



EDGE X FOUNDRY™

SAST for EdgeX Foundry

James Gregg / EdgeX DevOps

Wednesday July 17, 2019

Static Application Security Testing (SAST) (Fuji scope)

Initial evaluation focused on tools / languages / cost / ease of integration

- *Missing requirements from Security WG*

Tool Selection

What's the right SAST tool for EdgeX Foundry?

- *Define the criteria for tool selection*

Implementation / Integration

What should happen if the scan finds a vulnerability?

- Define Build Behavior
 - Break the build, Continue build but Alert, Scan outside of Build separately
 - How to Alert and Who receives notification? Slack, Email

Developer Feedback Loop / Remediation

What happens after tool is up and running scans?

- Code remediation timelines based on severity
- Roles and Responsibility of Code Reviewer within tool
- Measurement of improved security of code base over time (ties into certification)

Seeking input, recommendation and direction from EdgeX Security experts

Initial Analysis of SAST Tools

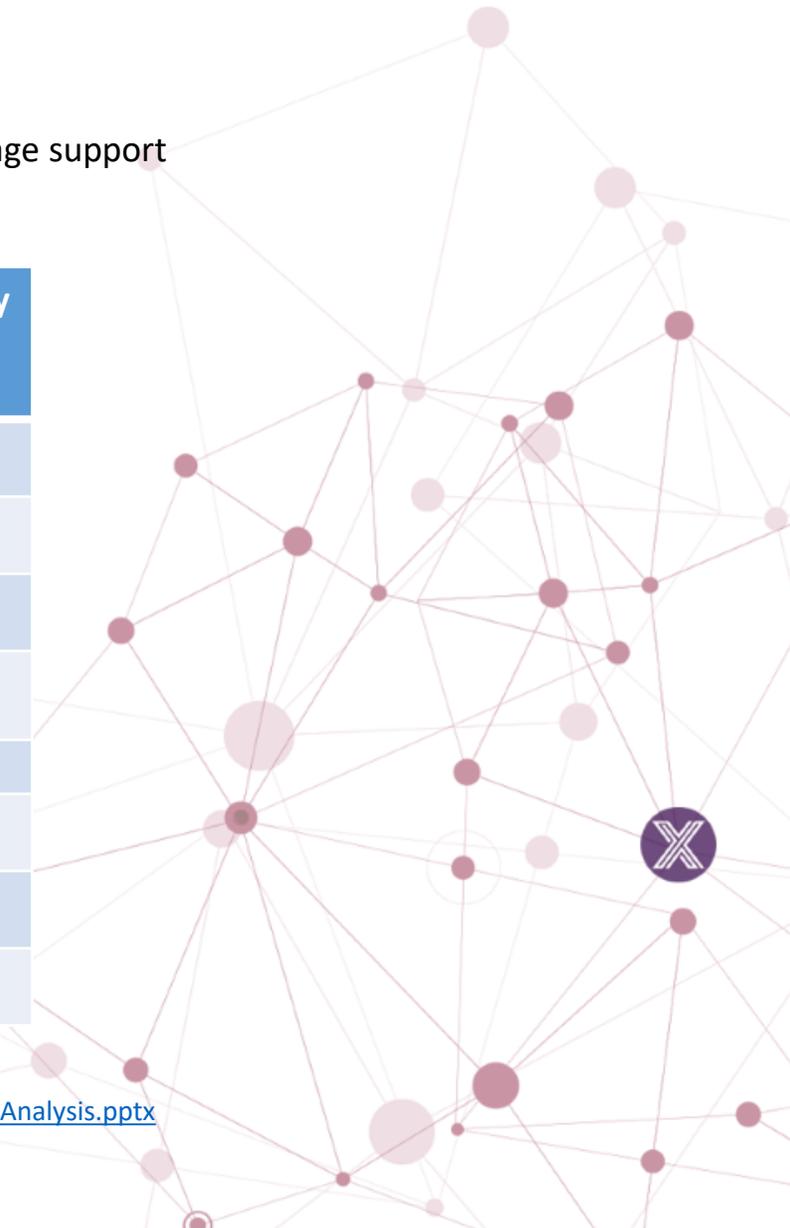
- Previous discussions and review of findings in EdgeX DevOps WG meetings
- Focused on tools that were FOSS and supported EdgeX Foundry code base including Go language support



Languages	GoLang	C	C++	JavaScript	Python	Groovy
Tools						
Fortify	✓	✓	✓	✓	✓	✓
Coverity	✓	✓	✓	✓	✓	✓
Kritika		✓	✓	✓	✓	
SonarQube / SonarCloud	✓			✓	✓	
Checkmarx	✓			✓		
Snyk	✓			✓	✓	
Guardrails	✓			✓	✓	
Whitehat				✓		

Recordings / Meeting Minutes are available [online](#) (WW26-28)

<https://wiki.edgexfoundry.org/display/FA/DevOps+Working+Group?preview=/329486/31588576/StaticCodeAnalysis.pptx>



Analysis / Ranking and Rating

Based on the previous table, the analysis are captured for the tools which supports GoLang.

