

Securing Serverless - By Breaking In

Guy Podjarny, Snyk
[@guypod](#)

About Me

- Guy Podjarny, **@guypod** on Twitter
- CEO & Co-founder at Snyk
- History:
 - Cyber Security part of Israel Defense Forces
 - First Web App Firewall (AppShield), Dynamic/Static Tester (AppScan)
 - **Security**: Worked in Sanctum -> Watchfire -> IBM
 - **Performance**: Founded Blaze -> CTO @Akamai
- O'Reilly author, speaker

Serverless Security: The Theory

(talk from ServerlessConf)

Serverless Security: What's Left to Protect?

Guy Podjarny, Snyk
[@guypod](#)

<https://snyk.io/blog/serverless-security-implications-from-infra-to-owasp/>

https://www.youtube.com/watch?v=CiyUD_rI8D8

Today - straight to practice!

Agenda

- Show a demo serverless app
- Hack it
- Explain the security flaws and how to fix them
- Summary
- Q&A

Going Terminal...

Vulnerable Libraries



Example: Fetch file & store in s3 (Serverless Framework Example)

```
'use strict';

const fetch = require('node-fetch');
const AWS = require('aws-sdk'); // eslint-disable-line import/no-extraneous-dependencies

const s3 = new AWS.S3();

module.exports.save = (event, context, callback) => {
  fetch(event.image_url)
    .then((response) => {
      if (response.ok) {
        return response;
      }
      return Promise.reject(new Error(
        `Failed to fetch ${response.url}: ${response.status} ${response.statusText}`));
    })
    .then(response => response.buffer())
    .then(buffer => (
      s3.putObject({
        Bucket: process.env.BUCKET,
        Key: event.key,
        Body: buffer,
      }).promise()
    ))
    .then(v => callback(null, v), callback);
};
```

19 Lines of Code

```
"dependencies": {
  "aws-sdk": "^2.7.9",
  "node-fetch": "^1.6.3"
}
```

2 Direct dependencies

19 dependencies (incl. indirect)

191,155 Lines of Code

Security

Equifax's disastrous Struts patching blunder: THOUSANDS of other orgs did it too

Those are just the ones known to have downloaded outdated versions

EQUIFAX DATA BREACH

Equifax's Mega-Breach Was Made Possible by a Website Flaw It Could Have Fixed

Failure to patch two-month-old bug led to massive Equifax breach

Critical Apache Struts bug was fixed in March. In May, it bit ~143 million US consumers.

DAN GOODIN - 9/13/2017, 11:12 PM

Serverless does secure OS dependencies

Just not app dependencies

1. Beware Vulnerable Libraries

(test during dev, monitor over time)

The screenshot displays the Snyk dashboard for a project named 'JeffConf-Demo'. The interface includes a navigation bar with 'Dashboard', 'Projects', 'Integrations', and 'Settings'. A search bar for repositories is present, along with a 'Search' button and a 'View last import log' link. A green 'Add projects' button is also visible. Below the search bar, there is a filter for 'AWS Lambda' with a count of 10. The main content area lists four AWS Lambda functions, each with a 'package.json' link, a vulnerability count (0 H, 1 M, 1 L), a 'View report' link, a 'Test weekly' dropdown, and a 'Tested 12 hours ago' timestamp.

Function	Package	Vulnerabilities	Report	Test Frequency	Last Tested
us-east-1/aws-node-twilio-dev-sendText(\$LATEST)	package.json	0 H, 1 M, 1 L	View report	Test weekly	Tested 12 hours ago
us-east-1/serverless-goof-dev-delete(\$LATEST)	package.json	3 H, 3 M, 1 L	View report	Test weekly	Tested 12 hours ago
us-east-1/serverless-goof-dev-get(\$LATEST)	package.json	3 H, 3 M, 1 L	View report	Test weekly	Tested 12 hours ago
us-east-1/serverless-goof-dev-render(\$LATEST)	package.json	3 H, 3 M, 1 L	View report	Test weekly	Tested 12 hours ago

Side Note: Snyk isn't only for Serverless



Which GitHub repositories do you want to test?

Snyk supports Node, Ruby, Java, Scala and Python projects. We require [relevant manifest files](#) to be able to test your repository.

