STORAGE DEVELOPER CONFERENCE
SDC 21
BY Developers FOR Developers
Virtual Conference
September 28-29, 2021

Transparent Encryption and Dual Endpoint Access Controls to Secure AWS S3 Buckets

Carlos Wong, Sr. Principal Software Engineer

Contributing Team Members: Tirthankar Das, Kyoungbong Koo, Feng Pan, Jose Santos, Mihai Spatar, Sri Sudarsan

THALES

Amazon S3 Data Leaks Continue

US municipalities suffer data breach due to misconfigured Amazon S3 buckets

July 2021

Data of three million elderly citizens exposed in cloud security oversight

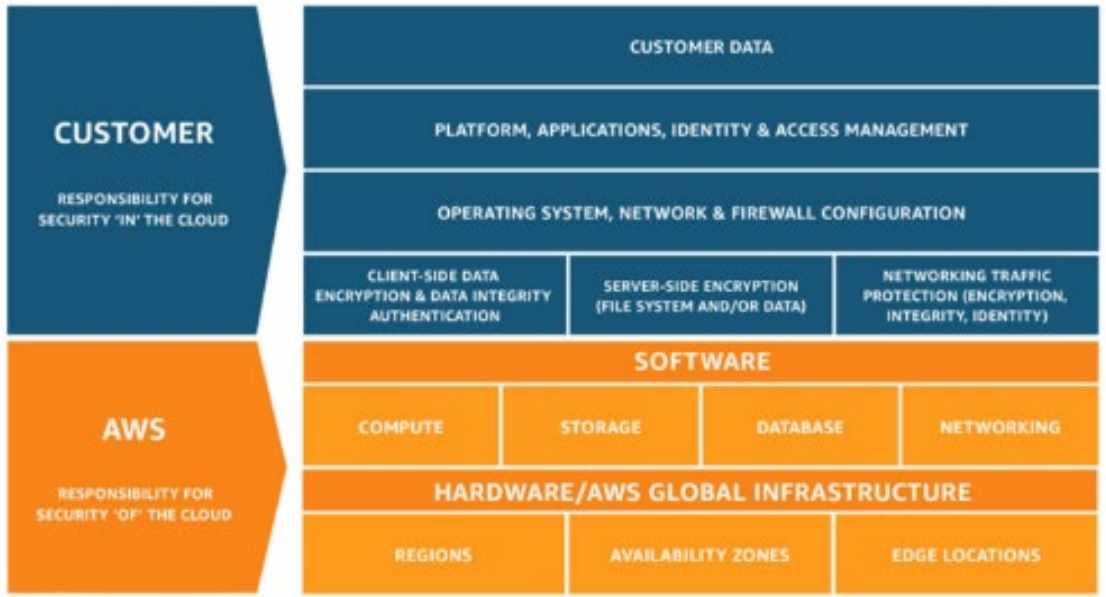
August 2021

Data Breach Affects 300,000 Reindeer Customers Data

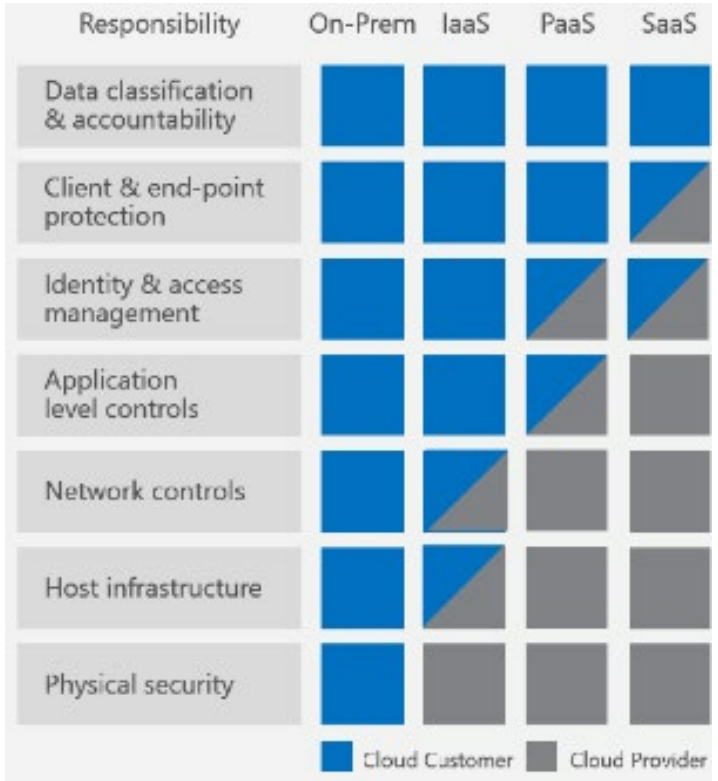
August 2021

Cloud Provider shared responsibilities models are very clear: Customers always own the security of their data

AWS



Azure



THALES

Why do cloud data leaks continue?

Human error caused by complexity or convenience

- Multicloud means learning many policy, cloud security and identity solutions
- Complexity often results in cloud storage left open for easy access

Vulnerabilities happen in software

- Open source AWS WAF *modsecurity* bug reported at Black Hat **2015**
- Imperva Cloud WAF customer database compromise

Insiders include cloud admins and your admins

- Root users have too much visibility
- Former AWS employee leveraged AWS admin knowledge for CapOne breach

So what are some possible next steps?

The plan could be

1. End Human error
2. Vulnerabilities always will be found by the good guys first
3. Insiders always will be trusted and never compromised

