

Are Your Organization's CI/CD Pipelines Secure?

Angel Rivera - @punkdata

Developer Advocate



**DEVOPS
WORLD**
by CloudBees

AGENDA

- Intro
- SDLC + CI/CD
- DevSecOps + Pipeline Integrations
- Securing Pipelines
- Recap

@punkdata

Hi I'm **Angel**

Developer Advocate.

My job is to **inspire**
developers!



@punkdata

@punkdata

History : Software : Development

@punkdata

Software : Development : Practices

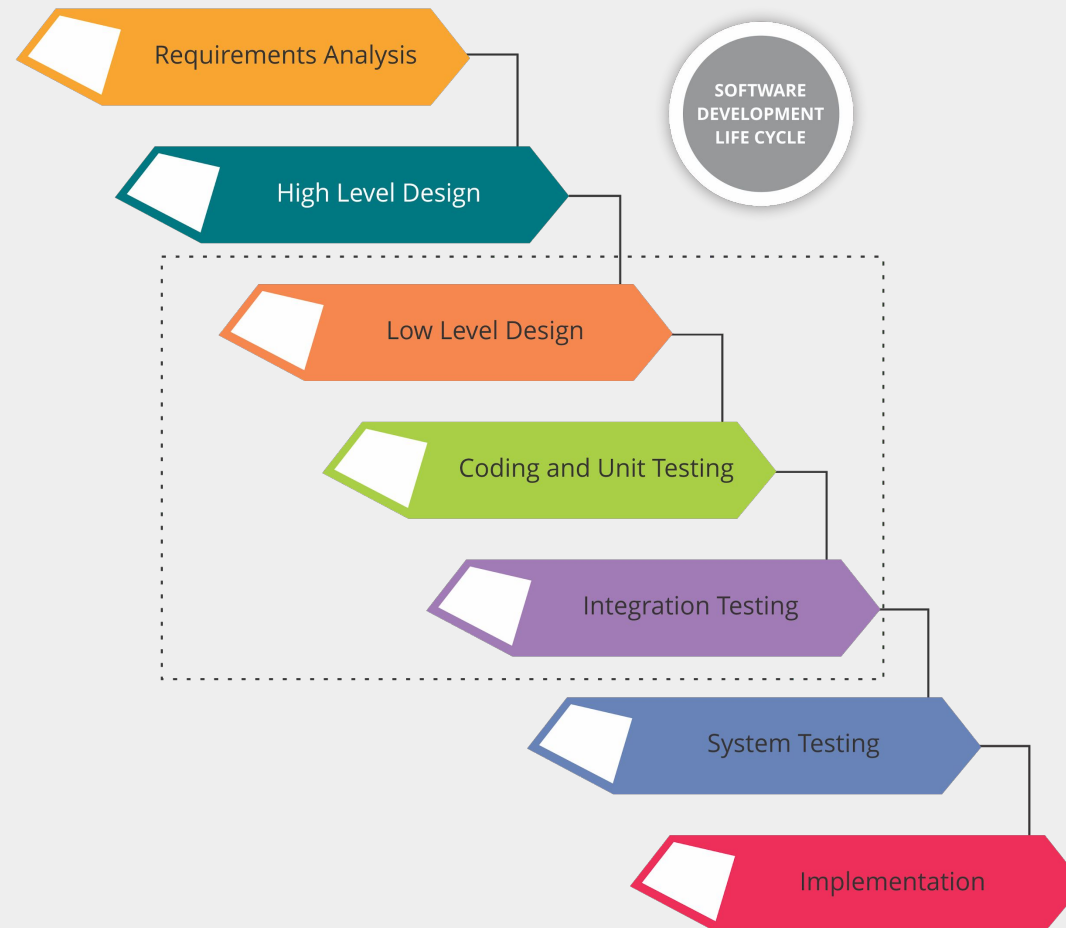
.circleci > ! config.yml > ...

```
576 - setup_ci_environment
577 - run:
578     name: Test
579     no_output_timeout: "1h"
580     command: |
581         set -e
582         # TODO: merge this into Caffe2 test.sh
583         cat >/home/circleci/project/ci_test_script.sh \<<EOL
584         # ===== The following code will be executed inside Docker container =====
585         set -ex
586
587         export BUILD_ENVIRONMENT="$BUILD_ENVIRONMENT"
588
589         # libdc1394 (dependency of OpenCV) expects /dev/raw1394 to exist...
590         sudo ln /dev/null /dev/raw1394
591
592         # conda must be added to the path for Anaconda builds (this location must be
593         # the same as that in install_anaconda.sh used to build the docker image)
594         if [[ "${BUILD_ENVIRONMENT}" == conda* ]]; then
595             export PATH=/opt/conda/bin:$PATH
596         fi
597
598         # Upgrade SSL module to avoid old SSL warnings
599         pip -q install --user --upgrade pyOpenSSL ndg-httpsclient pyasn1
600
601         pip -q install --user -b /tmp/pip_install_onnx "file:///var/lib/jenkins/workspace/third_party/onnx#egg=onnx"
602
603         # Build
604         ./jenkins/caffe2/test.sh
605
606         # Remove benign core dumps.
607         # These are tests for signal handling (including SIGABRT).
608         rm -f ./crash/core.fatal_signal_as.*
609         rm -f ./crash/core.logging_test.*
610         # ===== The above code will be executed inside Docker container =====
611         EOL
612         chmod +x /home/circleci/project/ci_test_script.sh
613
614         if [[ "${BUILD_ENVIRONMENT}" == *cmake* ]]; then
```

