Aalto University
School of Science

Master's Programme in Computer, Communication and Information Sciences

Modernizing key management in the transition to public cloud

Venla Kuosa

Master's Thesis
2025

Aalto University
School of Science

Author Venla Kuosa

Title Modernizing key management in the transition to public cloud

Degree programme Computer, Communication and Information Sciences

Major Security and Cloud Computing

Supervisor Prof. Tuomas Aura

Advisor Janne Kauppinen

Collaborative partner OP Financial Group

Date 3 September 2025 **Number of pages** 77 **Language** English

Abstract

Cryptography is a core enabler of cybersecurity, but without the effective management of cryptographic keys, the security of services may be compromised. In the financial sector, requirements for cryptography arise from complex regulations and intricate IT environments, demanding robust key management practices. The transition from on-premises datacenters to a public cloud platform represents a fundamental architectural shift, creating opportunities to improve existing practices while introducing new challenges in meeting evolved requirements.

The goal of this thesis was to develop evaluation criteria based on requirements in the financial sector, propose architectural approaches for a cloud-based key management system (KMS), and examine their differences against the defined criteria. The proposed designs comprised three options: relying solely on the cloud service provider's platform-native KMSs, combining platform-native and external KMSs, and hosting a self-managed KMS within the cloud platform. The research identified architectural choices that significantly influence resiliency, flexibility, developer experience, and expenses in providing financial services.

The findings indicate that all the evaluated approaches can deliver secure and effective key management. Relying on platform-native solutions provides the most seamless user experience and synergies with other platform components but creates a strong dependency on a single service provider and its KMS offerings. Incorporating an external KMS can substantially improve resiliency and flexibility, particularly under uncertain regulatory conditions, but also adds complexity to governance and system integrations. A self-hosted KMS enables greater control over technology choices and customizability but shifts more responsibility to the financial organization and increases operational costs. The study concluded that organizations can determine the optimal approach based on their use cases, risk tolerance, available assets, and technology choices.

Keywords key management, key management system, cryptography, public cloud, cloud transition, cybersecurity

Aalto-yliopisto
Perustieteiden
korkeakoulu

Tekijä Venla Kuosa

Työn nimi Avainhallinnan modernisointi osana julkipilvisiirtymää

Koulutusohjelma Computer, Communication and Information Sciences

Pääaine Security and Cloud Computing

Työn valvoja Prof. Tuomas Aura

Työn ohjaaja Janne Kauppinen

Yhteistyötaho OP Ryhmä

Päivämäärä 3. syyskuuta 2025 **Sivumäärä** 77 **Kieli** englanti

Tiivistelmä

Kryptografia on keskeinen osa kyberturvallisuutta, mutta ilman tehokasta avainhallintaa palvelujen turvallisuus voi vaarantua. Finanssialan organisaatioiden avainhallintaan kohdistuu lainsäädännön ja monimutkaisten teknisten ympäristöjen asettamia vaatimuksia, mikä edellyttää luotettavia avainhallintakäytäntöjä. Siirtyminen paikallisista konesaleista julkiselle pilvialustalle merkitsee perustavanlaatuista muutosta organisaation IT-arkkitehtuuriin. Tämä tarjoaa oivan mahdollisuuden parantaa olemassa olevia avainhallintakäytäntöjä luoden kuitenkin samalla uusia haasteita avainhallinnalle vaatimusten muuttuessa ympäristön mukana.

Tämän diplomityön tavoitteena oli kehittää vertailukriteeristö arkkitehtuuriratkaisulle tutkimalla avainhallintaan kohdistuvia vaatimuksia finanssialan organisaatioissa, laatia ratkaisuvaihtoehtoja pilvipohjaiselle avainhallinta-arkkitehtuurille sekä arvioida näitä vaihtoehtoja vertailukriteeristöä hyödyntäen. Ehdotetuissa ratkaisuissa huomioitiin kolme vaihtoehtoa: pilvialustan natiivien avainhallintakyvykkyyksien käyttö, natiivien kyvykkyyksien sekä ulkoisen avainhallintajärjestelmän yhdistäminen sekä itse hallinnoidun avainhallintajärjestelmän rakentaminen pilvialustalle. Tutkimus osoitti, että arkkitehtuurivalinnalla voidaan merkittävästi vaikuttaa finanssipalveluiden häiriönsietokykyyn, joustavuuteen, kehittäjäkokemukseen ja kustannuksiin.

Tulokset osoittavat, että kaikki esitetyt ratkaisuvaihtoehdot voivat tarjota turvallisen ja tehokkaan avainhallintaratkaisun. Pilvialustan natiivit kyvykkyydet tarjoavat suoraviivaisimman käyttäjäkokemuksen ja synergioita alustan muiden komponenttien kanssa, mutta luovat vahvan riippuvuuden yksittäiseen palveluntarjoajaan. Ulkoisen avainhallintajärjestelmän hyödyntäminen voi parantaa merkittävästi häiriönsietokykyä sekä joustavuutta muuttuvien regulaatiovaatimusten vallitessa, mutta monimutkaistaa hallintotoimia ja järjestelmäintegraatioita. Itse hallinnoitu avainhallintajärjestelmä mahdollistaa vapaamman teknologiavalinnan ja räätälöitävyyden, mutta siirtää enemmän vastuuta organisaatiolle ja kasvattaa operatiivisia kustannuksia. Johtopäätöksenä voidaan todeta, että finanssialan organisaatiot voivat valita itselleen optimaalisen arkkitehtuurin arvioimalla käyttötapauksiaan, riskinsietokykyään, käytettävissä olevia resurssejaan sekä teknologiavalintojaan.

Avainsanat avainhallinta, avainhallintajärjestelmä, kryptografia, julkinen pilvi, pilvisiirtymä, kyberturvallisuus

Preface

I wish to thank Professor Tuomas Aura for his valuable feedback, expertise, and perspective on the topics discussed in this thesis.

I also want to express my gratitude to my instructor Janne Kauppinen for his time, insights, and patience in advising me throughout this writing process, as well as for serving as a role model in my professional career.

Helsinki, 3 September 2025

Venla Kuosa

Contents

Abbreviations

API	Application Programming Interface
AWS	Amazon Web Services
BC	Business Criteria
BPR	Business Process Re-engineering
BYOK	Bring Your Own Key
CA	Certificate Authority
CAF	Cloud Adoption Framework
CCM	Cloud Controls Matrix
CIA	Confidentiality, Integrity, Availability
CI/CD	Continuous Integration and Continuous Deployment
CLI	Command-line Interface
CNAPP	Cloud Native Application Protection Platform
CSA	Cloud Security Alliance
CSF	Cybersecurity Framework
CSP	Cloud Service Provider
CSPM	Cloud Security Posture Management
DORA	Digital Operational Resilience Act
DPF	Data Privacy Framework
DR	Disaster Recovery
DSS	Digital Signature Standard
EBA	European Banking Authority
eIDAS	Electronic Identification, Authentication and Trust Services
EMV	Europay, Mastercard, Visa
FC	Functional Criteria
FIPS	Federal Information Processing Standard
GCP	Google Cloud Platform
GDPR	General Data Protection Regulation
GUI	Graphical User Interface
HSM	Hardware Security Module
IaaS	Infrastructure as a Service
IAM	Identity and Access Management
ICT	Information and Communication Technology
ISO	International Organization for Standardization
KMS	Key Management System
MAC	Message Authentication Code
NIST	National Institute of Standards and Technology
OC	Operational Criteria
PaaS	Platform as a Service
PCI	Payment Card Industry
PIN	Personal Identification Number
PSD2	Second Payment Services Directive

PQC	Post Quantum Cryptography
RTS	Regulatory Technical Standards
SaaS	Software as a Service
SIEM	Security Information and Event Management
SLA	Service Level Agreement
SRM	Shared Responsibility Model
SSH	Secure SHell
SSO	Single Sign-On
SWIFT	Society for Worldwide Interbank Financial Telecommunication
TLS	Transport Layer Security

1 Introduction

Key management is an essential component in secure cryptography. Without effective key management practices, the confidentiality of encrypted data, the integrity of digital signatures and the availability of encrypted data or ability to verify signatures are at significant risk due to potential key compromise or key loss. During a transition from an on-premises architecture to a public cloud platform, the environment experiences a comprehensive reengineering, which requires modernization of the organization's key management approach and presents an opportunity to evaluate and enhance existing key management practices. In the financial sector, security and resilience are essential due to its critical role in society. Therefore, the key management strategy also encounters various requirements.

The goal of this thesis is to examine different approaches to key management during and after transitioning from on-premises datacenters to the public cloud. Furthermore, this thesis explores which factors affect the selection of the key management approach, and proposes a plan to apply the selection criteria for choosing the most suitable key management approach.

The motivation for this research is to take the opportunity to assess strengths and weaknesses of the current key management capabilities, evaluate how to retain current best practices, and strengthen the weak spots in the current key management capabilities while mitigating possible risks related to key management during and after the cloud transition. The methodology used in this thesis involves solution design, definition of evaluation criteria, and systematic assessment of solutions using technical analysis. The goal of this thesis is to answer the following questions:

1. What are the requirements for a cloud-based key management solution in the financial sector?

2. What are the possible approaches to key management during and after a public cloud transition?

3. How can approaches be evaluated to find the most suitable one for key management for a financial organization?

The outcomes of this thesis outlined the most significant functional, operational, and business considerations for a key management approach in a financial organization and developed evaluation criteria based on the requirements. Then, three architecture suggestions were constructed using cloud-native, external, and self-hosted key management solutions. The suggestions were evaluated using developed criteria, and the research showed that different key management architectures differ in resiliency, flexibility, developer experience, and cost of the solution.

The scope of this thesis covers key management approaches for a financial organization's transition from a complex on-premises environment towards a single-CSP architecture. It also provides insights to decision-making on implementing key management on a public cloud platform in EU-based organizations.

The rest of this thesis is structured as follows. Chapter 2 defines core concepts related to cloud and key management. Chapter 3 describes the challenges in key management migration. Chapter 4 explores the factors that may affect the choice of the key management approach during and after the cloud transition. Chapter 5 investigates different solutions to key management architecture and proposes three approaches for key management during and after cloud transition. Chapter 6 evaluates the differences based on the criteria defined in in Chapter 4. Chapter 7 discusses remaining topics and further research.

Artificial intelligence was used in selected parts of this thesis to improve clarity and readability by suggesting word choices, alternative expressions, and revised sentence structures.

2 Background

This chapter presents an overview of key management and introduces the fundamental concepts of cloud computing necessary for understanding the target environment in the cloud transition. It then outlines the initial conditions typically found in on-premises architectures and describes the process as well as expected outcomes in a public cloud transition. The focus is placed on the security implications of this transition, with particular emphasis on the process of transitioning key management from on-premises to cloud.

2.1 Key management

Key management refers to the administration of the lifecycle of cryptographic keys [1], which are most commonly used in data encryption and digital signatures. These cryptographic operations rely either on a single key for symmetric cryptography or a pair consisting of a secret key and a public key for asymmetric (or *public-key*) cryptography [1]. Key management involves the establishment of policies and procedures for key generation, storage, usage, distribution, validity period, revocation, deletion, access management, and auditability [1]. Moreover, it encompasses the technical solutions required to incorporate these requirements into a system context. Next, this chapter proceeds to elaborate on the purpose and dimensions that are expected to be covered by key management, presents tools for key management, and explores their use cases. Special attention is given to the different types of cryptographic keys typically managed with these tools, as well as those which fall outside the scope of such systems.

2.1.1 Purpose of key management

Key management plays an important role in cryptography-related operations, which are an essential part of cybersecurity. Security is often described as the combination of confidentiality, integrity, and availability – the CIA triad. In the context of cybersecurity, the most common applications of cryptography are encryption to ensure data confidentiality and digital signatures to ensure data integrity. However, if the keys used in these operations are exposed to unauthorized parties, both confidentiality and integrity are compromised. Although these cryptographic operations are not directly related to the availability of data, cryptographic key management is essential. Encrypted data becomes inaccessible if the encryption key is lost, and digital signatures cannot be verified in the absence of the corresponding verification key.

When discussing applications of cryptography, the CIA triad is often extended to include three additional concepts: authentication, authorization, and non-repudiation [1]. Table 1 summarizes the impact of key management in each of these use cases. Authentication and authorization are the two fundamental cryptographic services used in access control, which itself is a core element of cybersecurity, specifically within the domain of identity and access management (IAM). Authentication verifies the identity of the requester, and authorization verifies whether the identity has been granted access to the requested operation. Section 2.1.5 returns to the topic of different identity-related secrets.

Authentication of integrity, authentication of source, and non-repudiation – which refers to the proof of undeniability of messages or actions – are concepts associated with integrity verification. Although their cryptographic implementations differ, their functionality can be conceptually grouped under the *integrity* component of the CIA triad.

Various attack vectors have been identified to target cryptographic techniques in attempts to break encryption or forge signatures. Examples include side-channel attacks aimed at revealing a key or parts of it, replay attacks that reuse the key from valid transmissions, and brute-force attacks that attempt to guess the correct key. With current computing capacity, standardized algorithms, and recommended key lengths, such attempts require significant resources and sophistication [2]. When key management follows established standards and recommendations, these attack vectors remain largely theoretical. However, by gaining direct access to cryptographic keys, an attacker can execute unauthorized cryptographic operations, such as decrypting sensitive data or signing malicious messages or code with stolen keys.

Key compromise resulting from poor key management practices is the most probable attack vector against cryptographic security, often involving human error or inadequate knowledge of secure key handling procedures. It is important to recognize that not everyone handling secret keys is an expert in key management. Consequently, the organizational cybersecurity strategy must incorporate controls for key management in the cryptography domain. Effective key management requires a combination of governance practices, processes, and policies, and the provision of technical capabilities throughout the organization. Thus, key management requires both technical solutions and governance to align with key management best practices.

Although cryptographic algorithms provide mechanisms to ensure data confidentiality and integrity, key management must ensure practices to preserve the key material used in these operations, or the availability of data may be compromised. For instance, in typical ransomware attacks, an adversary encrypts the victim's data and demands ransom in exchange for the decryption key. A similar loss of data can occur unintentionally if encryption keys for databases or other types of storage are lost or destroyed without proper backup measures. In such cases, access to encrypted data is permanently lost. Key management addresses these risks through resilient and secure key storage, backup policies, disaster recovery strategies, and business continuity planning.

While cryptography acts as an enabler of security and privacy, it does not, in isolation, guarantee comprehensive security. Additional security mechanisms are essential to defend against diverse threats. In particular, IAM is closely linked to key management. This relationship extends beyond controlling who is authorized to use and manage cryptographic keys, as IAM also determines who is allowed to access data that is encrypted in storage.

In conclusion, key management has a significant effect on all three aspects of security – confidentiality, integrity, and availability – along with additional services to ensure the authenticity of operations conducted in digital environments, summarized

Confidentiality	Encryption is used to implement data confidentiality to prevent data being visible plaintext. Key management provides means to keep encryption keys safe and available to only authorized parties, and defines rules for using strong algorithms for key generation and requirements for key length. Stolen, exposed or weak encryption keys can compromise data confidentiality by creating an opportunity for an attacker to decrypt data and gain access to confidential information.
Integrity	Digital signatures and message authentication codes (MACs) are used to verify data integrity. Key management provides means to keep signing key material safe and available to only authorized parties, and defines rules for using strong algorithms for key generation and requirements for key length. Stolen, exposed or weak signing keys can compromise data integrity by forging signatures or MACs for tampered data.
Availability	Availability of cryptographic keys is a requirement for accessing encrypted data or verifying signatures. Key management implements safeguards, backup processes and disaster recovery practices to ensure preservation of key material in case of an incident. Lost encryption keys lead to losing encrypted material and the inability to verify signatures render signed data invalid.
Authentication	Key management protects authenticity of identity, integrity or origin. If an authentication key is stolen, it can be used to steal an identity, fake the origin of a message, or fake the proof of integrity of a message. Authentication is related to a more broad concept of credential management, discussed in section 2.1.5.
Authorization	Cryptographic keys can be used in authorization, i.e., granting an entity certain access to a resource. Key management outlines policies for frequent rotation of authorization keys. Typically, authorization keys and other authorization tokens relate to the broader concept of IAM.
Non-repudiation	Digital signatures are used for non-repudiation, referring to the proof of both integrity of data and the entity signing the data. Key management provides means to keep signing key material safe and available to only authorized parties and defines requirements for strong key generation. Stolen, exposed or weak signing keys can compromise non-repudiation by enabling an attacker to steal the identity of the key owner and sign their own messages.

Table 1: The impact of key management in the use of cryptographic solutions.

in Table 1. The foundational objectives of key management are to ensure that keys are accessible only to authorized entities, securely stored, and reliably available when needed. To meet these objectives, key management best practices must be seamlessly embedded into daily operations of development and IT administration. The next section describes processes involved in the key lifecycle.

2.1.2 Key lifecycle

The cryptographic key lifecycle begins at key generation and ends with the secure destruction of the key material. Figure 1 illustrates the various states throughout the key lifecycle, along with the transitions between these states. In the **pre-operational phase**, key generation and activation are the most important operations to support [1]. Keys must be generated with sufficient strength, meaning they must have adequate key length and entropy used with standardized algorithms. This ensures weak keys are not used in cryptographic operations.

Upon activation, it is essential to define an appropriate validity period (also referred to as *cryptoperiod* [1]). A validity period that is too long increases the likelihood of key exposure during its operational phase, extends the window during which a potentially compromized key can be used, and risks the generation parameters becoming obsolete during the key's lifetime [1]. Key activation may also involve key distribution, which is a crucial step in safeguarding secret key material. Secure key distribution ensures that keys are transmitted without interception and are delivered only to authorized recipients. Key management policies must define and enforce secure distribution mechanisms.

During the **operational phase**, secure storage of the key is crucial. Access must be restricted strictly to authorized entities and incorporate continuous processes for

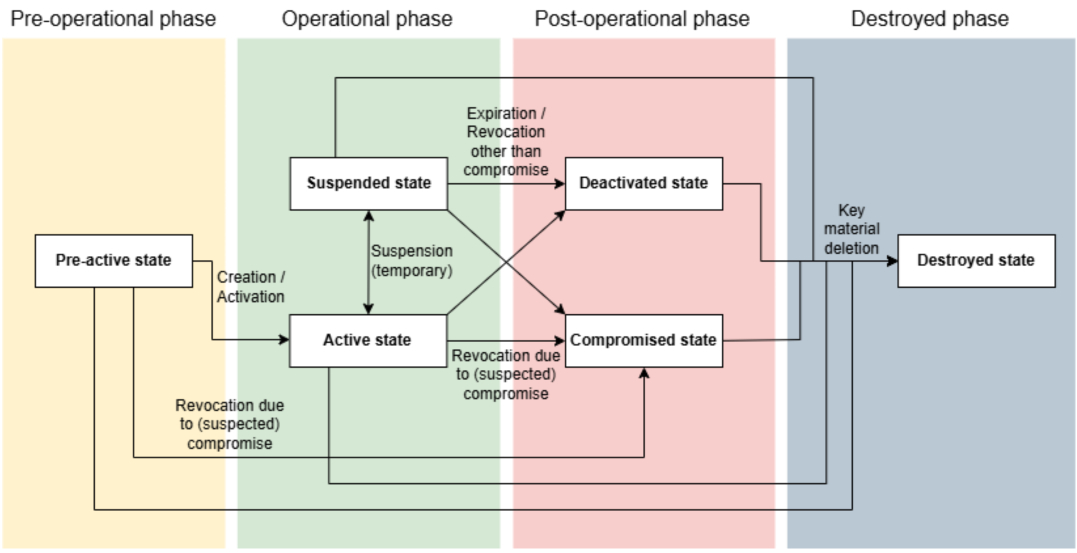

Figure 1: Key lifecycle phases, key states and transitions according to NIST Recommendation for Key Management [1].

maintaining and reviewing access rights. In addition to technical solutions for secure storage and usage, key management requires comprehensive logging of key usage and management operations, such as changes in access control. Audit logs must be maintained for all events involving the key – not only during its operational phase but throughout all phases of its lifecycle. In some cases, logs may be supplemented by *ceremony documents*, which serve both as a runbook and as a formal record of key management activities.

