**re:Invent 2019**

**ARC319**

**Security Vulnerability Identification and Remediation**

**Facilitator Notes**

[Introduction (Prerequisites) 2](#_Toc25755021)

[Problem Statement 2](#_Toc25755022)

[Solution Overview 3](#_Toc25755023)

[Task 1 – Enable AWS Config 3](#_Toc25755024)

[Task 2 – Deploy AWS Config rules & remediations 4](#_Toc25755025)

[Task 3 – Deploy 3 Tier 5](#_Toc25755026)

[Task 4 – Create insecure resources 5](#_Toc25755027)

[Task 5 – Manually create config rule 6](#_Toc25755028)

[Task 6 – Enable security hub 6](#_Toc25755029)

[Task 7 – Review AWS config deployed Rules 7](#_Toc25755030)

[Task 8 – Validate remediation 7](#_Toc25755031)

# Introduction (Prerequisites)

* Welcome Everyone
* Introduce yourself
* Recap the session number and title “ARC319 – Security Vulnerability Identification and Remediation”
* 300 level workshop, it is expected that you have experience working in the AWS console, if not don’t worry; detailed instructions and we’re here to help
* Mention pre-provisioned account (Provide Links)
* Provide Bit.ly for lab guide (bit.ly/awsarc319)

# Problem Statement

* Provide Overview of problem statement (devsecops engineer, mergers/acquisitions, bought blogging platform, IaC deployment)
* Overview of lab (read below)
  + This workshop focuses on integrating the new web platform and ensuring a proper security posture is maintained. The lab will involve learning how to monitor, alert, and remediate security events in your AWS environments; primarily focusing on [AWS Config](https://aws.amazon.com/config/) and [AWS Security Hub](https://aws.amazon.com/security-hub/).
  + How else could you use this? Evaluating 3rd party software? Taking hackathon projects to production? Just getting started with a foreign AWS environment?
* Read Disclaimer and ask for confirmation from attendees that they understand
  + **\*\*\*ARC319 will provide scripts and templates that intentionally create security holes, to be remediated. These templates should ONLY be deployed into temporary/sandbox AWS accounts and not into your corporate environment or anywhere with sensitive data.**

# Solution Overview

* Talk through architecture diagram and AWS services in scope
* Company has strict security posture
* Need to deploy this stack in a sandbox to validate security posture prior to deploying into production accounts
* Knows of some AWS services but new to AWS; mention services briefly but not that the lab focuses on Config and Security Hub as you have a defined list of rules you want to deploy in the sandbox.
* Ask how the architecture might be improved to be better, well architected
  + Route 53 for globally available DNS
  + Cloudfront distribution for S3
  + Caching solution for database reads
  + Additional Availability zones for resiliency

# Task 1 – Enable AWS Config

Goal – Enable AWS config = provable compliance at scale

* Have attendees enable AWS Config and walk through the steps in task 1
* What is Config?
  + AWS Config is a service that enables you to assess, audit, and evaluate the configurations of your AWS resources.
  + Config fits perfectly into a layered security model, always start with preventative controls and use services like Config for detective controls and ongoing compliance validation
  + Config continuously monitors and records your AWS resource configurations and allows you to automate the evaluation of recorded configurations against desired configurations.
  + This enables you to simplify compliance auditing, security analysis, change management, and operational troubleshooting.
  + We will be using Config today for identification and remediation of security configurations
* **Task 1 Checkpoint**
  + Should have config running, with an empty dashboard that looks like mine?
* Have attendees enable AWS Security Hub and walk through the steps in task 1
* Discuss what Security Hub is
  + Requires config
  + AWS Security Hub gives you a comprehensive view of your high-priority security alerts and compliance status across AWS accounts.
  + With all the different security tools available oftentimes this leaves your team switching back-and-forth between these tools to deal with hundreds, and sometimes thousands, of security alerts every day
  + With Security Hub, you now have a single place that aggregates, organizes, and prioritizes your security alerts, or findings, from multiple AWS services, such as Amazon GuardDuty, Amazon Inspector, and Amazon Macie, as well as from AWS Partner solutions.

