Orchestrating Security Tooling With AWS Step Functions

Background

Jules Denardou

@Pod_Sec

@JulesDT

Justin Massey

@jmassey09

@th3r3p0

- Security Engineers at Datadog
- Product Security team
- Improve security of product without detrimental impact to the developers workflows


SECURE
ALL THE THINGS!

Goals:

Developers own the security of their app

Goals:

Developers own the security of their app

Find bugs early. Fix bugs early.

Security Tooling and CI/CD
A Love/Hate Relationship

Security tooling and CI/CD

- Security tools often don’t integrate with CI/CD
  - Long running jobs
  - False positives
  - Each tool has its own way of giving a report

Security tooling and CI/CD

- Security tools often don’t integrate with CI/CD
  - Long running jobs
  - False positives
  - Each tool has its own way of giving a report

Security tooling and CI/CD

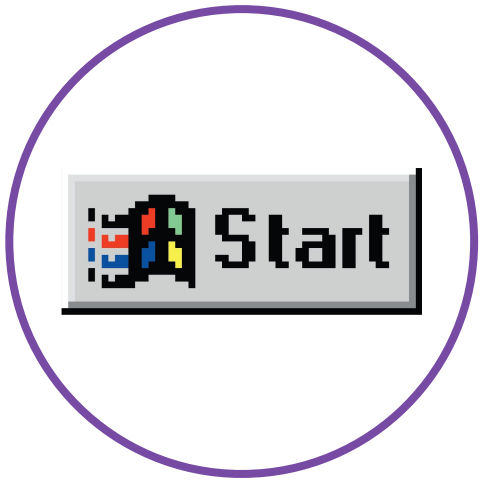
- Security tools often don’t integrate with Dev Workflows
  - Blocking deployment pipelines
  - Reports in a third party website

Security tooling and CI/CD

- Security tools often don’t integrate with Dev Workflows
  - Blocking deployment pipelines
  - Reports in a third party website

