AWS Cloud Practitioner - 3

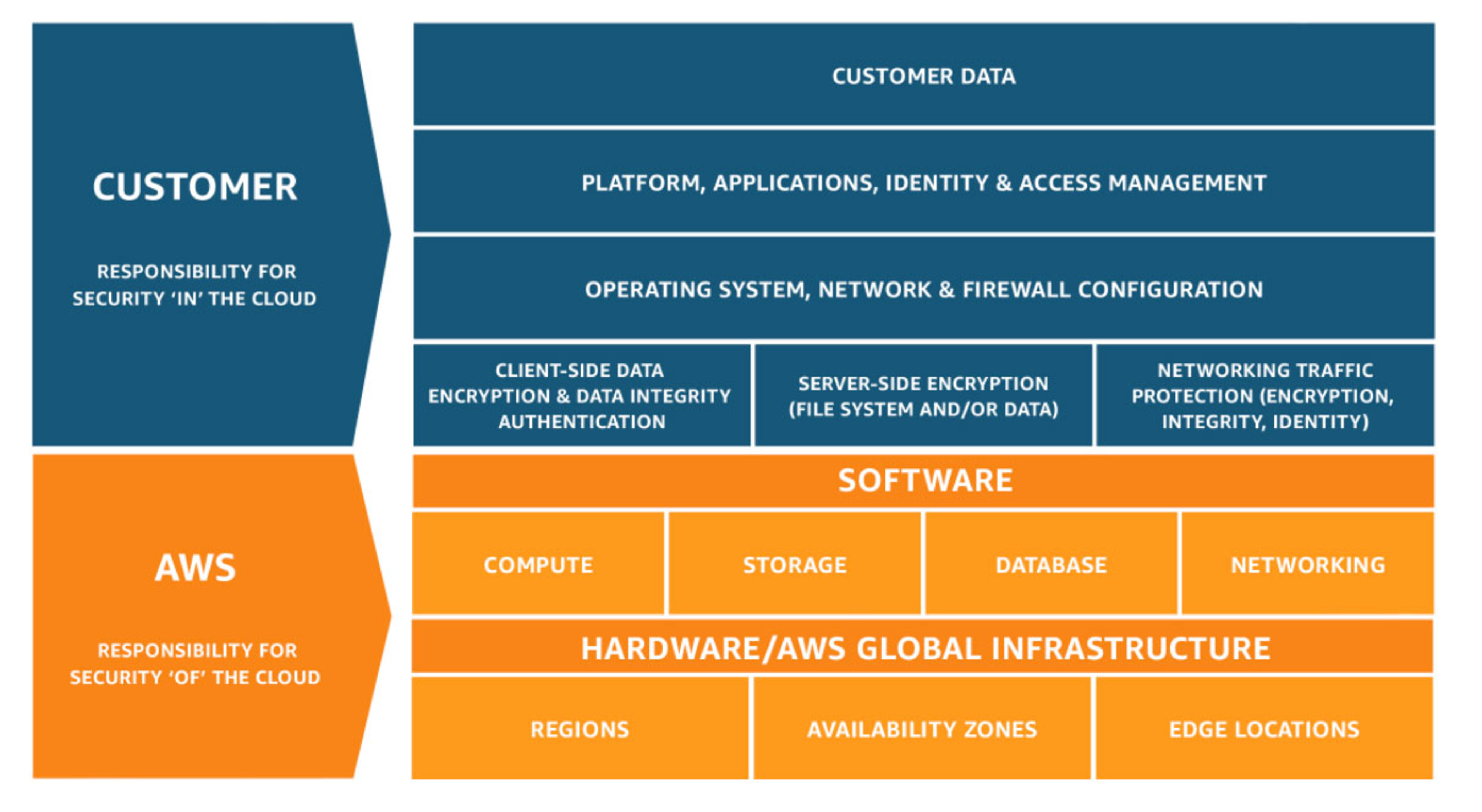
Security

Subhi Dweik. Chapter Eindhoven '22 X Rockademy

Index

- Shared responsibility model
- Well Architected
- IAM
- Organizations
- Encryption
- Sec services
- Questions