# Task 2 – Deploy AWS Config rules & remediations

Goal – Use CF to create a set of config rules & remediations. We’ll dig into them later in the lab and we’ll als create one by hand in task 3.

* Email address is for notifications
* **Tip** – conformance packs were recently released to make it easier to manage sets of config rules at scale across your entire aws organization. conformance packs make it easy to deploy, remediate, and report collections of config rules.
* **Checkpoint** - refresh your config console, while CF deploying you'll start to see things appear, resources get inventoried, rules get created. start receiving emails.

# Task 3 – Deploy 3 Tier

Goal - launch the template given to us from the acquisition we were talking about at the beginning of the session

* Reiterate disclaimer
* **Checkpoint -**  takes 5 minutes, monitor in CF console
* Discuss vulnerabilities they see in the code, queue them in to lines mentioned in the lab guide

# Task 4 – Create insecure resources

Goal - intentionally create an insecure resource by hand (we did it with CF, too)

* Have the attendees open a new tab for the Security Hub service
* Discuss how you are starting to see certain findings and entries, based on CIS (Center for Internet Security) hardening guidelines
* Briefly discuss the CIS items that are populated in the ‘findings’ section; mention it helps you to know what you don’t know
* Click on the integrations section and discuss various integration options
  + GaurdDuty – Machine learning backed solution to identify common behavior and flag/alert when things are out of the norm
  + Inspector – Vulnerability assessment service with two different options (network based/agentless) and (host based/with Agent)
  + Macie – Machine learning backed service that identifies and notifies you of sensitive data in S3 and learns user behavior as it relates to S3.
  + 3rd party options (currently 22 different 3rd party solutions)
* Look at ‘Compliance Standards’
* Once looking at results explain these are the CIS controls
* Filter on S3
  + Mention the lack of a rule for bucket encryption, not there but it is a company policy
* Discuss how you could add a rule for encrypting S3 buckets and also leverage automated remediation (Config)
* **Checkpoint** - now should have 2 public s3 buckets. hopefully y'all can appreciate the number of hoops you had to go through to manually make a bucket public...

# Task 5 – Manually create config rule

Goal - learn the mechanics that make up a config rule

* Have attendees work through creating their own config rule
* Answer questions as necessary
* Note that it could take some time for the rule to evaluate
* **checkpoint** - should have a new rule in the evaluating state...

# Task 6 – Enable security hub

Goal - create our single pane of glass, ties in other services (3rd party, macie, guard duty) - critical to a good security strategy (layers, etc)

* **checkpoint** –
  + 1) securityhub\* config rules should be there. If they're not, disable/enable security hub again.
  + 2) security hub dashboard should show % complete compliance
  + 3) config dashboard should start looking a little more useful

# Task 7 – Review AWS config deployed Rules

Goal - answer the question - so what is our compliance posture? what world of hurt are we in if we merge this company's application as-is?

* task 7 tip - may need to go into an s3 rule, and re-evaluate to pick things up if your new bucket still isn't showing
* task 7 tip - if you "re-evaluate" too many rules/resources at once you may get an error "We are unable to complete the request at this time. Try again later or contact AWS Support." - this is a throttling error (available in the network details of the request)
* task 7 tip - for the S3\_BUCKET\_PUBLIC\_READ\_PROHIBITED rule, click into the rule, manually remediate if you'd like. Wait a few seconds then go to the resource - notice that it has removed public access but config still shows noncompliant - has to wait to inventory the object again
* task 7 checkpoint - should have an idea for relationship between resources, rules, and remediations are related. Understands the flow for a custom remediation inside of Lambda (SNS -> Lambda -> AWS Service)

# Task 8 – Validate remediation

Goal - validate that the remediations we have in place are taking place. review and understand why some things show as noncompliant.

* tip - change the config rule filter to "compliant" and pull up the securityhub-iam-root-access-key-check rule for the AWS account to see one that's fully compliant, but then reviewing the "compliance" timeline for the resource as a whole will show it as "non-compliant"
* tip - after 5-10 minutes, the s3 prohibit public read rule was fully compliant with all 3 buckets listed... just have to wait. Encourage exploring other rules, remediations, screens, security hub while waiting.