Microsoft

AZ-203.4 Module 03: Implementing secure data solutions

Subtitle or speaker name

Microsoft

Topics

- Encryption options
- End-to-end encryption
- Implement Microsoft Azure confidential computing
- Manage cryptographic keys in Azure Key Vault
- Lab: Access resource secrets securely across services

Lesson 01: Encryption options

Microsoft Azure Security Spectrum

Identity & access	Encryption	Secure networking	Partner solutions	Unified security management
• Role-based access control (RBAC) • Strong authentication • Monitoring and alerting	• Encryption key management • Encryption at rest and in transit	• Virtual networks • Traffic rules • Secure connectivity	• Antimalware • Network appliances • Encryption • Monitoring • Application security • Authentication	• Security policy • Monitoring • Recommendations • Threat detection

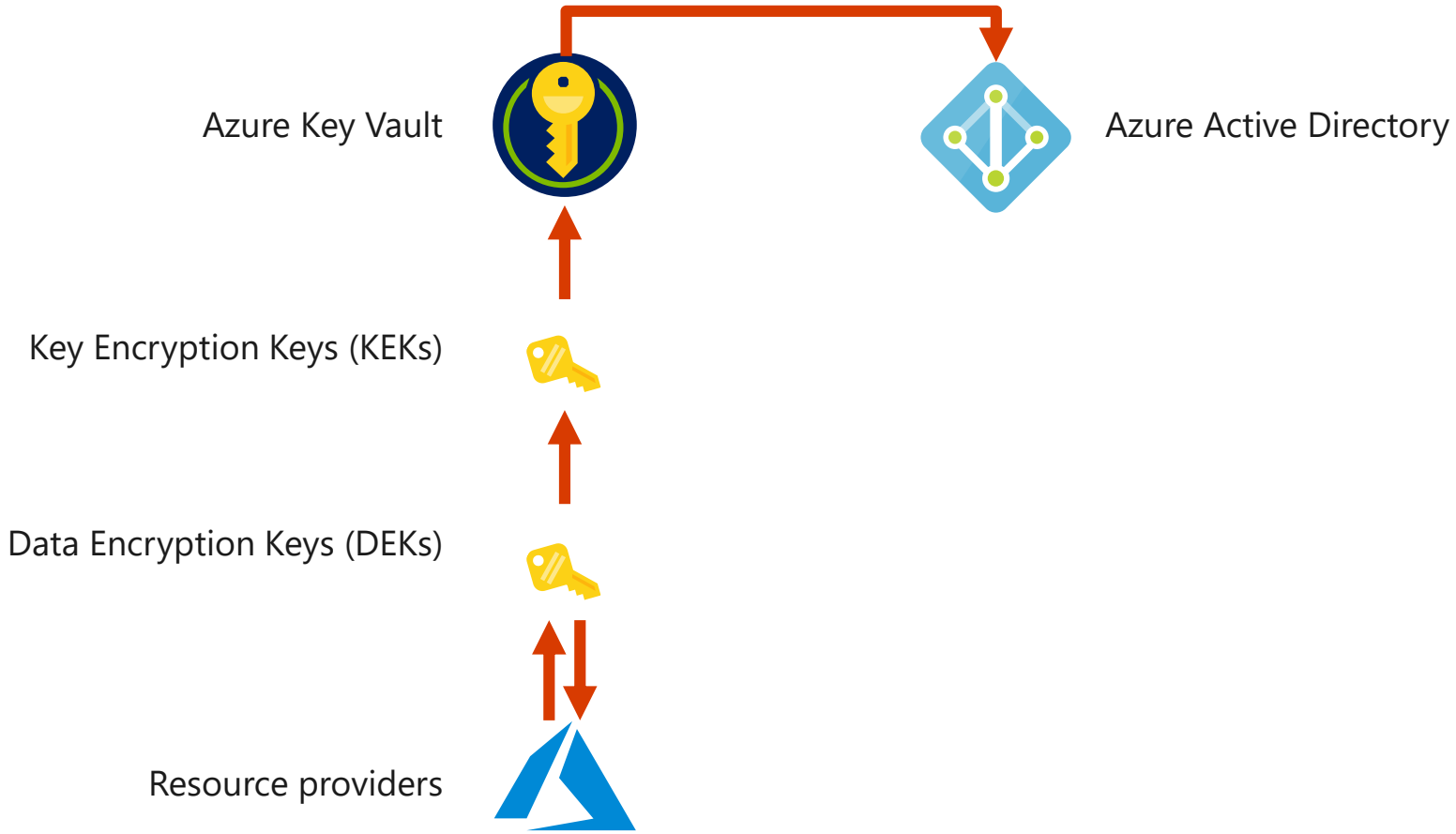
Encryption

- Encryption
  - Process of translating plain text data (**plaintext**) into something that appears to be random and meaningless (**ciphertext**)
- Decryption
  - Process of converting ciphertext back to plaintext
- Symmetric encryption is used to encrypt more than a small amount of data
  - A symmetric key is used to encrypt the data
  - The same key must be used to decrypt the data