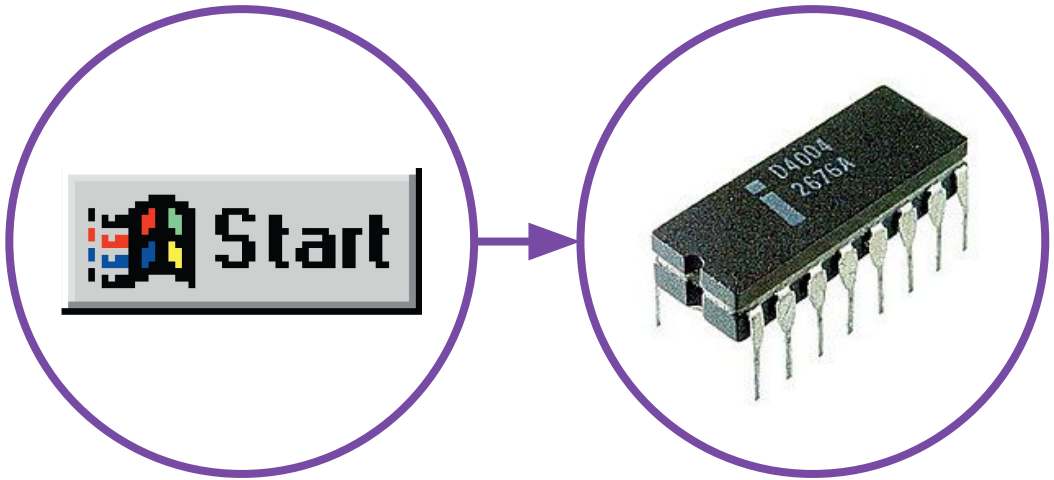
Plugin Based Workflow

Plugin Based Workflow

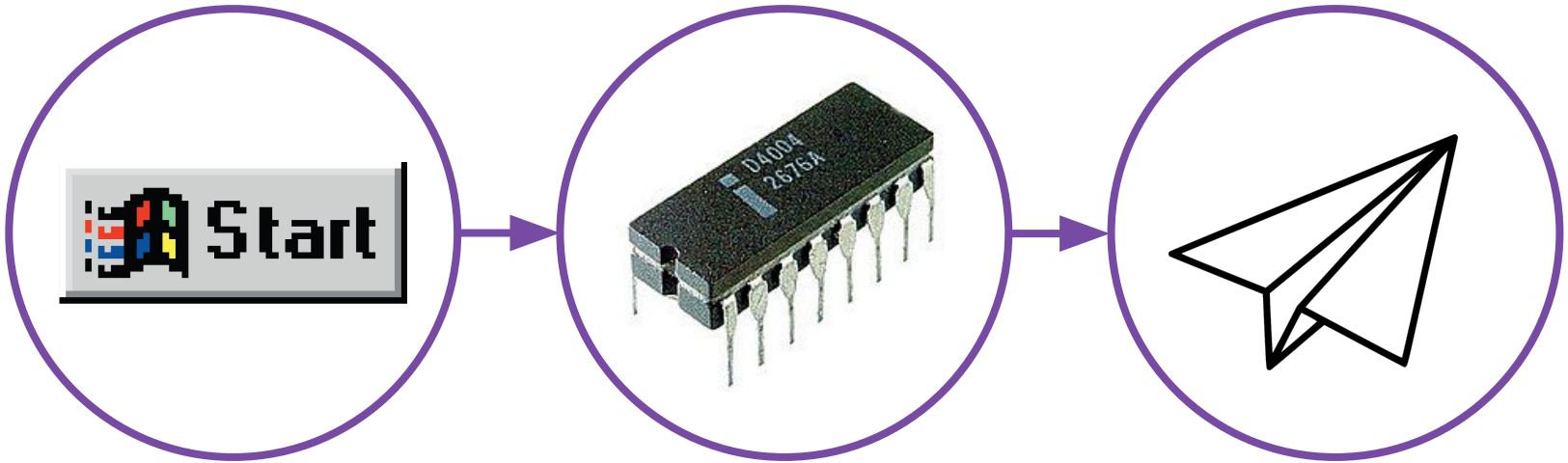


Triggers

Plugin Based Workflow



Plugin Based Workflow

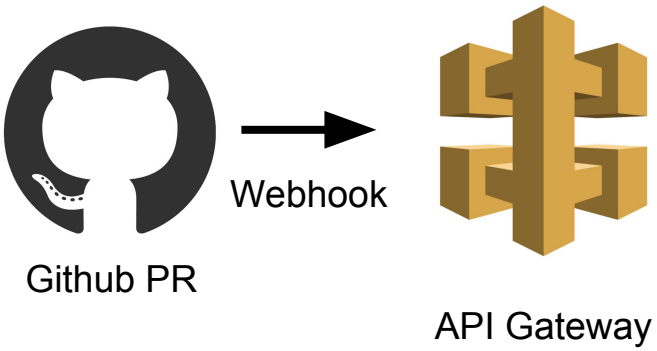


Our Solution

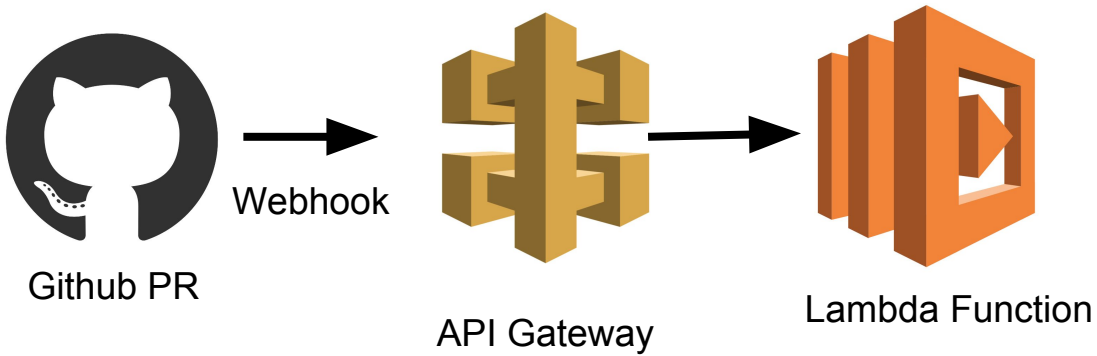
Our Solution

Github PR

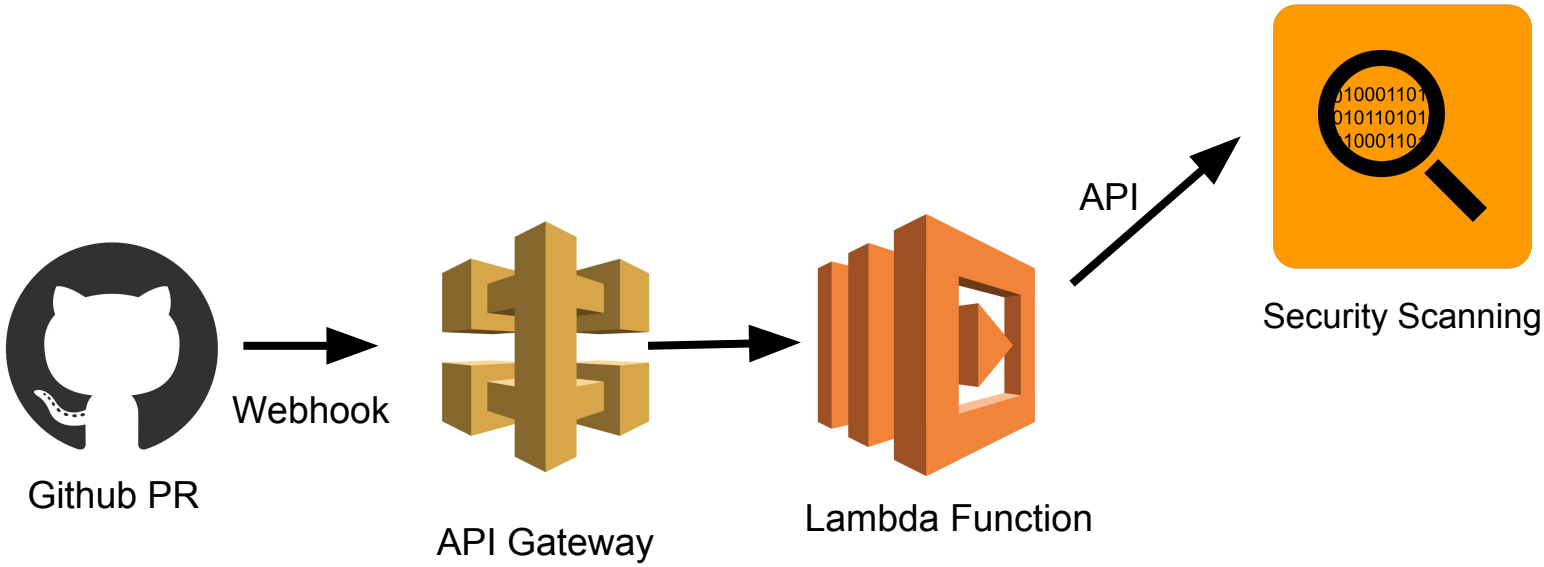
Our Solution



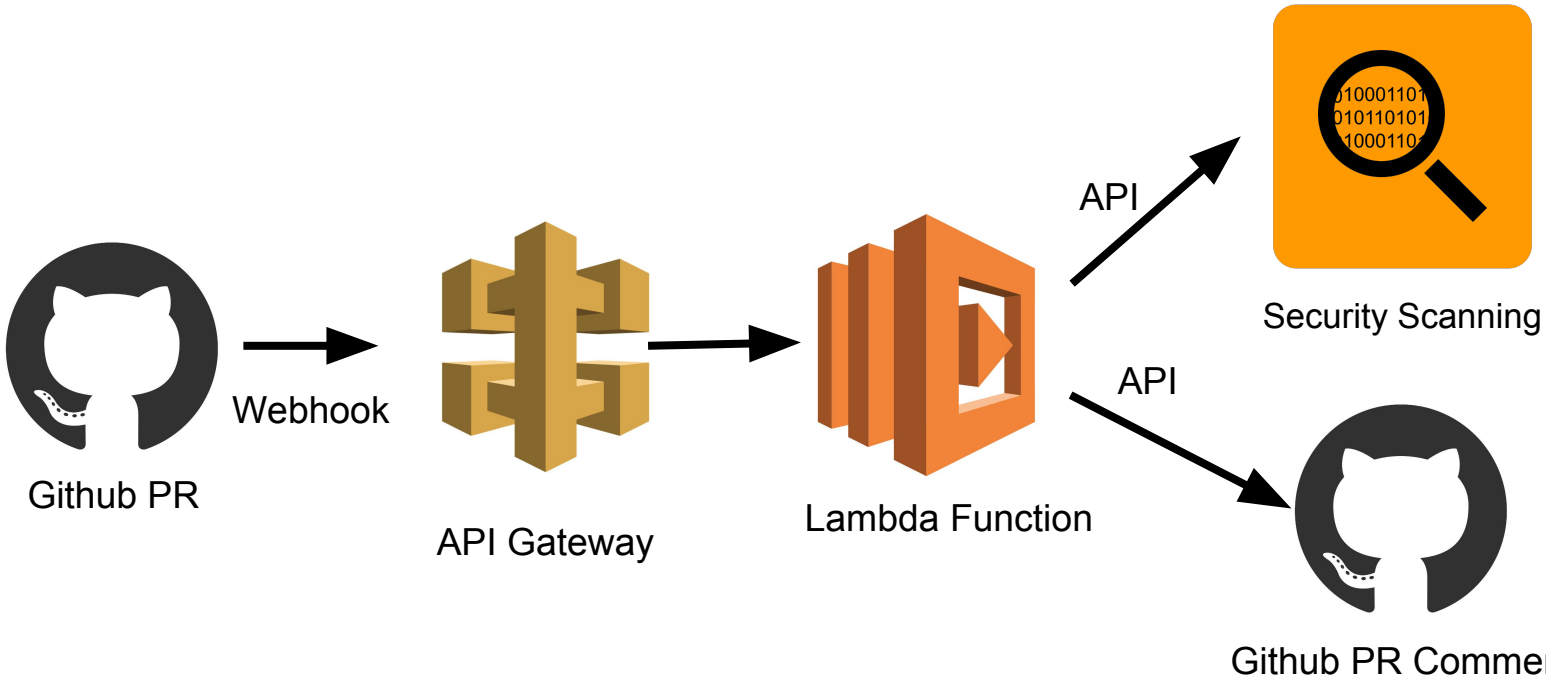
Our Solution



Our Solution



Our Solution



Issues With This Solution

Time

- Some scans take > 30 min
- Lambdas timeout out after 300 seconds

Issues with 1st Solution

Time

Scan Time	05h:11m:55s
Lines Of Code Scanned	658903
Files Scanned	3779

Issues with 1st Solution

Time

- Some scans take > 30 min
- Lambdas timeout out after 300 seconds

Notifications

- Some scans have no way to notify you when complete

Design

- First implementation

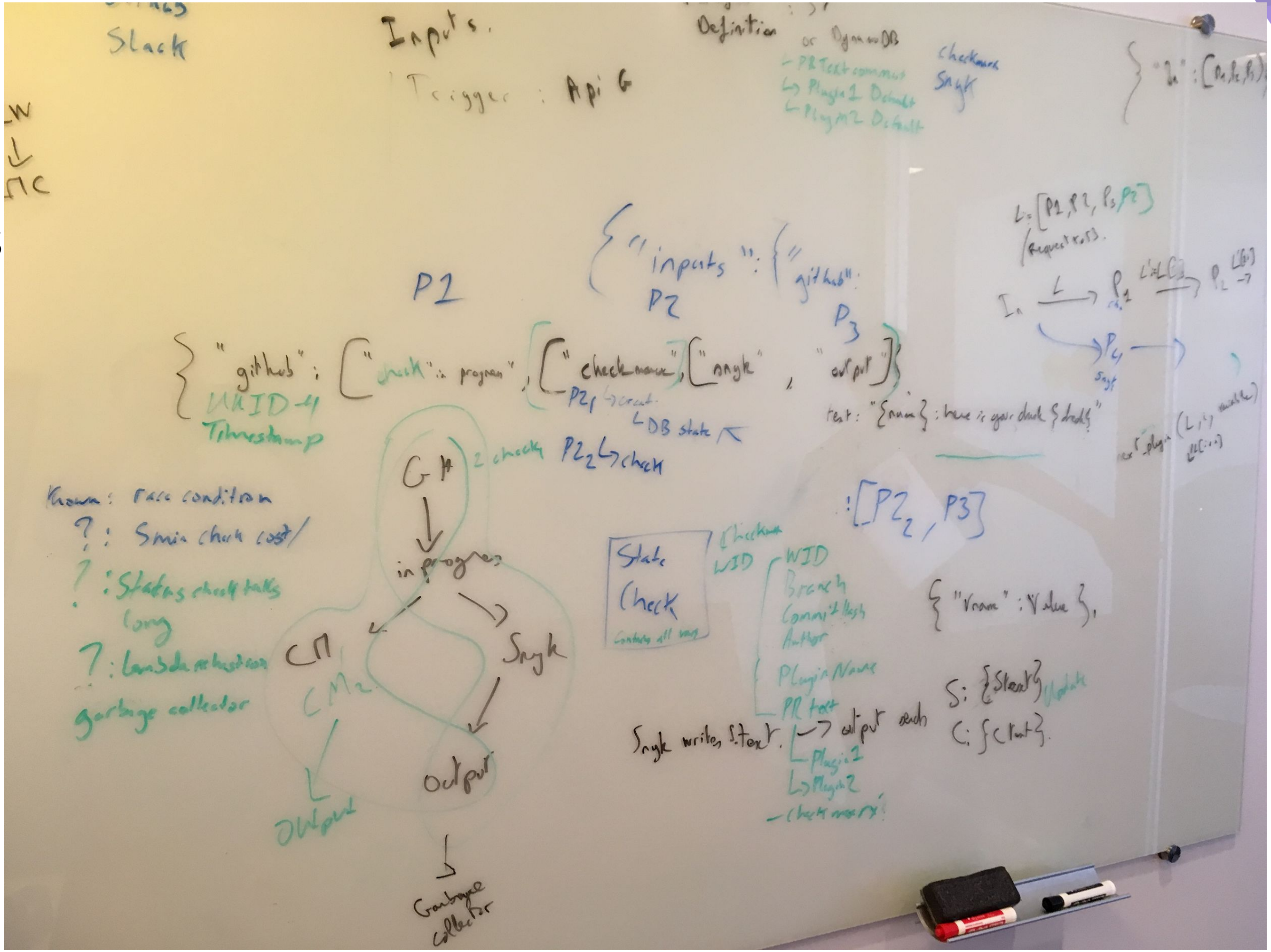
Design

- First implementation

AWS Lambdas

Des

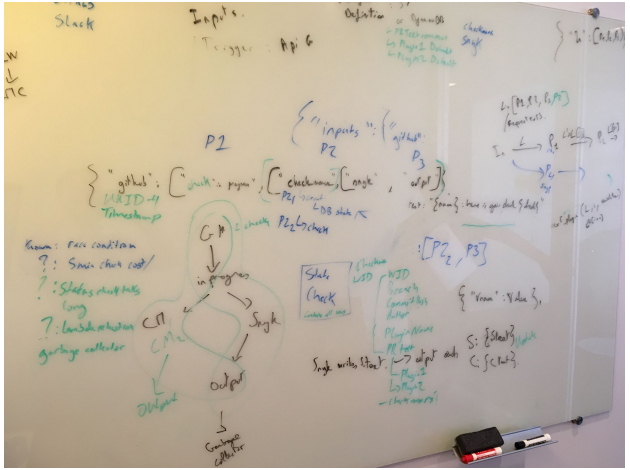
- Firs

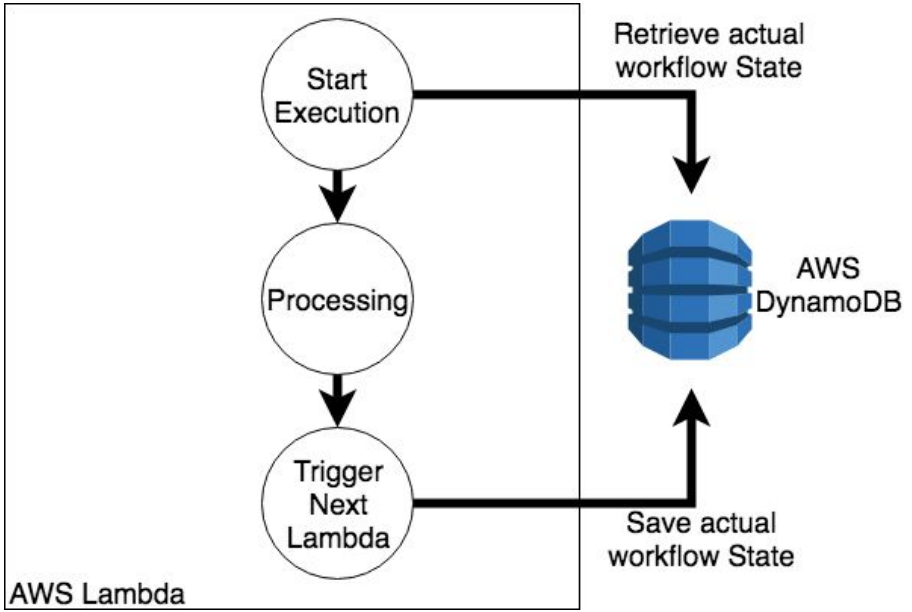


Slack
Inputs.
Trigger : Api G
Definition
P1
P2
State
Check
Known : race condition
garbage collector
in progress
Snyk
output
Garbage collector

Design

- First implementation
  - AWS Lambdas





Design

- Alternative: AWS Step Functions

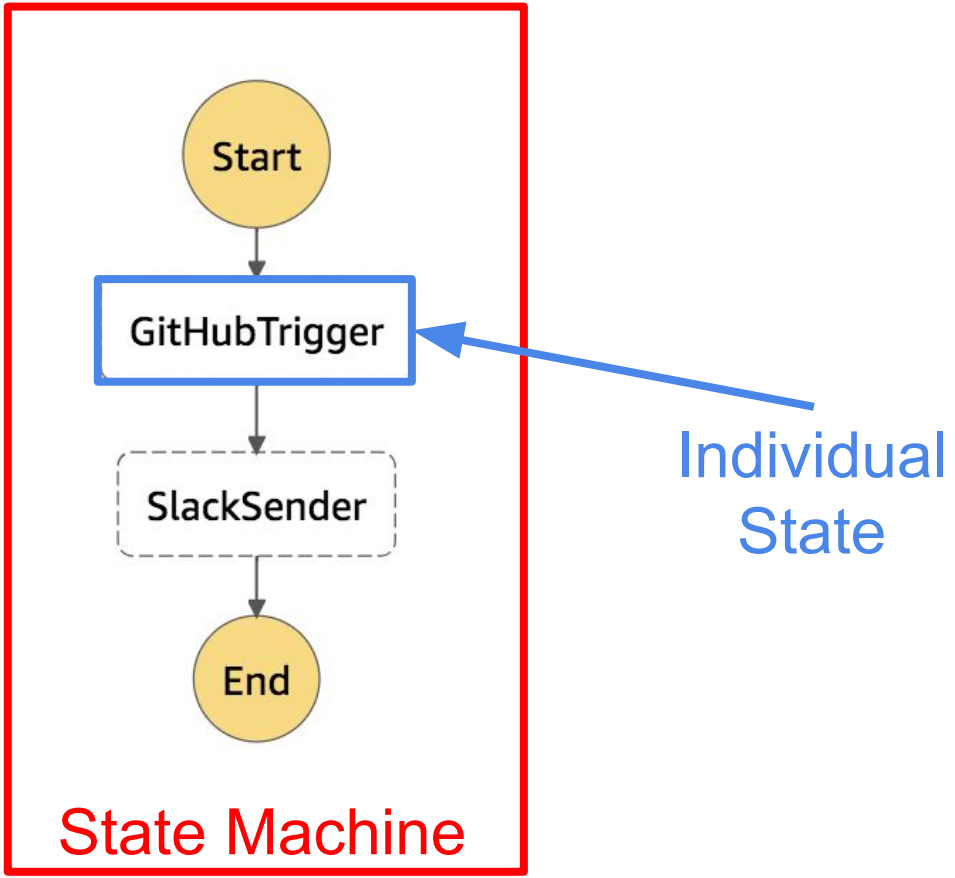
Design

- Alternative: AWS Step Functions
  - State Machine
    - Amazon States Language
  - Individual States
  - Step Function console = State Machine GUI

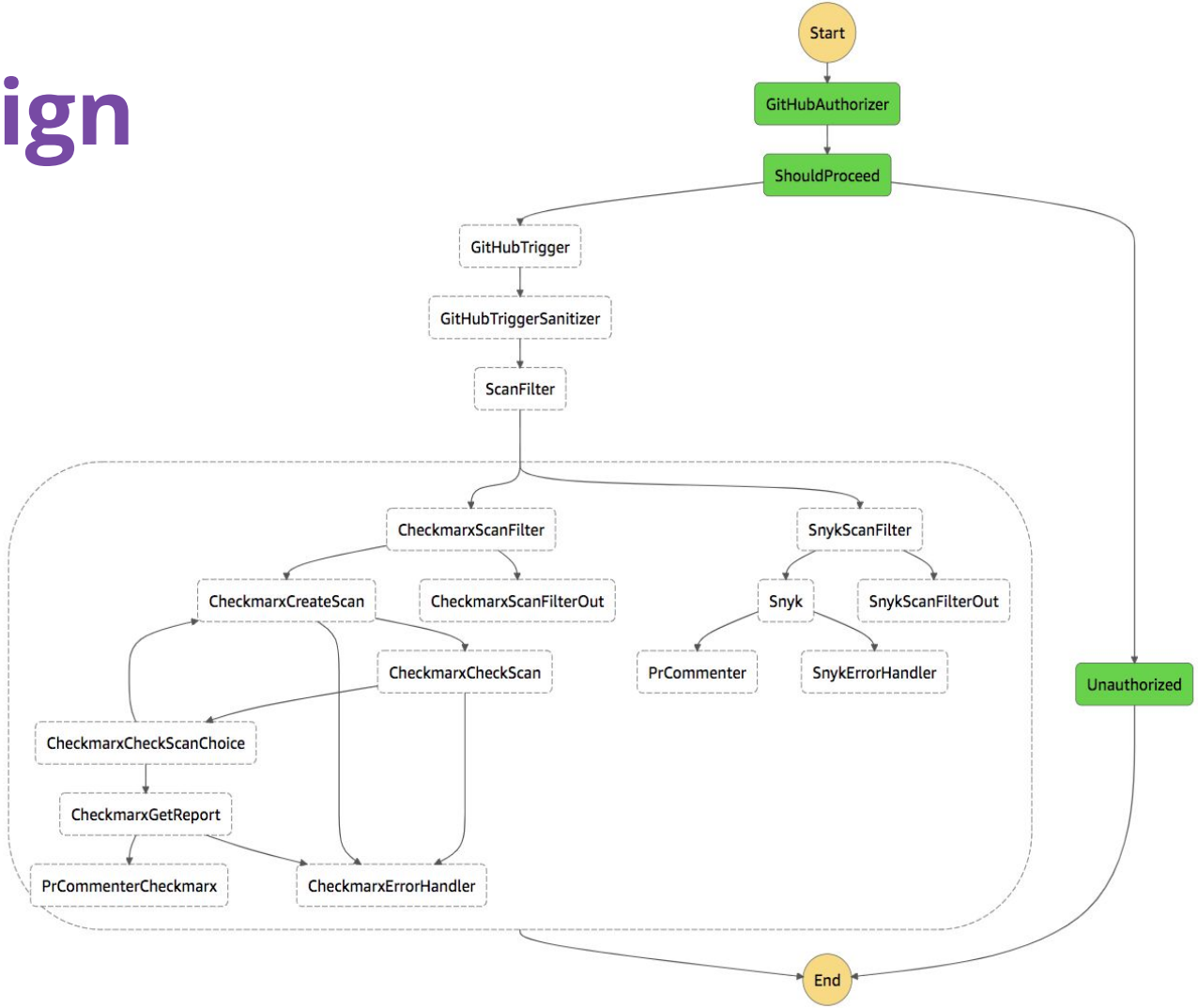
Design

- States
  - Task
  - Choice
  - Pass
  - Delay
  - Parallel
  - Success or Failure

