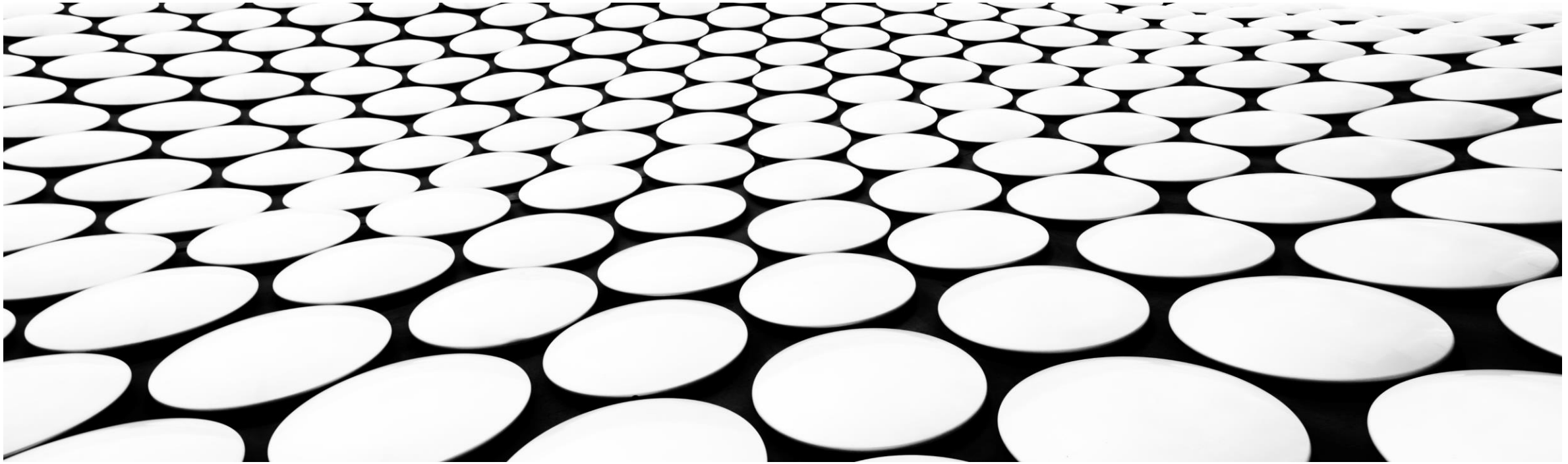
HACK YOURSELF FIRST - TÉCNICAS DE "BUG BOUNTY" E FERRAMENTAS PARA SEU DEVSECOPS

RAFAEL B. BRINHOSA



#ABOUT ME

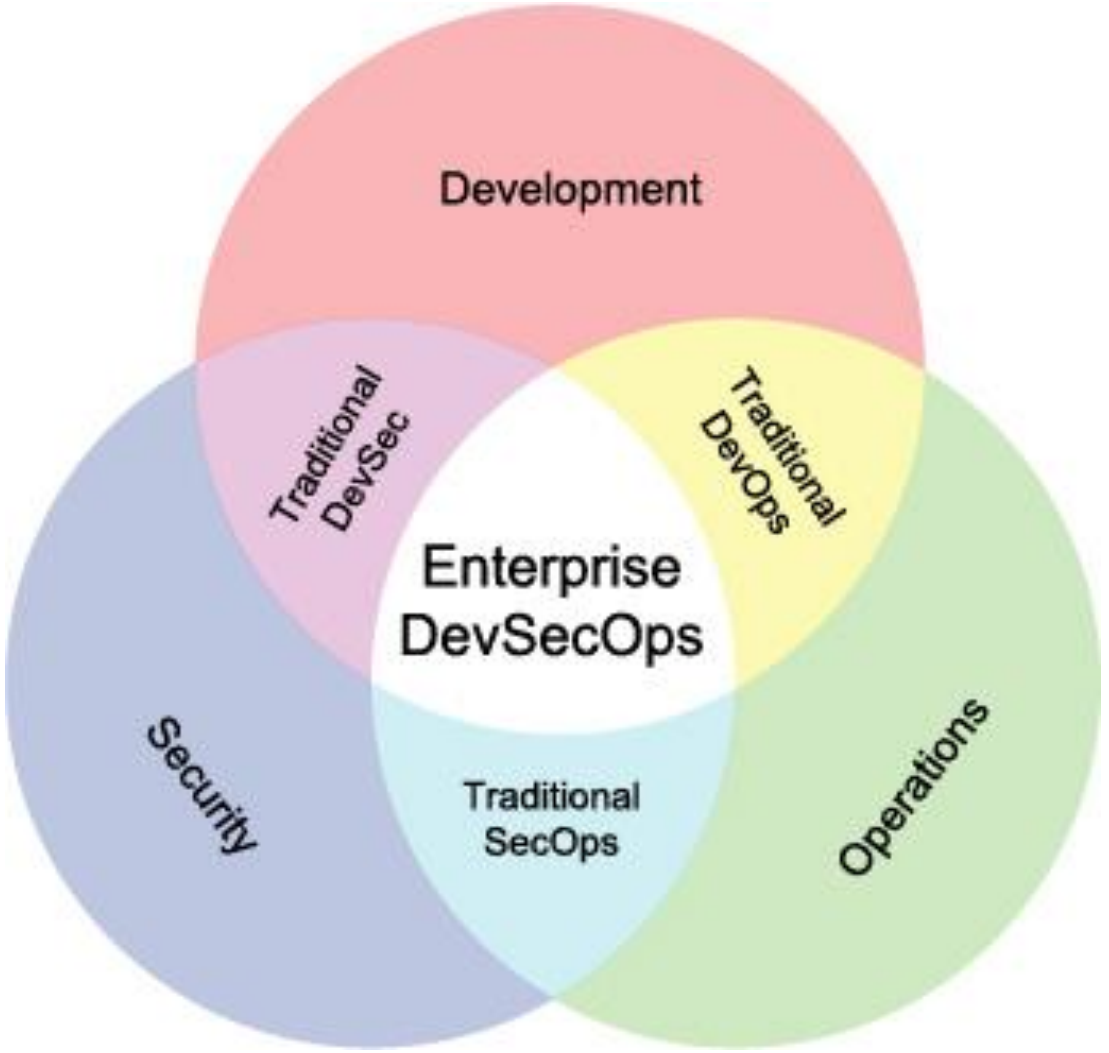
