
aws
COMMUNITY DAY
SPAIN

NTT DATA

AWS SECURITY by DESIGN

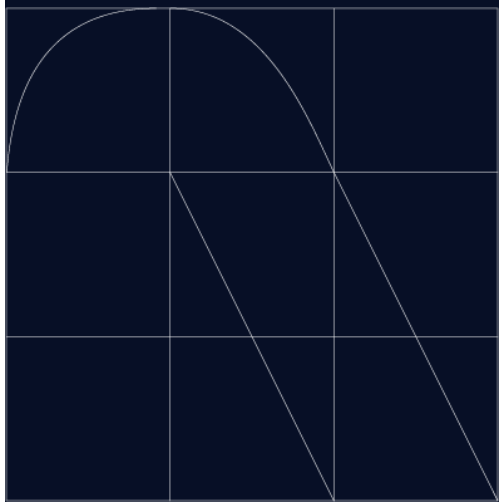
Alejandro Lazaro & Irene Aguilar

Cloud Architects, AWS Ambassadors, AWS Community Builders


SECURITY
LIGHTYEAR
SECURITY EVERYWHERE
imgflip.com

INDEX

NTT DATA



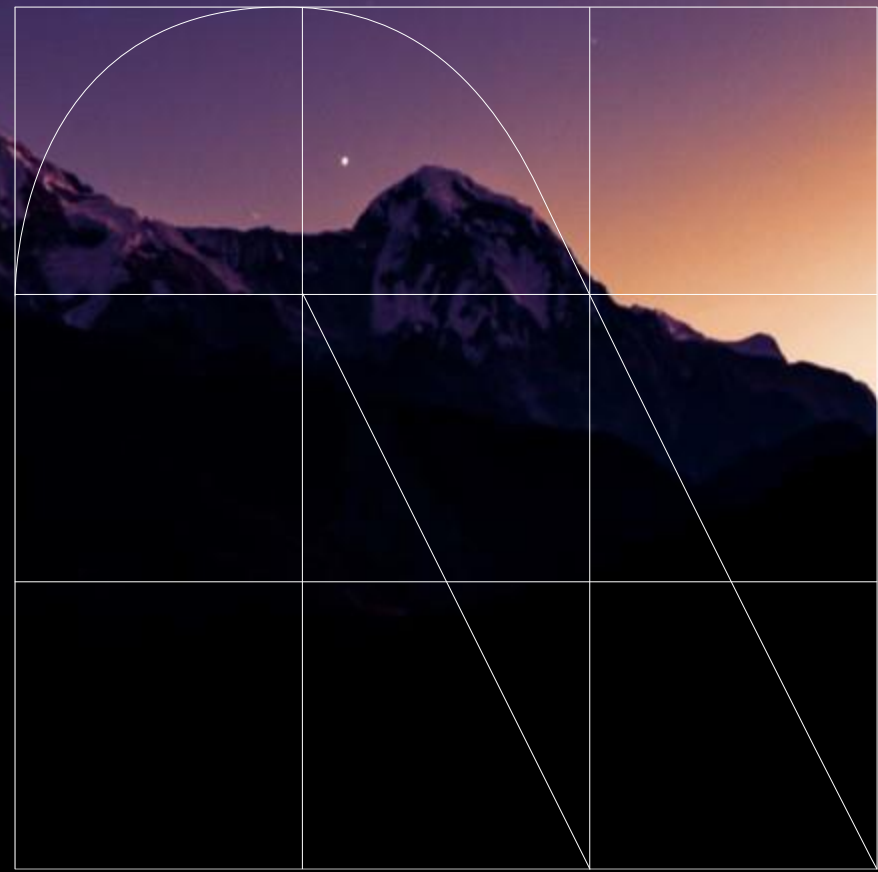
AWS Security Tools

NTT DATA

AWS Security Tools

Multiple tools to help us to design and provide security in the cloud:

