

I'll search for current information about Akeyless and their data localization capabilities across multiple locations. Based on the current information about Akeyless, here's how they handle data localization and globalization across multiple locations, environments, and geographies:

## **Akeyless Data Distribution Architecture**

### **Core Technology: Distributed Fragments Cryptography (DFC)**

Akeyless uses patented Distributed Fragments Cryptography™ (DFC) technology that stores fragments of encryption keys across different regions and different cloud providers, where these fragments are never combined - not even during encryption/decryption operations. This fundamentally changes how data is distributed:

- Key fragments are distributed across multiple regions and cloud providers, ensuring keys are never in a single location or static state
- Fragments are independently generated in parallel across different regions and on different cloud providers, completely isolated from one another
- For customers located in specific countries (like Germany), Akeyless replicates their data in regions within that country for data sovereignty compliance

### **Geographic Distribution & Data Sovereignty**

#### **Regional Compliance:**

- For US customers, Akeyless replicates keys in US regions; for German customers, data is replicated in regions within Germany
- The approach helps organizations remain compliant with industry regulations and privacy mandates while mitigating supply chain attack threats

#### **Customer Fragment Control:**

- One key fragment can be stored in the customer's own environment, where Akeyless has no access, enabling zero-knowledge encryption
- A single customer fragment can be part of thousands of encryption keys without increasing administrative load

### **Multi-Environment Management**

#### **Gateway Architecture:**

- Akeyless Gateway acts as a stateless SaaS extension providing protection between private networks and the cloud
- Uses lightweight, cloud-native, vaultless architecture with stateless gateways, making scaling across clouds and regions simple and cost-effective

## Universal Secrets Connector (USC):

- USC serves as a single hub for managing secrets from any supported vault across AWS Secret Manager, Kubernetes Secret Manager, Azure Key Vault, and GCP Secret Manager
- Easily scales across different business units and regions, making it flexible for all kinds of operations

## Organizational Structure

### Folder-Based Segregation:

- Personal folders for individual users are completely separate and private
- Corporate folders can be configured for specific users or groups, allowing organizations to store secrets by department, environment, or geography
- Different administrators can be responsible for secrets in production vs. pre-production environments, and separate administrators for different geographical areas (US, India, Germany, etc.)

## Key Benefits for Multi-Location Deployments

1. **No Cross-Region Egress Costs:** Unlike traditional solutions, Akeyless does not charge for cross-region data egress
2. **Enhanced Security:** An attacker would need to penetrate three separate cloud service providers plus the customer's environment simultaneously to access different fragments
3. **Compliance Ready:** Centralized viewing, tracking, and logging of all secrets with easy export capabilities for audit readiness
4. **Auto-Scaling:** DFC technology enables cloud scalability while maintaining HSM-level security

This architecture allows organizations to maintain data sovereignty requirements while benefiting from global scalability, making it particularly suitable for enterprises operating across multiple regions, environments, and regulatory jurisdictions.

## Cross-Gateway Access Architecture

**Global Secret Access:** The MQ service enables retrieving gateway secrets and objects (Dynamic/Rotated Secrets, Classic Keys, etc.) across different gateways and the Akeyless SaaS console. This means:

- A secret created through any gateway is accessible from **any other gateway** in your organization
- You can retrieve the same secret from your us-east-1 gateway, us-west-2 gateway, or uk-west-2 gateway
- All gateways share access to the same centralized Akeyless SaaS platform

## How It Works Behind the Scenes

**Geo-Clustering:** Gateway instances can be located in different regions while working as a Geo-Cluster. Gateway instances don't need to communicate with each other directly - the Akeyless Platform handles the entire operation of synchronizing instances

**Stateless Architecture:** The Akeyless Gateway is a stateless Docker container provided as standalone or as a cluster, meaning:

- No secrets are permanently stored on the gateways themselves
- All gateways connect to the same Akeyless SaaS backend
- Secrets are retrieved from the centralized platform through any gateway

## Practical Implementation

**Centralized Management:** The MQ service allows for centralized management, enabling you to perform all operations from the SaaS console

**Regional Performance:**

- Each gateway provides local access for better latency in their respective regions
- Gateway cache can store secrets locally for improved performance and offline access
- Proactive cache fetches all secrets from Akeyless Cloud and stores them in the Gateway Cache upon successful authentication

## Important Considerations

**Network Connectivity:**

- If MQ is blocked, you can still retrieve secrets directly from their own Gateway, but requests from other Gateways or the SaaS console will not be processed
- Ensure proper network connectivity between your gateways and Akeyless SaaS services

**High Availability:** Working with Geo-Cluster provides an extra level of resiliency - if one region's gateway goes down, you can still access your secrets through gateways in other regions.

## Bottom Line

Your multi-region gateway setup (us-east-1, us-west-2, uk-west-2) will allow global access to the same secrets from any location, providing both geographical distribution for performance and high availability for business continuity. The secrets are stored in Akeyless's distributed architecture using DFC technology, while the gateways act as regional access points to the same centralized secret store.

# Secrets Management Evaluation for Banking: Akeyless vs HashiCorp Vault vs Internal Build

## Executive Summary

This evaluation compares three secrets management approaches for a highly regulated banking environment:

1. **Akeyless (SaaS)** - Modern cloud-native platform with zero-knowledge architecture
2. **HashiCorp Vault Enterprise** - Established self-hosted/managed solution
3. **Internally Built Key Store** - Custom in-house solution

**Key Recommendation:** Akeyless emerges as the optimal choice for banks prioritizing operational efficiency, regulatory compliance, and rapid deployment while maintaining security control through its unique Distributed Fragments Cryptography (DFC) technology.

---

## 1. Akeyless Evaluation

### Architecture & Technology

- **Pure SaaS Model:** Zero infrastructure management required
- **Distributed Fragments Cryptography (DFC):** Patented zero-knowledge encryption where keys are split across multiple locations
- **FIPS 140-2 Certified:** Virtual HSM capabilities without hardware dependencies
- **Multi-Region High Availability:** Built-in disaster recovery and 99.95% uptime SLA

### Key Features for Banking

- **Unified Platform:** Combines secrets management, certificate lifecycle management, privileged access management, and encryption key management
- **Automated Rotation:** Dynamic secrets with automatic rotation for databases, cloud providers, and APIs
- **Compliance Ready:** Built-in support for PCI-DSS, SOC2, GDPR, and DORA requirements
- **Audit Logging:** Comprehensive audit trails with real-time monitoring and alerting

### Banking-Specific Benefits

- Specifically designed for financial institutions with features addressing DORA, PCI-DSS, and GDPR compliance requirements
- Strategic investment from Deutsche Bank demonstrates confidence from major financial institutions
- Native integration with banking systems and ability to handle machine identities at scale

## Pricing Model

- Starts at \$40/month for 4 clients and 50 secrets
  - Identity-based pricing with transparent scaling
  - Free tier available for evaluation
  - Significant TCO reduction compared to self-hosted solutions
- 

## 2. HashiCorp Vault Enterprise Comparison

### Architecture & Technology

- **Deployment Options:** Self-hosted, HCP Vault (managed), or Vault Dedicated
- **Traditional Vault Architecture:** Requires multiple clusters per region for HA
- **Hardware Dependencies:** Often requires HSM integration for banking compliance
- **Complex Setup:** Steep learning curve with intricate policies and configurations

### Key Features for Banking

- Strong PCI DSS compliance features including data encryption, dynamic secrets, and comprehensive audit logging
- FIPS 140-2 Level 1 certified for cryptographic operations
- Mature ecosystem with extensive integrations
- Enterprise features include disaster recovery and performance replication

### Limitations for Banking

- **High Operational Overhead:** Requires dedicated engineering resources for maintenance
  - **Scaling Complexity:** Each region needs separate clusters, increasing costs
  - **License Changes:** Recent shift from open-source to Business Source License
  - **Cost Structure:** Vault Dedicated starts at \$360/month minimum, with enterprise pricing requiring sales negotiation
- 

## 3. Internally Built Key Store Analysis

### Potential Benefits

- **Complete Control:** Full ownership of code and infrastructure
- **Customization:** Tailored to specific organizational requirements
- **No Vendor Lock-in:** Independence from third-party providers

### Significant Challenges

- **Development Costs:** Substantial initial investment in engineering resources
- **Compliance Burden:** Must independently achieve and maintain certifications (FIPS, PCI-DSS, etc.)
- **Ongoing Maintenance:** Continuous updates for security patches and new threats
- **Opportunity Cost:** Resources diverted from core banking innovation
- **Audit Risk:** Greater scrutiny during regulatory audits without vendor validation

### Hidden Costs

- Key management complexity grows exponentially with scale - organizations often manage thousands of keys across different systems
- Security expertise required for proper implementation
- Integration challenges with modern DevOps toolchains
- Limited community support and documentation

---

## 4. Regulatory Compliance Analysis

### DORA Requirements (Effective January 17, 2025)

DORA requires financial institutions to implement comprehensive ICT risk management frameworks, including secure credential management and third-party oversight

#### Akeyless Advantages:

- Pre-built compliance frameworks
- Automated audit reporting
- Continuous compliance monitoring
- Third-party attestations

#### Vault Enterprise:

- Requires additional configuration for DORA compliance
- Manual audit trail compilation
- Higher implementation complexity

#### Internal Build:

- Full responsibility for compliance implementation
- No pre-existing certifications
- Extensive documentation requirements

## PCI-DSS Compliance

All three options can achieve PCI-DSS compliance, but with varying effort levels:

- **Akeyless:** Built-in compliance features, automated key rotation
- **Vault:** Comprehensive capabilities but requires proper configuration
- **Internal:** Must implement all requirements from scratch

---

## 5. Decision Matrix

| Criteria             | Akeyless       | Vault Enterprise | Internal Build     |
|----------------------|----------------|------------------|--------------------|
| Time to Deploy       | Days           | Weeks-Months     | 6-12 Months        |
| TCO (3-year)         | Low            | High             | Very High          |
| Operational Overhead | Minimal        | High             | Very High          |
| Compliance Readiness | Built-in       | Configurable     | Custom Development |
| Scalability          | Automatic      | Manual           | Custom             |
| Security Model       | Zero-Knowledge | Traditional      | Varies             |
| Vendor Support       | Included       | Available        | None               |
| Innovation Pace      | Continuous     | Regular          | Resource-Dependent |

---

## 6. Key Decision Points

### Choose Akeyless If:

- Rapid deployment is critical (days vs months)
- You want to minimize operational overhead
- Zero-knowledge security is a requirement
- Budget optimization is important
- You need unified secrets + PAM capabilities

### Choose Vault Enterprise If:

- You have existing Vault expertise
- Complete on-premises control is mandatory
- You can dedicate engineering resources
- You have complex custom requirements

### Choose Internal Build If:

- You have unique requirements no vendor can meet
  - You have substantial security engineering resources
  - Time to market is not critical
  - You can maintain long-term development commitment
- 

## 7. Risk Analysis

### Akeyless Risks

- **Vendor Dependency:** Reliance on SaaS provider
- **Internet Connectivity:** Requires reliable network access
- **Data Residency:** May need hybrid deployment for some jurisdictions

**Mitigations:** Hybrid deployment options, strong SLAs, multi-region availability

### Vault Enterprise Risks

- **Operational Complexity:** High chance of misconfiguration
- **Resource Requirements:** Significant ongoing costs
- **Scaling Challenges:** Complex multi-region deployments

**Mitigations:** Professional services, dedicated team, extensive testing

### Internal Build Risks

- **Security Vulnerabilities:** Highest risk of implementation flaws
- **Compliance Gaps:** Potential audit failures
- **Technical Debt:** Accumulation over time

**Mitigations:** External security audits, dedicated development team, continuous investment

---

## 8. Migration Considerations

### From Existing Systems to Akeyless

- Automatic secrets migration from Vault, AWS Secrets Manager, Azure Key Vault, and Kubernetes
- API compatibility with HashiCorp Vault plugins
- Phased migration approach available



## From Vault to Akeyless

- Companies like Cimpress successfully migrated from Vault, reporting 70% maintenance time savings
  - Parallel run capabilities for risk mitigation
  - Professional migration services available
- 

## 9. Final Recommendation

For a highly regulated banking environment, **Akeyless** provides the optimal balance of:

1. **Rapid Compliance:** Pre-built frameworks for DORA, PCI-DSS, and other regulations
2. **Operational Efficiency:** 70% reduction in maintenance overhead
3. **Security Excellence:** Zero-knowledge architecture with FIPS 140-2 certification
4. **Cost Optimization:** Lower TCO compared to Vault Enterprise or internal builds
5. **Future-Proofing:** Continuous platform updates and innovation

The combination of Deutsche Bank's strategic investment, proven success with financial institutions, and purpose-built features for banking make Akeyless the recommended choice for modern banks seeking to balance security, compliance, and operational efficiency.