Encryption at rest

- Encryption (or encoding) of data when it is persisted
  - Very common security requirement to encrypt data with a secret encryption key anytime it is persisted to disk
  - Prevents attackers from accessing sensitive data when they have full access to a server's machine, storage, or drives
- Encryption at rest design in Azure uses symmetric encryption:
  - A symmetric encryption key is used to encrypt data as it is written to storage
  - The same encryption key is used to decrypt that data as it is readied for use in memory
  - Data may be partitioned, and different keys may be used for each partition
  - Keys are stored in a security-enhanced location with access-control policies
  - Data encryption keys are often encrypted with asymmetric encryption to further limit access

Encryption at rest in Azure



Encryption at rest for Azure services

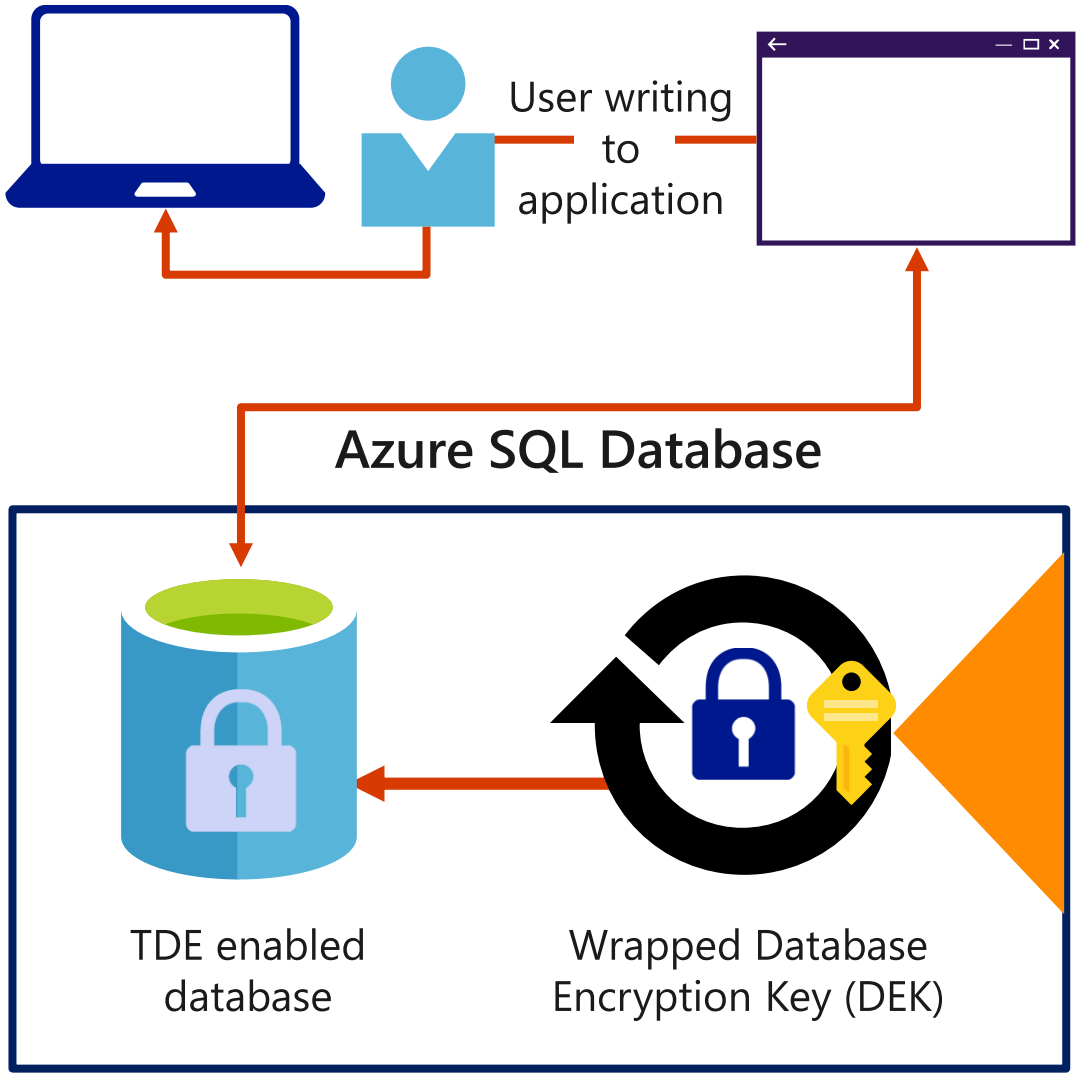
- Azure Storage 
  - Data is automatically encrypted server-side for all Storage services (Blob, Queue, Table, Files)
  - By default, keys are managed by the service
  - Supports customer-managed keys stored in Azure Key Vault
- Azure SQL Database 