- https://linkedin.com/in/brinhosa

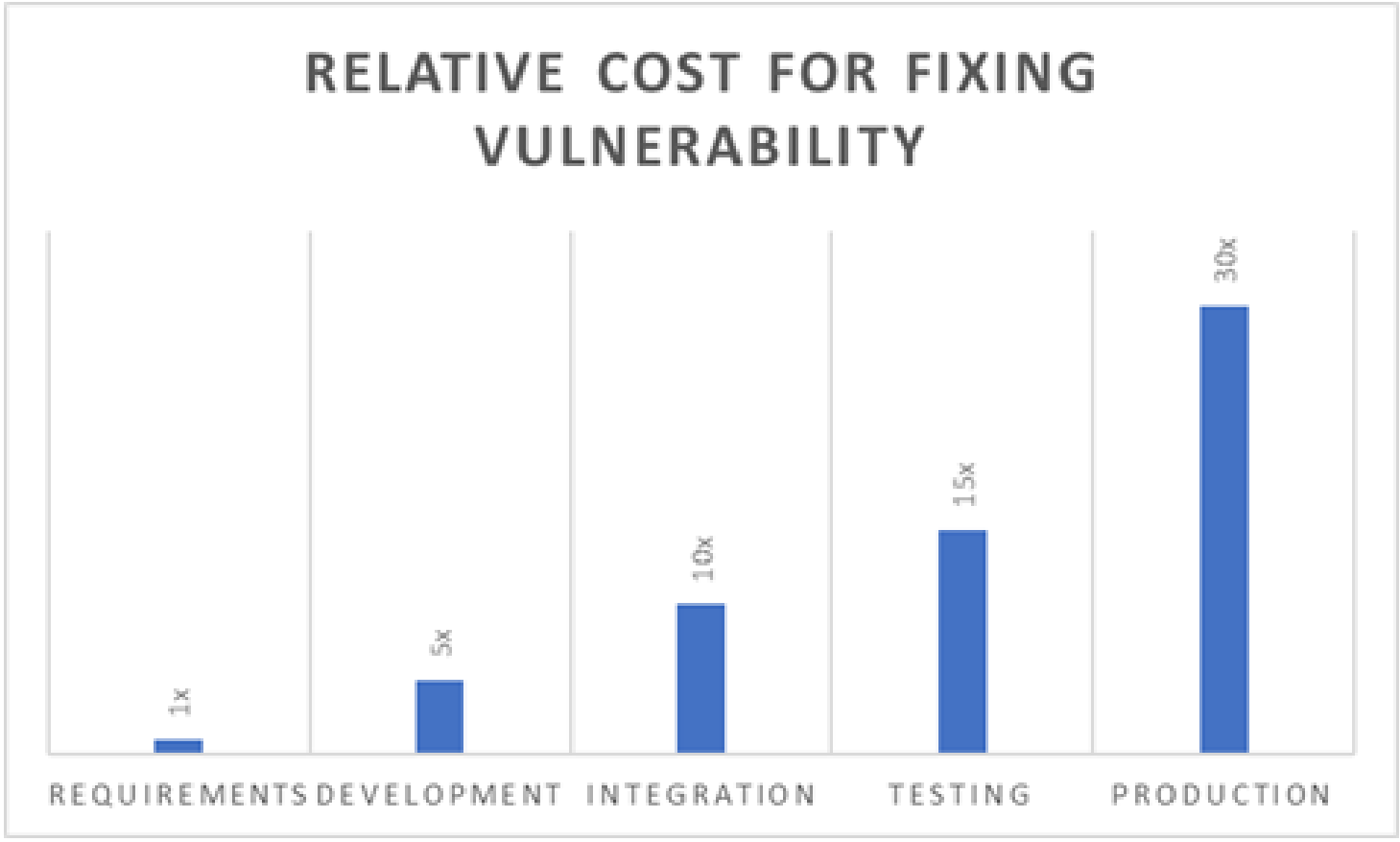
O QUE É BUG BOUNTY?

HACK YOURSELF FIRST

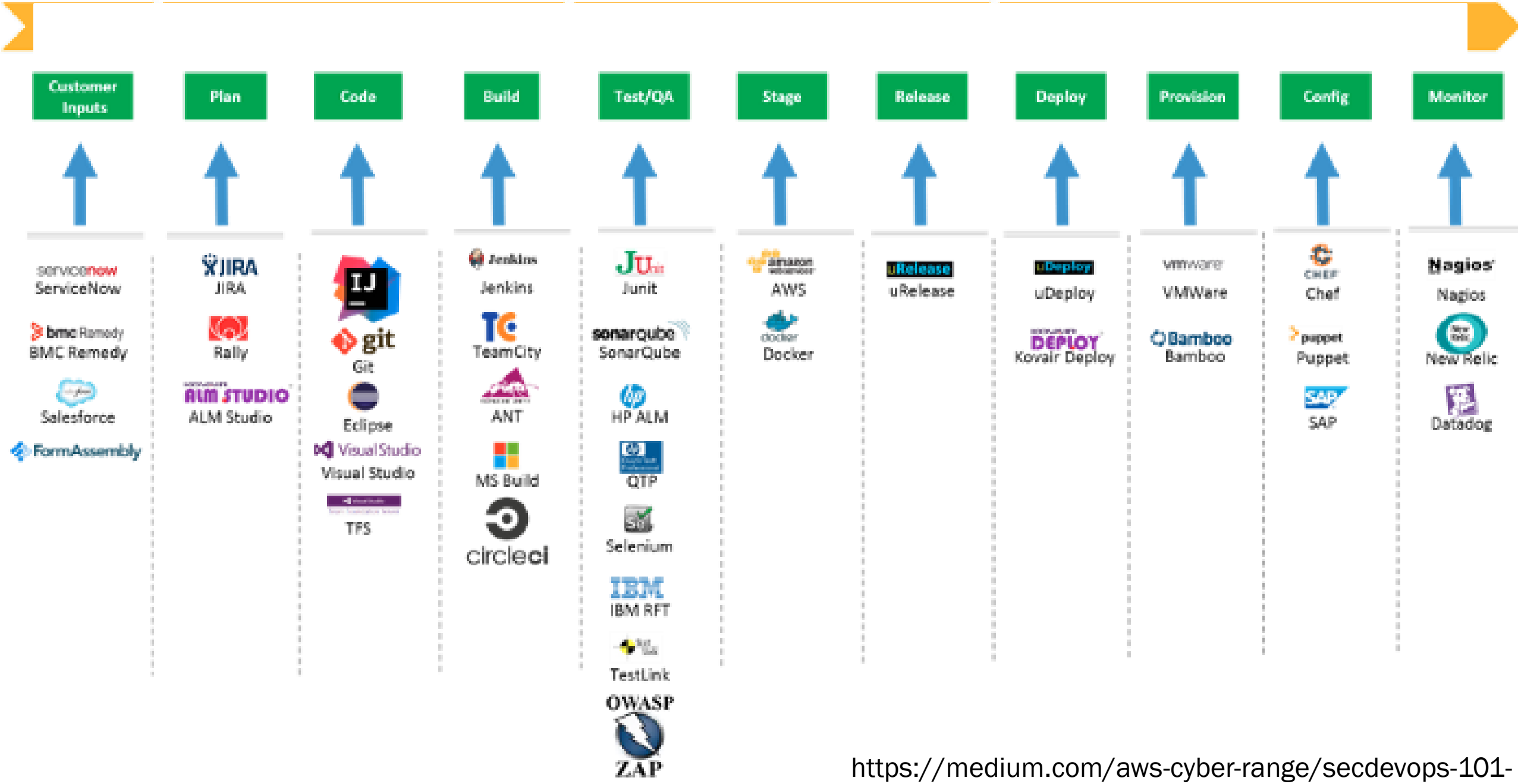
- O que é?
- Por que?

POR QUE DEVSECOPS?