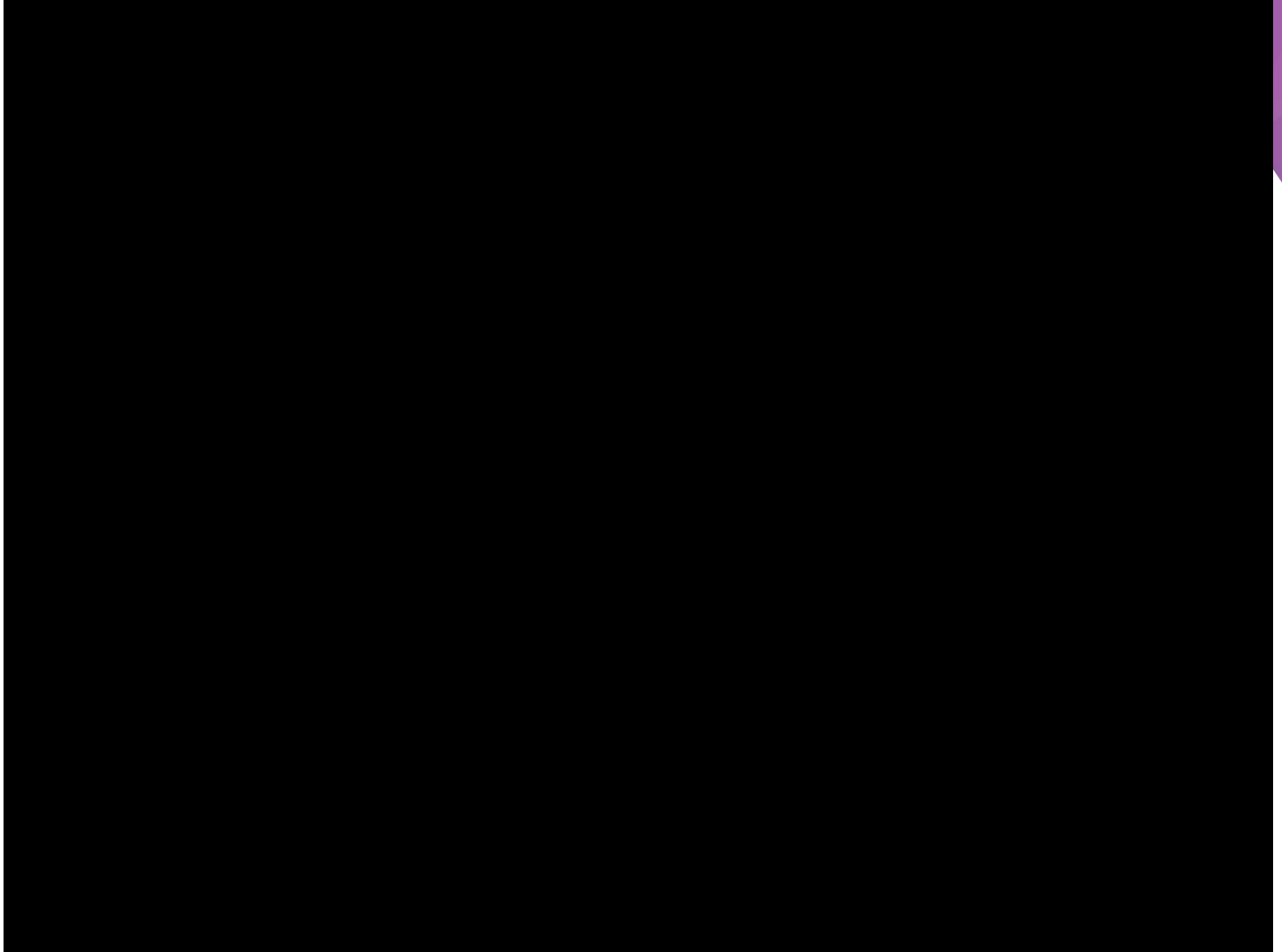
Design


Start
GitHubTrigger
SlackSender
End
State Machine
Individual State

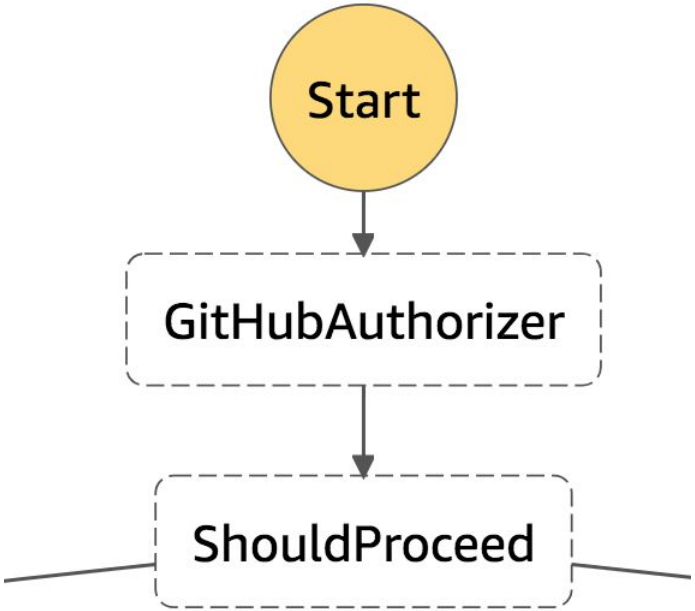
Design

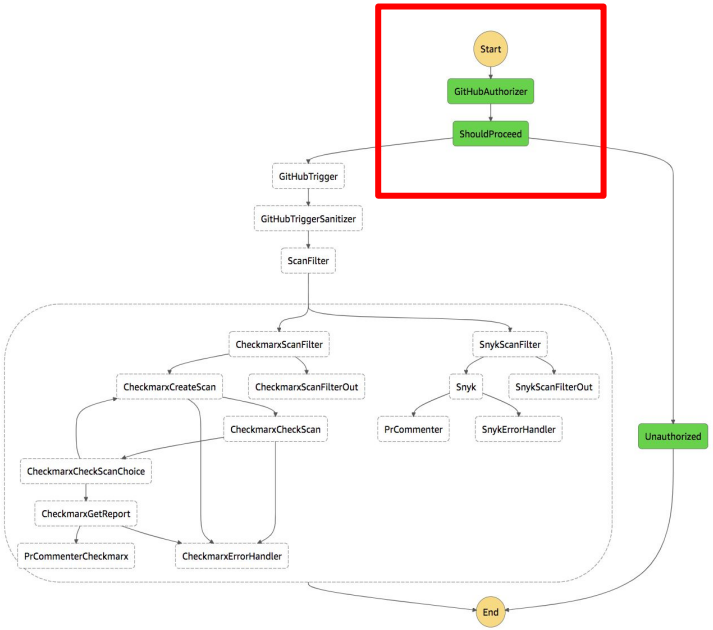

Start
GitHubAuthorizer
ShouldProceed
GitHubTrigger
GitHubTriggerSanitizer
ScanFilter
CheckmarxScanFilter
SnykScanFilter
CheckmarxCreateScan
CheckmarxScanFilterOut
Snyk
SnykScanFilterOut
CheckmarxCheckScan
PrCommenter
SnykErrorHandler
Unauthorized
CheckmarxCheckScanChoice
CheckmarxGetReport
PrCommenterCheckmarx
CheckmarxErrorHandler
End

Demo !



Design


Start
GitHubAuthorizer
ShouldProceed


Start
GitHubAuthorizer
ShouldProceed
GitHubTrigger
GitHubTriggerSanitizer
ScanFilter
CheckmarxScanFilter
SnykScanFilter
CheckmarxCreateScan
CheckmarxScanFilterOut
Snyk
SnykScanFilterOut
CheckmarxCheckScan
PrCommenter
SnykErrorHandler
Unauthorized
CheckmarxCheckScanChoice
CheckmarxGetReport
PrCommenterCheckmarx
CheckmarxErrorHandler
End

Design

- Integrating Github & AWS: Incompatible by default

Design

- Integrating Github & AWS: Incompatible by default
  - Github uses HMAC signed events

`X-Hub-Signature`	The HMAC hex digest of the response body. This header will be sent if the webhook is configured with a `secret`. The HMAC hex digest is generated using the `sha1` hash function and the `secret` as the HMAC `key`.

Design

- Integrating Github & AWS: Incompatible by default
    - Github uses HMAC signed events
    - AWS API Gateway uses an “Authorizer”

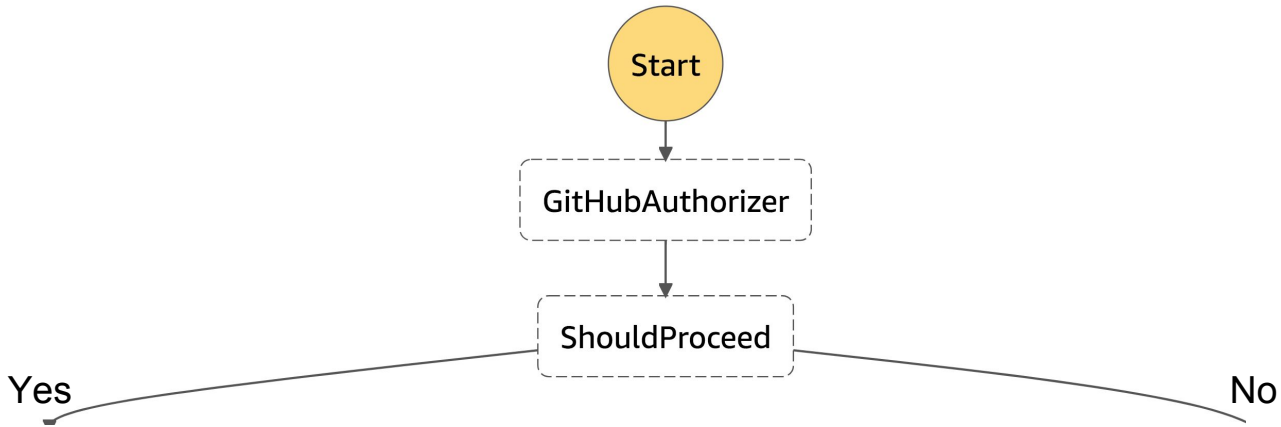
Types of API Gateway Lambda Authorizers

API Gateway supports Lambda authorizers of the `TOKEN` and `REQUEST` types:

- Lambda authorizers of the `TOKEN` type grant a caller permissions to invoke a given request using an authorization token **passed in a header**. The token could be, for example, an OAuth token.
- Lambda authorizers of the `REQUEST` type grant a caller permissions to invoke a given request **using request parameters, including headers, query strings, stage variables, or context parameters.**

Design

- Integrating Github & AWS: Incompatible by default
    - Implement a custom authorizer as a first step of the Step function