Supported repositories

Last synced a few seconds ago

[Refresh](#)

Filter repos

guypod

attachinary

bluebird

body-parser

boto

broker-snyk-client-example

node-uuid

okta-sdk-java

pysyft

qonduit

rage4service

🔗 Open a fix PR

github.com/guypod/goof

[View test report](#)

1. Choose which vulnerabilities you would like to fix.
2. Open a pull request with the upgrades and patches to address these vulnerabilities.
3. The repository will be watched for new vulnerabilities and you will receive an alert when a new vulnerability affects your project.

Vulnerabilities with a fix

An upgrade or patch is available to fix the vulnerable dependencies.

L Regular Expression Denial of Service

H Content & Code Injection (XXE)

H Cross-site Scripting (XSS) via

M Remote Memory Exposure in

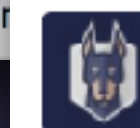
[Snyk Alert] Fix for 4 vulnerable dependency paths

Merged BerkeleyTrue merged 1 commit into staging from snyk-fix-9a4b3f6d on Jul 25, 2016

Conversation 1

Commits 1

Files changed 1



snyk-bot commented on Jul 24, 2016

Contributor +

The following newly disclosed vulnerabilities impact one or more of the npm packages this project uses:

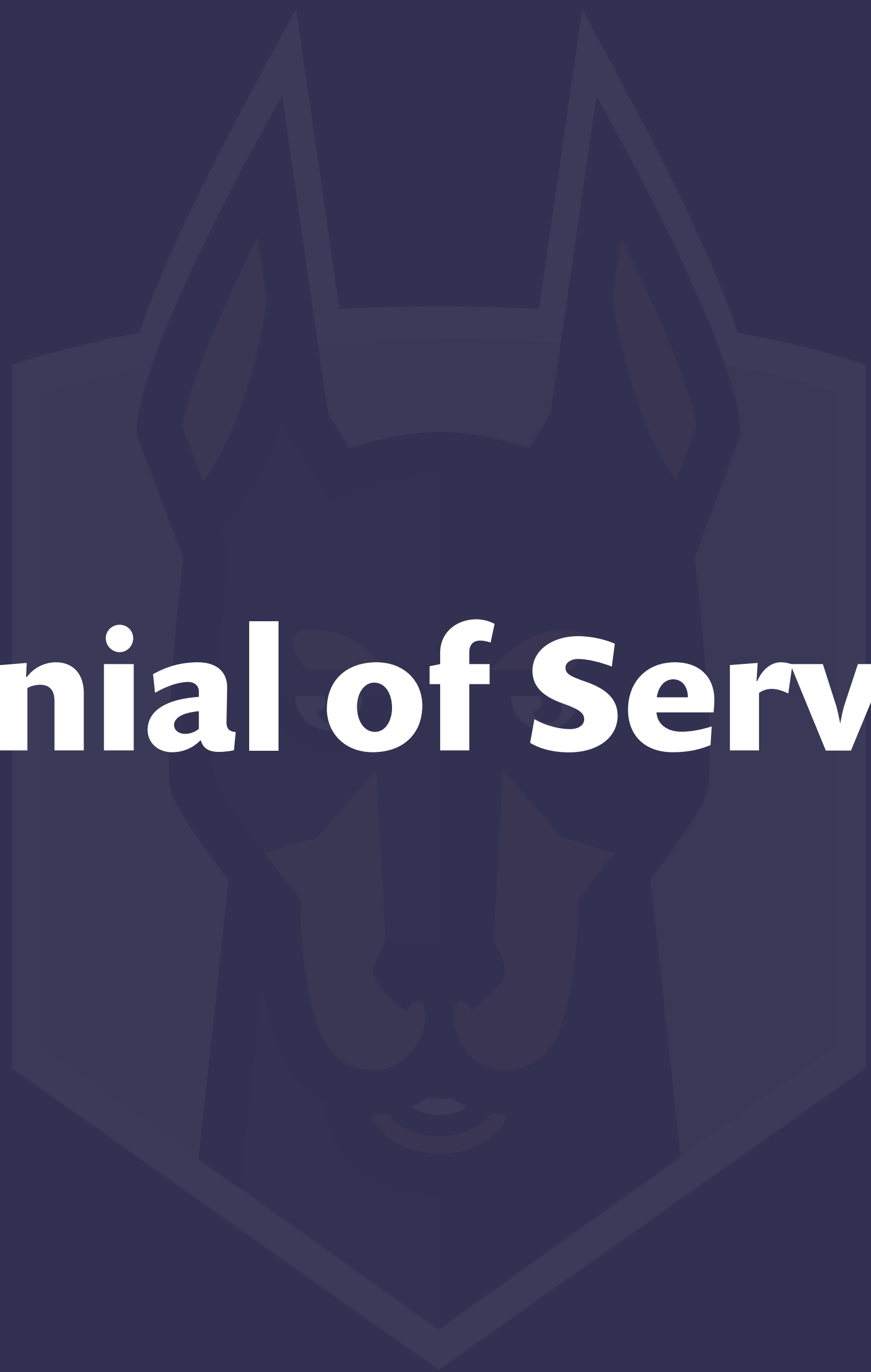
- [npm:minimatch:20160620](#)

