## Contents

Foreword ....................................................................................................................................................................5
Letter from the Co-Chairs...........................................................................................................................................6
Acknowledgments ......................................................................................................................................................7
1.0 Introduction ..........................................................................................................................................................8
   1.1 Intended Audience ...........................................................................................................................................8
   1.2 Scope ................................................................................................................................................................9
2.0 Requirements Addressed .................................................................................................................................. 10
   2.1.1 Data Availability ...................................................................................................................................... 10
   2.1.2 Key Management ................................................................................................................................... 10
   2.1.3 Data in the Cloud .................................................................................................................................... 11
   2.1.4 Securing the Client .................................................................................................................................. 12
   2.1.5 Policy and Enforcement .......................................................................................................................... 12
   2.1.6 Data Integrity .......................................................................................................................................... 13
3.0 Implementation Considerations and Concerns................................................................................................. 14
   3.1 Considerations ............................................................................................................................................... 14
      3.1.1 Key Management ................................................................................................................................... 14
      3.1.2 Availability ........................................................................................................................................... 14
   3.2 Concerns ........................................................................................................................................................ 14
4.0 Implementation ................................................................................................................................................. 16
   4.1 Architectural Overview .................................................................................................................................. 16
      4.1.1 Data Confidentiality ............................................................................................................................... 16
      4.1.2 Key Management ................................................................................................................................... 18
      4.1.3 Interoperability ....................................................................................................................................... 20
      4.1.4 Algorithms .............................................................................................................................................. 21
   4.2 Guidance and Implementation Steps ............................................................................................................ 21
      4.2.1 Data at Rest ............................................................................................................................................ 21
      4.2.2 Data in Transit ........................................................................................................................................ 22
      4.2.3 Data in Use .............................................................................................................................................. 23
      4.2.4 Data Destruction ..................................................................................................................................... 23
4.3 Key Management .......................................................................................................................................... 24
  4.3.1 Securing the Client .............................................................................................................................. 24
  4.3.2 Client-Side Encryption ......................................................................................................................... 25
  4.3.3 Endpoint Devices and Applications ..................................................................................................... 25
  4.3.4 Protection of Keys ............................................................................................................................... 25
  4.3.5 Policy and Enforcement ....................................................................................................................... 26
  4.3.6 Data Integrity .................................................................................................................................... 26
5.0 References and Useful Links .............................................................................................................................. 28
Foreword

Cloud Computing represents one of the most significant shifts in information technology many of us are likely to see in our lifetimes. We are reaching the point where computing functions as a utility, promising innovations yet unimagined. The major roadblock to full adoption of Cloud Computing has been concern regarding the security and privacy of information.

Much work has been done regarding the security of the cloud and data within it, but until now, there have been no best practices to follow when developing or assessing security services in an elastic cloud model—a model that scales as client requirements change.

One mission of the Cloud Security Alliance is to provide education on the uses of Cloud Computing to help secure all other forms of computing. To aid both cloud customers and cloud providers, the CSA SecaaS Working Group is providing Implementation Guidance for each category of Security as a Service, as delineated in the CSA’s SecaaS Defined Categories of Service. Security as a Service was added, as Domain 14, to version 3 of the CSA Guidance.

Cloud Security Alliance SecaaS Implementation Guidance documents are available at https://cloudsecurityalliance.org/research/working-groups/security-as-a-service/.

We encourage you to download and review all of our flagship research at http://www.cloudsecurityalliance.org.

Best regards,

Jerry Archer  Alan Boehme  Dave Cullinane
Nils Puhlmann  Paul Kurtz  Jim Reavis

The Cloud Security Alliance Board of Directors
Letter from the Co-Chairs

Security as a Service is a specialized area categorized two years ago as growing rapidly and in unbound patterns. Vendors were struggling. Consumers were struggling. Each offering had its own path. We felt it was urgent to address the needs and concerns common to the implementation of Security as a Service in its many forms.

The Defined Categories of Service helped clarify the functionalities expected from each Category. In this series, we hope to better define best practices in the design, development, assessment and implementation of today’s offerings.

We want to thank all of the many contributors worldwide who have worked so hard to produce these papers providing guidance for best practices in Cloud Computing Security. Many have been with the Security as a Service Working Group since the beginning; many others joined in this effort. Each has spent countless hours considering, clarifying, writing and/or editing these papers. We hope they help move forward toward those unimagined innovations.

Sincerely,

Kevin Fielder and Cameron Smith
SecaaS Working Group Co-Chairs
Acknowledgments

Co-Chairs

Vadim Saratovtsev, Mycroft
Geoff Webb, NetIQ

Contributors

Karen Lu HongQian, Gemalto
Jim Peterson, PKWare
Asad Ali, Gemalto
Price Oden, Microsoft
Anish Mohammed
Maurizio Divona, Gemalto
Karen Lu HongQian, Gemalto

Peer Reviewers

Mark Penny, NHS Informatics
Peter Warner, Sandia National Laboratories
Mitesh Soni, iGATE
Gert Hoevenaars, Acision b.v.
Yael Nishry, Vaultive

CSA Global Staff

Aaron Alva, Research Intern
Vicki Hahn, Technical Writer/Editor
Luciano JR Santos, Research Director
Kendall Scoboria, Graphic Designer
Evan Scoboria, Webmaster
John Yeoh, Research Analyst
1.0 Introduction

Encryption is one of the most effective data protection controls available today. Encryption integrity is based on the technologies and processes governing the cryptographic security services.

