Podman Image SCP

Design Discussion

Two distinct options

podman save | podman --remote load

Usages

- Similar functionality to podman load
- No ssh capabilities or scp
- Simpler, lighter weight

podman save | ssh HOST podman load

ssh HOST podman save | podman load

ssh HOST podman save | ssh HOST podman load

Usages

- Save and load to AND from remote host
- Save and load from remote host to remote host
- Utilizes podman system connection as well as new direct ssh connections
- New SCP library, uses the dialed ssh connection already established

Flow for proposed SCP

- init command
  - Format flag (debatable)
  - Quiet flag
- Parse options and args
  - args determine which kind of scp we are doing
    - **REM =>REM**
    - **LOC => REM**
    - **REM => LOC**
    - **LOC => LOC**
- Check if given connection is in **podman system connection** if not warn user, but accept the new one.
- Based on type, either execute on image engine or call **execCommand()** which creates an ssh connection given the data from the main scp function
- **execCommand()** takes identity, uri, imageFileIn, imageFileOut, image name, and a boolean to determine which way we are saving/loading
- **ToFrom = true**
  - does an **scp** to the remote host and then an **ssh HOST podman load …** and from the loaded file
- **ToFrom = false**
  - does a **ssh HOST podman save** and then a **scp** to the local host. The local host then uses an image engine back in the main **scp()** function to **podman save** the copied image.
- Once returned to **scp()** the code determines what is just did (save or load) and executes the opposite on the local client. (**ToFrom = F/T**)
- If BOTH hosts are specified as remote clients, the functions perform similar tasks but **execCommand()** executes twice once for the **save()** once for the **load()**

Example Commands

Podman image scp alpine FedoraHost:: => **local save and remote host load**

Podman image scp FedoraHost::alpine => **remote save and local load**

Podman image scp FedoraHost1::alpine FedoraHost2:: => **remote host save and load**

Podman image scp alpine FedoraHost::/home/user/Documents/alpine => **local save and remote host load to a temporary directory specified**

Podman image scp alpine => **local save and load**

Podman --remote image scp alpine => **local save and remote engine load**