Waterfall : Development

@punkdata

Waterfall: Development



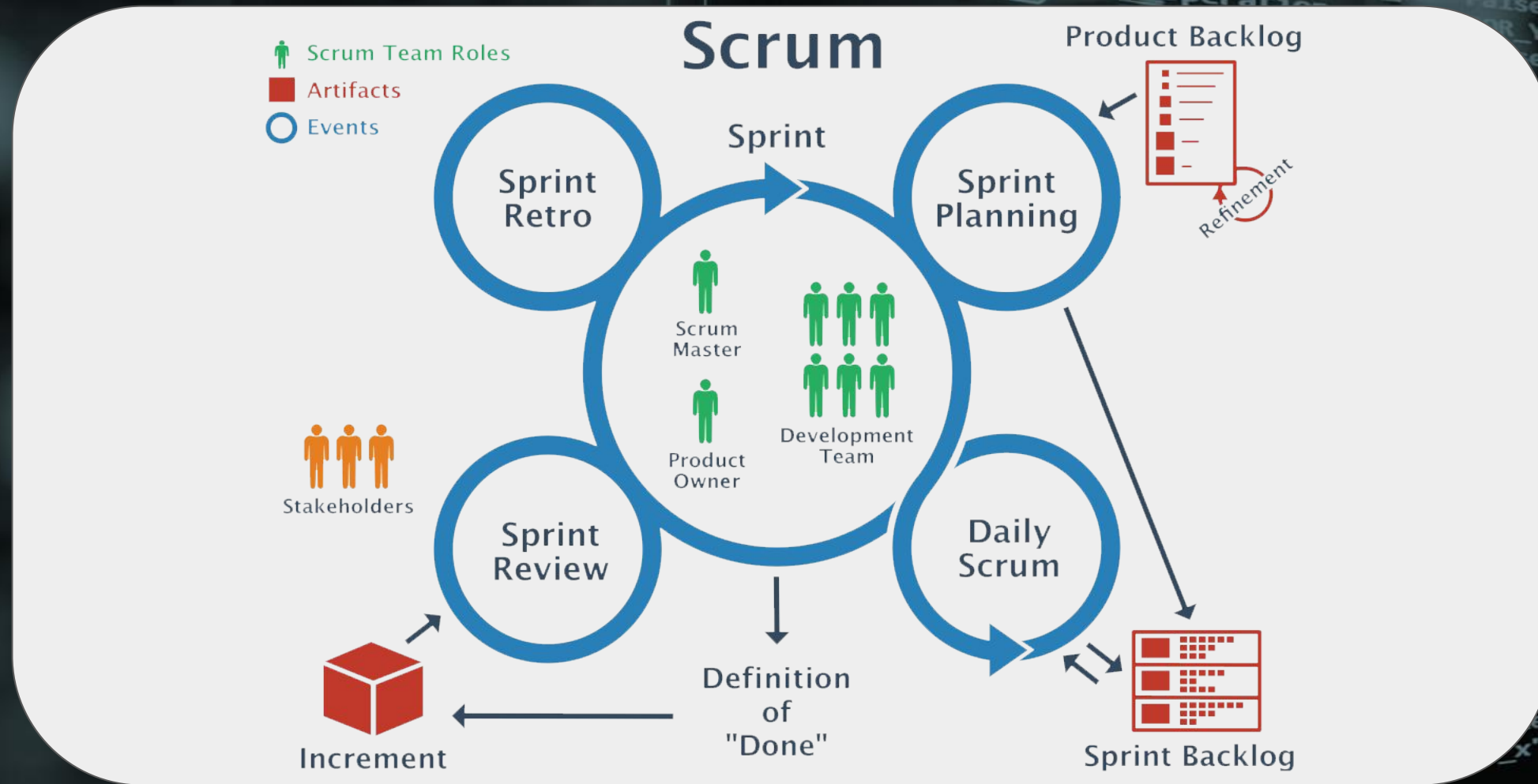
@punkdata



Agile : Development

@punkdata

Agile : Development



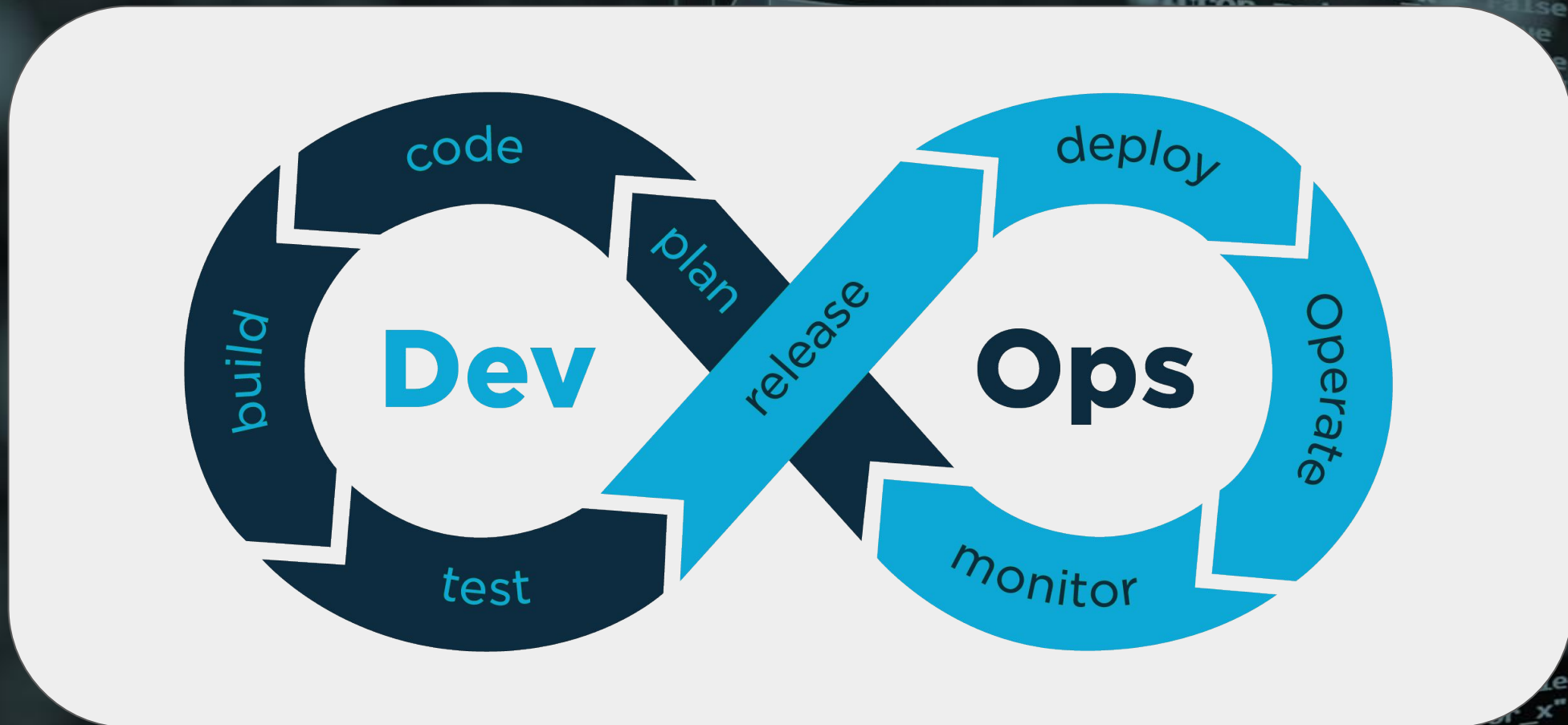
@punkdata

Continuous Integration : Continuous Delivery

CI /CD

@punkdata

CI/CD + DevOps



@punkdata

CI : Dev : Principles

- Write and commit code **often**
- Commit to a **shared** code repository
- **Test** code on every commit (unit + smoke level tests)
- Fast **Feedback** Loops

