AWS Certificate Manager (ACM)

- HTTP - simple and insecure
- HTTP**S** - **SSL/TLS** Layer of encryption added to HTTP
- Data is encrypted **in-transit**
- Certificates **prove identity**
- **Chain of trust** - Signed by a **trusted authority**
- ACM lets you run a **public** or **private** Certificate Authority (CA)
- **Private CA** - Applications need to **trust your private CA**
- **Public CA** - Browsers trust a list of provider, which can trust other providers.
- ACM can **generate** or **import** certificates
  - if generated, it can **automatically renew**
  - if imported, *you are responsible for importing it*
- Certificates can be deployed only to **supported services**
  - Cloudfront, ALB ... *NOT EC2*
- ACM is a **regional service**
- Certs cannnot leave the region once they are generated or imported in
- Eg. *To use a cert with **ALB** in **ap-southeast-2** you need a cert in **ACM** in **ap-southeast-2***
- Global Services such as **CloudFront** `operates as though within 'us-east-1'`

CloudHSM

- **HSM** - hardware security module to manage cryprtographic operations
- **AWS** provisioned it but full managed by customer
- Federal Information Processing Standard Publication 140-2, (FIPS PUB 140-2)
- Fully **FIPS 140-2 Level 3** (*KMS is L2 overall, some L3*)
- KMS can use **CloudHSM** as a **custom key store**, CloudHSM integration with KMS (newer feature)

CloudHSM	KMS
FIPS 140-2 level3	FIPS 140-2 level 2 (some L3)

CloudHSM	KMS
Industry Standard API - **PKCS#11**,Java Cryptography Extenstion (**JCE**), Microsoft **CryptoNG** (CNG) libraries	Communicated via AWS API

- Configure HSM in cluster mode to ensure HA mode (multi AZ). By default, it does not run on HA mode.

Use Cases

- No native AWS integration .. e.g. no s3 SSE
  - client side encryption can be used before uploading
- Offload the SSL/TLS Processing of Web Servers.
- Enable Transparent Data Encryption (TDE) for Oracle Databases
- Protect the Private Keys for an Issuing Certificate Authority (CA)

AWS Config

AWS Config is a service which records the configuration of resources over time (configuration items) into configuration histories.

- Record configuration changes over time on resources
- **Auditing** of changes, **compliance** with standards
- Deos not prevent changes from happening.. no protection -**Regional Service** .. supoorts **cross region** and **account** aggregation
- Changes can generate **SNS Notifications** and near-realtime events via **EventBridge** and **Lambda**
- All the config is **stored regionally in s3** bucket which can be interacted with aws config api

Amazon GuardDuty

Guard Duty is an automatic threat detection service which reviews data from supported services and attempts to identify any events outside of the 'norm' for a given AWS account or Accounts.

- **Continous** security monitoring service - once enabled
- Analyses **supported Data Sources**
  - uses **AL/ML** and **threat intelligence feeds**
- Identifies **unexpected** and **unauthorized** activity
  - either **notify** or **event-driven** protection/remediation
- Support multiple accounts (**MASTER** and **MEMBER**)

