How to Think About Data Security in the Cloud

Eric Docktor
Vice President, AWS Cryptography

Ken Beer
General Manager, AWS Key Management Service

What the Cloud Offers For Security

- Improves security for nearly all customers
- Delivers unprecedented visibility and control
- Simplifies the work of security and compliance
- Enables agility and speed through automation

aws

Building blocks for your workload

- Facilities
- Physical security
- Compute infrastructure
- Storage infrastructure
- Network infrastructure
- Virtualization layer
- Hardened service endpoints
- Rich API capabilities

- Network configuration
- Security groups
- OS firewalls
- Operating systems
- Application security
- Proper service configuration
- Authentication & account management
- Authorization policies

In Cloud, Security Is a Shared Responsibility

Customers concentrate on their stack while cloud provider manages infrastructure.

- Facilities
- Physical security
- Compute infrastructure
- Storage infrastructure
- Network infrastructure
- Virtualization layer
- Hardened service endpoints
- Rich API capabilities

+

- Network configuration
- Security groups
- OS firewalls
- Operating systems
- Application security
- Proper service configuration
- Authentication & account management
- Authorization policies

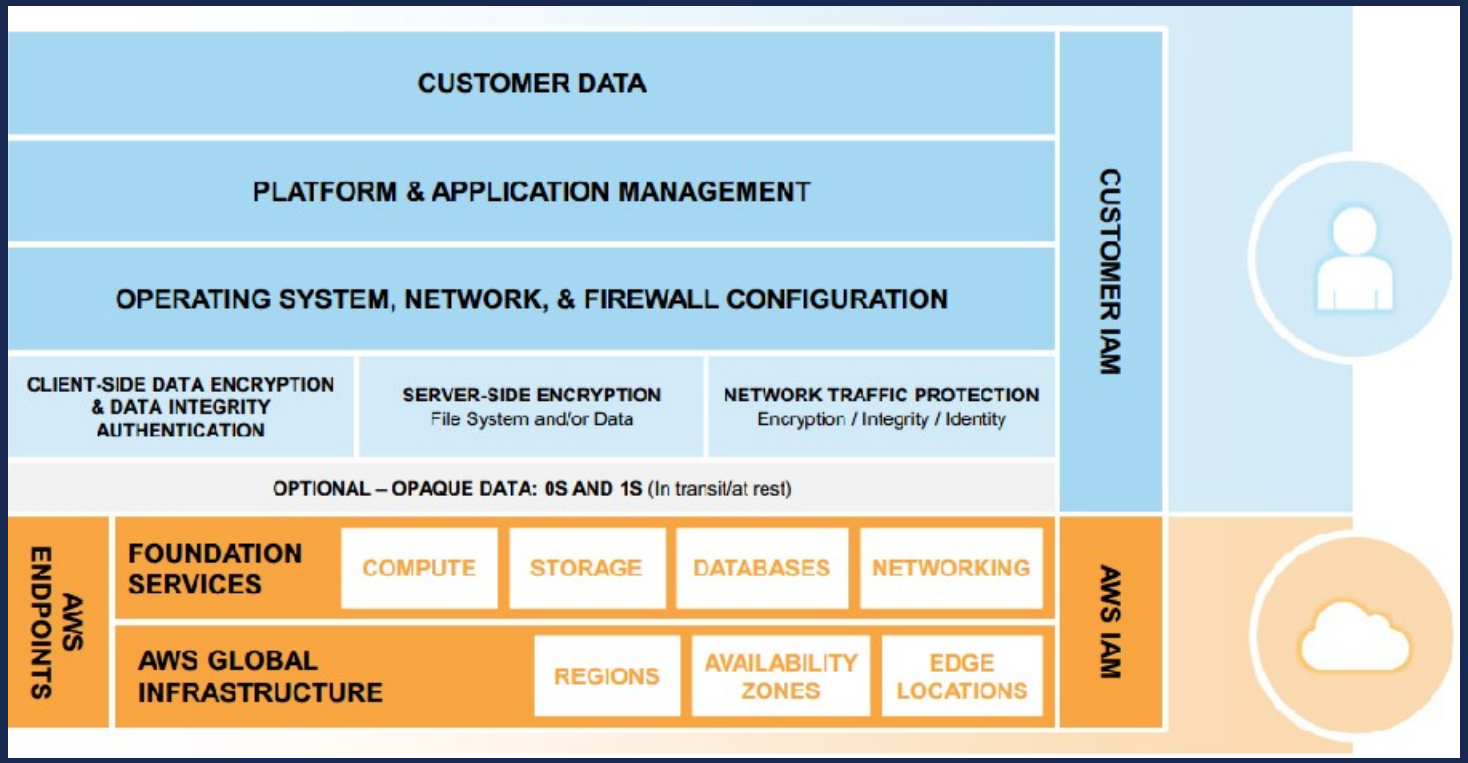
=

More secure and compliant systems than any single entity could normally achieve on its own

This allows your security team to focus on a subset of overall security needs that pertain directly to your data

aws

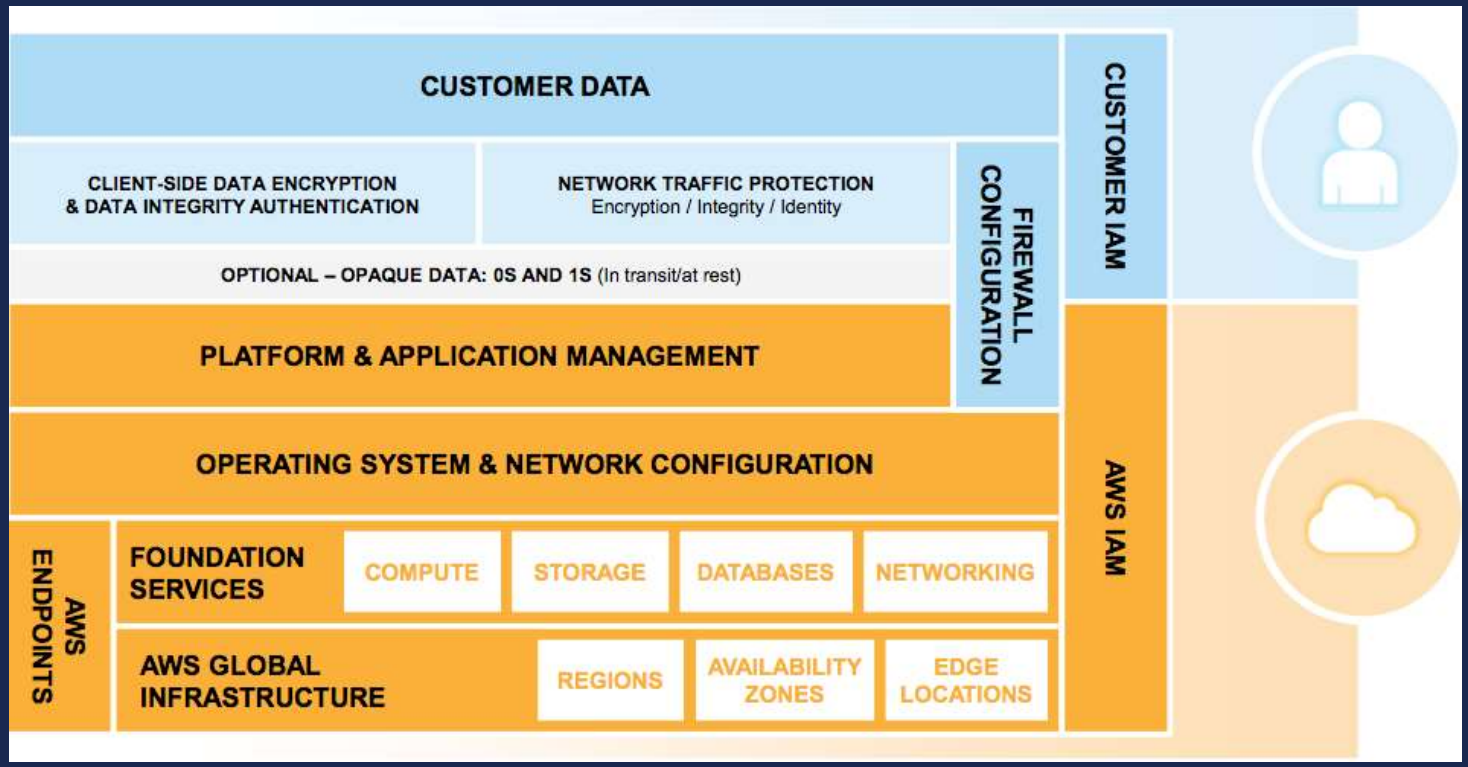
Shared Responsibility Model - Infrastructure