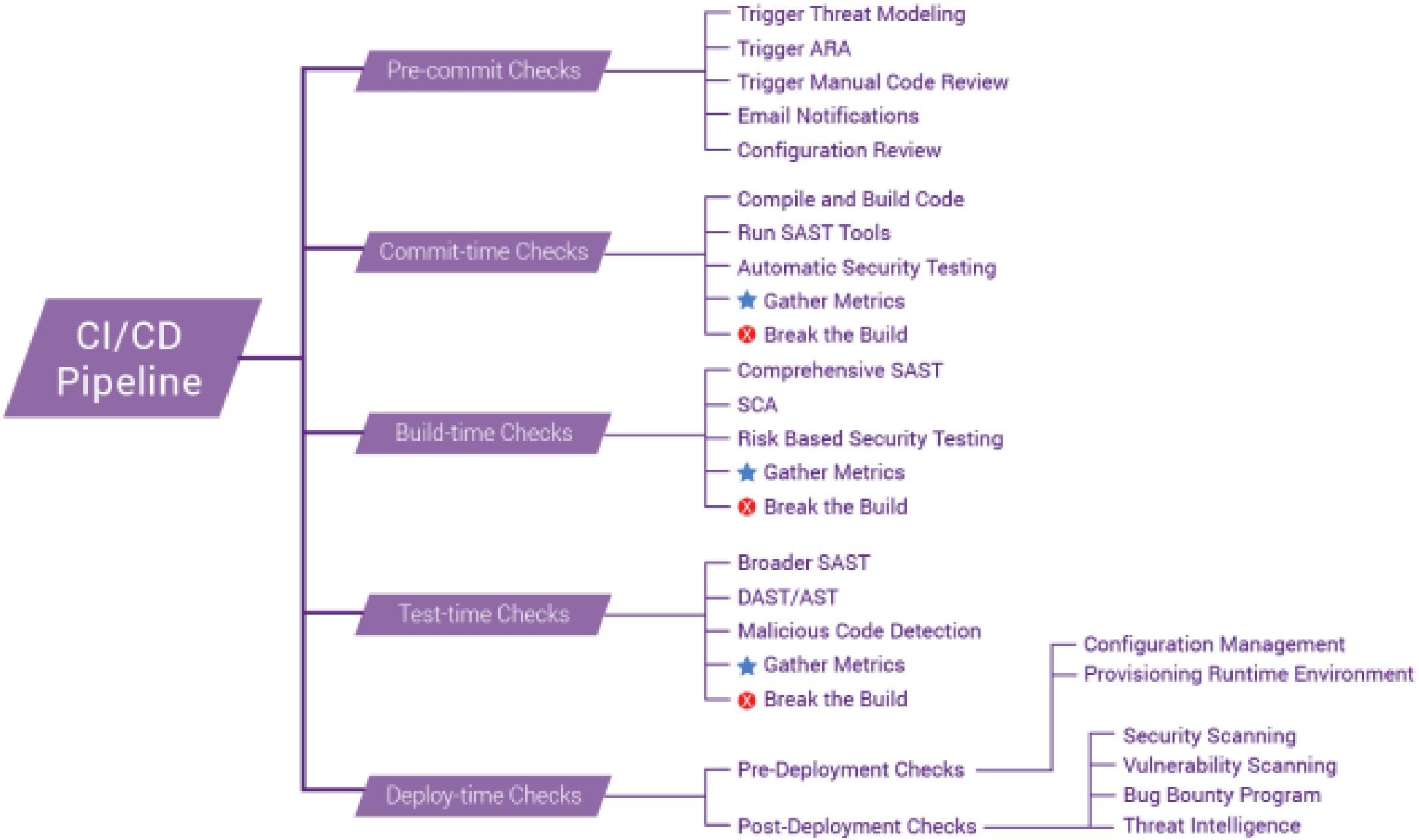



RELATIVE COST FOR FIXING VULNERABILITY
1x
5x
10x
15x
30x
REQUIREMENTS
DEVELOPMENT
INTEGRATION
TESTING
PRODUCTION

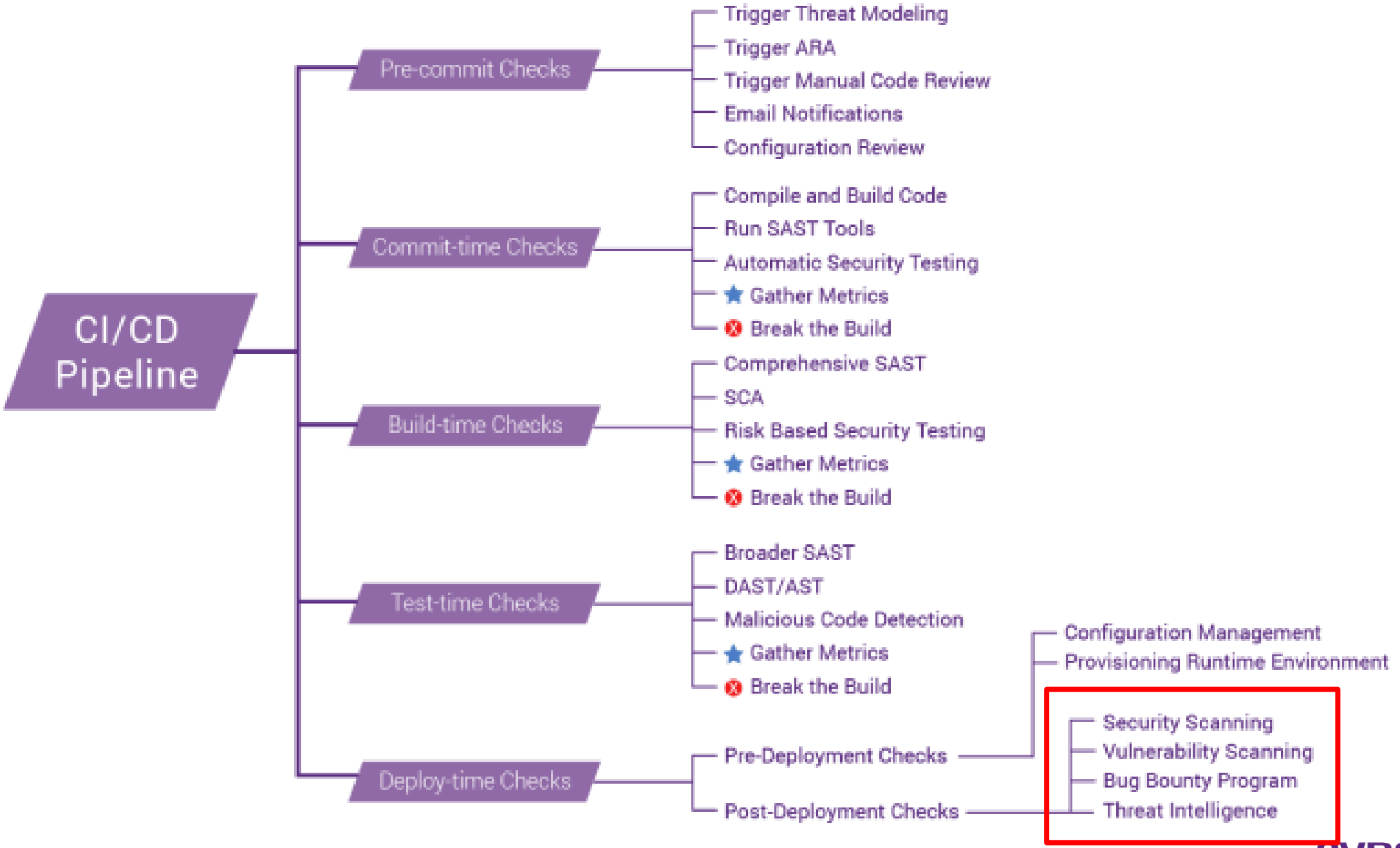
EXEMPLOS DE ESTEIRAS E FERRAMENTAS

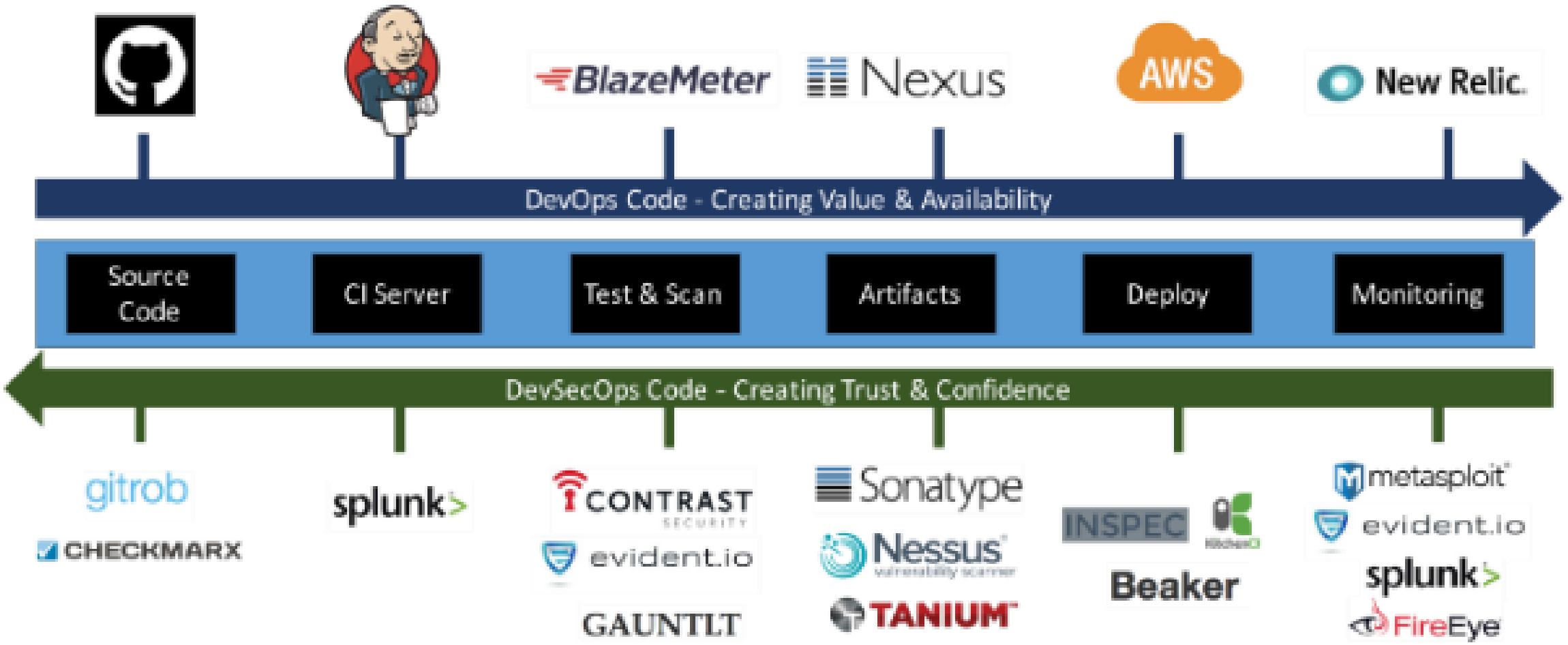


https://medium.com/aws-cyber-range/secdevops-101-strengthen-the-basics-20f57197aa1c

