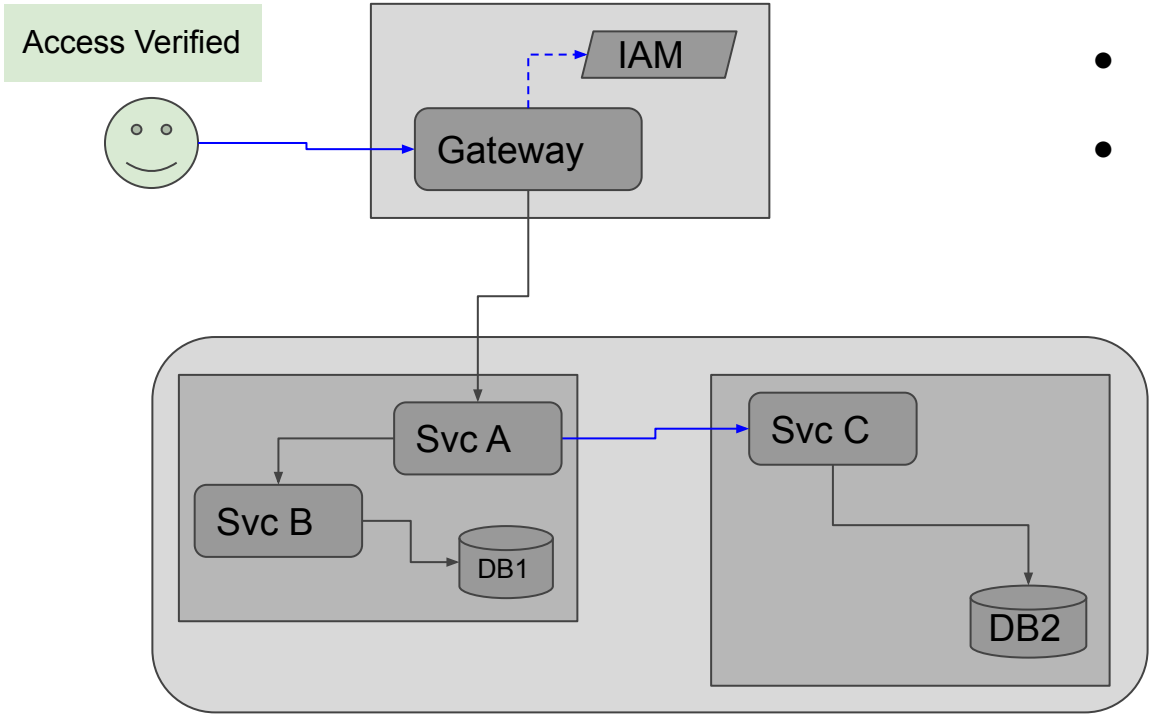
sonarcloud Free for Opensource

clean code throughout the development workflow

www.sonarsource.com

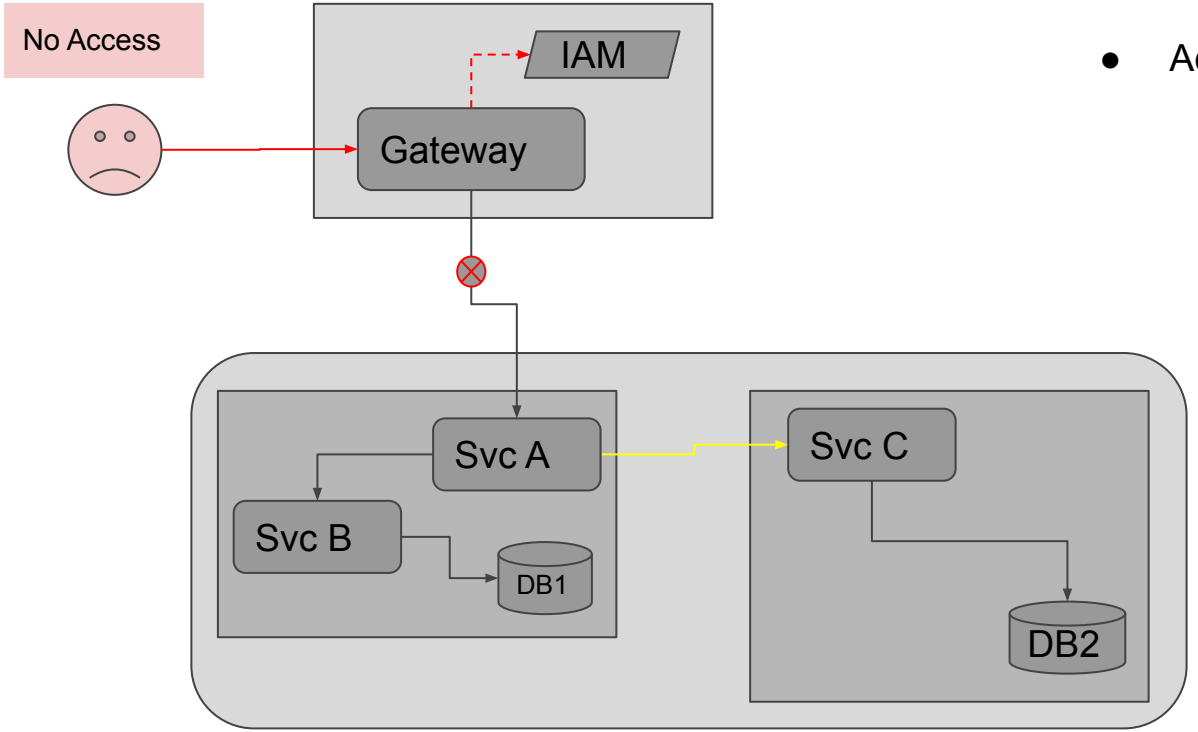