Good luck

A better plan to secure cloud data

Protect your data through

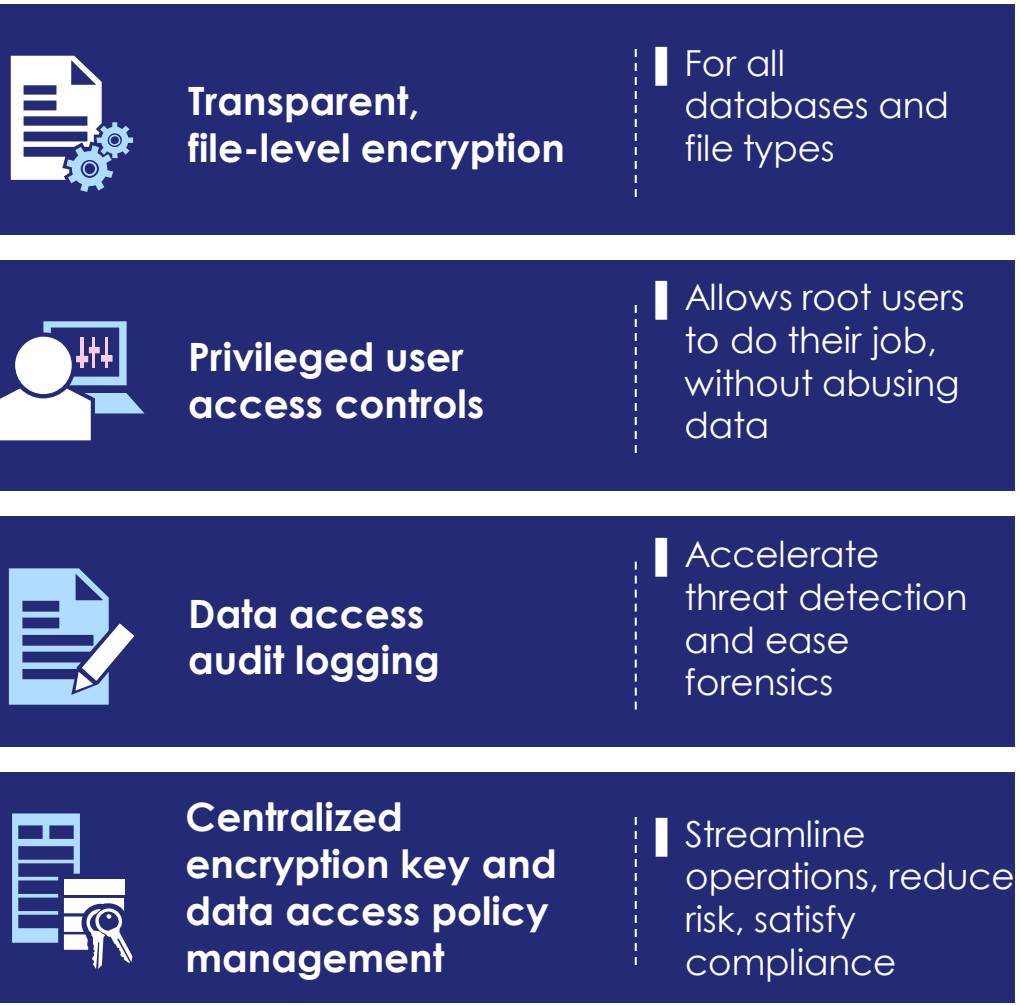
encryption,

tokenization,

and encryption keys you control

Gain safe harbor from breach disclosures

The layers of protection