Managed by AWS Customers

Managed by Amazon Web Services

Shared Responsibility – Platform Services



Managed by AWS Customers

Managed by Amazon Web Services

Global Compliance Schemes

aws

Global Compliance Schemes

Leverage the work your provider has already done

Certifications / Attestations	Laws, Regulations, and Privacy	Alignments and Frameworks
C5 [Germany]	CISPE	CIS
Cyber Essentials Plus [UK]	DNB [Netherlands]	CJIS
DoD SRG	EU Model Clauses	CSA
FedRAMP	FERPA	ENS [Spain]
FIPS	GLBA	EU-US Privacy Shield
IRAP [Australia]	HIPAA	FISC [Japan]
ISO 9001	HITECH	FISMA
ISO 27001	IRS 1075	G-Cloud [UK]
ISO 27017	ITAR	GxP (FDA CFR 21 Part 11)
ISO 27018	My Number Act [Japan]	ICREA
MLPS Level 3 [China]	U.K. DPA - 1988	IT Grundschutz [Germany]
MTCS [Singapore]	VPAT / Section 508	MITA 3.0
PCI DSS Level 1	EU Data Protection Directive [EU]	MPAA
SEC Rule 17-a-4(f)	Privacy Act [Australia & New Zealand]	NIST
SOC 1	PDPA - 2010 [Malaysia]	PHR
SOC 2	PDPA - 2012 [Singapore]	Uptime Institute Tiers
SOC 3	PIPEDA [Canada]	UK Cloud Security Principles
	Spanish DPA Authorization	

aws

Applying the Shared Responsibility Model

Security of the cloud

- Security measures the provider implements and operates
- Provider's security standards shown by certifications & attestations

Security in the cloud

- Security measures that the **customer** implements and operates
- **Certifications** and a**ttestations** can be used by customers when undertaking risk assessments or using **frameworks**

aws

Security Controls You Define and Operate

Access Control

- You control who can do what to which resources under which conditions

Visibility-Audit-Remediation

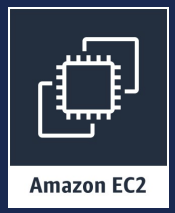
- No hidden resources – everything discoverable via an API

Automate

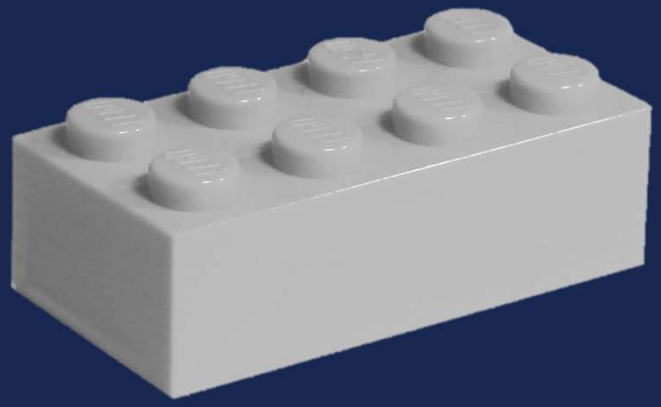
- Security through code
- Enforce the use of templates – no cowboy code

aws

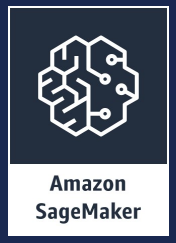
Building Blocks: Starting Small



Compute



Logging



AI/ML

Is it secure?

