Practice Production Hardening

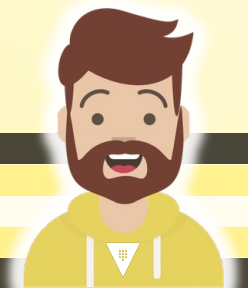
Intro to Production Hardening

There are many best practices for a production hardened deployment of Vault.

Practice defense in depth and follow the Vault security model

Product Hardening is broken down into multiple categories:

- General Recommendations
- Operating System Recommendations
- Vault-Specific Recommendations



Note for this Section

This will be a conceptual discussion with no demos

In the exam, you will NOT be expected to know how to perform Operating System-level configurations, like disabling swap or make file permission changes.

However, you may be asked questions (multiple choice) on HOW to make a configuration or deployment MORE secure based on provided examples.

General Topics

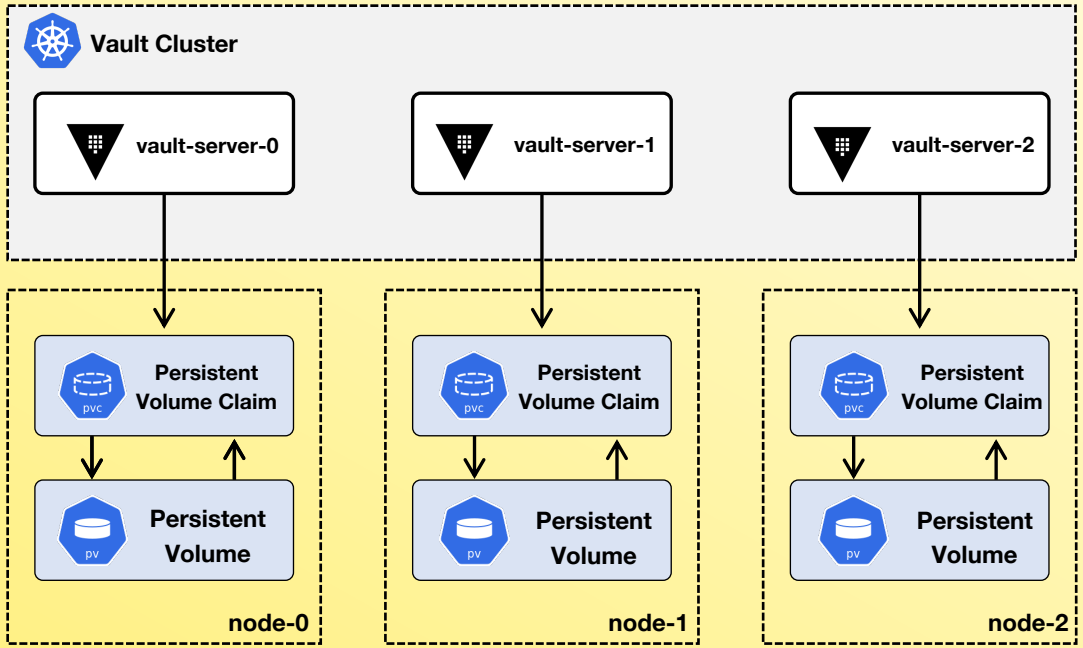
Deployment Model

- The fewer shared resources, the better
- Think "single tenancy" where possible
- Secure deployments: Hardware > VMs > Containers
- Ultimately comes down to protecting memory contents