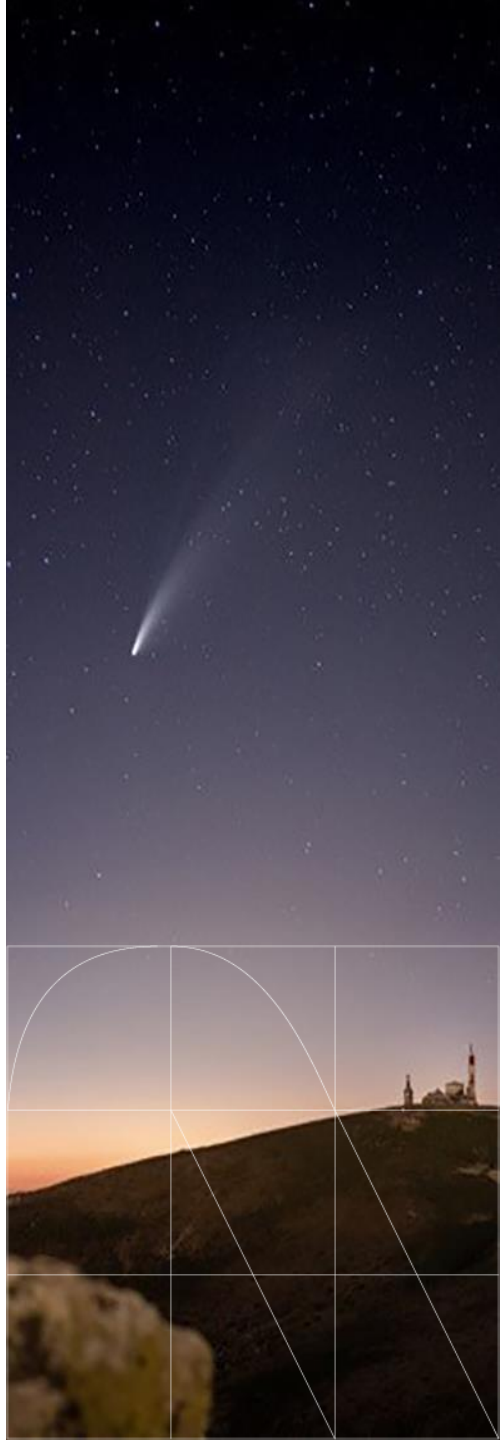
- Well Architected Framework
- Cloud Adoption Framework
- NIST Cybersecurity Framework
- Center for Internet Security (CIS) AWS Foundations
- AWS Security Checklist
- AWS Security Maturity Model



AWS Well-Architected Framework

AWS Well-Architected Framework describes key concepts, design principles, and architectural best practices for designing and running workloads in the cloud.

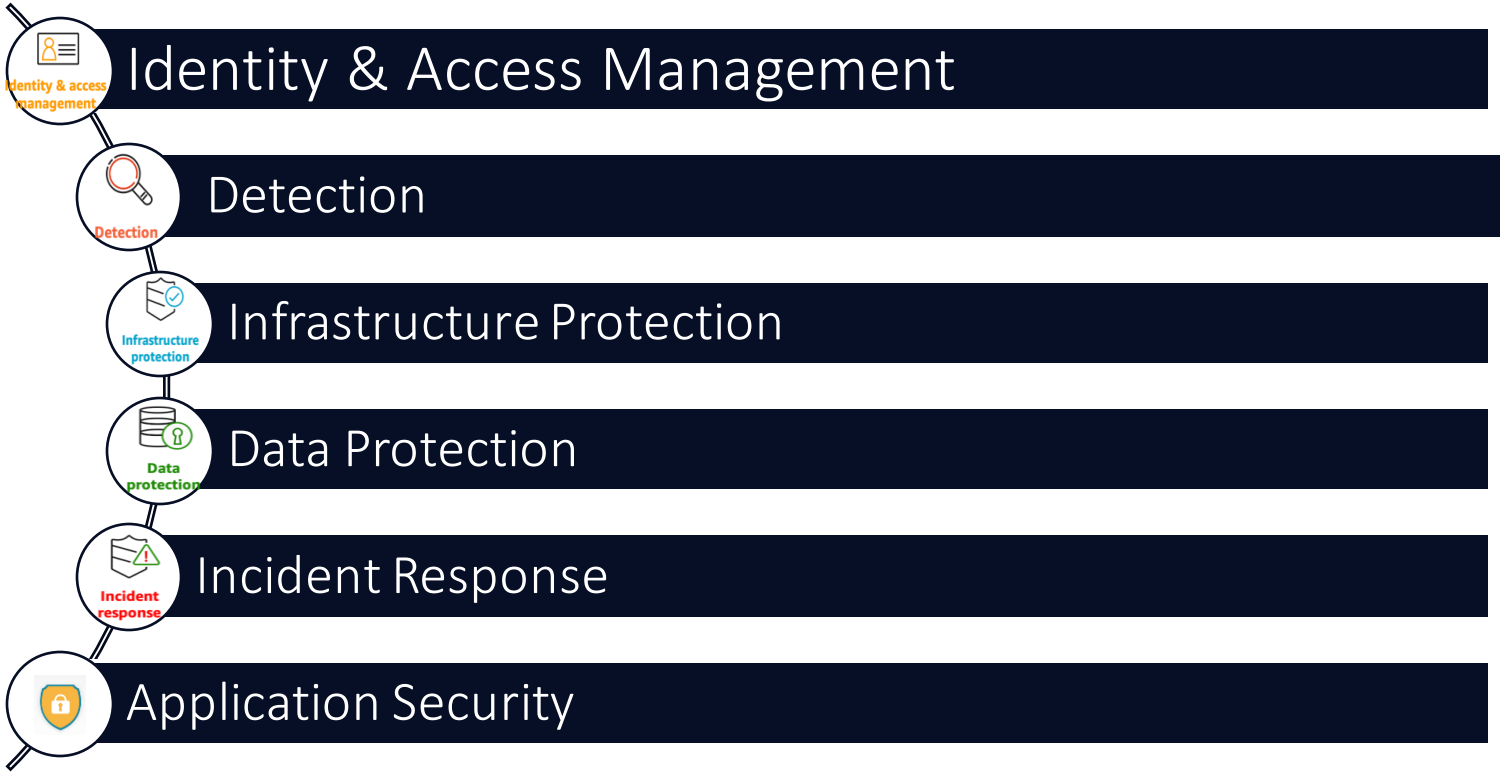
Composed of six main pillars:

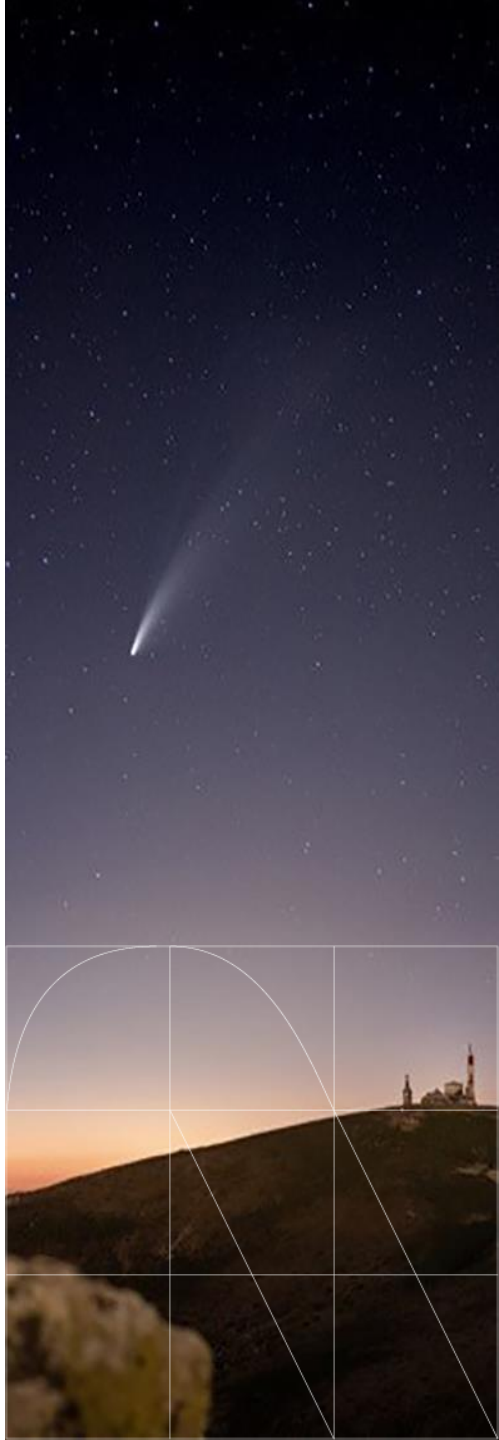


AWS Well-Architected Framework

Security Pillar provides guidance for secure AWS Workloads

Based on these areas:

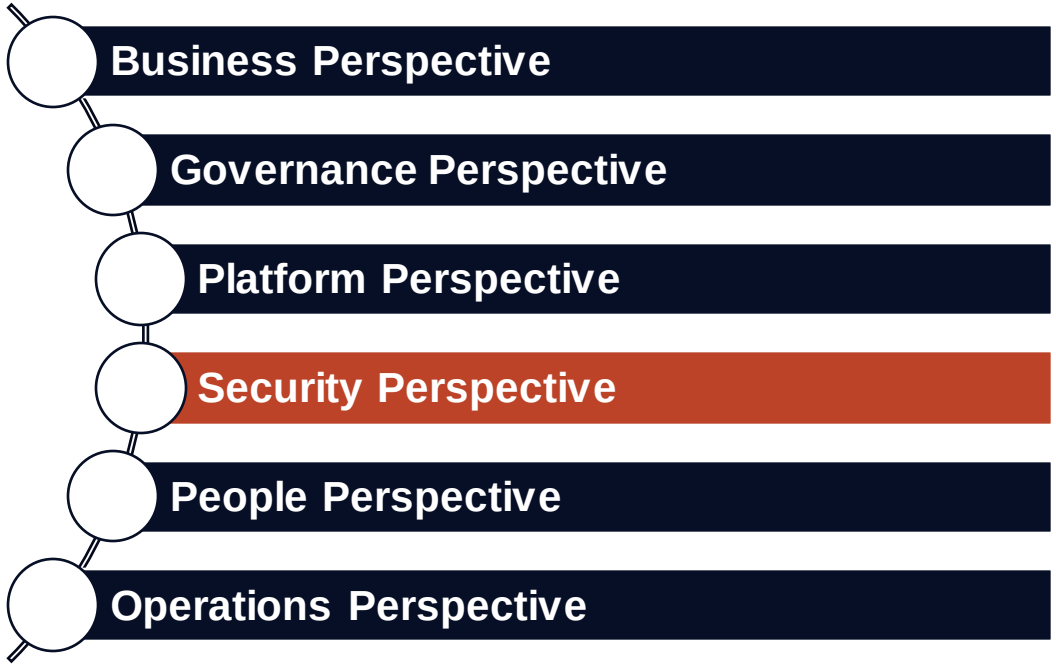




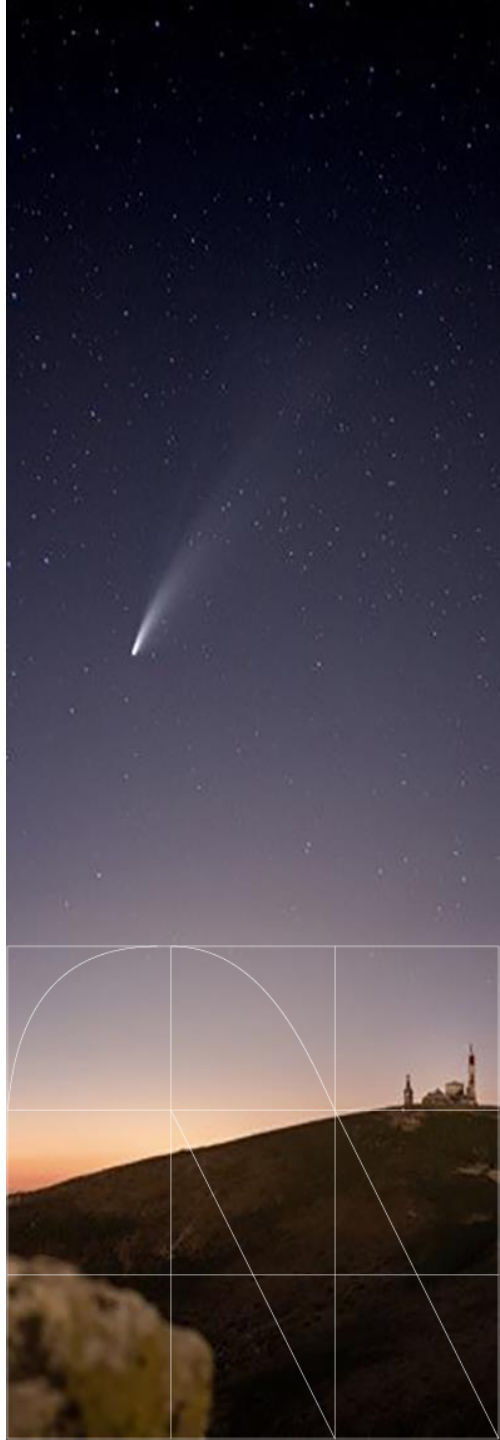
AWS Cloud Adoption Framework

AWS Cloud Adoption Framework is a guide provided by AWS for organizations to accelerate their cloud adoption journey while creating a structure that ensures business objectives are met.