Storage

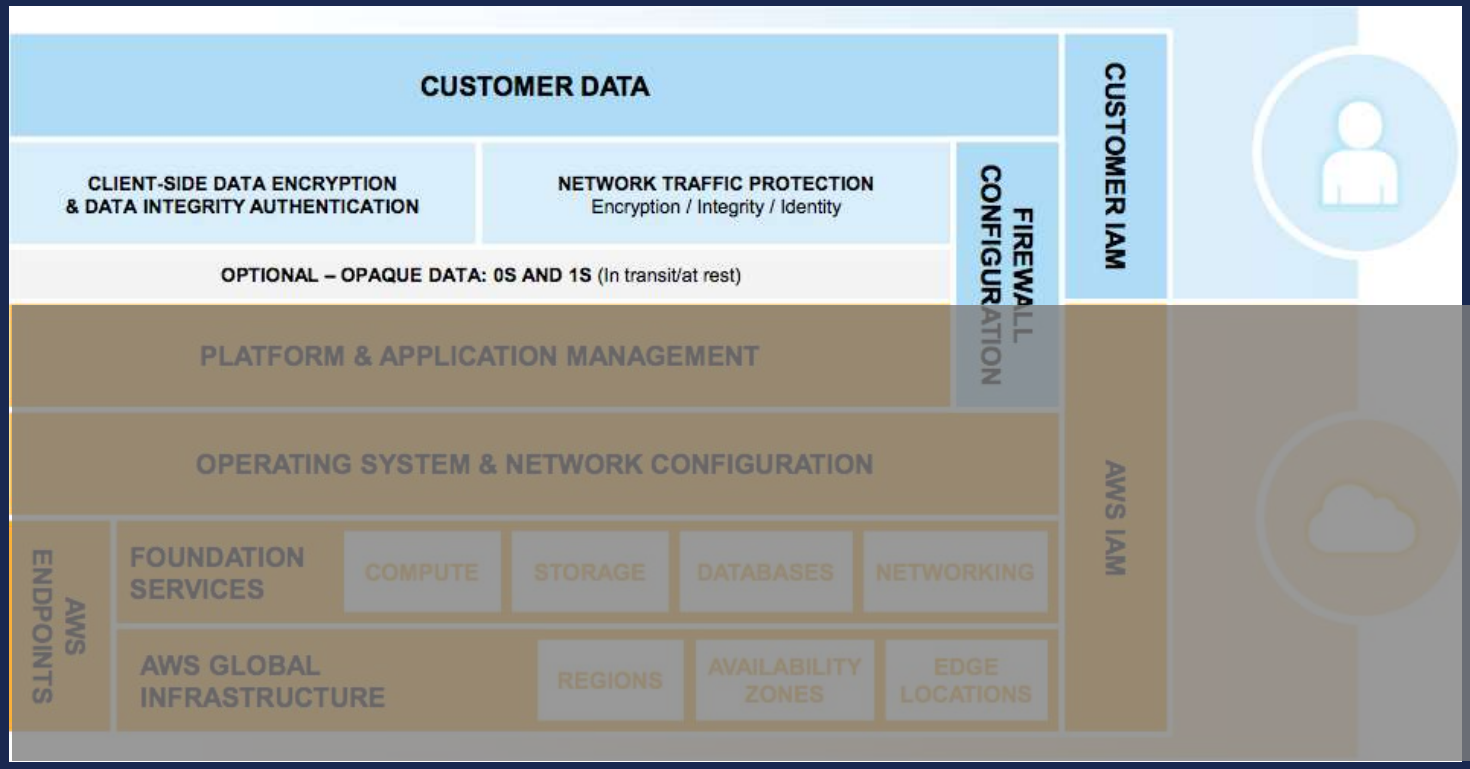
aws

Building Blocks: Finished Application

Re-using secure blocks minimizes chance of data breach across the entire workload

Securing Your First Workload

Shared Responsibility – Platform Services



Managed by AWS Customers

Managed by Amazon Web Services

Cloud Access Control – Expressing Policy

Identity

Must be authenticated by provider, may be federated from your network

Effect

Allow vs. Deny

Action/Scope

API to create, read, update, or delete a resource

Resource

Instance, storage, database, networking group, identity, encryption key, etc.