  - Transparent Data Encryption (TDE) is enabled by default on all new databases
  - Supports customer-managed 2048-bit keys stored in Azure Key Vault
- Azure Cosmos DB 
  - Backups and media attachments are stored in Blob storage
  - Databases are automatically encrypted on solid-state drives (SSDs)

Lesson 02: End-to-end encryption

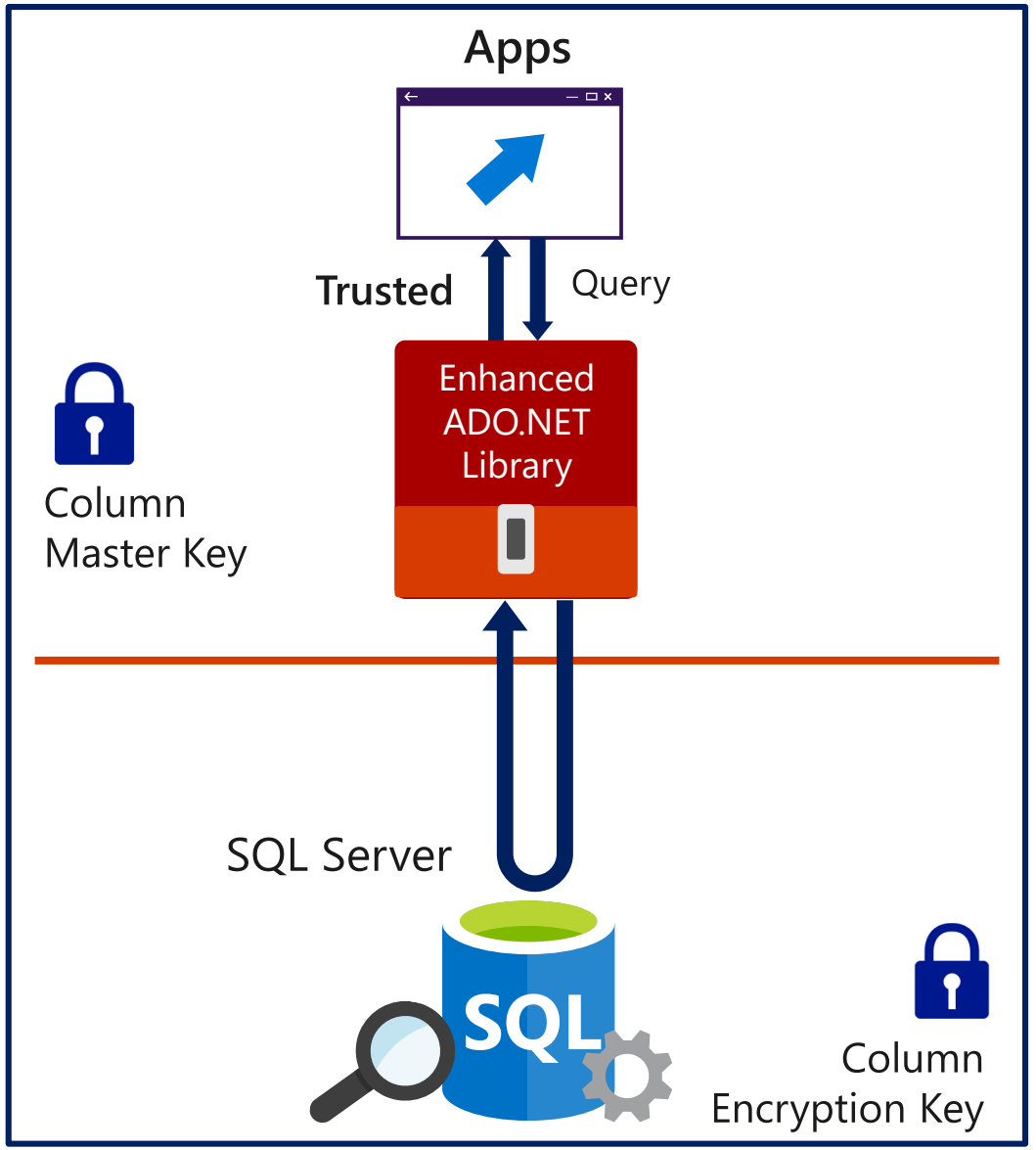
Transparent Data Encryption (TDE)