Encryption is a primary data (and application) protection technique. For encryption to be useful, encryption keys must be properly managed and protected. This document covers both the encryption and key management topics.

The emergence of cloud computing where critical customer and enterprise data could be held by third-party cloud providers in multi-tenant shared computing and storage environments highlights the need to call on encryption as a primary security control.

Storage, movement, and processing of digital information are commonly discussed in terms of “Data at Rest,” “Data in Transit,” and “Data in Use.” The application of encryption mechanisms can similarly be considered for each of these states.

When enterprises and individuals move their data and applications to the Cloud, protection of their confidential information (e.g. company secrets, intellectual properties) and sensitive information (e.g. personal identifiable information, sensitive personal information) in transit, at rest, and in use is critical. Inappropriate information disclosure could cost a data owner’s reputation, financial standing, and impact their regulatory and legal compliance requirements.

When cryptography is used to protect valued data, the risk is transferred from the content to the keys. Once encryption has occurred, protection of cryptographic keying material becomes paramount.

Furthermore, the emergence of cloud delivery of security services now means that encryption capabilities can not only be used to secure data in the cloud, but can also be offered through the cloud to enable organizations of all kinds to more easily protect sensitive data.

It is understood, therefore, that encryption is essential to cloud computing. It then becomes a question of who provides these encryption controls. Some may be available as part of widely used protocols such as SSL or IPsec for transiting data over the internet. Ideally, an Enterprise customer encrypts its data before moving it to a public cloud. If the customer cannot do this themselves, then encryption should be outsourced to a security service offered by the cloud provider or vendors who offer specialized encryption for cloud environments.

The purpose of the Encryption Category of the Security as a Service Initiative is to identify what Encryption Security as a Service means and to provide guidance to organizations on reasonable implementation practices.

1.1 Intended Audience

This document can be used by anyone working in the area of digital security with a focus on data encryption. This includes software architects who design such security systems, developers who implement them, and managers who are responsible for the overall project scope.
1.2 Scope

The scope of this reference architecture is the use of encryption as a service delivered through the cloud. This can involve a fairly wide range of applications. More specifically, the following are considered in scope:

- Key Management Systems (KMS);
- Control of Keys (Customer vs. Cloud Provider);
- Key Lifecycle;
- Shared Secret Encryption (Symmetric Ciphers);
- No-Secret or Public Key Encryption (Asymmetric Ciphers);
- Hashing Algorithms;
- Digital Signature Algorithms;
- Key Establishment Schemes,
- Protection of Cryptographic Key Material (FIPS 140-2; 140-3);
- Interoperability of Encryption Systems, Key Conferencing, Key Escrow Systems, etc.;
- Application of Encryption for Data at Rest, data in transit, and data in use;
- PKI (including certificate revocation “CRL”);
- Future application of such technologies as Homomorphic encryption, Quantum Cryptography, Identity-based Encryption, etc.;
- Cryptosystem Integrity (How bad implementations can undermine a cryptosystem); and
- Cryptographic Security Standards and Guidelines.

Out of Scope:

- Key Labels,
- Algorithm Strength Analysis,
- Non-cryptography-based Confidentiality Schemes (e.g., Steganography), and
- The Development of Encryption Algorithms.
2.0 Requirements Addressed

Encryption mechanisms should reflect the business/usage requirements and the sensitivity of the data to be protected. Typically, data must be managed differently depending on its state or location, whether at rest, in transit, or in use while in the Cloud. While the security of data in transit benefits from mature encryption tools such as SSL, protecting data at rest while ensuring its availability presents additional and ongoing challenges. For structured storage, one of the more challenging problems is how to encrypt data while retaining the ability to read indexes and key fields which may need to be in plaintext. For all data, the goal is to enable secure access and collaboration while protecting against unauthorized access.

2.1.1 Data Availability

Use of a cloud-encryption service presents potentially significant challenges for data availability, which should be carefully considered when implementing or selecting a cloud service. Specifically, a cloud encryption service must manage and escrow keys (where necessary), and provide tight controls over access to those keys.

Although all cloud services must consider availability (and performance) as a component of service management and delivery, the problem is especially acute for cloud encryption services, as poor performance or lack of service would potentially impact a large number of other services, both cloud-based and on premises. At the very least, disaster recovery and fail-over options should be a standard part of any encryption service offered through the cloud.

2.1.2 Key Management

Key management is the most complex part of any security system dealing with encryption of data. An organization must define its cryptographic security, and key lifecycle management policy. Access to these cryptographic keys should be granted to authorized users only and revoked when those users no longer need access.

Cryptographic keys for encryption of data should be generated in a secure way. This may include the use of an appropriately strong random number generator (RNG). Cryptographic keys should never be transmitted in the clear and should be stored inside a secure element, such as a smart card, or a service with the hardware security module (HSM).

It should be possible to escrow keys used for data encryption. Several options for the management and storage of encryption keys should be available. For example, encryption keys stored on the same server as data, on a different server than the data, or delegated to the owner of the data or to a third party provider of such services. Key Management Service should be compliant with then-current standards, such as NIST Special Publication 800-57 parts 1, 2, and 3.