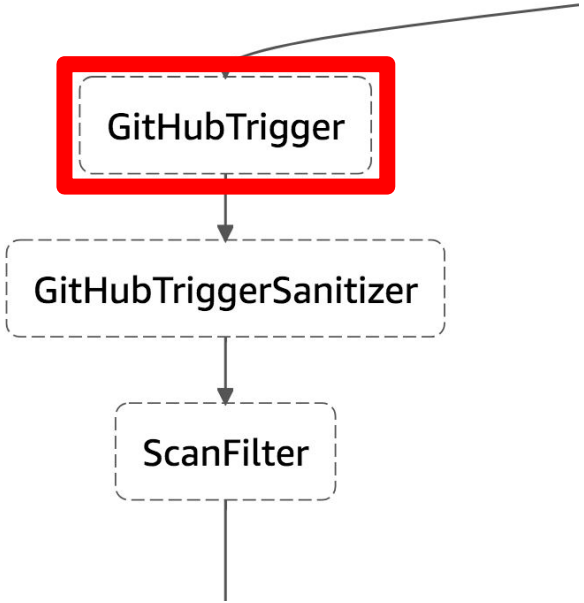


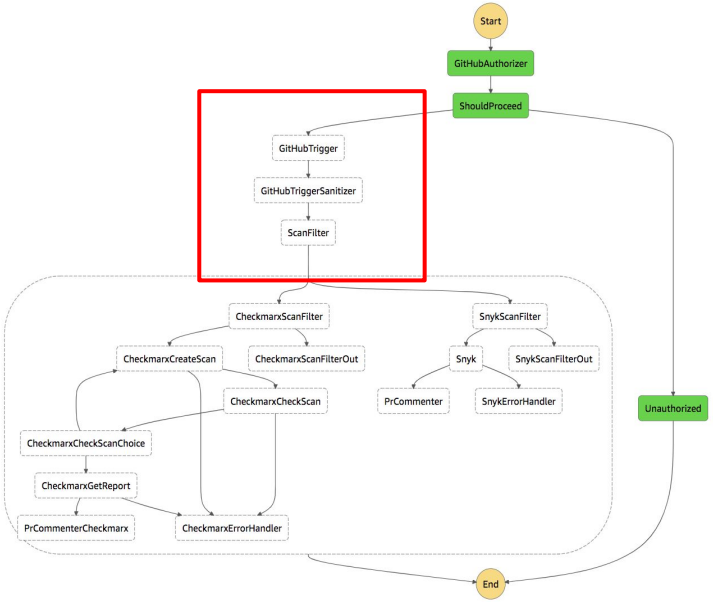
Design

- Integrating Github & AWS: Incompatible by default
    - Implement a custom authorizer as a first step of the Step function

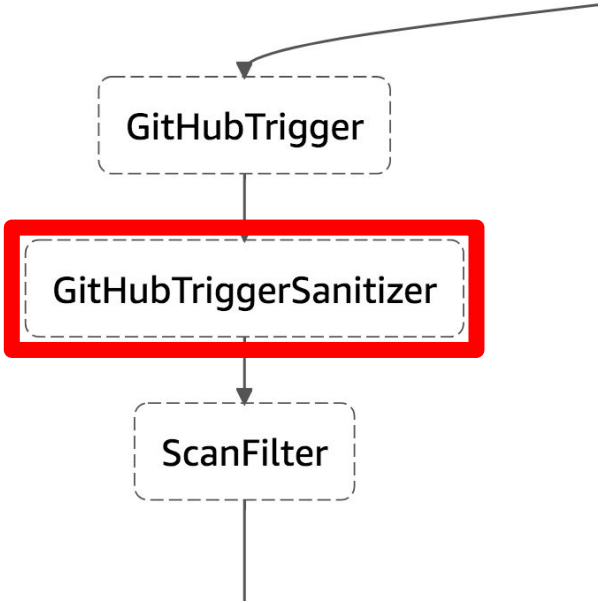
```
signature = hmac.new(github_webhook_secret, binascii.a2b_base64(event['body']), hashlib.sha1)
x_hub_signature = event['headers']['X-Hub-Signature'].split('=')[-1]
if not hmac.compare_digest(signature.hexdigest(), x_hub_signature):
    log.error("Digest not valid, GitHub didn't sent this payload")
    datadog_manager.send_metric("count", "prodsec_middleware.authorization.invalid", 1)
    return False
datadog_manager.send_metric("count", "prodsec_middleware.authorization.valid", 1)
return True
```

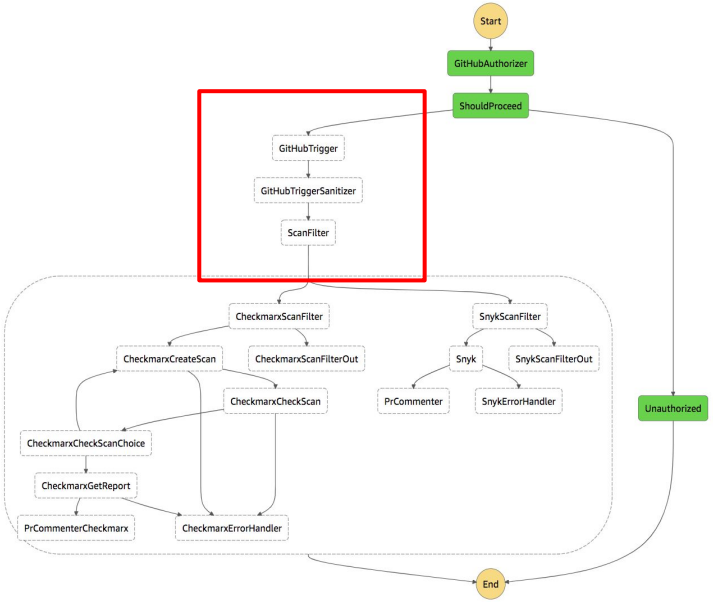
Design


GitHubTrigger
GitHubTriggerSanitizer
ScanFilter

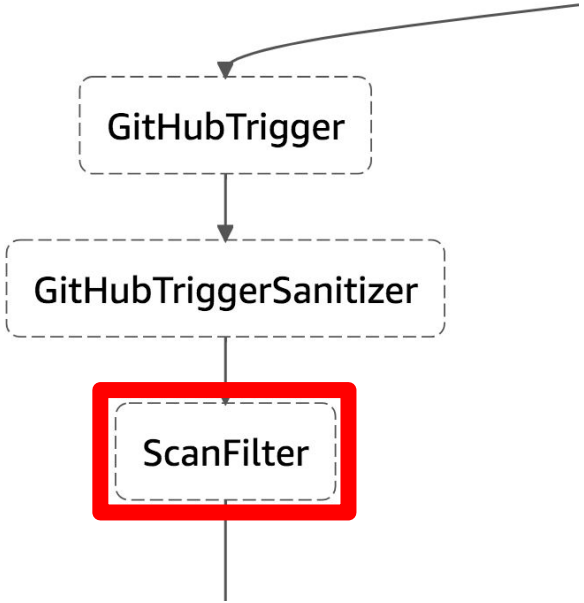

Start
GitHubAuthorizer
ShouldProceed
GitHubTrigger
GitHubTriggerSanitizer
ScanFilter
CheckmarxScanFilter
SnykScanFilter
CheckmarxCreateScan
CheckmarxScanFilterOut
Snyk
SnykScanFilterOut
CheckmarxCheckScan
PrCommenter
SnykErrorHandler
Unauthorized
CheckmarxCheckScanChoice
CheckmarxGetReport
PrCommenterCheckmarx
CheckmarxErrorHandler
End

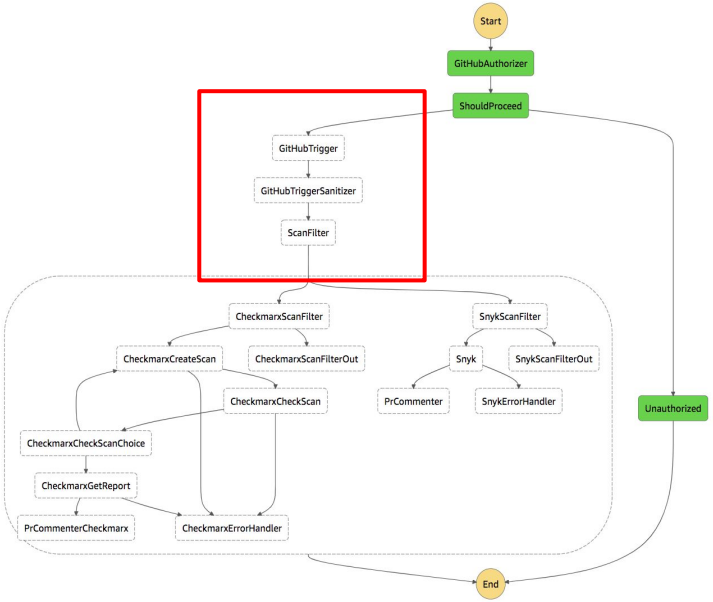
Design


GitHubTrigger
GitHubTriggerSanitizer
ScanFilter

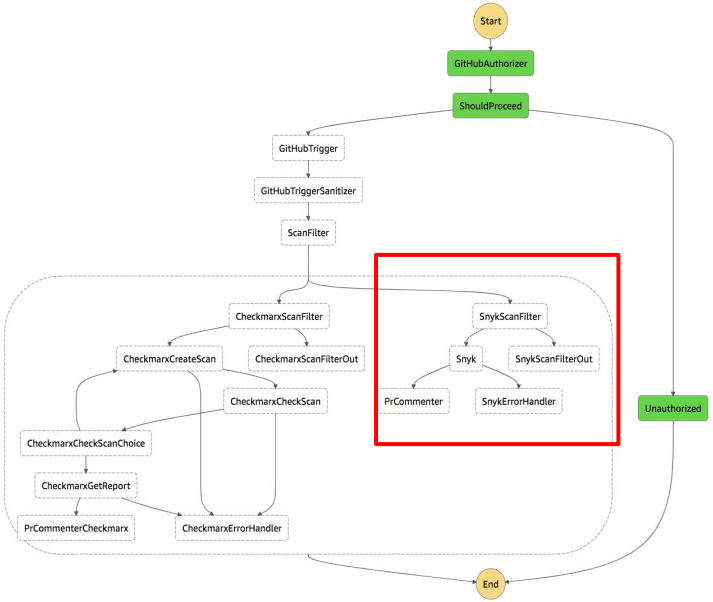

Start
GitHubAuthorizer
ShouldProceed
GitHubTrigger
GitHubTriggerSanitizer
ScanFilter
CheckmarxScanFilter
SnykScanFilter
CheckmarxCreateScan
CheckmarxScanFilterOut
Snyk
SnykScanFilterOut
CheckmarxCheckScan
PrCommenter
SnykErrorHandler
Unauthorized
CheckmarxCheckScanChoice
CheckmarxGetReport
PrCommenterCheckmarx
CheckmarxErrorHandler
End

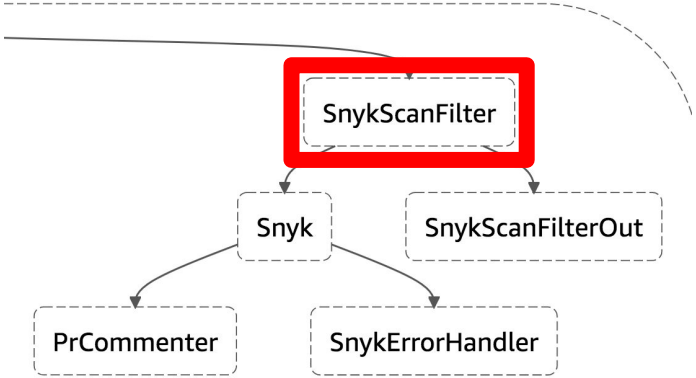
Design


GitHubTrigger
GitHubTriggerSanitizer
ScanFilter

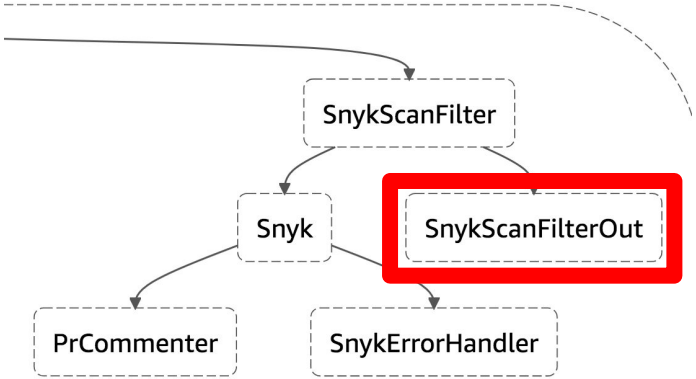

Start
GitHubAuthorizer
ShouldProceed
GitHubTrigger
GitHubTriggerSanitizer
ScanFilter
CheckmarxScanFilter
SnykScanFilter
CheckmarxCreateScan
CheckmarxScanFilterOut
Snyk
SnykScanFilterOut
CheckmarxCheckScan
PrCommenter
SnykErrorHandler
Unauthorized
CheckmarxCheckScanChoice
CheckmarxGetReport
PrCommenterCheckmarx
CheckmarxErrorHandler
End

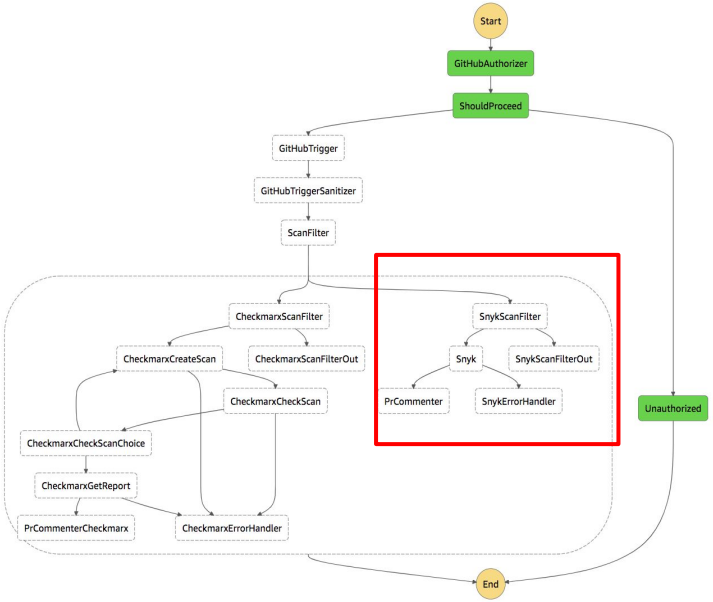
Design


Start
GitHubAuthorizer
ShouldProceed
GitHubTrigger
GitHubTriggerSanitizer
ScanFilter
CheckmarxScanFilter
SnykScanFilter
CheckmarxCreateScan
CheckmarxScanFilterOut
Snyk
SnykScanFilterOut
CheckmarxCheckScan
PrCommenter
SnykErrorHandler
Unauthorized
CheckmarxCheckScanChoice
CheckmarxGetReport
PrCommenterCheckmarx
CheckmarxErrorHandler
End