Shared responsibility model



CUSTOMER

RESPONSIBILITY FOR
SECURITY 'IN' THE CLOUD

CUSTOMER DATA

PLATFORM, APPLICATIONS, IDENTITY & ACCESS MANAGEMENT

OPERATING SYSTEM, NETWORK & FIREWALL CONFIGURATION

CLIENT-SIDE DATA ENCRYPTION & DATA INTEGRITY AUTHENTICATION	SERVER-SIDE ENCRYPTION (FILE SYSTEM AND/OR DATA)	NETWORKING TRAFFIC PROTECTION (ENCRYPTION, INTEGRITY, IDENTITY)

AWS

RESPONSIBILITY FOR
SECURITY 'OF' THE CLOUD

SOFTWARE

COMPUTE	STORAGE	DATABASE	NETWORKING

HARDWARE/AWS GLOBAL INFRASTRUCTURE

REGIONS	AVAILABILITY ZONES	EDGE LOCATIONS

Well Architected

- 6 Pillars leading to proper software and competent organizations
- A series of best practices AWS has learned over time operating the platform
- Has a tool which helps your org validate all parts of your organization/workload
- Meant as something you keep in mind when building software, and review regularly

Terminology

- Component: Piece of a workload.
- Workload: Generally the smallest unit the business talks about.
- Architecture: How components work together in a workload
- Milestone: Key changes to your achitecture
- Technology portfolio: Collection of workloads the business rquires to operate.

Pillars - 1

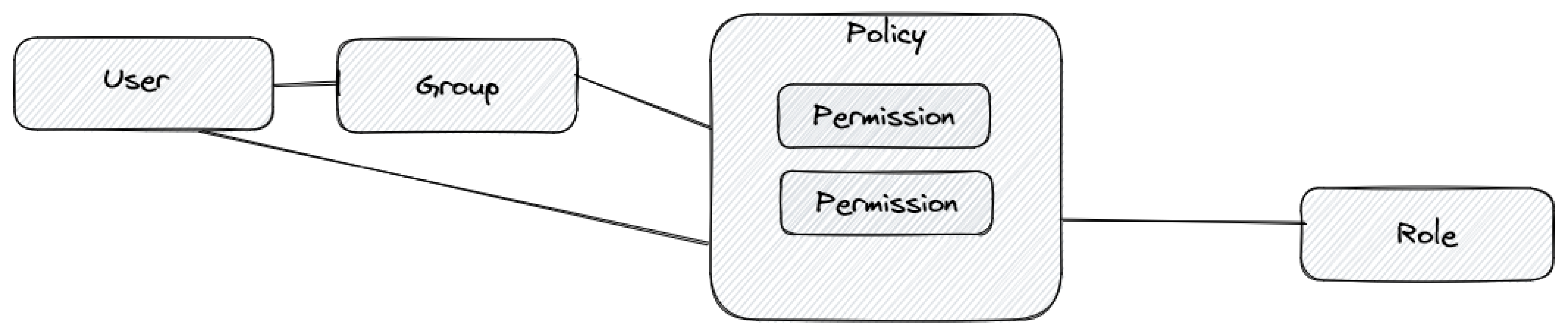
- Operational Excellence
  - Essentially how you deal with software and its deployment
- Security
  - Least privilege, Zero trust, Audit...
- Reliability
  - Automate failure recovery, Stop guessing capacity

Pillars - 2

- Performance Efficiency
    - Serverless first, multi-region, delegate when possible
- Cost optimization
    - Don't overprovision, make sure people have insights in costs
- Sustainability
    - Reduce waste, be aware of whether something is necessary