AL/ML - Artificial Intelligence/ Machine Learning

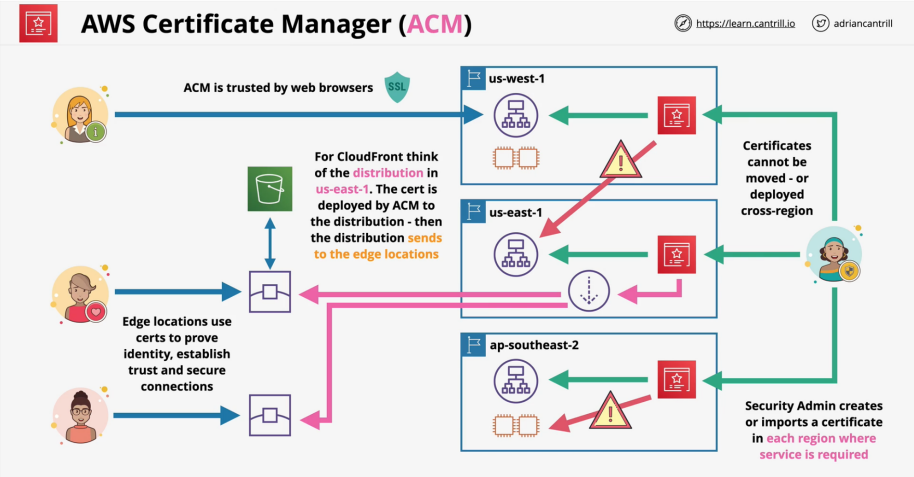


Figure 1: acm

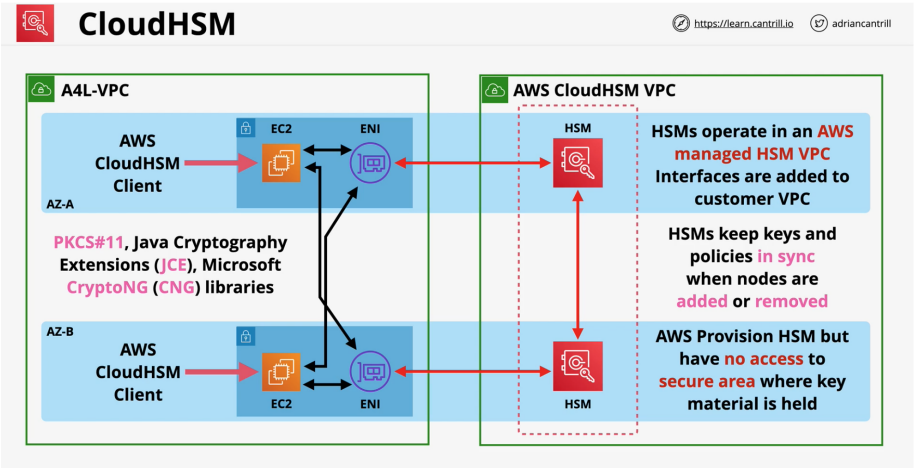


Figure 2: cloudhsm

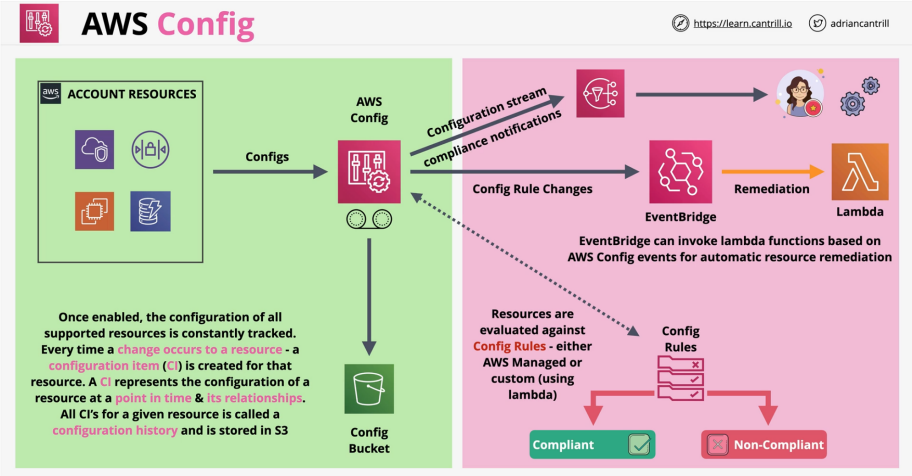


Figure 3: config

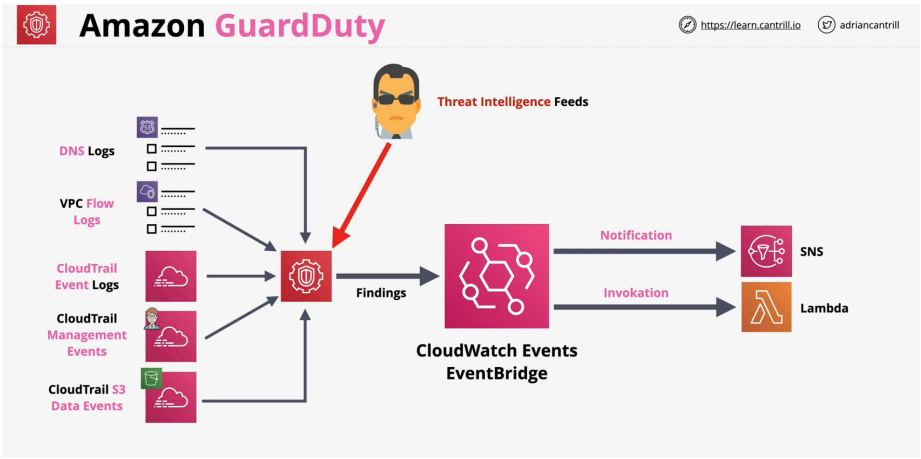


Figure 4: guardduty

AWS Inspector

- Scans **EC2** instances and the **instance** OS
  - scans for **vulnerabilities** and **deviations** against best practice
- Assessment varying period.. 15mins, 1 hours, 8/12 hours or 1 day
- Provides a **report of findings** ordered by priority
- Network Assessment (**Agentless**)
- Network + Host Assessment (**Agent**)
- Rules packages determine what is checked
- Network Reachibility package (**no agent required**)
  - agent can provide **additional OS visibility**
- Check reachibility end to end.
  - EC2, ALB, DX, ELB, ENI
  - IGW, ACLs, RT's, SG's, Subnets, VPCs, VGWs & VPC Peering
- Network Reachability package findings
  - `RecognizedPortWithListener`: exposed to public and OS listening to the port
  - `RecognizedPortNoListener`: exposed to public but OS not listening
  - `RecognizedPortNoAgent`: exposed but no agent to confirm if OS is listening
  - `UnrecognizedPortWithListener`: unrecognized port exposed and OS listening to the port
- **Host Assessment** package (**agent required**)
  - **CVE** - Common Vulnerabilities & Exposures
  - **CIS** benchmarks - Center for Internet Security