Composed of six main perspectives:



Security Perspective
Security Governance
Threat Detection
Data Protection
Security Assurance
Vulnerability Management
Application Security
Identity and Access Management
Infrastructure Protection
Incident Response



NIST Cybersecurity Framework

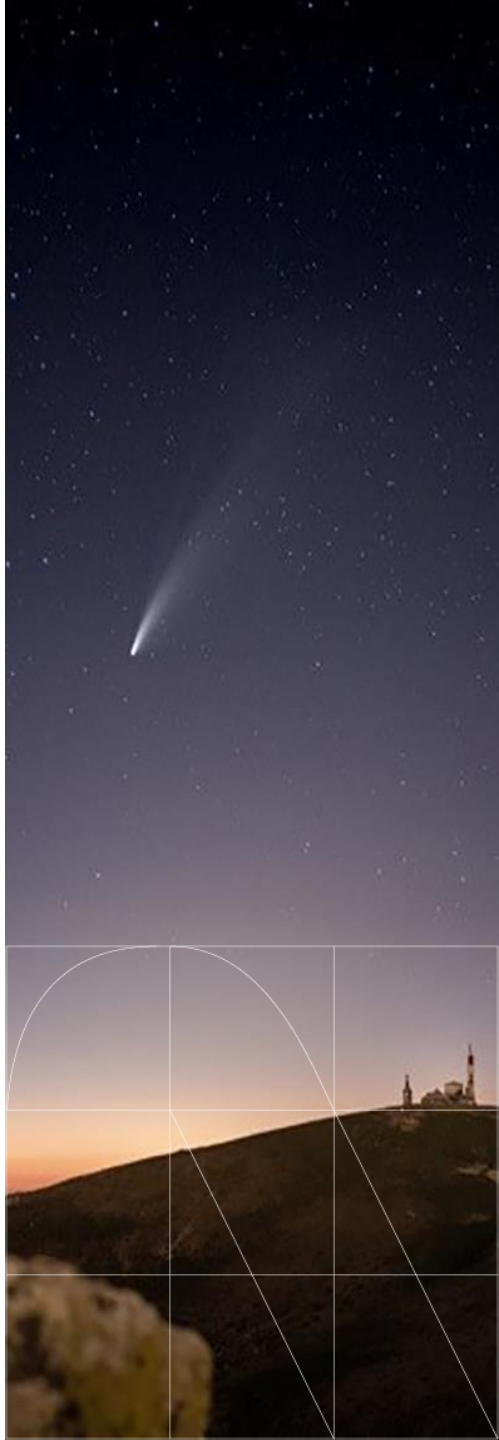
The NIST CSF provides a common language and a standardized methodology for managing cybersecurity risks.

Three most common scenarios:

- Used to evaluate an organization's cybersecurity posture and maturity
- Evaluate current and proposed products and services to identify capability gaps and opportunities.
- Restructuring security teams, processes, and training.

Focused on five key functions:

- Identify function
- Protect function
- Detect function
- Respond function
- Recover function



AWS CIS Foundations

The AWS CIS Foundations Benchmark is a set of security configuration best practices for users of AWS.