- Encrypts database, backups, and logs at rest and in flight
- Requires little to no code changes
  - Only requires a modification to connection string in most scenarios
- Can be used with many third-party SQL tools already in the market
- Supported in Microsoft Azure SQL Database, Azure SQL Data Warehouse, and SQL Server

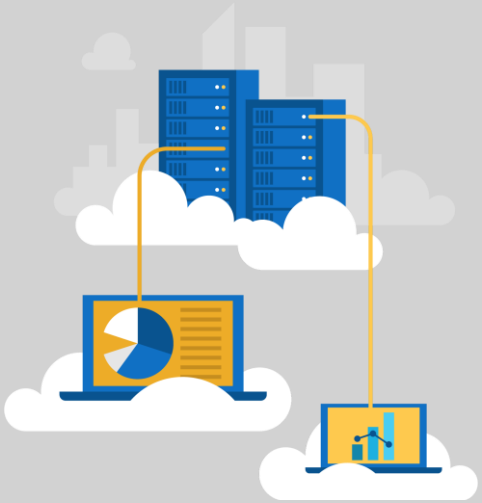


Always Encrypted

- Fully transparent encryption
  - Encrypted inside client applications
  - Encryption keys are not available to the database engine
- Encrypts data at rest, in flight, and in memory
- Requires the use of specific drivers
  - In most applications, requires some rewrites
  - Not compatible with every third-party tool



Lesson 03: Implement Azure confidential computing

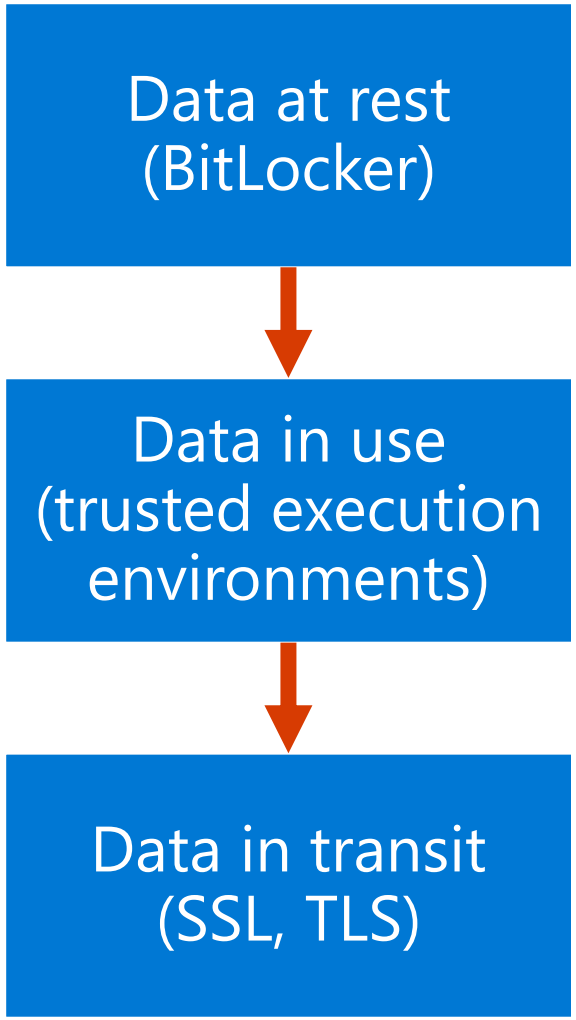


Trusted Execution Environments

- Data is processed within Trusted Execution Environments (TEEs)
  - Ensures that there is no way to view data or operations inside the TEE from the outside
  - If code is tampered with or altered, all operations are halted, and the environment is disabled
  - TEEs can be hardware based or software based
  - Many Azure services, such as Azure SQL Database, execute code in TEEs
  - There are frameworks available to take advantage of TEEs
    - Example: Confidential Consortium Blockchain Framework
- TEEs are being developed through collaboration among vendors:
  - Intel (SGX)
  - Microsoft Research