IAM

- Least privilege by default



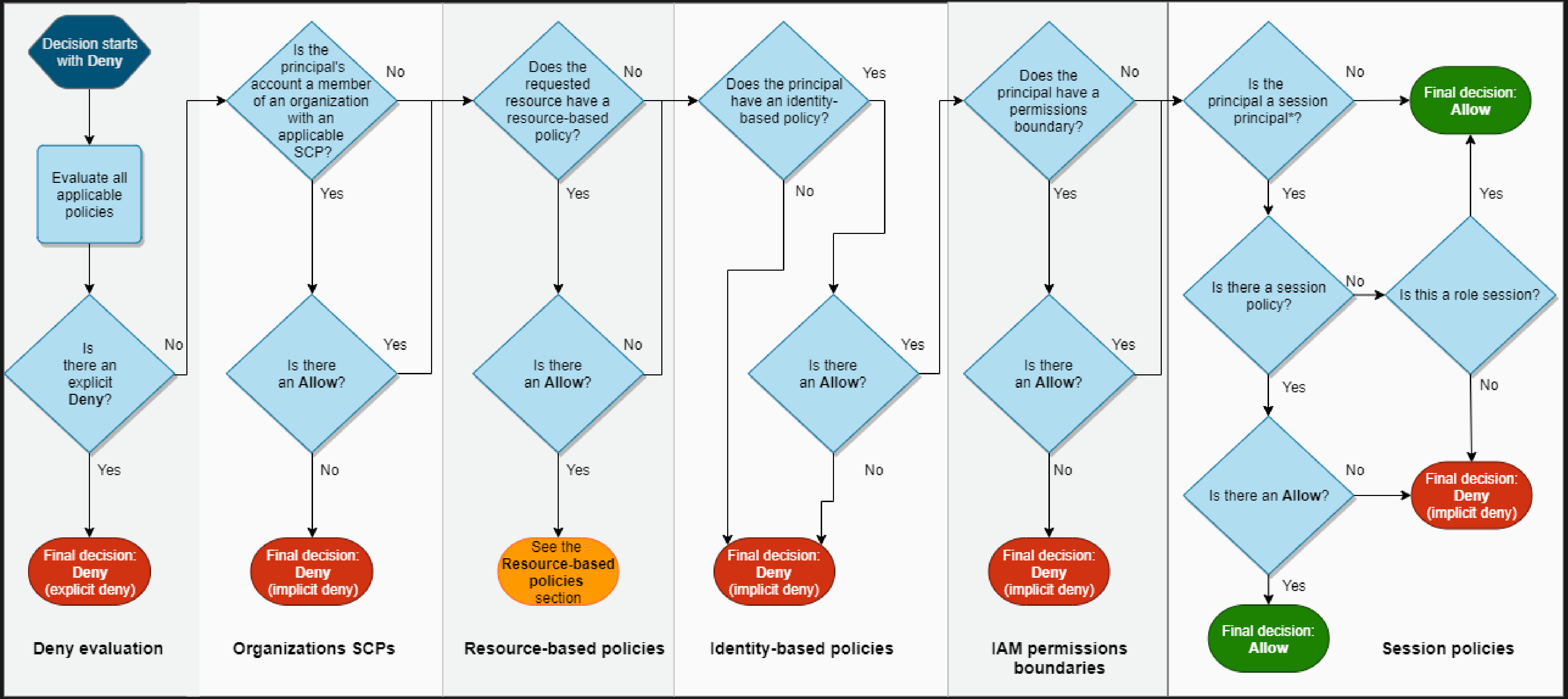
Roles

- A role can be assumed
- Services can assume roles
- Not all services can assume roles
- Role permissions are not in addition to your own roles

Policies

- Declaration of one or more permissions
- Evaluated at time of request
- IAM Policies only control access to AWS services

Policy eval order



Decision starts with Deny
Evaluate all applicable policies
Is there an explicit Deny?
No
Yes
Final decision: Deny (explicit deny)
Deny evaluation
Is the principal's account a member of an organization with an applicable SCP?
No
Yes
Is there an Allow?
Yes
No
Final decision: Deny (implicit deny)
Organizations SCPs
Does the requested resource have a resource-based policy?
No
Yes
Is there an Allow?
No
Yes
See the Resource-based policies section
Resource-based policies
Does the principal have an identity-based policy?
Yes
No
Is there an Allow?
Yes
No
Final decision: Deny (implicit deny)
Identity-based policies
Does the principal have a permissions boundary?
No
Yes
Is there an Allow?
Yes
No
Final decision: Deny (implicit deny)
IAM permissions boundaries
Is the principal a session principal*?
No
Yes
Final decision: Allow
Is there a session policy?
No
Yes
Is this a role session?
Yes
No
Is there an Allow?
No
Yes
Final decision: Deny (implicit deny)
Final decision: Allow
Session policies

*A session principal is either a role session or an IAM federated user session.