Condition

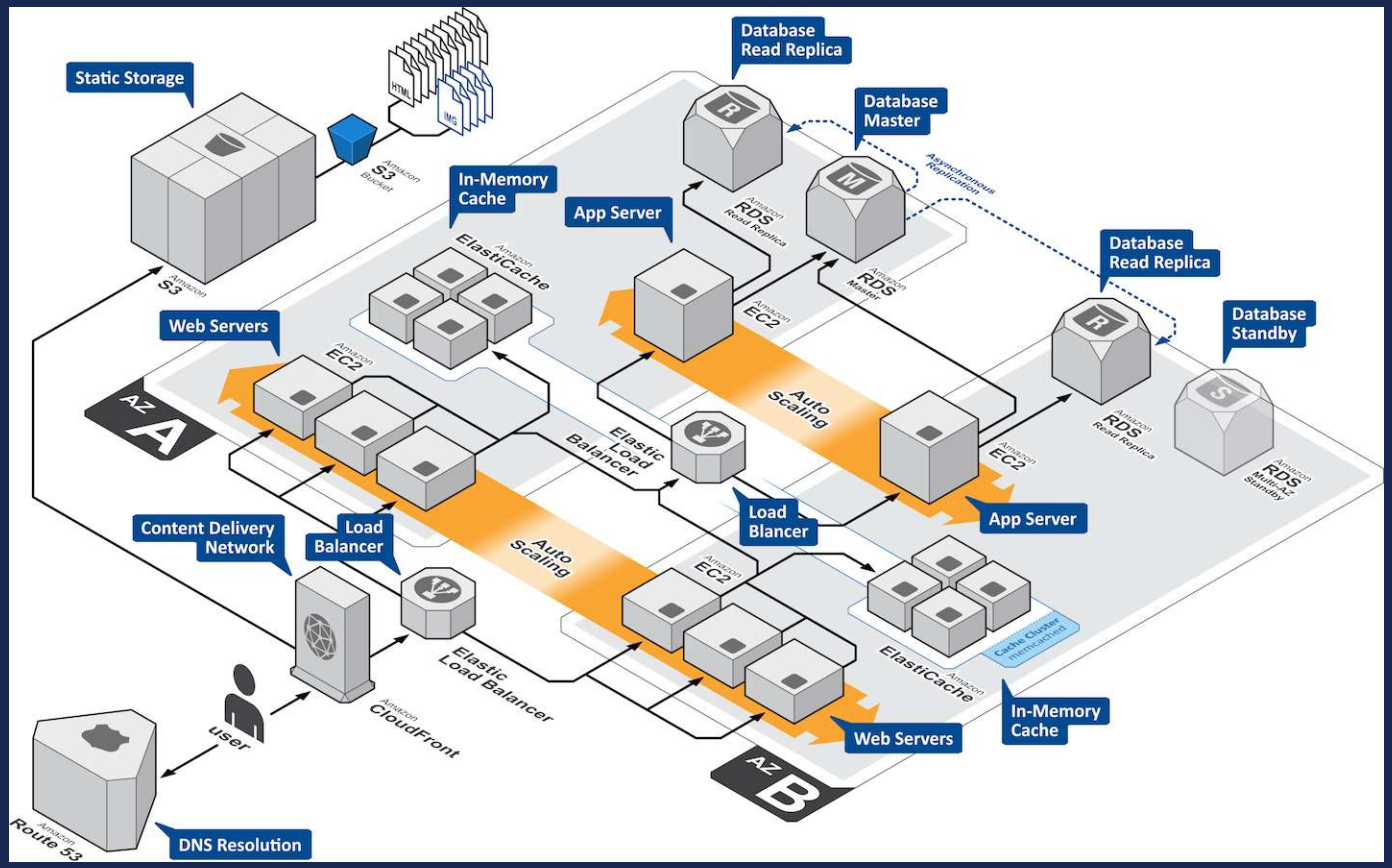
SourceIP, time, MFA, custom metadata in API request, etc.

