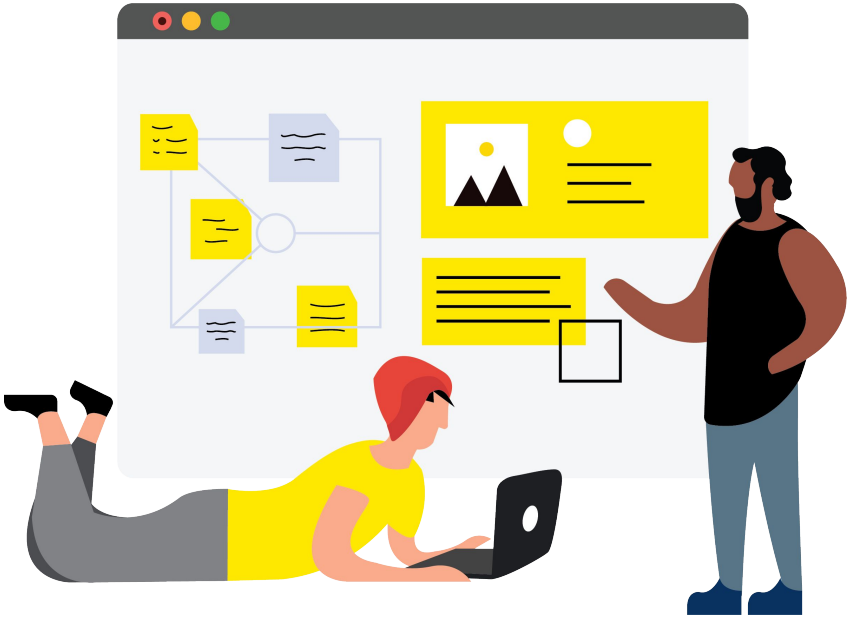


eficode

Secrets Management with Azure Key Vault

Finland Azure User Group 30.11.2021

Tapio Tuomisto

Cloud Architect
Email: tapio.tuomisto@eficode.com
Slack tapiot (msgurut.slack.com)
https://fi.linkedin.com/in/tapio-tuomisto

eficode

Agenda

What is secrets management?
What is Azure Key Vault?
How it works? Is it secure?
What should I do with it?
How to get started with Key Vault?

DEMO:
Automating Azure Key Vault secrets with Terraform.
Using secrets with virtual machines and App Service

Source code: https://github.com/t5t8/keyvault-demo

eficode

What is secrets management?

Automated tools, methods and processes to manage any authentication credentials.
Ways of keeping sensitive information secure.

Secrets can be anything you want to tightly control

- Passwords (user or software generated)
- Cryptographic keys, SSH keys, application keys
- Tokens, Connection strings
- Private Certificates
- One time password devices

eficode

What is secrets management?

Process of identifying and listing all secrets in organization and centralize the management.