Disabling the use of a key may occur temporarily through suspension, or permanently when transitioning a key to the **post-operational phase** though revocation or expiration. These actions disable the keys *protect functions*, referring to generative operations like encryption and signing. However, key material remains available for *process functions*, such as decryption or signature verification. This approach ensures that key suspension, revocation, or expiration does not result in data loss or invalidated signatures due to the unavailability of required keys.

Often, the need for a key extends further than the validity period of the key. Regular key rotation is a best practice which limits the length of the key's validity period. Key rotation refers to replacing the current key with a new one, either manually or automatically. Automated key rotation reduces the risk of human error and ensures timely rotation. Once the new key is active and in use, the old key can be securely retired.

The final step in the key lifecycle is key destruction, transitioning the key to the **destroyed phase**. Secure destruction ensures that the key material cannot be used for unintended purposes. Although the key material is destroyed, metadata and associated audit logs are archived for a required period of time. To prevent accidental destruction of critical keys, many key management tools offer a *soft delete* feature, allowing recovery of deleted keys within a defined grace period.

Central management of keys enables comprehensive key inventory management across the organization. The inventory provides visibility into key attributes such as key lengths, algorithms, and validity periods, supporting both vulnerability mitigation and regulatory compliance in the cryptography domain. Key inventory should also include metadata such as key ownership, associated use cases, and system information, which supports incident resolution and initiating changes needed for policy compliance. Automated monitoring and analytics can enhance the effectiveness of inventory management, and the remaining manual work done by the organizations' cryptography specialists is effectively supported by a central point of management.

2.1.3 Key management solutions

Key management can be supported through specialized tools designed to provide functionality for managing and using cryptographic keys throughout their lifecycle. A *key management system* (KMS) is an umbrella term that encompasses all solutions that implement such features. These typically include interfaces and functionality for key lifecycle management and the execution of cryptographic operations as well as a secure storage storage solution for key material [3]. Key management tools are available as software products, physical hardware appliances, and as a service.

At the baseline, key management tasks can be performed manually. However, as the scale and complexity of cryptographic operations increase, manual methods become impractical and increase the risk of cryptography-related vulnerabilities due to negligence or human error [4]. The primary goal of key management tools is to reduce reliance on manual processes and individual expertise, thereby ensuring that best practices are consistently applied. Next, this chapter discusses some of the most common tools to support effective key management.

Software-based key management systems run on conventional server hardware. These solutions range from vendor-specific tools designed for integration with certain products (e.g., IBM Security Key Lifecycle Manager [3]) or particular use cases (e.g., SSH key management [5]), to general-purpose systems that support a wide array of integrations (e.g., HashiCorp Vault [6]). While specialized solutions may not support a comprehensive centralized key management approach, they often provide seamless integration capabilities in their intended use case, minimizing the workload for developers and administrators compared to more general-purpose tools.

A **hardware security module** (HSM) is a physical appliance equipped with all the necessary features for key management operations throughout the key lifecycle [7]. HSMs allow cryptographic computations to be performed entirely within a secure execution environment inside the HSM, ensuring that secret keys never have to leave the device in plaintext [7]. HSM acts as secure key storage and guarantees that keys are never exposed through HSM appliance memory using security measures such as tamper resistance [7].

A **service-based KMS** refers to a KMS residing in a cloud, based on the Software as a Service (SaaS) model. The cloud service models are elaborated in Section 2.2.2. These systems are typically priced based on usage, rather than requiring the upfront investment on the software licenses or hardware, as these components are managed by the service provider. As a result, key management administrators can focus on tasks related to the keys themselves without the additional burden of maintaining the underlying infrastructure.

The Federal Information Processing Standard (FIPS) 140-3 (and its predecessor, 140-2, also still valid) defines security requirements for cryptographic modules, including HSMs, on different sensitivity levels [8]. Typically, HSMs are offered as FIPS 140-3 (and preceding 140-2) level 3 compliant appliances, which guarantee tamper resistance and deny the possibility to export keys in readable form [7]. Software-protected KMS solutions may qualify for FIPS 140-3 level 1 compliance [9], which, while offering a lower assurance level, still serves as a reference point for assessing security. Compliance with FIPS standards may be mandatory based on industry requirements, further discussed in Chapter 4.3.1. Importantly, these standards serve as references for the security guarantees provided by key management tools.

KMS providers are typically companies specializing in security, and more specifically, cryptography. Each category, software, hardware, and service-based KMS offers platform-agnostic KMS, meaning they can be integrated with both on-premises systems and major public cloud platforms. These services are designed to follow the key management best practices described in the earlier sections of this chapter, also enabling a central inventory of cryptographic keys.

In addition to security-specialized providers, all major public cloud service providers (CSP, see Chapter 2.2) offer platform-native services for key management within their own environment. They integrate seamlessly with the provider's other platform-native products while also supporting use cases beyond the cloud platform, including on-premises environments and other cloud platforms. Hence, the cloud platforms' KMS solutions can be used in hybrid or multi-cloud architectures, offering flexibility alongside native integration benefits.

2.1.4 Key types in scope

This thesis focuses on enterprise key management in the financial sector. The key management solution in this context must support a wide range of use cases – both those that are currently implemented and potential future applications – while also adapting to the evolving technological landscape and emerging threats, such as those posed by quantum computing. However, not all financial organizations are able to fully migrate to the public cloud. For example, the global infrastructure for payment communications and its direct customers have specific requirements for physical devices in key management. Organizations that directly connect to the backbone of the Society for Worldwide Interbank Financial Telecommunication (SWIFT) network are required to use a specific on-premises *Swift HSM* [10]. Therefore, such organizations fall outside the scope of this thesis.

Fortunately, physical appliance requirements affect only service bureaux providing SWIFT connectivity and large global institutions with direct access to the SWIFT network [10], not typical financial organizations which access the SWIFT network through said service bureaux. Instead, the KMS architecture designs presented in this thesis support use cases such as signing transferred data between financial organizations and SWIFT service bureaux. Beyond this, digital signatures are used in banks' internal payment infrastructures as well as inter-bank connections, further emphasizing the need for efficient key management. Payment card operations introduce additional needs for the KMS: keys for personal identification number (PIN) generation and verification, card verification, and the *Europay, Mastercard, Visa* (EMV) card master keys [11] – to name a few – are handled by many financial organizations and must be securely managed. Payment card keys face strict requirements from the Payment Card Industry (PCI), which are discussed later in sections 3.1 and 4.3.1.

Although the financial infrastructure presents specific use cases for cryptography, the underlying cryptographic functions follow the same standards as in any other sector. In the financial sector, the importance of data encryption is pronounced. Information that is stored, processed, and produced in financial organizations is highly sensitive. With the widespread adoption of online banking and digital financial transactions, financial organizations must continuously adapt to ensure the security of customer data and funds in the digital era.

In addition to data encryption, digital signatures play a vital role in ensuring data security. Financial services can use digital signatures, in combination with encryption, to maintain the integrity of transmitted information. Moreover, since developing software for digital services is a core function of modern financial institutions, code signing is commonly employed to guarantee the integrity and authenticity of the software. Data security in both internal and externally available services is essential. Especially in complex systems like financial organizations' IT environments, custom certificate authority (CA) infrastructures may be needed, and the CA root keys may be stored in the KMS, forming the basis for the chain of trust for issued end-entity certificates.

While data encryption and integrity protection are important across all industries, the financial sector is classified as a critical sector [12] and experiences prominent and constantly evolving regulatory obligations in addition to industry-driven self-regulation. The complexity arising from regulatory frameworks affects technical implementations in financial organizations, further discussed in section 3.3, and creates requirements for the KMS architecture as well. These requirements are outlined in section 4.3.1. As a consequence of the critical role in society, the financial sector is a frequent target of cyber threats. According to the CERT-EU Threat Landscape Report from 2024 [13], financial organizations are among the most frequent targets of cyber attacks. Robust key management tools and processes are required to protect organizations from successful attacks.

The key management modernization in this context focuses on the encryption keys used for data at rest and, in some cases, data in transit. Furthermore, signing keys used for digital signatures should be securely stored in a KMS. However, some use cases of cryptographic keys are not traditionally managed in a centralized key management solution. Notable exceptions include transport layer security (TLS) certificates (specifically, the private keys associated with certificates) and secure shell (SSH) keys, which are most commonly stored in encrypted form in device storage. Most incidents related to TLS certificates are caused by certificate expiry, particularly due to failures in timely renewal or deployment of new certificates [14]. As a result, certificate and SSH key management are typically implemented using specialized tools that enable automation, scanning and monitoring [14][5]. In cloud environments, however, the role of a KMS solution must be re-evaluated for these use cases. It is also important to note that KMS solutions are typically designed for managing the lifecycle of long-term master keys rather than short-lived session keys. There is no practical need for similar comprehensive management of ephemeral secrets.

Finally, to further address the keys used in authentication instead of data confidentiality and integrity protection, the concept of workload identity management must be considered. In this context, a workload refers to technical entities such as applications, devices, or containers. For access control purposes, workloads are assigned credentials, which can be implemented using cryptographic keys. In cloud environments, workload identities are typically managed using native IAM solutions, separate from the KMS [15]. Therefore, the KMS architectures presented in this thesis are not intended for entity authentication purposes. It is technically possible to use the key management architecture solutions discussed in this thesis as a component of workload identity management as underlying cryptographic algorithms follow the same standards, but a KMS alone does not offer a comprehensive solution for managing workload credentials.

2.1.5 Relation to secrets management

Whereas key management focuses on managing the lifecycle of cryptographic keys, secrets management is a broader concept referring to the handling of all types of secret values. These include cryptographic keys but also passwords, technical credentials, and application secrets, such as API keys and access tokens. Unlike the state-driven approach to lifecycle management presented in section 2.1.2, Cloud Security Alliance (CSA) describes five crucial aspects in secrets management: secure storage, access controls, auditing and logging, secret rotation, and secret versioning [7].

Traditionally, key management has been treated as a distinct discipline, separate from the management of other secrets. However, modern KMS solutions often include support for managing free-form secret values. It can be noticed that all of the secrets management aspects outlined by CSA are also essential in cryptographic key management, but the process implementations may vary. Secrets management is more rarely concerned with keeping the secret material within secure boundaries, such as inside a HSM, throughout its lifecycle. Instead, the focus is in maintaining the security of secret values while they are transferred between systems. In secrets management, access control appears as the main point of attention when it comes to secrets' use cases [4]. Referring to the previous chapter, the management of access credentials for authentication is divided between key management and IAM. The solution for managing such secrets must be done on a case-by-case basis.

2.2 Cloud concepts

Cloud computing refers to a wide range of services accessible over a network – typically the internet — that share a certain defining characteristics. Common features include resource pooling, where physical or virtual resources are dynamically allocated to customers according to demand, and automated self-service, which allows customers to provision and manage their resources independently [16]. Pricing models in the cloud are typically subscription-based, and the costs are determined by the measured usage or the allocation of resources provided by the cloud service provider (CSP) [16]. The following sections explore different cloud architecture models and examine the security considerations specific to operating in a public cloud environment.

2.2.1 Cloud deployment models

Cloud deployment models portray how cloud infrastructure is organized and delivered. Public cloud is the most widely recognized cloud deployment model, referring to a model where the computing environment and management interface is provided by a third party [16]. The infrastructure is owned by the CSP and the capacity is shared between the CSP's customers [16]. Services are generally available to the public on a pay-per-use basis [16]. Leading public CSPs are, at the time of writing, Microsoft (Azure), Amazon (AWS), and Google (GCP).

Other deployment models include the private cloud and the hybrid cloud. A private cloud is typically an on-premises environment with cloud computing characteristics, however retaining traditional on-premises limitations, such as fixed computing capacity [16]. The hybrid cloud model describes an architecture where the customer combines two or more distinct cloud environments of any deployment model [16]. Multi-cloud is a subtype of the hybrid cloud model, comprising of two or more public cloud environments. This thesis focuses on a transition to a single-CSP public cloud architecture. However, hybrid clouds present important considerations relating to the selection of an appropriate KMS architecture. The hybrid aspect is therefore discussed when comparing approaches for the modernized key management solution.

2.2.2 Cloud service models

Cloud service models define the division of responsibility between the CSP and the customer acros the layers of the technology stack. At minimum, the CSP manages the physical infrastructure including computing hardware, storage, and network components. The most common cloud service models are Software as a Service (SaaS), Platform as a Service (PaaS), and Infrastructure as a Service (IaaS), ordered from least to greatest customer responsibility over the technology stack [16]. Figure 2 illustrates the division of responsibilities between the CSP and the customer in each model. All major public CSP's offer services across these three models, although smaller vendors specialize in a narrower selection of cloud offerings, such as a single SaaS product.

The division is denoted as the *shared responsibility model* (SRM), which identifies the security and operational responsibilities between the CSP and the customer. The SRM also highlights areas of shared accountability, where both parties must take action to maintain security and ensure continuity of operations [17]. Although responsibilities shift depending on the service model, end-to-end security is never fully guaranteed by the CSP: Typical vulnerabilities in cloud environments stem from customer-side misconfigurations, mismanaged access control, and insufficient monitoring and visibility in cloud workloads – issues caused by skill gap and false assumptions on responsibilities [17].

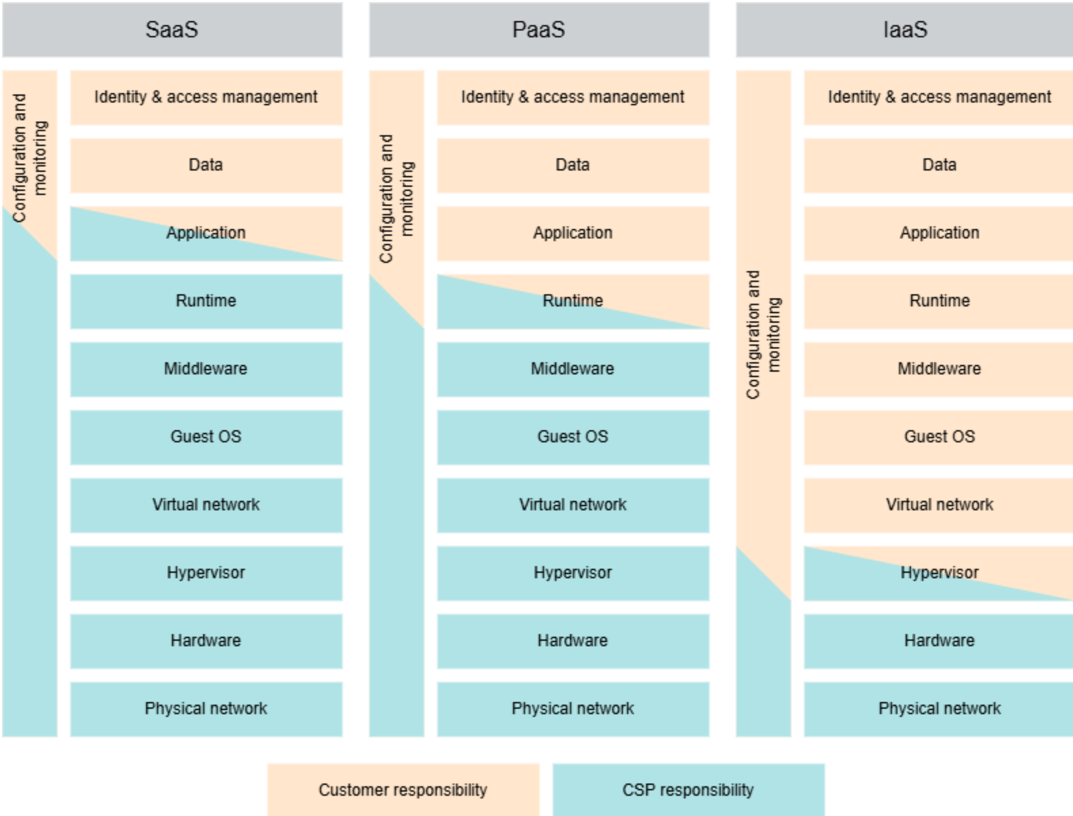

Figure 2: Shared responsibilities over the technology stack in cloud service models based on Tenable [17].

As mentioned in Chapter 2.1.3, KMSs may be offered under a SaaS model. In contrast, a cloud-based HSM is an example of IaaS. A cloud HSM refers to an HSM appliance hosted within the CSP's datacenter, available to customers in varying levels of responsibility. The HSM may be shared or dedicated – a *shared HSM* refers to multi-tenancy, where a single appliance is used by multiple customers. A *dedicated HSM* is allocated to a single customer. While a dedicated HSM may follow the same shared responsibility principles as the shared HSM (also referred to as a *managed HSM*), with the CSP retaining administrative control over the appliance, a more typical motivation for selecting a dedicated HSM is the need for complete control over the HSM. To facilitate this, the CSP grants the customer administrative privileges, eliminating CSP access to the HSM [9]. Cloud HSM offerings may extend to include even *payment HSMs* for real-time payment transactions [9][18], which constitute an essential component in the financial sector's infrastructure, although specific to a particular subset of financial organizations.

CSPs typically offer platform-managed keys used for their platform-native services, such as data storage encryption. These keys provide a convenient means to implement basic cybersecurity requirements, such as encryption of data at rest. In these cases, the CSP manages the entire key lifecycle, and the customer has no control or visibility into key attributes such as key length or rotation frequency. To address the need for greater control, CSPs offer an alternative where customers manage keys themselves

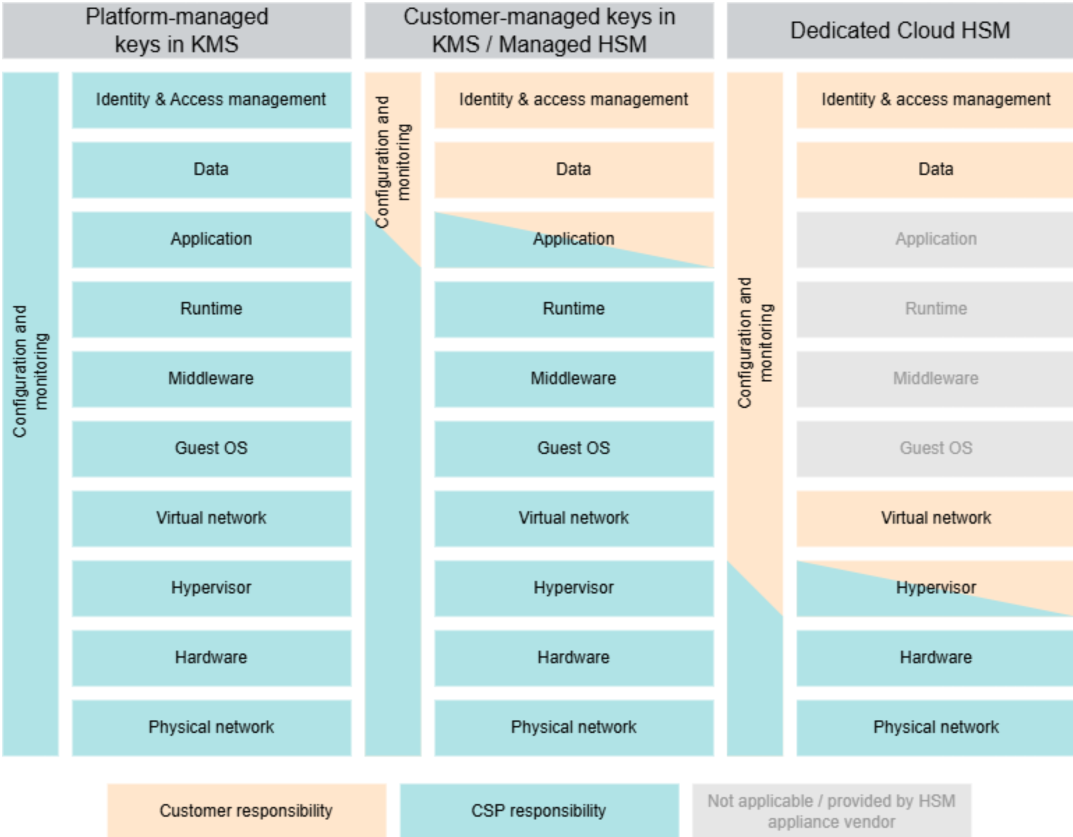

Figure 3: Shared responsibilities in different cloud key management solutions.

with *customer-managed keys*. For such keys, as well as any other keys that are not used for the security of platform-native services, the customer needs a key management solution.