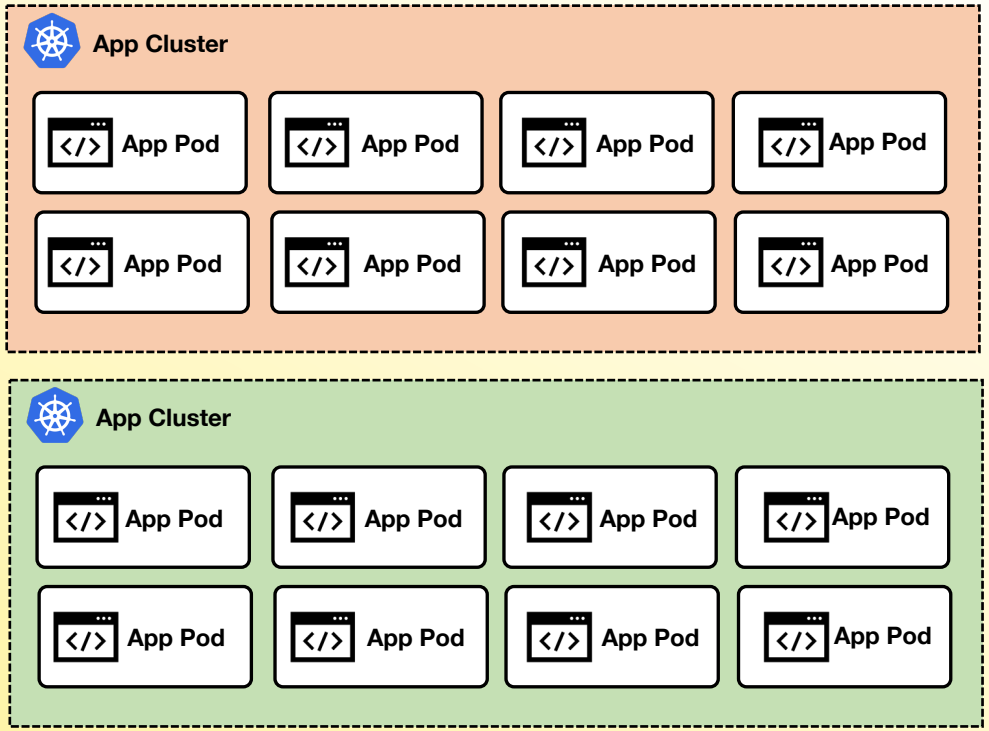
- Many customers will still use virtualization (VMware/Cloud) or containerization (Docker/K8s) but will deploy to dedicated clusters

Deployment Model

Dedicated Vault Cluster



Application Clusters



Limit Access to Vault Nodes

- Reduce or eliminate access to Vault nodes
- Includes SSH/RDP and through platform-based access (i.e., AWS SSM, `kubectl exec -it <pod>`, etc)
- Instead, access Vault via API or CLI from your workstation or jump box

- If you REALLY need access, use HashiCorp Boundary to limit/control access

Limit Services Running on Vault Nodes

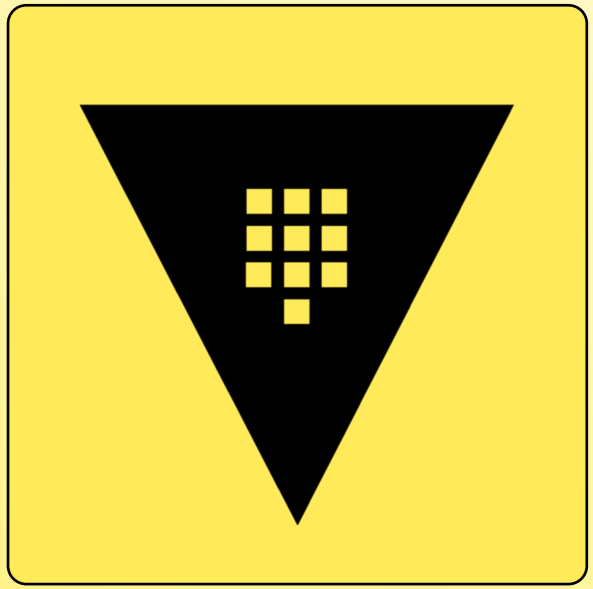
- Vault nodes should be dedicated to Vault services
- You should not have other services contending for resources
- More services = more firewall requirements
- Don't forget: Encryption keys are stored in memory