The usual context ….. Trust on perimeter

Trust on Perimeter



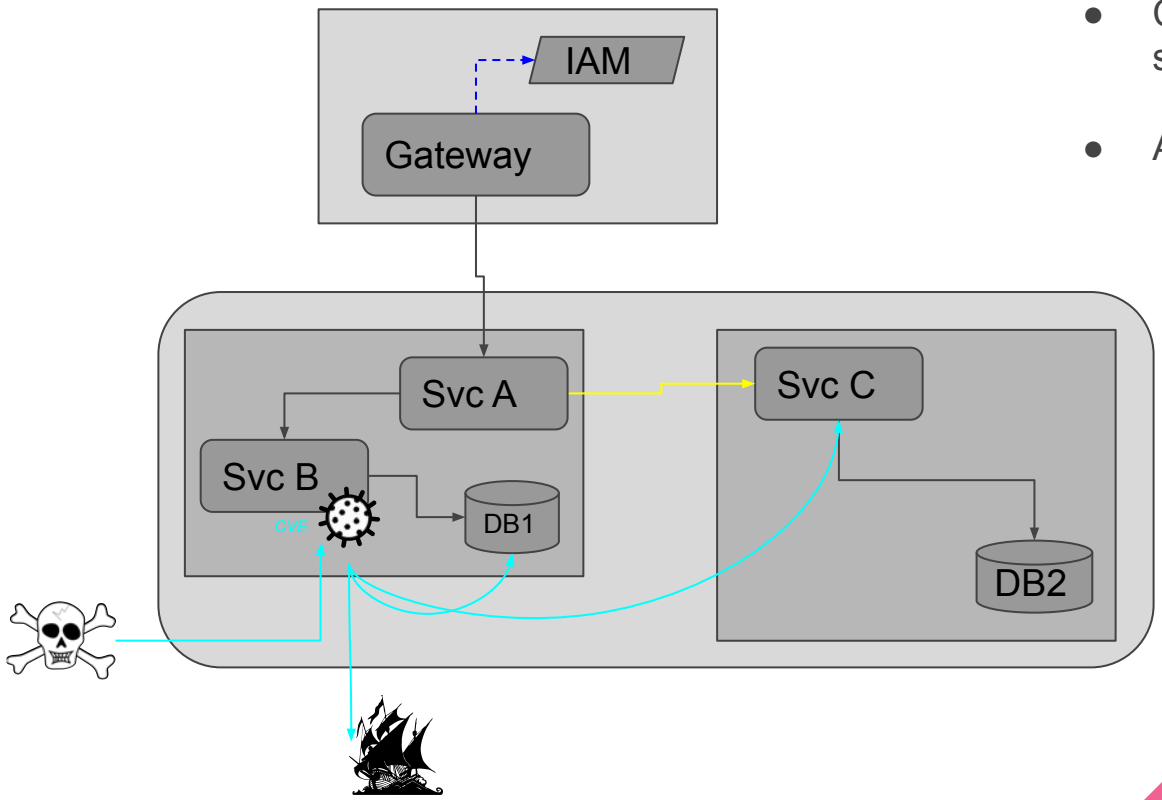
- 1 main secured door
- IAM / TLS verification on Gateway