As these vulnerabilities are now publicly known, attackers can try to use them against your application, making fixing them a matter of urgency.

To help expedite the fix, Snyk created this pull request with the necessary changes to address the vulnerabilities.

This pull request includes:

Denial of Service



2. ReDoS can still be costly (won't take you down, but can hike up bill)

```
REPORT RequestId: 80166153-6277-11e7-bf5f-3dd574c54fdd Duration: 24.91 ms Billed Duration: 100 ms Mem
START RequestId: 825afaaa-6277-11e7-aa62-b76fb81685d0 Version: $LATEST
END RequestId: 825afaaa-6277-11e7-aa62-b76fb81685d0
REPORT RequestId: 825afaaa-6277-11e7-aa62-b76fb81685d0 Duration: 344.10 ms Billed Duration: 400 ms Mem
START RequestId: 84fc8075-6277-11e7-b130-97509a23329e Version: $LATEST
END RequestId: 84fc8075-6277-11e7-b130-97509a23329e
REPORT RequestId: 84fc8075-6277-11e7-b130-97509a23329e Duration: 6000.14 ms Billed Duration: 6000 ms
```

Beware

Resource Exhaustion Attacks

Not all your services elastically scale



Secrets

3. Avoid secrets in deployed code

(env variables aren't enough - Use a KMS!)

```
const adminSecret = "ea29cbdb-a562-442a-8cc2-adbc6081d67c";

module.exports.api = (event, context, callback) => {

  if (!event.queryStringParameters ||
      !event.queryStringParameters.secret ||
      event.queryStringParameters.secret !== adminSecret) {
    // Return an unauthorized response
  }
}
```

Serverless platforms offer a **Key Management System**

Just use it!

Granularity



4. Deploy granular functions

(shared function code = greater exposure)

functions:

create: serverless-goof-dev-create

list: serverless-goof-dev-list

render: serverless-goof-dev-render

get: serverless-goof-dev-get

update: serverless-goof-dev-update

delete: serverless-goof-dev-delete

internalBackup: serverless-goof-dev-internalBackup

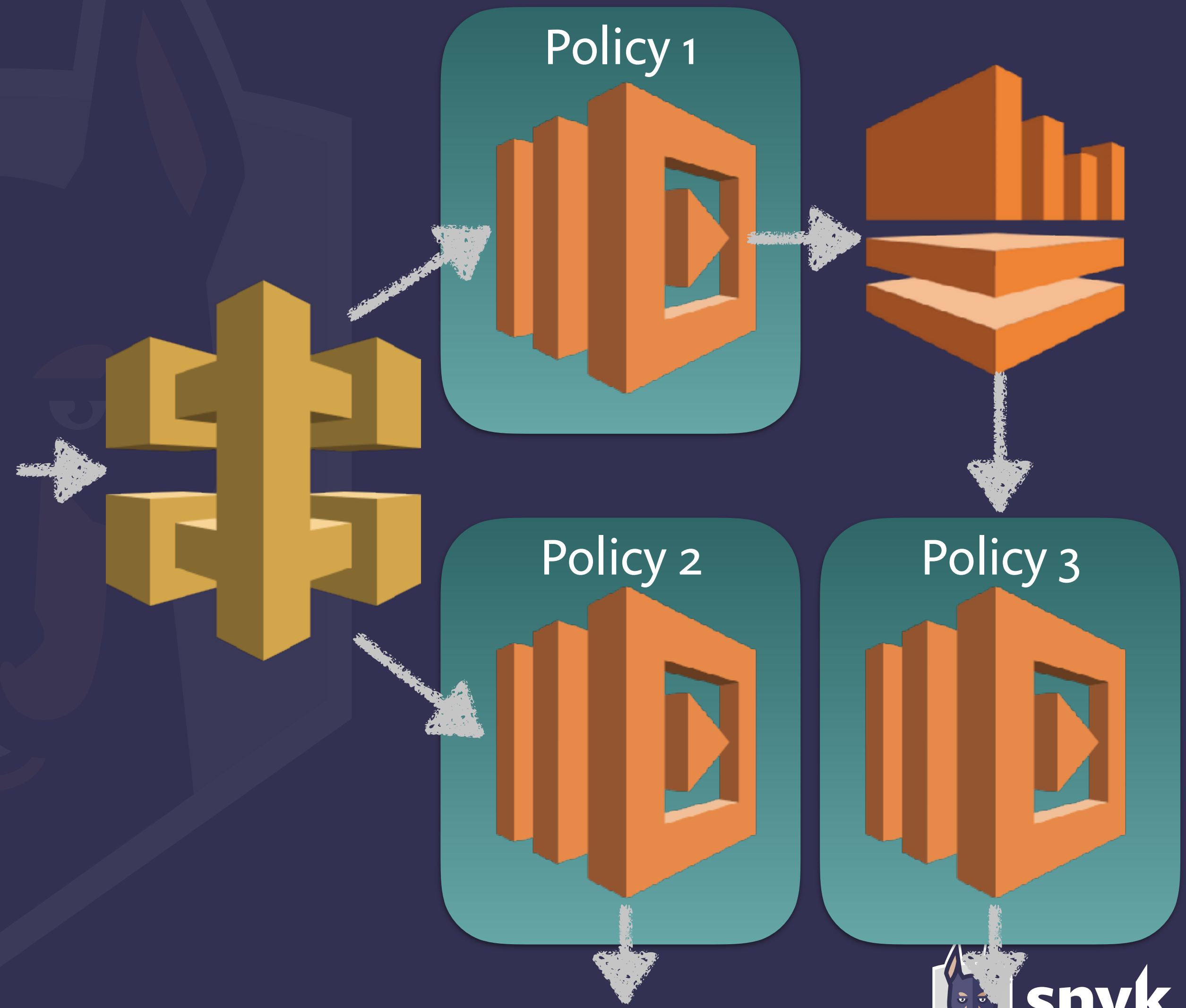
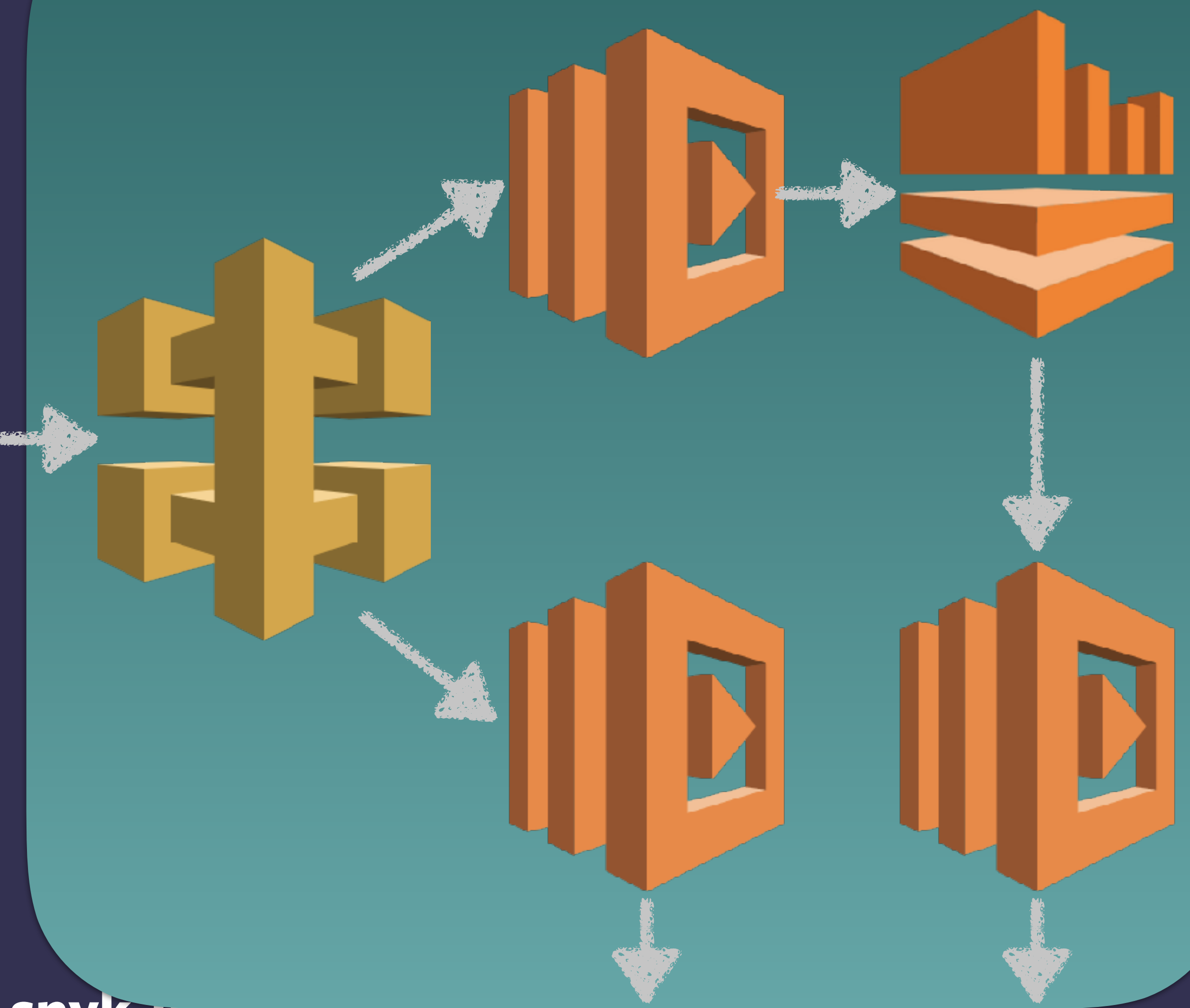
internalRestore: serverless-goof-dev-internalRestore