Cloud Access Control – Policy Requirements

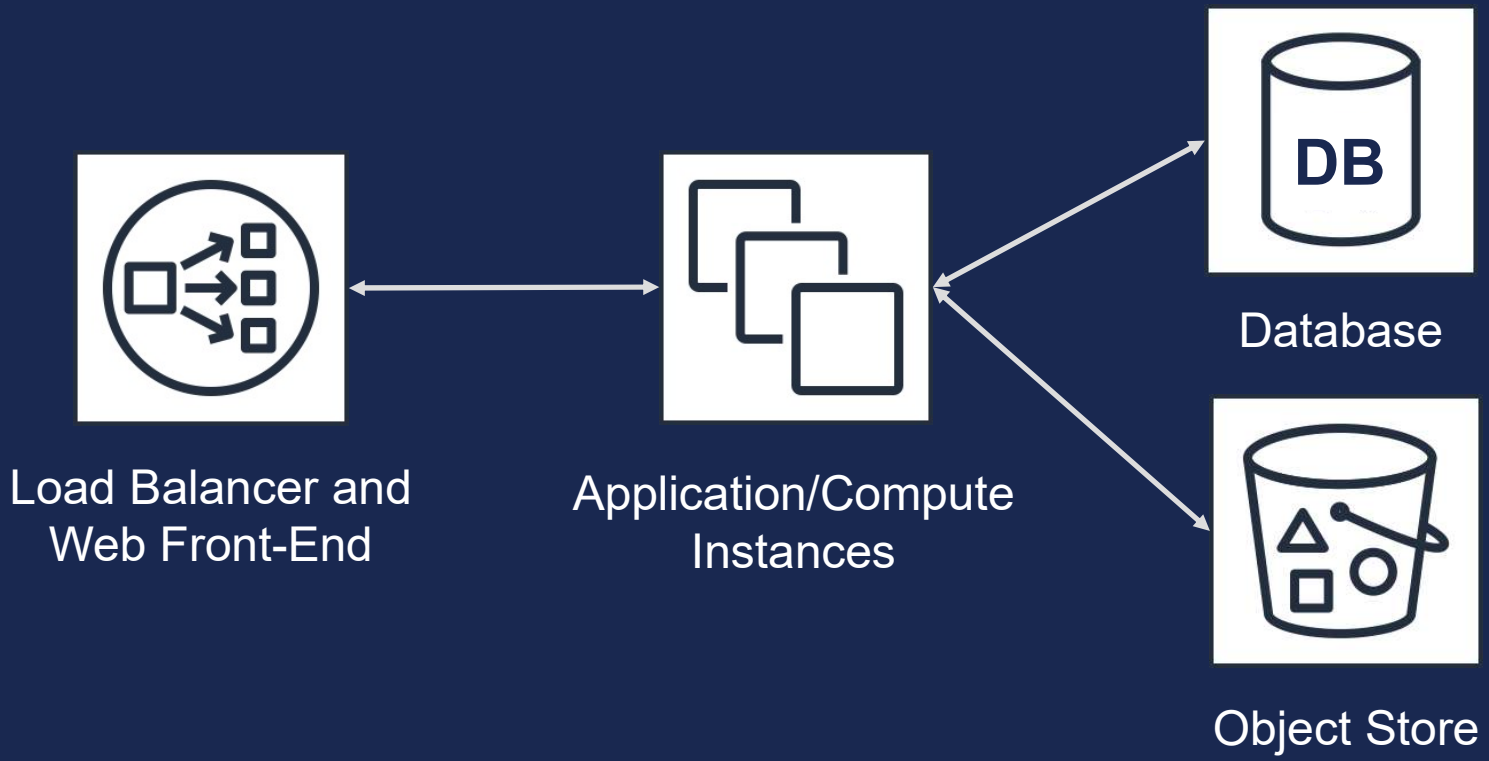
- Human-readable
- Flexible semantics
- Consistent across cloud provider's services
- Access control on access control
  - Only privileged users can create/edit policy
  - No escalation of privilege

aws

Where Your Architecture Might Be Headed



Applying Access Control to A Sample Workload



Load Balancer and Web Front-End

You own:

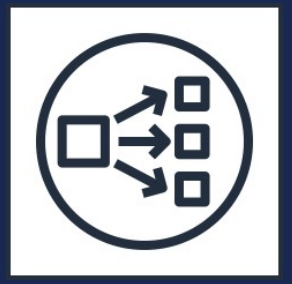
- Routing rules
- Protocols/Ports
- Targets behind load balancer

Cloud Provider owns:

- Availability, throughput, host security

Your choice:

- TLS config
  - Generate/store certificates?
  - Automate certificate rotation?