Trust on Perimeter



- Access denial on Gateway

Trust on perimeter - the problem



- CVE : user can access a service inside the cluster
- Access to call any service

ZTA and how to approach the solution …

bit.ly/vilojona-zerotrust

Ways of mitigating the problem

Enforce identity validation in every service

Enforce using mTLS or Token validation in every call

Constraint callers and destinations

Zero Trust (ZTA / ZTNA / perimeter-less security)

Every user is assumed to be an attacker

Eliminates trust → never trust, always verify

ZTA Core Principles

Create Single, Strong User Identities and Single, Strong Device Identities

Always Authenticate Access, Anywhere in a Network

Know All Architecture

Policy-Setting

Never Trust the Network

Always Use Services Designed for Zero Trust

ZTA Challenges

Timely and costly implementation

Legacy software compatibility issues

3rd party technologies integration problems

Continuous maintenance and monitoring requirements

Implementation

Add SSL transport

Add Authz and Authn validation

Add Observability

Add rules for sources and destinations

Use clean code approach

Inspect CVEs in your code and libraries

To every application

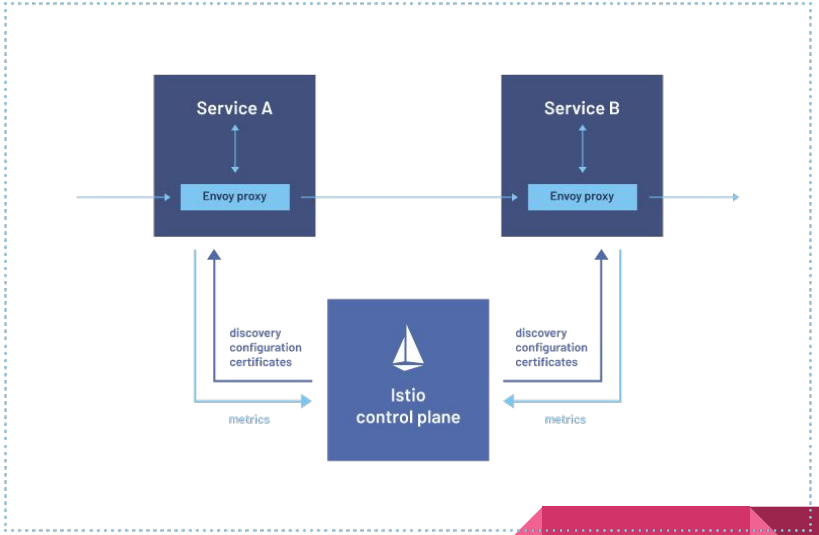
Or …. 