When transitioning to the cloud, it is essential to consider the SRM within key management alongside broader cloud security concerns. Both internal roles and responsibilities within the customer organization and the CSP's role in key management must be clearly defined and understood. Figure 3 illustrates how shared responsibility varies across different cloud key management solutions. Understanding and adapting to the SRM in key management is a critical factor when migrating from on-premises infrastructure to a cloud environment.

2.2.3 Cloud security

Although on-premises and cloud environments share the same fundamental security risks, their respective security landscapes differ significantly. The SRM reduces the responsibilities of the customer organization, but it also decreases the amount of control over the architecture. IAM and component configurations thus become critical, especially since the ease of asset creation – enabled by the ability to provision resources at the push of a button – can blur the boundaries between the competence areas of various professionals.

Major CSPs as well as third-party vendors have recognized the need for monitoring and automating cloud security tasks. The cloud-native approach to security is denoted as a Cloud-Native Application Protection Platform (CNAPP), which unifies multiple security functions into one system [19]. One of the core components of CNAPP is Cloud Security Posture Management (CSPM) [19]. CSPM tools are offered to cloud customers as a means of monitoring and preventing misconfigurations, detecting threats, ensuring compliance with best practices and regulatory standards, enforcing governance policies, and providing a centralized view of the overall security posture within the cloud platform – possibly even extending to other cloud platforms and on-premises environments in hybrid cloud setups [20].

While CSPM has become an essential tool for mitigating common cloud security risks, it is not a silver bullet. The implementation can be a complex process, even when using the CSP's native CSPM tools, which are typically dessigned for seamless integration within the provider's platform [20]. With a hybrid cloud solution, the complexity is further amplified [20]. CSPM solutions offer a wide variety of features and integration possibilities, including log monitoring, security information and event management (SIEM) systems, and identity and access management (IAM) solutions [20].

It is essential to remember that despite CSPM tools being provided by the CSP or third-party vendors, the customer remains responsible for configuring and monitoring their portion of the technology stack in accordance with the SRM. Lack of expertise or understanding, or unquestioningly adopting CSPM suggestions, is not a sufficient approach to cloud security. Instead, effective cloud security requires custom policies, monitoring, and detection practices supported by skilled professionals in cloud computing, security, and networking. Rahman et al. [20] emphasize that regulated industries, such as finance, must not rely solely on CSPM tools for regulatory compliance.

Although the CSP is responsible for implementing security controls for its own portion of the stack, it also serves as an advisor, providing guidance and best practices for secure use of its platform. A cloud environment comprising a vast array of different components, spanning across different service models, relies on well-thought-out planning of architecture and configurations to ensure a secure and maintainable environment.

An organization-wide key management policy forms a vital part of key management. In cloud environments, policies can be set to define controls for key management. These include restricting the use of weak keys and defining requirements for the length of the key's validity period [21]. However, these tools are effective only if the key information is accessible on the cloud platform. With third-party KMSs, key management policies and access control management must be implemented within the KMS itself.

2.3 Transition to public cloud

The transition from a traditional datacenter architecture to a fully modernized cloud-based architecture constitutes a significant transformation. This shift not only demands substantial refactoring of the entire technical environment but also requires a fundamental change in perspectives regarding responsibilities, security, and associated risks. The modernization addresses common challenges in on-premises environments and facilitates the development of financial services. This chapter outlines the primary drivers for cloud adoption and describes the process of migrating the environment from datacenters to the cloud, also incorporating the perspective of key management.

2.3.1 The paradigm shift

In a traditional on-premises architecture, the entire technology stack – as illustrated in Figure 2 in Chapter 2.2.2 – is possessed by the organization within a private datacenter environment. The organization is responsible for managing all components, from the physical infrastructure and network connectivity to the application hosted within the datacenter. Despite the complete control over the assets, it has become a common practice to utilize on-premises installations of licensed software instead of developing own software for each purpose as software products have become available and organizations have been able to make the *make-or-buy* decision.

In large and mature IT organizations, the on-premises environment typically comprises systems at different stages of their lifecycle. The systems range from legacy technologies to modern platforms, resulting in a complex environment with multiple generations of platform architecture. Lekkala [22] highlights the issue of *data silos*, which are a result of fragmented systems lacking interoperability and visibility. The same study identified scalability limitations and constrained analytics capabilities as the most significant challenges in on-premises architecture in the financial sector along siloed architecture. Furthermore, legacy components increase costs through the risk of vendor lock-in and expensive specialists due to the difficulty of changing the technology provider [22].

The first signs of a shift towards cloud architecture include the adoption of SaaS and PaaS systems as a part of the environment [23]. The benefits of these service models over purchasing or making the systems for the organizations' ownership include scalability, a flexible cost model, and outsourcing infrastructure administration and maintenance to the service provider. These acquisitions are often limited to single products or systems, and therefore integrating them with the existing environment through the Internet is relatively simple [23]. While SaaS acquisitions depict the paradigm shift from keeping all operations strictly in proprietary datacenters toward sharing the responsibility over the technology stack, they do not represent the comprehensive architectural transformation required in a large-scale migration to cloud infrastructure.

Transitioning from on-premises datacenters to a public cloud environment is a massive operation. One of the primary considerations is the transfer of control over physical infrastructure to the CSP, along with granting the CSP administrative access to the system components. In the early stages of cloud adoption, financial services were hesitant to host services off-site, perceiving such outsourcing as a significant security risk [23]. However, as cloud platforms have matured and attained certifications that meet stringent financial industry standards – such as ISO 27001 and PCI DSS [23] – this perspective has shifted. Organizations are increasingly acknowledging the long-term risks of maintaining aging on-premises infrastructure, including its inherent security vulnerabilities.

The perceived loss of control over infrastructure components and the reallocation of responsibilities are increasingly outweighed by the benefits offered by cloud platforms. CSPs prioritize the security of their environments [23][24], and tools such as CSPM have demonstrated effectiveness in improving security and mitigating typical on-premises vulnerabilities through advanced monitoring and automation. Capital One, one of the leading banking companies in the United States, is an example of a financial organization that adopted a full-cloud strategy in AWS [22], depicting this shift within the financial sector. Furthermore, other globally significant financial institutions, such as Goldman Sachs and JP Morgan, have adopted a *cloud-first strategy* to remain competitive in the evolution of the financial industry, while retaining certain workloads in on-premises datacenters, thus employing hybrid-cloud strategies [22]. Many emerging financial technology (*fintech)* companies utilize cloud-native technology, and trends indicate accelerated fintech development with other trending technologies, such as artificial intelligence and blockchain, combined with cloud-native environments [24].

From a key management perspective, this paradigm shift introduces critical considerations regarding control over the keys as well as restricting access to the keys. In cloud environments, customers are unable to strictly limit key access to designated personnel within their own organization. The CSP inherently has access to the customer's environment, including cryptographic keys. Even with dedicated HSMs, where administrative access is restricted to the customer, keys may still be accessible to the CSP through HSM credentials used by applications hosted within the same cloud. This raises the risk of key or data leakage through CSP access channels. However, as trust in CSP security controls has increased, concerns about the cloud's perceived insecurity have decreased. Attacks stemming from cloud platform vulnerabilities are now often viewed as an acceptable risk when compared to similar risks in on-premises infrastructure. According to a 2024 survey by CSA, 98 % of financial organizations use cloud computing in some form, with 59 % of respondents reporting that they "store or process regulated banking data in cloud services" [25].

With the SRM, many key management use cases shift to the CSP. Keys associated with the CSP-managed components and platform-managed keys are handled entirely by the provider, which implements its own key management policies and architectures. While this reduces the load of key management for customers and limits the number of use cases for key management on the customer side, it raises challenges regarding compliance and key ownership. This represents one of the major considerations related to key management in the cloud, and is further examined in Chapter 7.

Cryptographic techniques have been proposed as a solution to restrict CSP access to plaintext data stored in the cloud platform. Encrypting sensitive data will make the information inaccessible even when an entity gain access to it from the CSP side. However, in a single-CSP architecture, data typically flows directly from clients into the cloud environment, and therefore, encryption keys that are used within the platform can be also used by the CSP. Therefore, any cryptography solutions for hiding sensitive data from the CSP would require a hybrid cloud environment.

2.3.2 Cloud transition process and expected outcomes

The transition from an on-premises architecture to a cloud environment involves more than merely migrating workloads from physical servers to virtual machines via a lift-and-shift approach. As discussed in the earlier section, the paradigm shift toward favoring cloud computing – even with sensitive and regulated data – is driven by the need to address issues in current architecture and remain competitive in an IT industry that is rapidly innovating through cloud-enabled technologies. To fully gain the benefits of cloud computing, a comprehensive reengineering of the entire IT environment is required, ensuring that the benefits of the change are utilized efficiently.

All three major public cloud service providers offer their own Cloud Adoption Framework (CAF), which is intended for supporting a comprehensive onboarding to the cloud platform. This section briefly outlines the characteristics of AWS CAF [26], Azure CAF [27], and GCP CAF [28] to describe the elements of a cloud transition process. Although each framework incorporates vendor-specific characteristics, they share many foundational principles. Notably, all CAFs form a plan for technical execution based on the business objectives and goals established for the transition. They also address organizational and cultural changes and skill development, which is essential in such a major technology change.

Each framework emphasizes the importance of preparing the cloud environment before migrating any systems or workloads. The preparation phase includes the establishment of centralized governance, security, operational excellence, and IAM practices. These elements set the foundation for a secure and manageable cloud environment and serve as enables for the successful onboarding of systems from on-premises to the cloud. Key management strategies are also developed at this stage, considering its role as a centralized security service in the modernized environment.

Once the platform is prepared, system onboardings can begin. For each on-premises system, the most suitable modernization strategy is determined on a case-by-case basis. A system may be rearchitected or refactored to leverage cloud-native capabilities, replatformed (e.g., from on-premises containers to cloud containers), or replaced by a SaaS alternative. In some cases, systems may be rehosted with minimal changes or even retired altogether. However, systems that are merely rehosted may not fully gain the benefits of cloud computing. Ideally, each system is modernized to conform to cloud-native principles, meaning that the workloads can be deployed programmatically in a repeatable and automated manner – eliminating manual processes [29].

The structure of a cloud adoption process is iterative, allowing for continuous review and enhancement of each component as needed. Changes may be needed due to a change in objectives with regard to the cloud platform usage, maturing in cloud proficiency, or new compliance requirements.For large, mature organizations with complex on-premises environments – comprising numerous independent systems at varying stages of their lifecycle – the adoption process can span over several years. During the transitional period, the organization's environment represents a hybrid cloud approach, although the long-term objective is a full-cloud architecture.

Subramanyam [23] explores the added value of the cloud transition not only from the perspective of eliminating the current challenges of the on-premises environment, but also its role as an enabler of future development of financial services. The article highlights the modern approach to *business process re-engineering* (BPR) in the financial sector, where emphasis is placed on customer experience and agile ways of working. The modern approach relies on cloud-native tools for real-time monitoring and analytics, which facilitate accelerated service development cycles. From 1990 to 2020, the share of manual business processes has come down from 90 % to 10 %, replaced by digital processes. As services become increasingly digital, the importance of cybersecurity becomes more prominent. Whereas the expectations of cost savings from transitioning workloads to cloud may not be realistic in the long term [30][31], BPR with cloud-native tools can reduce operational costs by increasing efficiency. At the same time, more efficient processes improve customer experience by decreasing customer waiting times for services such as loan approvals and customer onboarding while also minimizing the risk of human errors during the process [23].

2.3.3 Key management migration process to public cloud

To understand the outcomes associated with the key management migration process, it is critical to define the phases, responsibilities, and most significant architectural differences in old and new approaches. This section outlines the migration process stages and differentiates between two core areas of responsibility: **KMS migration**, which focuses on designing and implementing the new KMS architecture, and **system-level migration**, which involves transferring keys used in different systems from the on-premises KMS to the modernized cloud-based KMS.

Cloud Security Alliance (CSA) provides guidance for arranging key management in the transition process from on-premises to cloud environment [32]. On a high level, the process describes the migration of encryption keys related to a particular use case. However, the process can be used to represent the phases of comprehensive KMS modernization. This guide reflects the perspectives of multiple stakeholders: especially in larger organizations, where key management responsibilities may be distributed across different systems and teams, it is important to define clear roles and responsibilities for the migration.

The key management architecture and strategy are typically designed by dedicated key management professionals within the organization. Their role is to ensure that the organization's key management practices are governed by a clearly defined set of rules that align with industry best practices and compliance requirements.

The cloud transition process is usually orchestrated by a designated team responsible for preparing the cloud platform for system onboarding, as described in the previous section. Since establishing the KMS is part of the preparation phase, close collaboration between key management professionals and cloud platform specialists is required to design and build the modernized KMS.

Finally, system developers and administrators have an important role as the utilizers of the KMS. They are responsible for implementing key management within their respective systems, in accordance with the organization's key management policy, and using the tools provided in the KMS solution.

CSA outlines four stages in the transition process:

1. **Assessment & Planning.** Evaluate current key management practices, identifying gaps, risks and future requirements.

2. **Preparation.** Prepare data and key material for migration, evaluating secure transfer strategies and testing for compatibility to prevent data loss or exposure.

3. **Migration.** Execute the implementation plan validated in phase 2.

4. **Post-migration.** Validate outcomes, establish continuous monitoring, and refine key management practices within the modernized environment.

The first stage, assessment and planning, includes the KMS architecture decision which this thesis aims to support. The architectural solution must be outlined and prepared for use in the platform preparation phase, which takes place prior to system-level onboarding. Therefore, KMS planning, preparation, and deployment occur much earlier than system-specific assessment and planning.

Once completed, the planning takes place on the system side – the responsibility shifts from the key management and cloud platfrom professionals to the system developers and administrators. With the support of organizational key management policy, they make the decision of whether to still retain the control of key management within the organization or to leverage platform-managed keys, and in case of the first option, they must choose an appropriate approach for key management within the offered solution. In case of encryption, CSA identifies data sensitivity, ownership, and regulatory requirements as factors in the risk assessment for the key management decision.

The second and third stages are incorporated in the system-level CAF, and they are executed in accordance with the chosen modernization approach. Based on the decision made in the planning phase, the responsibility of key management may be delegated to the CSP or retained within the organization. This may involve a proof-of-concept testing with the modernized KMS and updated use cases. A critical consideration is the secure transfer of keys and data: will keys be migrated from on-premises to the cloud, or will new keys be generated in the modernized KMS, eliminating the need for direct transfer of the keys?

The final stage, post-migration, involves evaluating the success of the migration from both system and centralized key management perspectives. This includes assessing compliance issues, monitoring KMS usage, and reviewing key management decisions. Technical aspects such as integrations and implementation challenges in different use cases must be evaluated. The assessments should determine whether the chosen KMS architecture is effective, review its security, and detect any possible instances of non-compliant KMS solutions. Additionally, the KMS should be reviewed for opportunities to optimize processes related to its use. System developers and administrators are expected to report their observations about limitations and challenges to the key management and cloud platform teams to support improvements and ensure alignment with key management policy.

3 Challenges in key management modernization

As the platform architecture undergoes modernization, the key management strategy, including architecture, must also be updated. Public cloud platforms offer robust key management tools, but third-party KMS providers present alternatives for the platform-native offerings. To examine the most suitable solution for key management modernization, the risks and challenges of the organizational and technical environment must be considered.

The goal of key management modernization is to implement best practices in a sustainable and efficient way, unifying key management policies across the organization and gaining visibility and control over keys' compliance to the policy, avoiding typical pitfalls often encountered in a complex on-premises environment.

This chapter discusses the unique characteristics and requirements for key management in the financial sector. Then, this chapter gives an overview of challenges with key management to which on-premises environments are prone to. With this information, this chapter describes the risks and challenges related to key management modernization during and after the cloud transition, divided into two parts: what are the possible pitfalls for the architecture design in the target environment, and what risks might be encountered during the transition phase related to key management modernization.

3.1 The role of key management in the financial sector

To place key management and its modernization in context, it is essential to examine its role within the financial sector. The sector plays a critical role in society, facilitating payments, savings, loans, investments, and insurance in ways that promote economic stability and growth. In certain European countries, including Finland, strong citizen authentication is implemented using banking credentials [33], further emphasizing the societal importance of the sector. The operational integrity of financial services impacts all individuals in the society, as well as corporate customers and even governments.

Consequently, disruptions related to service availability, failure to demonstrate the trustworthiness of the authenticity of its services, or incidents involving data breaches can have long-lasting consequences in an increasingly globalized society and may prove to be catastrophic for a financial organization in a matter of days. In the modern era, financial organizations must function as an integral part of the infrastructure but also remain competitive in the evolving market using emerging technology. Given these factors, the security of services, that is, confidentiality, integrity, and availability, must remain uncompromised.

As a result of its role in society, the financial sector is highly regulated. Although most sector-specific regulations do not directly address technical implementations, regulators are also increasingly motivated to set expectations for the sector's resiliency and security levels. Within the European Union, the European Banking Authority (EBA) supervises compliance with regulations applicable to financial organizations.

The most notable technical requirements found in regulation come from the Digital Operational Resilience Act (DORA) [34], which entered into force in 2025 in the

European Union, aiming to harmonize operational standards for critical services of the financial sector. DORA is one of the few regulations that specifically establishes requirements for key management. Furthermore, although not exclusively targeting the financial sector, the General Data Protection Regulation (GDPR) imposes obligations on the management of customers' personal data, which necessitates the use of cryptograhpy as a part of its implementation, hence also underscoring the essential role of effective key management.

In countries where governments have outsourced strong citizen authentication to banks, banks are responsible for acting as identity service providers when citizens authenticate to services using their bank credentials [33]. Strong authentication implementation framework is defined in the EU regulation on Electronic Identification and Trust Services for Electronic Transactions for Electronic Transactions in the Internal Market (eIDAS) [35]. In the theme of strong authentication, the EU Second Payment Services Directive (PSD2) sets requirements for strong customer authentication in online payments [35]. PSD2 and eIDAS requirements from the point of strong authentication align with each other. The regulations outline the use of different types of certificates; however, the technical requirements for keys or algorithms do not differ from industry standards [35].