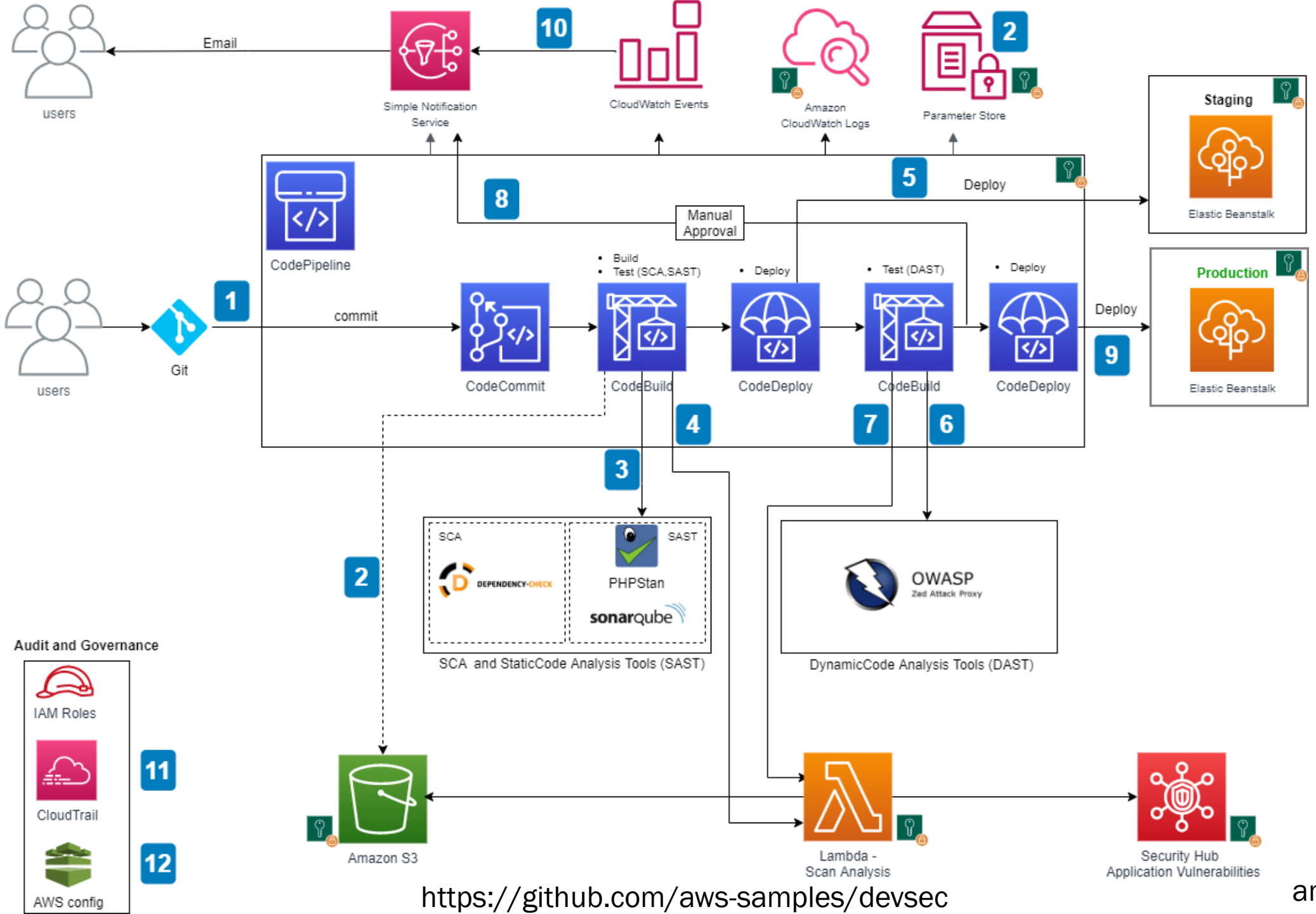

CI/CD Pipeline
Pre-commit Checks
Trigger Threat Modeling
Trigger ARA
Trigger Manual Code Review
Email Notifications
Configuration Review
Commit-time Checks
Compile and Build Code
Run SAST Tools
Automatic Security Testing
Gather Metrics
Break the Build
Build-time Checks
Comprehensive SAST
SCA
Risk Based Security Testing
Gather Metrics
Break the Build
Test-time Checks
Broader SAST
DAST/AST
Malicious Code Detection
Gather Metrics
Break the Build
Deploy-time Checks
Pre-Deployment Checks
Configuration Management
Provisioning Runtime Environment
Post-Deployment Checks
Security Scanning
Vulnerability Scanning
Bug Bounty Program
Threat Intelligence

SYNOPSYS®

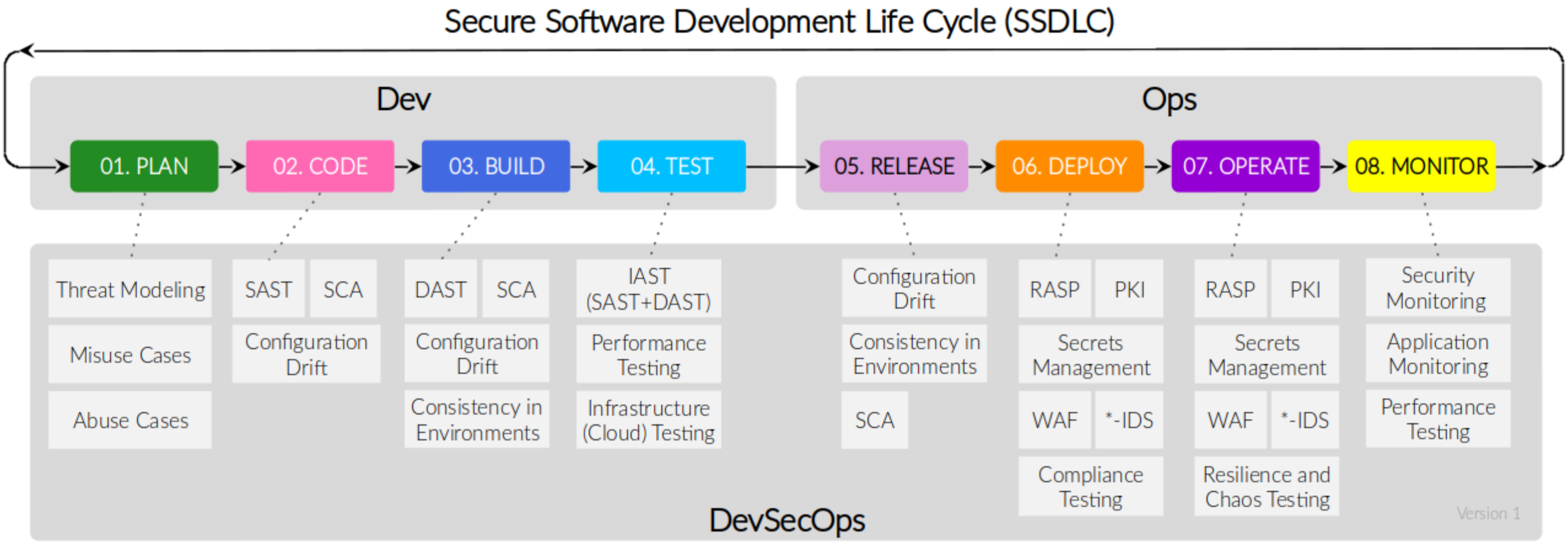

CI/CD Pipeline
Pre-commit Checks
Trigger Threat Modeling
Trigger ARA
Trigger Manual Code Review
Email Notifications
Configuration Review
Commit-time Checks
Compile and Build Code
Run SAST Tools
Automatic Security Testing
Gather Metrics
Break the Build
Build-time Checks
Comprehensive SAST
SCA
Risk Based Security Testing
Gather Metrics
Break the Build
Test-time Checks
Broader SAST
DAST/AST
Malicious Code Detection
Gather Metrics
Break the Build
Deploy-time Checks
Pre-Deployment Checks
Configuration Management
Provisioning Runtime Environment
Post-Deployment Checks
Security Scanning
Vulnerability Scanning
Bug Bounty Program
Threat Intelligence