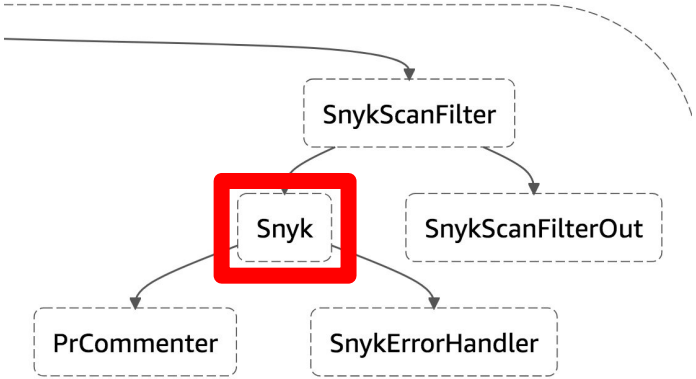

SnykScanFilter
Snyk
SnykScanFilterOut
PrCommenter
SnykErrorHandler

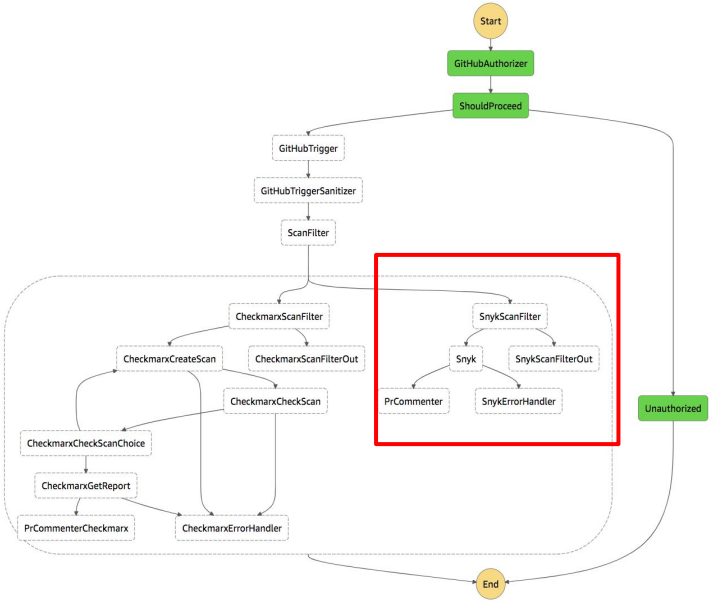
Design


SnykScanFilter
Snyk
SnykScanFilterOut
PrCommenter
SnykErrorHandler

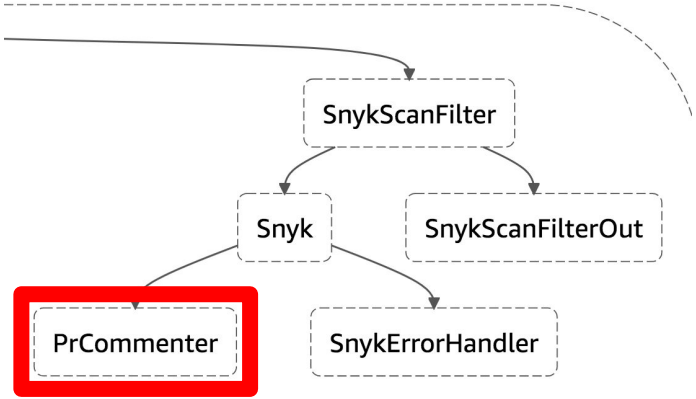

Start
GitHubAuthorizer
ShouldProceed
GitHubTrigger
GitHubTriggerSanitizer
ScanFilter
CheckmarxScanFilter
SnykScanFilter
CheckmarxCreateScan
CheckmarxScanFilterOut
Snyk
SnykScanFilterOut
CheckmarxCheckScan
PrCommenter
SnykErrorHandler
Unauthorized
CheckmarxCheckScanChoice
CheckmarxGetReport
PrCommenterCheckmarx
CheckmarxErrorHandler
End

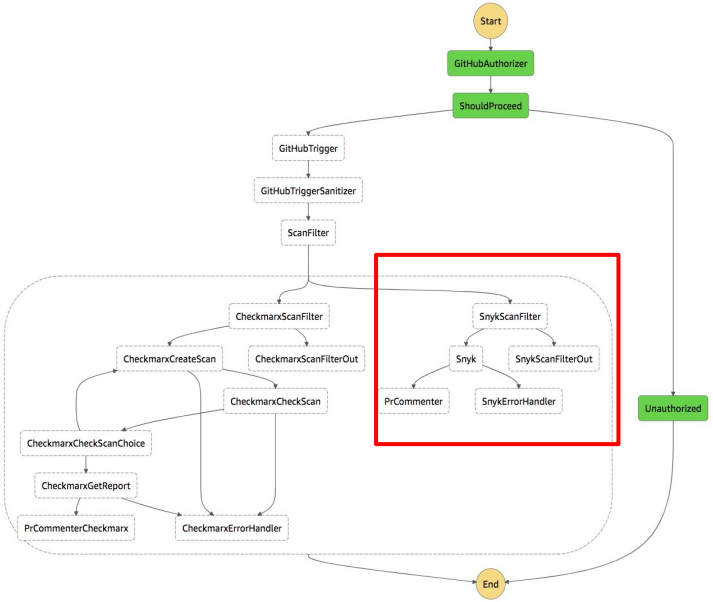
Design


SnykScanFilter
Snyk
SnykScanFilterOut
PrCommenter
SnykErrorHandler

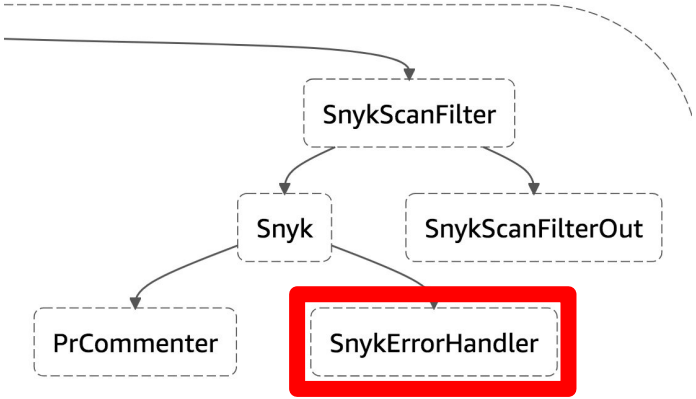

Start
GitHubAuthorizer
ShouldProceed
GitHubTrigger
GitHubTriggerSanitizer
ScanFilter
CheckmarxScanFilter
SnykScanFilter
CheckmarxCreateScan
CheckmarxScanFilterOut
Snyk
SnykScanFilterOut
CheckmarxCheckScan
PrCommenter
SnykErrorHandler
Unauthorized
CheckmarxCheckScanChoice
CheckmarxGetReport
PrCommenterCheckmarx
CheckmarxErrorHandler
End

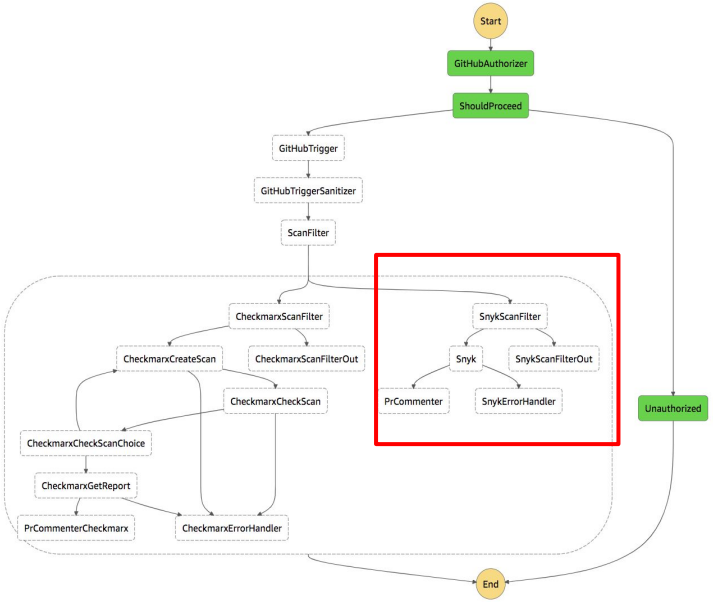
Design


SnykScanFilter
Snyk
SnykScanFilterOut
PrCommenter
SnykErrorHandler

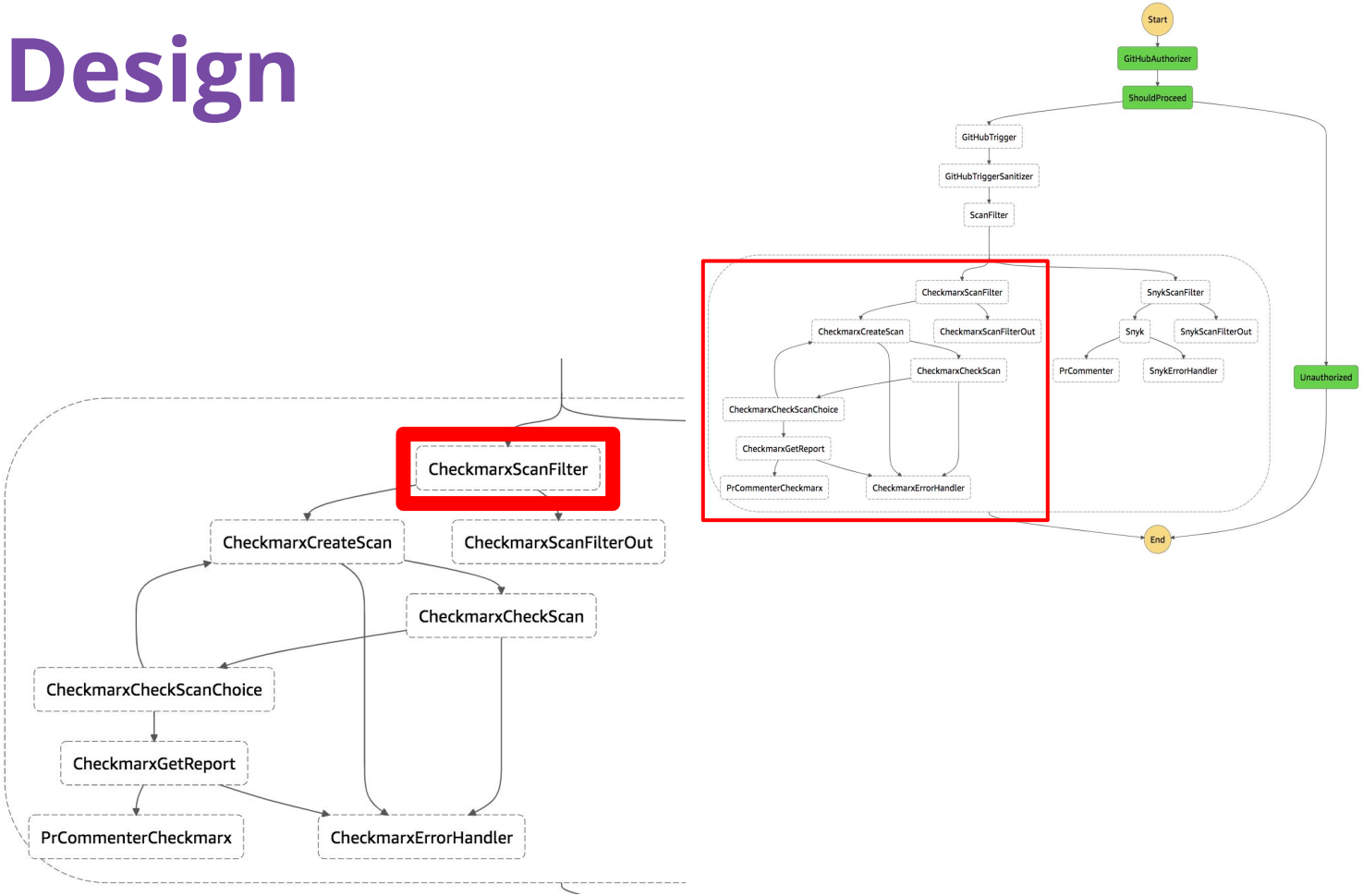

Start
GitHubAuthorizer
ShouldProceed
GitHubTrigger
GitHubTriggerSanitizer
ScanFilter
CheckmarxScanFilter
SnykScanFilter
CheckmarxCreateScan
CheckmarxScanFilterOut
Snyk
SnykScanFilterOut
CheckmarxCheckScan
PrCommenter
SnykErrorHandler
Unauthorized
CheckmarxCheckScanChoice
CheckmarxGetReport
PrCommenterCheckmarx
CheckmarxErrorHandler
End