Azure confidential computing

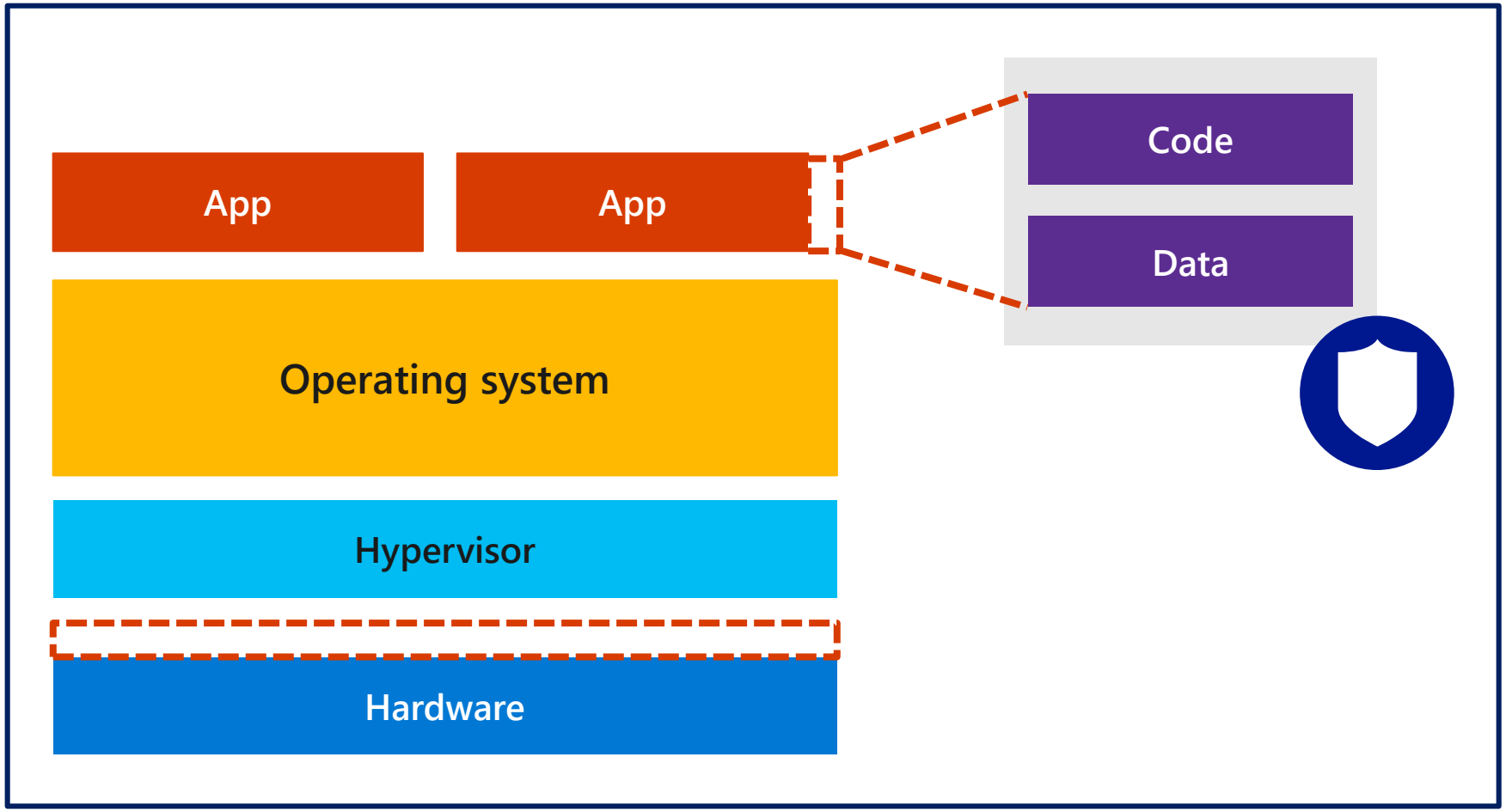
- Based on trusted execution enclaves:
  - Virtualization-based security
  - Intel® Software Guard Extensions (Intel® SGX)
- Secures all data while in use:
  - Workloads are invisible to host fabric
  - Data is now always encrypted
  - Protected while in use, in transit, and in storage
- Protects against multiple threats:
  - Malicious insiders, including admins, hackers and malware
  - Third-party access without consent



Azure confidential computing (continued 1)

- A collection of features across a broad spectrum of Azure services designed to encrypt data in use
- Ideal for scenarios where data needs to be processed in the cloud
  - The services maintain encryption that prevents the data from being exposed as plain text