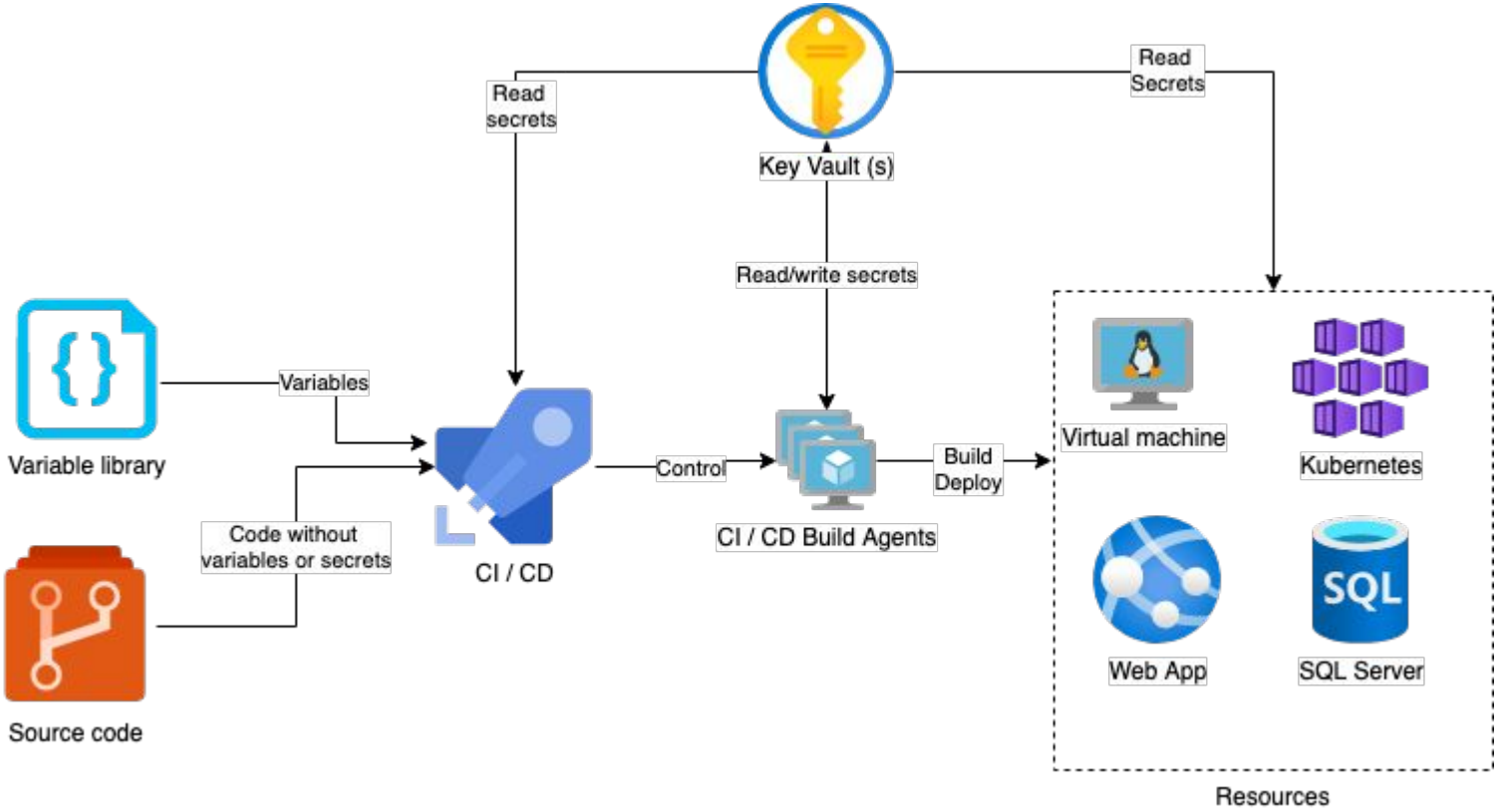
Enforcement of best practices

- Complexity of passwords, encryption methods
- Rotation and expiration
- Remove hardcoded secrets and default values from source code, variables and local machines

Auditing and monitoring secret usage
Extend to large teams with third parties
Crucial part of building security to DevSecOps

eficode

What is secrets management in DevOps?



eficode

What is Azure Key Vault?

Managed service available in all subscriptions

- Provides secure storage for
  - Keys
  - **Secrets**
  - Certificates
- Highly integrated to other Azure resources
- Centralizes and monitors secret usage
- Uses Roles or policies to control access
- Standard supports vaults with software keys
  - Premium can use hardware security module (HSM)
- Pricing based on requests
  - ~ 0.026 € / 10 000 requests (west europe today)

eficode

How Azure Key Vault works?