- Exception to this rule may include:
  - Telemetry agent
  - Log file agent (Splunk, SumoLogic, DataDog)

Permit Only Required Ports on Firewall

- Vault and Consul use dedicated ports for communication
- Permit only the required ports to reduce attack surface
- Many enterprise Vault deployments don’t even allow SSH or UI ports
- Default ports include:
  - Vault: 8200, 8201
  - Consul: 8500, 8201, 8301

More Ports May Be Needed for Your Implementation



Vault Cluster

tcp/443
Cloud API

Cloud Platform API

tcp/3306
Database

Database Servers

tcp/389
LDAP

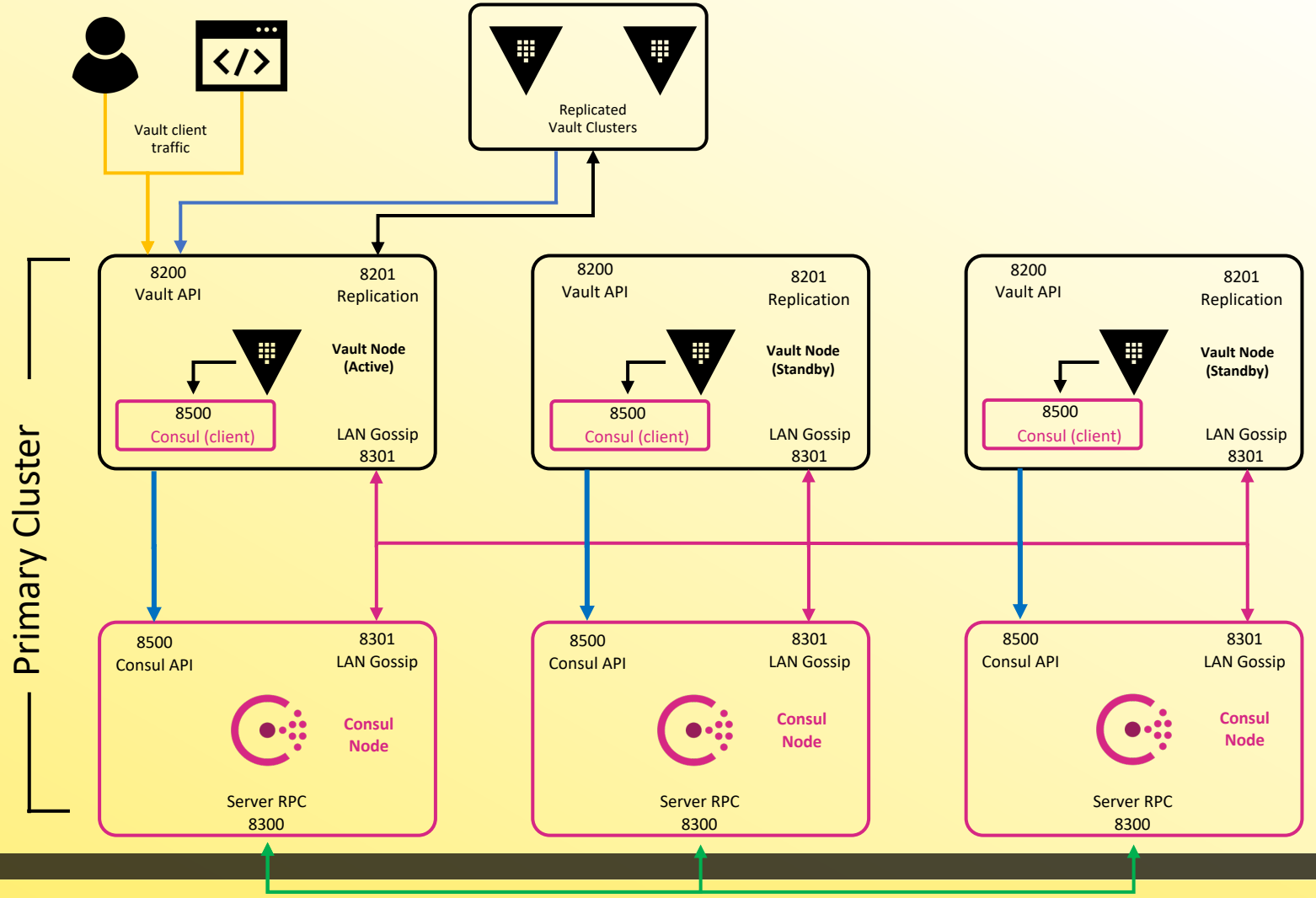
Active Directory

tcp/443
Cloud-Based File Storage

Vault Snapshots