Design

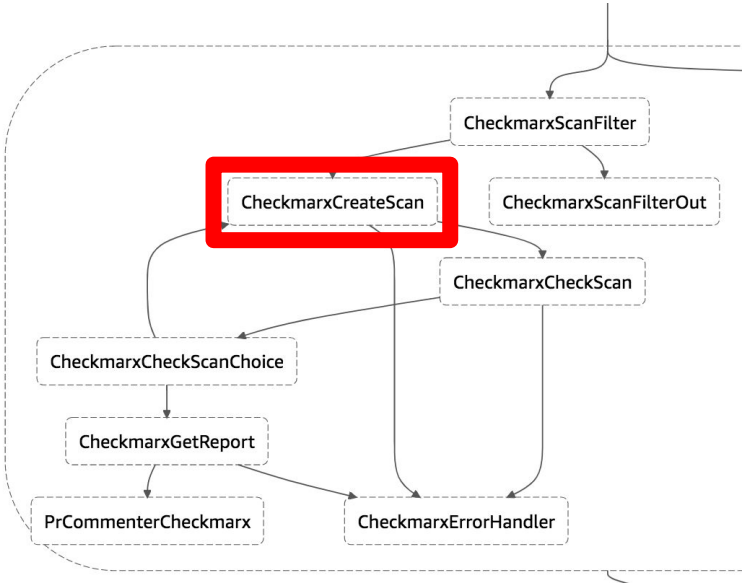

SnykScanFilter
Snyk
SnykScanFilterOut
PrCommenter
SnykErrorHandler

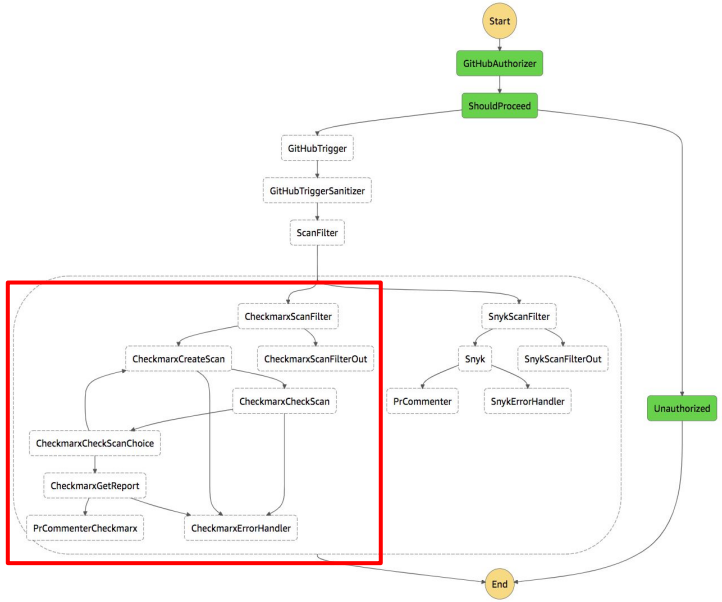

Start
GitHubAuthorizer
ShouldProceed
GitHubTrigger
GitHubTriggerSanitizer
ScanFilter
CheckmarxScanFilter
SnykScanFilter
CheckmarxCreateScan
CheckmarxScanFilterOut
Snyk
SnykScanFilterOut
CheckmarxCheckScan
PrCommenter
SnykErrorHandler
Unauthorized
CheckmarxCheckScanChoice
CheckmarxGetReport
PrCommenterCheckmarx
CheckmarxErrorHandler
End

Design

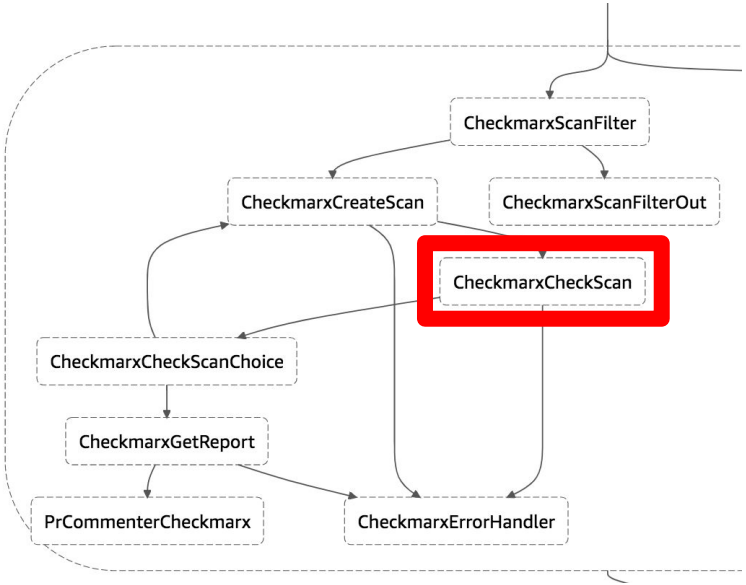

Start
GitHubAuthorizer
ShouldProceed
GitHubTrigger
GitHubTriggerSanitizer
ScanFilter
CheckmarxScanFilter
SnykScanFilter
CheckmarxCreateScan
CheckmarxScanFilterOut
Snyk
SnykScanFilterOut
CheckmarxCheckScan
PrCommenter
SnykErrorHandler
Unauthorized
CheckmarxCheckScanChoice
CheckmarxGetReport
PrCommenterCheckmarx
CheckmarxErrorHandler
End
CheckmarxScanFilter
CheckmarxCreateScan
CheckmarxScanFilterOut
CheckmarxCheckScan
CheckmarxCheckScanChoice
CheckmarxGetReport
PrCommenterCheckmarx
CheckmarxErrorHandler

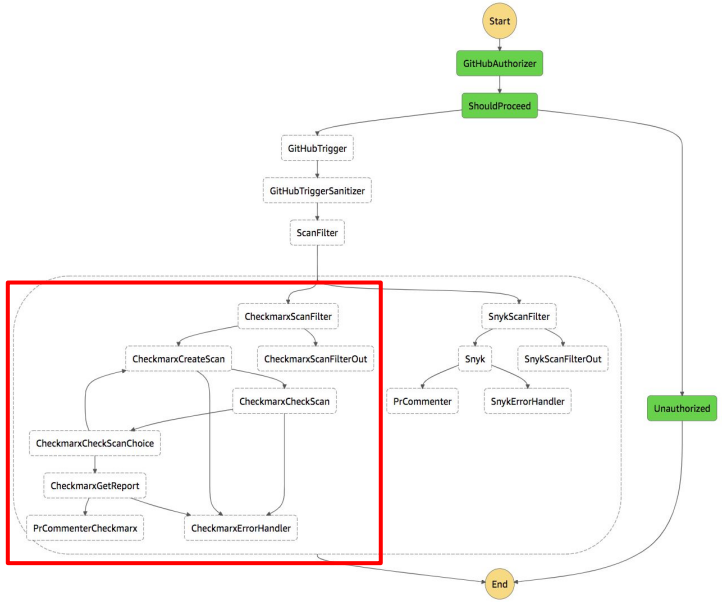
Design


CheckmarxScanFilter
CheckmarxCreateScan
CheckmarxScanFilterOut
CheckmarxCheckScan
CheckmarxCheckScanChoice
CheckmarxGetReport
PrCommenterCheckmarx
CheckmarxErrorHandler

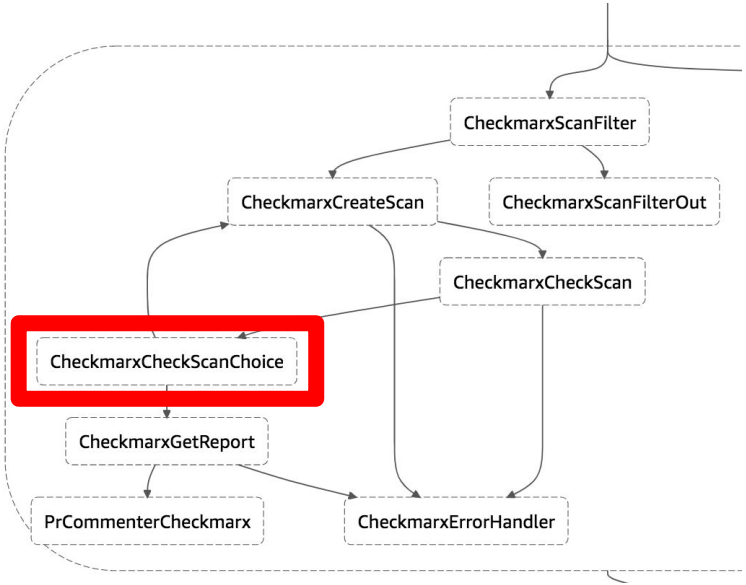

Start
GitHubAuthorizer
ShouldProceed
GitHubTrigger
GitHubTriggerSanitizer
ScanFilter
CheckmarxScanFilter
CheckmarxCreateScan
CheckmarxScanFilterOut
CheckmarxCheckScan
CheckmarxCheckScanChoice
CheckmarxGetReport
PrCommenterCheckmarx
CheckmarxErrorHandler
SnykScanFilter
Snyk
SnykScanFilterOut
PrCommenter
SnykErrorHandler
Unauthorized
End

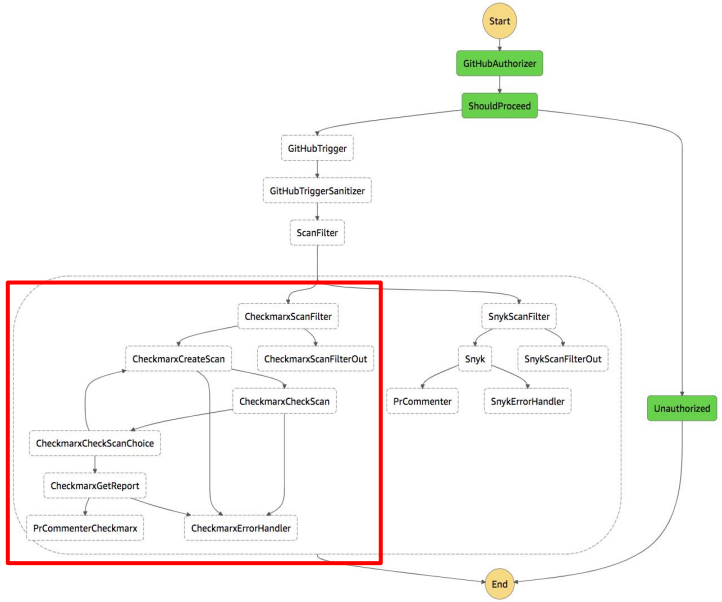
Design


CheckmarxScanFilter
CheckmarxCreateScan
CheckmarxScanFilterOut
CheckmarxCheckScan
CheckmarxCheckScanChoice
CheckmarxGetReport
PrCommenterCheckmarx
CheckmarxErrorHandler

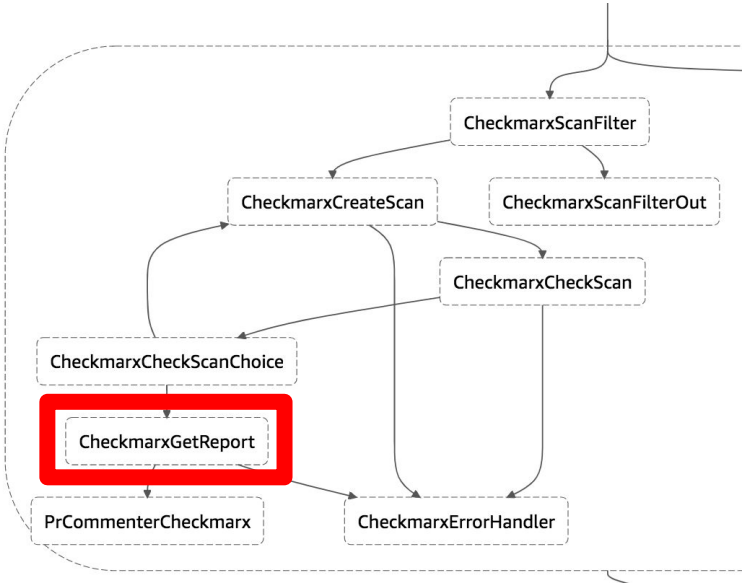

Start
GitHubAuthorizer
ShouldProceed
GitHubTrigger
GitHubTriggerSanitizer
ScanFilter
CheckmarxScanFilter
CheckmarxCreateScan
CheckmarxScanFilterOut
CheckmarxCheckScan
CheckmarxCheckScanChoice
CheckmarxGetReport
PrCommenterCheckmarx
CheckmarxErrorHandler
SnykScanFilter
Snyk
SnykScanFilterOut
PrCommenter
SnykErrorHandler
Unauthorized
End