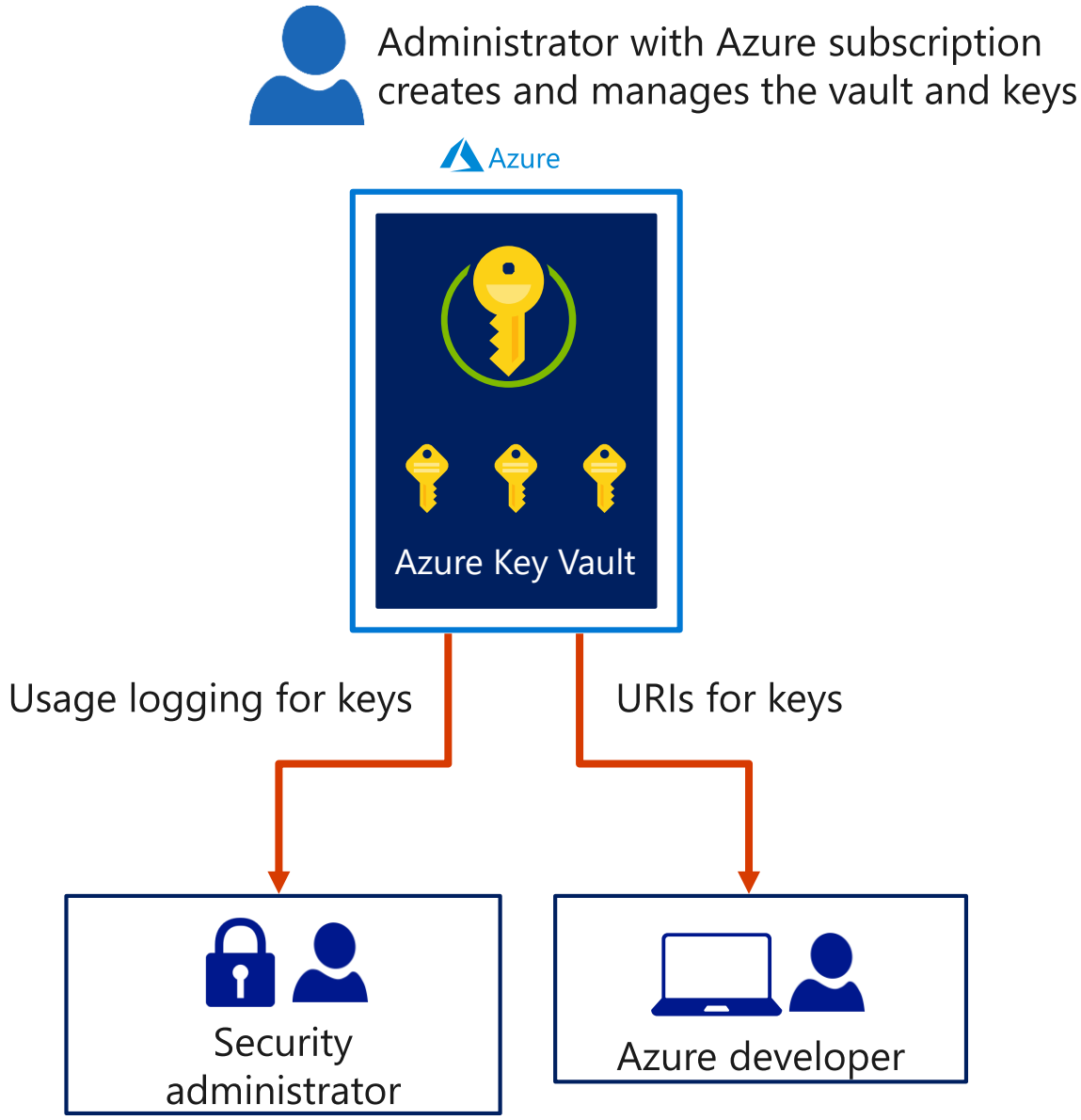
Azure Confidential Computing (continued 2)