The AWS CIS Foundations Benchmark includes recommendations for securing AWS services across five areas:

Identity and access management
Logging and monitoring
Network security
Configuration management
Data protection

AWS Security Tools

	AWS WAF	AWS CAF	AWS CIS	NIST Cybersecurity
Area	Architecture	Migration (J2C)	Compliance	Risk Management
Objective	Help AWS customers design and operate secure, resilient, efficient, and cost-effective systems	Help organizations create a comprehensive approach to cloud adoption across their organization	The AWS CIS Foundations Benchmark is a set of security configuration best practices for users of AWS	Provides guidelines for securing resources on AWS across different industries
Security approach	• Identity & Access Management • Detection • Infrastructure Protection • Data Protection • Incident Response • Application Security	• Security Governance • Threat Detection • Data Protection • Security Assurance • Vulnerability Management • Application Security • Identity and Access Management • Infrastructure Protection • Incident Response	• Identity and access management • Logging and monitoring • Network security • Configuration management • Data protection	• Identify • Protect • Detect • Respond • Recover
Key Features	Provides a consistent approach to evaluating architectures, and includes a full section of security with design principles and best practices	Offers a structured approach to planning and implementing cloud adoption across different areas of an organization, with a full section for security	Provides prescriptive guidance for configuring AWS resources according to best practices	Framework for managing cybersecurity risk to critical infrastructure, including cloud environments
Security Considerations	Security is integrated into the framework	Security is part of adoption stages	Provides security configuration best practices for hardening AWS accounts and web applications running on AWS	Emphasizes cybersecurity risk management

How to start:

AWS Security Checklist

AWS Security Checklist

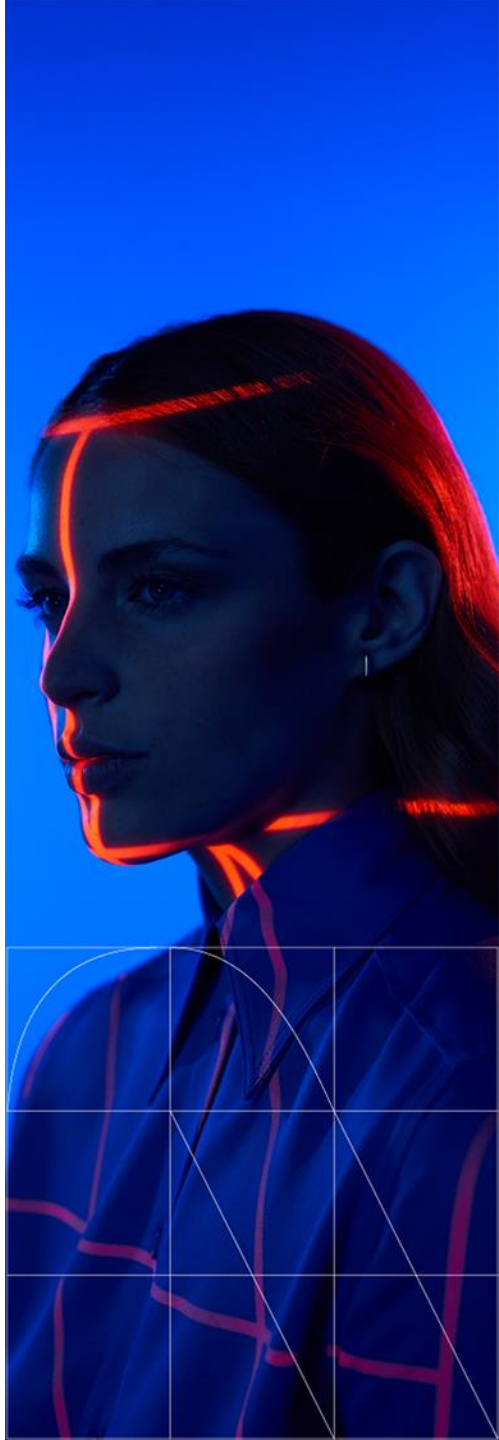
Identity management | Detection | Infrastructure Protection | Data protection | Incident response

How to improve:

AWS Security Maturity Model

Phase 1: Quick Wins

Level	Recommendation
Security governance	- Assign Security Contacts - Select the region(s)
Security assurance	- Automate alignment with best practices using AWS Security Hub
Identity and Access management	- Multi-Factor Authentication - Avoid using Root and audit it - Access and role analysis with IAM Access Analyzer
Threat detection	- Thread Detection with Amazon GuardDuty and review your findings - Audit API calls with AWS CloudTrail - Remediate security findings found by AWS Trusted Advisor - Billing alarms for anomaly detection
Vulnerability management	
Infrastructure protection	- Limit access using Security Groups
Data protection	- Amazon S3 Block Public Access - Analyze data security posture with Amazon Macie
Application security	- AWS WAF with managed rules
Incident response	- Act on Amazon GuardDuty findings