However, based on the Segregation of Duties security principle, key management ideally should be separated from the cloud provider hosting the data. This provides the greatest protection against both an external breach of the service provider as well as an attack originating from a privileged user/employee of the provider.
Additionally, this segregation of duties prevents the cloud provider from unauthorized disclosure of customer data, such as compliance with a subpoena, without the customer knowledge or approval. The customers should retain complete control over their data and only they should be able to comply with disclosure requests.

One design to consider is a Remote Key Management Service where the customer maintains the KMS or Enterprise Key Management (EKM) solution on-premise. Many enterprises already own, maintain and support their own KMS. Oftentimes the KMS uses an HSM for key generation and protection. While such solutions can involve the cloud provider for encryption/decryption services, the cloud provider may not be truly providing a full KMS as a service.

Ideally, the model of Client Side Key Management is followed. This model puts the customer in complete control of encryption/decryption keys. The cloud provider does not hold keys, has minimal knowledge of users, cannot decrypt customer data, and facilitates the storage of encrypted data. This type of solution can be used by cloud storage and SaaS providers as well as IaaS providers.

The Organization for the Advancement of Structured Information Standards (OASIS) Key Management Interoperability Protocol (KMIP) specification for interoperable key management solutions has drawn vendor interest and key management products that support KMIP are appearing in the market.

Key revocation policy and associated mechanisms are critical for all key management models. The key management service should enable key revocation. For example, when an employee leaves the organization or changes job function, key management should revoke all key(s) to which s/he no longer has a need to access.

### 2.1.3 Data in the Cloud

Data in the cloud refers to data while it is being transmitted, stored or processed by a Cloud Service Provider (CSP). The organization should apply the same data classification used when the data is resident within the organization and therefore apply necessary cryptographic security requirements to data stored, transmitted or processed by a CSP. Cryptographic security controls cannot be replaced by CSP Service Level Agreements. Once data is safely transmitted to a CSP, it should be stored, transmitted and processed in a secure way.

The CSP should ensure logical (and if necessary, physical) separation of data based on the owner. Data and cryptographic keying material belonging to company A must be kept separate from data belonging to company B, for example. Special care should be taken with temporary storage locations which the CSP may use to hold decrypted data to ensure that these locations are not pooled or shared between different customers. Careful analysis of all cryptographic functions should be undertaken to ensure that customer data is not put at risk in the various cloud service models (SaaS, PaaS and IaaS) being offered by the CSP.

Encryption of a database should not adversely affect the ability of applications to use this data. If so desired by the user, the CSP should allow data to be encrypted in a format preserving way.

Once data is removed from cloud storage, the CSP must ensure that it is properly purged, and in the case that the CSP is also managing cryptographic keys, that all related keys are destroyed in accordance with approved key lifecycle management standards and guidelines.
The destruction of data implies taking all appropriate measure to ensure that data cannot be accessed by anybody in future. All traces of the data, including any in the logs, should be removed from the system. It should not be possible to gain any future access to the deleted data. Cryptographic keys used during encryption of data should also be destroyed.

### 2.1.4 Securing the Client

Cloud computing services that store encrypted data are made possible by a combination of server and client side components. There often is a far greater emphasis on secure server side software and processes while client components are overlooked and potentially may be vulnerable to compromise. This is especially important as more and more users rely on personal client devices such as smart phones for accessing cloud services. These devices run applications that tout ease of use, and may add security as an afterthought. A backdoor Trojan, keystroke logger, or some other type of malicious software running on a client device can readily undermine the overall security of cloud storage services.

Because of their widespread use and universal acceptance as a de-facto client application, web browsers are a key element in providing client-side access to cloud computing services such as data encryption. The open architecture of web browsers can introduce security problems, allowing user sessions to be hijacked and encrypted cloud data can be compromised. Adequate measures must be taken to authenticate users in a secure manner to protect their online session from malicious use.

Additionally, mobile device applications increasingly provide a front-end to cloud-based services and may be the target of attacks designed to gain access to data and services in the cloud. Care should be taken to ensure that all access points are sufficiently secure to not introduce new vulnerabilities or compromise the security of cloud services, including encryption. Key management is especially important in this area, as keys stored on a mobile application could be exposed if the device is lost or stolen.

It is possible to encrypt the data before it is sent to the cloud, significantly limiting the damage if data is later compromised at the CSP. It also allows greater user control over the generation and storage of encryption keys. Client-side encryption may be necessary for regulatory compliance, such as HIPPA, PCI DSS, and other regulations, in addition to the prevention of unauthorized data disclosure and compliance with data residency and privacy mandates. Encryption of the data prior to its transmission to the CSP and Remote Key Management Service where the customer maintains the KMS or Enterprise Key Management (EKM) solution on-premise, ensure that the data and the ownership of it are retained by the client while the CSP is responsible for hosting and processing the data in its encrypted form.

### 2.1.5 Policy and Enforcement

Requirements for encryption to conform to established security policy do not stop at the on-premise perimeter. As it relates to the use of cloud services, the issues of what is encrypted, and when, should be addressed in a similar way to on-premise methods through the establishment of file and data classification strategies. This classification can then be implemented within information workflows using solutions such as DLP that can identify policy violations and then enforce appropriate use of information.
The issue of how data is encrypted in the cloud must also align appropriately with on-premise policy. Policies and the accompanying enforcement measures must extend to cover data as it moves to and from cloud providers to ensure that the use of encryption continues to meet all mandates for data protection.