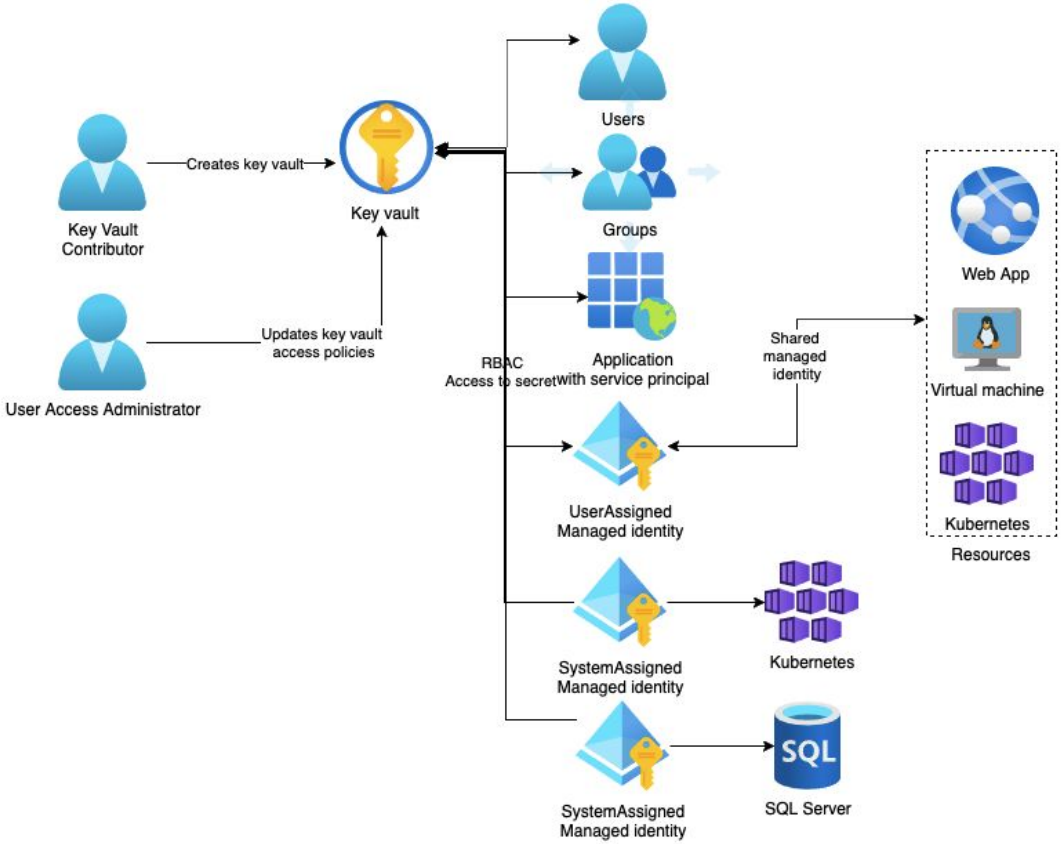
Key Vault Contributor can create multiple Key Vaults, that have own separate access control.

User Access Administrator defines access using either Role Based Access Control (RBAC) roles or Key Vault policies.

Key Vault can be accessed by Azure AD identities

- Users and security groups
- Managed identities
- Service principals with certificate or secret

Key Vault management and data plane access



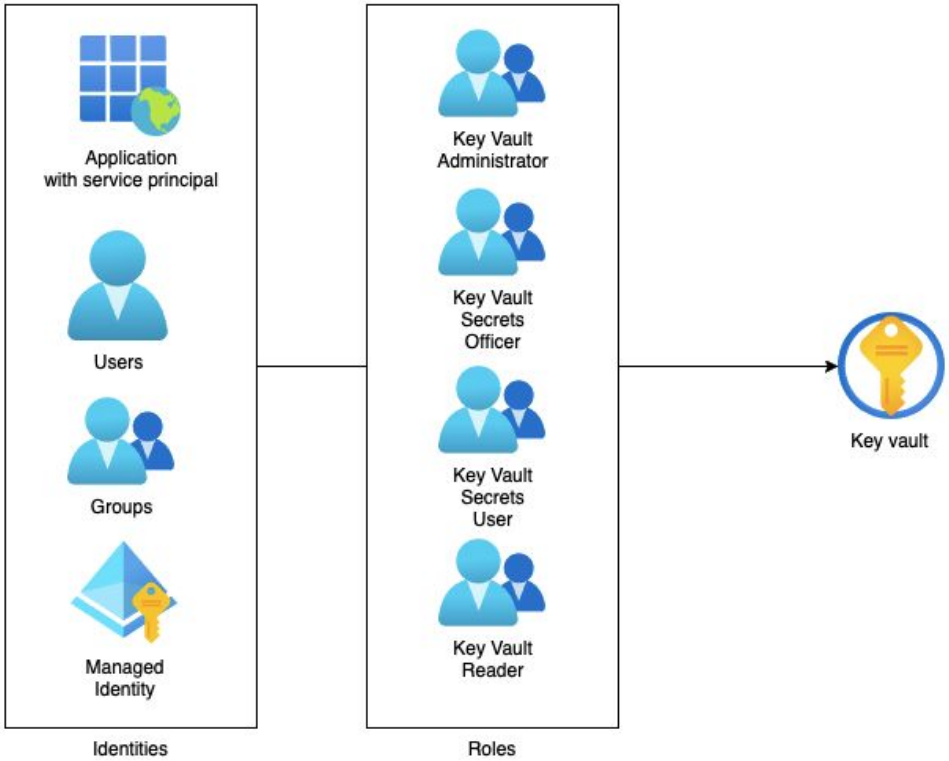
eficode

How to access Azure Key Vault secrets?