Rank	Tools	Github Integration	Support for Docker	Cost	Open Source	SCM Support	SaaS	Rules config	Remarks
1	Guardrails	Need to see how this could work for EdgeX Foundry.		Free for open source project Pricing	No	Yes	Yes	Yes	Paid clients have additional functionality for CLI and PR support
2	SonarQube <i>SonarCloud</i>	Easy github plugin Scanner Info Server Info	Yes Docker	Free for open source project, except for C, C++ Pricing	Yes	Yes • Get Started • Jenkins Configuratio n • Sonar Scanner	Yes	Yes	It supports GoLang, but C/C++ scanning works only on Developer / Enterprise version. Additional overhead to implement with LF resources needed to enable by FUJI.
3	Snyk	Integration not supported for GoLang (Details)	No for GoLang	Free for open source project Pricing	No	Yes Documentation	Yes	Yes	Snyk supports testing and monitoring Go projects that have their dependencies managed by dep or govendor . Go support is currently supported via the Snyk CLI and Git Integrations <i>Seems like limited support for GoLang—need supplier roadmap</i> Reporting requires paid plan = \$\$\$
<div style="border: 2px solid orange; border-radius: 15px; padding: 10px; background-color: #f9f9f9;"> <p style="background-color: #4CAF50; color: white; padding: 2px 5px; display: inline-block; margin-bottom: 5px;">Updates</p> <ul style="list-style-type: none"> Snyk now supports Go with go modules using CLI Coverity not considered in this initial analysis Linux Foundation doesn't support SonarQube instances but offers SonarCloud </div>									
4	Fortify	Don't see the proper documentation	Couldn't find it	Cost involved	No	Yes Data Sheet	Yes	Yes	It supports on demand scanning, but it is expensive.

POC Recap

Guardrails 1

detect-secretsv0.11.0 00m 00.645s

File: `edgex/charts/edgex-mongo/templates/deploy-mongo.yaml:26`
 Rule ID: Hex High Entropy String
 Finding Type: secret

Associated metadata

```
{
  "hashedSecret": "032813bf7f421b78b30901f32458b7b0f0ea15e6"
}
```

This is not a false positive

Initial Findings:

Not the right solution for this project.
 Oriented towards web based applications, not microservices.

Snyk 3

James Gregg | james.r.gregg@intel.com

Dashboard Reports **Projects** Integrations Settings

jamesgregg/hellonode:package.json

Overview History Settings

Snapshot taken by recurring test 4 days ago. Retest now

Vulnerabilities	0 via 0 paths	Dependencies	0	Source	GitHub
Taken by	Recurring	Tested with	package.json	Repository	hellonode
Branch	master	Manifest	package.json		

Issues Dependencies

Severity

- High 0
- Medium 0
- Low 0

Status

- Open 0
- Patched 0
- Ignored 0

No known vulnerabilities found

Add a Snyk Badge to the README file to show that this project is free of vulnerabilities.

vulnerabilities 0

HTML

```
<a href="https://snyk.io//test/github/jamesgregg/hellonode?targetFile=package.json"><img src="https://snyk.io/test/github/
```

Markdown

```
[[Known Vulnerabilities]](https://snyk.io//test/github/jamesgregg/hellonode/badge.svg?targetFile=package.json)|(https://sny
```

Initial Findings:

Ruled out due to lack of Go Lang support with go modules.
 There now appears to be a Snyk Docker based CLI that supports scan based on go.mod

SonarCloud 2

(Linux Foundation service offering)

https://sonarcloud.io/organizations/edgexfoundry/projects

sonarcloud My Projects My Issues

EdgeX Foundry Project

Projects Issues Quality Profiles Rules Quality Gates Members

Filters

Perspective: Overall Status Sort by: Name Search by project name or key 1 projects

Quality Gate

Passed	0	1
Warning	0	1
Failed	0	1

Reliability (Bugs)

A	0	1
B	0	1
C	0	1
D	0	1
E	0	1

Security (Vulnerabilities)

A	0	1
B	0	1
C	0	1
D	0	1
E	0	1

Maintainability (Code Smells)

A	0	1
---	---	---

edgex-go Public

Project is not analyzed yet. [Configure analysis](#)

1 of 1 shown

© 2008-2019, SonarCloud by SonarSource SA. All rights reserved.
News - Twitter - Terms - Pricing - Privacy - Security - Help - Contact us - Status - About

• SonarCloud SaaS version

- Free for Linux Foundation (Open Source)
- Easy implementation
- Offers Code Quality / Code Coverage / Code Smells
 - Same capability already in place now with Codecov.io for Code Coverage reporting
- Very Limited Security Checks for Go (see rules next slide)
- Other Languages include (mostly web oriented)
 - OWASP Top 10
 - Sans Top 25
 - CWE

Currently enabled on edgex-go repo but hasn't been scanned to date

SonarCloud Vulnerability Rules for Go

Rules

The screenshot displays the SonarCloud interface for the EdgeX Foundry Project. The URL <https://sonarcloud.io/organizations/edgexfoundry/rules> is highlighted in the address bar. The page shows a list of rules for the Go language, with two rules highlighted in a red box:

- Credentials should not be hard-coded
- IP addresses should not be hardcoded

The interface also shows a search bar, navigation controls, and a filter for 'Vulnerability' rules. The 'Vulnerability' filter is selected, and the 'Clear' button is visible. The 'Vulnerability' filter is also selected in the main content area.

Language	Count
Go	2
T-SQL	2
Apex	1
HTML	1
Kotlin	1
Ruby	1
Scala	1
VB.NET	1

Type	Count
Bug	11
Vulnerability	2
Code Smell	33
Security Hotspot	0

Wrap Up

• Discussion

- Developer Workflow should include recommendation from Security WG on developer tool for local scan

Malini suggestion to take a closer look at <https://github.com/golangci/golangci-lint> and gosec

Try - gometalinter with gosec

- Resourcing for role for reviewing would be handled within SIR team
- Need to focus on tools that provide true static code analysis
- Image scanning is addressed with Clair as pre and post release reporting process

• Next Steps / Recommendation

- Look at output from different tools
 - SonarCloud
 - Snyk
 - Coverity
 - Checkmarx
- James to coordinate a demo with Snyk within this forum
- Enable the SonarCloud integration using sonar.properties
- Bring reports into Security WG for review