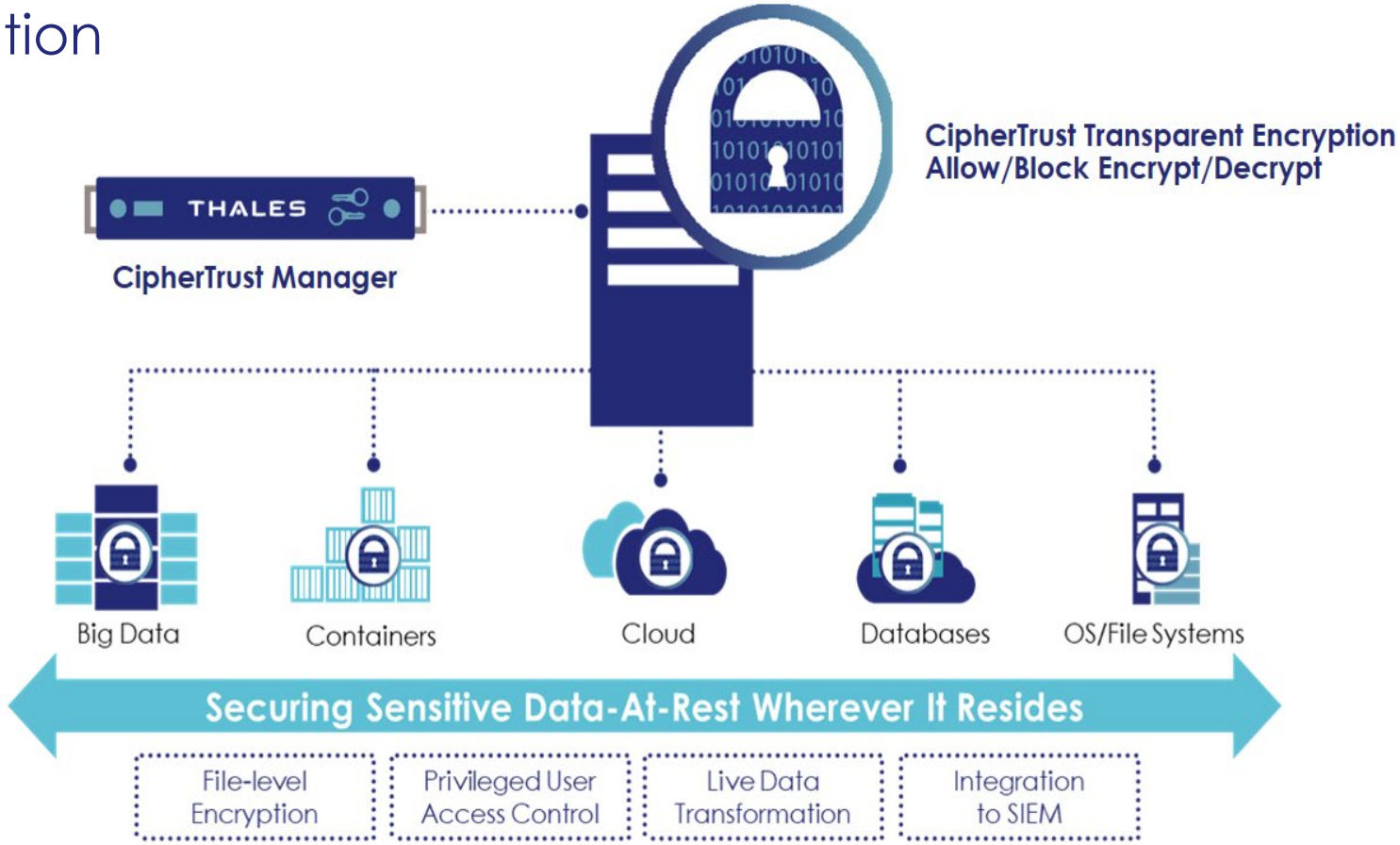
- File level enhanced encryption
- Fine-grain access control
- Device protection from unauthorized access
- Application whitelisting - identify "trusted applications"
- System level "audit logs"

