The Container Store®

Simplifying and Securing SSH Access with HashiCorp Vault

Ryan Breidenbach
John Jelinek
Matt Smith

Who are we?

- Ryan – aspiring amateur poker player
- Matt – big fan of Black Mirror
- John – just got out of bitcoin

- Working on
  - Self Service Ops tools for developers
  - Infrastructure automation/Infrastructure as Code
  - Faster Deployment Pipelines
  - Improved Observability/Monitoring
- We are hiring. If this type of work looks interesting, come talk to us afterwards.

The Container Store®

The challenge

- What was my password on that server?
- Can you create me a user on server xxx?
- Can you add me to sudoers?
- Who has access to this box?
- Someone switches teams. How to we clean up their accounts across all boxes they had access to?
- It's time to rotate passwords. How to manage across all servers?

Linux user management options

- **Local accounts**
  - authenticated by passwords / keys
  - authorized via groups
  - Manually or scripted via config mgmt tools
- **PAMD/SSSD**
  - Authenticated by AD/LDAP
  - Authorized by . . .
- **Vendor'd solutions**
  - ssh proxies that hold/provision access on demand
  - Centralized, based solutions that sync accounts

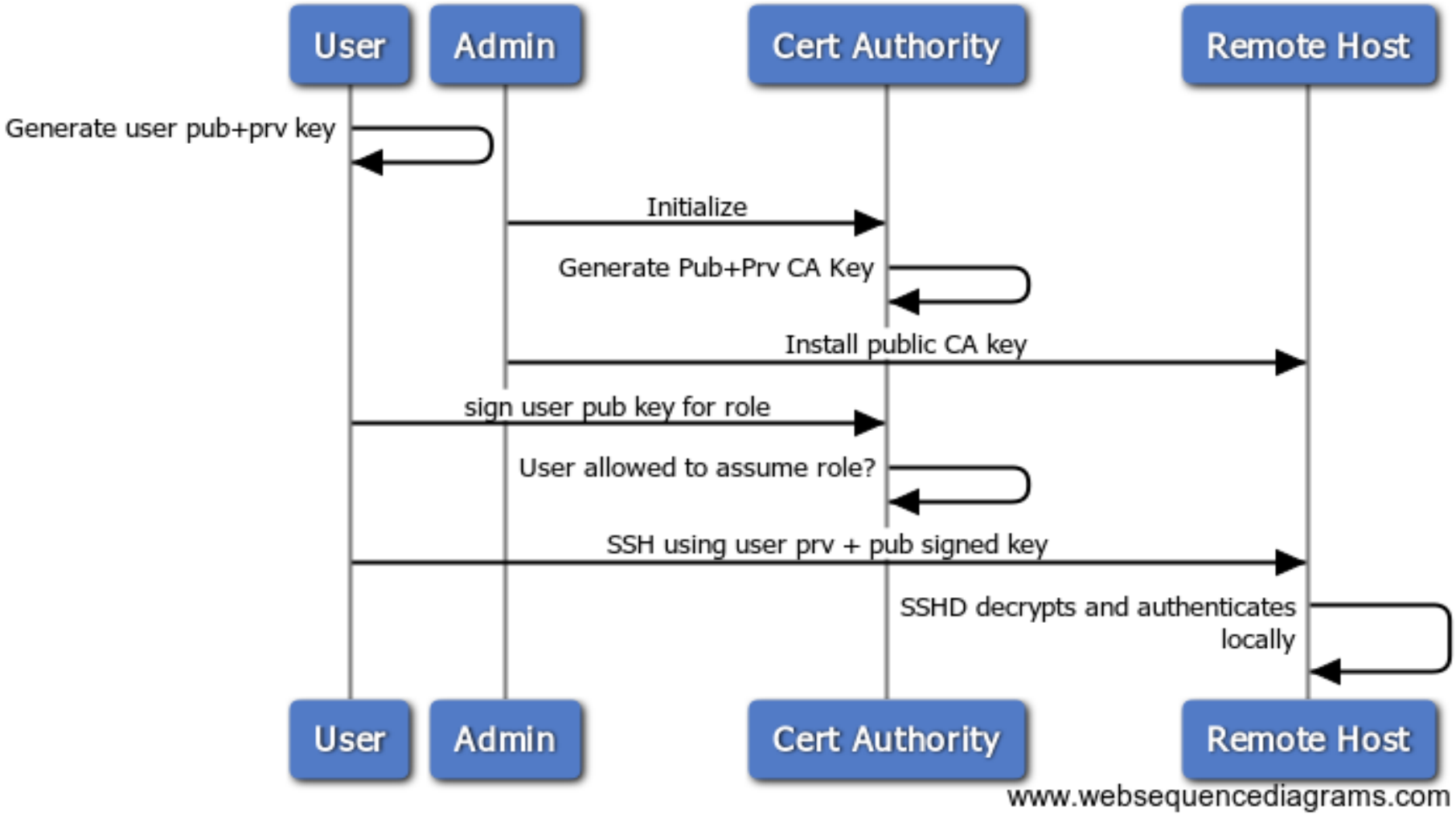
Problems with Other Approaches

- **Reliability**
- **Scalability**
  - Onboarding/termination is cumbersome
- **Vendor dependence**
- **Separation of Authentication/Authorization**
- **Configuration of servers falling out of date, error handling in updating all servers**
  - **Password drift**
- **PAM solutions could address these issues, but we have not found a good solution**

OpenSSH to the rescue

- An approach that appears to be pioneered (or at blogged about) by Facebook
- Uses built-in capabilities of OpenSSH
- Uses key signing to grant a user the authorization to login to a given server account for a limited time.

Overview of SSH with Signed Public Key



The Container Store®

Live Demo

Introduction to Vault

A Tool for Managing Secrets

Vault secures, stores, and tightly controls access to tokens, passwords, certificates, API keys, and other secrets in modern computing. Vault handles leasing, key revocation, key rolling, and auditing. Through a unified API, users can access an encrypted Key/Value store and network encryption-as-a-service, or generate AWS IAM/STS credentials, SQL/NoSQL databases, X.509 certificates, SSH credentials, and more.

Vault Concepts

- Authentication
- Tokens
- Policies
- Backends
  - Auth Backends
  - Secret Backends

Implementing SSH Key Signing with Vault

- Vault acts as the CA
- Vault provides and endpoint for key signing
- Access to this endpoint is controlled by Vault policies
- Policies are applied a user's token based on the authentication backend (e.g. LDAP backend integrating with Active Directory)

Live Demo

Thank You