One approach to ZTA without touching apps code

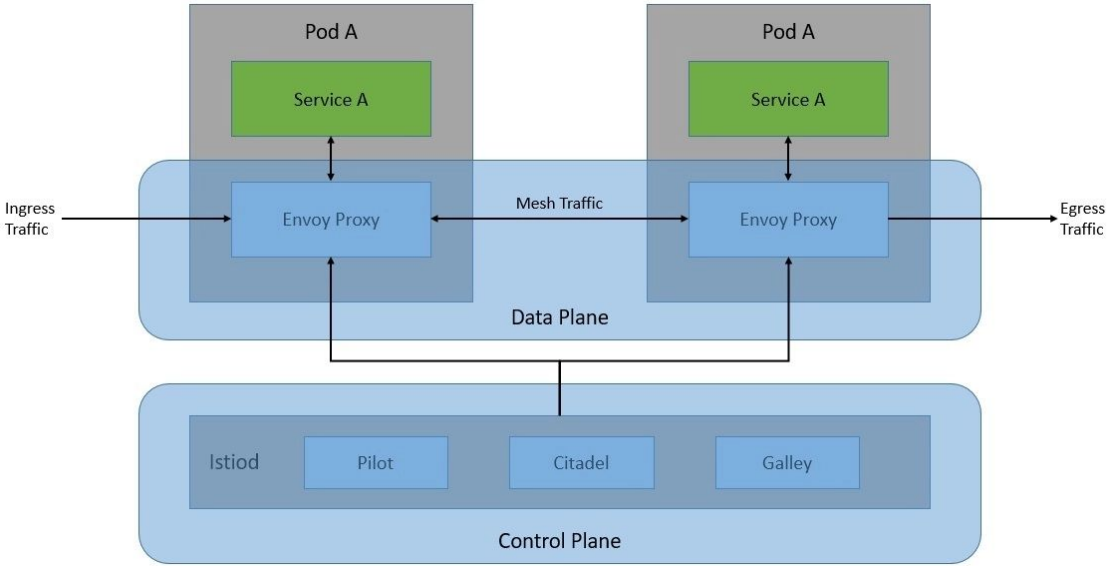
bit.ly/vilojona-zerotrust

Introducing Istio Service Mesh

- Collections of microservices
- Allows you to transparently add :
  - Observability
  - traffic management
  - security
- A/B testing
- Canary deployments
- Rate limiting
- Access control
- Encryption
- End-to-end authentication



How does Istio work ?

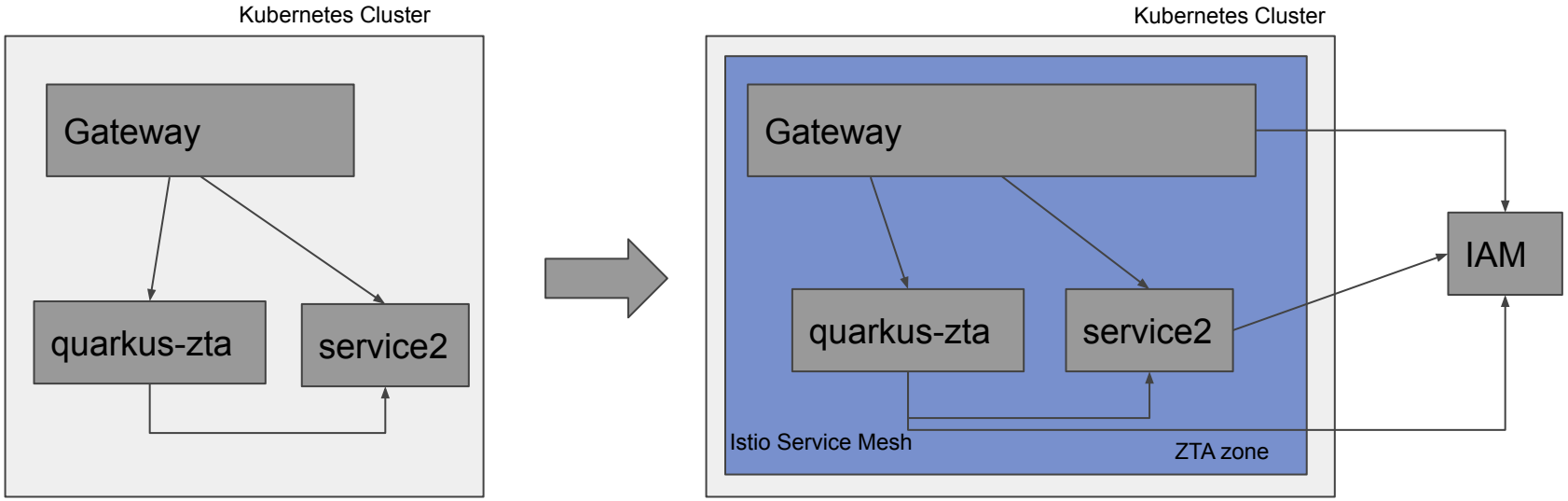


- Sidecar pattern
- Usage of Envoy proxies
- Intercepting network
- Adding filters and extensions over the networking