THALES

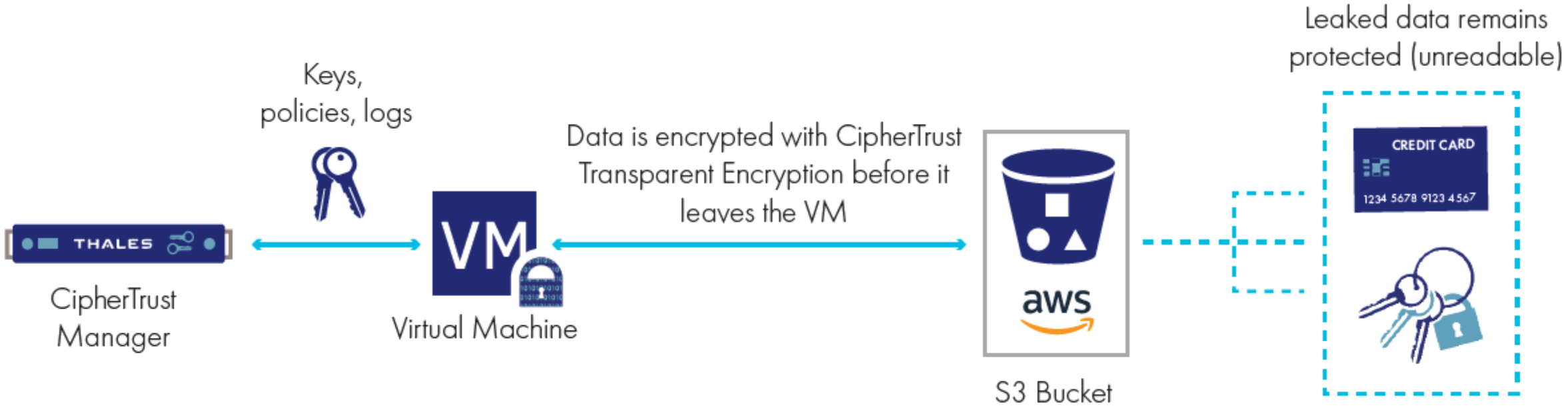
STORAGE DEVELOPER CONFERENCE SDC 21

CipherTrust Transparent Encryption

- Transparent data-at-rest encryption
  - AES-128 or AES-256
  - CBC-CS1 mode
  - XTS mode
- Fine grain access controls
  - Which users
  - What applications
  - What operations
  - Time based
- Audit logging
- Centralized Key Manager
  - CipherTrust Manager
  - Manages encryption keys
  - Manages the access policies



Amazon S3 with CipherTrust Transparent Encryption



- **Transparent to applications and Amazon S3 administrators**

Encryption and access controls are completely transparent to applications while Amazon S3 administrative procedures remain unchanged after software agent deployment. The encryption offered by this solution is independent of Amazon S3 server-side encryption.

- **Continuous protection even with misconfigured S3 buckets**

Continuously enforces policies that protect against unauthorized access by users and processes even in the case of Amazon misconfigurations. Data access to protected S3 buckets is tracked through detailed audit logs.