Policy example

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "limitedSize",
            "Effect": "Deny",
            "Action": "ec2:RunInstances",
            "Resource": "arn:aws:ec2:*:*:instance/*",
            "Condition": {
                "ForAnyValue:StringNotLike": {
                    "ec2:InstanceType": [
                        "*.nano",
                        "*.medium"
                    ]
                }
            }
        }
    ]
}
```

Policy actions

```
"Action": "ec2:StartInstances"
"Action": "iam:ChangePassword"
"Action":["sqs:SendMessage", "sqs:ReceiveMessage"]
"Action":"s3:List*"
```

Policy Conditions

```
"Condition": {
  "DateGreaterThan": {"aws:CurrentTime":"2016-11-30T11:00:00Z"},
  "DateLessThan": {"aws:CurrentTime":"2016-11-30T15:00:00Z"},
  "IpAddress": {"aws:SourceIp":["192.0.2.0/24", "203.0.113.0/24"]}
}
```

```
"Action":"s3:ListBucket",
"Effect":"Allow",
"Resource":["arn:aws:s3:::mybucket"],
"Condition":{"StringLike":{"s3:prefix":["home/${aws:username}/*"]}}
```

Policy Anatomy

```
{
   "Version":"2012-10-17",
   "Statement":[
      {
         "Effect":"Allow", # Can be explicit allow, or explicit deny
         "Action":[
            "s3:GetObject", # API action(s) to allow, supports wildcards
         ],
         "Resource":"arn:aws:s3:::awsexamplebucket1/*" # What resource(s) to allow this on
         "Principal": {
            "AWS":"arn:aws:iam::257973423188:root" # Allow this for a specific principal. Usually on the "Receiving side"
         }
         "Condition" : {
            "StringEquals" : {
               "aws:username" : "johndoe" # Only apply this policy if the username is "johndoe"
            }
         }
      }
   ]
}
```

AWS Credentials chain

The AWS SDK looks for credentials in a certain order, from top to bottom:

- Overrides
  - For cli: flags (e.g. `--profile`)
  - For SDK: arguments to constructor
- Environment variables (`AWS_ACCESS_KEY_ID` , `AWS_SECRET_ACCESS_KEY` , `AWS_PROFILE` , etc.)
- Java only: `aws.accessKeyId` and `aws.secretKey` properties
- Web identity token (used for EKS for example)
- The default credentials file (`~/.aws/credentials` and `~/.aws/config` IF AWS_SDK_LOAD_CONFIG is set)
- AWS ECS Container credentials
- EC2 Instance profile credentials

AWS Organizations

- Manage multiple accounts from a single root account
- Consolidated billing
- SCPs

Encryption

- KMS
  - AWS owned/AWS managed/C-KMS
- CloudHSM
  - Single tenant hardware encryption keys
  - Turns out, not that expensive anymore
- SSM
  - Secrets management, supports automatic rotation, optionally by a custom lambda

Traffic security

- WAF
  - Inspects traffic, drops malicious traffic
  - Has to be able to look into traffic, so ssl termination required
- Shield
  - Standard: applies to CloudFront, ELB, and Route53, free and automatic
  - Advanced: Supports EC2, GA, etc.. $3000/m, minimum of 1 year.

Sec adjacent

- Artifact
  - Stores compliance reports for aws services
- Cognito
  - User management. Register users, or federate via an IDP with SAML/OIDC

Assorted sec services

- Config: Realtime change monitoring in AWS
- Macie: Scan for PII
- GuardDuty: Anomaly detection
- Inspector: Scan the contents of VMs, lambda, and Containers for vulnerabilities.

Questions?