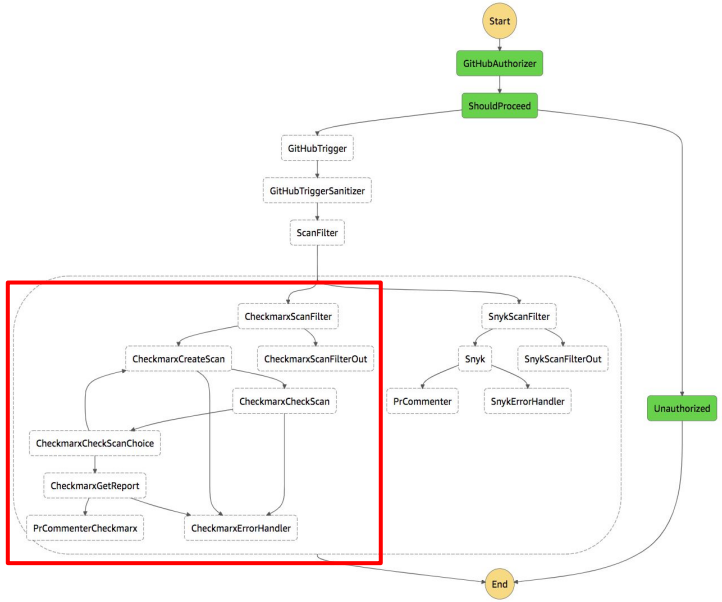
Design


CheckmarxScanFilter
CheckmarxCreateScan
CheckmarxScanFilterOut
CheckmarxCheckScan
CheckmarxCheckScanChoice
CheckmarxGetReport
PrCommenterCheckmarx
CheckmarxErrorHandler

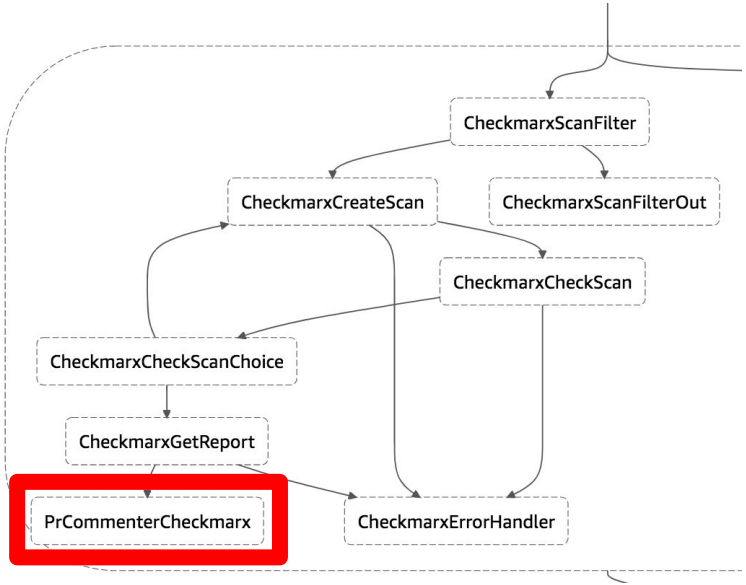

Start
GitHubAuthorizer
ShouldProceed
GitHubTrigger
GitHubTriggerSanitizer
ScanFilter
CheckmarxScanFilter
CheckmarxCreateScan
CheckmarxScanFilterOut
CheckmarxCheckScan
CheckmarxCheckScanChoice
CheckmarxGetReport
PrCommenterCheckmarx
CheckmarxErrorHandler
SnykScanFilter
Snyk
SnykScanFilterOut
PrCommenter
SnykErrorHandler
Unauthorized
End

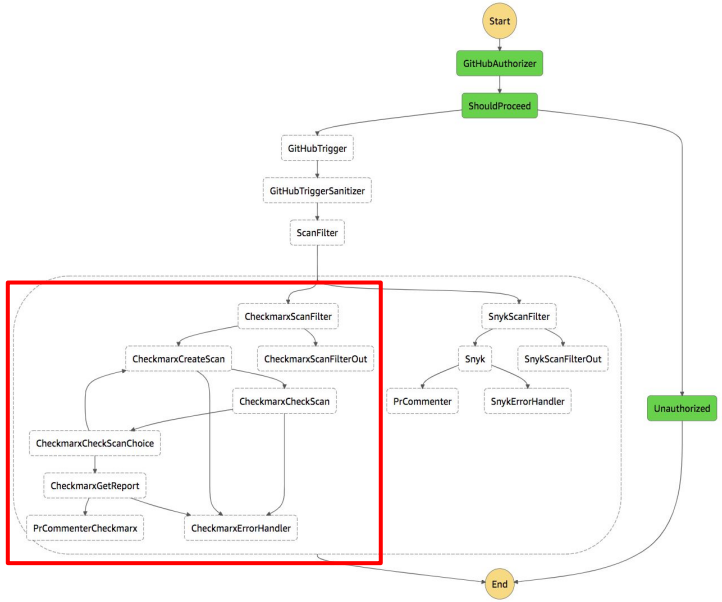
Design


CheckmarxScanFilter
CheckmarxCreateScan
CheckmarxScanFilterOut
CheckmarxCheckScan
CheckmarxCheckScanChoice
CheckmarxGetReport
PrCommenterCheckmarx
CheckmarxErrorHandler

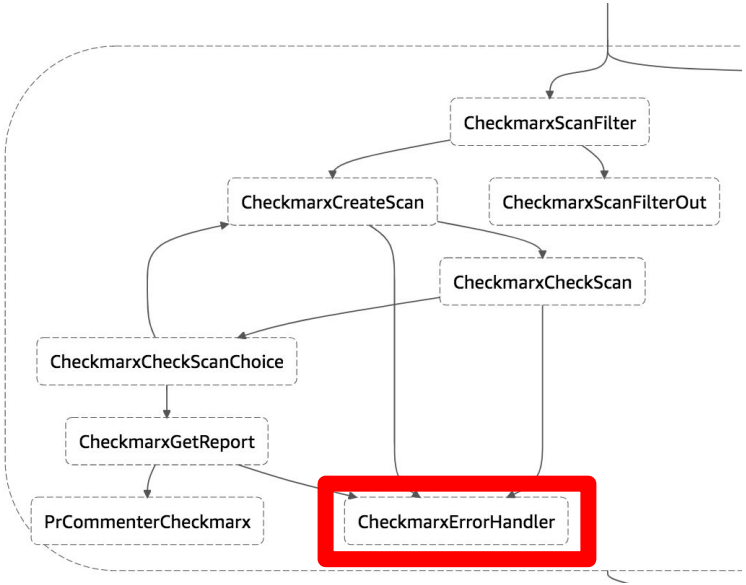

Start
GitHubAuthorizer
ShouldProceed
GitHubTrigger
GitHubTriggerSanitizer
ScanFilter
CheckmarxScanFilter
CheckmarxCreateScan
CheckmarxScanFilterOut
CheckmarxCheckScan
CheckmarxCheckScanChoice
CheckmarxGetReport
PrCommenterCheckmarx
CheckmarxErrorHandler
SnykScanFilter
Snyk
SnykScanFilterOut
PrCommenter
SnykErrorHandler
Unauthorized
End

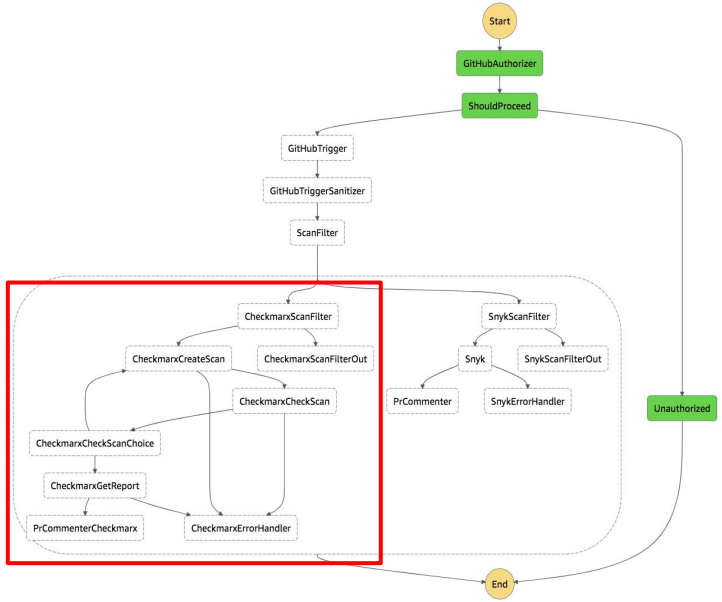
Design


CheckmarxScanFilter
CheckmarxCreateScan
CheckmarxScanFilterOut
CheckmarxCheckScan
CheckmarxCheckScanChoice
CheckmarxGetReport
PrCommenterCheckmarx
CheckmarxErrorHandler

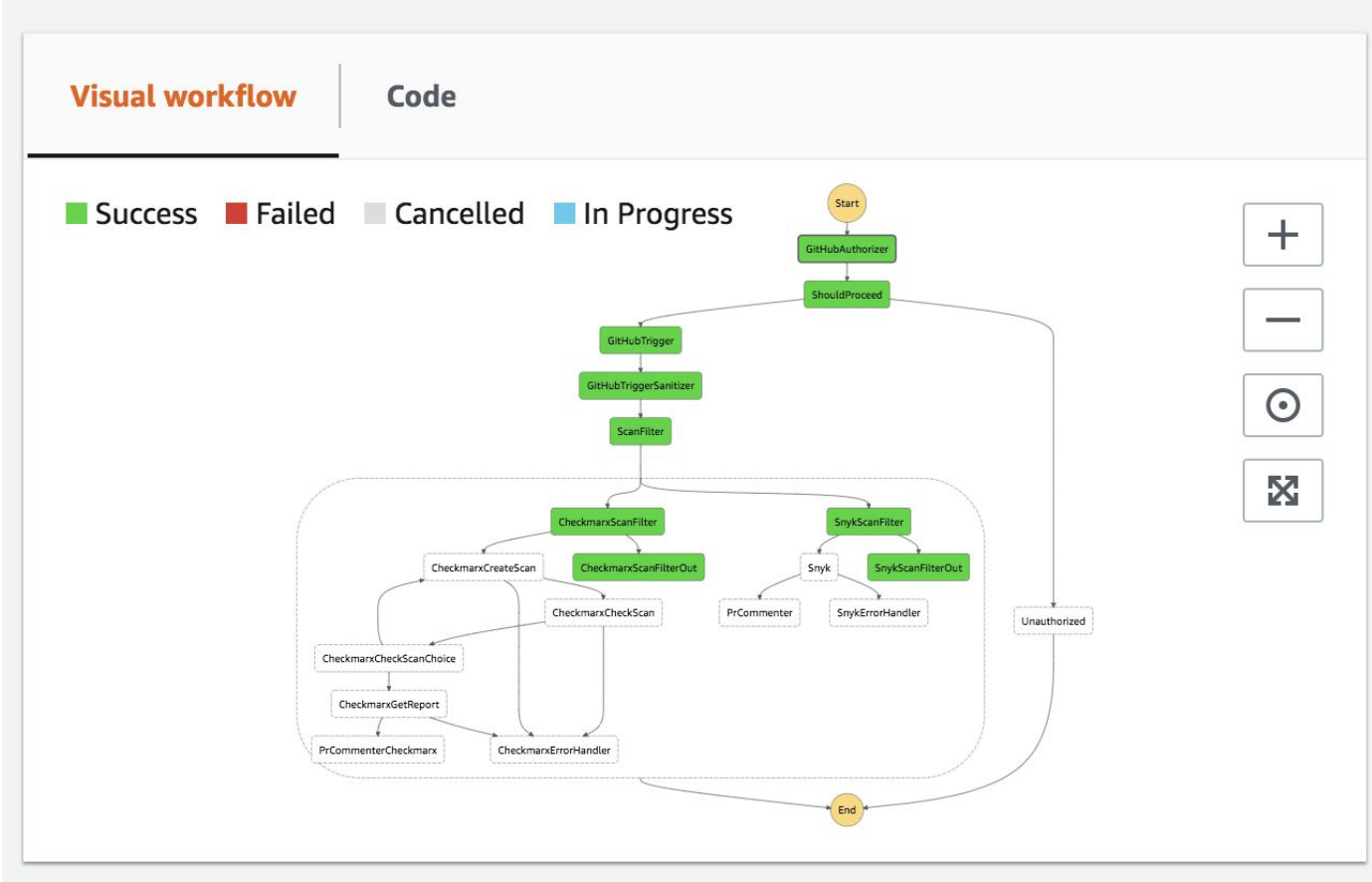

Start
GitHubAuthorizer
ShouldProceed
GitHubTrigger
GitHubTriggerSanitizer
ScanFilter
CheckmarxScanFilter
CheckmarxCreateScan
CheckmarxScanFilterOut
CheckmarxCheckScan
CheckmarxCheckScanChoice
CheckmarxGetReport
PrCommenterCheckmarx
CheckmarxErrorHandler
SnykScanFilter
Snyk
SnykScanFilterOut
PrCommenter
SnykErrorHandler
Unauthorized
End

Design


CheckmarxScanFilter
CheckmarxCreateScan
CheckmarxScanFilterOut
CheckmarxCheckScan
CheckmarxCheckScanChoice
CheckmarxGetReport
PrCommenterCheckmarx
CheckmarxErrorHandler


Start
GitHubAuthorizer
ShouldProceed
GitHubTrigger
GitHubTriggerSanitizer
ScanFilter
CheckmarxScanFilter
CheckmarxCreateScan
CheckmarxScanFilterOut
CheckmarxCheckScan
CheckmarxCheckScanChoice
CheckmarxGetReport
PrCommenterCheckmarx
CheckmarxErrorHandler
SnykScanFilter
Snyk
SnykScanFilterOut
PrCommenter
SnykErrorHandler
Unauthorized
End

