Identity Round Robin Workshop – External Security Services

Jeff Levine

Security & Compliance Solutions Architect

AWS Solutions Architecture

aws | Pop-up Loft

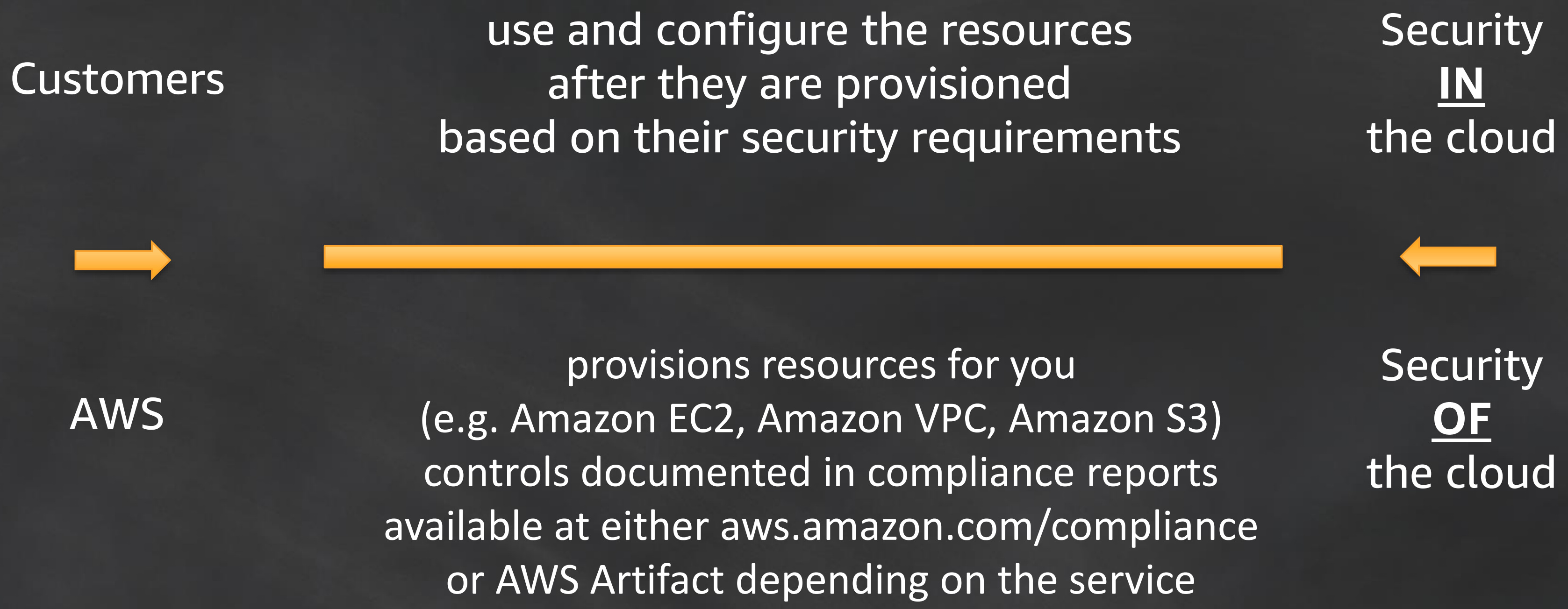
Who Is This Jeff Levine Guy?

- The first computer I used was an HP 3000 Series II
- I have used keypunchers and card sorters.
- My first programming language was FORTRAN.
- I came to AWS in 2016.
- I help customers with real world security issues.
- I write blog posts and whitepapers.
- I like to scuba dive!

Agenda

- Overview of the Shared Responsibility Model
- External Security Services
- Access Delegation
- Lab overview
- Q & A

Security is a Shared Responsibility



Customers

use and configure the resources
after they are provisioned
based on their security requirements

Security
IN
the cloud

AWS

provisions resources for you
(e.g. Amazon EC2, Amazon VPC, Amazon S3)
controls documented in compliance reports
available at either aws.amazon.com/compliance
or AWS Artifact depending on the service

Security
OF
the cloud

(the line of demarcation varies based on the service)

Here are some examples of what this means:

- When you ask AWS to provision an Amazon EC2 instance:
  - AWS provides an isolated instance and makes it available to you.
  - You decide what happens in the instance.
  - You control who can access the instance.

- When you ask AWS to provision an Amazon S3 bucket:
  - AWS provides an isolated S3 bucket.
  - You decide what goes into the bucket.
  - You control who can access the bucket.

In short:

Your data is *your data* and you decide who can access it.

Can AWS help me at a deeper level with security?

Yes, with your consent.

External Security Services

AWS offers security services that
with your consent
can get closer to your resources to give you more information.