RBAC access model

- Modern way to setup access
- Uses RBAC role assignments like other resources.
- Built in roles, supports custom roles
- Custom role can access a specific secret
- Supports Privileged Identity Management for Just In Time access
- Changing role assignments may take a few minutes.
- Limited to 2000 assignments per subscription.

RBAC access to Key Vault secrets



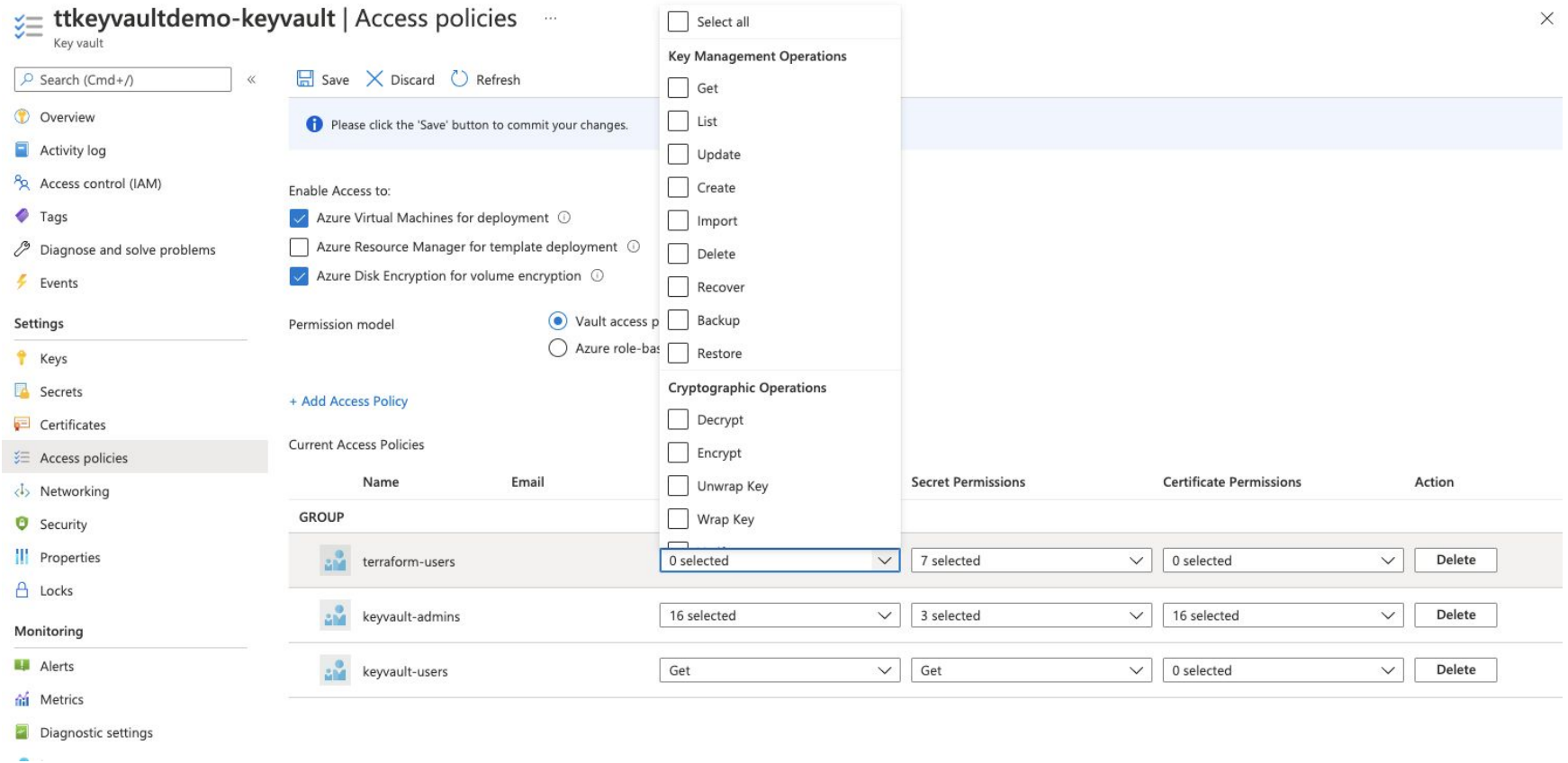
eficode

How to access Azure Key Vault secrets?