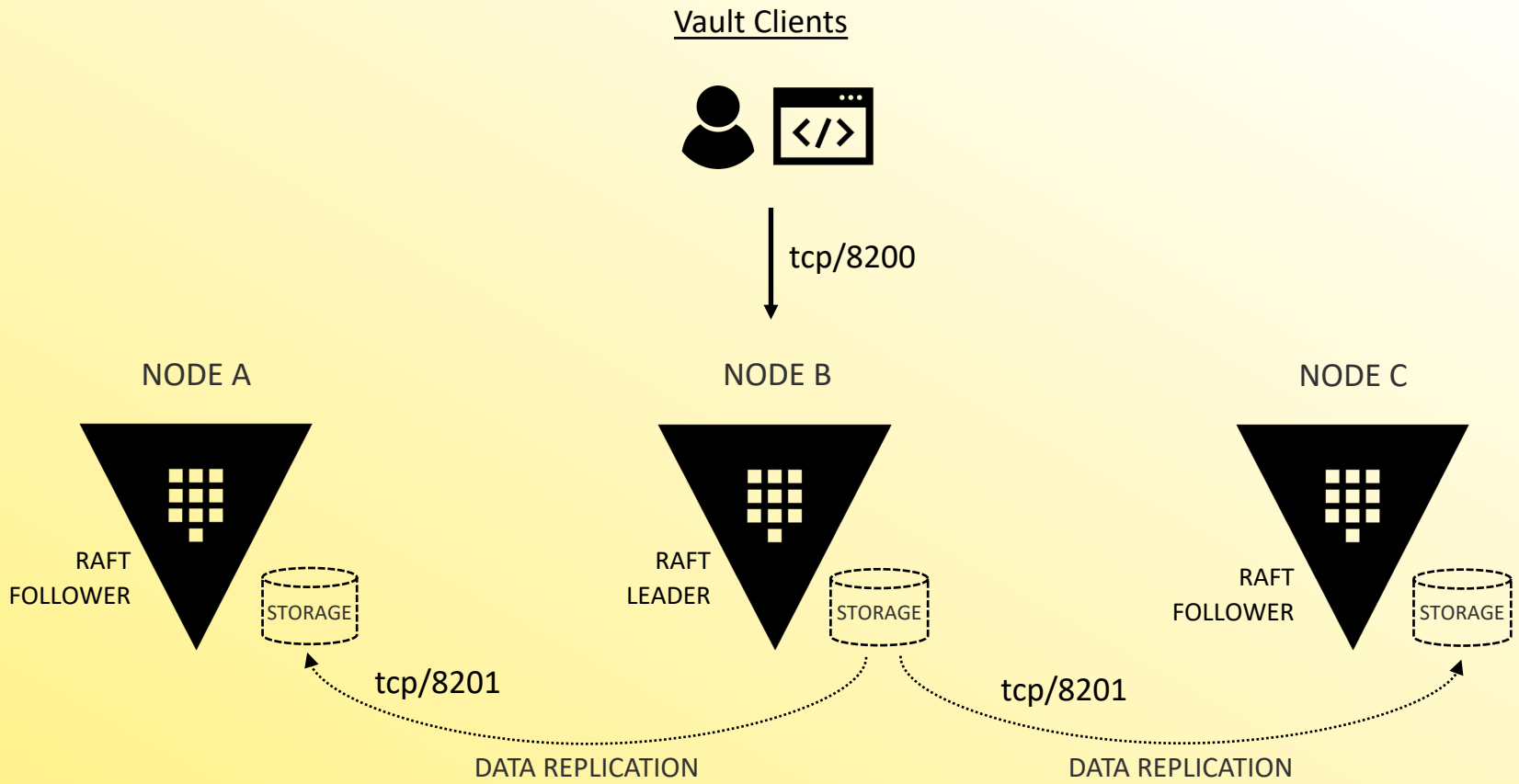
Permit Only Required Ports on Firewall

Consul Example



Permit Only Required Ports on Firewall

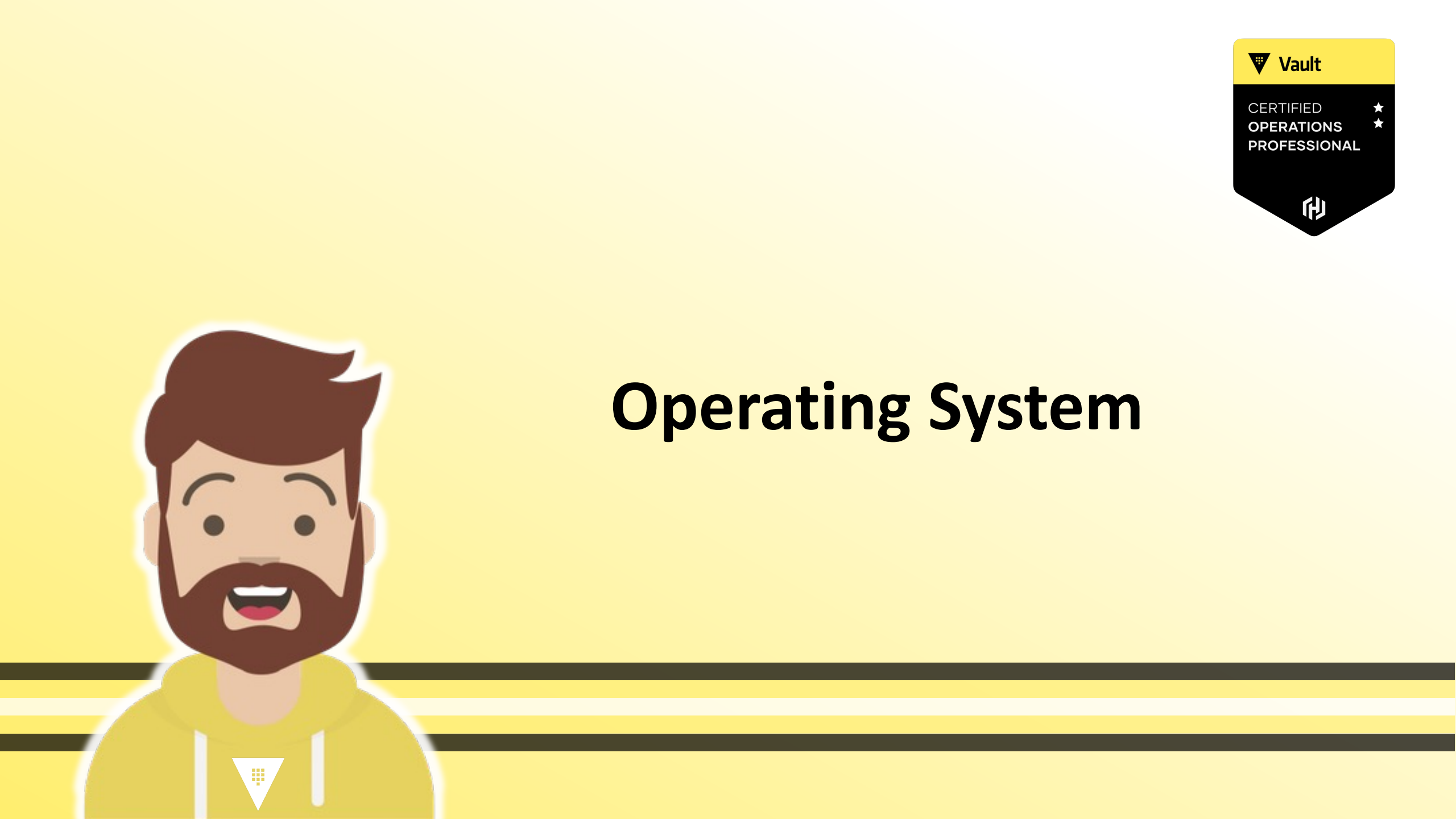
Integrated Storage Example



Immutable Upgrades

- Immutable upgrades guarantee a known state because you know the result of your automation configurations
- Easy to bring new nodes online, destroy the old nodes
- Consul and Raft can use AutoPilot to assist with upgrades

- Care must be taken when using Raft because you need to ensure replication has been completed to newly added nodes



Operating System

Run Vault as an Unprivileged User

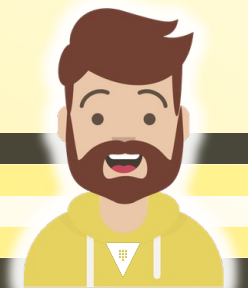
- Never run Vault as Root