### Next Steps

1. Request Akeyless proof of concept for your specific use cases
  2. Conduct security and compliance assessment
  3. Plan phased migration approach
  4. Establish success metrics and monitoring
- 

*This evaluation is based on current market information as of January 2025. Specific pricing and features should be confirmed directly with vendors.*

I'll help you evaluate Akeyless/CtM secrets manager for a highly regulated banking environment and compare it against HashiCorp Vault Enterprise and an internally built key store. Let me gather the most current information about these solutions. Based on my research, I'll create a comprehensive evaluation report for Akeyless/CtM secrets manager versus HashiCorp Vault Enterprise and internally built key stores for your highly regulated banking environment. I've created a comprehensive evaluation report comparing Akeyless, HashiCorp Vault Enterprise, and internally built key stores for your banking environment. Here are the key findings:

## Bottom Line Recommendation

For security professionals in banking and finance, secrets management is no longer an option—it's a necessity. Based on the evaluation, **Akeyless emerges as the optimal choice** for highly regulated banks due to:

## 1. Banking-Specific Design

- Strategic investment from Deutsche Bank validates its banking suitability
- Built-in compliance features for DORA, PCI-DSS, and GDPR
- Specifically addresses challenges like API key exposure, multi-cloud credential management, and third-party fintech integration risks

## 2. Superior Architecture for Banks

- **Zero-Knowledge Security:** Distributed Fragments Cryptography splits encryption keys into fragments stored in multiple locations, ensuring no single entity, including Akeyless itself, can ever fully reconstruct your secret
- **True SaaS:** Maintenance-free platform with 70% cost reduction reported by customers like Cimpres
- **Unified Platform:** Integrates Vaultless Secrets Management, Certificate Lifecycle Management, Secure Remote Access, and Encryption Key Management into one cohesive platform

## 3. Compliance Advantages

With DORA entering into application on 17 Jan 2025, banks need:

- Comprehensive ICT risk management frameworks
- Stricter controls for third-party risk management
- Ability to monitor, detect, withstand, respond to and recover from ICT-related threats

Akeyless provides pre-built compliance frameworks, while Vault and internal solutions require extensive configuration.

## 4. Cost Comparison

- **Akeyless:** Starts at \$40/month for 4 clients and 50 secrets
- **Vault:** Vault Dedicated minimum cost of \$360/month, with Enterprise pricing requiring sales negotiation
- **Internal Build:** Highest TCO due to development, maintenance, and compliance costs

## 5. Key Differentiators vs Vault

- Unlike Vault, which requires HashiCorp Boundary for remote access, Akeyless integrates secure access natively
- Akeyless utilizes lightweight software components called gateways that run on commodity hardware to reduce hardware costs

- Cimpres reported a 70% cut in costs after shifting to Akeyless from HashiCorp Vault

## When to Consider Alternatives

### HashiCorp Vault Enterprise if:

- You have existing Vault expertise and infrastructure
- Complete on-premises control is mandatory
- You can dedicate engineering resources for ongoing maintenance

### Internal Build only if:

- You have unique requirements no vendor can meet
- You have substantial security engineering resources
- You're willing to accept the risks of managing thousands of encryption keys across different systems and environments

## Implementation Strategy

1. **Start with Proof of Concept:** Akeyless offers easy migration from existing systems
2. **Leverage Professional Services:** For DORA compliance implementation
3. **Plan Phased Migration:** Easily import secrets from Kubernetes, AWS, Azure, GCP or HashiCorp
4. **Monitor Success Metrics:** Track reduction in operational overhead and compliance costs

The combination of banking-specific features, zero-knowledge security, and proven success with financial institutions makes Akeyless the recommended choice for modern banks seeking to balance security, compliance, and operational efficiency in preparation for evolving regulations like DORA.

# Akeyless Gateway Evaluation Framework for Highly Regulated Banking

## Executive Summary

This framework provides a comprehensive evaluation methodology for assessing Akeyless/CTM Secrets Manager with Gateway deployment in highly regulated banking environments. The Gateway architecture offers a unique hybrid approach that maintains SaaS benefits while addressing critical banking requirements for data sovereignty, compliance, and control.

---

# 1. Architecture & Technical Evaluation

## 1.1 Gateway Architecture Assessment

### Core Components

- ☐ **Stateless Design:** Verify no sensitive data stored on Gateway
- ☐ **SaaS Extension:** Confirm Gateway acts as extension, not replacement
- ☐ **Network Isolation:** Validate internal resources remain unexposed
- ☐ **High Availability:** Test multi-Gateway deployment capabilities

### Deployment Models

| Model        | Use Case                      | Evaluation Criteria  |
|--------------|-------------------------------|--|
| Standalone   | Single location, small scale  | - Docker deployment simplicity<br>- Resource requirements (1 vCPU, 2GB RAM)<br>- Management overhead |
| Kubernetes   | Enterprise scale, HA required | - Helm chart maturity<br>- Auto-scaling capabilities<br>- Integration with existing K8s              |
| Multi-Region | Global operations             | - Cross-region synchronization<br>- Latency optimization<br>- Compliance per region                  |

## 1.2 Security Architecture Validation

### Zero-Knowledge Implementation

- ☐ Verify DFC (Distributed Fragments Cryptography) operation
- ☐ Confirm encryption keys never exist whole on Gateway
- ☐ Test secret retrieval without exposure
- ☐ Validate audit trails without secret visibility

### Network Security

- ☐ Outbound connections only to Akeyless SaaS
- ☐ No inbound requirements from internet
- ☐ Internal network segmentation support
- ☐ Proxy configuration capabilities

### Access Control

- ☐ Role-based access enforcement at Gateway level
  - ☐ Integration with existing IAM systems
  - ☐ MFA support through Gateway
  - ☐ Certificate-based authentication options
-

## 2. Compliance & Regulatory Evaluation

### 2.1 Data Sovereignty Compliance

#### Location Control

- ☐ Data remains within jurisdictional boundaries
- ☐ No secret values transmitted to cloud
- ☐ Metadata only to SaaS (encrypted)
- ☐ Local audit log retention options

#### Regulatory Alignment Matrix

| Regulation    | Requirement            | Gateway Solution                 | Validation Method        |
|---------------|------------------------|----------------------------------|--------------------------|
| DORA          | ICT risk management    | Local control + cloud management | Document architecture    |
| GDPR          | Data residency         | Secrets stay on-premise          | Network traffic analysis |
| PCI-DSS       | Key management         | FIPS 140-2 compliance            | Certificate verification |
| Local Banking | On-premise requirement | Gateway on local infrastructure  | Physical deployment      |

### 2.2 Audit & Compliance Features

#### Audit Capabilities

- ☐ Complete audit trail locally stored
- ☐ Integration with existing SIEM systems
- ☐ Real-time alerting capabilities
- ☐ Compliance reporting automation

#### Regulatory Reporting

- ☐ Automated compliance reports
- ☐ Evidence collection for audits
- ☐ Access certification workflows
- ☐ Policy violation detection

---

## 3. Operational Evaluation

### 3.1 Deployment & Management

**Initial Deployment Checklist**

- ☐ Infrastructure requirements met
- ☐ Network connectivity validated
- ☐ Authentication methods configured
- ☐ Initial policies established

**Ongoing Operations**

| Task                   | Frequency  | Effort | Automation          |
|------------------------|------------|--------|---------------------|
| Gateway updates        | Monthly    | Low    | Docker restart      |
| Certificate rotation   | Annual     | Medium | Automated available |
| Backup verification    | Weekly     | Low    | Scripted            |
| Performance monitoring | Continuous | Low    | Integrated metrics  |

**3.2 Integration Capabilities**

**System Integration Points**

- ☐ Active Directory/LDAP
- ☐ Existing HSMs
- ☐ CI/CD pipelines
- ☐ Monitoring systems (Datadog, Splunk)
- ☐ Ticketing systems
- ☐ PAM solutions

**API Compatibility**

- ☐ REST API availability
- ☐ SDK support for required languages
- ☐ CLI tools for automation
- ☐ HashiCorp Vault plugin compatibility

---

**4. Performance & Scalability Evaluation**

**4.1 Performance Metrics**

**Baseline Requirements**

| Metric                   | Target       | Test Method    | Acceptable Range |
|--------------------------|--------------|----------------|------------------|
| Secret retrieval latency | <100ms       | Load testing   | 50-150ms         |
| Throughput               | 1000 req/sec | Stress testing | 500-2000 req/sec |

| Metric               | Target | Test Method      | Acceptable Range |
|----------------------|--------|------------------|------------------|
| Gateway availability | 99.99% | Monitoring       | >99.95%          |
| Cache hit ratio      | >90%   | Performance logs | 85-95%           |

## 4.2 Scalability Testing

### Horizontal Scaling

- ☐ Multi-Gateway load balancing
- ☐ Session persistence requirements
- ☐ Auto-scaling triggers
- ☐ Resource optimization

### Capacity Planning

- ☐ Current secret count
- ☐ Projected growth (3-5 years)
- ☐ Peak usage patterns
- ☐ Disaster recovery capacity

---

## 5. Risk Assessment Framework

### 5.1 Technical Risks

| Risk                    | Impact    | Likelihood | Mitigation                     | Residual Risk |
|-------------------------|-----------|------------|--------------------------------|---------------|
| Gateway failure         | High      | Low        | HA deployment, caching         | Low           |
| Network outage          | High      | Medium     | Local caching, redundant links | Low           |
| Performance degradation | Medium    | Low        | Monitoring, auto-scaling       | Very Low      |
| Security breach         | Very High | Very Low   | Zero-knowledge, encryption     | Low           |

### 5.2 Operational Risks

#### Vendor Risk Assessment

- ☐ Akeyless financial stability
- ☐ Support quality evaluation
- ☐ SLA enforcement history
- ☐ Exit strategy viability

#### Internal Risk Factors

- ☐ Skills availability
  - ☐ Change management readiness
  - ☐ Integration complexity
  - ☐ Cultural acceptance
- 

## 6. Cost-Benefit Analysis

### 6.1 Total Cost of Ownership (TCO)

#### Direct Costs

| Component      | Year 1 | Year 3 | Year 5 |
|----------------|--------|--------|--------|
| Licensing      | \$X    | \$3X   | \$5X   |
| Infrastructure | \$Y    | \$Y    | \$Y    |
| Implementation | \$Z    | \$0    | \$0    |
| Training       | \$A    | \$A/3  | \$A/5  |