Complete Maturity Level

Access to the original page to view the table in the original site

CAF Category	Phase 1: Quick Wins	Phase 2: Foundational	Phase 3: Efficient	Phase 4: Optimized
Security governance	Assign Security contacts Select the region(s)	Identify security and regulatory requirements Cloud Security Training Plan	Perform threat modeling	Forming a Chaos Engineering team (Resilience) Sharing security work and responsibility
Security assurance	Automate alignment with best practices using AWS Security Hub	Configuration monitoring with AWS Config	Create your reports for compliance (such as PCI-DSS)	
Identity and access management	Multi-Factor Authentication Avoid using Root and audit it Access and role analysis with IAM Access Analyzer	Centralized user repository Organization Policies - SCPs	Privilege review (Least Privilege) Tagging strategy Customer IAM: security of your customers	Context-based access control IAM Policy Generation Pipeline
Threat detection	Threat Detection with Amazon GuardDuty and review your findings Audit API calls with AWS CloudTrail Remediate security findings found by AWS Trusted Advisor Billing alarms for anomaly detection	Investigate most Amazon GuardDuty findings	Integration with SIEM/SOAR Network Flows analysis (VPC Flow Logs)	Amazon Fraud Detector Integration with additional intelligence feeds
Vulnerability management		Manage vulnerabilities in your infrastructure and perform pentesting Manage vulnerabilities in your applications	Security Champions in Development	
Infrastructure protection	Limit Security Groups	Manage your instances with Fleet Manager Network segmentation - Public/Private Networks (VPCs) Multi-account management with AWS Control Tower	Image Generation Pipeline Anti-Malware/EDR Outbound Traffic Control Use abstract services	Process standardization with Service Catalog
Data protection	Amazon S3 Block Public Access Analyze data security posture with Amazon Macie	Data Encryption - AWS KMS Backups Discover sensitive data with Amazon Macie	Encryption in transit	
Application security	AWS WAF with managed rules	Involve security teams in development No secrets in your code - AWS Secrets Manager	WAF with custom rules Shield Advanced: Advanced DDoS Mitigation	DevSecOps Forming a Red Team (Attacker's Point of View)
Incident response	Act on Amazon GuardDuty findings	Define incident response playbooks - TableTop Exercises Redundancy using multiple Availability Zones	Automate critical and most frequently run Playbooks Automate deviation correction in configurations Using infrastructure as code (CloudFormation, CDK)	Automate most playbooks Amazon Detective: Root cause analysis Forming a Blue Team (Incident Response) Multi-region disaster recovery automation

Lessons Learned

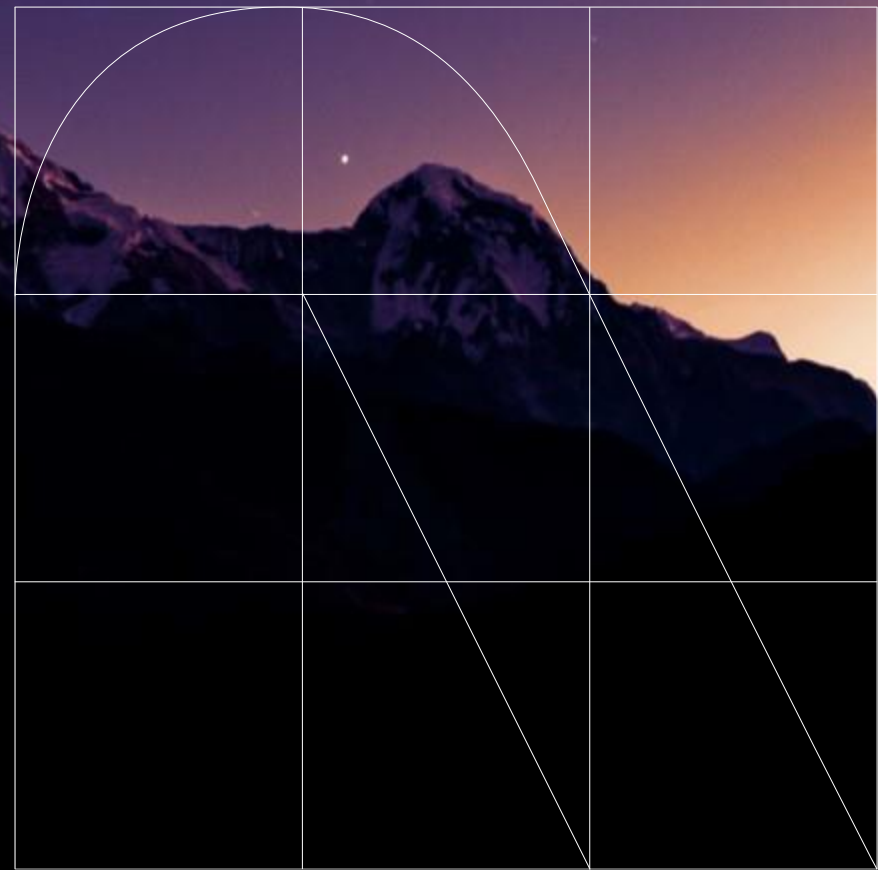
1. BEFORE anything else: know your client and your processes
2. AS-IS: AWS Security Checklist
3. TO-BE: AWS Security Maturity Model
4. Estimation → As always, don't be optimistic…
5. Start with Quick Wins
6. Don't try to fix everything (AWS best practices) → criticity
   - Hard requirements / MUST
   - Soft requirements / NICE TO HAVE
7. Be careful with AWS WAF
8. Security has a cost
9. Not only enable detection services. Monitor them and act on findings
10. Security: the sooner the better