In addition to formal regulatory frameworks, self-regulation has produced standards to ensure secure operations. Payment Card Industry (PCI), formed by industry market leaders, has developed particularly detailed technical standards to support the security of the payment infrastructure. Requirements that are relevant to key management mandate the use of HSMs that are validated with certain FIPS levels, described in Chapter 2.1.3. PCI standards are a common concern in financial organizations, and therefore, further discussed in Chapter 4.3.1.

As described in Chapter 2.1.1, key management plays an important role in the security of IT systems. The modern financial domain is dependent on technology, and failures in these systems can have significant consequences, which act as motivators to commit to securing the systems, using key management, among other tools and practices, based on enterprise risk management. The technical requirements for financial systems address resilience against system malfunctions, configuration mistakes, and cyberattacks.

Regulatory, industry, and risk-based requirements create a unique environment for key management within financial organizations. Although there are many rules to be followed, direct instructions are lacking, and each organization must come up with their own interpretation of the role of key management in fulfilling all needs in the IT systems for financial services. Regulatory requirements and industry standards affecting key management are discussed in more detail in Chapter 4.3.1.

3.2 On-premises key management challenges

To understand the requirements and limitations of key management solutions, it is beneficial to assess current challenges faced in on-premises environments. Past experience and existing issues with key management provide a foundation for setting requirements for the modernized approach. This section presents the primary reasons for key management modernization and illustrates why a cloud transition necessitates reengineering rather than directly replicating legacy practices in the cloud environment.

In traditional on-premises environments, key management capabilities are typically limited. According to Somasundaram [4], the most notable issues are security risks arising from manual handling and unsecured storage, scalability issues from diverse systems lacking a unified key management approach, insufficient automation, and difficulties in meeting compliance requirements. In environments composed of multiple generations of architecture, built on different technology stacks, key management approaches tend to vary between systems. Some systems may implement automation and store keys on secure hardware, while others are burdened with manual key management practices and unsecured secret keys. In the latter case, the need for modernization is especially urgent.

A manual approach to key management becomes a bottleneck as the rotation frequency and number of keys increase. This approach not only demands specialized expertise and a thorough understanding of key management from those managing the keys but also introduces an increased risk of incidents caused by human error, negligence, or insufficient resources allocated to secure key management [4]. In more modern systems, automation can be used to address the limitations of manual processes.

The diverse and fragmented key management approaches complicate governance. A centralized key management system can enhance visibility, auditability, and enforce governance by ensuring that the systems adhere to the organizational key management policy. However, in siloed and heterogeneous environments, a centralized key management approach may not be successful, as integrations with different systems are found difficult [22]. A cloud platform provides monitoring and automation capabilities to govern key management throughout the environment.

The role of the modernized key management approach in the on-premises environment requires evaluation. Specifically, the organization must decide whether to start using the modernized KMS within on-premises systems while waiting for the system's cloud transition to complete, or to migrate to the modernized KMS during the system's cloud adoption process. The latter option presents a logical point of such changes, as the system is undergoing broader reengineering. However, regulation, vulnerabilities, and cyber threats are imposing increasingly stricter security requirements for on-premises systems as well. Consequently, key management in the old environment must also evolve and improve continuously. Therefore, the feasibility of integrating on-premises systems with the modernized key management architecture must be assessed on a case-by-case basis.

3.3 Risks in key management in the cloud

Existing research reveals a gap in understanding the risks associated with key management in environments that rely on a single public CSP. The topic presents several challenges. Importantly, the adoption of single-CSP environments remains relatively uncommon in the financial sector. According to the CSA 2024 survey [25], also mentioned in Chapter 2.3.1, the majority of financial organizations utilize a hybrid cloud model that incorporates both on-premises and cloud environments, with approximately half of them distributing cloud workloads between two or more CSPs. This section examines the risks in adopting a single-CSP architecture and explores their implications on key management. Furthermore, this section briefly touches upon the cryptography techniques enabled by a hybrid cloud approach.

The CSA survey [25] identified regulation and compliance as the primary blocker for cloud adoption within financial services. The findings indicate an increase in these concerns since 2020, reflecting growing concerns about the regulatory landscape and finding compliant solutions for cloud adoption with regulated workloads. Zeghmouli [31] further elaborates this issue, noting that the industry lacks clear and consistent regulations concerning data security, sovereignty, residency, and the use of cloud services in general. Zeghmouli highlights that the requirements are constantly changing due to political developments and advancements in technology. Notably, the article reports that changes in legislation or compliance act as the primary reason for 19 percent of cloud exits done by organizations in the financial sector.

In addition to – and partly as a result of – regulatory challenges, fully migrating services to a single cloud platform raises concerns regarding resiliency. CSPs offer extensive backup and disaster recovery capabilities, along with service redundancy, facilitated by the extensive network of datacenters across multiple regions. The infrastructure allows for high availability and enables organizations to change the physical location of their workloads with a configuration change. This level of physical and operational flexibility cannot be provided by traditional on-premises environments. However, reliance on a single CSP introduces a potential single point of failure. Ryan et al. [36] present a notable increase in incidents on CSP platforms between 2020 and 2023. An example of a CSP-caused incident was a configuration mistake in Google's cloud platform, which led to permanent deletion of an entire environment belonging to UniSuper, a major financial institution [36]. This incident highlights the risks associated with CSP dependency.

Due to the limited experience and research on single-CSP architectures and key management, there are no definitive answers as to how key management should or can be arranged in a heavily regulated environment characterized by constantly evolving requirements and a geopolitical landscape that continues to introduce new uncertainties. In recent years, there has been a growing emphasis on cloud exit strategies [30]. The trend does not necessarily indicate a desire to abandon cloud environments altogether but rather reflects a strategic shift to embrace hybrid architectures or preparing to mitigate emerging risks. These considerations are particularly relevant for key management and other centralized services, which may need to be migrated to other environments within a limited time frame.

Despite strong evidence indicating that financial organizations favor the hybrid approach, the proportion of cloud usage even for critical and regulated workloads has increased in all aspects in 2023 compared to 2020 [25]. This may reflect ongoing cloud transitions, with many organizations aiming to decommission on-premises datacenters once the transition is completed. With maturing solutions for sovereign cloud technologies, the future of regulatory compliance within a full-cloud architecture increasingly appears as an opportunity rather than a threat. Nevertheless, it remains uncertain whether this is a risk organizations are willing to accept – and whether the decision can be determined by cryptographic key management considerations.

3.4 Risks in key management modernization

The approach for key management modernization is not evident. Although cloud platforms offer tools for key management, their suitability for a complex environment with a variety of technical requirements must be assessed. Furthermore, the hybrid cloud phase during the transition, combined with constantly evolving compliance requirements and deadlines, adds a layer of complexity to the key management strategy. The on-premises environment must continue to meet compliance obligations regardless of the upcoming reengineering driven by the cloud transition, which may still be years from completion. This raises a critical question: How should key management of the two environments be supported during the transition years, and how can the chosen solution be future-proofed for the cloud-native target architecture?

Unifying key management tools and processes is a comprehensive project, and platform modernization through a cloud transition offers excellent circumstances for such an undertaking. However, inadequate planning and inaccurate risk assessments can result in a key management solution that does not align with the organization's requirements or the modernized IT environment. Modifying the key management architecture during or after the cloud transition not only incurs additional work but may also lead to distributed key management solutions, which have been seen as a major challenge in on-premises environments.

The decision between centralizing all services, including on a single platform to realize operational synergies, or using third-party KMS solutions while adding complexity in order to enhance resilience and security is not straightforward. In practice, the assessment is much more nuanced. While there are multiple different solutions to organizing key management in the cloud, each with their own strengths and weaknesses, it is challenging to measure and compare different aspects and their impacts on the overall suitability of the KMS architecture.

When designing the key management solution, architectural decisions must incorporate the long-term strategic goals. For instance, is the organization planning to remain with a single CSP, or retain the possibility to switch service provider? It is also essential to consider exit scenarios for the cloud architecture and form a plan from the KMS side to support different architectures, such as hybrid cloud. A central consideration in this context is whether it is desired to use platform-specific tools or remain platform-agnostic with the choice of a KMS solution.

4 Criteria definition for selecting KMS

Key management best practices, cloud migration risks, and financial sector requirements collectively establish high expectations for the modernized key management architecture. These expectations are formed by the identified challenges, improvement opportunities, and the cloud-native target architecture. As with all IT development, leveraging cloud-native components is preferred to maximise the benefits of platform synergies. The goal of this chapter is to define a comprehensive set of evaluation criteria that assess the suitability of the cloud platform's native capabilities for use in financial organizations and to facilitate comparison with alternative architectural solutions.

When assessing the requirements for the key management architecture, it is essential to consider the entire transition process, described in section 2.3.3. The lengthy transition process to the cloud environment creates a hybrid cloud infrastructure, which requires retaining the security of this type of environment as well.

CSA [8] suggests evaluating the desired functional, operational, and business outcomes for the key management solution. Functional outcomes describe the expectations for the technical capabilities of the solution, which must align with the best practices of key management. Operational outcomes assess the manageability of the solution and how it complements the roles and responsibilities of key management within the organization. Business outcomes include the organization's risk tolerance and its adherence to regulatory and other legally binding obligations. Using these categories, this chapter further explores the key management requirements and considerations.

4.1 Functional requirements

This section discusses the requirements for supported features in the key management solution, focusing on the identified risks and challenges in the on-premises environment and the transition to the cloud platform. The assessment begins with an overview of the features based on key management best practices. Next, requirements related to the transition process are reviewed. Finally, this section explores the role of secrets management and its relationship with key management, including considerations for its implementation in the scope of this thesis.

4.1.1 Key management best practices

Section 2.1 outlines the purpose and lifecycle phases of key management, which define the expectations for the supported functionality of any chosen KMS approach. Current challenges in adhering to best practices, identified in section 3.2, arise from decentralized key management with insufficient automation capabilities. These issues contribute to compliance and security risks by complicating governance and monitoring processes and creating bottlenecks due to the reliance on manual processes. Addressing these challenges should be a core objective of the modernized key management solution. An effective KMS should offer a unified approach that is applicable in all applications regardless of their sensitivity level. This involves implementing stronger controls, such as quorum policies, referring to an activity that requires approval from m out of n predefined approvers, and ensuring hardware protection where necessary. At the same time, the solution should accommodate less stringent controls for non-sensitive or non-production use cases, balancing security with usability and efficiency.

CSA 2024 survey on cyber resilience in the financial industry shows that the most widely used cybersecurity frameworks are the National Institute of Standards and Technology Cybersecurity Framework (NIST CSF) and the International Organization for Standardization 27001 (ISO 27001) [37], which both mandate the use of cryptography as a protective measure to ensure security. While these strategy-level frameworks do not provide implementation guidelines for complying with the controls, CSA provides a rather useful framework in the cloud computing context. The Cloud Controls Matrix (CCM) developed by CSA offers comprehensive guidance on key management, explicitly addressing the shared responsibility model [38]. The CCM requirements on key management align closely with NIST key management best practices and provide clarity on which responsibilities are on the CSP side and which remain the customer's obligation.

The CSA CCM emphasizes the importance of clearly defined roles and responsibilities, as well as the separation of duties in key management [38]. The KMS should provide comprehensive access control capabilities that support appropriate access rights for different stakeholders. These controls should account for separation of duties and quorum approvals with the most sensitive keys, as mentioned earlier. To minimize the attack surface and human error in the access rights configuration, the KMS should facilitate seamless access rights management integrated with the platform's centralized IAM. Crucially, cryptographic keys must be accessible only to authorized entities.

To conclude, the evaluation criteria from key management best practices can be summarized into three points: the KMS approaches' functionality to support key operations, both cryptographic and management-related, throughout the key's lifecycle; the unified policy definitions and governance in a centralized manner; and granular access control with separation of duties in the key management context.

4.1.2 Key management transition challenges

The duration of the cloud transition phase must be considered in the selection of the approach. Given that platform preparation and system onboardings span over several years, on-premises key management must continue to be maintained and developed to address emerging security threats and evolving compliance requirements. Consequently, it is desirable for the modernized KMS to support on-premises integrations and expand the unified management and policies across the environments, covering also systems which are yet to be migrated to the cloud.

The duration of the deployment of the modernized KMS approach also affects the duration of the cloud transition as a whole. Especially in case of implementing highly customized KMSs or complex integrations with systems, completing them takes more time and acts as a blocker for executing system-level onboardings. The effort and extensiveness of the key management solution's modernization and integration act as a criterion for the KMS.

Another important aspect of the transition phase is the onboarding of systems into using the modernized KMS solution. In some cases, it is feasible not to rotate keys that are in use when migrating them to the modernized solution. Therefore, it is important to support key import. Consequently, such keys may also need to be exported in possible re-migration scenarios, requiring also export functionality in the modernized KMS. On the other hand, FIPS requirements and organizational policies may deny key export, and thus disabling such functionality should also be possible.

Upon assessing the transition requirements, the challenges are adequately addressed by feature requirements of NIST best practices. Furthermore, the delays caused by solution complexity are further discussed in operational considerations. Finally, business considerations discuss the KMS flexibility to support on-premises integrations and other platform architectures. Hence, the transition challenges are found to be minor and as such do not generate additional criteria.

4.1.3 Secrets management

Although not strictly a requirement from the key management perspective, modern KMSs include functionality for managing other types of secrets, also extending to certificates. In the cloud, secrets management may be integrated as part of the KMS [9], implying that the same infrastructure is also used for this use case. Key management and secrets management share several core principles, such as access control, lifecycle management, and auditability. Moreover, an organization needs a policy and infrastructure for secrets management regardless of how key management is implemented.

Despite the strong relationship and similarities in key management and secrets management, it must be considered during architecture design whether keys and other secrets should share common tools, policies, and strategy, or arrange the management of secrets, such as application credentials, using separately defined practices and tools. Therefore, secrets management cannot be treated as an isolated concern from the design of a KMS solution even if the result of this evaluation is to separate their implementation into different systems.

4.1.4 Functional criteria summary

To summarize the findings related to functional factors that affect the key management decision, the solution's technical capabilities must be able to support best practices and offer a solution to the most prominent challenges found in complex on-premises architectures. Therefore, the criteria aim to highlight the differences in solutions in terms of their implemented features and ability to unify governance related to key management. Furthermore, the criteria raise roles and responsibilities, access management, and the relationship with secrets management for discussion. The derived functional criteria are presented in Table 2.

Criterion	Description
FC1	Provides essential features that align with key management best practices and support the core operations in each phase of the key lifecycle.
FC2	Enforces centralized key management policy through technical controls, including restrictions on the use of weak algorithms, inadequate key lengths and validity periods.
FC3	Supports granular access control with respect to separation of duties in key management.
FC4	Enables automated key rotation.
FC5	Includes secrets management capabilities.

Table 2: Functional criteria (FC) summarized.

4.2 Operational requirements

This section focuses on the operational requirements, encompassing aspects related to the delivery, maintenance, governance, and use of the KMS. The goal of these requirements is to assess the suitability of the processes associated with the KMS solution in the organizational context. The discussion addresses both the end-user perspective and the effort required from key management professionals responsible for the system's administration and monitoring. Finally, the section examines how the selected KMS approach influences the organization's ability to implement crypto-agility.

4.2.1 User experience

In on-premises environments, decentralized key management – where keys are distributed into technology or use case-specific key management solutions – is typical. However, as outlined in functional criteria, the goal is to align with best practices by implementing a unified key management approach. In the transition, it is critical to prevent forming the re-emergence of fragmented system-specific key management strategies. This can be enforced using an organization-wide policy, but the policy alone may not be sufficient to eliminate instances of *shadow KMS*, a type of shadow IT where key management does not comply with the organization's policy.

Klotz et al. [39] collected drivers, consequences, and governance practices related to shadow IT management. Their findings indicate that the developers, administrators, or architects of a specific system may intentionally act against organization-wide policy to gain productivity, agility, and innovation through their own decentralized solution, or they might lack awareness of policies and best practices, leading to non-compliance.

Employee attitudes and peer behavior significantly influence the emergence of intentional shadow IT [39], including the use of shadow KMSs. According to the paper, most prominent reasons to opt for non-compliant systems include a mismatch between system functionality and requirements, slow processes related to system onboarding and use, loss of power related to system administration, and lack of trust for system administrator competence [39]. These challenges are particularly pronounced during a period of organizational change, where roles and responsibilities are redefined and the environment architecture is experiencing a comprehensive transformation. Employees may be tempted to retain previous key management practices which are familiar from the on-premises environment.

Unintentional policy violations, on the other hand, are the result of insufficient monitoring, inadequate technical controls, and lack of training or awareness. Addressing these issues requires automated monitoring and policy enforcement, which can be supported by CSPM. The KMS migration and key management processes must be clarified, also considering possible skill gaps in cloud technologies and complex cloud adoption processes in a larger picture.

While policy and process clarity is independent of the KMS architecture itself, the usability and suitability of the selected architecture play a critical role in adoption of the unified solution. The convenience and integration flexibility must be considered in the design to support a variety of different requirements, minimizing excessive work for using the modernized solution, which might lead to seeking unofficial alternatives. When considering the differences in the modernization approaches, the trade-offs must be carefully considered: what are the risks of choosing usability over added security?

4.2.2 Delivery and administration

The manual effort associated with the KMS is an important factor. Although the available assets, such as time and experts, the organization is willing to assign into the delivery and maintenance of the KMS may vary, it must be noted that maintaining heavy KMS systems is not the core business of a financial organization. While security is pronounced in these organizations, more manual work does not necessarily mean a more secure solution. On the contrary, manual processes in key management are seen as one of the greatest challenges in on-premises key management [4]. This section addresses the delivery of the KMS solution and, once operational, the ongoing efforts required to ensure adherence to best practices and correct usage of the system – considering the administrative tasks regarding the KMS.

During the cloud transition, the enablement of key management capabilities should not become a major source of delay for the transition process. If the delivery of the key management solution is expected to last longer, the delay should be justified by cases where alternative solutions are either incompatible with KMS requirements or the proposed KMS provides significantly superior solution. Possible delays must be anticipated early in the planning process to avoid unexpected waiting time in the cloud transition, and initiate implementation as early as possible. In some cases, accelerating KMS delivery may require additional workforce, which adds costs.

Once operational, manual customer-side responsibilities for managing the centralized KMS should be minimized. Analytics, alerts, and automatic policy enforcement mechanisms should be used to eliminate the need for manual governance and monitoring where possible. Such automation not only enhances security and compliance but also promotes the principles of cloud nativity, leveraging the scalability, efficiency, and agility offered by cloud services [29]. The amount of effort in the maintenance is also influenced by the deployment model of the KMS. It is important to determine to which extent the KMS technology stack should be managed by the organization, and whether the deployment model should vary depending on use cases.

4.2.3 Crypto-agility

Crypto-agility has gained prominence as a result of the emerging threat posed by quantum computing and its potential to break cryptographic techniques that are currently considered strong. While some consider quantum computing to be a distant concern that does not necessitate immediate action, crypto-agility offers many benefits for tackling current threats and challenges related to cryptography and key management specifically. Crypto-agile practices include the ability to seamlessly change the key algorithms and key lengths used in the systems, as well as highly automated key management practices and key inventory [7]. Such measures improve the response time in incidents related to cryptographic keys.