DEMO

bit.ly/vilojona-zerotrust

Demo


Kubernetes Cluster
Gateway
quarkus-zta
service2
Kubernetes Cluster
Gateway
quarkus-zta
service2
IAM
Istio Service Mesh
ZTA zone

Demo steps

1. No security
   a. Curl from outside
   b. Curl from inside
   c. Curl from inside to outside

2. With security
   a. Show Keycloak
   b. Curl from outside
   c. Curl from inside
   d. Show Kiali
   e. Curl from inside to outside

Files

a. Quarkus Service

b. Gateway
c. VirtualService
d. ConfigMap “istio-system.istio”

e. RequestAuthentication
f. AuthorizationPolicy
g. ServiceEntry

Our service

```java
@Path("/")
public class GreetingResource {

    @GET
    @Produces(MediaType.TEXT_PLAIN)
    @Path("hello")
    public String hello() {
        return "Hello from RESTEasy Reactive";
    }

    @GET
    @Produces(MediaType.TEXT_PLAIN)
    @Path("echo/{input}")
    public String echo(@PathParam("input") String input) {
        return "Echo from RESTEasy: " + input;
    }
}
```

https://github.com/jonathanvila/quarkus-simple-rest/blob/master/src/main/java/org/vilojona/GreetingResource.java

Demo Files … preparing Istio

```yaml
apiVersion: networking.istio.io/v1alpha3
kind: Gateway
metadata:
 name: demo-gateway
spec:
 selector:
   istio: ingressgateway # istio default
controller
 servers:
 - port:
     number: 80
     name: http
     protocol: HTTP
   hosts:
   - "*"
```

Creates a LoadBalancer accepting connections

```yaml
apiVersion: networking.istio.io/v1alpha3
kind: VirtualService
metadata:
 name: demo
spec:
 hosts:
 - "*"
 gateways:
 - demo-gateway
 http:
 - match:
   - uri:
       prefix: /echo
   route:
   - destination:
       host: quarkus-zta
       port:
         number: 80
```

Traffic routing roule to our service

Istio Config (ConfigMap istio-system.istio)

```
meshConfig.outboundTrafficPolicy.mode = REGISTRY_ONLY  # vs ALLOW_ANY
```

Restrict connections only to known services

Demo Files …. enforcing security

RequestAuthentication

```
apiVersion: security.istio.io/v1beta1
kind: RequestAuthentication
metadata:
 name: requestauth
 namespace: default
spec:
 selector:
   matchLabels:
     app: quarkus
 jwtRules:
  - issuer:
https://lemur-2.cloud-iam.com/auth/realms/qua
rkus-demo
    jwksUri:
https://lemur-2.cloud-iam.com/auth/realms/qua
rkus-demo/protocol/openid-connect/certs
```

Connects to Keycload token issuer

AuthorizationPolicy

```
apiVersion: security.istio.io/v1beta1
kind: AuthorizationPolicy
metadata:
 name: authpolicy
 namespace: default
spec:
 selector:
   matchLabels:
     app: quarkus
 rules:
 - from:
   - source:
       requestPrincipals: ["*"]
```

Forces requests to have valid Token

ServiceEntry

```
apiVersion: networking.istio.io/v1alpha3
kind: ServiceEntry
metadata:
 name: google
 namespace: default
spec:
 hosts:
 - www.google.com
 ports:
 - number: 443
   name: https
   protocol: HTTPS
 resolution: DNS
 location: MESH_EXTERNAL
```