@punkdata

CD : Ops : Principles

- Create release **artifacts** for CD process
- **Deploy** code artifacts to resources
- **Validate** apps and services are functioning
- **Monitor** to verify state and recover if failing

@punkdata



Automation

SEARCH



Innovation
Branding
Solution
Marketing
Analysis
Ideas
Success
Management

Logistics



INDUSTRY

Manufacturing
Supply chain
Product
Cargo
Customer
Delivery
Inventory
Management
Freight

@punkdata

The background features a dark gray rectangle on the left and a large black circle on the right, both containing faint, horizontal white lines that resemble code or data. The text "CI/CD : Pipelines" is centered in white, bold font across the middle of these shapes.

CI/CD : Pipelines

@punkdata

CI/CD : Pipelines

nodejs-circleci > test-hashiterraform > build_test

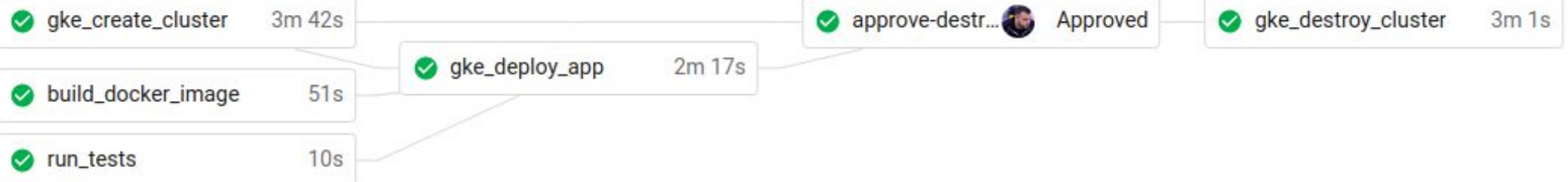
build_test ✓ SUCCESS

🕒 Duration 10m 17s
Finished 2 months ago

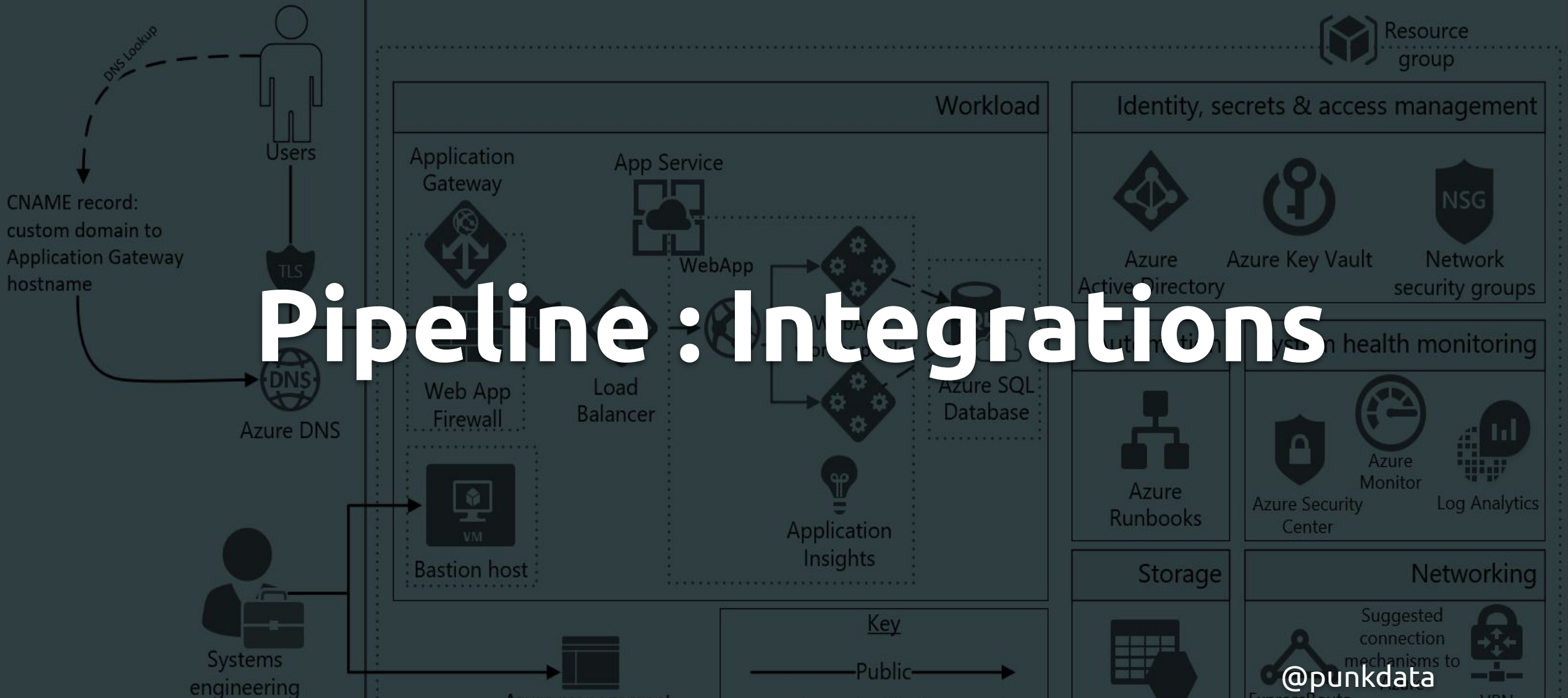
🔑 f7df391

test-hashiterraform

 punchdata



Pipeline : Integrations





Integration : Types

@punkdata

Pipeline : Integration : Types

- App Code Coverage Tools
- Vulnerability Scanning tools
- **API** Requests
- Database Access
- **Cloud** Provider CLI tools
- Infrastructure as Code tools

@punkdata

Dev : Sec : Ops

@punkdata



Shift : Security : Left

@punkdata

A dark, moody photograph of a young child with light-colored hair, looking through a magnifying glass. The child's face is partially obscured by the lens, which is held close to their eye. The background is blurred, showing some foliage. The overall tone is serious and investigative.