Crypto-agility is heavily dependent on the capabilities of systems integrated with the KMS. Furthermore, these systems may be part of a broader network of interconnected systems with dependencies between each other, all of which must adapt to evolving cryptographic requirements to support mutual cryptographic suites. The web of dependencies forms the true challenge of crypto-agility, and hence, the KMS architecture may not be the main contributor in crypto-agility but rather the implementation within the connected systems. However, the key management system must be able to support a variety of key lengths and emerging algorithms as technology progresses.

4.2.4 Operational criteria summary

This section evaluated the efforts associated with key management deployment, maintenance, use, and auditing. To support users with adopting the centralized KMS, user experience should be considered, while still accommodating a variety of use cases. Compliance monitoring and governance should require minimal manual effort and leverage the benefits of cloud nativity. Delivery and maintenance efforts should be minimal, leveraging shared responsibility where such tasks are outsourced to the service provider. Finally, the system should be updated to support new algorithms and key lengths as PQC requirements advance. These topics form the basis for the evaluation criteria listed in Table 3.

Criterion	Description
OC1	Demonstrates high usability and seamless integrability with other systems to decrease friction in KMS adoption.
OC2	Supports a range of use cases with varying levels of sensitivity, latency, capacity, and operational management.
OC3	Facilitates governance and policy compliance monitoring with minimal reliance on manual processes.
OC4	Deployment and maintenance responsibilities are feasible with allocated resources.
OC5	Accommodates updates to cryptographic algorithms and key lengts in response to evolving technical standards.

Table 3: Operational criteria (OC) summarized.

4.3 Business requirements

Business considerations reflect regulatory compliance and related risks, which have been widely recognized as the most challenging aspects of key management in the context of public cloud adoption. The considerations are followed by an examination of the KMS's role in business continuity, emphasizing the importance of system resiliency and disaster recovery. Finally, the discussion addresses the cost of the KMS solution.

4.3.1 Regulation

Regulatory requirements can be viewed from two distinct perspectives. First, there are explicit legal obligations that financial organizations must comply with. Second, there are regulatory risks, referring to topics where current regulations do not require action in the present but where future regulatory changes may necessitate considerable changes to key management. To begin with regulatory and other currently legally binding requirements, this section outlines the main considerations related to PCI standards, DORA, GDPR, and Schrems II.

In the context of legally binding technical requirements, the PCI standards introduced in Chapter 3.1 are not mandated by any regulation within the EU. However, contracts made with payment card providers, such as Visa and MasterCard, mandate PCI compliance. While PCI DSS (Payment Card Industry Digital Signature Standard) requires the implementation of strong encryption, this can be achieved by following NIST best practices described in Chapter 2.1. Although the use of an HSM facilitates secure storage, the standard does not require any specific FIPS validation or level [40]. In contrast, standards such as PCI PIN Security, governing the processing of payment card PINs, require the use of HSMs certified with FIPS 140-2 or FIPS 140-3 level 3 or higher [41]. While PCI DSS requirements apply to a larger audience, more specialized PCI standards, such as the PCI PIN Security mentioned earlier, affect a much smaller subset of financial organizations. Similar contractual requirements for FIPS 140-2/140-3 may arise from other contracts between financial organizations and their collaborative partners, vendors, or customers. It is important to note that PCI standards only cover use cases related to established contracts; however, organizations with such use cases must be able to comply with the requirements.

Briefly touched upon in section 3.1, DORA is an EU regulation that requires financial entities to apply sufficient measures to ensure the security and resilience of their IT environment [34]. It aims to harmonize requirements for ICT (information and communication technology) management in the financial sector and therefore goes to details in many areas of security and risk management. Its regulatory technical standards (RTS) on ICT risk management framework outline the controls for cryptography and key management [42]. The standards, again, align with NIST key management best practices with respect to key lifecycle management, policy definition, access management, logging, and monitoring. The RTS emphasize key management resiliency in the sense that methods for replacing lost, compromized, or damaged keys must be defined, and the key management policy enforcement must be monitored and any exceptions to the policy require an explanation and a description of countermeasures for possible vulnerabilities caused by the non-compliant key or process. Furthermore, there is also a requirement to define approaches for changing technology, such as algorithms, to remain resilient against evolving attacks, such as quantum computing.

While most of the RTS are already outlined in the functional criteria, the management of policy exceptions pointed out by the RTS should be considered in the KMS modernization. The key management solution should provide the possibility of documenting policy exceptions so that tracking them would not be all over the place in separate places where documentation could be stored.

Whereas DORA is intended for harmonizing financial organizations' resilience requirements in the EU, GDPR and Schrems II pursue the control of EU citizens' personal data, not only in the case of the financial sector but concerning all organizations managing the data of EU citizens [43]. The EU considers personal data management a human right, meaning individuals must retain control over their data, and therefore organizations processing and storing customer data must be able to control data access even when it is transferred to third parties. This type of control is especially important for financial information. The regulations consider data privacy, data sovereignty, data residency, and data ownership.

In the context of cryptographic key management, access to encrypted data depends on access to the encryption keys. According to Article 32 of GDPR, organizations must be able to assess the risks to "rights and freedoms of natural persons" with required severity and implement security controls accordingly [44]. In general, platform-managed keys meet the cryptography-related requirements arising from EU regulations. With platform-managed keys, the customer's only obligation is to enable the functionality; the CSP is responsible for all key management. While these keys do not protect against CSP-side access, as the CSP retains control over the key material, the cryptographic measures prevent unauthorized access through attack vectors targeting the cloud infrastructure.

As discussed earlier, regulatory challenges remain one of the primary obstacles to cloud adoption in financial organizations [25][31], creating a complex and constantly evolving landscape for financial organizations to operate in. The second half of this section discusses data privacy with a particular focus on cross-border data transfers and the role of key management in managing the associated risks.

The regulation on personal data privacy has been subject to frequent changes for the past decade. Currently, GDPR and Schrems II limit data transfers outside EU borders. Although the major CSPs have established datacenters across the globe, including an increasing number of datacenters within the EU, the companies are US based, making them subject to US law including cooperation with the National Security Agency (NSA). Therefore, the CSPs themselves cannot assure that the data in their European datacenters will not be accessed from the United States; upon access, the data is transferred to a non-EU country, violating the regulations. However, agreements have been developed to promote cooperation between the US and the EU markets.

The current enabler for cross-border data transfer between EU member states and the United States is the EU-US Data Privacy Framework (DPF) [45]. This framework entered into force in 2023 [45]. To be certified with the framework, US organizations must comply with privacy requirements corresponding to EU legislation [46]. The predecessors of the DPF, namely, the Safe Harbor framework (2000-2016) and the EU-US Privacy Shield (2016-2020), have been invalidated as found inefficient [43], and therefore changes or successors to the DPF can be expected. Currently, the major CSPs are certified with the DPF, and but any future changes will again have an impact on the possibility to use major CSPs services in EU organizations.

International regulatory dynamics are continuously evolving, and the regulations on cross-border data transfers reflect the changes. Data privacy regulation has a different emphasis in each nation. While the EU emphasizes the rights of citizens and strict privacy protection, the US prioritizes competitiveness with fewer constraints on cross-border data transfer [43]. China, on the other hand, enforces restrictions on data transfer for national security and strategic reasons [43]. Although the DPF currently facilitates data flows between organizations based in the EU and the US, similar frameworks have not been established between the EU and China. This makes legal compliance for data transfers between the EU and China significantly more challenging [43] and showcases the situation which might follow from future plot twists with relations between the US and EU and their views on data privacy.

Under these shifting regulatory conditions, financial organizations are increasingly concerned about where data is stored and who has access to it. Anticipating tighter regulations and responding to internal governance requirements, organizations question whether a single-CSP strategy will support the needs for data management. Regulators may become increasingly concerned with data sensitivity, such as in the case of financial data, reflecting recent regulatory advancements with financial organizations' resilience through DORA.

CSPs recognize the risk of changes in regulation and have begun to develop strategies to support EU customers with strict cross-border data transfer requirements. The concept of *sovereign cloud* supports retaining access to data within the EU borders and improving the integrability and support of external key management solutions, which allows customers to take technical control of data access using encryption with externally managed keys [47]. External key management is widely recognized as the most important solution to prevent CSP-side access to customer data [48], along with confidential computing for protecting data in use [25]. In response to regulatory challenges and changes, organizations are advised to encrypt or tokenize data stored in the cloud and use *key segregation*, i.e, to keep keys separate from the cloud platform [48][49].

However, as previously discussed, key segregation is effective only when data never enters CSP premises in unencrypted form. For this reason, organizations develop cloud exit strategies [30], and a single-CSP approach may evolve into a hybrid architecture. It must be noted that hybrid architecture is still the most common approach in financial organizations [25]. Such a shift may occur even during ongoing cloud migration programs. Therefore, the suitability of the key management solution's suitability for the hybrid cloud approach must be assessed not only as a temporary measure during the transition but as a permanent solution in seamless cloud exit scenarios.

4.3.2 Business continuity

Key management is a critical component of business continuity. Beyond the resiliency requirements addressing internal key management practices and policies defined in DORA RTS, the key management solution itself must be resilient. Given its integrations throughout the environment, the key management system must ensure high availability and meet service level agreement (SLA) requirements of connected systems. Within the context of critical services in the financial sector, it is crucial to avoid unexpected service interruptions caused by KMS disruptions. Accordingly, comprehensive disaster recovery (DR) strategies must be developed, documented and tested.

The disaster recovery strategy for the KMS depends significantly on the KMS architecture. The division of responsibilities between the CSP and the customer dictates how much the customer may affect the DR process and rehearsals. Different vendors offer varying levels of competence and technical capability to support rapid recovery in a disaster situation. Additional considerations for KMS resilience include the system's geographical distribution, the ability to relocate service across regions, and the reliability of KMS backups.

Moreover, the KMS architecture can influence disaster recovery capabilities across other systems. In event of a CSP disruption, off-site backups can be used if keys related to the backups are stored in an external KMS rather than a platform-native one. Therefore, key segregation also enhances the broader resilience of critical systems.

4.3.3 Cost factors

As a final consideration for the KMS criteria, cost remains a fundamental factor in architectural design. In cloud-based environments, billing models are typically based on consumption, reflecting actual usage of the KMS with no costs up-front. However, alternative pricing strategies exist; for example, third-party software-based solutions may operate on licensing models with fixed or tier-based models. Furthermore, the

KMS architecture may include additional infrastructure components, which must be accounted for in the total cost of the KMS.

Aside from the KMS infrastructure expenses, personnel costs can significantly impact the overall cost of the solution. Particularly in a highly customized or complex KMS solution, the design, implementation, and maintenance of the system may necessitate a dedicated team of specialists. These requirements can increase the operational costs throughout the lifespan of the system.

While security is recognized as a priority within financial services, making it both possible and necessary to allocate sufficient resources to key management, it is not always feasible to maintain control over a large portion of the technology stack either from practical or financial perspectives. When possible, outsourcing the maintenance and development of the KMS itself should be considered, as it decreases operational costs and may result in overall smaller costs of the KMS.

4.3.4 Business criteria summary

Regulation poses direct requirements on cryptography and key management. However, it can be observed that the regulation primarily demands adherence to best practices already defined in the functional criteria. DORA emphasizes policy management, and legally binding contracts involved in operating in the financial environment may require compliance with PCI standards, primarily the use of cryptographic modules of certain FIPS levels.

A greater risk for key management architecture comes from evolving regulatory requirements and uncertainties arising from changes in geopolitics. Financial institutions are aware that regulations are stringent and may require exit strategies for the cloud architecture, which must be considered in KMS selection. Finally, business continuity considerations and alignment with service-level agreements (SLAs) of dependent systems must be taken into account. These factors were used to form the list for business criteria, presented in Table 4.

Criterion	Description
BC1	Supports the use of FIPS-validated hardware where required by regulatory, contractual or internal obligations.
BC2	Enables handling and tracking policy exceptions centrally.
BC3	Offers architectural flexibility to support changes between CSPs or to hybrid cloud models.
BC4	Meets resiliency, high-availability, and SLA requirements for systems that depend on the KMS.
BC5	Presents a cost structure that is in proportion with the provided benefits.

Table 4: Business criteria (BC) summarized.

5 Potential KMS solutions

This chapter examines possible solutions for key management architecture in the cloud. First, the options for cloud KMS *patterns*, summarized by CSA [8], are examined. Based on these options, this chapter presents three approaches assembled from the components presented, focusing on architectural differences and their abilities to meet the criteria defined in Chapter 4. The solutions consider both the transition phase as well as the target architecture.

Although a hybrid cloud architecture during the transition phase would allow partial on-premises key management solutions, the scope of this thesis excludes on-premises infrastructure as a part of the modernized platform. Hence, this chapter does not discuss on-premises HSM appliances or other on-premises KMS components in the KMS patterns or approach suggestions. It is assumed that the in the modernization process of systems that are on-premises HSM clients will refactor their system to use the modernized KMS approach.

Although the role of HSM is important, especially from key storage and regulation viewpoints, this section discusses key management as a whole using the term *key management system* (KMS). At this abstraction level, it is not specified whether the KMS is HSM-based or not. None of the suggested cloud-based approaches rule out or require the use of HSM. The need for different types of HSMs is further assessed in Chapter 6.

5.1 CSA KMS patterns

CSA describes the most common approaches to the KMS architecture with four *KMS patterns*. The abstraction level of the patterns does not make statements on the cloud deployment model or service model used but instead generalizes the options for all applicable models. In some cases, the CSA paper [8] discusses the use of on-premises components. However, they can be presented as any external KMSs, including cloud-based solutions. This thesis considers only cloud-based options.

Figure 4 illustrates each pattern in the scenario where the cloud transition has been completed and the on-premises assets no longer exist. The following subsections go through the main characteristics of each pattern. The patterns are intended to simplify the KMS model for specific use cases and are not generic solutions for all environments with varying key management needs. Therefore, they are used selectively as a reference for the actual architecture designs.

5.1.1 Cloud Native KMS

The first pattern describes an approach where the same CSP provides all cloud services, including the KMS. The key management architecture may consist of multiple different components: software- or hardware-backed KMS instances which can be accessed via graphical user interface (GUI), application programming interfaces (APIs), or command-line interfaces (CLIs), or different HSM instances, such as managed or dedicated HSM.

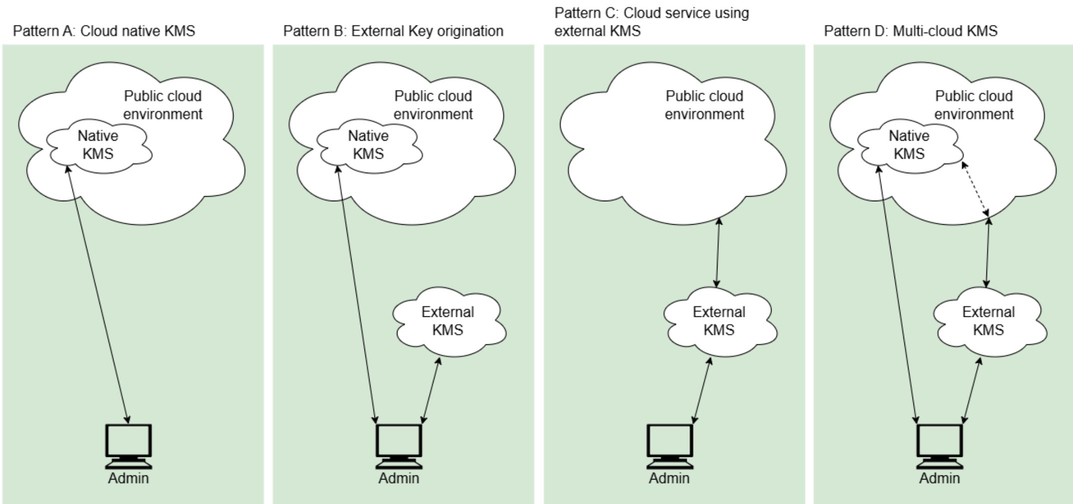

Figure 4: CSA patterns [7].

While residing on the same CSP's premises with other components, the cloud-native KMS offers highest performance and scalability, lowest latency, as well as the most seamless compatibility with other platform products in comparison to other KMS patterns. Furthermore, the service-level agreement (SLA) enforcement is not made more complex with third-party products. Of the four patterns, the cloud-native KMS has the least time-consuming deployment process. [8]

The most significant drawback in the cloud-native pattern is the lack of responsibility and, therefore, also control on the customer side [8]. While the solution is straightforward to use and manage from the customer perspective, the CSP dictates and executes maintenance, development, backup processes and disaster recovery rehearsals, as well as their share of configuration and monitoring of the system, with the customer having little to no visibility to what's happening *under the hood* and little control over the functionality of the service. Additionally, cloud-native KMSs typically offer limited possibilities for separation of duties and organization for key ceremonies [8], e.g., ones which require designated key officers for key export.

As discussed earlier, concentrating all components of the environment into the same cloud platform eliminates the possibility to control CSP access to data within the platform. This KMS pattern is dependent on one CSP; hence, the change of the architecture into, e.g., multi-cloud or changing the cloud platform altogether is not very convenient compared to other patterns.

Regardless of the drawbacks, CSA suggests using the cloud-native KMS pattern unless the customer has a clear reason to use other models [8]. According to CSA, the platform-native synergies, low cost, and implementation speed as well as performance outperform other patterns. Trust in CSP resiliency lowers the perceived risk of issues caused by the downsides of the pattern.

5.1.2 External Key Origination

External Key Origination is a pattern for very specific use cases where the cloud-native KMS does not offer the required functionality in the key generation phase, such as the separation of duties or key ceremonies mentioned in the above section. The procedure is also referred to as *bring your own key* (BYOK), enabling customers to import externally generated key material to the cloud-native KMS for use in the cloud environment. [8]

Outside the extended functionality, the pattern does not provide significant advantages, but having an external system for key origination may bring additional costs and complexity. Therefore, this pattern is recommended only when there are clear use cases defined for the external KMS [8].

5.1.3 Cloud Service Using External KMS

The third pattern describes an approach which replaces the platform-native KMS with an external KMS as the central point for managing all keys within the organization instead of the platform's native key management tools. The external KMS may be owned and managed by the customer or a third party. In this pattern, CSA assumes single-tenancy for the hardware, which means that the HSM appliance(s) are reserved solely for the one customer. [8]

The greatest benefit of an external KMS is the ability to restrict CSP's access to the keys. In a single-cloud architecture, this offers little benefits, since the access tokens to the KMS are stored in the cloud platform for every integration that uses the keys. However, in a multi-cloud or hybrid cloud environment, the separation of keys into an external KMS can support data privacy requirements, store keys used in backups, and provide an unified key management approach for distributed environments. Furthermore, CSA highlights improved configurability of algorithms, protocols and key lengths [8].

CSA mentions high cost as a downside in this pattern [8]. Costs incur from the dedicated hardware as well as the external system and its cost model. In comparison to a platform-native KMS, the scalability, performance and latency may be worse in this pattern [8]. Furthermore, the integrations with platform-native systems take longer and add complexity to key management [8]. Therefore, a larger budget and staff are needed for arranging key management with this pattern.

CSA generally recommends this approach if the CSP does not offer a native KMS [8]; however, all major public CSPs offer a variety of KMS products. The external KMS is not dependent on platform availability – therefore, the deployment of an external KMS can be executed before the platform is configured and ready for use, enabling its use in the on-premises environment as well. The choice between the native key management tools and externally managing keys is a major architectural decision requiring thorough planning.