adminApi: serverless-goof-dev-adminApi

Easier

Safer

AWS Security Policy



Permissions

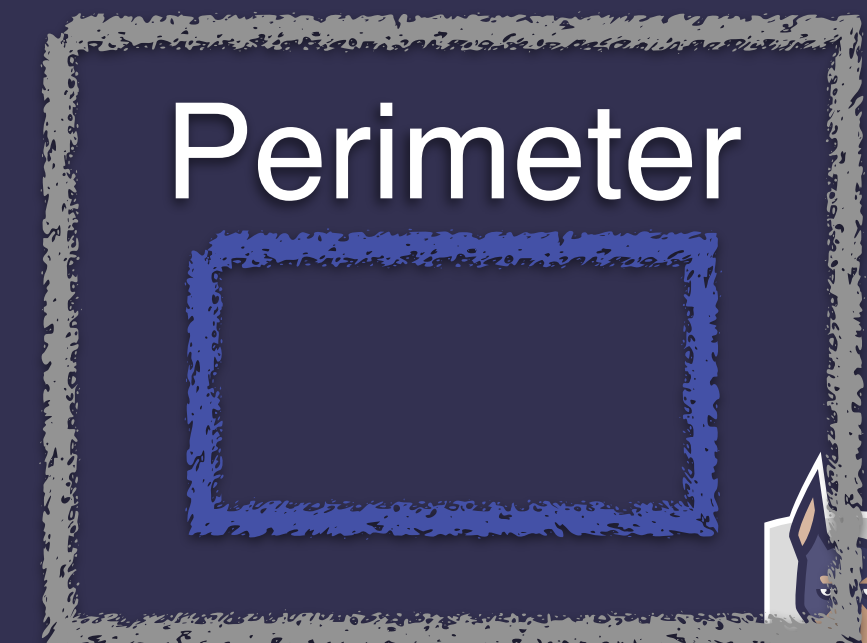
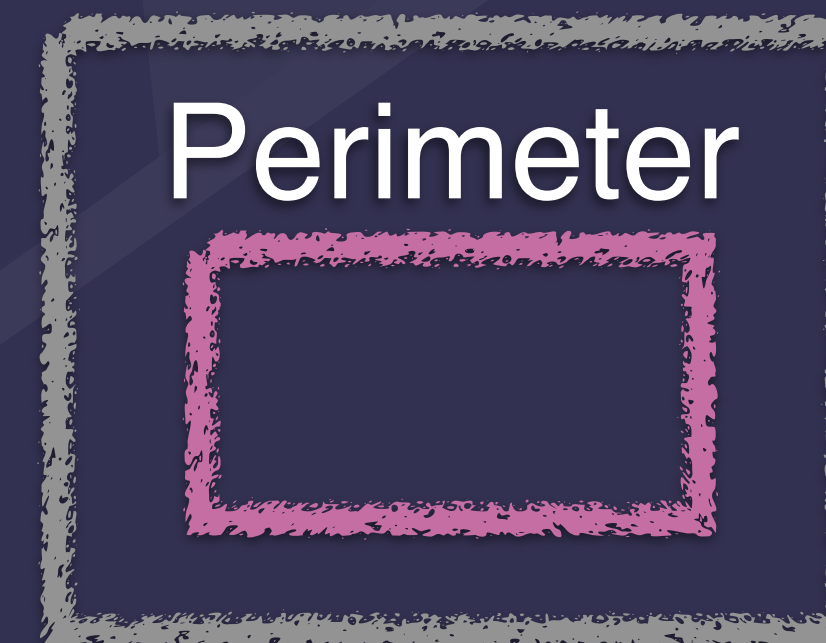