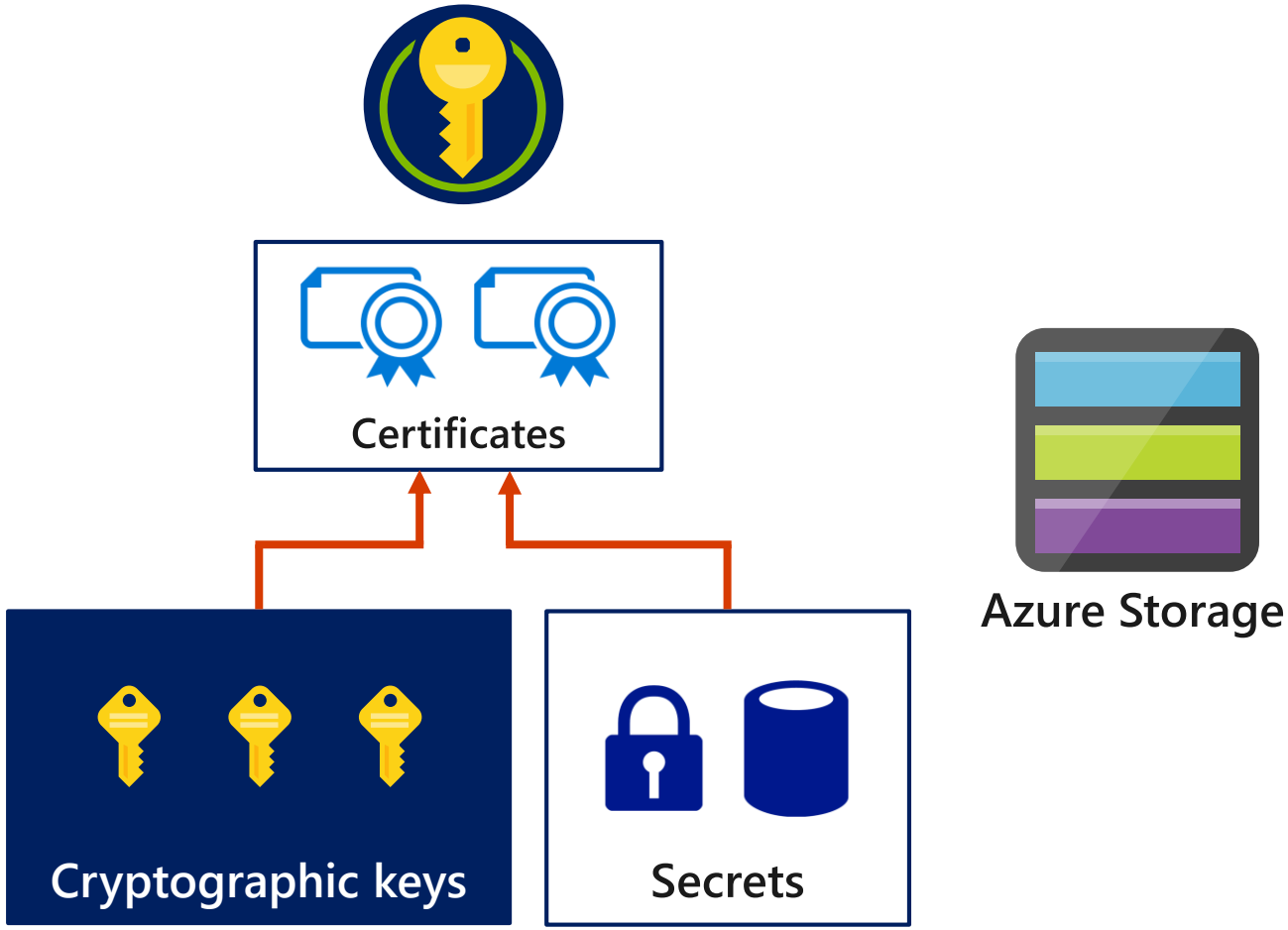
Lesson 04: Manage cryptographic keys in Azure Key Vault

Azure Key Vault

- Safeguard cryptographic keys and other secrets that cloud apps and services use
  - Increase security and control over keys and passwords
  - Applications have no direct access to keys
  - Use FIPS 140-2 Level 2 validated hardware security modules (HSMs)
  - Create and import
    - Encryption keys
    - API keys
    - Secrets
    - Passwords
    - SSL/TLS certificates

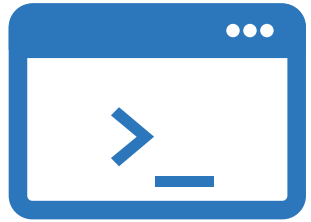


Key Vault secret types



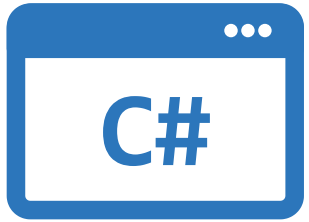
Create Key Vault secret by using Azure CLI

```
# Create resource group
az group create --name SecurityGroup --location westus

# Create Key Vault resource
az keyvault create --name contosovault --resource-group SecurityGroup --location westus

# Set secret in Key Vault
az keyvault secret set --vault-name contosovault --name DatabasePassword --value
'Pa5w.rd’

# Show value of secret in Key Vault
az keyvault secret show --vault-name contosovault --name DatabasePassword
```

