Security Best Practices

AWS Well-Architected

Sara Gray, Security Specialist

Public Sector

Security design principles

- Implement a strong identity foundation
- Enable traceability
- Apply security at all layers
- Automate security best practices
- Protect data in transit and at rest
- Keep people away from data
- Prepare for security events

Security Best Practice Areas

- Identity and access management
- Detective controls
- Infrastructure protection
- Data protection
- Incident response

Today's Security Approach

Preventative = P

Detective = D

Corrective = C

AWS Security Solutions

P	D	P+D	P+D	C
				
Identity	Detective control	Infrastructure security	Data protection	Incident response

AWS Security Solutions

P	D	P+D	P+D	C
				
Identity	**Detective control**	**Infrastructure security**	**Data protection**	**Incident response**
AWS Control Tower	AWS CloudTrail	AWS Systems Manager	AWS Key Management Service (KMS)	AWS Config Rules
AWS Identity & Access Management (IAM)	AWS Config	AWS Shield	AWS CloudHSM	AWS Lambda
AWS Directory Service	Amazon CloudWatch	AWS WAF – Web Application Firewall	Amazon Macie	
AWS Organizations	Amazon GuardDuty	AWS Firewall Manager	AWS Certificate Manager + Private CA	
AWS Secrets Manager	Amazon VPC Flow Logs	Amazon Inspector	Server-Side Encryption	
AWS Single Sign-On	AWS Security Hub	Amazon Virtual Private Cloud (VPC)	Encryption SDK	
Amazon Cognito				

aws

Poll: In which area are you the weakest or need the most improvement?

a. Identity and Access Control

b. Detective Control

c. Infrastructure Security

d. Data Protection

e. Incident Response

Lab prerequisites

- You will use your own AWS Account to build out the labs today
- You must have an AWS role with administrative privilege
- You must have the AWS CLI installed and configured
- It is your responsibility to delete any AWS resources after today to prevent ongoing costs!

Security Labs

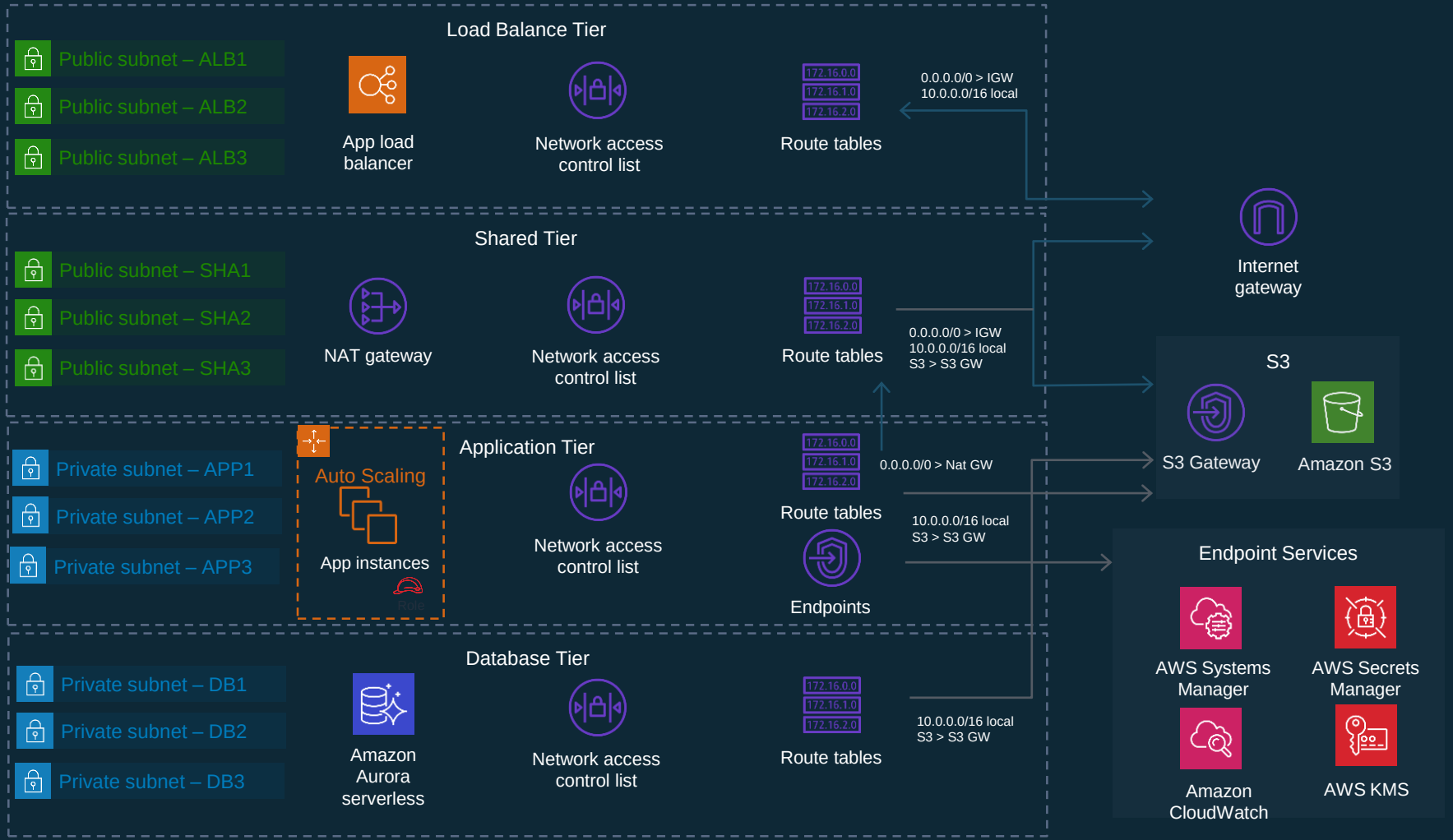
- **Preventative Controls**
  - Lab 1: Automated Deployment of VPC
- **Detective Controls**
  - Lab 2: Automated Deployment of Detective Controls
  - Lab 3: Enable Security Hub
- **Corrective Controls**
  - Lab 4: Incident Response with AWS Console and CLI