- Running as root can expose Vault's sensitive data
- Limit access to configuration files and folders to the Vault user
- Normally, I create a user named "vault"

- Don't forget that the Vault user will need access to write local files, such as the database, audit logs, and snapshots
  - `chmod -R vault:vault /opt/vault/data`

Secure Files and Directories

- Protect and audit critical Vault directories and files, including directory for snapshots
- Ensure unauthorized changes can’t be made
- Includes binaries, config files, plugins files and directory, service configurations, audit device files and directory, etc.

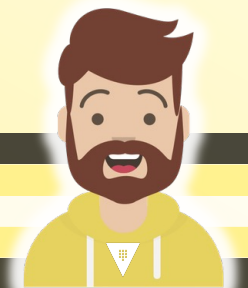
```
# Set permissions on Vault folder

$ chmod 740 -R /etc/vault.d
```



Protect the Storage Backend

- Vault writes all configuration and data to the storage backend
- No storage backend = No Vault!

- Consul Storage Backend:
  - Use Consul ACLs when running on Consul
  - Limit access to any Consul node
  - Enable `verify_server_hostname` in config file



Disable Shell History

- Disabling history prevents retrieval of commands
- Possible to discover credentials/tokens in history
- Can also disable just 'vault' command in history

```
$ history
 1365  vault login hvs.RTMd9YZ5Np9WGjvTfARaqffQ
 1366  vault policy list
 1367  vault list sys/policies/acl
 1368  vault secrets list
 1369  vault list pki/roles
```