Policy based access model

- Classic way to setup access
- Uses Key Vault policies, not RBAC roles
- Many permission levels configurable separately
- Templates available for different access needs.
- Changing policies is fast operation
- Limited to 1024 policies per vault

Key Vault Access policies



eficode

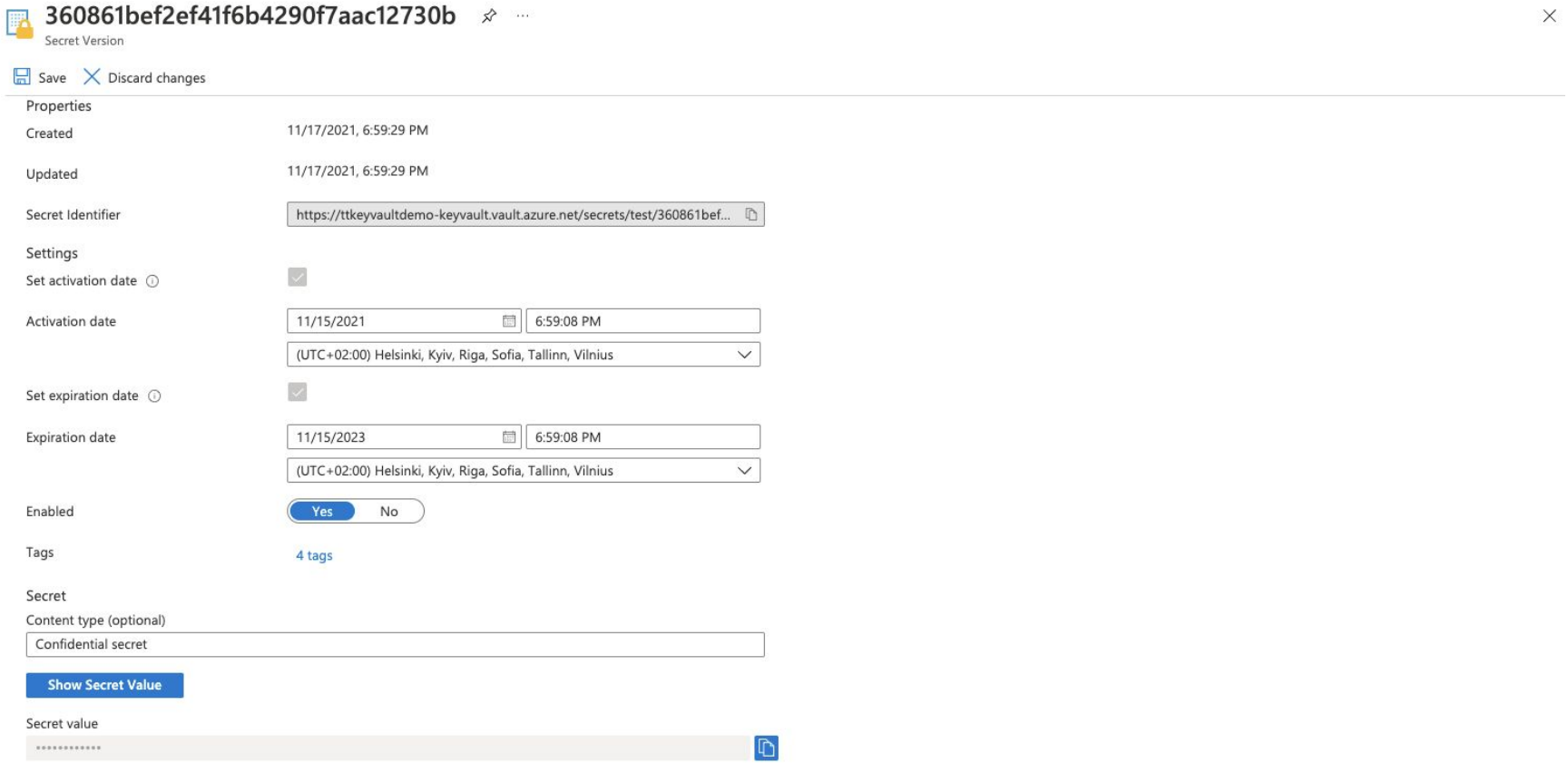
What can we do with a secret?