2.1.6 Data Integrity

If the encryption key is stored with the CSP, the use of data encryption alone may not provide sufficient assurance that encrypted information has not been altered. Files placed into the cloud for storage may be subject to tampering or replacement in this case, and encryption alone cannot detect this. Combining data encryption with integrity protections such as digital signatures can ensure that data in the cloud remains both private and authentic. Where available, use of trusted time should be considered by using time-stamped signatures on data.
3.0 Implementation Considerations and Concerns

3.1 Considerations

There are a number of areas that should be considered when evaluating the applicability or effectiveness of any encryption solution delivered through the cloud. These are similar to considerations that apply to encryption solutions generally, with the added concerns over availability for the service itself.

3.1.1 Key Management

Providing key management as a security service can be a challenging situation for Cloud Service Providers, Cloud Service Vendors/Brokers and their customers, and especially for enterprises that have existing people, process and technology resource investment in on-premise datacenter encryption solutions. Ideally, the customer maintains control of the encryption keys however, customers need to choose the approach that best matches their risk tolerance and the compliance, government, audit, and/or executive mandated requirements they need to follow.

3.1.2 Availability

Use of a cloud-encryption service presents potentially significant challenges for data availability, which should be carefully considered when implementing or selecting a cloud service.

Specifically, a cloud encryption service that manages and escrows keys must provide tight controls over access to those keys. In the event of a cloud encryption service being unavailable, keys stored with the cloud provider will also be unavailable. This would make access to the data impossible without an alternate key store.

Cloud encryption services that enable the customer/user to store the key locally must still carefully address availability issues, since it is assumed that the decryption process would be unavailable even if the keys are locally managed and stored, or a local copy is available.

Although all cloud services must consider availability (and performance) as a component of service management and delivery, the problem is especially acute for cloud encryption services, as poor performance or lack of service would potentially impact a large number of other services, both cloud-based and on premises.

At the very least, business continuity, disaster recovery and fail-over options should be a standard part of any encryption service offered through the cloud.

3.2 Concerns

The perception often held, whether accurate or not, is that the confidentiality of on-premise data processing, data movement and data storage can be viewed as operating in an enclave that can rely on logical and physical segmentation perimeters for protection. As a result, encryption is sometimes not as well-utilized as perhaps it should be. When these organizations choose to adopt the cloud and move assets outside of that seemingly
protected environment, the need and value of encryption is recognized. Encryption as a Service is still evolving. When migrating common applications, such as email, solutions already exist which enable a standard integration of the encryption to existing environments. However, as the applications are more complex, there can be a dependence on the businesses’ application developers to insert calls to crypto APIs and routines into their code to enable encryption. This may not be a significant concern for organizations that have coders and that use Platform as a Service (PaaS) cloud computing where there is some expectation for application development. However, when the intention is to “forklift” existing applications from on-premise to Infrastructure as a Service (IaaS) providers where integrated or easy to use encryption capabilities may not be readily available, the customer can be left with the task of modifying code, seeking a third-party encryption solution, or daring to operate with suboptimal confidentiality protection.

There may also be legal problems where cryptographic key management is concerned and where an organization chooses to store its keys “in the cloud.” This may have ramifications where using cryptographic keys has a legal significance (such as digital signing of documents) and where users are required to keep those keys under their “sole control.” It may be difficult to prove “sole control” in a cloud hosted key management solution (such as might be necessary in the UK for compliance with the UK’s “Electronic Signature Regulation Act (2002)” (see: http://www.legislation.gov.uk/uksi/2002/318/regulation/2/made).
4.0 Implementation

This Guidance discusses best practices with regard to encryption and key management services for data in all its forms. Typically, data must be managed differently depending on its state, whether at rest, in transit, or in use in the Cloud.

This document addresses encryption and key management as services for server-side, client-side, and hybrid cryptosystems, and assumes that an organization has implemented appropriate data classification standards and requirements.

4.1 Architectural Overview

Encryption solutions must be architected to achieve the goals of both data protection (confidentiality and integrity) as well as availability of the data, the service, and the capability to collaborate and share easily.

As is the case with any approach to implementing an encryption solution, considerations around key management will play a central role in the architecture of encryption delivered through the cloud as well as any legal and jurisdictional requirements which must be complied with.

4.1.1 Data Confidentiality

This reference architecture aims at providing the reader with guidance that is meaningful to whatever implementation scenario the reader is tackling. At the highest level, scenarios are broken into three areas relating to data state:

- Data at Rest
- Data in Transit
- Data in Use

An aspirational solution would be to have a sort of “self-protecting data” where data is an encrypted blob with attached metadata. The metadata contains policy such as an access (who or what can access the data and what operations can they execute, e.g. read-only, modify, etc.) and other tags such as data classification (high business impact, top secret, etc.) and type of data (PII, financial, medical, etc.). This sort of solution could address the protection needs of data at rest and data in transit which would not be as dependent on VPN protection, physical protection, or many other confidentiality controls. Rights Management approaches are an embodiment of this idea; however, they do not appear to be comprehensive and pervasive.

4.1.1.1 Data at Rest

The Data at Rest scenario captures the use of encryption to protect the confidentiality of valued data while in storage. Storage takes many forms for cloud providers, including storage devices, databases, and a variety of unstructured objects. One of the challenges of encrypting structured database storage is how to encrypt data while retaining the ability to read indexes and key fields which may need to be in plaintext. There are multiple
schemes, such as column-level encryption for structured data using Transparent Data Encryption (TDE), and dispersal techniques that split data and send pieces to various locations, so that if one repository is compromised the captured data yields no information.