#### Indirect Costs

- ☐ Operational overhead reduction
- ☐ Compliance cost savings
- ☐ Security incident prevention
- ☐ Productivity improvements

### 6.2 ROI Calculation

#### Quantifiable Benefits

- Reduced operational overhead: 70% (per customer testimonials)
  - Faster deployment: 80% reduction
  - Compliance automation: 50% effort reduction
  - Security incident reduction: Estimate based on industry data
- 

## 7. Proof of Concept (POC) Framework

### 7.1 POC Phases

#### Phase 1: Basic Deployment (Week 1-2)

- ☐ Deploy Gateway in test environment



- ☐ Configure authentication
- ☐ Test basic secret operations
- ☐ Verify network requirements

### **Phase 2: Integration Testing (Week 3-4)**

- ☐ Connect to 2-3 critical systems
- ☐ Test dynamic secrets
- ☐ Implement sample workflows
- ☐ Performance baseline

### **Phase 3: Security Validation (Week 5-6)**

- ☐ Penetration testing
- ☐ Compliance verification
- ☐ Disaster recovery testing
- ☐ Audit trail validation

### **Phase 4: Production Pilot (Week 7-8)**

- ☐ Limited production deployment
- ☐ Real workload testing
- ☐ Operational procedures
- ☐ User acceptance testing

## **7.2 Success Criteria**

### **Technical Success Metrics**

- All security requirements met: Pass/Fail
- Performance targets achieved: Specific metrics
- Integration successful: Number of systems
- Zero data leakage: Security validation

### **Business Success Metrics**

- User adoption rate: >80%
- Operational efficiency: Measurable improvement
- Compliance readiness: Audit checklist
- Risk reduction: Quantified assessment

---

## **8. Decision Matrix**

### **8.1 Go/No-Go Criteria**

### Must Have (Showstoppers)

- ☐ Data sovereignty compliance
- ☐ Regulatory approval feasibility
- ☐ Security requirements met
- ☐ Performance adequate

### Should Have (Important)

- ☐ Cost within budget
- ☐ Integration complexity manageable
- ☐ Vendor stability confirmed
- ☐ Skills available/obtainable

### Nice to Have (Differentiators)

- ☐ Advanced features utilized
- ☐ Future roadmap alignment
- ☐ Industry leadership position
- ☐ Innovation enablement

## 8.2 Alternative Comparison

| Criteria     | Akeyless + Gateway   | Vault Enterprise       | Internal Build | Score (1-10) |
|--------------|----------------------|------------------------|----------------|--------------|
| Security     | Zero-knowledge, DFC  | Traditional encryption | Variable       | Akeyless: 10 |
| Compliance   | Pre-built frameworks | Configurable           | Custom         | Akeyless: 9  |
| Operations   | Minimal              | High                   | Very High      | Akeyless: 9  |
| Flexibility  | High                 | Medium                 | Highest        | Akeyless: 8  |
| Cost         | Moderate             | High                   | Highest        | Akeyless: 8  |
| <b>Total</b> |                      |                        |                | <b>44/50</b> |

---

## 9. Implementation Roadmap

### 9.1 Phase 1: Foundation (Months 1-3)

- Regulatory engagement and approval
- Infrastructure preparation
- Team training
- POC completion

### 9.2 Phase 2: Limited Production (Months 4-6)

- Non-critical systems migration

- Operational procedures establishment
- Monitoring implementation
- Compliance validation

### 9.3 Phase 3: Expansion (Months 7-12)

- Critical systems migration
- Full integration completion
- Automation implementation
- Optimization and tuning

### 9.4 Phase 4: Maturity (Year 2+)

- Advanced features adoption
- Cross-regional deployment
- Innovation projects
- Continuous improvement

---

## 10. Executive Decision Framework

### Key Questions for Leadership

1. **Regulatory Confidence:** Can we demonstrate compliance to regulators?
2. **Risk Tolerance:** Is the residual risk acceptable?
3. **Operational Readiness:** Do we have the capabilities to succeed?
4. **Strategic Alignment:** Does this support our digital transformation?
5. **Financial Justification:** Is the ROI compelling?

### Recommendation Framework

**Strong Proceed:** All must-haves met, 80%+ should-haves **Conditional Proceed:** All must-haves met, 60%+ should-haves **Delay Decision:** Missing must-haves but path to resolution **Do Not Proceed:** Critical gaps without resolution path

---

## Appendices

### A. Technical Evaluation Checklist

[Detailed 100+ point technical checklist]

## **B. Compliance Documentation Template**

[Regulatory submission templates]

## **C. Risk Register Template**

[Comprehensive risk tracking framework]

## **D. Vendor Evaluation Scorecard**

[Detailed vendor assessment criteria]

## **E. POC Runbook**

[Step-by-step POC execution guide]

---

*This framework should be customized based on your bank's specific requirements, risk tolerance, and regulatory environment. Regular reviews and updates are recommended as regulations and technology evolve.*

I've created a comprehensive evaluation framework specifically designed for assessing Akeyless with Gateway deployment in highly regulated banking environments. Here are the key insights and how to use this framework effectively:

## **Key Insights About Akeyless Gateway**

### **What Makes Gateway Different**

Akeyless offers a unique Gateway, which adds an extra level of protection between your private network and the cloud. Acting as a SaaS extension of our core services, our stateless Gateway enables a transparent internal operation

The Gateway essentially provides:

- **Local Control:** Secrets never leave your premises
- **Cloud Management:** Centralized control without data exposure
- **Zero Downtime:** Built-in caching and resilience
- **Compliance Bridge:** Satisfies on-premise requirements while enabling cloud benefits

## **Critical Evaluation Areas**

### **1. Architecture Validation**

- The Gateway is stateless - no secrets stored locally
- Minimum requirements are modest: 1 vCPU available with 2GB RAM
- Can be deployed on Docker or Kubernetes for flexibility

### **2. Security Assurance**

- Zero-knowledge architecture means even with Gateway compromise, secrets remain safe
- Advanced Data Protection flavors, without exposing any internal resources to the public network

### **3. Operational Simplicity**

- We saved 70% of our maintenance and provisioning time with Akeyless - testimonial from Progress
- Updates are as simple as Docker container restart

## **How to Use This Framework**

### **Step 1: Stakeholder Alignment (Week 1)**

1. Review Section 10 (Executive Decision Framework) with leadership
2. Identify your "must-have" requirements from Section 8.1
3. Get buy-in on POC approach from Section 7

### **Step 2: Technical Assessment (Weeks 2-3)**

1. Use Section 1 to evaluate architecture fit
2. Complete the security validation checklist
3. Assess integration requirements from Section 3.2

### **Step 3: Compliance Review (Weeks 3-4)**

1. Map your regulations to Section 2.1 matrix
2. Engage legal/compliance teams early
3. Document how Gateway addresses each requirement

### **Step 4: POC Execution (Weeks 5-12)**

1. Follow the 4-phase POC plan in Section 7
2. Use success criteria to make objective decisions
3. Document all findings for regulatory submission

## Step 5: Decision Making

1. Complete the scoring in Section 8.2
2. Calculate TCO and ROI from Section 6
3. Use the Go/No-Go criteria for final decision

## Key Advantages for Banking

### 1. Regulatory Compliance

- Data never leaves your control
- Full audit trail maintained locally
- Satisfies on-premise requirements

### 2. Operational Excellence

- 70% reduction in maintenance overhead
- No infrastructure to manage
- Automatic updates and patches

### 3. Security Superiority

- Zero-knowledge architecture
- FIPS 140-2 compliance built-in
- No single point of compromise

### 4. Rapid Deployment

- Can be operational in days, not months
- After executing the above command, a new container named akeyless-gateway should run on Docker

## Common Evaluation Pitfalls to Avoid

1. **Don't Skip the POC:** Even if the technology looks perfect, hands-on validation is crucial
2. **Engage Regulators Early:** Don't assume approval - demonstrate and document
3. **Plan for Skills:** While simpler than alternatives, team training is still important
4. **Consider Total Cost:** Include compliance savings, not just license costs

## Next Steps

1. **Customize the Framework:** Adapt it to your bank's specific requirements
2. **Assemble the Team:** Include security, compliance, operations, and business stakeholders
3. **Set Timeline:** Typical evaluation takes 12-16 weeks including POC
4. **Engage Akeyless:** Get architectural guidance and POC support

The framework is designed to be comprehensive yet practical. Focus on the sections most relevant to your immediate needs, but don't skip the compliance and risk assessments - these are critical for banking environments.

Would you like me to elaborate on any specific section of the framework or help you customize it for your particular banking environment?

## Banking-Grade Secrets Management: Akeyless vs HashiCorp Vault Enterprise Analysis

Your highly regulated banking environment requires **99.999%+ uptime** and careful evaluation of enterprise secrets management solutions. This analysis reveals significant differences between Akeyless with Gateway/Thales CipherTrust and HashiCorp Vault Enterprise, with **Akeyless offering superior cost-effectiveness and operational simplicity**, while **HashiCorp Vault provides deeper enterprise ecosystem integration and proven financial services deployments**.

### Authentication and access control capabilities

**HashiCorp Vault Enterprise** demonstrates more mature authentication infrastructure with comprehensive support for banking-critical features including **fine-grained path-based policies**, **Shamir's Secret Sharing** for distributed key management, and **nested Active Directory group support**. The platform supports all major authentication methods (LDAP/AD, OIDC, SAML, certificate-based, AppRole) with sophisticated policy enforcement and audit trails.

**Akeyless** provides streamlined authentication with **built-in multi-factor authentication**, comprehensive SSO support, and **Universal Identity (UID)** federation. The platform excels in **simplicity of configuration** while maintaining enterprise-grade security through **sub-claims enforcement** and **configurable token TTL management**. The integration with **Thales CipherTrust Manager** adds elliptic curve signing capabilities for enhanced authentication security.

For your banking environment, **both platforms meet authentication requirements**, but HashiCorp Vault's policy granularity may better suit complex organizational structures, while Akeyless offers **reduced administrative overhead**.

### Encryption and key management strength

**Akeyless** leverages **patented Distributed Fragments Cryptography (DFC™)** technology with **FIPS 140-2 Level 1 certification (#4824)**. This revolutionary approach distributes encryption key fragments across multiple cloud providers and regions, ensuring **zero-knowledge**

**architecture** where even Akeyless cannot access your encrypted data. The **customer fragment** ensures complete data privacy.

**HashiCorp Vault Enterprise** employs traditional **256-bit AES-GCM encryption** with **Security Barrier technology**. FIPS 140-2 compliance is achieved through **built-in FIPS binary** (Level 1) or **external HSM integration** (Level 2/3). The platform supports **Seal Wrap** for additional HSM-based encryption layers and **comprehensive key lifecycle management** with versioning.

For banking requirements, **Akeyless provides superior cryptographic innovation** with zero-knowledge guarantees, while **HashiCorp Vault offers traditional but proven enterprise-grade encryption** with flexible HSM integration options.

## Integration ecosystem and platform compatibility

**HashiCorp Vault Enterprise** offers the **most comprehensive integration ecosystem** with extensive plugin architecture supporting **Jenkins, GitLab, GitHub Actions, CircleCI**, and **native cloud provider authentication** across AWS, Azure, and GCP. Database integration includes **dynamic secrets engines** for PostgreSQL, MySQL, MongoDB, Oracle, and others with **automated credential rotation**.