Amazon GuardDuty – threat detection

Amazon Inspector – security assessment

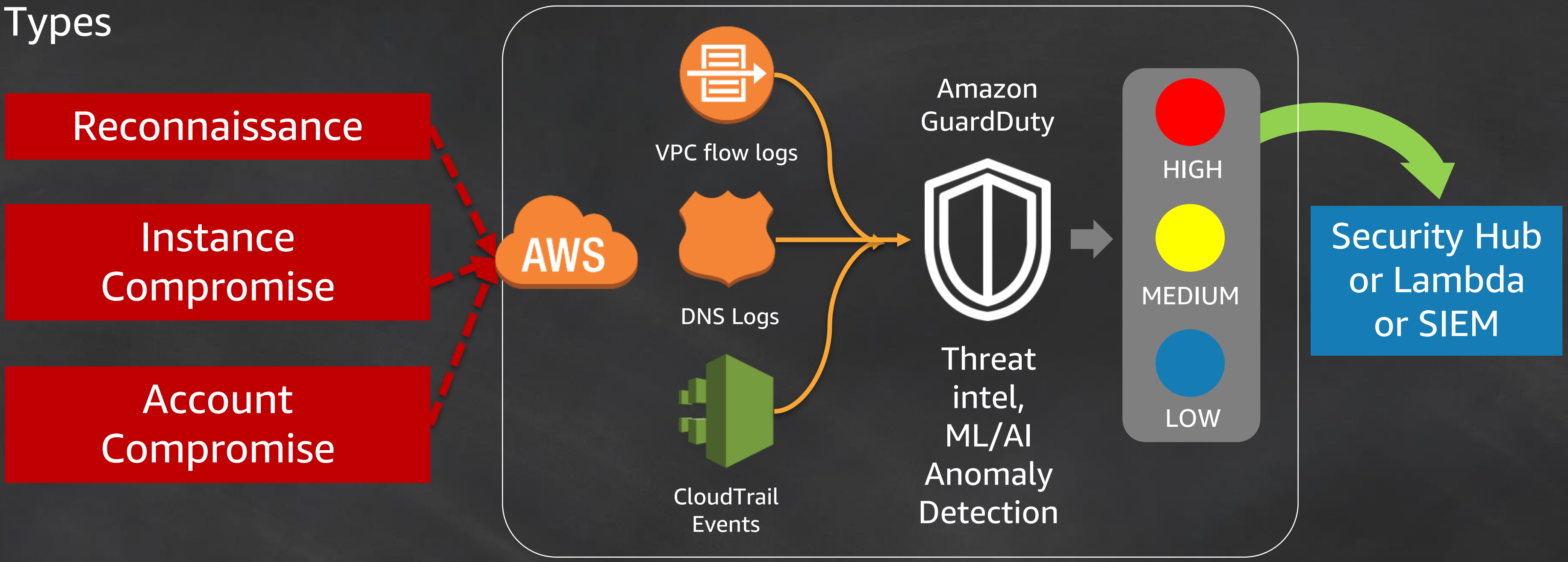
Amazon Macie – data classification

How GuardDuty Works

Threat Detection Types

Data Sources

Findings



Reconnaissance

Instance Compromise

Account Compromise

AWS

VPC flow logs

DNS Logs

CloudTrail Events

Amazon GuardDuty

Threat intel, ML/AI Anomaly Detection

HIGH

MEDIUM

LOW

Security Hub or Lambda or SIEM

Amazon Inspector

EC2 Host assessment

Host assessment rules packages

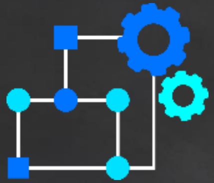
Using an Agent installed on EC2, Amazon Inspector can assess:

- Vulnerabilities in software (CVE)
- Host hardening guidelines (CIS Benchmark)
- AWS Security Best Practices.

Amazon Macie - Data Classification & Visibility

Amazon Macie uses machine learning-based classification of your Amazon S3 objects to provide visibility into your S3 environment. Macie can identify:

- PII – Names, credit card numbers, social security numbers, etc.
- programming languages to detect source code
- logging formats
- database backup formats
- credentials
- API key formats



Service Delegation

Using AWS IAM, you can delegate the capabilities of External Security Services to different classes of users. For example:

- Security Operators may need the ability to display a finding associated with a security service.

- Security Administrators may need complete access to the capabilities of a service.

AWS Identity and Access Management (IAM)

AWS Identity and Access Management (IAM) enables you to manage access to AWS External Security Services using principals, actions, and resources.

- An IAM principal is an actor.
  - An IAM user principal is a permanent, single entity, usually a person.
    An IAM role principal is assumed by a user or a service.

- In this workshop, we will deal with two roles:
  - A Security Administrator role has full access to security services.
  - A Security Operator role has read-only access to security services.
  - You will switch between these roles and test your access.

AWS Identity and Access Management (IAM)

- An IAM action corresponds to an AWS API call associated with an AWS service, for example: inspector:StartAssessmentRun

- Actions can be taken by:
  - The AWS console
  - The AWS CLI
  - AWS services
  - Applications

- Resources are AWS objects such as EC2 instances, S3 buckets, etc.