Pipeline : Vulnerability : Scans

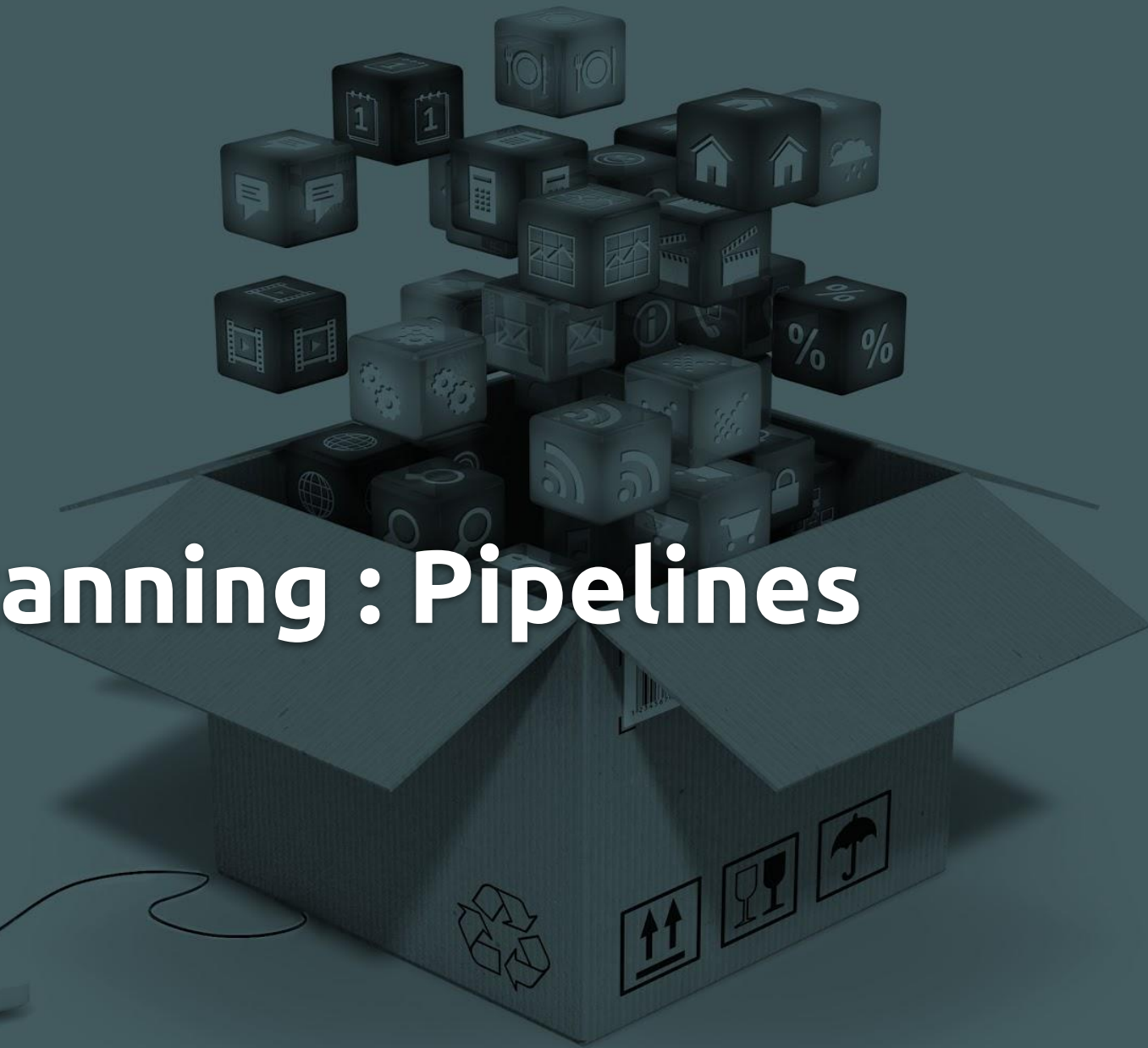
@punkdata



App : Scanning : Pipelines

@punkdata

Container : Scanning : Pipelines



@punkdata

Integration : Access

The background is a dark, monochromatic illustration. On the right, a silhouette of a person stands holding a laptop. To their left, several overlapping rectangular frames, resembling code editor windows, are shown. Behind these frames, a large gear is visible. The overall theme is technical and related to software development or system integration.

@punkdata

Integration : Access : Credentials

A stylized illustration in the background shows a person sitting at a desk, leaning back with their hand on their head, suggesting deep thought or problem-solving. Above their head are several gears and a glowing lightbulb, symbolizing ideas and technology.

- OAuth
- Access Keys
- API Tokens
- Certificate-based authentication
- JWT - JSON Web Tokens
- Username + Password

@punkdata

A person in a dark suit is seen from behind, standing in the center of a complex, three-dimensional maze. The person's right hand is on their head, and their left hand is on their lower back, suggesting a state of confusion or frustration. The maze is made of dark, rectangular blocks, creating a complex network of paths and dead ends. The lighting is dramatic, with strong shadows and highlights, emphasizing the depth and complexity of the maze.