devsecops.org



https://github.com/aws-samples/devsec


Secure Software Development Life Cycle (SSDLC)
Dev
Ops
01. PLAN
02. CODE
03. BUILD
04. TEST
05. RELEASE
06. DEPLOY
07. OPERATE
08. MONITOR
Threat Modeling
Misuse Cases
Abuse Cases
SAST
SCA
Configuration Drift
DAST
SCA
Configuration Drift
Consistency in Environments
IAST (SAST+DAST)
Performance Testing
Infrastructure (Cloud) Testing
Configuration Drift
Consistency in Environments
SCA
RASP
PKI
Secrets Management
WAF
*-IDS
Compliance Testing
RASP
PKI
Secrets Management
WAF
*-IDS
Resilience and Chaos Testing
Security Monitoring
Application Monitoring
Performance Testing
DevSecOps
Version 1

DevSecOps Security Controls

DEVELOPMENT

OPERATIONS

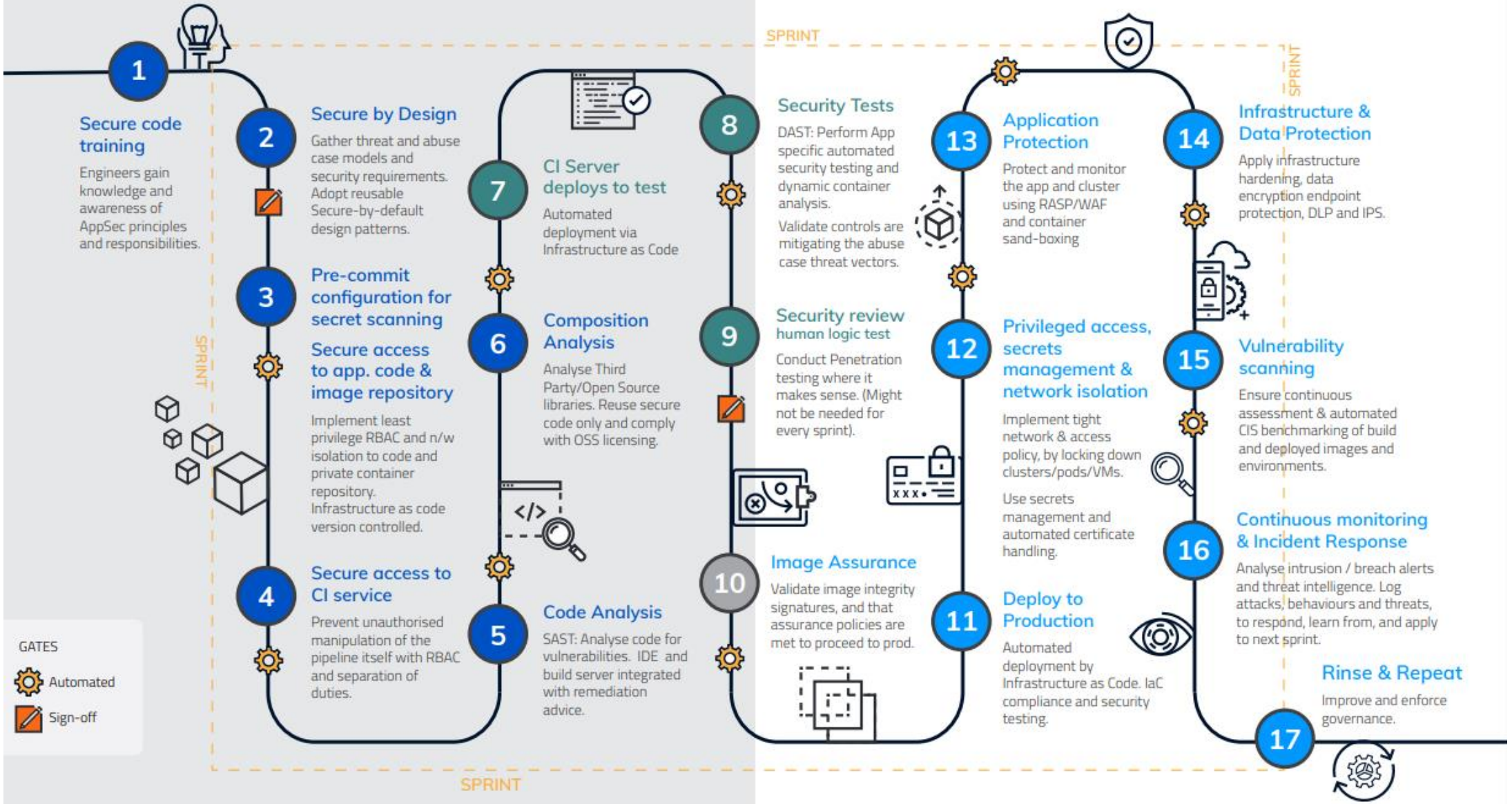
SECURITY

SPRINT

SPRINT

SPRINT

SPRINT



https://accelera.com.au/what-the-sec-is-devsecops/

v1.0 - Sep 2020

TÉCNICAS DE BUG BOUNTY

TÉCNICAS DE BUG BOUNTY

- FERRAMENTAS NÃO CONVENCIONAIS
- ONE-LINERS
- PARALELISMO
- TESTE EM MASSA

FERRAMENTAS E ONE-LINERS

ANÁLISE DE VULNERABILIDADES DO CONTAINER - TRIVY

- Trivy

- Trivy é um scanner simples e abrangente para vulnerabilidades em imagens de contêiner, sistemas de arquivos e repositórios Git, bem como para problemas de configuração. Trivy detecta vulnerabilidades de pacotes de sistema operacional (Alpine, RHEL, CentOS, etc.) e pacotes específicos da linguagem (Bundler, Composer, npm, yarn, etc.). Além disso, o Trivy verifica arquivos de infraestrutura como código (IaC), como Terraform, Dockerfile e Kubernetes, para detectar possíveis problemas de configuração que expõem suas implantações ao risco de ataque.

```
2021-11-18T10:38:38.410-0300    [34mINFO[0m    Detected OS: ubuntu
2021-11-18T10:38:38.411-0300    [34mINFO[0m    Detecting Ubuntu vulnerabilities...
2021-11-18T10:38:38.445-0300    [34mINFO[0m    Number of language-specific files: 0

infoslack/dvwa (ubuntu 14.04)
=============================
Total: 1626 (UNKNOWN: 0, LOW: 450, MEDIUM: 1148, HIGH: 28, CRITICAL: 0)
```

LIBRARY	VULNERABILITY ID	SEVERITY	INSTALLED VERSION	FIXED VERSION	TITLE
apache2	CVE-2016-0736	MEDIUM	2.4.7-1ubuntu4.9	2.4.7-1ubuntu4.14	httpd: Padding Oracle in Apache mod_session_crypto -->avd.aquasec.com/nvd/cve-2016-0736
	CVE-2016-5387			2.4.7-1ubuntu4.13	Apache HTTPD: sets environmental variable based on user supplied Proxy request header... -->avd.aquasec.com/nvd/cve-2016-5387
	CVE-2016-8743			2.4.7-1ubuntu4.14	httpd: Apache HTTP Request Parsing Whitespace Defects -->avd.aquasec.com/nvd/cve-2016-8743
	CVE-2017-3167			2.4.7-1ubuntu4.16	httpd: ap_get_basic_auth_pw() authentication bypass -->avd.aquasec.com/nvd/cve-2017-3167
	CVE-2017-3169				httpd: mod_ssl NULL pointer dereference -->avd.aquasec.com/nvd/cve-2017-3169
	CVE-2017-7668				httpd: ap_find_token() buffer overread -->avd.aquasec.com/nvd/cve-2017-7668
	CVE-2017-9788			2.4.7-1ubuntu4.17	httpd: Uninitialized memory reflection in mod_auth_digest -->avd.aquasec.com/nvd/cve-2017-9788