Preventative Controls Lab

Preventative Controls Lab

Lab 1: Automated Deployment of VPC

- http://bit.ly/2KzgMmS


Load Balance Tier
Public subnet – ALB1
Public subnet – ALB2
Public subnet – ALB3
App load balancer
Network access control list
172.16.0.0
172.16.1.0
172.16.2.0
Route tables
0.0.0.0/0 > IGW
10.0.0.0/16 local
Shared Tier
Public subnet – SHA1
Public subnet – SHA2
Public subnet – SHA3
NAT gateway
Network access control list
Route tables
0.0.0.0/0 > IGW
10.0.0.0/16 local
S3 > S3 GW
Internet gateway
S3
S3 Gateway
Amazon S3
Auto Scaling
Application Tier
Private subnet – APP1
Private subnet – APP2
Private subnet – APP3
App instances
Network access control list
Route tables
0.0.0.0/0 > Nat GW
10.0.0.0/16 local
S3 > S3 GW
Endpoints
Endpoint Services
AWS Systems Manager
AWS Secrets Manager
Amazon CloudWatch
AWS KMS
Database Tier
Private subnet – DB1
Private subnet – DB2
Private subnet – DB3
Amazon Aurora serverless
Network access control list
Route tables
10.0.0.0/16 local
S3 > S3 GW

Detective Controls Labs

Detective Controls Labs

Lab 2: Automated Deployment of Detective Controls

- http://bit.ly/31mpCLv

Detective Controls Labs

Lab 3: Enable Security Hub

- http://bit.ly/2MK1crq

Poll: In which do you find the most challenging for managing your logs?

a. Volume of log data to analyze
b. Ability to correlate logs between sources
c. Defining organizational log requirements
d. Timeliness to identify and respond to suspicious events
e. Configuration of service and application logging

Corrective Controls Lab

IR Principles

- Establish Goals
- Respond using the cloud
- Know what you have and what you need
- Do things that scale
- Use redeployment mechanisms
- Iteratively automate everything
- Learn and improve your process

Clean room

- Pre-provision access to workloads for security team
- Use tags to quickly determine impact and escalate
- Use AWS API operations to automate and isolate instances
- AWS CloudFormation – create clean environments for investigation

Poll: Do you leverage runbooks in your business?

a. No, I don’t see the need
b. No, but I’d like to
c. Yes, but not consistently and/or they are out of date
d. Yes, we use them consistently and keep them up to date

Runbooks!

Working Definition:

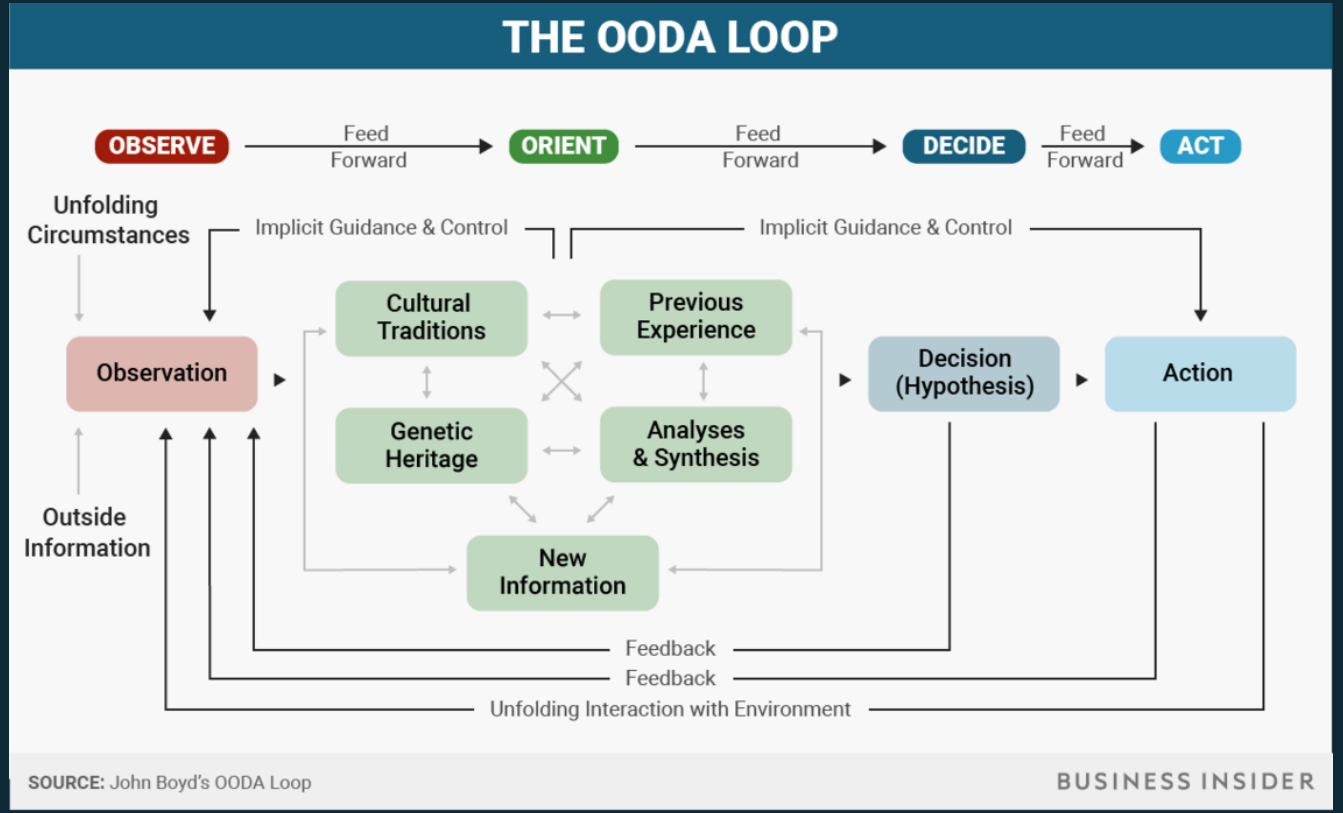
A way to have an employee actively and succinctly remediate an issue in an enterprise's infrastructure, application and/or service layer.

Wikipedia:

In a computer system or network, a runbook is a compilation of routine procedures and operations that the system administrator or operator carries out. System administrators in IT departments and NOCs use runbooks as a reference. Runbooks can be in either electronic or in physical book form. Typically, a runbook contains procedures to begin, stop, supervise, and debug the system. It may also describe procedures for handling special requests and contingencies. An effective runbook allows other operators, with prerequisite expertise, to effectively manage and troubleshoot a system. Through runbook automation ,these processes can be carried out using software tools in a predetermined manner.

aws

Runbook – OODA!



https://www.businessinsider.com/ooda-loop-decision-making-2017-8

GuardDuty + Planning (Run books * Game Days) * Partners = More Sleep

- This pattern holds regardless of product
- GuardDuty's importance is multiplied with CloudWatch, Config or custom Lambdas
- Notification and remediations allow you, the administrator to better meet uptime and DR goals
- Run books help with knowledge and training, but also to feel in control of a situation. Both as a coordinator and as an engineer.

Runbooks – Things to consider

1. Attribution – Catching or at least knowing who caused the incident
   - Speak to your legal counsel and follow their suggestion
   - What steps can you take to ensure chain of custody
2. Third party involvement e.g. regulator.
3. Forensics
   - List of tools
   - List of data
   - Reasons for use case, e.g. When does an incident need forensics
4. What timed procedures are being run, e.g. end of month
5. Review the ground rules that you have found, build these as your guard rails

Runbook Example

Problem description
[Your organization here] is under a [Attack Type]

[Attack Description]

Data to gather for troubleshooting
[Evaluation of current data.]

Steps to troubleshoot and fix
1.Log in to AWS
2.Do stuff
3.Correct Issue
4.Jump to forensics environment?

Urgency category
[Critical, Important, moderate, informational]
Escalation path / communication plan:
Unable to fix, escalate to these individuals or groups in this order:
1.Someone, email and phone number
2.Someone Else, email phone number
3.Distribution List/Slack?
4.CTO/CISO?
5.CEO?

Continuous Improvement

Reviewing the issues that occurred, and harden the application, infrastructure or procedures, so that the event can’t happen again.

Prevention vs Reaction

Least permissions

- Profiles
  - Lambda Functions
  - Containers
  - EC2
- Roles
- Users
- Everything!

Security as code

- Keep Humans away from the data
- Production is set apart, cleaner patterns means better threat detection.
- The pipeline is a no human zone

Incident Response Lab

Corrective Controls Lab

Lab 4: Incident Response with AWS Console and CLI

- http://bit.ly/2GR1vga

Teardown

Tear Down

In reverse order, go through each lab and remove resources using instructions.

Resources

AWS Well-Architected Labs

- https://wellarchitectedlabs.com/Security/

Comprehensive list of AWS security-focused content

- https://amzn.to/2SDTYV8