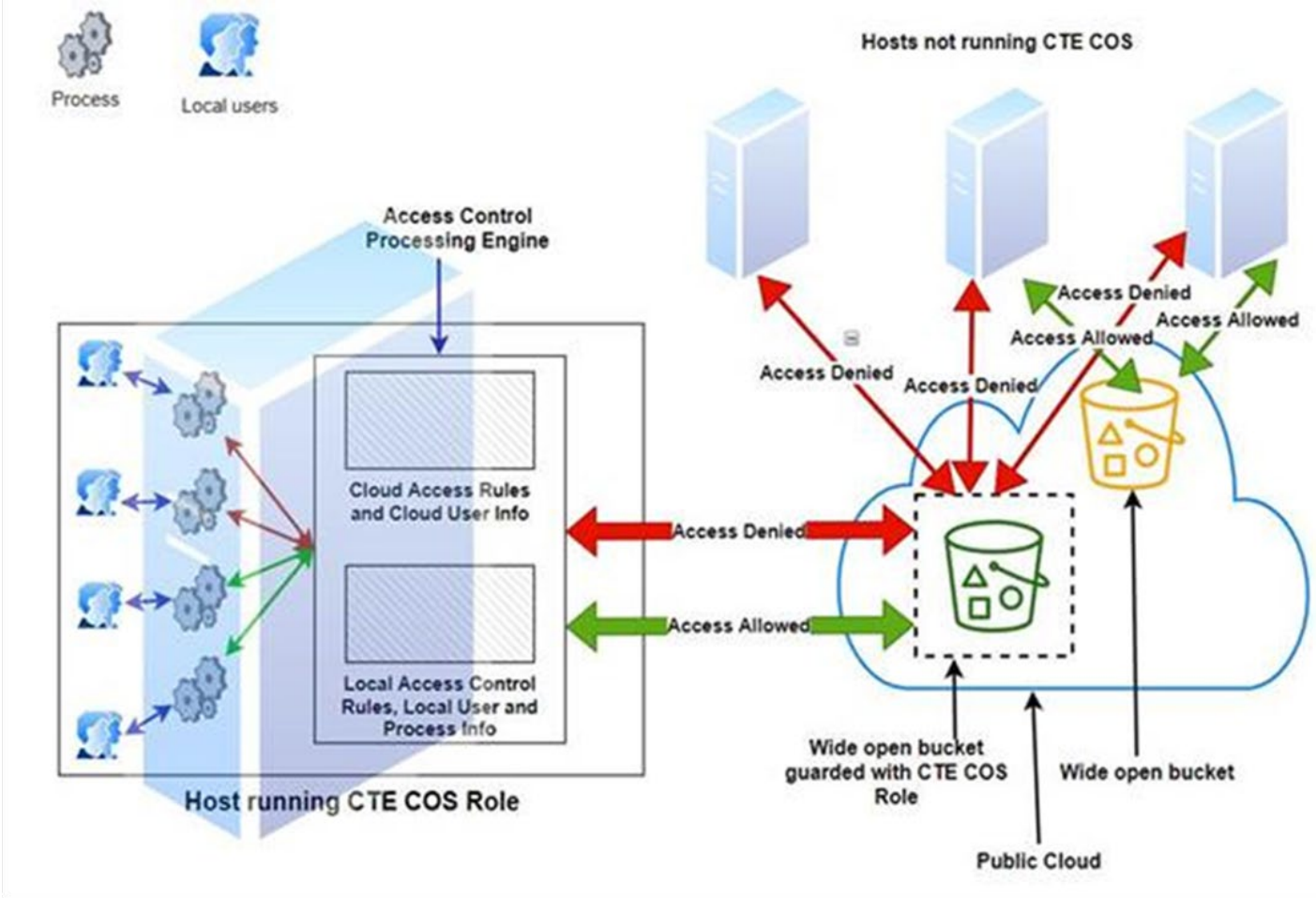
CTE for Amazon S3

- Cloud Object Storage (COS)
- Proxy TLS interception
- Leveraging the Squid Proxy but customized and hardened, e.g.
  - Rejects all clients that are not localhost in origin
  - Only TLS 1.2, TLS 1.3 allowed
  - OpenSSL version that is more securely hardened via compile time options
  - Private CTE Guardpoint to protect key files and other sensitive data
    - Content encrypted and access restricted to root, the proxy, CTE administrative applications
- Proxy uses eCap to interface with the COS S3 adaptation module
- Buckets are secured by applying a "GuardPoint" via the centralized key manager
  - Addressed using same S3 formats, e.g. https://my-s3-bucket.s3.amazonaws.com
- Dual endpoint feature, named End-to-End Access Controls, available as an optional setup