  - **Security best practices** for Amazon Inspector

AWS KMS

Key Management Service

- Regional and Public Service
- Create, Store and Manage Keys
- Supports both **symmetric** and **asymmetric** keys
- Also supports cryptographic operations - **encrypt**, **decrypt** & ...
- Keys never leave KMS - Provides FIPS 140-2 (L2)
  - FIPS 140-2 & FIPS 140 -3

Customer Master Keys (CMK)

- **CMK** - **C**ustomer **M**aster **K**ey
- CMK is **logical** - ID, date, policy, desc and date
  - backed by **physical** key material
  - key material can be **generated** or **imported**
- CMK can be used (**encrypt/decrypt**) for up to **4KB of data**
- CMK is encrypted by **AWS** before storing in the disk.

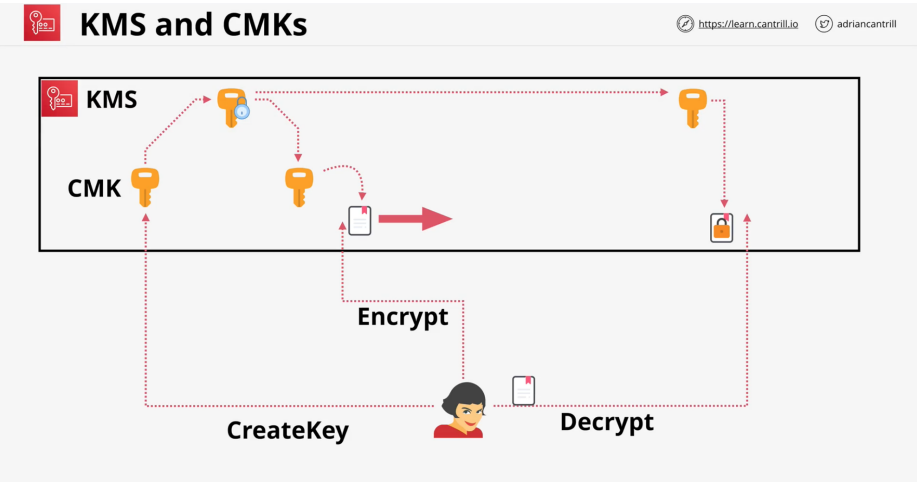


Figure 5: cmk

Data Encryption Keys (DEKs)

- Workaround for `4KB data` limitation
- GenerateDataKey - works on > **4KB**
  - linked to specific CMK
  - KMS does not store DEK in any way
  - generates and discard it once user uses it
- **DEK** is provided in 2 versions
  - **Plaintext** version
  - **Ciphertext** version
- **Encrypt** data using **plaintext** key
  - then **discard** the **plaintext** key
- **Store encrypted key** with encrypted ddata

  S3 create `DEK` for every object

Key Concepts

- CMKs are isolated to a **region** and never leave
- **AWS** Managed or **Customer** Managed CMKs

  - Customer keys and AWS keys
  - Customer Managed Keys are more configurable
- Both types of keys support rotation