Access secret using identifier and do operations
- Management: get, list, set/update/create, …
- Cryptographic and rotation for keys
- Manage certificates and certificate authority

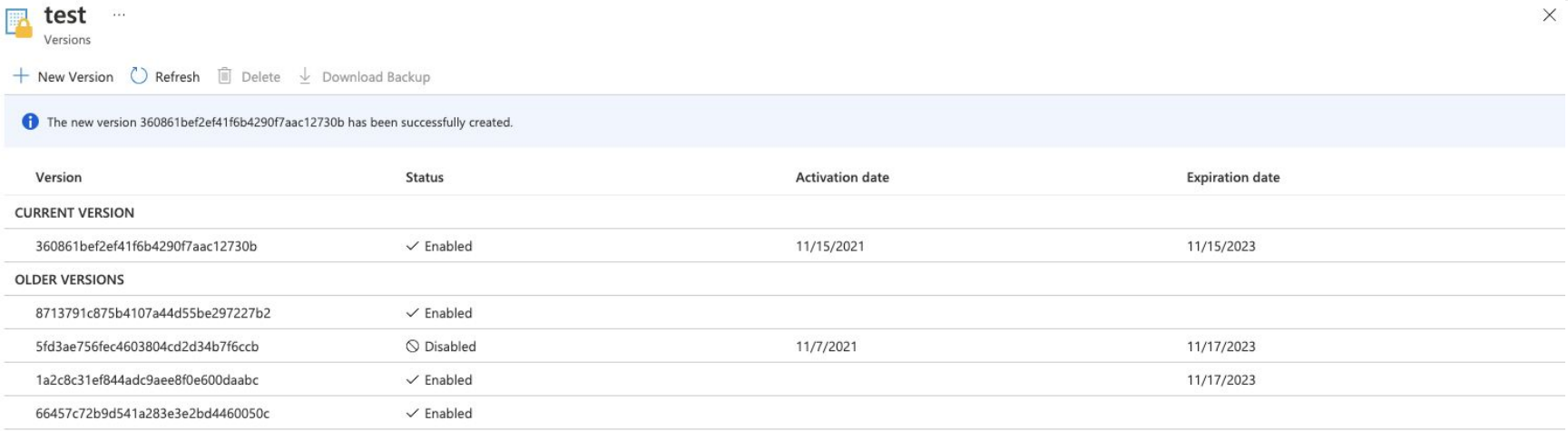
Log and monitor secret usage:
- storage account, event hubs, Azure Monitor

Create new versions, see old versions and rotate
- Activation and expiration dates.
- Versions, secret can be enabled and disabled.
- Use tags

Creating a new secret version



Secret versions, status, dates.



eficode

Is Azure Key Vault Secure?

Secured by Azure AD

- Central point to control identities.
- MFA, identity protection, and other features
- RBAC Roles, security groups

For extra security of Key Vault:

- Backups and recovery
- Split secrets to multiple Key Vaults
- Allowlist network connections
- Private endpoints
- Soft delete and purge protection
- Azure policies for governance