End-to-End Access Controls

Effects of End-to-End Access Controls

- End-to-End access controls in action
  - CTE access controls in effect on the local host client end
  - Special bucket Role policy applied on the AWS S3 server end
  - Bucket policy remains in effect where there is no CTE running
  - Buckets (misconfigured) as public, essentially become private



Setup End-to-End Access Controls on AWS

- On the AWS end perform an IAM Role related setup
- Create an COS Role Policy defining the access to a bucket(s)
- Create the COS IAM Role and assign it the COS Role Policy
- Create a delegated COS IAM User that will assume that COS IAM Role

Sample role policy

- Here is an example of the role policy that can be used. Essentially a simple, wide open policy that permits the IAM Role all actions to the bucket. The idea is that the access controls are offloaded and controlled by CTE on the local client end.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "VisualEditor1",
                "Effect": "Allow",
                "Action": "s3:*",
                "Resource": [
                    "arn:aws:s3:::my-s3-bucket",
                    "arn:aws:s3:::my-s3-bucket/*"
                ]
            }
        ]
}
```


Setting up End-to-End Access Controls on local host

- The COS IAM Role must be registered into COS environment
  - The COS IAM User credentials (key id and secret key)
    - Authenticates the COS IAM User that is permitted to assume the COS IAM Role
  - ARN of the COS IAM Role
  - ARN of the COS IAM User
- The bucket can now be secured
  - Requires the COS IAM User's credentials to enable or disable
  - Triggers an AWS policy on bucket
  - That allows access only to the IAM Role
- CTE administrative tool available for these tasks
- S3 request to bucket will be authenticated to assume the COS IAM Role
  - S3 requests using the IAM User credentials

S3 bucket security offloaded locally

- With bucket locked down on server, enforcement of access policies controlled on client end by CTE