Data can also be at rest in a variety of unstructured forms such as spreadsheets, reports, folders, files, images, documents and emails. Encryption can be applied through Digital Rights Management (DRM) schemes, native file system encryption, and application software development tools.

Hard drives can be encrypted with full disk encryption implementations available as an OS feature as well as self-encrypting solutions following the Trusted Computing Group’s OPAL standard.

4.1.1.2 Data in Transit

The Data in Transit scenario includes the following situations:

- Data transiting from an end user endpoint computer (e.g., laptop) on the Internet to a web-facing service in the cloud
- Data moving between machines within the cloud (including between different cloud services), e.g., between a web Virtual Machine and a Database
- Secure channels across boundaries such as hybrid topologies where there is a connection between a customer on-premise network and the cloud provider

Perhaps the best known use of cryptography for the Data in Transit scenario is Secure Socket Layer (SSLv3) and Transport Layer Security (TLS). These cryptographic protocols have been in use for many years (in the form of https) typically to provide communication security over the Internet and are the de facto encryption approach for browser-to-web host and host-to-host communication. They use asymmetric cryptography for key exchange followed by symmetric encryption for content confidentiality. IPsec is another transit encryption protocol used for VPN tunnels which can utilize cryptography algorithms such as 3DES and AES. In addition, Secure/Multipurpose Internet Mail Extensions (S/MIME) is available for protecting email from sender to recipient.

4.1.1.3 Data in Use

Data in Use refers to the actual dynamic processing of data, in addition to on-screen display and presentation of data. Historically, processing needed to be done on readable, unencrypted data. For example, encrypted data needed to be decrypted prior to the CPU fetching the data from memory or storage. This created a serious vulnerability unless additional steps were taken to address storage of clear-text in system memory as attacks involving memory dumps could be executed and sifted through to locate the desired information. Also, this exposed an opportunity for an insider attack by cloud provider administrators and vendors.

From a high-level functional description, it is desirable to protect data by presenting encrypted data in a way that enables it to be processed. As described in more technical detail, what is needed is the ability to perform operations on cipher-text without needing to decrypt it first. Clearly such a solution is extremely desirable for cloud computing. Addressing the challenge of protecting data, while at the same time presenting it in a way
that enables it to be processed, is still relatively new when compared with encryption-at-rest and in-transit. Such approaches as tokenization go some way to addressing this need.

4.1.2 Key Management

Based on the Segregation of Duties security principle, key management should be separated from the cloud provider hosting the data. Described below are three different scenarios which depict the possible configurations. Organizations should elect the appropriate scenario for them, based on the classification of the data and their risk tolerance level. The scenario illustrated in Figure 2 (in which the customer maintains the KMS or Enterprise Key Management (EKM) solution on-premise and the encryption keys are stored on-premise) is the optimal scenario. It allows the customer to leverage the benefits of migrating sensitive data to a CSP for storage and processing while maintaining control and ownership over the keys and therefore the data.

Hosting a Key Management Service (KMS) in a multi-tenant environment (Figure 1) can be a costly proposition for a cloud provider, especially if Hardware Security Modules (HSM) are used. As a result, cloud providers may attempt to protect keys with software-based solutions that do not meet the physical security requirements specified in the National Institute of Standards and Technology (NIST) Federal Information Processing Standards Publication FIPS 140-2 or 140-3 specifications. For example, the ability for software to provide tamper evidence is unlikely. It is possible therefore that some Enterprise and Government customers will not accept software-based key protection solutions.

4.1.2.1 Remote Key Management Service

One design to consider is a Remote Key Management Service where the customer maintains the KMS or Enterprise Key Management (EKM) solution on-premise, as illustrated in Figure 2. The ideal scenario is enterprises owning, maintaining and supporting their own KMS, leaving the ownership and control to the customer while the hosting and the processing are sourced out to the cloud provider. Oftentimes the KMS uses an HSM for key generation and protection. This arrangement requires hybrid connectivity between the cloud
provider and the On-Premise KMS. An application runs in the cloud provider’s compute virtual machines and makes calls to the customer’s HSM. The cloud provider can then upload encrypted content to cloud storage and later pull that encrypted content for computation which requires access to the on-premise KMS. While such solutions can involve the cloud provider for encryption/decryption services, the cloud provider is not truly providing a full KMS as a service.

![Figure 2: Remote Key Management Service](image)

### 4.1.2.2 Client-Side Key Management

A similar decentralized approach puts the customer in complete control of encryption/decryption keys. As shown in Figure 3, almost all processing and control is done on the customer side. The cloud provider does not hold keys, has minimal knowledge of users, cannot decrypt customer data, and facilitates the storage of encrypted data. The KMS is provided and run by the cloud provider, but the KMS resides on customer’s premise and the keys are generated and held by the customer. This type of solution can be used by cloud storage and SaaS providers.

![Figure 3: Cloud Storage Providers](image)
4.1.3 Interoperability

The Organization for the Advancement of Structured Information Standards (OASIS) Key Management Interoperability Protocol (KMIP) specification for interoperable key management solutions has drawn vendor interest and key management products that support KMIP are appearing in the market.

Interoperability of encryption systems is an important consideration when implementing cloud solutions. Cloud encryption services protecting data at rest, in motion, or in use must effectively support the eventual need for decrypting encrypted data. Moving encrypted data into and out of the cloud without interoperable encryption solutions capable of processing the encrypted data wherever it is needed will result in disruption of business operations.