5.1.4 Multi-Cloud KMS

The last pattern refers to an approach that combines multiple KMS patterns described above. In practice, this could mean using both platform-native and external KMSs, or describe a multi-cloud architecture with KMSs in both platforms. CSA highlights the resilience improvements with this approach – having key backups in another KMS.

The pattern is described as a strategic approach to key management [8]. When combining different KMSs, it is desired to get the best out of all worlds and not to stumble on the complexity of the implementation. Therefore, the pattern requires the highest level of expertise and is suggested to organizations with stringent compliance requirements and a mature environment [8].

The downsides of this approach are the highest cost due to overlapping solutions along with the possibility of getting tangled into complicated web of integrations with poor visibility between each other [8], as integrating multiple different KMSs with each other to support central management of keys creates challenges.

5.2 Architectural considerations of third-party KMSs

The CSA patterns offer a rather limited perspective to the use of third-party products in the KMS architecture. While the patterns acknowledge the use of external KMSs, the use of dedicated hardware is assumed.

Chapter 2.1.3 presented the option to purchase KMS as a service from third-party providers. This model provides the customer with a wider selection of technology options while also separating the main CSP's responsibility from key management, keeping keys and data in separate platforms. In comparison to the third CSA KMS pattern – *Cloud Service Using External KMS* – the service-based KMS may also incorporate multi-tenancy in the hardware, which is a typical characteristic in cloud-native services, as per CNCF definition of cloud-nativity [29].

Aside from service-based KMSs, it is also possible to host one's own KMS in the cloud. In a single-CSP scenario, this would mean hosting the KMS in the same platform as the rest of the components in the environment, replacing the platform-native key management tools with third-party products. Although such a solution is more common in on-premises or hybrid environments, this approach provides a wider technology selection. The customer can use the IaaS components of the cloud platform, including HSMs, to support best practices of secure storage and comply with possible FIPS requirements by making the system HSM-backed instead of only software-backed storage.

The use of third-party key management products affects the key management processes as well as the transition on a foundational level, but regardless of the differences, all of them can serve the purpose of modernizing key management into a unified approach.

5.3 Suggested approaches for further evaluation

The requirements for a KMS in complex environments, which are typically found in financial organizations planning a cloud transition, are variable. This section presents suggestions for the modernized key management architecture. The solution designs are based on CSA patterns extended by the possibility of using third-party key management products. In the solution design, cloud-nativity and criteria defined in Chapter 4 were considered. Based on these conditions, three suggestions were formed for further evaluation: platform-native KMSs, extending the platform-native KMSs with an external KMS, and a self-hosted KMS.

These three approaches represent distinct architectural solutions due to differences in where the key material resides, how centralized management is organized, and how various systems integrate with the KMS. While the requirements would allow for a broader set combinations of platform-native and third-party key management solutions, the suggested architectural designs were chosen for their clear distinctions and practical feasibility. For instance, combining a self-hosted KMS with an external SaaS KMS would be possible, but its characteristics are not sufficiently distinct or relevant for evaluation against the defined criteria. Limiting the designs to a set of three allows for a more detailed level of discussion about their differences with regard to each criterion.

5.3.1 Approach A: Native KMS

Approach A describes a KMS architecture that relies on the cloud platform's native key management tools. This approach is identical to CSA's cloud-native KMS pattern and may consist of software- or hardware-backed KMSs or different management models of HSMs or a combination of these.

The range of key management products offer flexibility to address varying key management requirements of different systems within the environment. As the products are offered by the same CSP, this flexibility does not compromise the central governance of key management through policies, logging, analytics, and automation. Figure 5 illustrates the environment during the transition phase using approach A.

The cloud-native KMSs can be taken into use when the platform preparation is done. Key management related policies and monitoring can be considered part of the preparation work, which means collaboration between the cloud platform specialists and key management specialists. When system-level cloud adoption journeys begin and each system goes through the reengineering process, the cloud-native KMS can be taken into use. While the platform offers a variety of options for the KMS, the appropriate product must be decided based on the system's requirements, such as data sensitivity and regulatory requirements.

This approach aims to provide a cost-efficient solution to key management, eliminate the complexity of integrations using the synergies of platform-native tools, and unified policy requirements and monitoring.

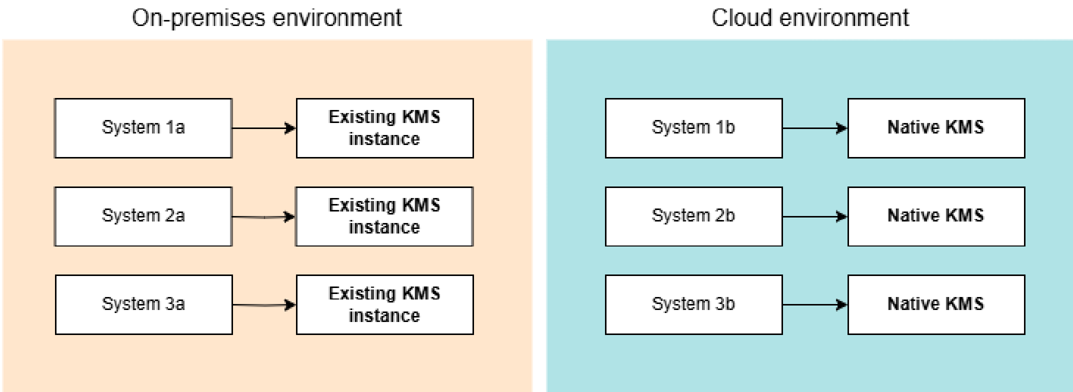

Figure 5: Approach A during a transition from on-premises environment to public cloud.

5.3.2 Approach B: Combining native and external KMS

Approach B extends the platform-native offerings with an external KMS, offered in the SaaS model by a third party. Out of the four CSA patterns, the last one best describes this approach. The most significant advantage of the additional KMS is the separation of data and keys into different places of storage, thereby clarifying roles and responsibilities and improving resiliency. It also offers added flexibility for transitioning from a single-CSP model to multi- or hybrid cloud architectures, or even migrating to another provider.

The choice of external KMS has an impact on the geographical location, type of storage, operational performance, technical functionality, and access control. This thesis assumes that the chosen KMS can support FIPS 140-3 level 3 compliance, provides all technical functionality outlined by key management best practices, supports secrets management, and is located in the same country as the organizations' main area of business. Furthermore, as the systems are specialized purely in key management, it can be expected that they support the requirements of typical key ceremonies [8].

Figure 6 illustrates the architectural structure of approach B. Since the external KMS is not dependent on the adoption of the new cloud environment, it can support key management improvements in the on-premises environment while the systems wait for their cloud adoption processes to begin. The acquisition and deployment of a SaaS KMS can be simple assuming a multi-tenant KMS where the customer is simply provisioned an account. However, reengineering the system can significantly change the requirements for key management, and integration to the KMS may have to be re-built during the process, so it is natural that most systems take the modernized approach into use during the cloud adoption.

As shown in Figure 6, keys can be originated in the external KMS and integrated with either cloud systems directly or through a native KMS, or they can be originated in the platform-native KMS with no traffic to the external KMS. The distribution of key management between two systems creates challenges with unifying policies. Furthermore, centrally managing and monitoring a KMS architecture consisting of products delivered by different vendors creates challenges. As CSA notes, successful

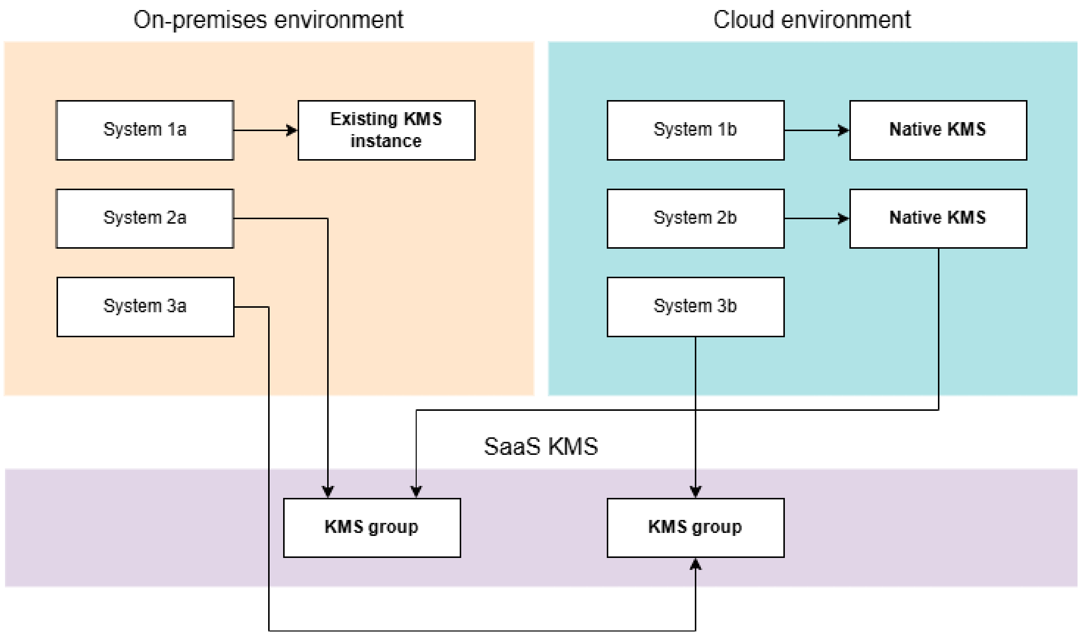

Figure 6: Approach B during a transition from on-premises environment to public cloud.

implementation requires thorough planning and a high level of expertise in both key management and cloud technologies to avoid difficulties arising from system complexity [8].

Although the external KMS could completely replace the platform-native KMS as presented in the second CSA pattern, the usability perspective and latency requirements necessitate the use of the platform-native KMS. It is not feasible to force the requirement of double integration, first between the system and the native KMS, and then between the native and external KMS, which would enable transferring monitoring completely to the external KMS. Therefore, central management and policy compliance monitoring must inevitably be done separately in the native and the external KMS.

5.3.3 Approach C: Hosting own KMS in the cloud environment

Approach C presents a solution that entirely eliminates the use of platform-native KMSs, replacing them with self-hosted third-party key management software within the same cloud platform. The main goal is to provide greater control over technology choices and address deficiencies in the functionality of the cloud platform's native KMSs. For this reason, this approach may be selected by organizations willing to build more extensive customization around the solution. This solution may offer improved portability and expandability across other cloud deployment models, such as hybrid cloud [3]. However, it loses the synergies with the cloud platform's native tools, such as policy management and governance.

Similarly to Approach B, the technology choices affect the solution's capability to support key management best practices and secrets management. Many third-party solutions support this functionality [3], and therefore this thesis assumes that such a product is chosen in the architectural design. Furthermore, HSM support depends on the selection of technology, although it is available in many products [3].

While the self-hosted KMS provides more control over technology choices, it does not offer additional segregation of key storage and data storage like Approach B does. Since it resides on the public cloud platform, CSP-side access to keys cannot be restricted, and the geographical location of keys remains tied to the CSP's datacenter locations.

Figure 7 presents the adoption of the self-hosted KMS during the cloud transition. The system-level adoption is rather straightforward, as there is no need or possibility to choose from a variety of KMS solutions. However, the deployment, configuration, and integration processes are significantly heavier in comparison to the other approaches since the customer responsibility is much higher: a self-hosted KMS can be considered an IaaS model from the public cloud service provider's perspective, as the CSP is only providing the underlying infrastructure and not responsible for application-level maintenance.

The cloud platform preparation phase must be completed before the deployment process of the KMS can be started. Due to the longest duration to achieve readiness of the KMS, this approach acts as a blocker for other systems' cloud adoption processes in case they have dependencies on the KMS. This delay must be considered when making the architecture decision.

Since approach C provides the most limited selection of key management products to integrate with, it is likely highest in cost. The customer must have more personnel allocated for the deployment and management of the KMS as well as pay for the infrastructure fees and software licensing. Often, a platform-native HSM-backed KMS is significantly more affordable than a dedicated HSM [50][9].

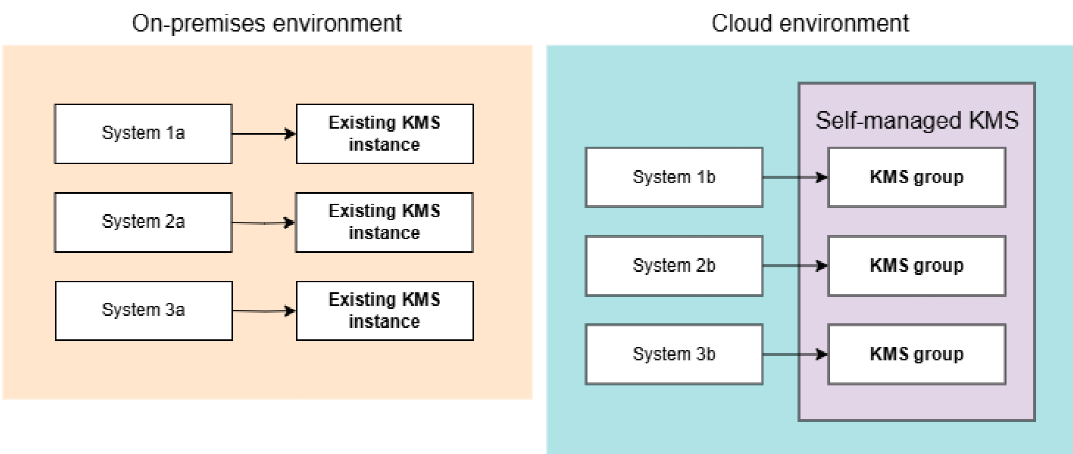

Figure 7: Approach C during a transition from on-premises environment to public cloud.

6 Approach comparison

This chapter reviews the functional, operational, and business evaluation criteria presented in Chapter 4 and discusses the differences between approaches A, B, and C defined in section 5.3. Rather than focusing on specific technology selections, the comparison is based on architectural differences. After the evaluation of the approaches with regard to the criteria, the chapter provides a summary of each approach's ability to meet the criteria – fully, partially, or insufficiently. However, the summary does not explicitly rank the approaches from the most to least suitable as each organization may place different emphasis on individual criteria, potentially or even likely leading to different conclusions. The outcome of this comparison aims to highlight the main differences in the most critical factors influencing the choice of an architectural approach for key management.

6.1 Functional criteria

FC1	A	B	C
Provides essential features that align with key management best practices and support the core operations in each phase of the key lifecycle.	Yes	Yes	Yes

The fundamental functionality for managing key lifecycle is provided in all solutions. The foundation for developing key management systems is typically based on the NIST Recommendation for Key Management, also describing key management best practices. Therefore, key management systems incorporate features needed to follow the best practices. However, platform-native KMSs assume that key related processes are primarily automated and focus on supporting operations within cloud workloads. As a result, they often lack comprehensive key distribution mechanisms, which are more commonly found in third-party KMS products designed for a wider selection of key management use cases.

Although KMS developments shares a common set of best practices, there are differences among solutions in terms of supported algorithms, available key lengths, and related attributes [8]. Approach A, which relies exclusively on platform-native KMSs, supports currently recommended algorithms and key lengths, and generating strong keys for encryption and signing is not an issue. Rather, the primary challenge lies in integrating with legacy systems that require non-standard or outdated key configurations, or that lack common APIs with cloud-based KMSs optimized for modern technologies. Such legacy systems should have a plan for updating the cryptographic technology under DORA RTS on ICT risk management framework [42]; therefore, in the worst case, their keys can be managed using existing practices until the system modernization has been completed in the cloud adoption.

While the selection of a cloud platform in general is not determined by the features of its native KMS, the choice of a third-party KMS can be heavily influenced by the characteristics of a particular product. Organizations may opt for a third-party KMS due to its architectural preferences, broader use case coverage, or specific functionality which has been identified as a requirement. These considerations are typically based on known or anticipated use cases within the unique technical environment and cannot be generalized across the financial sector as a whole.

In terms of key lifecycle management and cryptographic operations, the solutions deliver an equivalent level of functionality as a result of their shared foundation in best practices and expected functionality. However, Approaches B and C allow organizations to select technology tailored to specific needs, including cases where certain requirements are not supported by the platform-native solutions. These approaches also may also offer greater customizability and transparency, whereas platform-native KMSs often operate in a more *black-box* manner [8].

FC2	A	B	C
Enforces centralized key management policy through technical controls, including restrictions on the use of weak algorithms, inadequate key lengths and validity periods.	Yes	Partial	Partial

The unified approach to defining, enforcing, and governing a key management policy was found to be one of the most significant deficiencies in on-premises key management and therefore important to address in the key management modernization. The cloud-native approach to policies enables using technical controls to support enforcement of the policy [21], reducing reliance on purely manual processes that are relying on policy awareness. Typical on-premises approaches that involve decentralized key management have proven the manual approach inefficient in practice [4]. However, extending a cloud-native policy across third-party products is challenging. For this reason, key management policy configurations should reside within the chosen KMS itself. Fortunately, KMS products offer their own policy management capabilities, although the coverage of the product-specific policies is less likely to be as extensive as platform-native options and may require a greater amount of manual governance.

Approach A leverages the cloud platform's native policy mechanisms, and correspondingly, Approach C centralizes key management policy definitions to be implemented within the third-party KMS. Approach B, which uses both platfrom-native and third-party KMSs, introduces complexity in monitoring and policy definition as the key management policy cannot be centralized within a single service. This duplication increases the manual workload required for policy management. Nevertheless, the use of two systems, combined with a higher level of automation, still represents a significant improvement over the manual policy management common in on-premises environments.

Platform-native controls may be used to detect non-compliant key usage regardless of whether platform-native KMSs are employed. This is due to the fact that third parties lack the capability to monitor keys and operations within the cloud environment. Consequently, Approach A is the only one that enables a fully centralized policy enforcement model.

FC3	A	B	C
Supports granular access control with respect to separation of duties in key management.	Patial	Yes	Yes

Platform-native KMS solutions integrate seamlessly with the platform's IAM, and KMSs support the definition of granular access controls. In cloud architectures, it is typical to maintain a separate instance of the KMS for each system, thus limiting access to only the personnel responsible for that system. While platform-native solutions typically follow general-purpose IAM patterns, third-party KMSs offer access controls tailored specifically for key management. These controls include traditional roles such as key custodians, approvers, and administrators, which can better support key ceremony requirements than platform-native KMSs [8]. In Approach A, a dedicated HSM is operated using the appliance-specific interface rather than the platform-specific one [9], thereby providing a more traditional approach to key management roles. However, dedicated HSMs are resource-intensive and expensive solutions if such roles are required widely across the organization, and their integration with systems demands a high level of expertise compared to the platform's SaaS-based KMSs.

In third-party products, it is more typical to have a single instance – perhaps separate instances for production and non-production environments – where access to individual keys is controlled within the KMS. The convenience of access management in this model depends on the chosen technology and its compatibility with the organization's broader access management practices.

In general, platform-native KMSs support a sufficient level of granularity with access controls. Approach A simplifies access control by aligning it directly with the cloud platform's IAM model, which can be seen as an advantage for operational consistency. Approach B combines this approach with a third-party SaaS KMS and its separate access management, introducing additional workload. Specialized third-party products may improve key management specific access control features, but may also add unnecessary complexity. Approach C centralizes all key-related access management within a single third-party product, eliminating the need to manage platform-native KMS permissions. Third-party KMSs are expected to support single sign-on (SSO), removing the need for separate credentials to access the system.