Load Balancer and Web Front-End

aws

Application/Compute Instances

You own:

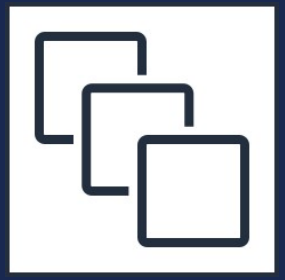
- Protocols/Ports
- CPU/Memory/Storage size
- OS-level authentication
- Automatic scaling rules
- Application logs

Cloud Provider owns:

- Availability, IOPS, host security

Your choice:

- Operating system deployment/patching?



Application Instances

aws

Data Storage

You own:

- Storage size
- Access rules

Cloud Provider owns:

- Availability, durability, host security, logs

Your choice:

- DB engine?
- Storage type (block, file, object)?

Database

Object Store

aws

Sample Access Policy - Who Can Create Compute Resources

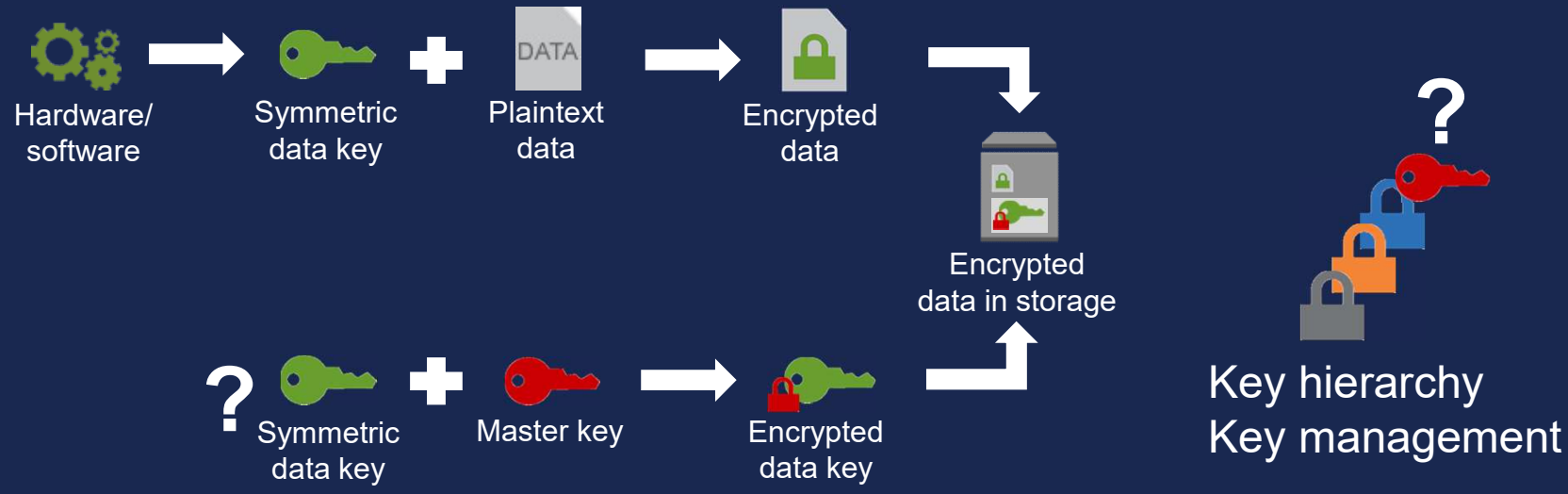
```
{"Statement":[{
        "Identity": “ComputeAdministrator”,
        "Effect": "Allow",
        "Action": [
                "CreateImage",
                "RunInstance",
                "CreateSnapshot"],
        "Resource":
                "instance/i-1234567890abcdef0",
        "Condition": {
                "StringEquals": {"ResourceTag": "ProjectName"}}}]}
```


aws

When One Set of Access Controls Isn’t Enough

Encryption

Data at Rest Encryption Primer



Key hierarchy
Key management

Your cloud provider ensures the plaintext key(s) can only be used by identities you define

aws

Encryption/Decryption as Access Policy

```
{"Statement":[{
      "Identity": “ComputeAdministrator”,
      "Effect“ : "Allow",
      "Action“ : "RunInstance“]}
```

+

```
{"Statement":[{
      "Identity": “ComputeAdministrator”,
      "Effect“ : "Allow",
      "Action“ : “Decrypt“
      "Resource" : "TheKey "]}
```