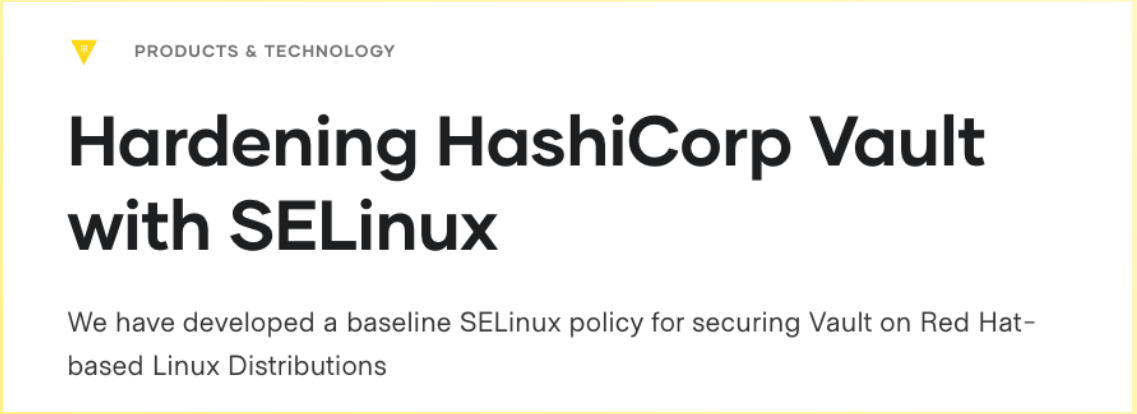
to prevent this...

```
# Disable a command history system wide

$ echo 'set +o history' >> /etc/profile
```

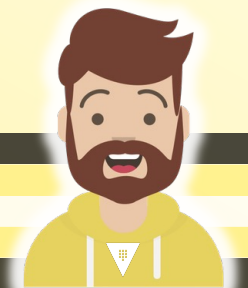
do this....

Configure SELinux/AppArmor

- Don’t disable to make install/management easier
- Provides additional layers of protection for the OS
- Adhere to CIS or DISA to improve posture of the host OS



hashicorp.com/blog/hardening-hashicorp-vault-with-selinux



Turn Off Core Dumps

- Core dumps could reveal encryption keys
- Different process to disable depending on the OS

- You won't have to do this in the exam, just know it's something that you SHOULD do in a production environment

Protect and Audit the vault.service File

- Make sure you know if this file is modified or replaced
- An attacker could point to compromised binaries to leak data

- This assumes you are running systemd, but any "service" file should be monitored and secured

Patch the Operating System Frequently

- Make sure to patch the OS frequently
- Follow your standards or be more stringent for Vault
- Options include Satellite, SpaceWalk, or other solution

- If you use an immutable architecture, then replace the nodes often with known good "patched" images. Use Packer to help simplify this workflow.

Disable Swap

- Vault stores sensitive data in-memory, unencrypted
- That data should never be written to disk
- Disabling swap provides an extra layer of protection

- Different process for different operating systems but again, you don't have to do this on the exam
- Example is enabling `mlock` to prevent memory swap


Vault
CERTIFIED
OPERATIONS
PROFESSIONAL

Vault-Specific Configurations

Secure Vault with TLS

- Vault contains sensitive data
- Communications should never occur without TLS in place
- Load Balancers used with Vault can terminate TLS or instead use pass through to the Vault nodes.
- Verify `tls_disable` configuration does not equal `true` or `1`
    - Default is `false` (not disabled)

Secure Consul

- Consul contains your sensitive data
- Communications should never occur without TLS in place
- Issue a certificate from a trusted CA
- Enable Consul ACLs
- Configure gossip encryption (use `consul keygen`)

- In short, follow the Consul Security Model

Enable Auditing

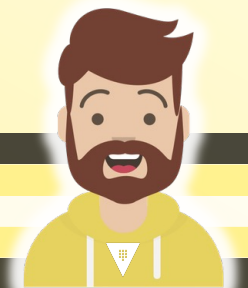
- Use multiple Audit Devices to log all interactions
- Send that data to a collection server
- Archive log data based upon security policies
- Create alerts based on certain actions

Say No to Cleartext Credentials

- Don’t put credentials in configuration files
- Use Environment Variables, where supported
- Use cloud-integrated services, such as AWS IAM Roles or Azure Managed Service Identities

```
1  seal "awskms" {
2    region = "us-east-1"
3    kms_key_id = "12345678-abcd-1234-abcd-123456789101",
4    endpoint = "example.kms.us-east-1.vpce.amazonaws.com"
5    access_key="AKIAIOSFODNN7EXAMPLE"
6    secret_key="wJalrXUtnFEMI/K7MDENG/bPxRfiCYEXAMPLEKEY"
7  }
```

Don't do this....