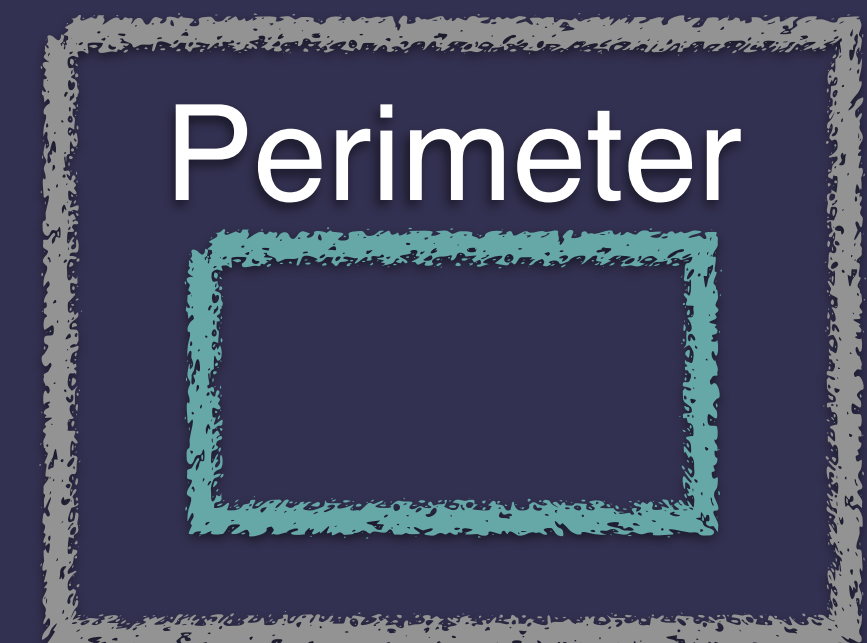
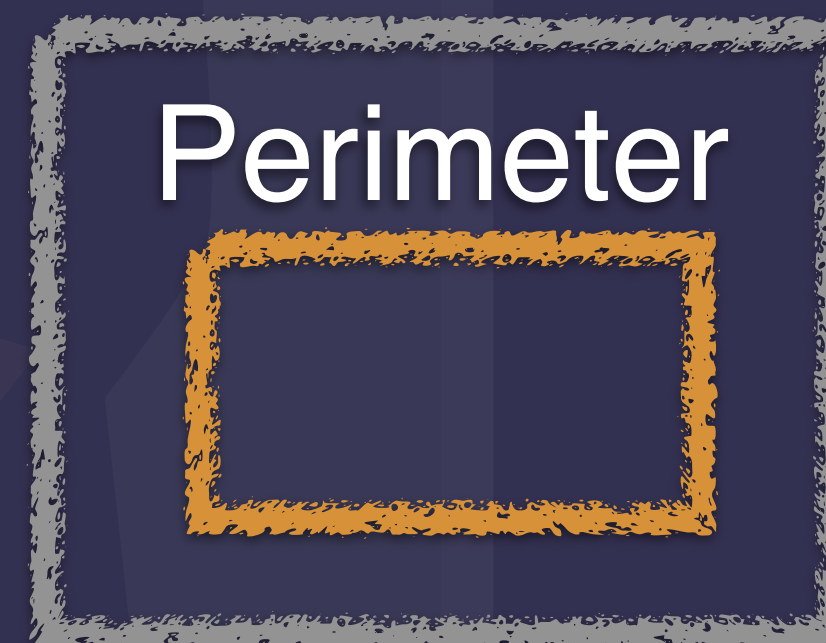
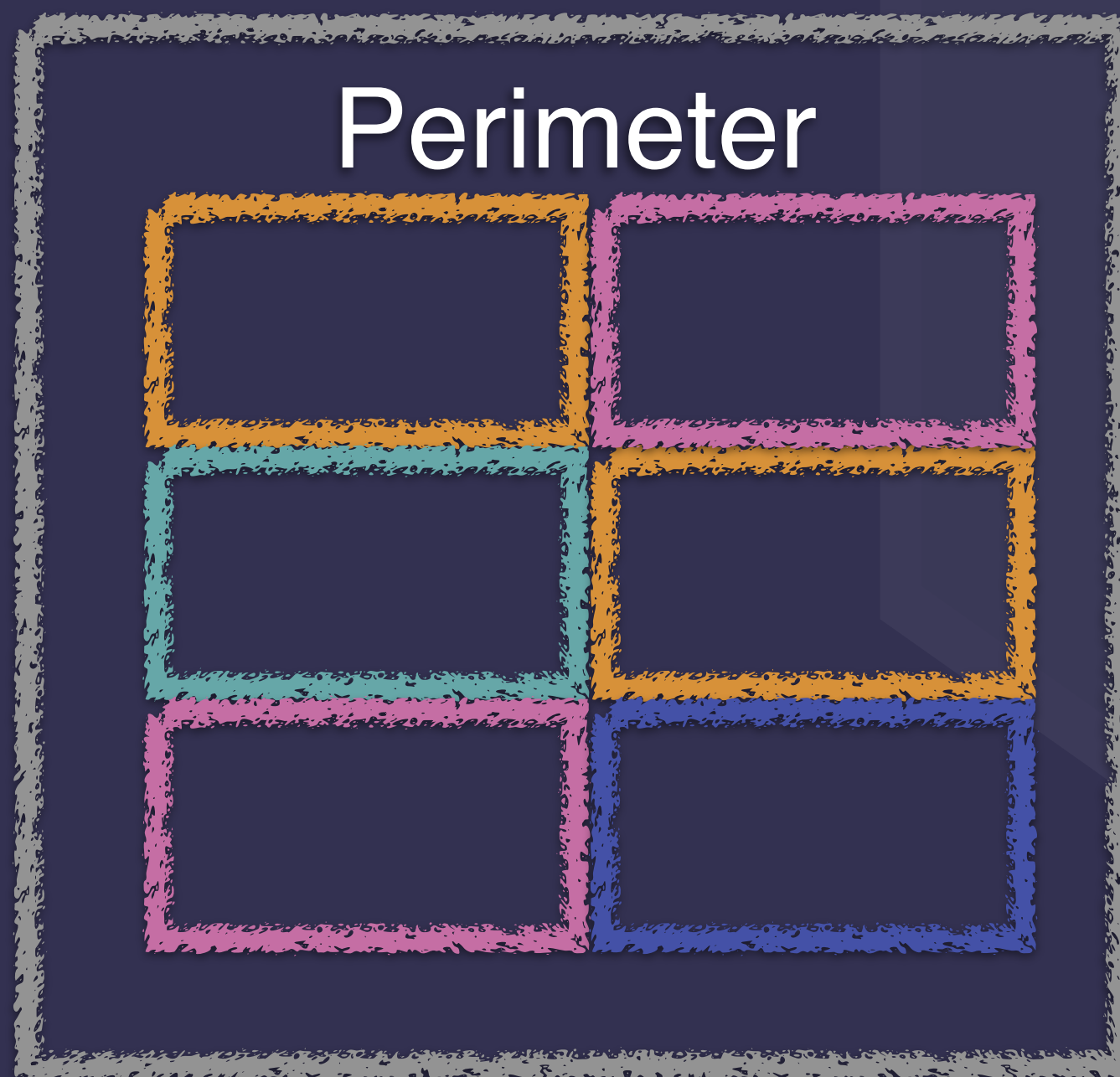
5. Use Granular Policies