Encryption and decryption cycles may fluctuate within different processing environments, by different users, and over varying periods of time. Scenarios where data is encrypted for long-term cloud archiving or storage typically have only an infrequent need to decrypt protected information and only within a narrow range of environmental variation.

Alternately, information having a higher frequency of access (e.g., databases, reports and other business documents) might need to be opened much more frequently, and across a potentially wider range of operating environments and users. For example, a local end-user may be sending a confidential file to multiple business partners. Partners may be geographically dispersed and each may work for separate companies having different types of computing platforms available for receiving the file.

Similarly, a mainframe application process may create sensitive reports requiring encryption. These reports may be encrypted for delivery and may then be received on other mainframe platforms, on Linux, or even by mobile users. The encrypted form of the report must be easily opened by authorized users or applications that may receive the data on an incompatible operating platform such as a mobile device.

Regardless of where or when encrypted information is accessed, users of that information must be able to successfully decrypt it before they can use it to achieve their business objectives. Elements necessary for successful decryption consist of both the keying mechanisms used to deliver and manage the encryption keys, and the encryption file format applied to the data by the encryption service. Both must be interoperable with the expected range of users and environments to ensure that wherever encrypted data is accessed, it can be opened when presented with a correct key. Interoperability of keying methods can be achieved using standards-based key formats (i.e. X.509, OpenPGP) with interoperable key management systems such as those based on KMIP.

Interoperability of encryption file formats can be achieved through support for any of the widely established encrypted file formats such as OpenPGP or secure ZIP. These formats should be considered for protecting files or unstructured data, to ensure interoperability across environments using readily available and easily integrated encryption tools, to avoid extensive customization or re-engineering of systems to support cloud services use. Encryption service providers should offer support for multiple encrypted file formats.
Protecting information using established encryption file formats provides a number of additional benefits which can facilitate efforts to securely move data to and from the cloud in an interoperable manner. These include:

- Support for symmetric and asymmetric keying needs. Keys can be passphrase, public/private key, or forms of Identity-Based Encryption (IBE) [7] and can be used as a combination of any of these types.
- Support for standard encryption algorithms such as AES, Blowfish and RSA which are readily available across all major operating platforms such as AES and Conformance to regulatory requirements such as FIPS 140-2 and 140-3.
- The ability to apply a master key to ensure data access for recovery or inspection purposes.
- Storage and transfer efficiencies to reduce costs through built-in data compression.
- Trust and integrity of data through digital signature use.
- Encryption of information metadata.
- Potential for reduced re-engineering or replacement costs when adding encryption to business processing.

4.1.4 Algorithms

Encryption solutions implement cryptosystems that utilize one or more cryptography algorithms. Very often, these solutions combine asymmetric and symmetric cryptography where asymmetric keys are used to set up symmetric keys on both ends of a communication path and then the symmetric keys are used for content encryption. Different encryption algorithms have different strengths. Encryption-as-a-service providers should choose encryption algorithms based on standards and the needs of their target customers as well as the regulatory markets that their customers work in and will need compliance with.

4.2 Guidance and Implementation Steps

The following guidelines represent areas to be considered when evaluating the suitability of an encryption service delivered through the cloud, or when planning to implement such a service. Ideally, all data is encrypted automatically without any manual intervention from the user.

The key principle should focus on control and ownership of the data. This should be kept in mind when migrating to the cloud, and controls should be applied throughout the entire lifecycle (in transit, at rest and in use) to allow the customer to maintain control over the data while the CSP hosts and processes it.

4.2.1 Data at Rest

Data at rest refers to data while it is inside persistent storage in structured and unstructured forms such as databases and file systems. Security best practices indicate that:

1. All data classified as needing protection should be encrypted while in storage.
2. The use of encryption should be based on the sensitivity and value of the data (based upon corporate data classification requirements). For example, a possible data handling standard or policy might
specify that the use of encryption is optional for lower value data but mandatory for high value digital assets.

3. Access to stored data should be granted on a need-to-know and least privilege basis.

4. All access to classified/encrypted data must be compliant with established access control policy. In some cases, the policy for who or what is granted access is defined by the owner or a delegated owner of the data; in other cases the policy decision may be set by a custodian of the data, e.g. an IT department.

5. In addition to procedural access control, a cryptosystem must support native integrity checking.

6. It should be possible to compare some encrypted data (one-way hashed passwords) without jeopardizing the overall integrity of the cryptosystem.

7. Once data is safely transmitted to a CSP, it is stored in a secure way.

4.2.2 Data in Transit

Data in transit refers to data while it is being transferred from one data repository to another. Data in transit includes data sent by back-end servers, applications and databases over the network. Two data repositories can be within the same corporate network, in the cloud, or part of completely different networks. Data in Transit should be managed so that:

1. All sensitive data is protected while in transit over the wire, air, or any other transport medium.

2. If data intended to be confidential to any degree is being transmitted unencrypted, secure network communication protocols such as IPSec or SSL/TLS should be used, especially when transmitting data over shared or public networks, including shared networks of the Cloud Service Provider (CSP). This is commonly referred to as “encrypting the pipe” vs. encrypting the data itself.