Upgrade Vault Frequently

- New updates frequently include security fixes
- New cipher suites can be enabled or supported
- New functionality enabled

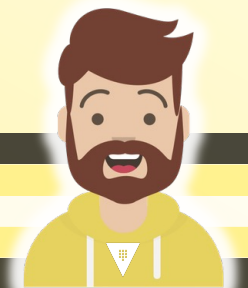
releases.hashicorp.com/vault/

```
../
vault_1.10.0+ent.hsm
vault_1.10.0
vault_1.10.0+ent
vault_1.10.0-rc1
vault_1.10.0-rc1+ent
vault_1.10.0-rc1+ent.hsm
vault_1.9.4+ent
```

Stop Using Root Tokens, Seriously

- Root tokens have unrestricted access to Vault **with no TTL**
- Not bound by ACL policies, ERPs, or RGPs

- Get rid of the initial root token after initial setup
  - `vault token revoke hvs.xxxxxxxxx`



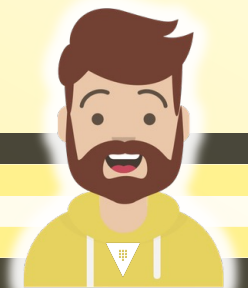
Verify the Integrity of the Vault Binary

- Always get Vault binaries directly from HashiCorp
- Use the HashiCorp checksum to validate
- Modified version of Vault binary could leak data

Disable the UI – if Not in Use

- Vault UI is disabled by default
- Configured in the Vault configuration file
- Do the same for Consul UI, as well

```
1  ui = false
2  log_level = "INFO"
3  license_path = "/opt/vault/vault.hcl"
```



Encrypt the Gossip Protocol (Consul)

- TLS only secures the interfaces, not Consul gossip traffic
- Use the –encrypt flag in the Consul configuration file
- Uses a 32-byte key – can use `consul keygen` to generate

Secure the Unseal/Recovery Keys

- Initialize Vault using PGP keys, such as keybase.io
- Distribute to multiple team members, no single person should have all the keys
- Don’t store the keys in Vault itself

Minimize the TTLs for Leases and Tokens

- Use the smallest TTL possible for tokens
- Define Max TTLs to prevent renewals beyond reasonable timeframe

- Minimizing TTL also helps reduce burden on the storage backend

Follow the Principle of Least Privilege

- Only give tokens access to paths required for business function
- Separate policies for applications and users
- Limit use of * and + in policies, where possible
- Templated policies can help with policy creation and maintenance

Perform Regular Backups

- Backup configuration files and directories
- Automate Vault backup using snapshots or equivalent depending on the storage backend
- Regularly test backups to ensure functionality

- `vault operator raft snapshot save monday.snap`
- `vault write sys/storage/raft/snapshot-auto/config/daily` (Enterprise)

Integrate with Existing Identity Providers

- Use your existing IdP to provide access to users
- If/when users leave, they immediately lose access to Vault
- The fewer places a user has credentials, the better
- Using locally defined credentials is an administrative burden

Monitoring and Alerting

Vault Security Monitoring

- ❑ Use of a root token
- ❑ Creation of a new root token
- ❑ Vault policy modification
- ❑ Enabling a new auth method
- ❑ Modification of an auth method role
- ❑ Creation of a new auth method role
- ❑ Permission denied (403) responses
- ❑ Use of Vault by human-related accounts outside of regular business hours
- ❑ Vault requests originating from unrecognized subnets
- ❑ Transit Minimum Decryption Version Config
- ❑ Seal Status of Vault
- ❑ Audit Log Failures
- ❑ Resource Quota Violations
- ❑ Updates to Vault Policies
- ❑ Transit Key Deletion
- ❑ Cloud-based resource changes

Done via Audit Log and Log Collection Tool