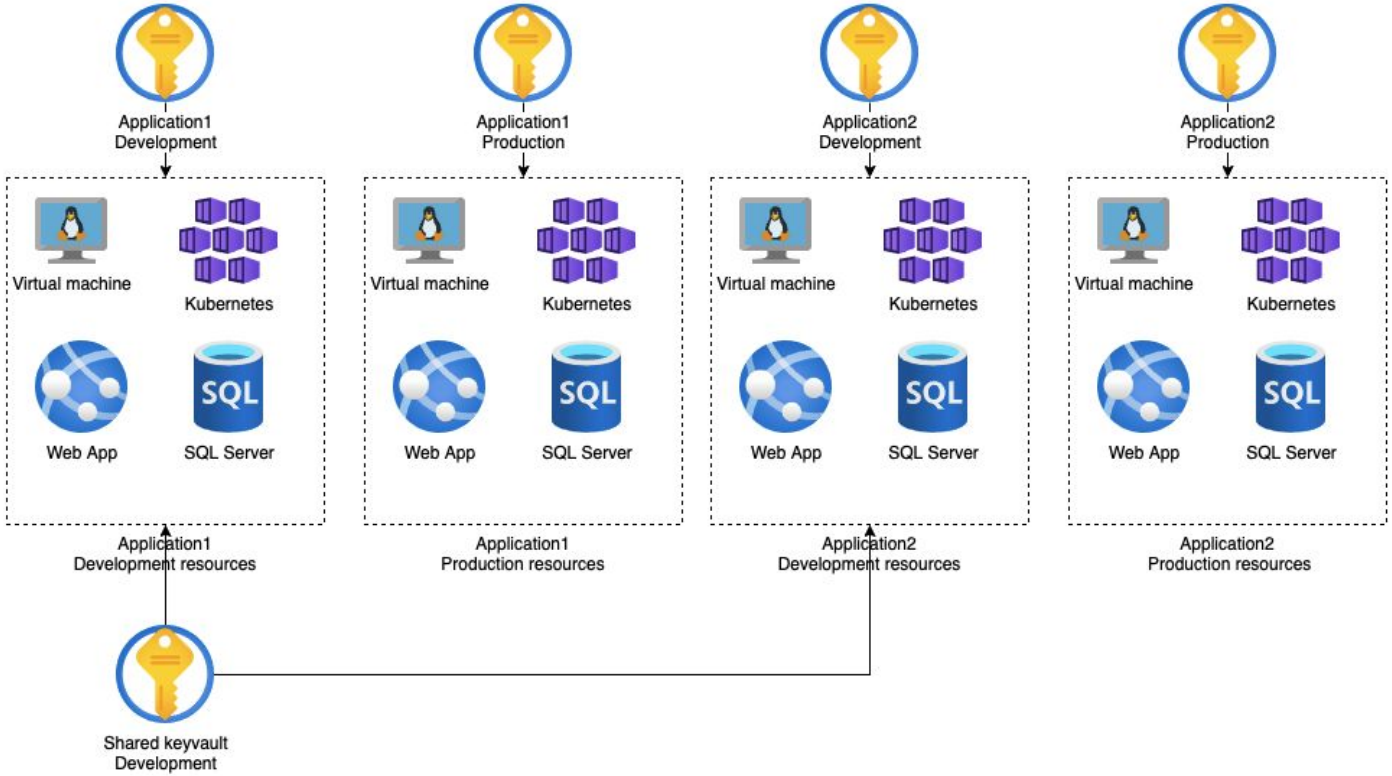
eficode

How to get started using Azure Key Vault?

Deploy multiple key vaults
- Based on purpose
- Shared vaults for teams including 3rd parties
- Keep the amount of vaults manageable

Automate and manage everything as code:
- Vaults and roles/policies
- Secret creation
- Secret consumption

Using multiple vaults in 2 projects



eficode

What should I do with Azure Key Vault?

Enable Teamwork
- Store secrets in central location
- Use roles and security groups
- Automate secret creation

Enable DevSecOps
- Use single source of truth for secrets
- Replace environment variables with secrets
- Use managed Identities to access secrets

Common use cases
- Store db connection strings for webapps
- Encrypt virtual disks with own key