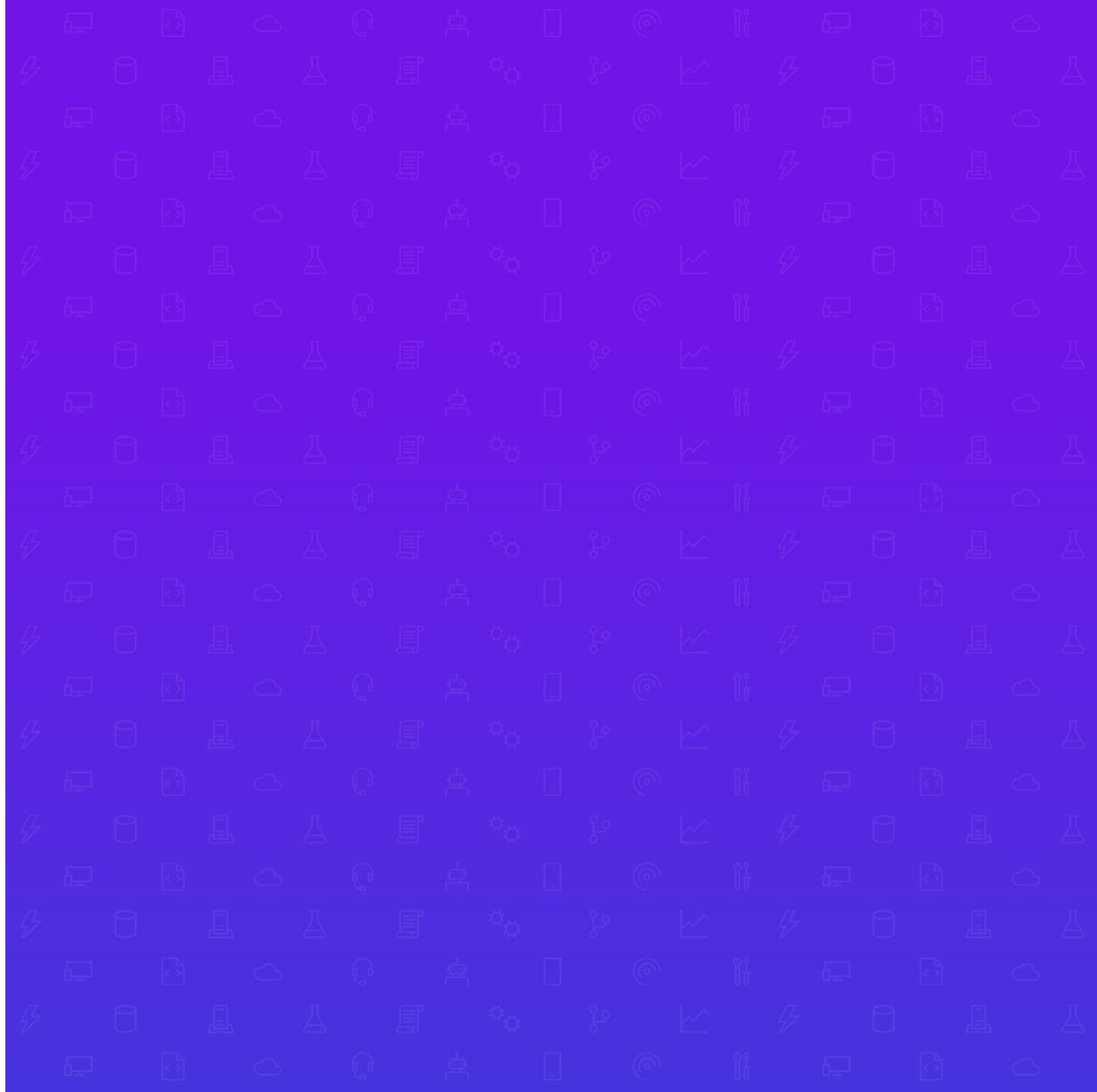


Get Key Vault secret by using C#

```
string secretUri = "https://contoso-vault2.vault.azure.net/secrets/example/0932840309";
var securityToken = "...";

// Create Key Vault client
var client = new KeyVaultClient(
    new KeyVaultClient.AuthenticationCallback(securityToken)
);

// Get secret
var secretBundle = await client.GetSecretAsync(secretUri);

// Get value of secret
var secret = secretBundle.Value;
```



Lab: Access resource secrets securely across services



Lab Login Information

- Virtual Machine

Microsoft

Review

- Encryption options
- End-to-end encryption
- Implement Microsoft Azure confidential computing
- Manage cryptographic keys in Azure Key Vault
- Lab: Access resource secrets securely across services