Type of KMS key	Can view KMS key metadata	Can manage KMS key	Used only for my AWS account	Automatic rotation
Customer managed key	Yes	Yes	Yes	Optional. Every 365 days (1 year).
AWS managed key	Yes	No	Yes	Required. Every 1095 days (3 years).
AWS owned key	No	No	No	Varies

- CMK itself contains **Backing Key** as well as **previous** backing keys
- Aliases for CMKs - **Per Region**

Key Policies and Securities

- Key Policies (Resource)
- Every CMK has a key policy
  - Customer manage CMK policy can be adjusted
  - CMK key policy explicity told to trust AWS account
  - IAM policies to ensure IAM role/user have access to KMS operation on the key

AWS Systems Manager Parameter Store

- Storage for configuration and stores
- Different types of parameters to be stored
  - String
  - StringList
  - SecureString
- Ex: Licencse codes, Database Strings, Full Configs & Passwords
- **Hierrarchies & Versioning**
  - Allows to store parameters using **Hierrarchial** Order
  - Allows to store different **versions** the parameter
- Can strore both **Plaintext** and **Ciphertext**
  - KMS in turn can use **Ciphertext**
- **Public Parameters** are available. **E.x.** - Latest AMI per region

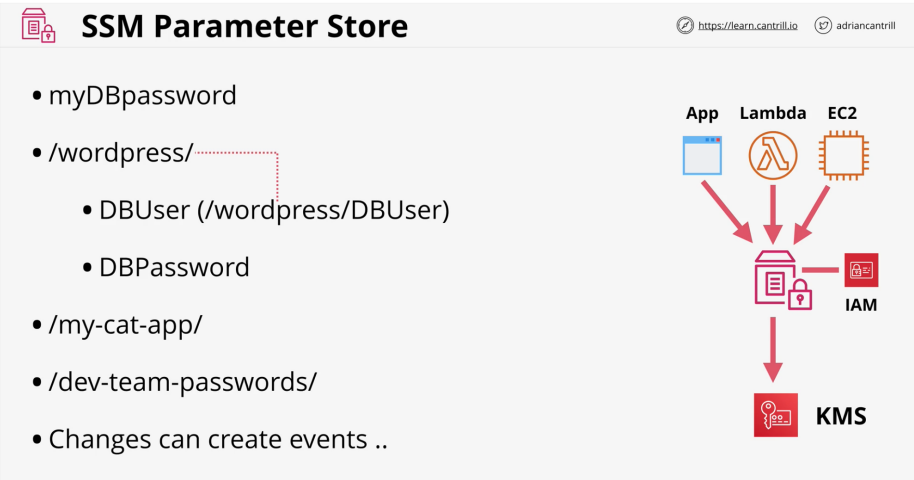


Figure 6: parameter-store

AWS Secrets Manager

- Shares **functionality** with Parameter Store
- Designed for **secrets** .. (passwords, API Keys)
- Usable via **Console, CLI, API or SDK's** (integration)
- Supports **automatic rotation** .. this uses **Lambda**
- Directly **integrates** with some AWS produts like **RDS**
  - sync the authectication creds as well
- Permissions controlled by IAM
- Secrets are encrypted using **KMS**

VPC Flow Logs

- Capture **packet Metadata** .. **NOT packet contents**
- Different monitoring points to apply...
  - **VPC** - all interfaces in the VPC
  - **Subnet** - all interfaces in that subnet
  - **Interface directly**
- VPC Flow Logs are **NOT realtime**
- Destination can be **S3** or **Cloudwatch Logs**