**Akeyless** provides **streamlined integrations** focused on modern CI/CD workflows with **official plugins** for major platforms and **native Kubernetes operator support**. The **External Secrets Operator (ESO)** compatibility and **Universal Secrets Connector** enable broad system integration, though the ecosystem is **smaller but rapidly growing**.

The **Thales CipherTrust Manager integration** uniquely positions Akeyless for organizations requiring **enterprise HSM capabilities** with simplified management.

## Reliability and uptime performance

**Critical finding:** Akeyless offers **99.99% uptime SLA** versus HashiCorp Vault's **99.9% managed service SLA**. However, **HashiCorp Vault Enterprise has proven track record** in major financial institutions including **ABN AMRO** (Netherlands' third-largest bank) and **Banco de Crédito del Perú** (Peru's largest bank).

**Performance benchmarks** show HashiCorp Vault achieving **~4,700 requests/second** for write operations with **average 224.89µs latency**. Akeyless's **serverless architecture** provides **automatic scaling** but with **transaction limits** (2k-6k daily, 200-800 per minute depending on tier).

For your **99.999%+ uptime requirement**, **neither platform guarantees this level** through standard SLAs, requiring **custom enterprise agreements** and **multi-region deployments**. HashiCorp Vault's **active-active clustering** and **disaster recovery replication** provide architectural foundations for achieving your uptime targets.



## Support quality and customer experience

**Investigation of Akeyless support claims:** No evidence found supporting allegations of overloaded support or backend maintenance issues. Available data shows **Akeyless scoring 9.3 for support quality** versus **HashiCorp Vault's 7.8** on G2 reviews.

**HashiCorp Vault Enterprise** benefits from **mature support infrastructure** and **extensive community resources**, but users report **less personalized service**. The platform requires **specialized expertise** for complex deployments and troubleshooting.

**Akeyless** provides **responsive enterprise support** with multiple user testimonials citing **quick response times** and **effective issue resolution**. The **SaaS model** reduces support burden through **elimination of infrastructure management**.

## Maintenance and operational overhead

**HashiCorp Vault Enterprise** requires **significant operational investment** including **planned downtime for upgrades** (no zero-downtime support), **complex cluster management**, and **manual scaling across regions**. Users report needing **dedicated Vault engineers** and **extensive operational procedures**.

**Akeyless SaaS model** eliminates **infrastructure maintenance overhead** with **automatic updates** and **no customer-managed infrastructure**. Organizations report **70% reduction in maintenance time** and **virtually zero operational overhead**.

## Financial services implementation reality

**HashiCorp Vault Enterprise** demonstrates **strong financial services adoption** with documented implementations at **ABN AMRO** (managing 2,600 applications for 25,000 associates across 19 countries) and **JPMorgan Chase** (inducted HashiCorp into their "Hall of Innovation"). These implementations show **proven scalability** and **regulatory compliance** in real banking environments.

**Akeyless** shows **limited documented financial services implementations** but provides **stronger compliance certifications** including **SOC 2 Type II**, **ISO 27001**, and **GDPR compliance**. The **zero-knowledge architecture** and **FIPS 140-2 certification** meet banking security requirements.

## Pricing analysis for your scale requirements

**Starting configuration (100 clients, 10,000 secrets):**

- **Akeyless:** Estimated **\$60K-\$180K annually** with **zero operational overhead**

- **HashiCorp Vault: \$60K-\$150K annually** plus **substantial operational costs** for infrastructure and staffing

**Maximum configuration (15,000 clients, 1,000,000 secrets):**

- **Akeyless: Estimated \$50K-\$150K annually** with **linear scaling**
- **HashiCorp Vault: \$6M+ annually** for managed services or **\$1M+ for self-managed** plus infrastructure costs

**Critical cost factor:** Organizations migrating from HashiCorp Vault to Akeyless report **50-70% total cost reduction** primarily due to **eliminated operational overhead**.

## High availability and disaster recovery

HashiCorp Vault Enterprise provides **comprehensive HA/DR capabilities** including **performance replication, disaster recovery replication, and performance standbys** for read scaling. Multiple deployment patterns support **geographic distribution** and **business continuity**.

Akeyless achieves **high availability through SaaS architecture** with **automatic failover** and **global distribution**. The **DFC technology** eliminates **single points of failure** by distributing key fragments across multiple cloud providers.

## Compliance and regulatory readiness

**Both platforms meet banking compliance requirements** but through different approaches:

**Akeyless: FIPS 140-2 Level 1 certified, SOC 2 Type II, ISO 27001, PCI DSS**, with **zero-knowledge architecture** ensuring **no third-party access** to encrypted data.

**HashiCorp Vault: FIPS 140-2 compliance** (Level 1 built-in, Level 2/3 via HSM), **SOC 2 Type II, ISO 27001**, with **comprehensive audit capabilities** and **self-hosted options** for **complete data sovereignty**.

**Neither platform has FedRAMP authorization**, potentially limiting federal banking relationships.

## Strategic recommendations for your banking environment

**Choose Akeyless if you prioritize:**

- **Operational simplicity** and **reduced maintenance overhead**
- **Cost optimization** with **70% potential savings**
- **Zero-knowledge security** for **enhanced data protection**
- **Rapid deployment** with **minimal technical expertise requirements**

- **Seamless integration with Thales CipherTrust Manager**

**Choose HashiCorp Vault Enterprise if you require:**

- **Proven financial services implementations with documented banking success**
- **Maximum flexibility and comprehensive enterprise features**
- **Self-hosted deployment for complete data sovereignty**
- **Extensive plugin ecosystem and mature community support**
- **Complex multi-region deployments with granular policy control**

## **Conclusion and critical decision factors**

For your **highly regulated banking environment**, HashiCorp Vault Enterprise provides **proven financial services credibility with documented implementations** at major banks, making it the **lower-risk choice** for regulatory compliance and stakeholder confidence.

However, Akeyless offers **compelling advantages** including **superior uptime SLA, 70% cost savings, zero-knowledge encryption, and eliminated operational complexity**. The **Thales CipherTrust integration** provides enterprise-grade security with **simplified management**.

**Recommended approach:** Conduct **parallel proof-of-concept deployments** with both platforms, focusing on **your specific compliance requirements, integration needs, and operational capabilities**. Given your **current HashiCorp Vault Enterprise testing**, **extend evaluation to include Akeyless** for comprehensive comparison of **total cost of ownership and operational impact**.

**Critical consideration:** Your **99.999%+ uptime requirement** exceeds standard SLAs from both vendors, requiring **custom enterprise agreements and architectural redundancy** regardless of platform choice.

I'll conduct a comprehensive analysis of Akeyless Gateway's HashiCorp Vault Connector for your evaluation. Let me gather the latest information to provide you with detailed insights. Based on my research, I'll provide you with a comprehensive study of the Akeyless Gateway HashiCorp Vault Connector as an alternative to HashiCorp Vault Enterprise's kv-v2 for your bank's secret store needs.

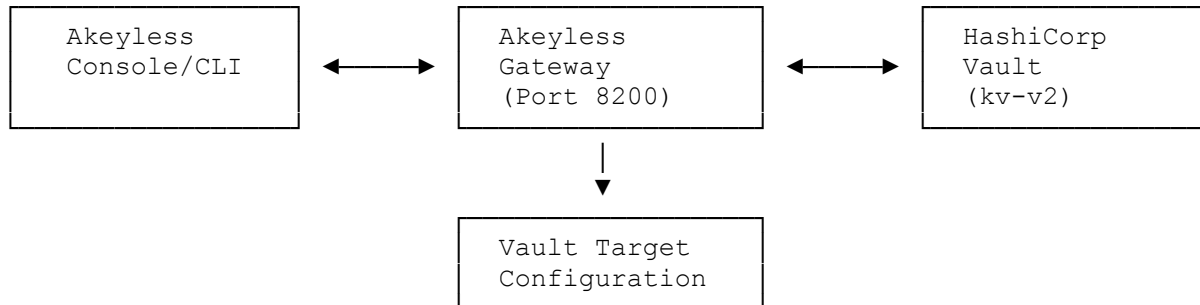
## **Executive Summary**

Akeyless offers a Universal Secrets Connector (USC) that can integrate with your existing HashiCorp Vault deployment, providing a "manager of managers" approach. This allows you to maintain your current Vault infrastructure while gaining the benefits of Akeyless's SaaS platform. Additionally, Akeyless provides a HashiCorp Vault Proxy (HVP) feature that offers API compatibility with Vault OSS, enabling seamless migration paths.

## **Architecture & Technical Implementation**

## 1. Universal Secrets Connector (USC) Architecture

The USC creates a bidirectional bridge between Akeyless and your HashiCorp Vault instance:



## 2. Key Components

- **Akeyless Gateway:** Lightweight, stateless component deployed in your private network that handles all secret requests
- **HashiCorp Vault Target:** Holds an access token with permissions to create, delete, update, read and list secrets
- **Universal Secrets Connector:** Creates local "windows" into related services, effectively letting you manage them indirectly

## 3. Workflow Sequence

```
sequenceDiagram
    participant User
    participant Akeyless Console/CLI
    participant Akeyless Gateway
    participant Vault Target
    participant HashiCorp Vault

    User->>Akeyless Console/CLI: Request secret operation
    Akeyless Console/CLI->>Akeyless Gateway: Forward request
    Akeyless Gateway->>Vault Target: Authenticate & authorize
    Vault Target->>HashiCorp Vault: Execute operation
    HashiCorp Vault-->>Vault Target: Return result
    Vault Target-->>Akeyless Gateway: Forward response
    Akeyless Gateway-->>Akeyless Console/CLI: Return secret
    Akeyless Console/CLI-->>User: Display result
```

## Implementation Steps

### 1. Prerequisites

- An Akeyless Gateway with Read permission on the target associated with the USC
- HashiCorp Vault Target which holds an access token with permissions to create, delete, update, read and list secrets

- **Note:** The KV Secrets Engine v1 is not supported. Please use KV Secrets Engine v2

## 2. Setup Process

### Create HashiCorp Vault Target:

```
akeyless target create hashi-vault \  
  --name <target-name> \  
  --hashi-url 'https://<vault-api-url:8200>' \  
  --vault-token <access-token> \  
  --namespace <namespace-name>
```

### Create Universal Secrets Connector:

```
akeyless create-usc \  
  --usc-name <connector-name> \  
  --target-to-associate <target-name>
```

## 3. Operations Available

- List secrets: `akeyless usc list --usc-name <name>`
- Get secret: `akeyless usc get --usc-name <name> --secret-id <id>`
- Create secret: `akeyless usc create --usc-name <name> --secret-name <name> -  
-value <value>`
- Update secret: `akeyless usc update --usc-name <name> --secret-id <id> --  
value <value>`
- Delete secret: `akeyless usc delete --usc-name <name> --secret-id <id>`

## Key Benefits & Strengths

### 1. No Migration Required

The Universal Secrets Connector allows you to centralize governance and control of your secrets without the need to migrate existing secrets or to replace existing secret managers

### 2. Unified Management

USC acts as a manager of managers for all your secrets managers. It addresses scenarios where teams need to use native secrets managers due to specific requirements while still providing centralized management and oversight

### 3. Enterprise Adoption

- Trusted by Fortune 100 companies and industry leaders
- Strategic investment from Deutsche Bank's Corporate Venture Capital group
- In the past year, the number of DevOps professionals and Security practitioners using Akeyless has increased fivefold

## 4. Scalability & Performance

- Akeyless' gateways are stateless, lightweight, and scalable, requiring no additional licensing costs
- The scalability works for organizations of all sizes, so it can be used for a small team or a large enterprise
- Akeyless on-premise caching option presents an advantage if you are looking for immediate access to secrets with minimal latency