Internal Service for external endpoint

Almost finished

bit.ly/vilojona-zerotrust

Conclusions

Take Security very seriously

ZTA is the way to go to minimize security issues

It can be costly to implement

It involves security inside your cluster

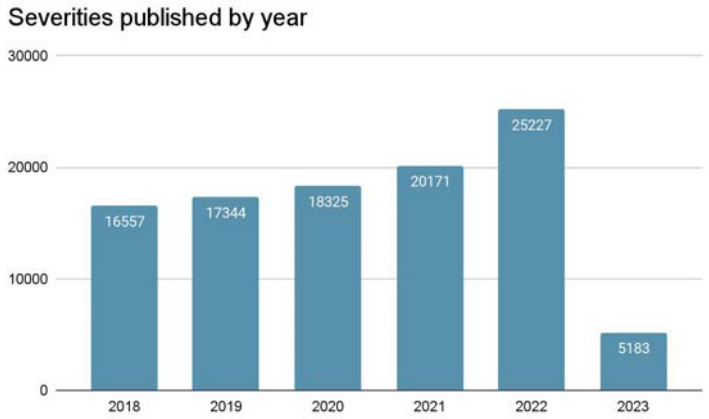
Service Mesh can help you

Transparent for existing applications

Introduces networking complexity

Allows to implement gradual security steps

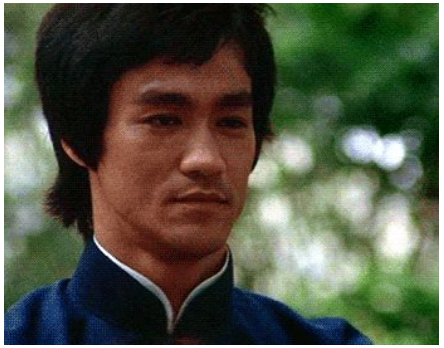
High level of customization



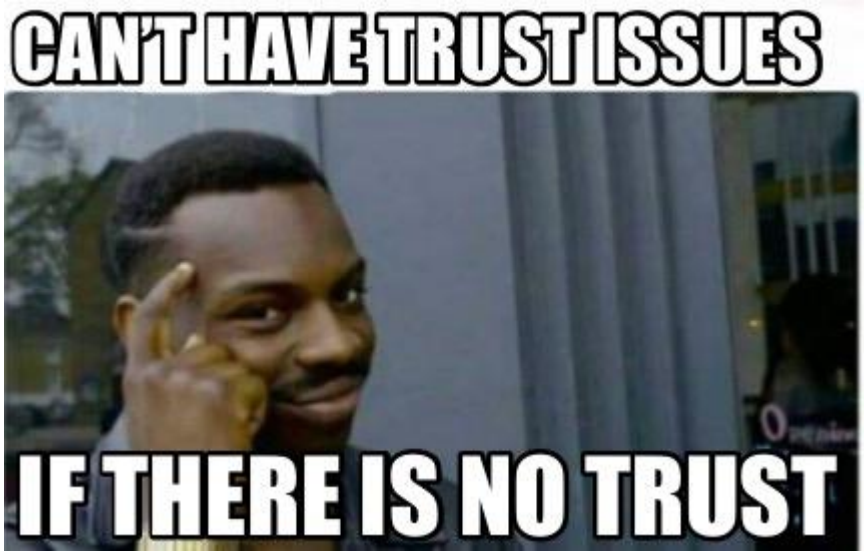
2022 → 866 CVEs with Score ~> Log4Shell

References

- CVEs and Data Bridges in 2021
  https://www.securitymagazine.com/articles/97046-over-22-billion-records-exposed-in-2021
- CVE List
  https://www.cvedetails.com/
- Log4Shell CVE Explained
  https://en.wikipedia.org/wiki/Log4Shell
- NIST Zero Trust Architecture definition
  https://www.nist.gov/publications/zero-trust-architecture
- Istio Service Mesh
  https://istio.io/latest/docs
- Quarkus
  https://quarkus.io/
- Minikube
  https://minikube.sigs.k8s.io/docs/
- Steps connecting Istio and Keycloak
  https://aytartana.wordpress.com/2023/03/02/adding-authentication-with-no-code-istio-and-keycloak/
- Code Quality and Security
  https://www.sonarsource.com/blog/tag/security/
- Source code Quarkus service
  https://github.com/jonathanvila/quarkus-simple-rest



Thank you :)



@vilojona

jonathan.vila@gmail.com

aytartana.wordpress.com

github.com/jonathanvila