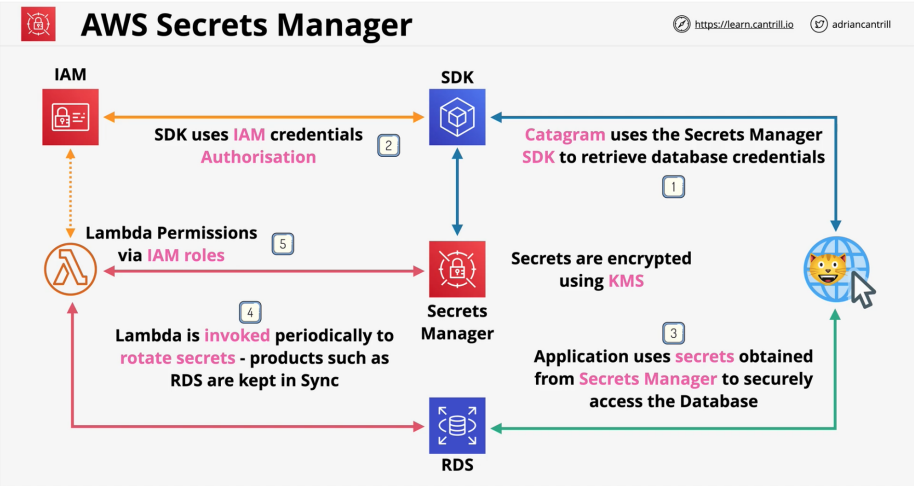
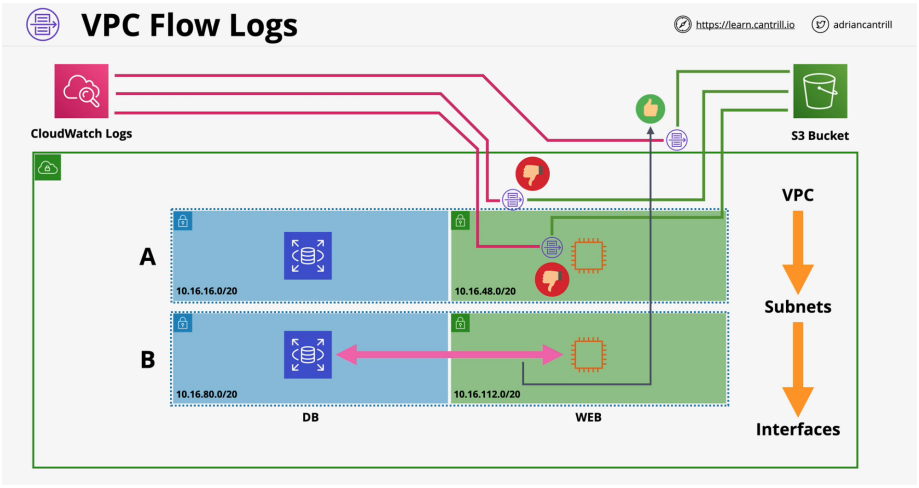
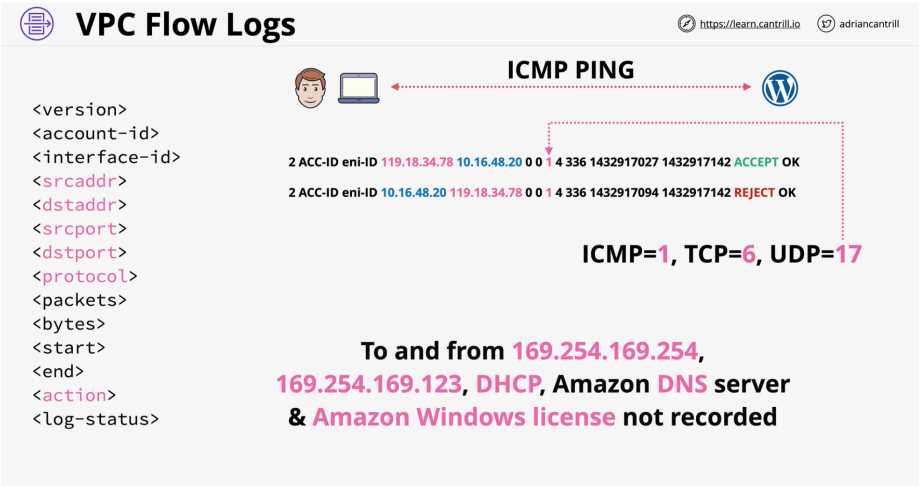


Figure 7: secrets-manager





AWS Shield

- Provides AWS resources with **DDoS protection**

AWS Shield Standard	AWS Shield Advance
Free with **Route53** and **CloudFront**	**$3000/month/organization** which also supports - **EC2, ELB, Global Accelerator** in addition to **Cloudfront & R53**
Protect against **Layer 3** and **Layer 4 DDoS** attacks	**same**
	Provides **24/7 (365 days) advance response team** to deal with DDoS atack
	Financial Insurance for any increased AWS cost incurred by the attack

AWS WAF

Web Application Firewall

- **Layer 7** (HTTP/s) Firewall
- Protect against complex Layer 7 attacks/exploits
- **SQL Injections, Cross-Site Scripting**, Geo Blocks, Rate Awareness
- Web Access Control List (**WEBACL**) integrated with **ALB, API Gateway and CloudFront**
-