AWS IAM Policies

AWS Identity and Access Management (IAM) policies are combinations of principals, actions, and resources.

- Identity policies are attached to principals and define what the principal can do.
- Resource policies are attached to resources and define who can do what to a resource.
- Permission boundary policies are like a “fence” around identity policies.
- **In this workshop, we will only deal with identity policies.**

Policy types

- AWS-managed policies are supplied by AWS. You then attach these to users, groups, and roles to grant access (e.g. AmazonGuardDutyFullAccess).

- Customer-managed policies are created by you. You then attach these to users, groups, and roles to grant access. You can create a set of corporate policies and then re-use them. When you want to change the policy, you can update it in one location.

- Inline policies are added directly to a principal. You should generally not use these.

Policy types

- It is a best practice to places users into groups and use managed policies (either AWS-managed or Customer-managed) whenever possible. You can also attach policies to a role.

- AWS Config Rules can help you enforce group membership and also check for the use of managed policies:

- iam-user-group-membership-check

- iam-user-no-policies-check

Permission Delegation with External Security Services

The External Security Services support delegation through either managed or custom policies depending on the service. Each service has a documentation page entitled “Access Control in Amazon (servicename).”

- GuardDuty – Managed and custom policies
- Inspector – Managed and custom policies
- Macie – Managed (for full access) and custom (for read-only)

The AmazonGuardDutyReadOnlyAccess policy

```
{
        "Version": "2012-10-17",
        "Statement": [
                {
                        "Effect": "Allow",
                        "Action":  [
                                "guardduty:Get*",
                                "guardduty:List*“
                        ],
                        "Resource": "*“
                }
        ]
}
```

Permission Delegation with Roles

- You will be working with roles. A role consists of two components:
  - Policy definitions – the set of policies that describe what the role does
  - Trust relationships – the party that can assume a role.

Permission Delegation with Roles

- You can switch between roles to in the console to change your *effective* access.

- Role switching enforces the principle of *least privilege.* Principles should have the fewest privileges needed to perform their duties.

- For example, suppose Maria is a Security Administrator with many responsibilities. When she needs elevated privileges, she temporarily *assumes* the Security Administrator role in AWS and relinquishes the role when she is performing other duties.

- Role switching in AWS is similar to *sudo* in Linux.

Permission Delegation with Roles - YAML

```yaml
SecAdministratorRole:
   Type: AWS::IAM::Role
   Properties:
     AssumeRolePolicyDocument:
       Version: "2012-10-17"
       Statement:
         - Effect: "Allow"
           Principal:
             AWS:
             - 123456789012
           Action:
           - "sts:AssumeRole"
```

Permission Delegation with Roles - YAML

```
ManagedPolicyArns:
      - arn:aws:iam::aws:policy/AWSCloudTrailFullAccess
      - arn:aws:iam::aws:policy/AmazonGuardDutyFullAccess
      - arn:aws:iam::aws:policy/AmazonInspectorFullAccess
      - arn:aws:iam::aws:policy/AmazonSNSFullAccess
      - !Ref SecAdministratorMaciePolicy
```

Permission Delegation with Roles - YAML

```
SecAdministratorMaciePolicy:
   Type: AWS::IAM::ManagedPolicy
   Properties:
     Description: 'Policy for Security admin role'
     Path: '/'
     PolicyDocument:
       Version: '2012-10-17'
       Statement:
       - Effect: Allow
         Action: macie:*
         Resource: '*’
```

Lab Overview

- In the lab, you will experiment with IAM access delegation for the External Security Services as well as AWS CloudTrail.
- You will be given access to an AWS account.
- You will build an environment with CloudFormation that sets up Amazon GuardDuty, Amazon Macie, Amazon Inspector, and AWS CloudTrail.

Lab Overview

- The environment creates two roles:
  - A Security Administrator role with policies that grant full access to security services.
  - A Security Operator role.
    - Initially, the policies are similar to those of the Security Administrator role.
    - You will change the policies to provide read-only access to security services.

Lab Overview

- You will learn how to switch between roles to change your *effective* access.
- Role switching enforces the principle of *least privilege.*
- Role switching in AWS is similar to *sudo* in Linux.

Lab Overview

- Make sure you use only the us-west-2 (Oregon) region.

- The lab has two phases:
  - In the Build Phase, you will build the environment and configure the Security Operator role.   You will do some testing then turn over your credentials to another team who will do the verification.

  - In the Verify Phase, you will receive someone else’s credentials and then perform verification to ensure they did the lab properly.

Lab Overview

- When doing this in a team setting, remember that your're sharing an account.
- All resources in AWS are owned by an account.
- Consider dividing the steps up across the team and collaborating.

?

https://awssecworkshops.com

1. Click on Workshops
2. Click on Identity Round-Robin
3. Click on External Security Services

Final review

- You have learned to do the following:
  - Configure roles
  - Work with policies
  - Switch between roles

?

aws | Pop-up Loft

Everything and Anything Startups Need to Get Started on AWS

aws.amazon.com/activate