Sifting through logs…


Visual workflow
Code
Success
Failed
Cancelled
In Progress
Start
GitHubAuthorizer
ShouldProceed
GitHubTrigger
GitHubTriggerSanitizer
ScanFilter
CheckmarxScanFilter
SnykScanFilter
CheckmarxCreateScan
CheckmarxScanFilterOut
Snyk
SnykScanFilterOut
CheckmarxCheckScan
PrCommenter
SnykErrorHandler
Unauthorized
CheckmarxCheckScanChoice
CheckmarxGetReport
PrCommenterCheckmarx
CheckmarxErrorHandler
End

Step details (GitHubAuthorizer)

Status

Succeeded

Resource

arn:aws:lambda:us-east-1:[redacted]:function:prodsec-middleware-github-authorizer-dev | CloudWatch logs

▶ **Input**

▶ **Output**

▶ **Exception**

Searching for an ID unique to the state machine:

```
print(event)
print(context)
```

It

It doesn’t

It doesn’t exist

Solution

1. Generate a Unique ID in the first lambda (state_id)

Solution

1. Generate a Unique ID in the first lambda (state_id)
2. Override `logging.Filter.filter()`

```
class ContextFilter(logging.Filter):
    def __init__(self, *args, **kwargs):
        self.state_id = kwargs.pop('state_id', None)
        super(ContextFilter, self).__init__(*args, **kwargs)

    def filter(self, record):
        record.state_id = self.state_id
        record.stage = os.getenv('STAGE', 'Unknown')
        return True
```

Solution

1. Generate a Unique ID in the first lambda
2. Override `logging.Filter.filter()`
3. Use custom filter

```
log_format = '%(asctime)s %(levelname)s %(stage)s %(state_id)s %(message)s'
level = logging.INFO
logger = logging.getLogger()
ch = logging.StreamHandler()
formatter = logging.Formatter(log_format)
ch.setFormatter(formatter)
custom_filter = ContextFilter(state_id=state_id)
ch.addFilter(custom_filter)
logger.addHandler(ch)
logger.setLevel(level)
```

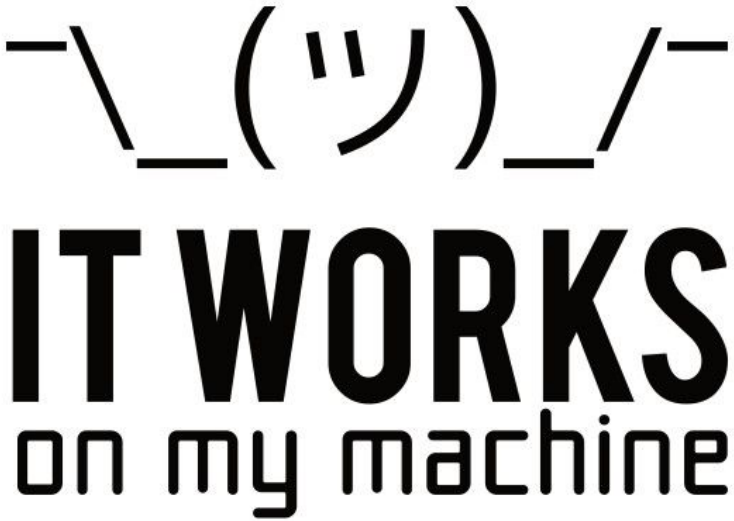
DD Execution ID: 326e2250f055dc8fb593 ×

Show sidebar | **148 results found**

DATE ↓	HOST	SERVIC...	DD EXECUTION ID	MESSAGE
Sep 07 16:07:53.110		lambda	326e2250f055dc8fb5…	> 2018-09-07 20:07:53,110,110 INFO [pr_commenter] -prod-326e2250f055dc8fb593 - PR Commenter Finished
Sep 07 16:07:52.098		lambda	326e2250f055dc8fb5…	> 2018-09-07 20:07:52,097,097 INFO [root] -prod-326e2250f055dc8fb593 - InstallationAuthorization(expires_a
Sep 07 16:07:51.820		lambda	326e2250f055dc8fb5…	> 2018-09-07 20:07:51,819,819 INFO [pr_commenter] -prod-326e2250f055dc8fb593 - PR found. Writing PR commen
Sep 07 16:07:35.038		lambda	326e2250f055dc8fb5…	> 2018-09-07 20:07:35,038,038 INFO [root] -prod-326e2250f055dc8fb593 - InstallationAuthorization(expires_a
Sep 07 16:07:34.480		lambda	326e2250f055dc8fb5…	> 2018-09-07 20:07:34,480,480 INFO [botocore.vendored.requests.packages.urllib3.connectionpool] -prod-326e
Sep 07 16:07:34.202		lambda	326e2250f055dc8fb5…	> 2018-09-07 20:07:34,188,188 INFO [botocore.vendored.requests.packages.urllib3.connectionpool] -prod-326e
Sep 07 16:06:33.881		lambda	326e2250f055dc8fb5…	> 2018-09-07 20:06:33,841,841 INFO [botocore.vendored.requests.packages.urllib3.connectionpool] -prod-326e
Sep 07 16:06:33.580		lambda	326e2250f055dc8fb5…	> 2018-09-07 20:06:33,580,580 INFO [botocore.vendored.requests.packages.urllib3.connectionpool] -prod-326e
Sep 07 16:06:33.319		lambda	326e2250f055dc8fb5…	> 2018-09-07 20:06:33,263,263 INFO [botocore.vendored.requests.packages.urllib3.connectionpool] -prod-326e
Sep 07 16:06:32.982		lambda	326e2250f055dc8fb5…	> 2018-09-07 20:06:32,982,982 INFO [botocore.vendored.requests.packages.urllib3.connectionpool] -prod-326e
Sep 07 16:06:32.702		lambda	326e2250f055dc8fb5…	> 2018-09-07 20:06:32,701,701 INFO [botocore.vendored.requests.packages.urllib3.connectionpool] -prod-326e
Sep 07 16:06:32.673		lambda	326e2250f055dc8fb5…	> 2018-09-07 20:06:32,673,673 INFO [pr_commenter] -prod-326e2250f055dc8fb593 - PR Commenter Starting
Sep 07 16:06:32.488		lambda	326e2250f055dc8fb5…	> 2018-09-07 20:06:32,488,488 INFO [checkmarx.handlers] -prod-326e2250f055dc8fb593 - Checkmarx Get Report
Sep 07 16:06:32.488		lambda	326e2250f055dc8fb5…	> 2018-09-07 20:06:32,488,488 INFO [checkmarx.handlers] -prod-326e2250f055dc8fb593 - Found 11 new vulnerab
Sep 07 16:06:32.446		lambda	326e2250f055dc8fb5…	> 2018-09-07 20:06:32,446,446 INFO [checkmarx.SOAP] -prod-326e2250f055dc8fb593 - Response Status Code: 200
Sep 07 16:06:32.306		lambda	326e2250f055dc8fb5…	> 2018-09-07 20:06:32,277,277 INFO [checkmarx.SOAP] -prod-326e2250f055dc8fb593 - Making SOAP auth request
Sep 07 16:06:32.276		lambda	326e2250f055dc8fb5…	> 2018-09-07 20:06:32,276,276 INFO [checkmarx.checkmarx] -prod-326e2250f055dc8fb593 - Configured odata req
Sep 07 16:06:32.276		lambda	326e2250f055dc8fb5…	> 2018-09-07 20:06:32,276,276 INFO [checkmarx.checkmarx] -prod-326e2250f055dc8fb593 - User authenticated s
Sep 07 16:06:32.276		lambda	326e2250f055dc8fb5…	> 2018-09-07 20:06:32,276,276 INFO [checkmarx.checkmarx] -prod-326e2250f055dc8fb593 - Authentication Statu
Sep 07 16:06:32.131		lambda	326e2250f055dc8fb5…	> 2018-09-07 20:06:32,130,130 INFO [checkmarx.handlers] -prod-326e2250f055dc8fb593 - Getting report id 511
Sep 07 16:06:31.667		lambda	326e2250f055dc8fb5…	> 2018-09-07 20:06:31,666,666 INFO [botocore.vendored.requests.packages.urllib3.connectionpool] -prod-326e
Sep 07 16:06:31.648		lambda	326e2250f055dc8fb5…	> 2018-09-07 20:06:31,648,648 INFO [checkmarx.handlers] -prod-326e2250f055dc8fb593 - Checkmarx Get Report
Sep 07 16:06:30.289		lambda	326e2250f055dc8fb5…	> 2018-09-07 20:06:30,288,288 INFO [checkmarx.handlers] -prod-326e2250f055dc8fb593 - Checkmarx Check Scan

```
[pr_commenter] -prod-326e2250f055dc8fb593 - PR Commenter Starting
[checkmarx.handlers] -prod-326e2250f055dc8fb593 - Checkmarx Get Report
[checkmarx.handlers] -prod-326e2250f055dc8fb593 - Found 11 new vulnerab
[checkmarx.SOAP] -prod-326e2250f055dc8fb593 - Response Status Code: 200
[checkmarx.SOAP] -prod-326e2250f055dc8fb593 - Making SOAP auth request
[checkmarx.checkmarx] -prod-326e2250f055dc8fb593 - Configured odata req
[checkmarx.checkmarx] -prod-326e2250f055dc8fb593 - User authenticated s
[checkmarx.checkmarx] -prod-326e2250f055dc8fb593 - Authentication Statu
[checkmarx.handlers] -prod-326e2250f055dc8fb593 - Getting report id 511
[botocore.vendored.requests.packages.urllib3.connectionpool] -prod-326e
[checkmarx.handlers] -prod-326e2250f055dc8fb593 - Checkmarx Get Report
[checkmarx.handlers] -prod-326e2250f055dc8fb593 - Checkmarx Check Scan
```

Developer feedback



Developer feedback

- Important part of the project: the developer
  - Developer engagement is crucial

Developer feedback

- Important part of the project: the developer
  - Developer engagement is crucial
  - We need to give the results in the best way possible

Developer feedback

- Important part of the project: the developer
  - Developer engagement is crucial
  - We need to give the results in the best way possible
  - Low-noise, easy to read, where the developer is

Developer feedback

- Important part of the project: the developer
  - Developer engagement is crucial
  - We need to give the results in the best way possible
  - Low-noise, easy to read, where the developer is
  - Make issues found actionable

Developer feedback

- Important part of the project: the developer
  - Developer engagement is crucial
  - We need to give the results in the best way possible
  - Low-noise, easy to read, where the developer is
  - Make issues found actionable
- The developer is our “customer”

Developer feedback

- We reach out to them
  - Send organization wide emails asking for feedback
  - Application Security focus group
  - Direct discussion with developers

Developer feedback

- We reach out to them
    - Send organization wide emails asking for feedback
    - Application Security focus group
    - Direct discussion with developers

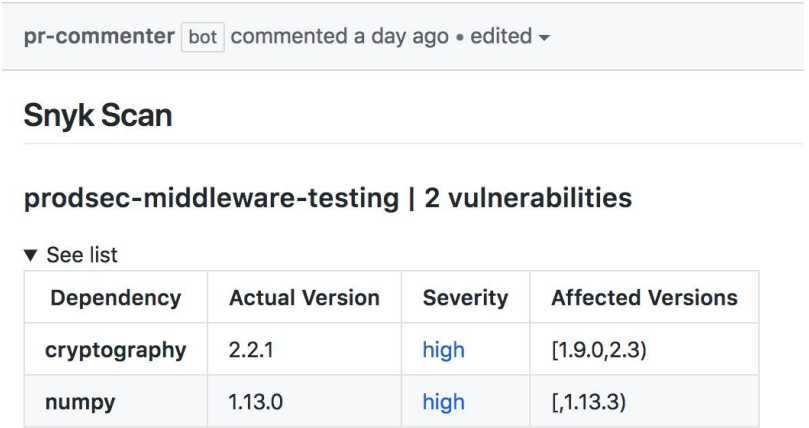
```
def feature_flag(event, datadog_manager):
    branch_name = event.get('github', {}).get('branch', "")
    # Feature flag to roll progressively
    if not (branch_name.startswith('julesdt') or branch_name.startswith('jmassey') or branch_name.startswith(
            pratikgs-dd') or branch_name.startswith('justin.massey')):
        log.info("Not the right branch")
        datadog_manager.send_metric("count", "prodsec_middleware.scan_filter.invalid", 1)
        return False
    return True
```

Developer feedback

- Follow their workflows
  - Use the same CI/CD tools
  - Try to use the same technologies
- Helps understanding their constraints

Developer feedback

- Our current solution
  - Comment on the PR
    This is where the dev is looking



pr-commenter bot commented a day ago • edited

Snyk Scan

prodsec-middleware-testing | 2 vulnerabilities

▼ See list

Dependency	Actual Version	Severity	Affected Versions
cryptography	2.2.1	high	[1.9.0,2.3)
numpy	1.13.0	high	[,1.13.3)

Developer feedback

- Our current solution
  - Comment on the PR
    This is where the dev is looking
- Our plan for the future
  - All the scans in a single comment
  - “Auto-fix” PR (when possible) created and linked for review

Deployment

Deployment

- GitHub Authorizer
- GitHub Trigger
- Plugin that uses Go Security Scanner
- GitHub PR Commenter
- Slack Output

Deployment

- Lambdas
  - Serverless framework

Deployment

- Lambdas
  - Serverless framework
- IAM, Step Function, API Gateway
  - Terraform

Thank you! Questions?

Jules Denardou

@Pod_Sec

@JulesDT

Justin Massey

@jmassey09

@th3r3p0

We are hiring: Paris, New York, and remote!