CI/CD : Pipeline : Obstacles

@punkdata



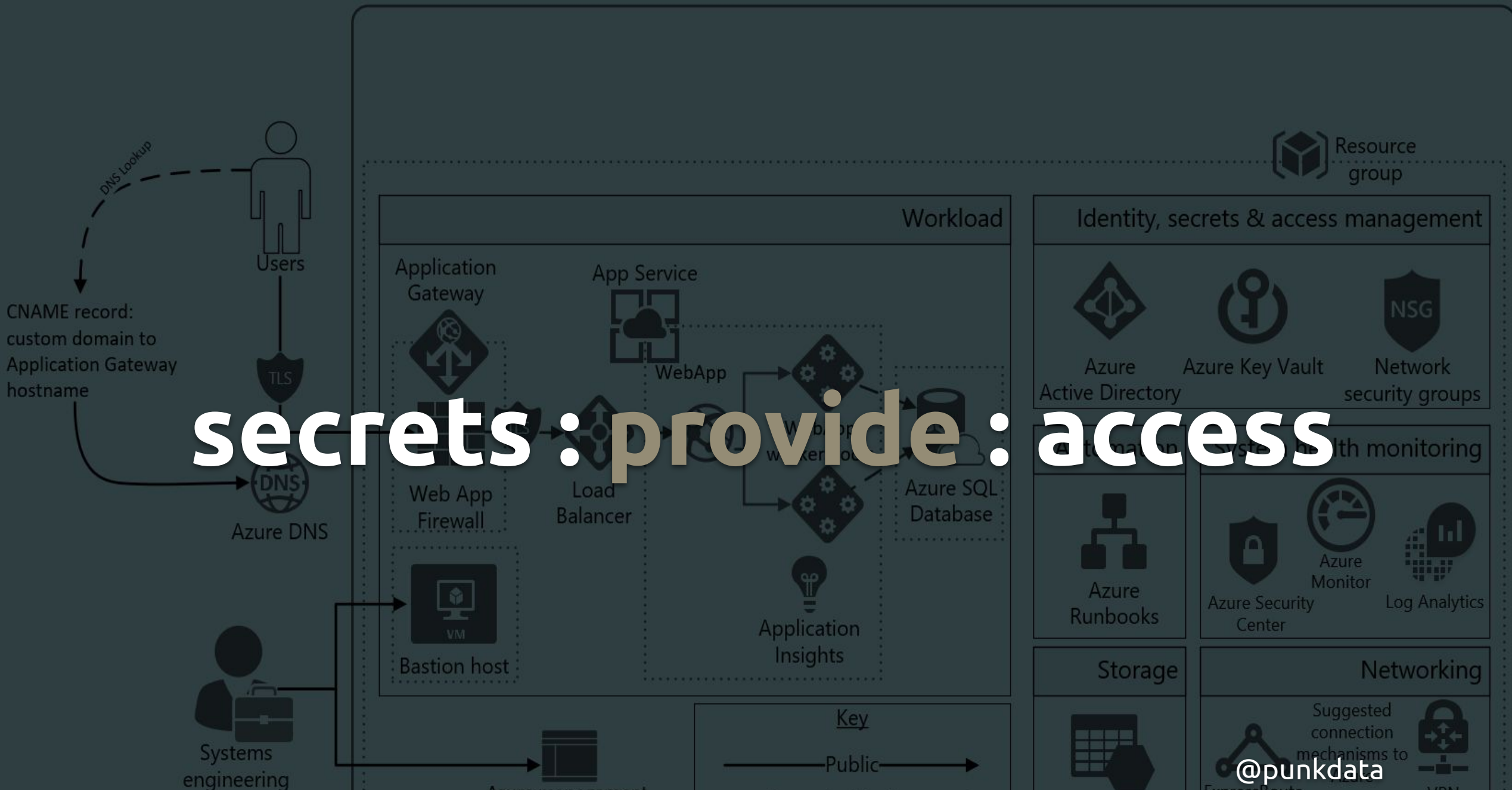
Securing Pipelines

@punkdata



Secrets

@punkdata





exposed : pipelines

@punkdata



weak : stale : secrets

@punkdata



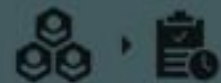
clear text

inadequate : mechanisms

unencrypted
@punkdata

secrets management

@punkdata



SECURITY TEAM
DEFINES POLICIES
Enforced at Runtime

CLIENT



TOKEN

tooling : protect : secrets

AUTHENTICATION

Identity-Based Access/Backends



SECRETS

Secrets Management/Backends

aws

Azure

GitHub



okta



LDAP



User +
Password



TLS
Certificate

Custom +

aws



Custom +



TOTP



Key/Values



Database



random : pwd: generation

@punkdata



auto : rotate : passwords

@punkdata

granular : access : controls



@punkdata



recap

@punkdata

recap : CI Principles

- Write and commit code often
- Commit to a shared code repository
- Test code on every commit (unit + smoke level tests)
- Fast Feedback Loops

@punkdata

recap : CD Principles

- Create release artifacts for CD process
- Deploy code artifacts to resources
- Validate apps and services are functioning
- Monitor to verify state and recover if failing

@punkdata

recap : DevSecOps

- Shift Security **Left**
- **App** Scanning within Pipelines
- Container **Scanning** within Pipelines

@punkdata

recap : Securing : Pipelines

- implement secrets management policies and tooling
- **integrate** secrets management tools into pipelines
- generate strong **random** secrets + credentials
- **auto rotate** + replace secrets used in pipelines

@punkdata



thank: you

Angel Rivera
Developer Advocate
 circleci

@punkdata