Top 10 recommendations

Top 10 recommendations

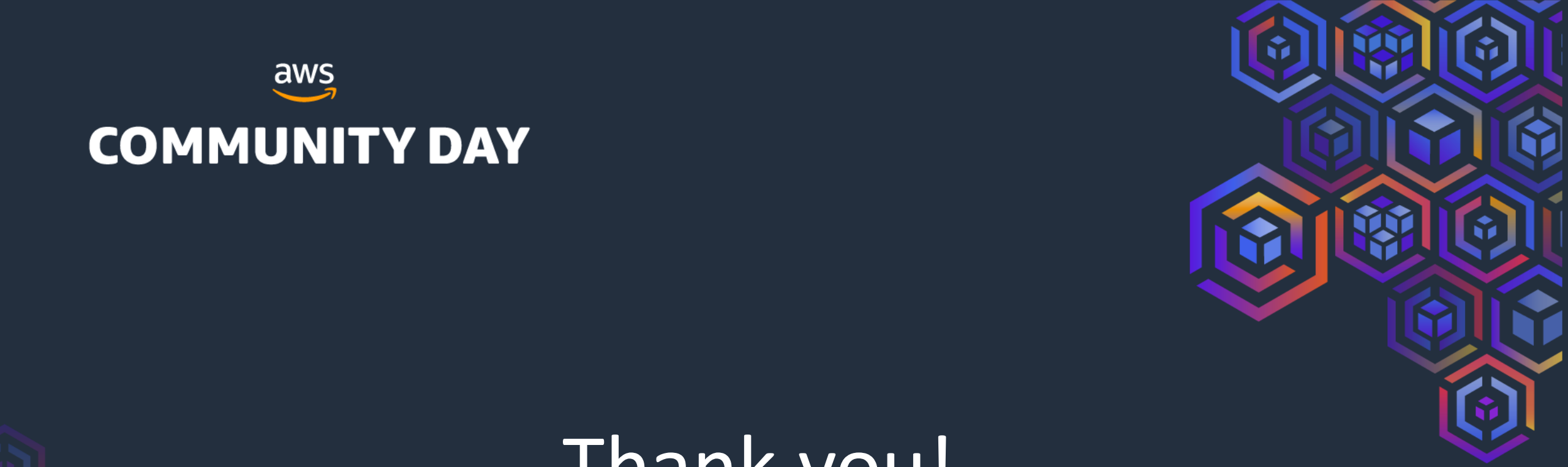
Most important cloud security tips:

1. Configure account contacts
2. Use multi-factor authentication (MFA)
3. No hard-coding secrets
4. Limit security groups
5. Intentional data policies
6. Centralize CloudTrail logs
7. Validate IAM roles
8. Take action on findings
9. Rotate keys
10. Be involved in the dev cycle



aws

COMMUNITY DAY



Thank you!

NTT DATA

aws

COMMUNITY DAY

NTT DATA