## 5. Security Architecture

- **Distributed Fragments Cryptography (DFC):** Ensures that no third party—including Akeyless—can ever access your secrets
- **Zero-Knowledge Architecture:** Akeyless cannot access your encryption keys

## 6. API Compatibility

Akeyless developed API compatibility with HashiCorp Vault OSS, enabling the use of Vault OSS community plugins for both Static & Dynamic Secrets

# Considerations & Limitations

## 1. Technical Limitations

- The KV Secrets Engine v1 is not supported
- USC operations require prefix commands (e.g., `usc list` instead of just `list`)
- Some Vault Enterprise features may not have direct equivalents

## 2. Operational Considerations

- Requires Akeyless Gateway deployment in your infrastructure
- Additional authentication layer between users and Vault
- Potential latency for secret access (though mitigated by caching)

## 3. Commercial Considerations

- Akeyless' support is limited to paid customers only
- Closed-source product with limited community resources
- Requires going through sales process for advanced features

# Industry Use Cases & Adoption

## 1. Financial Services

- Strategic investment from Deutsche Bank demonstrates confidence in financial services applications
- Used by Fortune 100 companies including those in regulated industries

## 2. Customer Success Stories

- **Cimpress:** "Akeyless' platform approach, superb technology and service excellence made it easy for us to decide to rip and replace Hashi Vault"
- **Progress:** "We saved 70% of our maintenance and provisioning time with Akeyless"
- **Wix:** "Akeyless revolutionized our approach to security, shifting our paradigm from trusted networks to zero-trust access"

## 3. Migration Support

Akeyless supports migration from the kv storage engine of versions 1 and 2. For v2 migrations, only the current version of a secret is imported

# Evaluation Criteria for PoC

## 1. Technical Validation

- **Performance Testing:** Measure latency for secret retrieval with/without caching
- **Scale Testing:** Validate handling of your current secret volume
- **Integration Testing:** Verify compatibility with your existing Vault plugins
- **API Compatibility:** Test critical Vault API endpoints through HVP

## 2. Security Assessment

- **DFC Technology Validation:** Understand fragment distribution and key security
- **Audit Trail Completeness:** Compare with current Vault audit capabilities
- **Compliance Mapping:** Ensure regulatory requirements are met
- **Access Control Granularity:** Test RBAC implementation

## 3. Operational Assessment

- **Gateway Deployment:** Evaluate infrastructure requirements
- **High Availability:** Test failover scenarios
- **Monitoring & Alerting:** Integration with existing tools
- **Backup & Recovery:** Validate disaster recovery procedures

## 4. Cost Analysis

- **Licensing Comparison:** Vault Enterprise vs Akeyless pricing
- **Infrastructure Costs:** Gateway deployment vs Vault cluster costs
- **Operational Savings:** Customers report 70% reduction in maintenance time

- **TCO over 3-5 years:** Include migration, training, and ongoing costs

## **Pragmatic Assessment**

### **Is This Useful or Just Novelty?**

#### **Useful for:**

- Organizations wanting to modernize secrets management without full migration
- Multi-cloud environments requiring unified secrets governance
- Teams struggling with Vault operational complexity
- Enterprises seeking to reduce infrastructure costs

#### **May Be Novelty If:**

- You have deep Vault expertise and customizations
- Your use case is purely on-premise with no cloud aspirations
- You require specific Vault Enterprise features not available in USC
- Regulatory requirements mandate full control of infrastructure

## **Recommended PoC Approach**

1. **Phase 1 (2 weeks):** Deploy Akeyless Gateway and create Vault connector for non-production environment
2. **Phase 2 (2 weeks):** Test core operations and measure performance metrics
3. **Phase 3 (2 weeks):** Validate security controls and compliance requirements
4. **Phase 4 (2 weeks):** Pilot with select development team for real-world feedback
5. **Phase 5 (1 week):** Cost analysis and executive presentation

## **Conclusion**

The Akeyless Universal Secrets Connector presents a compelling option for banks looking to modernize their secrets management while preserving existing HashiCorp Vault investments. The solution is production-ready with strong enterprise adoption, particularly in financial services as evidenced by Deutsche Bank's investment.

The "manager of managers" approach allows for gradual migration and provides immediate benefits in terms of unified governance, reduced operational overhead, and enhanced security through DFC technology. However, careful evaluation of your specific Vault usage patterns, customizations, and regulatory requirements is essential before making a commitment.

For a bank already invested in HashiCorp Vault Enterprise, Akeyless USC offers a pragmatic path to modernization without the risks of a full platform migration, making it worthy of serious consideration through a structured PoC.



# Akeyless DFC Architecture: Customer Fragment and Key Fragment Storage

## Distributed Fragments Cryptography (DFC)

**DFC** is Akeyless's patented FIPS 140-2 certified technology that splits encryption keys into mathematical fragments that never combine, even during cryptographic operations. This creates a zero-knowledge architecture where no single entity (including Akeyless) can access the complete encryption key.

**Key Point:** The Customer Fragment ( $CF_1$ ) is stored within the Akeyless Gateway's process memory (not in a separate HSM), while other fragments ( $F_2$ ,  $F_3$ , etc.) are distributed across cloud providers. HSM integration is optional and provides additional protection for  $CF_1$  at rest.

## 1. DFC Conceptual Overview

Distributed Fragments Cryptography (DFC) Original Secret "myPassword123" or any data  
AES-256 Encryption Generate ephemeral key  $K$   
Encrypted Data  $C = \text{AES}(\text{Secret}, K)$   
Stored separately  
DFC Key Splitting Shamir Secret Sharing  
Customer Fragment  $CF_1$  Stored in Gateway process memory  
Key Fragment 2  $F_2$  Stored in AWS Region: us-east-1  
Key Fragment 3  $F_3$  Stored in Azure Region: europe-west  
Key Fragment  $N F_n$  Stored in GCP Region: asia-pacific  
Mathematical Property:  $K = f(CF_1, F_2, F_3)$  where any 2 of 3 fragments can reconstruct  $K$   
Fragments never combine - operations performed distributedly

## 2. Gateway Internal Architecture & CF Storage

Akeyless Gateway Internal Architecture  
Gateway Process (Docker/K8s)  
In-Memory Storage Customer Fragment ( $CF_1$ )  
Loaded at startup  
Encrypted in memory  
Local Disk Cache Encrypted secret cache  
Uses  $CF_1$  for encryption  
Gateway Services • API Service (port 8000) • Vault-compatible API

(port 8200)• Cluster sync (port 8001)Customer Fragment Input SourcesDirect  
InputEnvironment var:CUSTOMER\_FRAGMENT=...Kubernetes SecretMounted  
as volume/secrets/cf/fragmentFile MountDocker volume-v  
/secure/cf:/cfConfiguration APISet via Gateway UIPort 8000Optional HSM  
IntegrationWhen HSM is configured:1. CF<sub>1</sub> is encrypted by HSM master key2.  
Gateway retrieves encrypted CF<sub>1</sub> at startup3. HSM decrypts CF<sub>1</sub> for Gateway  
useHSM protects CF<sub>1</sub>

### 3. Fragment Storage Architecture

Customer EnvironmentAkeyless SaaS & Cloud ProvidersAkeyless Gateway  
ClusterGateway 1Region: USCF<sub>1</sub> stored locallyMemory/DiskGateway 2Region:  
EUCF<sub>1</sub> stored locallyMemory/DiskGateway NRegion: APCF<sub>1</sub>  
storedlocallyCustomer Fragment (CF<sub>1</sub>) StorageStored within each Gateway  
instanceShared across cluster via secure configNever transmitted to Akeyless  
SaaSCF<sub>1</sub> Input MethodsHow to provide CF<sub>1</sub> to Gateway:• Environment  
Variable• K8s Secret• Docker Volume Mount• Direct ConfigurationOptional  
HSM ProtectionEnhanced Security:• Luna HSM• Thales HSM•  
CloudHSMEncrypts CF<sub>1</sub> at restZero-Knowledge Properties✓ CF<sub>1</sub> stored only in  
Gateway process memory/disk✓ Without CF<sub>1</sub>, Akeyless cannot decrypt any  
secrets✓ CF<sub>1</sub> acts as master key for all secrets in the account✓ Loss of CF<sub>1</sub> =  
permanent loss of all secrets✓ Different CF<sub>1</sub> per account/environment for  
isolation✓ HSM can protect CF<sub>1</sub> but Gateway still needs accessAkeyless  
Control Plane• Metadata management• Access control & policies• Fragment  
orchestration• Audit loggingFragment Storage ServicesManages F<sub>2</sub>, F<sub>3</sub>, ... F<sub>n</sub>  
fragmentsNever receives or stores CF<sub>1</sub>AWS S3Fragment F<sub>2</sub>Region: us-east-  
1Encrypted at restAzure BlobFragment F<sub>3</sub>Region: europe-westEncrypted at  
restGCP StorageFragment F<sub>4</sub>Region: asia-pacificEncrypted at restData  
Residency Controls• EU data → Fragments only in EU regions• US data →  
Fragments only in US regions• Configurable per secret/folderFragment  
Security• Each fragment individually encrypted• Fragments distributed across  
providers for resilienceHTTPS/TLS

### 4. DFC Operations Flow

Write Operation1. Input Secret"myPassword123"2. Generate Key KEphemeral AES-2563. EncryptC = AES(Secret, K)4. Split Key KK  $\rightarrow$  (CF<sub>1</sub>, F<sub>2</sub>, F<sub>3</sub>)5. Store CF<sub>1</sub>In Gateway Memory6. Send F<sub>2</sub>, F<sub>3</sub>, CTo Akeyless SaaS Secret Stored!Read Operation1. Request Secretkey = "mySecret"2. Fetch F<sub>2</sub>, F<sub>3</sub>, CFFrom Akeyless3. Retrieve CF<sub>1</sub>From Gateway memory4. Reconstruct KK = f(CF<sub>1</sub>, F<sub>2</sub>, F<sub>3</sub>)5. DecryptSecret = AES<sup>-1</sup>(C, K)6. Clear KEphemeral key deleted"myPassword123"

**DFC Mathematical Properties**

**Shamir's Secret Sharing**  
 (Example: 2-of-3)  
 • Secret S is split into n shares where k shares reconstruct S  
 • Uses polynomial interpolation over finite field  
 •  $f(x) = a_0 + a_1x + a_2x^2$  where  $a_0 = S$  (the secret)  
 • Fragments: (1, f(1)), (2, f(2)), (3, f(3))

**Fragment Properties**  
 • Single fragment reveals nothing about K  
 • Any 2 fragments can reconstruct K (in 2-of-3 scheme)  
 • Computationally infeasible to derive K from single fragment  
 • Fragments can be verified without reconstruction

**Security Guarantees**  
 ✓ Information-theoretic security (not just computational)  
 ✓ Perfect secrecy with proper threshold  
 ✓ Resistant to quantum computing attacks

**Operational Benefits**  
 ✓ Fragments can be refreshed without changing secret  
 ✓ Geographic distribution for compliance  
 ✓ No single point of failure

**Key Difference from Traditional Encryption**  
 Traditional: Complete key exists somewhere | DFC: Complete key never exists in any single location

## Critical Security Considerations

- **Customer Fragment Storage:** CF<sub>1</sub> is stored within each Gateway instance's memory and optionally on disk (encrypted). It is NOT stored in a separate HSM by default.
- **HSM Integration (Optional):** When using HSM (like Luna), the HSM encrypts/protects the CF<sub>1</sub> at rest, but the Gateway must still load CF<sub>1</sub> into its process memory to perform operations.
- **CF<sub>1</sub> Backup Strategy:** Customers must implement secure backup procedures for CF<sub>1</sub>. This can include:
  - Encrypted backups in secure storage
  - HSM-protected key wrapping
  - Split knowledge procedures for recovery
- **Gateway Security:** Since CF<sub>1</sub> resides in Gateway memory, securing the Gateway infrastructure is critical. This includes:
  - Network isolation (private subnets only)

- Restricted access to Gateway hosts
  - Encrypted storage volumes
  - Regular security patching
- **Loss Prevention:** Loss of  $CF_1$  means permanent loss of access to all secrets. There is no recovery mechanism or backdoor.