ANÁLISE DE VULNERABILIDADES DO CONTAINER - GRYPE

- Grype

- Grype é um projeto de código aberto para escanear seu projeto ou contêiner em busca de vulnerabilidades conhecidas. Grype usa as informações mais recentes dos mesmos serviços de feed da Anchore que o Anchore Engine. Você pode usar Grype para identificar vulnerabilidades na maioria dos pacotes do sistema operacional Linux e artefatos de linguagem, incluindo NPM, Python, Ruby e Java.
- docker run --rm -it -v /var/run/docker.sock:/var/run/docker.sock anchore/grype:latest infoslack/dvwa
- ✓ Vulnerability DB [updated]
- ✓ Loaded image
- ✓ Parsed image
- ✓ Cataloged packages [288 packages]
- ✓ Scanned image [2013 vulnerabilities]

ANÁLISE DE VULNERABILIDADES DO CONTAINER - GRYPE

NAME	INSTALLED	FIXED-IN	VULNERABILITY	SEVERITY
apache2	2.4.7-1ubuntu4.9	2.4.7-1ubuntu4.22	CVE-2018-17199	Low
apache2	2.4.7-1ubuntu4.9	2.4.7-1ubuntu4.20	CVE-2018-1312	Low
apache2	2.4.7-1ubuntu4.9	2.4.7-1ubuntu4.22	CVE-2019-0217	Medium
apache2	2.4.7-1ubuntu4.9	2.4.7-1ubuntu4.16	CVE-2017-7679	Low
apache2	2.4.7-1ubuntu4.9	2.4.7-1ubuntu4.18	CVE-2017-9798	Medium
apache2	2.4.7-1ubuntu4.9	2.4.7-1ubuntu4.13	CVE-2016-5387	Medium
apache2	2.4.7-1ubuntu4.9	2.4.7-1ubuntu4.20	CVE-2018-1283	Low
apache2	2.4.7-1ubuntu4.9	2.4.7-1ubuntu4.16	CVE-2017-7668	Medium
apache2	2.4.7-1ubuntu4.9	2.4.7-1ubuntu4.20	CVE-2017-15710	Low
apache2	2.4.7-1ubuntu4.9	2.4.7-1ubuntu4.20	CVE-2017-15715	Low
apache2	2.4.7-1ubuntu4.9	2.4.7-1ubuntu4.14	CVE-2016-4975	Low
apache2	2.4.7-1ubuntu4.9	2.4.7-1ubuntu4.14	CVE-2016-8743	Medium
apache2	2.4.7-1ubuntu4.9	2.4.7-1ubuntu4.16	CVE-2017-3167	Medium
apache2	2.4.7-1ubuntu4.9	2.4.7-1ubuntu4.22	CVE-2019-0220	Low
apache2	2.4.7-1ubuntu4.9	2.4.7-1ubuntu4.17	CVE-2017-9788	Medium
apache2	2.4.7-1ubuntu4.9	2.4.7-1ubuntu4.16	CVE-2017-3169	Medium
apache2	2.4.7-1ubuntu4.9	2.4.7-1ubuntu4.20	CVE-2018-1301	Low
apache2	2.4.7-1ubuntu4.9	2.4.7-1ubuntu4.14	CVE-2016-2161	Low
apache2	2.4.7-1ubuntu4.9	2.4.7-1ubuntu4.20	CVE-2018-1303	Low
apache2	2.4.7-1ubuntu4.9	2.4.7-1ubuntu4.14	CVE-2016-0736	Medium

SAST

ANÁLISE DE VULNERABILIDADES NO CÓDIGO

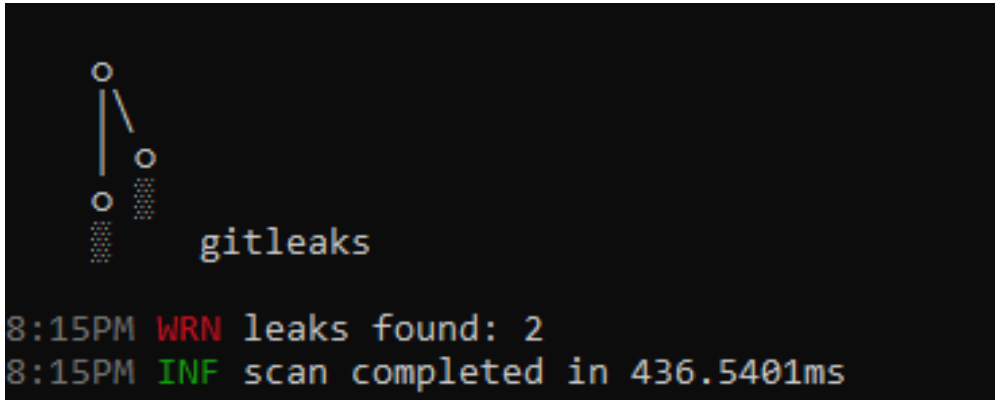
- SEMGREP
- INSIDER
- SHIFT-LEFT

ANÁLISE DE VULNERABILIDADES NO CÓDIGO

- SHIFT-LEFT
- sh <(curl https://slscan.sh)
- https://github.com/ShiftLeftSecurity/sast-scan

ENCONTRAR SEGREDOS

- SecretFinder
- python3 SecretFinder.py -i https://example.com/ -e
- GitLeaks
- docker run -v ${path_to_host_folder_to_scan}:/path zricethezav/gitleaks:latest [COMMAND] --source="/path" [OPTIONS]



SCA

ENCONTRANDO BIBLIOTECAS VULNERÁVEIS

- OWASP Dependency-check

DAST

ANÁLISE DAST – OWASP ZAP

- OWASP ZAP
- docker run --memory=1g -v $(pwd):/zap/wrk/:rw -t owasp/zap2docker-stable zap.sh -quickurl URL -quickout zapreport_APP_VERSION.html -cmd
- Command to start the ZAP api accessible by any host without requiring token:
- *nohup /opt/zaproxy/zap.sh -daemon -config api.addrs.addr.regex=true -config api.addrs.addr.name=.* -config api.disablekey=true -host 0.0.0.0 -port 9292 & > /dev/null*

ZAP Scanning Report

Generated with ZAP on Tue 19 Oct 2021, at 12:04:30

Contents

Summaries

Alert counts by risk and confidence

This table shows the number of alerts for each level of risk and confidence included in the report.

(The percentages in brackets represent the count as a percentage of the total number of alerts included in the report, rounded to one decimal place.)

		Confidence				
		User Confirmed	High	Medium	Low	Total
Risk	High	0 (0.0%)	0 (0.0%)	0 (0.0%)	0 (0.0%)	0 (0.0%)
	Medium	0 (0.0%)	0 (0.0%)	2047 (37.8%)	0 (0.0%)	2047 (37.8%)
	Low	0 (0.0%)	0 (0.0%)	1477 (27.3%)	1888 (34.9%)	3365 (62.2%)
	Total	0 (0.0%)	0 (0.0%)	3524 (65.1%)	1888 (34.9%)	5412 (100%)

DAST - JAELES

- JAELES
- Jaeles é uma estrutura poderosa, flexível e facilmente extensível escrita em Go para construir seu próprio Web Application Scanner.
- `jaeles scan -c 100 -s "cves,common,dns,fuzz,mics,probe,routines,sensitive" -u "demo.testfire.net"`

Exemplo de resultado:

```
Jaeles beta v0.17.0 by @j3ssiejjj
[Vulnerable][json-file-exposed][Potential]
[Vulnerable][common-route-01][Potential]
[Vulnerable][common-03-02][Medium]
[Vulnerable][cors-fuzz-01][Medium]
[Vulnerable][cors-fuzz-02][Medium]
[Vulnerable][Errors-Vulns-01][Critical]
[Vulnerable][dot-secret-no-ext][Potential]
[Vulnerable][git-leak-01][Medium]
[Vulnerable][scripts-file-exposed][Potential]
[Vulnerable][common-forbidden-bypass][Potential]
[Vulnerable][CVE-2019-7192][Medium]
[Vulnerable][common-directorylisting][Medium]
[Vulnerable][joomla-sqli-hdwplayer-01][High]
[Vulnerable][sensitive-secret-01][Potential]
```