- Local access controls take effect when GuardPoint for bucket is activated
- A combination of policy control points using local identities and local application context
  - Local user identity enforcement
  - Privilege user escalation tracking and prevention
  - Application binary path and signature for integrity
- Added benefit by deploying controls locally along with IAM based protection
  - E.g. the case of stolen AWS credentials

More details on CTE COS access controls

- Proxy is pre-configured to pass into COS adaptation module the unique connection port used by the requestor
- Port helps identify who is making the S3 request
- Access automatically denied if port number not passed in
  - E.g. proxy misconfigured – intentional or not
- CTE COS access control validations are done securely in kernel space
- COS S3 adaptation module interfaces with CTE SecFS kernel module's policy engine
- If access is not denied, request continues towards S3
  - Where AWS performs the other end of the End-to-End access controls
- Following S3 operations are access checked
  - Read object; Write object; Delete object; List bucket

Enhanced Threat Protection with End-to-End Access Controls

S3 and ransomware

- Securing S3 bucket on AWS
  - Configuring AWS S3 buckets can be complicated process
  - Misconfiguration open up buckets to potentially ransomware
    - Gains ability to copy, delete and other permissions
    - Attackers able to encrypt with KMS and replace original object
- S3 buckets are global resources
  - Potentially accessible from anywhere, increasing attack path surface
  - Attackers can probe for buckets that are accessible
- And there are always the insider threats
  - The privilege user accounts are prime targets
- AWS does provide security features for S3
  - Examples: object locking, MFA delete, versioning, IAM user policies, bucket policies

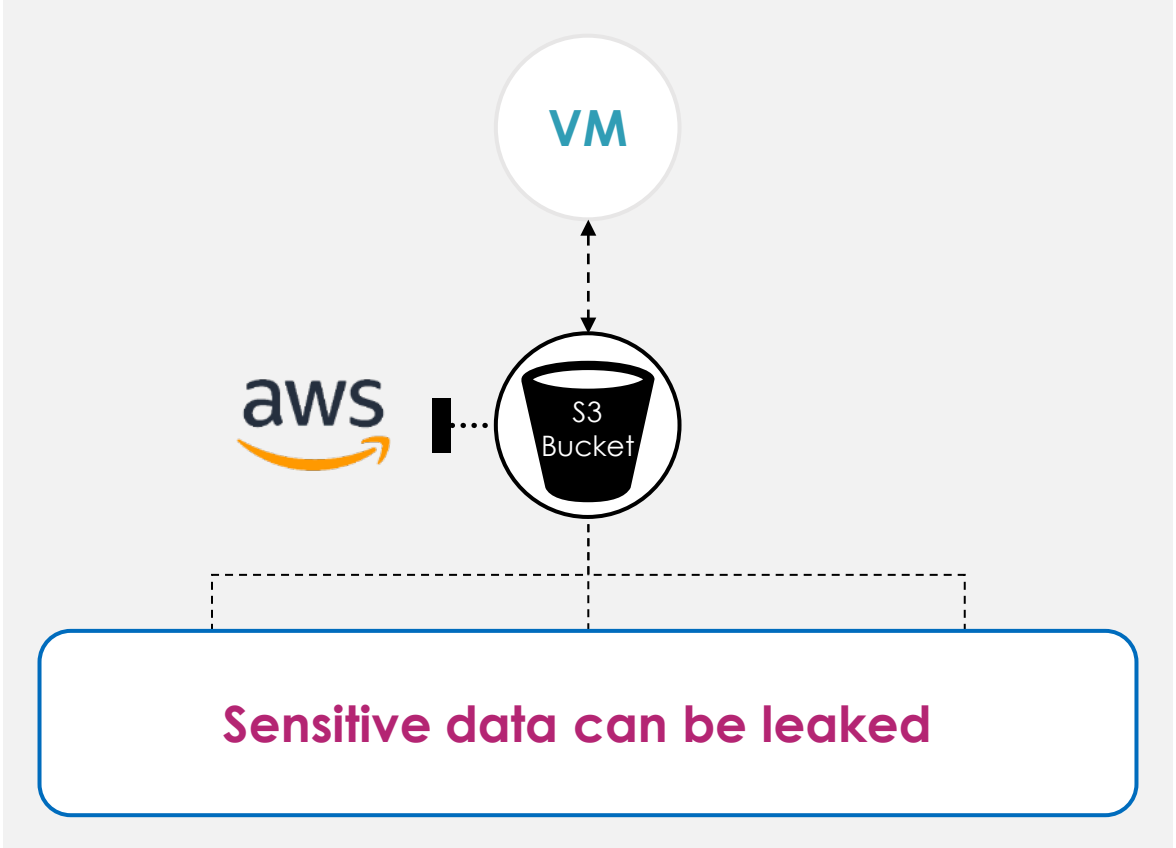
How can End-to-End Access Controls mitigate ransomware?

- End-to-End Access Controls effectively locks down the S3 bucket
  - Avoid complex bucket policies and other AWS measures