(only allow each function its minimum permissions)

```
iamRoleStatements:  
- Effect: Allow  
  Action:  
    - dynamodb:Query  
    - dynamodb:Scan  
    - dynamodb:GetItem  
    - dynamodb:PutItem  
    - dynamodb:UpdateItem  
    - dynamodb>DeleteItem  
  Resource: "arn:aws:dynamodb:${opt:region,aws}:  
- Effect: Allow  
  Action:  
    - lambda:InvokeFunction  
    - lambda:InvokeAsync  
  Resource: "*"
```

A function is a perimeter

That needs to be secured



Immutability



6. Don't rely on immutability

(Lambda - and others - reuse servers)

```
total 20
-rw-rw-r-- 1 sbx_user1060 486 1859 Jul  6 21:41 goof-todos.0.36851305747404695
-rw-rw-r-- 1 sbx_user1060 486 1859 Jul  6 21:41 goof-todos.0.44848913163878024
-rw-rw-r-- 1 sbx_user1060 486 1859 Jul  6 21:41 goof-todos.0.4663676058407873
-rw-rw-r-- 1 sbx_user1060 486 1859 Jul  6 21:41 goof-todos.0.5816180012188852
-rw-rw-r-- 1 sbx_user1060 486 1859 Jul  6 21:41 goof-todos.0.7717393657658249
-----06c5f7ccc07c2625-----
```

Serverless user is typically **Low Privilege**

Reducing impact substantially, but not eliminating it

7. Worry about *all* functions

(Every available function increases your attack surface)

Which AWS Lambda functions do you want to test?

Snyk supports Node, Ruby and Java projects. We require a `package.json`, a `Gemfile.lock` or a `pom.xml` to be able to test your function.

us-east-1

us-east-1

- | | |
|---|---|
| <input type="checkbox"/> aws-node-twilio-dev-sendText | <input type="checkbox"/> serverless-goof-dev-internalRestore |
| <input type="checkbox"/> \$LATEST | <input type="checkbox"/> \$LATEST |
| <input type="checkbox"/> serverless-rest-api-with-dynamodb-dev-create | <input type="checkbox"/> serverless-rest-api-with-dynamodb-dev-delete |
| <input type="checkbox"/> \$LATEST | <input type="checkbox"/> \$LATEST |
| <input type="checkbox"/> serverless-goof-dev-update | <input type="checkbox"/> serverless-goof-dev-create |
| <input type="checkbox"/> \$LATEST | <input type="checkbox"/> \$LATEST |
| <input type="checkbox"/> serverless-goof-dev-render | <input type="checkbox"/> serverless-goof-dev-list |
| <input type="checkbox"/> \$LATEST | <input type="checkbox"/> \$LATEST |
| <input type="checkbox"/> serverless-goof-dev-get | <input type="checkbox"/> vulnerable-func-dev-currentTime |
| <input type="checkbox"/> \$LATEST | <input type="checkbox"/> \$LATEST |
| <input type="checkbox"/> serverless-goof-dev-internalBackup | <input type="checkbox"/> prod |
| <input type="checkbox"/> \$LATEST | <input type="checkbox"/> aws-autho-protected-endpoints-dev-auth |
| <input type="checkbox"/> serverless-rest-api-with-dynamodb-dev-list | <input type="checkbox"/> \$LATEST |

Security in Serverless

Better

Vulnerable OS Dependencies

Denial of Service

Long-lived Compromised Servers

Neutral

Permissions

Securing Data at rest

Vulnerabilities in your code

Vulnerable App Dependencies

Worse

Third Party Services

Attack Surface

Security Monitoring

Serverless is **defined now.**
Let's **build Security in.**

Thank You!

Don't forget:
OSS Security AMA
2:55pm, Waterfront CDE

Guy Podjarny, Snyk
[@guypod](#)