3. There are several options for controlling the distribution of encryption keys for protecting data in transit. The CSP should offer multiple options. For example, encryption keys could be distributed to the CSP and both data depository owners, to the data repository owners and a third party provider of such services, or to only the data depository owners. Transmit paths include traffic between a customer endpoint (e.g. a browser on a laptop) and a publishing presence in the cloud (e.g. a web site), traffic between VMs within the cloud, and traffic over a hybrid connection between VMs in the cloud and machines on a corporate network. In a typical browser-to-web site connection, the web site holds/presents a server certificate; rarely does the browser end of the connection present a client certificate. It is important therefore that the client understands the certificate presented to it by the cloud web site so that it can make the decision as to whether or not the web site is trusted. Communications between VMs within the cloud may not be protected by CSP-provided encrypted pipes or customer controlled encrypted pipes. Rather the provider offers segmentation between each customer. Ideally, even transmissions within a single customer’s subscription should be encrypted. This might be accomplished using SSL/TLS or end-to-end IPsec solutions. Hybrid VPN connections will likely have one termination point in the cloud and the other at the customer on-premise location. Again, SSL/TLS or IPsec tunnels should be used to protect the data in transit.
4.2.3 Data in Use

Data in use refers to data while it is being processed by a Cloud Service Provider (CSP). Best practices would ensure that:

1. The same data classification and cryptographic security requirements used inside an organization are applied to data processed by a CSP. It’s possible that the data handling requirements need to be made even more stringent when data is moved from a “protected” on-premise environment to a public multi-tenant cloud. It is critical that the customer, and not the CSP, is responsible for the security and encryption protection controls necessary to meet their requirements.

2. Cryptographic security controls may not be replaced by CSP Service Level Agreements.

3. The CSP should ensure logical (and if necessary, physical) separation of data based on the owner. For example, data and cryptographic keying material belonging to company A must be kept separate from data belonging to company B. Ideally, the keys are stored and maintained by the customer.

4. Once data is removed from cloud storage, the CSP should ensure that it is properly purged, and, all related cryptographic keys the CSP holds or manages must be destroyed in accordance with approved key lifecycle management standards and guidelines.

5. If so desired by the user, the CSP should allow data to be encrypted in a format preserving manner. For example, a 16-digit credit card number should appear as a string of 16 digits, with each digit represented by ‘x’.

6. Encryption of a database should not adversely affect the ability of applications to use this data.

4.2.4 Data Destruction

The destruction of data implies taking all measures necessary to ensure that data cannot be accessed by anybody in future. The extent to which data destruction processes remove data and keys may depend on the type of service and the requirements of the data owner. Such requirements would therefore be based on the data owner having both their own data classification method and communicating such classification to the CSP. Alternatively the customer may simply specify that all data and access to it be destroyed as part of a standard data lifecycle plan.

Acceptable data destruction techniques ensure:

1. When so desired by an authorized user, all traces of the data, including in the logs, should be removed from the system.

2. It will not be possible to gain any future access to the deleted data.

3. Memory and storage disks are overwritten (wiped) before being put back in use for the next customer. It is not sufficient to simply removed pointers. If the data was truly encrypted, the key(s) necessary to decrypt the data became the critical asset to protect, and therefore destruction of the key(s) should suffice. Nevertheless, even the ciphertext should be destroyed.

4. The cryptographic keys used during encryption of data will also be destroyed. This can be accomplished by overwriting (zeroization of) the media that holds the keys.
4.3 Key Management

Key management is the most complex part of any security system dealing with encryption of data. The following are best practice guidelines for managing cryptographic keys:

1. An organization must define its cryptographic security and key lifecycle management policy before engaging with a CSP to store data. NIST, specifically the Computer Security Division’s (CSD) Security Technology Group (STG), provides ample guidance such as SP 800-57 Part 1, SP 800-131A & B, Cryptographic Algorithm Validation Program (CAVP) and Cryptographic Module Validation Program (CMVP).
2. Cryptographic keys for encryption of data should be generated in a secure way. This includes the use of an appropriately strong random number generator (RNG) and key management should be centralized.
3. Cryptographic keys should never be transmitted or stored in the clear.
4. The cryptographic keys should be stored inside a secure element, such as a smart card, or a service with the hardware security module (HSM).
5. Access to cryptographic keys should be granted to authorized users only. When possible, quorum approaches should be implemented that require a minimum of two people, teams or organizations so that no single entity can access keys and to force conspiracy or collusion to be required for malicious key compromise and/or key misuse.
6. It should be possible to escrow keys used for data encryption. Private Keys which are to be used for legal purposes such as digital signing should never be escrowed.
7. Several options for the management and storage of encryption keys should be available. For example, encryption keys may be stored, on a different server than the data, or delegated to the owner of the data or to a third party provider of such services.
8. Key Management Service shall be compliant with current standards such as NIST and FIPS. For example, Key Management Service should follow the recommendations in NIST Special Publication 800-57 part 1, 2, and 3.