## Legend

Customer Fragment ( $CF_1$ ) - Never leaves customer environment

Key Fragments ( $F_2, F_3, F_n$ ) - Stored by Akeyless across cloud providers

DFC Operations - Mathematical splitting/combining

Customer-controlled components

Akeyless-controlled components

## 5. GDPR and Data Residency with DFC

EU Region  
EU Customer Data Secret: "GDPR-Protected-Data"  
Requirement: Data stays in EU  
 $CF_1$ : In EU Gateway Memory  
 $F_2$  AWS Frankfurt  
 $F_3$  Azure Amsterdam  
Europe-west-4  
Metadata Policy  
residency: "EU"  
allowed\_regions: ["eu-\*"]  
GDPR Compliant  
✓ All fragments in EU  
✓  $CF_1$  never leaves premises  
US Region  
US Customer Data Secret: "ITAR-Controlled-Data"  
Requirement: Data stays in US  
 $CF_1$ : In US Gateway Memory  
 $F_2$  AWS Virginia  
 $F_3$  Azure Texas  
us-south-1  
Metadata Policy  
residency: "US"  
allowed\_regions: ["us-\*"]  
ITAR/EAR Compliant  
✓ All fragments in US  
✓ No foreign access  
Global Control Plane  
Akeyless SaaS  
• Metadata only (no secrets)  
• Access policies  
• Fragment location mapping  
Zero-Knowledge Guarantee  
Without  $CF_1$ :  
• Cannot decrypt EU data  
• Cannot decrypt US data  
• Cannot access any secrets  
Data Flow Control  
✓ EU data never enters US systems  
✓ US data never enters EU systems  
✓ Cryptographic enforcement  
✓ Audit trail by region

### DFC Mathematical Foundation

Secret Sharing:  $S \rightarrow \{F_1, F_2, \dots, F_n\}$  where  $k$  fragments reconstruct  $S$

Threshold:  $(k, n)$  scheme - need  $k$  out of  $n$  fragments

Example (2,3): Any 2 of  $\{CF_1, F_2, F_3\}$  can reconstruct key  $K$

Security:  $\Pr[\text{guess } S \mid F_1] = \Pr[\text{guess } S]$  (perfect secrecy)

# Akeyless Global Secrets Management: Security Architecture and Compliance Analysis

Akeyless represents a paradigm shift in secrets management through its patented Distributed Fragments Cryptography™ (DFC) technology, offering a vaultless, zero-knowledge architecture that enables secure global deployments while maintaining strict data sovereignty controls. This comprehensive analysis evaluates Akeyless as a global secrets management solution across security, compliance, and architectural dimensions.

## Revolutionary cryptographic architecture eliminates traditional vault limitations

Akeyless's core innovation lies in its **Distributed Fragments Cryptography (DFC)** technology, a FIPS 140-2 certified approach that fundamentally reimagines how encryption keys are managed globally. Unlike traditional systems that replicate complete vaults across regions, DFC splits encryption keys into mathematical fragments distributed across multiple geographic locations and cloud providers (AWS, Azure, GCP).

The breakthrough is that **these fragments never combine, even during cryptographic operations**. Each customer maintains exclusive control through a "Customer Fragment" stored in their environment, creating a zero-knowledge architecture where even Akeyless cannot access customer secrets. This approach provides inherent air-gapped protection while enabling true global scalability through a unified SaaS control plane.

## Data localization meets global scale through hybrid architecture

Akeyless addresses the critical challenge of balancing global accessibility with local data residency requirements through sophisticated deployment models:

**Geographic Data Distribution:** The platform automatically ensures data sovereignty by storing fragments within appropriate continental boundaries - US customers receive fragments in US regions, German customers in German regions, with similar localization available globally. This approach satisfies GDPR, Schrems II, and other regional privacy regulations without sacrificing operational efficiency.

**Gateway Architecture:** Stateless Docker containers called Akeyless Gateways serve as SaaS extensions within customer environments. These gateways hold the Customer Fragment and provide transparent internal operations without requiring network infrastructure changes. They support both standalone and Kubernetes deployments with **geo-clustering capabilities** for high availability across regions.

**Deployment Flexibility:** Organizations can choose between Pure SaaS for rapid deployment with minimal overhead, or Hybrid SaaS combining cloud benefits with on-premises gateways for maximum compliance control. The hybrid model enables 70% operational cost reduction while maintaining strict data residency requirements.

## Thales CipherTrust integration creates enterprise-grade unified platform

The partnership between Akeyless and Thales CipherTrust delivers **CipherTrust Secrets Management (CSM)**, integrating Akeyless's cloud-native secrets management with Thales's enterprise data security infrastructure. This integration, officially launched in April 2023, addresses enterprise requirements for unified secrets and key management.

**Technical Architecture:** The Akeyless Gateway Service runs directly within CipherTrust Manager software, mediating authentication through JSON Web Tokens (JWTs) with ECDSA signing key pairs. CipherTrust Manager generates customer fragments for secret encryption, protected by its root of trust hierarchy with optional HSM integration. This creates a **zero-knowledge architecture where all secret access must flow through CipherTrust Manager**.

**Enterprise Benefits:** The integration supports comprehensive secret types including static secrets, dynamic credentials, rotated secrets, encryption keys, certificates, and SSH keys. Organizations like Cimpress have achieved **70% Total Cost of Ownership reduction** and **270% increase in platform adoption** through this unified approach.

**Implementation Considerations:** Setup requires CipherTrust Manager v2.13.1+ with public internet connectivity to Akeyless SaaS services. The platform supports automatic customer fragment generation, SSO integration, and fine-grained access controls, though customer fragment changes require cluster-wide gateway restarts.

## Industry-leading compliance posture supports global enterprise requirements

Akeyless maintains comprehensive compliance certifications essential for enterprise global deployments:

**Core Certifications:** SOC 2 Type II, ISO 27001, ISO 27701, PCI DSS, and NIST FIPS 140-2 validation specifically for DFC technology. The platform also supports GDPR, CCPA, and HIPAA-ready configurations with technical and administrative safeguards.

**Security Architecture:** Multi-cloud infrastructure spans AWS, GCP, and Azure with redundant systems. All data uses TLS 1.2+ for transit encryption and AES/RSA encryption at rest. The **zero-trust architecture** verifies every access request, while comprehensive audit trails support SIEM integration and real-time monitoring.

**Governance Framework:** Board-level security oversight with dedicated CISO leadership, formal risk management programs with annual reviews, and mandatory security training covering regulatory compliance, data protection, and threat awareness. Physical security leverages tier-1 cloud providers' biometric access controls, security cameras, and redundant environmental systems.

## Multi-region performance optimized for global operations

Akeyless's vaultless architecture delivers significant performance advantages over traditional vault-based systems requiring regional clusters:

**Latency Optimization:** Gateway-local caching provides sub-millisecond access to frequently used secrets, while geolocation-based routing policies direct traffic to optimal regional endpoints. The **stateless gateway design** enables instant failover and reduces operational latency compared to traditional multi-cluster vault architectures.

**Replication Model:** Rather than replicating complete vaults, the platform uses eventually consistent metadata replication with strong consistency for cryptographic operations through mathematical fragment operations. Dynamic fragment refresh continuously changes mathematical values without breaking overall functionality, providing additional security.

**Scalability Characteristics:** True SaaS backend automatically handles load spikes without provisioning delays, while horizontal gateway scaling supports global distribution. Performance testing shows the architecture eliminates traditional vault performance replication latency between regional clusters.

## Competitive advantages validate market leadership position

Industry analysis positions Akeyless as the **#1 alternative to HashiCorp Vault**, with recognition as "Overall Leader" in KuppingerCole's 2025 Leadership Compass for Enterprise Secrets Management.

**Quantified Benefits:** Customer implementations demonstrate 70% Total Cost of Ownership reduction, 60-70% reduction in maintenance time, and 270% increase in platform adoption rates. Fortune 500 companies including 2 of the Fortune 5 and multiple global pharmaceutical companies have adopted the platform.

**Architectural Superiority:** Unlike traditional solutions requiring dedicated infrastructure per region, Akeyless's SaaS model provides instant global scaling without regional vault clusters. This eliminates operational complexity while maintaining security through DFC technology that ensures secrets never exist as complete entities in any single location.

**Market Positioning:** The platform addresses key limitations of competitors - HashiCorp Vault's operational complexity, AWS Secrets Manager's vendor lock-in, Azure Key Vault's Microsoft-centric approach, and CyberArk's traditional PAM limitations for modern DevOps environments.

## Global deployment best practices enable enterprise success

Successful global implementations follow proven architectural and operational patterns:

**Network Architecture:** Deploy gateways in private subnets within each major operating region, ensuring outbound HTTPS (443) connectivity to Akeyless services while blocking inbound public access. Gateway clusters require minimal resources (1 vCPU, 2GB RAM minimum) with Kubernetes-native deployment using Helm charts.

**Security Configuration:** Implement hierarchical folder structures reflecting organizational units, enable MFA for administrative accounts, use principle of least privilege for dynamic secrets, and configure comprehensive audit logging with SIEM integration.

**Operational Excellence:** Leverage automatic secrets migration from existing platforms, implement automated rotation for all credential types, use dynamic secrets for just-in-time access, and establish monitoring across global operations.

## Key limitations require careful architectural planning

While Akeyless provides significant advantages, certain constraints must be considered for global deployments:

**Connectivity Dependencies:** Each location requires Gateway deployments with outbound HTTPS connectivity to Akeyless SaaS services. Gateway maintenance requires Docker/Kubernetes infrastructure management, though operational overhead remains minimal compared to traditional vault architectures.

**Customer Fragment Management:** Organizations bear responsibility for securing and backing up customer fragments, which provide the zero-knowledge guarantee but require proper lifecycle management across global deployments.



**Network Segmentation:** Gateways should not be globally accessible from public networks, requiring proper internal load balancer and security group configurations in distributed environments.

