Encryption Policy

Purpose

The purpose of this policy is to establish practices for protecting <Company Name> data in the event of unauthorized access through the use of encryption. The policy describes the different components that can be configured to utilize encryption, the algorithm that must be used for each, and how encryption keys should be managed.

Scope

The policy applies to all systems that store or process <Company Name> data classified as Customer Confidential as per the Data Classification Policy.

Ownership

[POLICY OWNER]

is responsible for implementing and maintaining this policy.

Policy Statement

All sensitive data classified according to <Company Name>s Data Classification policy is encrypted at rest and in transit using strong, industry-recommended algorithms. Encryption at rest is used across multiple systems and layers of the stack including file systems, file object stores, databases, third-party SaaS services, and directly in <Company Name>'s own developed components. Encryption in motion is primarily achieved through the use of Transport Layer Security (TLS), but may include other secure protocols.

File store encryption

Third-party cloud filestores such as S3 and GCS are configured with a minimum server-side encryption using the vendor's key.

All <Company Name> files stored in S3 are encrypted using industry-standard AES-256 encryption with AWS-managed keys. S3 encrypts each object on the server, using a unique key, and then further encrypts the keys themselves with a master key stored in AWS KMS.

File systems encryption

File systems for databases and other sensitive data storage require at least block level encryption.

<Company Name> encrypts all data volumes on AWS using AWS EBS encryption. This solution applies encryption at rest on a block-level, and encryption in transit between the data volume and the compute instance. The algorithm used is industry-standard AES-256. The key used is stored alongside the data, but encrypted on-disk using a master key stored in AWS KMS.

Data store encryption

Data stores are configured to enable encryption at rest.

Data stored in <Company Name>'s AWS RDS instances, as well as its automated backups and snapshots, is encrypted using industry-standard AES-256. Encryption keys are AWS-managed, stored in AWS KMS, and key usage is audited.

TLS certificates and endpoints

TLS usage is evaluated on a quarterly basis using tools such as ssllabs and any grades lower than A are promptly corrected.

Strong encryption of data in transit based on TLS requires up-to-date cipher suites on any TLS-enabled endpoints. The list of suite components that must be kept updated includes the TLS version, configuration options, as well as available algorithms and key lengths. Critical vulnerabilities in older SSL and TLS versions, such as the Beast and Poodle attacks, as well as subsequent deprecations of TLS v1.0 and v1.1 over the past several years, have made securing TLS termination endpoints a necessary major focus of any strong security program.

<Company Name> maintains a secure and updated configuration of its TLS endpoint, and performs continuous external tests. A passing test requires TLS v1.2 or higher, and AES-128 or higher.

Data in transit encryption

Data in transit over the public Internet is encrypted with industry-standard algorithms.

All public interfaces must only be accessible over secure ports and protocols, such as TLS and ssh. <Company Name> performs continuous testing to ensure that ports associated with plain-text protocols, such as port 80 for http, are closed. All communication between clients and <Company Name>'s network components and applications is encrypted with AES-256 to ensure any accidental or malicious exposure of a communication channel is unreadable to unauthorized parties.

Data in transit encryption at <Company Name> is applied universally to all data, without consideration of that data's classification or its status as Sensitive or Regulated.