FC4	A	B	C
Enables automated key rotation.	Yes	Yes	Yes

All of the solutions improve automation capabilities, addressing one of the most significant challenges of key management in on-premises environments. However, a more critical consideration is the system-side support for key rotation, which must be considered as a part of each system's re-engineering process. While automated rotation is rather straightforward to implement within the KMS itself, the automatic adoption of a new key version is not always simple from the system perspective. Therefore, rather than focusing solely on the automation feature, greater emphasis should be placed on monitoring key validity periods and the age of cryptographic keys. The monitoring capabilities can also be enhanced by cloud technologies; however, the implementations differ based on chosen technology and are further discussed in **OC3**.

FC5	A	B	C
Includes secrets management capabilities.	Yes	Yes	Yes

Public cloud platforms may offer secrets management tools either integrated with the KMS or separately from key management. Regardless of the different implementations, all CSPs provide functionality for secrets management in some form. The same applies to third-party KMS vendors, whose products also include support for secrets management. The emphasis on secrets, however, varies between solutions. With third-party products, organizations may even choose a secrets management system that incorporates key management, rather than a dedicated key management system that also extends support to other forms of secrets.

Although both platform-native and third-party KMSs support secrets management in general, organizations may choose to implement secrets management using processes and tools for secrets in comparison to what is outlined in key management policy. For example, with Approaches B and C, secrets management may be excluded from the scope of third-party KMS solutions, and instead, managed in platform-native secrets management systems. When migrating existing secrets, the capabilities of each option must be evaluated. As the number of secrets in an organization is typically much higher than the number of cryptographic keys, a manual approach for migration may not be feasible. Consequently, secrets management may have significant influence on the KMS architecture decision and requires more evaluation in the technology selection phase.

Secrets management often requires more comprehensive capabilities than those offered by key management solutions. Beyond managing the secrets' lifecycle, the topic covers prevention of unintentional exposure of secrets in, e.g., source code which requires secrets detection in CI/CD processes and version control [51]. Furthermore, the cloud platform's threat detection capabilities should be used to identify exposed secrets, including keys, in cloud workloads.

6.2 Operational criteria

OC1	A	B	C
Demonstrates high usability and seamless integrability with other systems to decrease friction in KMS adoption.	Yes	Partial	Partial

When aiming to maximize platform synergies in a single-CSP architecture, the most intuitive approach for users is to adopt the cloud-native KMS. Any alternative requires comprehensive awareness training and introduction to the chosen key management solution. In approach A, the solution design is straightforward: limit KMS selection to platform-native tools and manage policies and access through cloud-native mechanisms. However, in security, simplicity does not always lead to optimal results. Simplified key management processes may lead to a higher risk of misconfigurations, unintended key usage, and insufficient metadata or documentation on key use cases, which can complicate incident response involving cryptographic keys.

Approach B is architecturally the most complex. While this introduces challenges in governance and administration, users can in fact benefit from the variety of KMS tools to choose from. The flexibility of the approach allows users to choose between simple integrations with native KMSs and more safeguarded operations via an external KMS, depending on the sensitivity of their requirements. While integrations involving the external KMS may be more demanding, their use is not mandatory when the native KMS is sufficient for the intended purpose.

Approach C offers a single platform for key management, aiming to support the organization's all key management use cases. From the perspective of usability and coverage for different use cases, this approach presents a greater risk due to the elimination of use-case specific KMSs. However, if the technical design is carefully validated and all organizational use cases are clearly identified and confirmed as feasible with the selected technology, a self-hosted KMS can be highly effective. However, system integrations may require more effort compared to the platform-native solution.

Overall, the usability of each solution depends both on the alignment with the organization's processes and the chosen technology. Especially in Approaches B and C, the technology selection greatly affects usability and integrability. Furthermore, possible customizations in the self-hosted solution in approach C can either improve or impair usability. While each architectural approach can deliver a high-usability solution, third-party systems present higher risk of unsuitable technology choices. Conversely, with clearly defined use cases, the technology selection can provide a more suitable fit than platform-native tools.

OC2	A	B	C
Supports a range of use cases with varying levels of sensitivity, latency, capacity, and operational management.	Yes	Yes	Partial

Platform-native KMS offerings are not limited to a single type of KMS, but rather, provides a selection of KMSs ranging from software-based KMSs to payment-grade HSMs [9][18]. Consequently, Approach A inherently provides multiple options for implementing key management within the cloud platform. In Approach B, the external KMS is unlikely to match the latency and performance of platform-native solutions [8]. However, it offers distinct advantages for managing highly sensitive keys outside the public cloud platform, including those used for encrypting off-site backups that enable recovery on another platform.

Approach C, hosted on the same platform as the rest of the environment, can achieve latency and performance levels comparable to platform-native KMS products, assuming equivalent infrastructure capacity is allocated. While replacing a wider selection of KMSs with a centralized system, it simplifies the operational management of the KMS, and as mentioned earlier, this might even be one of the main drivers for selecting a self-hosted solution. However, this also restricts the flexibility to select KMS solutions to specific use cases.

Given the greater degree of customer responsibility in Approach C, in-house personnel must demonstrate a high level of expertise to ensure organizational confidence in the system's ability to support even the most sensitive use cases. Inadequate administration could lead to significant operational risks, including the unavailability of critical financial services due to KMS-related failures.

OC3	A	B	C
Facilitates governance and policy compliance monitoring with minimal reliance on manual processes.	Yes	No	Partial

Automation and advanced log analytics are among the most significant advantages of cloud nativity and they have become a critical component in cloud security. Therefore, it is essential that key management also adheres to cloud-native principles, whether implemented through platform-native tools or using third-party products.

The platform-native approach to governance and monitoring is the most straightforward and comprehensive as it naturally integrates with other governance policies beyond key management. Although the configuration of policies and their enforcement can be complex – often requiring several iterations before meeting all the requirements – Approach A has the strongest capability to meet these criteria through platform-native features.

For third-party KMS solutions, the level of automation is determined by the chosen technology. These solutions often prioritize key management functionality over the governance of system usage. As a result, the automation capabilities may be more limited and less customizable compared to cloud platforms' policy definitions, log queries, alerting mechanisms, and monitoring capabilities.

Although log data from third-party key management systems may be integrated into the cloud platform's centralized log management system, organizations must manually configure alerts for external systems. In case of Approach B, where policies need to be defined and enforced in both native and external KMS, the manual workload increases considerably. This makes Approach B the weakest option in terms of leveraging automated workflows in monitoring.

OC4	A	B	C
Deployment and maintenance responsibilities are feasible with allocated resources.	Yes	Yes	No

The different architectural approaches vary significantly in the effort required for deployment and maintenance of the KMS solution. Approach A supports both SaaS and IaaS models for key management. The IaaS approach refers to a dedicated HSM with administrative privileges allocated to the customer, where the HSM is operated via its built-in interface without CSP-provided KMS interfaces in between. In the majority of use cases, it is feasible to use the SaaS alternative to transfer the maintenance operations of the infrastructure to the CSP. Regardless, it is beneficial that the CSP offers the IaaS alternative for very specific use cases where the customer requires greater control.

Approach B introduces marginally more responsibility related to the configuration of the external SaaS KMS, but the maintenance workload remains relatively low since the third-party provider is responsible for application and infrastructure updates and development.

In contrast, Approach C is based on the self-hosted model, placing the full burden of configuring and maintaining the underlying infrastructure on the customer side. Although CSPs offer automated updates and patching configurable for IaaS components, the KMS software itself must be maintained and updated by in-house personnel. Moreover, any customization, development, or extensions of the KMS are the sole responsibility of the organization itself. This represents a significant factor affecting the choice of the KMS approach. While some organizations may appreciate this level of control, it lacks the benefits of shared responsibility found in Approaches A and B. Instead, the added responsibility can rather be seen as an additional challenge, often leading organizations to minimize internal personnel for KMS maintenance and to outsource these responsibilities whenever feasible.

OC5	A	B	C
Accommodates updates to cryptographic algorithms and key lengts in response to evolving technical standards.	Yes	Yes	Yes

CSPs themselves are responsible for massive numbers of keys. Not only those related to the infrastructure and services but also platform-managed keys of customers' cloud environments. Consequently, CSPs have a pronounced interest in ensuring that any advancements in cryptanalysis do not expose vulnerabilities in their offerings. As stronger algorithms and key lengths are created and standardized to retain effective cryptography, even in case of the emerging post-quantum computing, it is reasonable to expect these features to be made available to customers through platform-native KMS solutions. Therefore, platform-native KMSs can be regarded as a secure and future-proof choice.

Smaller, specialized KMS providers may not have the extensive resources of CSPs but often possess deep expertise in the topic of key management. These providers are typically proactive in supporting post-quantum cryptography (PQC) algorithms and key lengths. However, a poor technology choice among third-party KMSs may lead to infrequent system upgrades, introducing additional risks while also indicating lower level of support for emerging algorithms. In most cases, the future-proofness of third-party KMSs with respect to PQC readiness is expected to be comparable with platform-native KMSs.

6.3 Business criteria

BC1	A	B	C
Supports the use of FIPS-validated hardware where required by regulatory, contractual or internal obligations.	Yes	Yes	Yes

Approach A offers key management tools ranging from HSM-backed KMS products to dedicated hardware, which are all FIPS certified. All major CSPs offer level 3 validated HSMs. In Approach B, along with platform-native offerings, organizations can choose a technology vendor whose SaaS product includes FIPS-validated hardware. It is important to note that FIPS certification can influence the features of the overall KMS, such as key exportability, whereas platform-native KMSs have varying FIPS levels depending on the solution.

Approach C, being a self-hosted KMS deployed in the cloud, runs on infrastructure provided by the CSP. To implement a HSM-backed KMS in this model, the organization must integrate the KMS with the platform's HSM offerings. The KMS product itself must support such HSM integration. Many KMS solutions, including leading products like HashiCorp Vault, offer this capability [3], making this a feasible approach. However, similar to the external KMS in Approach B, the behavior of the overall system must be assessed when configuring the use of a HSM as the detailed settings might influence the features.

BC2	A	B	C
Enables handling and tracking policy exceptions centrally.	Yes	Partial	Partial

As discussed in **FC2** and **OC3**, each approach provides mechanisms for defining policies, although the comprehensiveness of the features vary. While policies are defined to support the security of cryptographic solutions, specific use cases may require exceptions to the policy. It is essential to consider how exceptions are handled, the reasons for granting them, and the countermeasures implemented to compensate for potential vulnerabilities arising from the policy violations.

Third-party KMS products with more limited functionality for policy management may lack built-in support for handling exceptions. This may result in manual processes and distributed documentation. In contrast, public cloud platforms enable central management of exceptions as part of their policy features, forming a core component in cloud compliance.

Although this may appear a minor detail, clear awareness and documentation of exceptions during an incident are essential in rapid recovery. Additionally, allowing controlled exceptions to strict deny rules offers stronger protection than disabling the policy and merely making audit notices, for example, to allow weak keys in specific use cases. Since exception handling capabilities can vary with technology choices in third-party solutions, this factor deserves careful consideration. Overall, platform-native solutions provide robust and effective support in this regard.

BC3	A	B	C
Offers architectural flexibility to support changes between CSPs or to hybrid cloud models.	No	Yes	Partial

As concluded in Chapter 4.3.1, organizations may face the need to switch from a full-cloud single-CSP architecture towards a hybrid cloud environment. Rearrangement of key management is part of the cloud exit strategy. It must be noted that not necessary all keys or systems are to be migrated from the cloud platform in all cloud exit scenarios – this might affect only regulated or highly critical data and workloads. Therefore, the use of platform-native KMSs is not inherently high-risk. Instead, the importance of an inventory of keys and their use cases is pronounced, since identifying the keys that need to be migrated in the cloud exit scenario is essential.

With approach A and its platform-native KMS architecture, the KMS must be completely changed in cases where keys need to be migrated outside the cloud platform. This means that new KMS capabilities are needed, in which case a different approach, such as an on-premises KMS solution or an external SaaS KMS as in approach B must be taken into use. The exit scenario requires further planning from the architectural perspective to mitigate the risk of creating de-centralized key management or a shadow KMS.

Approach B provides the possibility for more far-reaching planning with each use case of the KMS. System-level risk assessments can be made to consider whether the keys can be stored in the CSP's platform-native KMS or to keep them in an entirely separate location. Approach B may require building new integrations as workloads are moved from the cloud platform; however, the external KMS can still be considered a suitable option for the key management needs that cannot be taken care of by platform-native KMSs. The risk of further de-centralizing key management in a cloud exit scenario can be lower than with Approach A, especially if the changes need to be made within a short time frame.

Approach C and its self-hosted KMS can be migrated entirely out of the cloud environment and hosted on-premises or, perhaps, on another cloud platform. Such a change made at once can pose significant challenges with integrations and may create risks for service downtime when compared to system-by-system migrations. Another option is to build another instance off-premises or extend the KMS architecture with a separate system; however, this adds complexity and requires careful planning beforehand.

The role of key management exit scenarios related to external events and their more detailed implementation options could be a topic for further research. This evaluation is based on very high-level considerations of the exit scenarios. Approach B is found to provide the best flexibility and continuity of operations even with fast changes to cloud deployment models. Approach A is fairly well extendable with external KMSs, such as pivoting to Approach B, but in itself forms dependency to a single CSP. While approach C offers the possibility of migrating the system to another platform, doing so with no service downtime can be a challenge.

BC4	A	B	C
Meets resiliency, high-availability, and SLA requirements for systems that depend on the KMS.	Yes	Yes	Yes

One of the most significant factors in architecture decisions is the perceived resilience of the solution. When considering the target environment residing in public cloud, its resilience has two sides. On one hand, it provides greater redundancy on the geographical location of the services with backups across different datacenters and the possibility to change the location of hosted assets via configuration. On the other hand, the CSP – or external factors impacting the CSP's ability to meet service requirements – may become the weakest link, as demonstrated by the UniSuper incident discussed in section 3.3.

Approach A relies on platform-native components, simplifying the SLAs as they align with the rest of the platform, since the entire environment is hosted within the same provider. However, disaster recovery scenarios and off-site backups require an external key management solution in case the environment needs to be restored on a different platform. Approach B presents a significant improvement in resiliency with the external KMS. This enables risk-based decisions at the system level on whether keys should reside within the platform-native KMS, the external KMS, or even both as a backup mechanism.

Major CSPs can be expected to have the competence and incentive to ensure continuity of their platform-native products. In Approach B, the external KMS provider's capabilities for ensuring sufficient measures for DR and backup must be closely examined, considering the highly critical nature of financial services. This is one of the most critical considerations in Approach B as one of the external KMS's purposes is to improve resiliency; hence, the external KMS itself must not under any circumstances turn out to become a weak point.

Approach C, hosted on the same platform as other environment components, lacks inherent support for off-site backup keys, similar to Approach A. Although the resilience of the KMS itself can reach a similar level to platform-native KMSs, this approach introduces risks related to technology choice and its potential malfunctions, as well as reliance on in-house personnel to maintain high availability and minimize downtime. Consequently, the organization must be committed to investing in its key management function internally to ensure consistent service availability.

BC5	A	B	C
Presents a cost structure that is in proportion with the provided benefits.	Yes	Partial	No

To highlight the differences in the pricing of KMS components, platform-native software-based solutions are typically very inexpensive and multi-tenant HSM-backed KMSs incur somewhat higher costs whereas dedicated HSM appliances are significantly more expensive [50][9]. Third-party alternatives range from open-source offerings to high-cost enterprise solutions [3]. However, it can be assumed that the openly available alternatives are not applicable for the central management of a financial organization's keys; instead, organizations are expected to opt for solutions that provide a more extensive support model and comprehensive SLAs.

The costs in Approach A primarily derive from the usage of platform-native KMSs. Customers have limited ability to influence these costs as they have to choose from the limited selection of KMS products provided by the CSP. Nevertheless, the platform-native software-based KMSs tend to be the most cost-effective option across both platform-native and third-party offerings – or more specifically, the CSPs have included their cost into other fees. Approach A is generally the least expensive option due to the concentration of KMSs within one platform, although extensive use of dedicated HSMs may increase costs.

In Approach B, the use of platform-native KMSs might be smaller, resulting in lower costs associated with the cloud platform's KMS offerings. The cost savings will, however, not cover the price of an external KMS. Therefore, the overall cost of the KMS architecture typically becomes higher, although the total expense depends heavily on the selected external KMS vendor.

In Approach C, the cost is formed by the combination of software licensing and other fees from the technology vendor, cloud infrastructure expenses for hosting KMS in the cloud, and the internal workforce needed for KMS deployment, maintenance, and possible development related to customizations, extensions, and integrations. As a result of greater share of operational responsibility in the organization, this approach generally results in the highest total expenses among these three options.

6.4 Evaluation summary

The greatest differences between the architectural approaches are evident in solution complexity, resiliency, flexibility, and responsibility, leading to varying levels of effort, costs, and risks. Table 5 presents a generic comparison of the ability of the different approaches to meet the evaluation criteria. Approach A fully satisfies most criteria, with Approach B achieving nearly the same level despite complexity-related challenges. Approach C fully meets roughly half of the criteria while partially fulfilling most others.

Approach A offers robust KMS capabilities with strong synergies across platform-native components and seamless integrations with systems that depend on the KMS. This solution is the strongest candidate for pursuing cloud-native solutions with cen-

	A	B	C
FC1	Yes	Yes	Yes
FC2	Yes	Partial	Partial
FC3	Partial	Yes	Yes
FC4	Yes	Yes	Yes
FC5	Yes	Yes	Yes
OC1	Yes	Partial	Partial
OC2	Yes	Yes	Partial
OC3	Yes	No	Partial
OC4	Yes	Yes	No
OC5	Yes	Yes	Yes
BC1	Yes	Yes	Yes
BC2	Yes	Partial	Partial
BC3	No	Yes	Partial
BC4	Yes	Yes	Yes
BC5	Yes	Partial	No
Total Yes	13	10	7
Total Partial	1	4	6
Total No	1	1	2

Table 5: Summary of results.

tralized management, automation, and advanced analytics. For developers experienced in cloud technologies, cloud-native KMSs are likely to provide the most intuitive user experience.

The main challenge with Approach A is the dependency on a single CSP. Future architectural changes – such as those driven by regulatory restrictions on cross-border data transfer or requirements to limit CSP access to customer data – cannot be easily incorporated due to the lack of support for key segregation. In addition, the centralized approach does not support the management of off-site key backups. While these risks are significant, they may be acceptable for many financial organizations since it can be observed that cloud adoption continues to increase. Cloud providers are also working on sovereign cloud solutions.

Approach B aims to combine the strengths of platform-native KMSs and external cloud-based KMS. Its greatest benefit in comparison to other approaches is flexibility: systems can store keys within the public cloud platform, in an external KMS, or even both for redundancy. Furthermore, adopting a hybrid model or changing the CSP altogether poses smaller challenges when using a platform-agnostic KMS.

However, this flexibility comes at the cost of complexity, affecting users, administrators, and auditors. Specialized personnel and user training are required, and policy management becomes distributed, making it harder to leverage automation and analytics for KMS monitoring. Despite this, the approach still improves these aspects compared to de-centralized on-premises solutions. Without careful planning and governance, the solution risks becoming fragmented.

Despite demonstrating the weakest capabilities for unified key management – identified as one of the main criteria based on past experience, best practices, and framework recommendations – the added flexibility and resilience remain strong arguments in favor of this approach. Incorporating a separate external KMS can significantly reduce risks in key management when encountering regulatory, contractual, or threat-driven changes, provided the organization can manage the solution's complexity effectively and fully leverage its benefits.