Both policies must be true to grant access to run your workload

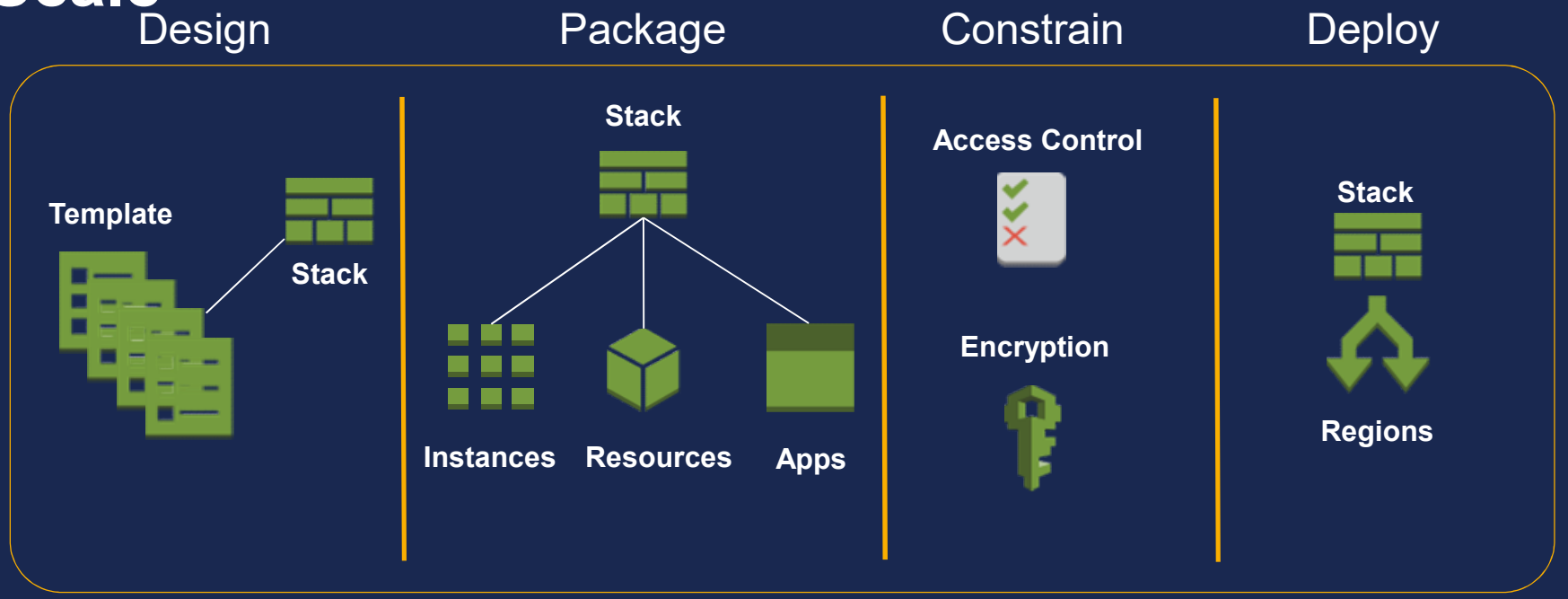
aws

Quis custodiet ipsos custodes?

1. You define the resource configuration and access policies.
2. Your cloud provider faithfully executes your configuration and access policies.
3. The cloud provider's auditors ensure the cloud providers are faithfully executing your configuration and access policies and not looking at your data.
4. Your auditors ensure you define your resource configuration and access policies correctly.
5. GOTO 1

aws

Automating Security Is The Only Way To Safely Scale



Security by Design allows you to automate deployment, configuration, audit, and remediation of your workloads

Thank you!