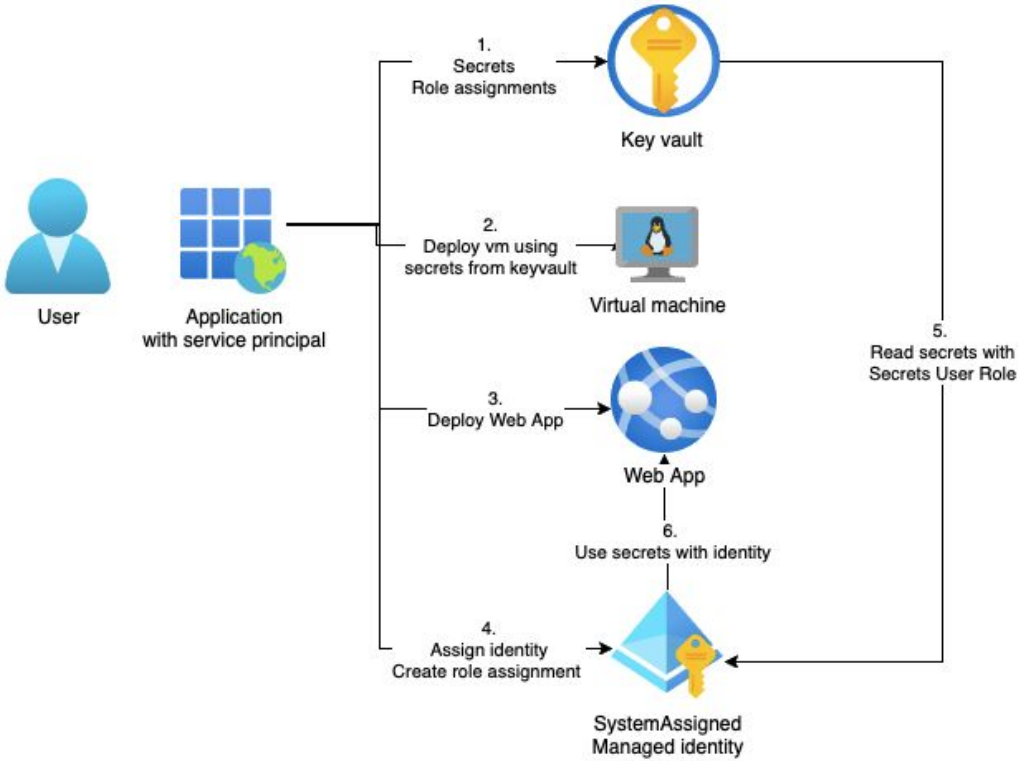
eficode

What should I do with Azure Key Vault?

Some more use cases:

- Access vms with Bastion host and ssh key
- Mount secrets, keys, and certificates to a pod by using a CSI volume in Azure Kubernetes Service
- Integrate Azure DevOps library to Key Vault
- Access secrets in VS Code or Visual Studio
- Generate certificates directly from Key Vault
- Build your own code or scripts with ready libraries.
- Automate infrastructure secrets with Terraform (or ARM, Bicep)

Demo


1.
Secrets
Role assignments
Key vault
User
Application
with service principal
2.
Deploy vm using
secrets from keyvault
Virtual machine
5.
Read secrets with
Secrets User Role
3.
Deploy Web App
Web App
6.
Use secrets with identity
4.
Assign identity
Create role assignment
SystemAssigned
Managed identity

```
## Set secret using random password
resource "random_password" "vm"
## Create tls private key
resource "tls_private_key" "vm"
## Set secrets
resource "azurerm_key_vault_secret" "vm_password"
resource "azurerm_key_vault_secret" "vm_private_key"
resource "azurerm_ssh_public_key" "vm"

## Set secrets to be used in virtual machine
resource "azurerm_virtual_machine" "vm"
 ## Read public key from SSH KEYS in vm
 admin_ssh_key
  public_key = azurerm_ssh_public_key .vm.public_key
 ## Read Password from key vault secrets
  admin_password =
    azurerm_key_vault_secret .vm_password.value
```

eficode

Demo, setup secrets for Virtual Machine with Terraform

In the demo we create randomized password and a key. These are stored in Key Vault as secrets and public key as ssh public key resource.

These secrets are then used to build a virtual machine.

Full source code:

https://github.com/t5t8/keyvault-demo

```
## Set app service identity
 identity { type = "SystemAssigned" }

## Grant app service access to keyvault secrets
resource "azurerm_role_assignment" "demo" {
 scope                = data.azurerm_key_vault.demo.id
 role_definition_name = "Key Vault Secrets User"
 principal_id         =
azurerm_app_service.demo.identity.0.principal_id
}
## Set these values from keyvault to app settings
 app_settings = {
   "VMPassword_from_keyvault"   = "@Microsoft.KeyVault(
VaultName=${data.azurerm_key_vault.demo.name};
SecretName=${var.prefix}-vm-admin-password)"
   "VMPrivatekey_from_keyvault" = "@Microsoft.KeyVault(
VaultName=${data.azurerm_key_vault.demo.name};
SecretName=${var.prefix}-vm-ssh-private-key)"
```

eficode

App Service With Secrets and system managed identity

In the demo we create a webapp with SystemAssigned managed identity. Then we make a role assignment to enable reading of secrets from Key Vault. These secrets can then be read by the webapp and used in app settings.

The webapp will update the values once a day from Key Vault if there are changes

Full source code:
https://github.com/t5t8/keyvault-demo

eficode

Questions or comments?

www.eficode.com