Approach C achieves its greatest strength through its highly centralized architecture, offering a single KMS for all users. Furthermore, the solution enables organizations to freely choose the KMS technology instead of sticking with platform-native KMSs and their selection of features. This improves usability but comes with high maintenance demands and costs, which may discourage choosing this approach. The unified design limits flexibility and use case diversity, making it less attractive given the required resource investment.

It may still be the most suitable option for specific needs, particularly for organizations already operating a centralized self-hosted KMS on-premises. In cases where this is found to be a good strategy, it is feasible to migrate the KMS to the cloud during the transition. Additionally, the desire to customize and possess greater responsibility over the KMS technology stack supports the selection of Approach C.

In conclusion, organization-specific needs, assets, and constraints strongly influence the optimal approach. Architectural decisions outline the capabilities of the KMS in terms of functionality, resiliency, flexibility, and maintainability. The evaluation criteria have proven effective in highlighting the architectural differences, yet the results show that all three approaches are fundamentally viable. While none of the approaches presents inherently insecure or unsuitable architectures in the current landscape, their long-term suitability for organizational needs depends on technology choices, administrative competence, and alignment with technical use cases in the organizational context.

7 Discussion

This thesis examined key management requirements, outlined three distinct key management approaches, and evaluated their differences. The results emphasized the architectural solution's implications for organizational IT management efficiency, the ability to leverage cloud native benefits, and the capability to provide resilient services, completing its critical role in financial services. This section further discusses some areas that were left for less attention during the evaluation.

Organization-specific requirements necessitate further examination, including more detailed review of applicable national legislation and the financial organization's specific role in the society, such as the Nordic countries' model for strong citizen authentication. Additionally, an investigation of common solutions in IT architecture within financial organizations, both on-premises and when targeting modern cloud environments, would provide a more constrained set of requirements for technology selection. Clearly defined use cases remain the foundation for any KMS architectural decision. Further research could involve a case study, incorporating technology selections with both the public cloud service provider and third-party products, and building proof-of-concept architectures with a defined set of features for comparison.

The relationship between secrets management and key management experiences a shift during a cloud transition. The role of secrets management may, in fact, be more significant than key management in modernized systems. The SRM leads to a notable decrease in infrastructure-related administrative processes, which also transfers many key management responsibilities to the CSP. Moreover, the use of platform-managed keys further reduces customer-side responsibilities over cryptographic keys. Secrets are more typically related to cloud workloads at the application level, which is generally less subject to division of control during cloud migration. However, secrets and keys experience different concerns related to their management. The question of secret residency may not be as relevant since many secrets, such as API keys or other application secrets, are sent around in web requests as a form of authentication. However, the backups of secrets may be crucial, and hence, external system may also be a justified consideration. Secrets management and its implementation either aligned with key management processes and tools or separately requires further evaluation.

While key management responsibilities are increasingly shifting towards the CSP's side, key ownership and related compliance has risen concerns. The topic has been addressed by CSA: What is relevant from the viewpoint of *ownership* is the actual access to keys, and it can be considered irrelevant who has created the key or been nominated as the *key owner* [8]. CSA uses the analogy of a storage unit, locked by the lessee-owned padlock, with the lessor holding a duplicate key in order to deliver shipments directly to the storage unit without requiring the lessee's presence [8]. In this scenario, both parties have access despite the lessee retaining ownership of the keys. Ownership is based on agreements, access is based on the concrete control measures. While keys are managed by the CSP, all control for managing data access for customers lies within the CSP's hands. Organizations must evaluate whether the convenience and reliability of this arrangement justify the potential risks associated with reduced direct control.

Existing studies and surveys do not provide a definitive picture of financial organizations' long-term strategies regarding single-CSP, fully cloud-based architectures – the focus of this thesis. This showcases the uncertainty of such an architectural decision in the current regulatory and geopolitical environment. While centralizing services under a single CSP can yield operational synergies, simplify workflows, reduce costs, and eliminate redundant systems, the dependency on a single platform increases risks.

Importantly, the potential need for architectural changes is not necessarily rooted in a lack of trust in CSPs' maturity, competence, or intentions. Their role in the global economy is significant, and regulatory or political changes that complicate the use of public cloud services could have wide-ranging effects. However, the relatively brief history of data regulation demonstrates that requirements can change frequently and unpredictably, influenced by shifts in political agendas and relationships.

In addition to navigating evolving regulatory conditions, CSPs remain attractive targets for cyber crime. While robust safeguards are in place, no organization is entirely immune to successful attacks that may cause service disruption or data breaches. Compromise of CSP credentials could grant attackers direct access to customer accounts without exploiting vulnerabilities in customer workloads, configurations, or processes. Given the substantial resources CSPs can allocate to security and resilience, the likelihood of a successful large-scale attack can be considered lower than the risks faced by an organization hosting their services in on-premises datacenters, taking full responsibility of its own IT assets.

The discussion expanded on the need for further evaluation of technology choices, the evolving role of secrets management, and the risks arising from shifting responsibilities in key management and architectural control. These considerations highlight the inherent complexity of adopting a single-CSP architecture in the financial sector. Even though the public cloud is now a mature and established technology, its application in critical financial services continues to demand a careful, context-specific risk assessment.

8 Summary

The transition from on-premises datacenters to a public cloud platform is an ambitious project for financial organizations. This shift demands fundamental changes to key management while also presenting opportunities to improve existing practices. As key management is essential to ensuring effective cryptography, a thorough evaluation of its implementation is required during the transition process.

The primary challenges in designing the new key management architecture arise from regulatory uncertainty, both regulation specific to the financial sector and broader data protection legislation. The geopolitical landscape further complicates the outlook for outsourcing infrastructure to the US-dominated cloud provider market. Additionally, large-scale reengineering of the IT environment, especially during the hybrid cloud phase, introduces evolving requirements for key management.

This thesis examined the requirements for key management within the financial sector and presented a set of evaluation criteria covering functional, operational, and business aspects. Functional criteria discussed technical capabilities and alignment with best practices, emphasizing centralized approaches to mitigate known issues encountered in de-centralized and siloed on-premises environments. Operational criteria addressed usability, automation, and analytics, focusing on how these capabilities can enhance the efficiency of key management leveraging the benefits of cloud-nativity. Business criteria considered regulatory requirements and the risks associated with changes in those requirements, along with the role of the KMS in resiliency and business continuity as well as risks related to changing regulations and their impact on the architecture decisions in financial organizations.

Using these criteria, three cloud-native architectural approaches for a single-CSP environment were defined and evaluated:

Approach A A platform-native approach, consisting of key management products offered by the CSP. These may include software- and hardware-based KMSs as well as managed and dedicated HSMs depending on the use cases.

Approach B Combination of platform-native tools and an external SaaS KMS provided by a third-party vendor. Based on the use case, systems may use only platform-native tools, external KMS, or integrate both for added resiliency and segregation.

Approach C A self-hosted KMS within the cloud platform, using CSP-provided infrastructure with a third-party-provided KMS software, completely replacing the use of platform-native KMSs.

After evaluation, none of the approaches presented immediate blockers for use as an architecture for key management in financial organizations. Approach A offered a seamless and intuitive user experience in cloud development but demonstrated a lack of flexibility and limited abilities to respond to changes in cloud architecture. Approach B provided a significant improvement in key management resiliency as well as support for key segregation. However, splitting key management across two distinct platforms weakens central management of keys and policies and complicates key management decisions and integrations. Approach C enabled greater control over the technology stack, allowing organizations to select the technology most suitable for organizational processes and use cases. Nevertheless, this solution incurs higher costs through added responsibilities within the organization and limits the cloud-native automation, analytics capabilities, and synergies with platform-native components.

While this research highlights the strengths and weaknesses of each architectural design, risks remain for all scenarios. Further research involving proof-of-concept implementations of selected technologies is required to support decision-making with more detailed use case descriptions. Such implementations could also help determine whether secrets management should be integrated with key management tools and processes or treated as a distinct domain. Moreover, an approach that prioritizes secrets management may potentially address key management requirements as well, given the substantial transfer of key management responsibilities to the CSP in the shared responsibility model.

The results indicate that organizations have the opportunity to choose from different architectural approaches while complying with regulatory, operational, and risk-based requirements coming from both internal and external sources. Each architectural solution inherently affects resiliency, flexibility, usability, and expenses; however, their impact can be managed through careful planning, alignment with organizational use cases, and robust technical implementation.

References

[1] National Institute of Standards and Technology, "Recommendation for Key Management," U.S. Department of Commerce, Washington, D.C., Tech. Rep. NIST Special Publication 800-57 Part 1 Revision 5, 2020.

[2] A. Mardon, G. Barara, I. Chana, A. D. Martino, I. Falade, R. Harun, A. Hauser, J. Johnson, A. Li, J. Pham, and N. Varghese, *Cryptography Online*. GM Press, 2021.

[3] I. Kuzminykh, B. Ghita, and S. Shiaeles, "Comparative Analysis of Cryptographic Key Management Systems," in *Internet of Things, Smart Spaces, and Next Generation Networks and Systems*, 2020, pp. 80—94.

[4] P. Somasundaram, "Unified Secret Management Across Cloud Platforms: A Strategy for Secure Credential Storage and Access," in *International Journal of Computer Engineering and Technology (IJCET)*, vol. 15, no. 2, 2024, pp. 5–12.

[5] National Institute of Standards and Technology, "Security of Interactive and Automated Access Management Using Secure Shell (SSH)," U.S. Department of Commerce, Washington, D.C., Tech. Rep. NIST IR 7966, 2015.

[6] HashiCorp, "HashiCorp Product Pricing," 2025, Available at: https://www.hashicorp.com/en/pricing?tab=vault (Accessed 9.8.2025).

[7] Cloud Security Alliance, "Key Management Lifecycle Best Practices," Cloud Security Alliance, 2023.

[8] Cloud Security Alliance, "Key Management in Cloud Services: Understanding Encryption's Desired Outcomes and Limitations," Cloud Security Alliance, 2020.

[9] Microsoft, "How to choose the right Azure key management solution," 2025, Available at: https://learn.microsoft.com/en-us/azure/security/fundamentals/key-management-choose (Accessed 28.7.2025).

[10] Swift, "Swift Customer Security Controls Framework v2026 – Customer Security Programme," Society for Worldwide Interbank Financial Telecommunication, 2025.

[11] Amazon Web Services, "AWS Payment Cryptography – Industry terminology," 2025, Available at: https://docs.aws.amazon.com/payment-cryptography/latest/userguide/terminology.html (Accessed 1.9.2025).

[12] European Union, "Directive (EU) 2022/2555 of the European parliament and of the council of 14 December 2022 on measures for a high common level of cybersecurity across the Union, amending Regulation (EU) No 910/2014 and Directive (EU) 2018/1972, and repealing Directive (EU) 2016/1148 (NIS 2 Directive)," in *Official Journal of the European Union*, 2022.

[13] CERT-EU, “Threat Landscape Report 2024: A Year in Review,” CERT EU, the Cybersecurity Service for Union entities, Rue de la Loi 107, 1049 Brussels, Belgium, Tech. Rep., 2024, available at: https://cert.europa.eu/publications/threat-intelligence/tlr2024/ (Accessed 9.7.2025).

[14] National Institute of Standards and Technology, “Securing Web Transactions: TLS Server Certificate Managment,” U.S. Department of Commerce, Washington, D.C., Tech. Rep. NIST Special Publication 1800-16, 2020.

[15] J. Anderson, “The Role of Identity and Access Management (IAM) in Securing Cloud Workloads,” *Obafemi Awolowo University*, 12 2022, Available at: https://www.researchgate.net/publication/389518277_The_Role_of_Identity_and_Access_Management_IAM_in_Securing_Cloud_Workloads (Accessed 9.7.2025).

[16] National Institute of Standards and Technology, “The NIST Definition of Cloud Computing,” U.S. Department of Commerce, Washington, D.C., Tech. Rep. NIST Special Publication 800-145, 2011.

[17] Tenable, “Deconstructing the shared responsibility model: How to avoid coverage gaps and confusion to better protect your business,” Available at: https://www.tenable.com/whitepapers/deconstructing-the-shared-responsibility-model (Accessed 14.8.2025).

[18] Amazon Web Services, “What is AWS Payment Cryptography?” 2025, Available at: https://docs.aws.amazon.com/payment-cryptography/latest/userguide/what-is.html (Accessed 6.8.2025).

[19] S. Chugh, “Bridging the Gap: Industry Perspectives and Trends in Cloud Security, and Opportunities for Collaborative Research,” in *5th IEEE International Conference on Trust, Privacy and Security in Intelligent Systems and Applications (TPS-ISA)*, 2023, pp. 400–404.

[20] A. Rahman, M. Ashrafuzzaman, M. M. I. Jim, and R. Sultana, “Cloud Security Posture Management Automating Risk Identification and Response in Cloud Infrastructures,” in *Academic Journal on Science, Technology, Engineering And Mathematics Education*, vol. 04, no. 03, 2024, pp. 151–162.

[21] Microsoft, “Azure Policy built-in definitions for Key Vault,” 2025, Available at: https://learn.microsoft.com/en-us/azure/key-vault/policy-reference (Accessed 9.8.2025).

[22] C. Lekkala, “Modernizing Legacy Data Infrastructure for Financial Services,” in *International Journal of Science and Research (IJSR)*, vol. 10, no. 1, 2021, pp. 1634–1638.

[23] S. V. Subramanyam, "Cloud Computing and Business Process Re-Engineering in Financial Systems: The Future of Digital Transformation." in *International Journal of Information Technology and Management Information Systems (IJITMIS)*, vol. 12, no. 01, 2021, pp. 126–143.

[24] P. Nutalapati, "A Review on Cloud Computing in Finance – Transforming Financial Services in the Digital Age," in *International Research Journal of Engineering & Applied Sciences*, vol. 12, no. 3, 2024, pp. 35–45.

[25] Cloud Security Alliance, "State of Financial Services in Cloud," Cloud Security Alliance, 2023.

[26] Amazon Web Services, "AWS Cloud Adoption Framework (AWS CAF)," 2025, Available at: https://aws.amazon.com/cloud-adoption-framework/ (Accessed 15.7.2025).

[27] Microsoft, "Microsoft Cloud Adoption Framework for Azure," 2025, Available at: https://learn.microsoft.com/en-us/azure/cloud-adoption-framework/overview (Accessed 15.7.2025).

[28] Google, "The Google Cloud Adoption Framework," 2025, Available at: https://services.google.com/fh/files/misc/google_cloud_adoption_framework_whitepaper.pdf (Accessed 15.7.2025).

[29] Cloud Native Computing Foundation, "CNCF Cloud Native Definition v1.1," 2024, Available at: https://github.com/cncf/toc/blob/main/DEFINITION.md (Accessed 6.8.2025).

[30] A. George, "The Cloud Comedown: Understanding the Emerging Trend of Cloud Exit Strategies," in *Partners Universal International Innovation Journal*, vol. 02, no. 05, 2024, pp. 1–32.

[31] N. Zeghmouli, "The cloud computing dilemma for financial services institutions," in *Journal of Securities Operations & Custody*, vol. 17, no. 2, 2024, pp. 166–177.

[32] Cloud Security Alliance, "Key Management for Public Cloud Migration," Cloud Security Alliance, 2024.

[33] S. Suoranta, L. Haataja, and T. Aura, "Electronic Citizen Identities and Strong Authentication," in *Secure IT Systems, 20th Nordic Conference, NordSec 2015, Stockholm, Sweden*, 2015, pp. 213–230.

[34] European Union, "Regulation (EU) 2022/2554 of the European parliament and of the council of 14 December 2022 on digital operational resilience for the financial sector and amending Regulations (EC) No 1060/2009, (EU) No 648/2012, (EU) No 600/2014, (EU) No 909/2014 and (EU) 2016/1011," in *Official Journal of the European Union*, 2022.

[35] W. Krupa, M. Parkkonen, K. Stempel, N. Baglan, and V. Kontopoulou, "A critical assessment of the strong authentication system using bank credentials," in *University Of Helsinki*, 2022.

[36] M. Ryan, G. Withers, and F. den Harthog, "The Cloud Conundrum: Are Financial Institutions Heading for a Catastrophic Disruption Event?" in *IEEE Transactions on Technology and Society*, vol. 6, no. 1, 2025, pp. 80–89.

[37] Cloud Security Alliance, "Cyber Resiliency in the Financial Industry 2024," Cloud Security Alliance, 2024.

[38] Cloud Security Alliance, "Cloud Controls Matrix (CCM)," 2025, Available at: https://cloudsecurityalliance.org/research/cloud-controls-matrix (Accessed 30.7.2025).

[39] S. Klotz, M. Westner, A. Kopper, and S. Strahringer, "Causing factors, outcomes, and governance of Shadow IT and business-managed IT: a systematic literature review," in *International Journal of Information Systems and Project Management*, vol. 7, no. 1, 2019, pp. 15–43.

[40] Payment Card Industry Security Standards Council, "Payment Card Industry Data Security Standard, Version 4.0.1," 2024.

[41] Payment Card Industry Security Standards Council, "Payment Card Industry PIN Security, Version 3.1," 2021.

[42] European Union, "Commission delegated regulation (EU) 2024/1774 of 13 March 2024 supplementing Regulation (EU) 2022/2554 of the European Parliament and of the Council with regard to regulatory technical standards specifying ICT risk management tools, methods, processes, and policies and the simplified ICT risk management framework," in *Official Journal of the European Union*, 2024.

[43] K. Rong, Y. Ling, T. Yang, and C. Huang, "Cross-border data transfer: patterns and discrepancies," in *Journal of International Business Policy*, vol. 8, 2025, pp. 10—32.

[44] European Union, "Regulation (EU) 2016/679 of the European parliament and of the council of 27 April 2016 on the protection of natural persons with regard to the processing of personal data and on the free movement of such data, and repealing Directive 95/46/EC (General Data Protection Regulation)," in *Official Journal of the European Union*, 2016.

[45] International Trade Administration, "Data Privacy Framework (DPF) Overview," US Department of Commerce, 2025, Available at: https://www.dataprivacyframework.gov/Program-Overview (Accessed 31.7.2025).

[46] S. Batlle and A. van Waeyenberge, "EU–US Data Privacy Framework: A First Legal Assessment," in *European Journal of Risk Regulation*, vol. 15, 2024, pp. 191–200.

[47] Microsoft, “Announcing comprehensive sovereign solutions empowering European organizations,” 2025, Available at: https://blogs.microsoft.com/blog/2025/06/16/announcing-comprehensive-sovereign-solutions-empowering-european-organizations/ (Accessed 30.7.2025).

[48] S. Mamidi, “Privacy and Security in Cloud-Based Banking: A Technical Perspective,” in *International Journal on Science and Technology (IJSAT)*, vol. 16, no. 1, 2025, pp. 10—32.

[49] M. H. Murphy, “Assessing the Implications of Schrems II for EU-US Data Flow,” in *International and Comparative Law Quarterly*, vol. 71, 2022, pp. 245–262.

[50] Amazon Web Services, “AWS Key Management Service pricing,” 2025, Available at: https://aws.amazon.com/kms/pricing/ (Accessed 28.7.2025).

[51] Uber, “Building Uber’s Multi-Cloud Secrets Management Platform to Enhance Security,” 2025, Available at: https://www.uber.com/en-GB/blog/building-ubers-multi-cloud-secrets-management-platform/ (Accessed 11.8.2025).