4.3.1 Securing the Client

Cloud computing services that store encrypted data are made possible by a combination of server and client side components. In general, there has been a far greater focus on server side components and, since resources are always at a premium, this typically is done at the expense of client side components which may be vulnerable to compromise. Clients should not be the “forgotten half” of the cloud story. Discussion is building on the issue of consumer devices and the practice of Bring Your Own Device (BYOD) to business environments. The most comprehensive coverage therefore integrates Client and Server as complementary components of the larger context. This is especially important as more users rely on personal client devices, such as smart phones, for accessing cloud services. These devices run applications that tout ease of use, and may address security as an afterthought. Designers of cloud storage should keep this reality in mind and create data protection systems that are both appropriately secure and remain easy to use with as little impact on cloud users as feasible. Cloud providers, vendors and device OEMs should offer end-user device security controls and compliance “device health” checking. Clients and consumer devices need to implement local storage encryption. Content should
not be decrypted at the cloud edge so that it to be consumed by devices that can’t download and decrypt the encrypted content it receives from cloud storage. A backdoor Trojan, keystroke logger, or some other type of malicious software running on a client device can undermine the overall security of cloud storage services.

4.3.2 Client-Side Encryption

Encrypting data before it is sent to the cloud storage provides higher levels of security comparing to encrypting in the cloud, because it prevents the cloud provider from seeing or accessing the original data and limiting the damage if data is later compromised at the CSP. It also allows greater user control over the generation and storage of encryption keys. For example, keys can be stored in a smart card or other hardware tokens that remain in the possession of the user, thus providing greater assurance that no one but the user can decrypt data stored in the cloud. Client-side encryption may be necessary for regulatory compliance, such as HIPPA, PCI DSS, and other compliance series. Given the current state of maturity of cloud provider encryption services, encryption of content by the customer is an excellent mitigation. However, not all cloud customers have the resources to self-provide an encryption capability. Though mature enterprise IT departments may be able, many start-ups and small and medium sized businesses may not.

4.3.3 Endpoint Devices and Applications

Because of their widespread use and universal acceptance as de-facto client applications, web browsers are key elements in providing client-side access to cloud computing services such as data encryption. However, the open architecture of web browsers, where additional plug-ins and browser helper objects can be installed by users, can introduce security problems. The multitude of available and often required browser plugins and mobile code can create an opportunity for user sessions to be hijacked (for example, through man in the middle attacks) and thereby access to encrypted cloud data can be compromised. Adequate measures should be taken to authenticate users in a secure manner to protect their online session from malicious use. For example, cloud providers and/or vendors should enable multi-factor authentication capability for end-users. Refer to the Identity Security as a Service document.

Additionally, mobile device applications increasingly provide a front-end to cloud-based services and may be the target of attacks designed to gain access to data and services in the cloud. Care should be taken to ensure that all access points are sufficiently secure to not introduce new vulnerabilities or compromise the security of cloud services, including encryption. Key management is especially important in this area, as keys stored on the mobile application could be exposed if the device is lost.

4.3.4 Protection of Keys

Because keys are critical components in any cloud storage, adequate key management measures should be adopted to handle them securely, which include provisioning, storage, usage, archive, and deprovisioning. In addition, keys must be carefully protected. Customers should be given the option to store keys separately using a Hardware Security Module (HSM) or secure elements such as a smart card, TPM, a secure USB token, etc. Root keys which are ultimately responsible for the encryption of all content or a broad domain of content should
be protected in hardware. This has been a challenge for cloud providers thereby leaving customers with less assurance than they want or are aware of.

4.3.5 Policy and Enforcement

Security policy is an important consideration when implementing cloud encryption solutions. Requirements for encryption to conform to established security policy do not stop at the on-premise perimeter. The use of encryption through cloud providers must effectively extend and support the same security policy requirements as are already in use for on-premise processing. Addressing policy issues requires appropriate consideration as to what is encrypted, when it is encrypted, and how. If the customer has been operating without sound guidance and policy documentation based on its own determination of acceptable risk tolerance and expected controls and behaviors, the customer needs to invest time to understand the value of the data it is about to move to the cloud, determine its risk appetite, and create policy and requirements that it expects the cloud provider to meet. Encryption is one of the most important and necessary controls for protection of valued data. The customer must locate encryption as a service from the cloud provider, third party, or create its own encryption capability.

As it relates to the use of cloud services, the issues of what is encrypted, and when, should be addressed similarly to on-premise methods through the establishment of data classification and handling strategies. This classification can then be implemented within information workflows using solutions such as Data Loss Prevention (DLP) mechanisms that can identify policy violations and then enforce appropriate use of information or at least inform data owners that their data was not adequately protected and/or labeled.

The issue of how data is encrypted in the cloud must also align appropriately with on-premise policy. For example, on-premise policy requirements such as those that define which encryption algorithms are approved for use, or that determine what is the minimum complexity required for passphrase use, must be applied consistently across all on-premise and cloud compute and storage touch points for that organization’s valued data. These policies issues and the accompanying enforcement measures must extend to cover data as it moves to and from cloud providers to ensure that the use of encryption continues to meet all mandates for data protection.

Key revocation policy and the associated mechanism are critical for all key management models. Therefore, the key management service should enable key revocation. For example, when an employee leaves the organization or changes job function, the key management service needs to revoke the key(s) that the employee uses to access that encrypted data that he should no longer have access to.

4.3.6 Data Integrity

A consideration that should not be overlooked when implementing cloud services is data integrity. The use of data encryption alone may not provide sufficient assurance that encrypted information has not been altered. Files placed into the cloud for storage, particularly if stored over long periods of time or that are in transit through cloud services, may be subject to tampering or replacement. Encryption alone cannot detect this. Combining data encryption with integrity protections such as digital signatures can ensure that data in the cloud
remains both private and authentic. Where available, use of trusted time should be considered by using time-stamped signatures on data.
5.0 References and Useful Links