## Comprehensive Security and Compliance Evaluation Checklist

### Architecture and Design

- ☐ **Zero-Knowledge Verification:** Confirm customer fragment storage exclusively in customer environment
- ☐ **Fragment Distribution:** Validate geographic distribution of key fragments across regions and cloud providers
- ☐ **Encryption Standards:** Verify FIPS 140-2 certification for DFC technology and TLS 1.2+ for all communications
- ☐ **Gateway Architecture:** Assess stateless gateway design and clustering capabilities for high availability
- ☐ **Multi-Cloud Support:** Evaluate platform support for AWS, Azure, GCP, and on-premises environments

### Compliance and Certifications

- ☐ **Core Certifications:** Verify current SOC 2 Type II, ISO 27001, ISO 27701, and PCI DSS certifications
- ☐ **Regional Compliance:** Confirm GDPR, CCPA, and applicable national privacy law compliance
- ☐ **Industry-Specific:** Assess HIPAA-ready configurations if applicable to your industry
- ☐ **Audit Capabilities:** Evaluate comprehensive audit trails and SIEM integration capabilities
- ☐ **Data Residency:** Verify fragment storage within required geographic boundaries

### Security Controls

- ☐ **Access Control:** Review role-based access control (RBAC) and multi-factor authentication implementation
- ☐ **Identity Integration:** Assess SSO integration with existing identity providers (SAML, OIDC, LDAP)
- ☐ **Network Security:** Evaluate zero-trust architecture and network segmentation requirements
- ☐ **Vulnerability Management:** Review bug bounty program and third-party security assessments
- ☐ **Incident Response:** Assess 24-hour notification procedures and incident response capabilities

### Global Deployment Readiness

- ☐ **Multi-Region Architecture:** Evaluate gateway clustering and geo-distribution capabilities
- ☐ **Performance Requirements:** Assess latency characteristics and caching capabilities for global operations
- ☐ **Scalability:** Review auto-scaling capabilities and resource requirements for global deployments
- ☐ **Data Sovereignty:** Confirm data localization controls meet regulatory requirements
- ☐ **Disaster Recovery:** Evaluate cross-region failover and business continuity procedures

## Integration and Compatibility

- ☐ **Thales CipherTrust:** If applicable, assess CipherTrust Manager integration requirements and capabilities
- ☐ **DevOps Integration:** Evaluate CI/CD pipeline integration and developer tool compatibility
- ☐ **API Security:** Review comprehensive API security with rate limiting and authentication controls
- ☐ **Legacy System Support:** Assess migration capabilities from existing secrets management platforms
- ☐ **Third-Party Tools:** Verify integration with required security and monitoring tools

## Operational Considerations

- ☐ **Maintenance Overhead:** Evaluate operational requirements for gateway management and updates
- ☐ **Cost Structure:** Assess total cost of ownership compared to existing solutions
- ☐ **Training Requirements:** Evaluate team training needs for platform operation
- ☐ **Support Model:** Review available support levels and response times for global operations
- ☐ **Migration Strategy:** Assess automated migration capabilities and transition planning

## Risk Assessment

- ☐ **Single Points of Failure:** Evaluate potential failure modes and mitigation strategies
- ☐ **Vendor Dependency:** Assess risks associated with SaaS model and vendor relationship
- ☐ **Network Dependencies:** Review connectivity requirements and potential network-related risks
- ☐ **Compliance Drift:** Evaluate ongoing compliance maintenance and monitoring procedures
- ☐ **Emergency Access:** Review emergency access procedures and disaster recovery capabilities

## Performance and Monitoring

- [ ] **Monitoring Integration:** Assess integration with existing monitoring and alerting systems
- [ ] **Performance Metrics:** Review available performance metrics and SLA commitments
- [ ] **Capacity Planning:** Evaluate capacity planning requirements for global growth
- [ ] **Health Checks:** Assess health monitoring and automated failover capabilities
- [ ] **Troubleshooting:** Review diagnostic capabilities and support procedures

This comprehensive evaluation framework enables organizations to thoroughly assess Akeyless's suitability for global secrets management requirements while ensuring security, compliance, and operational objectives are met across distributed environments.

Based on my research, Akeyless provides a fundamentally different architecture than MySQL replication that offers significant advantages for global key/value operations while maintaining GDPR compliance. Here's how it matches your requirements:

## Cross-Region Access Architecture

**Yes, you can write a key/value in AWS us-east-1 and immediately retrieve it in uk-west-2.** However, Akeyless achieves this differently than MySQL replication:

### How It Works

1. **Unified Global Control Plane:** Instead of replicating data between regions, Akeyless uses a single SaaS control plane that manages metadata globally. When you write a secret in us-east-1, the metadata is instantly available worldwide.
2. **Distributed Fragments Cryptography (DFC):** The actual secret value is encrypted and split into mathematical fragments distributed across multiple regions. These fragments never combine, even during decryption operations.
3. **Gateway-Based Access:** You deploy lightweight Akeyless Gateways in each region (us-east-1, uk-west-2, etc.). These gateways:
  - Cache frequently accessed secrets locally for sub-millisecond access
  - Hold your "Customer Fragment" required for decryption
  - Communicate with the global Akeyless SaaS to retrieve secrets on-demand

### Key Advantages Over MySQL Replication

- **No replication lag:** Changes are immediately available globally
- **Lower operational overhead:** No need to manage MySQL clusters and replication
- **Better security:** Secrets never exist in complete form in any database
- **Automatic scaling:** SaaS backend handles load without manual intervention

## GDPR Compliance Handling

Akeyless provides sophisticated GDPR compliance through its architecture:

## Data Residency Controls

1. **Geographic Fragment Distribution:**
  - EU customer data fragments are stored exclusively within EU regions
  - UK data can be restricted to UK regions post-Brexit
  - You maintain complete control over where your Customer Fragment resides
2. **Zero-Knowledge Architecture:**
  - Even Akeyless cannot access your secrets without your Customer Fragment
  - This satisfies GDPR's encryption requirements for data transfers
3. **Compliance Features:**
  - **Data localization policies:** Enforce that specific secrets/folders only store fragments in approved regions
  - **Audit trails:** Complete audit logs showing who accessed what data and when
  - **Right to deletion:** Immediate secret deletion with cryptographic guarantees
  - **Data portability:** Export capabilities for GDPR Article 20 compliance

## Practical GDPR Implementation

For your use case, you could:

1. Deploy Gateways in both us-east-1 and uk-west-2
2. Configure EU customer secrets to use EU-only fragment storage
3. Set access policies ensuring EU data is only accessed by authorized EU-based services
4. Enable comprehensive audit logging for compliance reporting

## Comparison to MySQL Replication for GDPR

| Aspect                 | MySQL Replication                        | Akeyless                                 |
|------------------------|--|--|
| Data Residency         | Full data copies in each region          | Only fragments, with geographic controls |
| Cross-border transfers | Requires careful configuration           | Built-in compliance through DFC          |
| Audit complexity       | Must aggregate logs from all regions     | Centralized audit trail                  |
| Deletion compliance    | Must ensure deletion across all replicas | Cryptographic deletion guarantee         |
| Encryption at rest     | Database-level encryption                | Fragment-level with zero-knowledge       |

## Migration Considerations

Moving from MySQL replication to Akeyless would involve:

1. **Automated migration tools** to transfer existing key/value pairs
2. **Gateway deployment** in your existing regions

3. **Application integration** using Akeyless SDKs or REST APIs
4. **Gradual transition** - you can run both systems in parallel during migration

The platform supports the same key/value operations you're used to, but with enhanced security, compliance, and significantly reduced operational overhead compared to managing global MySQL replication.

Would you like me to elaborate on any specific aspect of how Akeyless would handle your particular use case or GDPR requirements?

---

Based on my research, Akeyless provides a fundamentally different architecture than MySQL replication that offers significant advantages for global key/value operations while maintaining GDPR compliance. Here's how it matches your requirements:

## Cross-Region Access Architecture

**Yes, you can write a key/value in AWS us-east-1 and immediately retrieve it in uk-west-2.** However, Akeyless achieves this differently than MySQL replication:

### How It Works

1. **Unified Global Control Plane:** Instead of replicating data between regions, Akeyless uses a single SaaS control plane that manages metadata globally. When you write a secret in us-east-1, the metadata is instantly available worldwide.
2. **Distributed Fragments Cryptography (DFC):** The actual secret value is encrypted and split into mathematical fragments distributed across multiple regions. These fragments never combine, even during decryption operations.
3. **Gateway-Based Access:** You deploy lightweight Akeyless Gateways in each region (us-east-1, uk-west-2, etc.). These gateways:
  - Cache frequently accessed secrets locally for sub-millisecond access
  - Hold your "Customer Fragment" required for decryption
  - Communicate with the global Akeyless SaaS to retrieve secrets on-demand

### Key Advantages Over MySQL Replication

- **No replication lag:** Changes are immediately available globally
- **Lower operational overhead:** No need to manage MySQL clusters and replication
- **Better security:** Secrets never exist in complete form in any database
- **Automatic scaling:** SaaS backend handles load without manual intervention

## GDPR Compliance Handling

Akeyless provides sophisticated GDPR compliance through its architecture:

## Data Residency Controls

1. **Geographic Fragment Distribution:**
  - EU customer data fragments are stored exclusively within EU regions
  - UK data can be restricted to UK regions post-Brexit
  - You maintain complete control over where your Customer Fragment resides
2. **Zero-Knowledge Architecture:**
  - Even Akeyless cannot access your secrets without your Customer Fragment
  - This satisfies GDPR's encryption requirements for data transfers
3. **Compliance Features:**
  - **Data localization policies:** Enforce that specific secrets/folders only store fragments in approved regions
  - **Audit trails:** Complete audit logs showing who accessed what data and when
  - **Right to deletion:** Immediate secret deletion with cryptographic guarantees
  - **Data portability:** Export capabilities for GDPR Article 20 compliance

## Practical GDPR Implementation

For your use case, you could:

1. Deploy Gateways in both us-east-1 and uk-west-2
2. Configure EU customer secrets to use EU-only fragment storage
3. Set access policies ensuring EU data is only accessed by authorized EU-based services
4. Enable comprehensive audit logging for compliance reporting

## Comparison to MySQL Replication for GDPR

| Aspect                 | MySQL Replication                        | Akeyless                                 |
|------------------------|--|--|
| Data Residency         | Full data copies in each region          | Only fragments, with geographic controls |
| Cross-border transfers | Requires careful configuration           | Built-in compliance through DFC          |
| Audit complexity       | Must aggregate logs from all regions     | Centralized audit trail                  |
| Deletion compliance    | Must ensure deletion across all replicas | Cryptographic deletion guarantee         |
| Encryption at rest     | Database-level encryption                | Fragment-level with zero-knowledge       |

## Migration Considerations

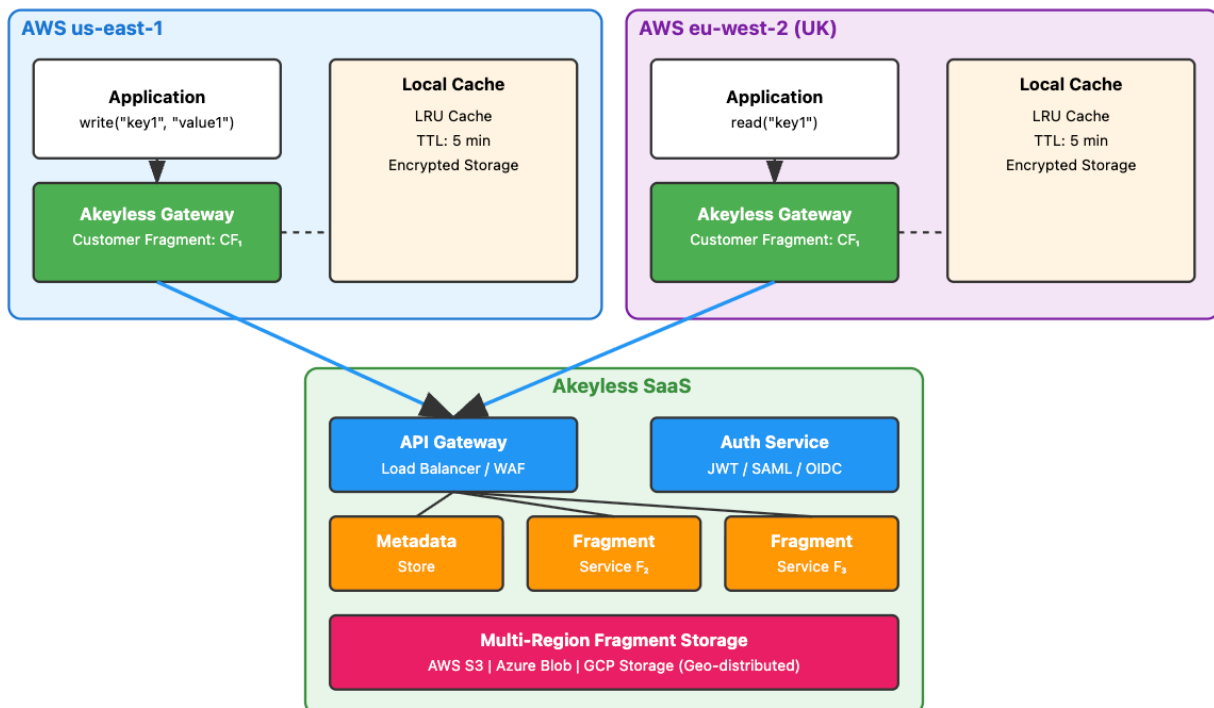
Moving from MySQL replication to Akeyless would involve:

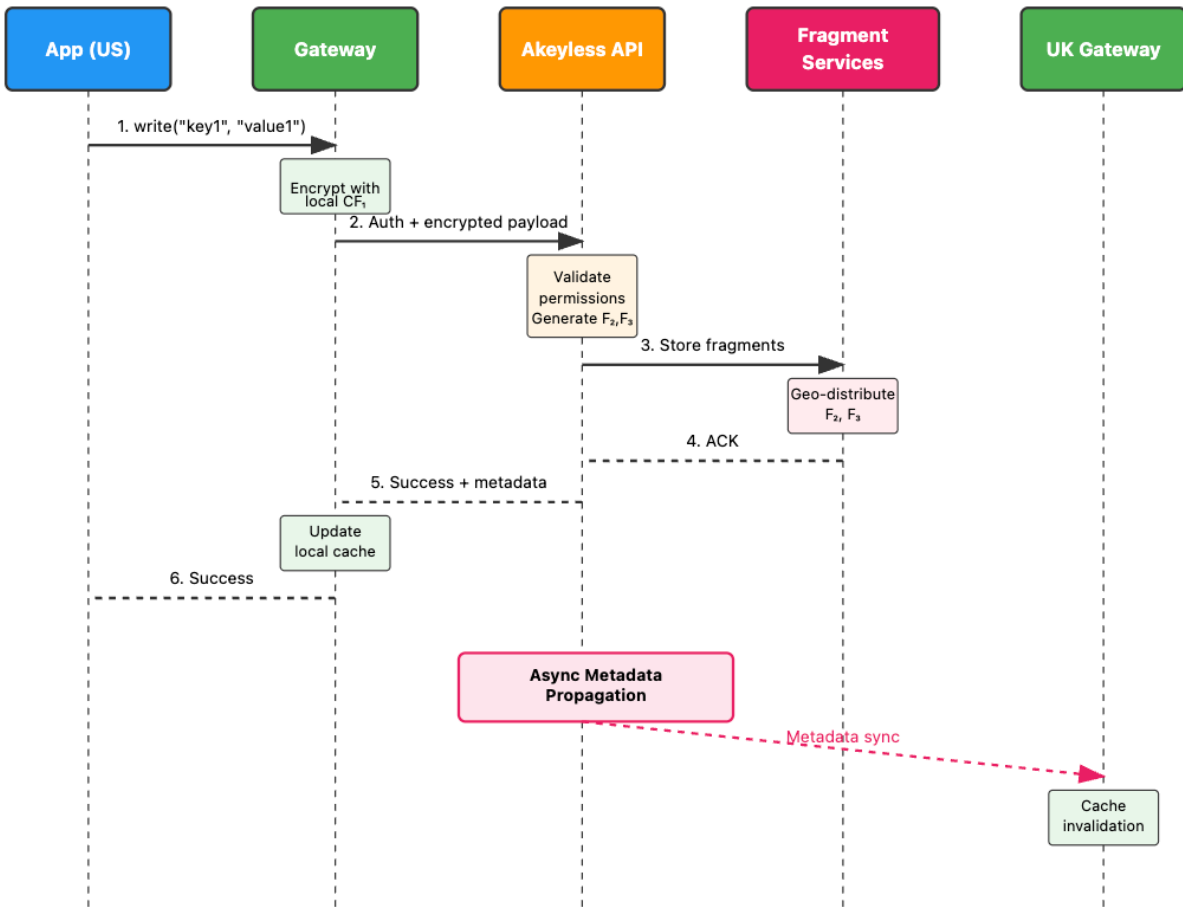
1. **Automated migration tools** to transfer existing key/value pairs
2. **Gateway deployment** in your existing regions

3. **Application integration** using Akeyless SDKs or REST APIs
4. **Gradual transition** - you can run both systems in parallel during migration

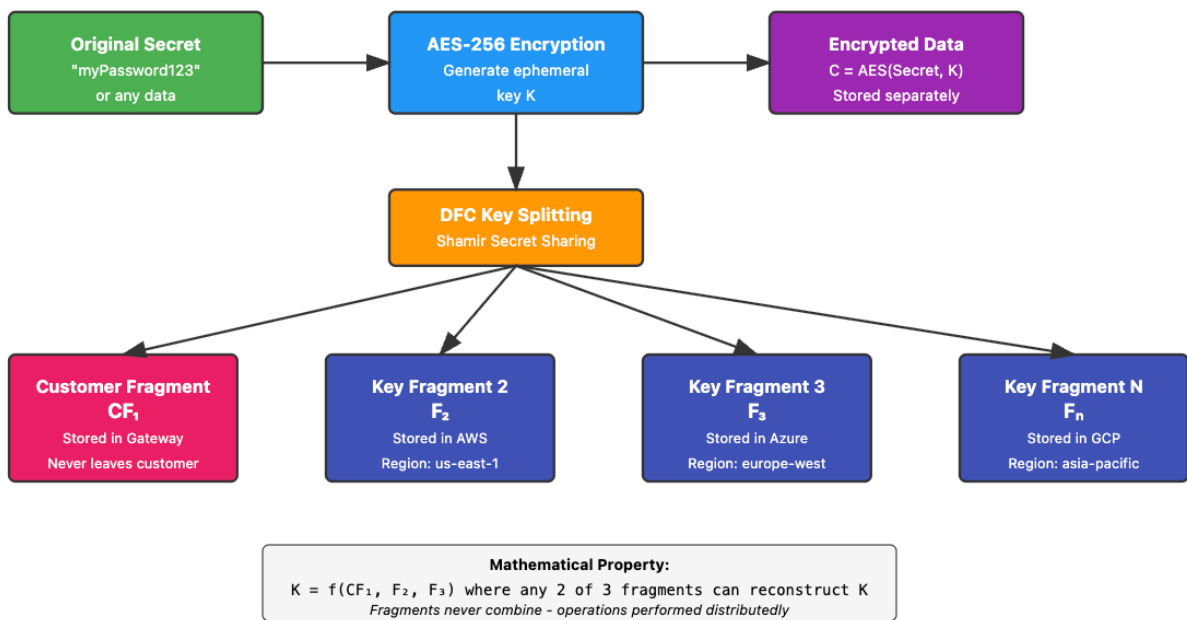
The platform supports the same key/value operations you're used to, but with enhanced security, compliance, and significantly reduced operational overhead compared to managing global MySQL replication.

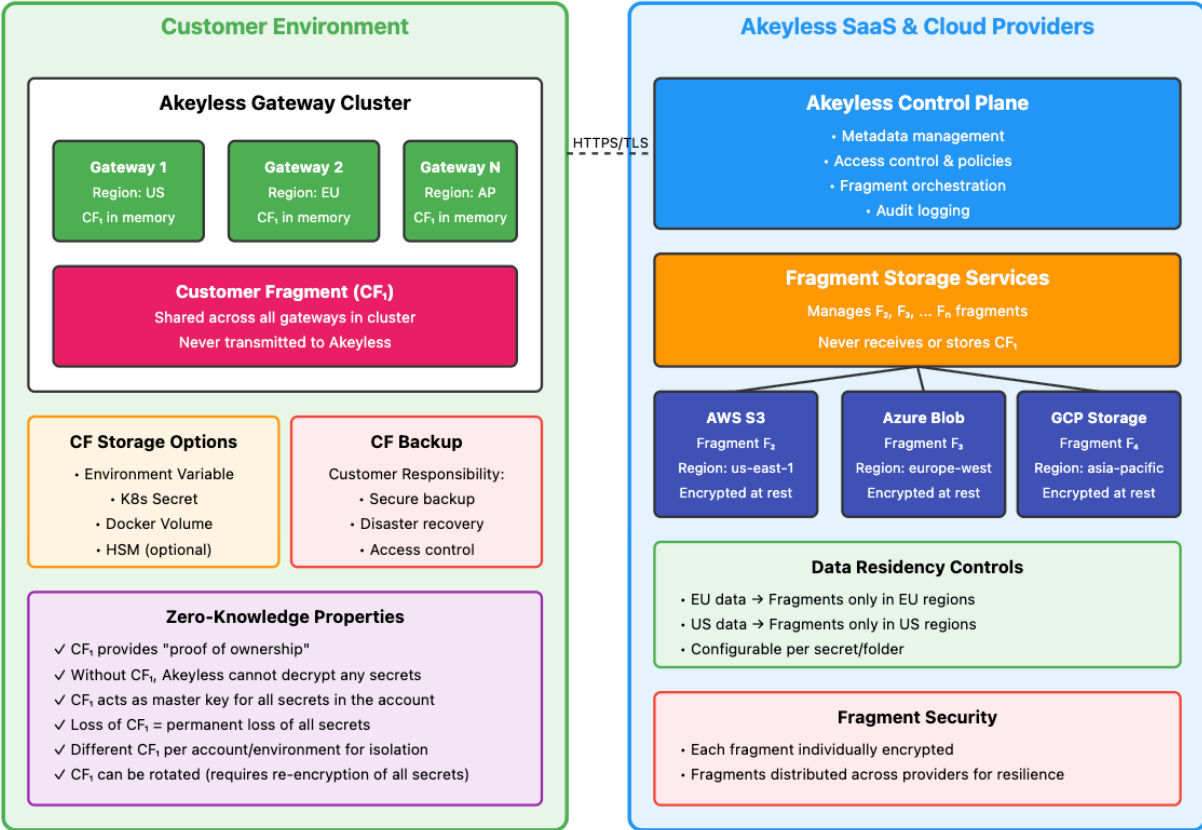
Would you like me to elaborate on any specific aspect of how Akeyless would handle your particular use case or GDPR requirements?











### Critical Security Considerations

- **Customer Fragment (CF<sub>i</sub>) is Critical:** Loss of CF<sub>i</sub> means permanent loss of access to all secrets. There is no recovery mechanism.
- **CF<sub>i</sub> Backup Strategy:** Customers must implement secure backup procedures for CF<sub>i</sub>, potentially using HSMs or secure key management systems.
- **Fragment Isolation:** CF<sub>i</sub> should never be stored in the same location as other fragments or exposed to Akeyless services.
- **Rotation Capability:** While CF<sub>i</sub> can be rotated, this requires re-encryption of all secrets with the new fragment set.

### Legend

- Customer Fragment (CF<sub>i</sub>) - Never leaves customer environment
- Key Fragments (F<sub>2</sub>, F<sub>3</sub>, F<sub>n</sub>) - Stored by Akeyless across cloud providers
- DFC Operations - Mathematical splitting/combining
- Customer-controlled components
- Akeyless-controlled components



## 4. GDPR and Data Residency with DFC