DAST - JAELES

- JAELES ONE-LINER
- `cat domains.txt | anew | httpx -silent -threads 500 | xargs -I@ jaeles scan -c 100 -s "cves,common,dns,fuzz,mics,probe,routines,sensitive" -u @`

BUG BOUNTY TOOLS

- `wget -O - https://raw.githubusercontent.com/KingOfBugbounty/DockerHunt/main/install_hacktools.sh | bash`
- Credits to KingOfBugbounty
- https://github.com/KingOfBugbounty/
- https://github.com/KingOfBugbounty/KingOfBugBountyTips

DESCOBRIR ARQUIVOS E DIRETÓRIOS - DIRSEARCH

- Dirsearch
- `!dirsearch -x 302 -r --random-agent -u "http://demo.testfire.net/" -o report.txt`

DESCOBRIR ARQUIVOS E DIRETÓRIOS - DIRSEARCH

```
 _|. _ _  _  _  _ _|_    v0.4.2
(_||| _) (/_(_|| (_| )

Extensions: php, aspx, jsp, html, js
HTTP method: GET
Threads: 30
Wordlist size: 10903

Output File: /content/report.txt

Error Log: /usr/local/lib/python3.7/dist-packages/dirsearch/logs/errors-21-11-27_01-11-40.log

Target: http://demo.testfire.net/

[01:11:41] Starting:
[01:11:52] 400 -    0B  - /\..\..\..\..\..\..\..\..\..\etc\passwd
[01:11:52] 400 -    0B  - /a%5c.aspx
[01:12:07] 400 -    0B  - /faces/javax.faces.resource/web.xml?ln=..\\WEB-INF
[01:12:07] 200 -    8KB - /feedback.jsp
[01:12:10] 200 -    9KB - /index.jsp
[01:12:12] 200 -    8KB - /login.jsp
[01:12:19] 200 -    7KB - /search.jsp
[01:12:22] 200 -    1KB - /swagger/index.html

Task Completed
<dirsearch.dirsearch.Program object at 0x7f9d6704d790>
```

DESCOBRIR PARÂMETROS

- ARJUN
- ParamSpider

XSS

- Dalfox
- KXSS
- XSStrike
- XSSHunter

XSS

- XSStrike
- python xsstrike.py --params --fuzzer -u "http://demo.testfire.net/search.jsp?query=s"

```
XSStrike

[+] WAF Status: Offline
[!] Fuzzing parameter: query
[!] [passed]   <test
[!] [passed]   <test//
[!] [passed]   <test>
[!] [passed]   <test x>
[!] [passed]   <test x=y
[!] [passed]   <test x=y//
[!] [passed]   <test/oNxX=yYy//
[!] [passed]   <test oNxX=yYy>
[!] [passed]   <test onload=x
[!] [passed]   <test/o%00nload=x
[!] [passed]   <test sRc=xxx
[!] [passed]   <test data=asa
[!] [passed]   <test data=javascript:asa
[!] [passed]   <svg x=y>
[!] [passed]   <details x=y//
[!] [passed]   <a href=x//
[!] [passed]   <emBed x=y>
[!] [passed]   <object x=y//
[!] [passed]   <bGsOund sRc=x>
[!] [passed]   <iSinDEx x=y//
[!] [passed]   <aUdio x=y>
[!] [passed]   <script x=y>
[!] [passed]   <script//src=//
[!] [passed]   ">payload<br/attr="
[!] [passed]   "-confirm``-"
[!] [passed]   <test ONdBlcLicK=x>
[!] [passed]   <test/oNcoNTeXtMenU=x>
[!] [passed]   <test OndRAgOvEr=x>
```

BXSS - XSSHUNTER

- https://xsshunter.com/app

XSS - DALFOX

- cat urls.txt| dalfox pipe --mining-dom --deep-domxss --ignore-return -b 'bxssurlfromxsshunter' --follow-redirects -w 300 — multicast — mass — only-poc -o xss_vulns.txt

XSS - DALFOX

- cat liveurls.result | waybackurls | uro | gf xss | kxss | dalfox pipe --mining-dom --deep-domxss --ignore-return -b 'bxssurlfromxsshunter' --follow-redirects -w 300 — multicast — mass

XSS – ONE-LINER

- waybackurls testphp.vulnweb.com| grep '=' |qsreplace '"><script>alert(1)</script>' | while read host do ; do curl -s --path-as-is --insecure "$host" | grep -qs "<script>alert(1)</script>" && echo "$host \033[0;31m" Vulnerable;done

@HacktifyS

INJEÇÃO SQL

- SQLMAP
- ```
  sqlmap -u "http://demo.testfire.net" --thread=5 --random-agent --level=5 --
  risk=3  --batch --crawl=3
  ```
  ```

```
        ___
       __H__
 ___ ___[.]_____ ___ ___  {1.5.11#pip}
|_ -| . [.]     | .'| . |
|___|_  [)]_|_|_|__,|  _|
      |_|V...       |_|   https://sqlmap.org

[!] legal disclaimer: Usage of sqlmap for attacking targets without prior mutual consent is illegal. It is the end user's responsibility to obey all

[*] starting @ 01:46:33 /2021-11-27/

[01:46:33] [INFO] fetched random HTTP User-Agent header value 'Mozilla/5.0 (Windows; U; Windows NT 6.1; en-US) AppleWebKit/532.0 (KHTML, like Gecko)
do you want to check for the existence of site's sitemap(.xml) [y/N] N
[01:46:33] [INFO] starting crawler for target URL 'http://demo.testfire.net'
[01:46:33] [INFO] searching for links with depth 1
[01:46:34] [INFO] searching for links with depth 2
[01:46:34] [INFO] starting 5 threads
[01:46:36] [INFO] searching for links with depth 3
[01:46:36] [INFO] starting 5 threads
do you want to normalize crawling results [Y/n] Y
do you want to store crawling results to a temporary file for eventual further processing with other tools [y/N] N
[01:46:38] [INFO] found a total of 6 targets
[1/6] URL:
GET http://demo.testfire.net/index.jsp?content=inside_contact.htm
do you want to test this URL? [Y/n/q]
> Y
[01:46:38] [INFO] testing URL 'http://demo.testfire.net/index.jsp?content=inside_contact.htm'
[01:46:38] [INFO] using '/root/.local/share/sqlmap/output/results-11272021_0146am.csv' as the CSV results file in multiple targets mode
[01:46:38] [INFO] testing connection to the target URL
you have not declared cookie(s), while server wants to set its own ('JSESSIONID=7D63A013F33...BDCCC8EB1F'). Do you want to use those [Y/n] Y
[01:46:38] [INFO] checking if the target is protected by some kind of WAF/IPS
[01:46:39] [INFO] testing if the target URL content is stable
[01:46:39] [INFO] target URL content is stable
[01:46:39] [INFO] testing if GET parameter 'content' is dynamic
[01:46:39] [INFO] GET parameter 'content' appears to be dynamic
[01:46:39] [WARNING] heuristic (basic) test shows that GET parameter 'content' might not be injectable
[01:46:39] [INFO] heuristic (XSS) test shows that GET parameter 'content' might be vulnerable to cross-site scripting (XSS) attacks
[01:46:39] [INFO] testing for SQL injection on GET parameter 'content'
[01:46:39] [INFO] testing 'AND boolean-based blind - WHERE or HAVING clause'
[01:46:40] [WARNING] reflective value(s) found and filtering out
[01:46:53] [INFO] testing 'OR boolean-based blind - WHERE or HAVING clause'
[01:47:04] [INFO] testing 'OR boolean-based blind - WHERE or HAVING clause (NOT)'
[01:47:17] [INFO] testing 'AND boolean-based blind - WHERE or HAVING clause (subquery - comment)'
[01:47:26] [INFO] testing 'OR boolean-based blind - WHERE or HAVING clause (subquery - comment)'
[01:47:33] [INFO] testing 'AND boolean-based blind - WHERE or HAVING clause (comment)'
[01:47:35] [INFO] testing 'OR boolean-based blind - WHERE or HAVING clause (comment)'
[01:47:37] [INFO] testing 'OR boolean-based blind - WHERE or HAVING clause (NOT - comment)'
[01:47:40] [INFO] testing 'AND boolean-based blind - WHERE or HAVING clause (MySQL comment)'
```

CSP CHECK – ONE-LINER

- curl -v -silent https://$domain --stderr - | awk '/^content-security-policy:/'

CSP CHECK – ONE-LINER RESULTS