  - Access available only thru hosts with CTE COS IAM Role in effect
  - Limits potential external access paths
  - Bucket remains locked down and inaccessible if CTE is shutdown or bucket GuardPoint disabled
- Fine grain access control policies
  - Permit only authorized users defined in the policy
  - Use a whitelist of trusted signed, applications that are only allowed access
  - Blocks untrusted ransomware binaries from successfully executing
  - Prevent escalation of privileges needed to gain system access
  - Prevent ransomware from encrypting the objects
- But in the end, the data is still encrypted
  - Worthless to attacker who plans to publish the sensitive data if not paid

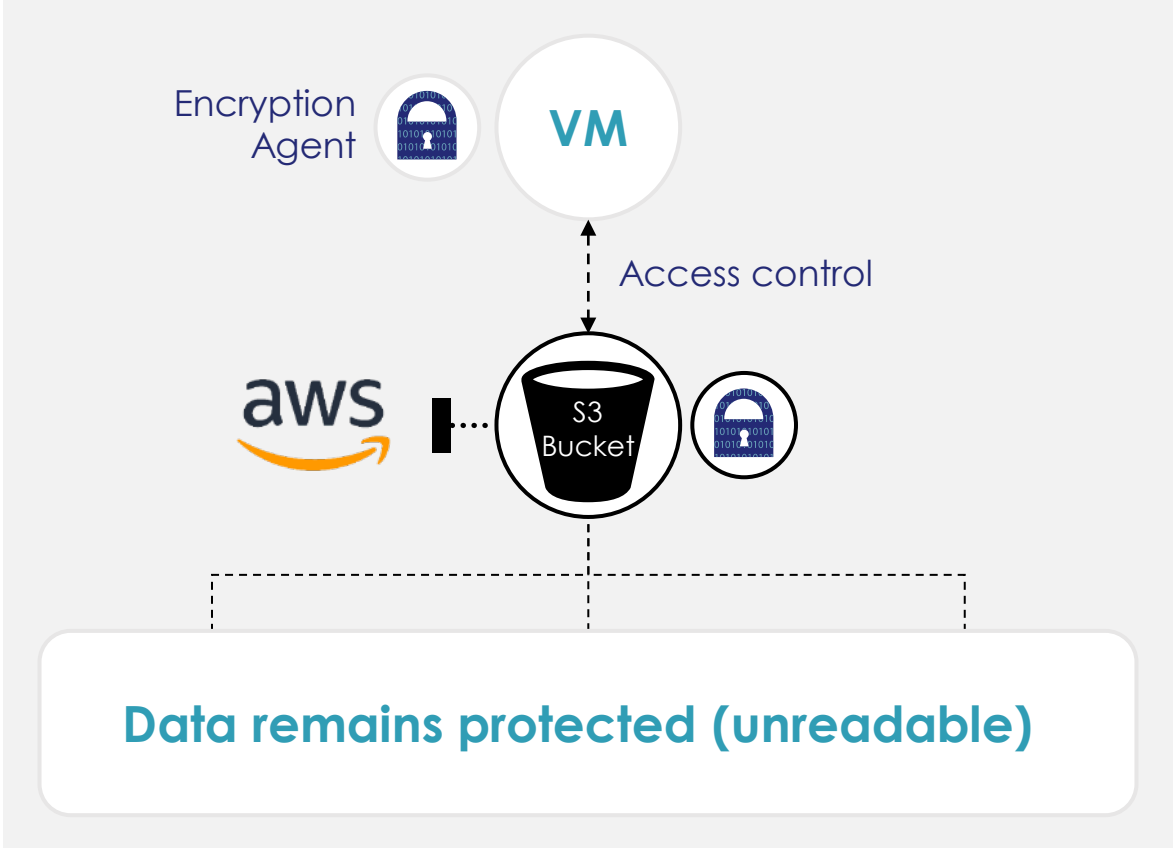
Final Thoughts

Avoid Data Leaks Caused by Misconfigured Amazon S3

Amazon S3 with Amazon Encryption



Amazon S3 with CipherTrust Transparent Encryption



THALES

STORAGE DEVELOPER CONFERENCE SDC 21

CipherTrust Data Security Platform from Thales

- CipherTrust Transparent Encryption for Amazon S3 is part of the CipherTrust Data Security Platform
- The platform's scalability, flexibility and efficiency drives down data security TCO and prepares organizations to rapidly meet the next security challenge and new compliance requirement

THALES

STORAGE DEVELOPER CONFERENCE SDC 21

Stop data leaks and data breaches in the cloud

- The last line of defense is your encryption
  - Bring your own Encryption to control data access and keys
  - Avoid breach notifications by encrypting your data
- Even if your data does get into the wrong hands, because it's encrypted
  - It's worthless!
  - No risk of newsworthy headlines
  - No fines
- Prevent becoming the next Amazon S3 data leak headline
  - With CipherTrust Transparent Encryption support for Amazon S3, organizations can ensure that volumes of data stored in the S3 buckets are safe and comply with the strictest security regulation

Please take a moment to rate this session.

Your feedback is important to us.

THALES

STORAGE DEVELOPER CONFERENCE SDC 21