Secrets Management in DevOps (Vault, AWS Secrets Manager)

Secrets such as API keys, database passwords, TLS certificates, and SSH keys are critical assets in modern software systems. Hardcoding or mismanaging these secrets can lead to severe security breaches. Secrets management tools help securely store, access, and manage these sensitive credentials.

This writeup explores the **principles of secrets management** and compares **HashiCorp Vault** and **AWS Secrets Manager**.

🔧 HashiCorp Vault

Vault is a popular open-source tool designed to manage secrets and protect sensitive data.

Features:

- **Dynamic secrets**: Generates secrets on demand (e.g., DB credentials).
- **Secret engines**: Support for multiple backends like AWS, PostgreSQL, PKI, etc.
- **Policy-based access control**: Uses HCL for defining who can access what.
- **Audit logs**: All access can be logged for compliance.
- **Encryption as a Service**: Secure storage for app secrets.

Architecture:

Vault supports both:

- **Dev mode**: For local testing

- **HA mode**: With integrated storage or external backends like Consul

Example Workflow:

```
# Start Vault in dev mode
vault server -dev

# Authenticate
vault login <token>

# Store a secret
vault kv put secret/db password='mysecret'

# Retrieve it
vault kv get secret/db
```

🔍 Comparison: Vault vs AWS Secrets Manager

Feature	HashiCorp Vault	AWS Secrets Manager
Hosting	Self-hosted or cloud-managed	Fully managed (AWS only)
Secret Rotation	Dynamic secrets, customizable	Built-in for RDS, Lambda-based extensible
Integration	Cloud-agnostic	Deep AWS integration
Audit Logging	Built-in	CloudTrail/AWS logging
Use Case Fit	Multi-cloud, advanced security needs	AWS-native apps
Cost	Free/self-hosted (open-source)	Pay-per-secret and per-access

🔚 Conclusion

Secrets management is a foundational pillar in DevOps and cloud security. Tools like **Vault** provide flexibility and strong security controls for multi-cloud environments, while **AWS Secrets Manager** offers ease of use for AWS-native systems. Choosing the right solution depends on your architecture, team skillsets, and compliance needs.