```
content-security-policy: default-src * data: blob:; script-src 'self' data: 'unsafe-inline' 'unsafe-eval' blob: https://js.hsadspixel.net https://up.pixel.ad
https://unpkg.com https://static.hotjar.com https://script.hotjar.com https://bat.bing.com https://cdn.jsdelivr.net https://www.trustradius.com
https://ssl.google-analytics.com https://www.storyqize.net https://cdn.storyqize.net https://s.yimg.com https://js.hs-scripts.com https://js.hs-analytics.net
```

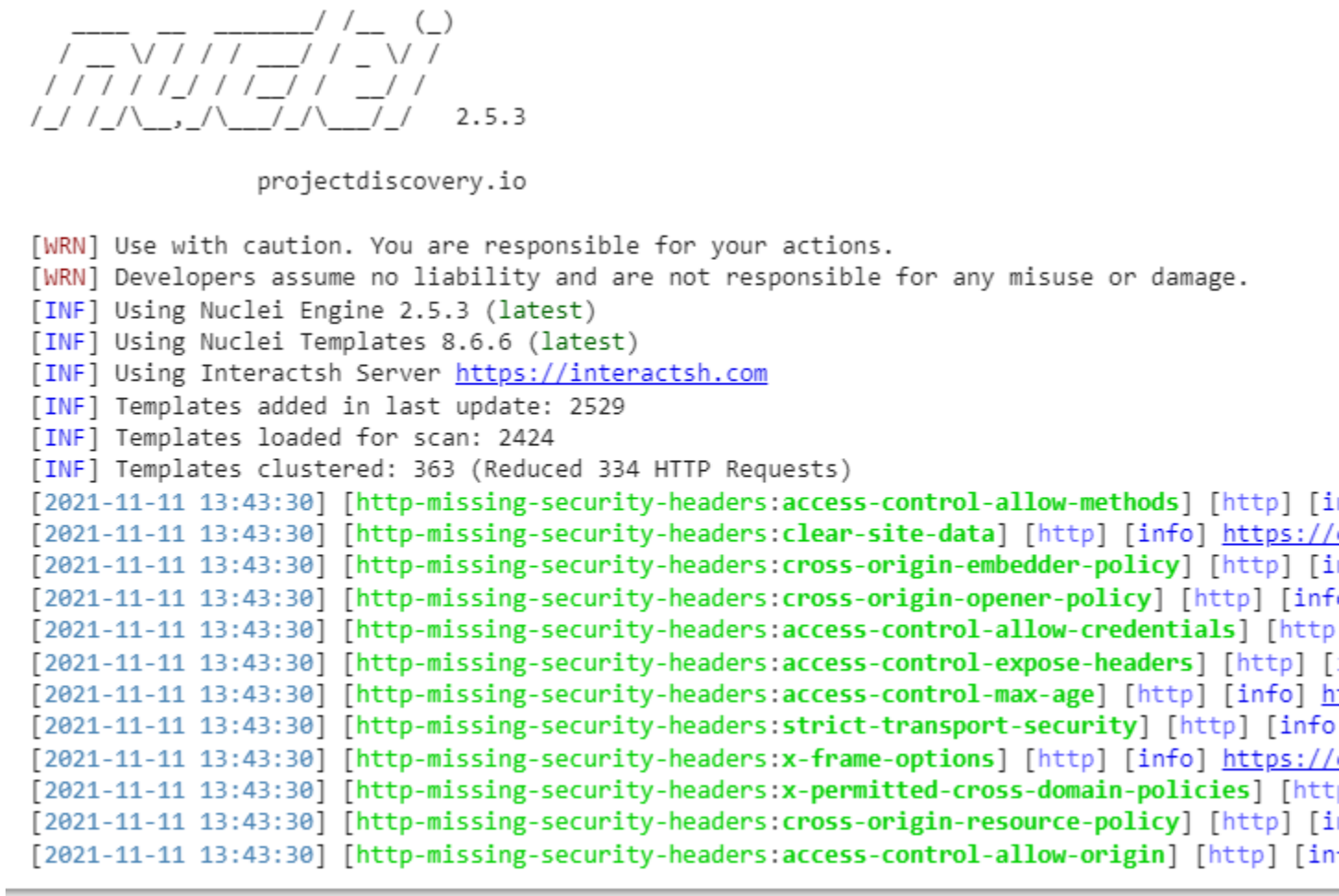
XSS-PROTECTION HEADER CHECK – ONE-LINER

- curl -v -silent https://www.google.com --stderr - | awk '/^x-xss-protection:/'

x-xss-protection: 0

ENCONTRAR VULNERABILIDADES CONHECIDAS

- NUCLEI
- `nuclei -u "http://demo.testfire.net" | tee -a "output.txt"`



ENCONTRAR VULNERABILIDADES CONHECIDAS

- TSUNAMI

INFRA AS A CODE

- Checkov
- docker run --tty --volume /projeto_iac:/tf bridgecrew/checkov --directory /tf

```
Check: CKV_AWS_58: "Ensure EKS Cluster has Secrets Encryption Enabled"
        FAILED for resource: AWS::EKS::Cluster.EKSCluster
        File: /eks.yaml:265-279
        Guide: https://docs.bridgecrew.io/docs/bc_aws_kubernetes_3

                265 |   EKSCluster:
                266 |     Type: AWS::EKS::Cluster
                267 |     Properties:
                268 |       Name: !Ref EKSClusterName
                269 |       RoleArn:
                270 |         "Fn::GetAtt": ["EKSIAMRole", "Arn"]
                271 |       ResourcesVpcConfig:
                272 |         SecurityGroupIds:
                273 |         - !Ref ControlPlaneSecurityGroup
                274 |         SubnetIds:
                275 |         - !Ref PublicSubnet01
                276 |         - !Ref PublicSubnet02
                277 |         - !Ref PrivateSubnet01
                278 |         - !Ref PrivateSubnet02
                279 |     DependsOn: [EKSIAMRole, PublicSubnet01, PublicSubnet02, PrivateSubnet01, PrivateSubnet02, ControlPlaneSecurityGroup]


secrets scan results:

Passed checks: 0, Failed checks: 2, Skipped checks: 0

Check: CKV_SECRET_2: "AWS Access Key"
        FAILED for resource: 25910f981e85ca04baf359199dd0bd4a3ae738b6
        File: /cfngoat.yaml:582-583
        Guide: https://docs.bridgecrew.io/docs/git_secrets_2

                582 |           access_key: "AKIAIOSFODNN7EXAMPLE"


Check: CKV_SECRET_6: "Base64 High Entropy String"
        FAILED for resource: d70eab08607a4d05faa2d0d6647206599e9abc65
        File: /cfngoat.yaml:583-584
        Guide: https://docs.bridgecrew.io/docs/git_secrets_6

                583 |           secret_key: "wJalrXUtnFEMI/K7MDENG/bPxRfiCYEXAMPLEKEY"
```

OBRIGADO!

- https://github.com/brinhosa
- https://github.com/brinhosa/awesome-pentest-tools-in-colab
- Ferramentas desta apresentação: shorturl.at/fvF58

REFERÊNCIAS

- https://github.com/KingOfBugbounty/KingOfBugBountyTips
- https://github.com/accurics/terrascan
- https://www.checkov.io/4.Integrations/Docker.html
- https://github.com/aws-samples/devsecops-cicd
- https://github.com/bridgecrewio/terragoat
- https://github.com/bridgecrewio/cfngoat
- https://github.com/aws-samples/devsecops-cicd
- https://github.com/zricethezav/gitleaks
- https://www.checkov.io/4.Integrations/Jenkins.html
- https://xsshunter.com/app
- https://github.com/brinhosa/awesome-pentest-tools-in-colab