Bootstrap automatique d'un poste Fedora d'administration

Cette page décrit la mise en place d'un poste Fedora destiné à l'administration de l'infrastructure. L'objectif est de pouvoir :

- installer les outils nécessaires ;
- générer une clé SSH ;
- créer automatiquement le fichier `~/.ssh/config` ;
- déployer la clé publique sur les hôtes utilisant `root` ;
- déployer la clé publique sur les hôtes utilisant `nancre` ;
- valider rapidement que tout fonctionne.

Prérequis

- Un poste Fedora avec accès administrateur local.
- Accès réseau aux serveurs du LAN `10.151.151.0/24`.
- Le mot de passe du compte `root` sur les hôtes concernés.
- Le mot de passe du compte `nancre` sur les hôtes concernés.

Script de bootstrap

Créer un fichier, par exemple `bootstrap-fedora.sh`, puis coller le script ci-dessous.

```
#!/usr/bin/env bash
set -euo pipefail

echo "=== Bootstrap Fedora Administration ==="

SSH_DIR="$HOME/.ssh"
KEY_FILE="$SSH_DIR/id_ed25519"
CONFIG_FILE="$SSH_DIR/config"

ROOT_HOSTS=(
```

```
  10.151.151.230
  10.151.151.100
  10.151.151.220
  10.151.151.231
  10.151.151.40
  10.151.151.239
  10.151.151.249
)

NANCRE_HOSTS=(
  10.151.151.244
  10.151.151.245
  10.151.151.246
  10.151.151.247
  10.151.151.240
)

echo
echo "[1/6] Installation des paquets nécessaires"
sudo dnf install -y openssh-clients sshpass

echo
echo "[2/6] Préparation du dossier ~/.ssh"
mkdir -p "$SSH_DIR"
chmod 700 "$SSH_DIR"

echo
echo "[3/6] Génération de la clé SSH si absente"
if [[ ! -f "$KEY_FILE" ]]; then
  ssh-keygen -t ed25519 -a 100 -f "$KEY_FILE" -N ""
else
  echo "Clé déjà existante : $KEY_FILE"
fi

echo
echo "[4/6] Création de ~/.ssh/config"
cat > "$CONFIG_FILE" <<'EOF'
Host nas
    HostName 10.151.151.201
    User n.ancre
```

```
    Port 22022
    IdentityFile ~/.ssh/id_ed25519

Host 10.151.151.* !10.151.151.201
    User root
    IdentityFile ~/.ssh/id_ed25519
    StrictHostKeyChecking no
    UserKnownHostsFile /dev/null

Host proxmox
    HostName 10.151.151.249
    User root
    IdentityFile ~/.ssh/id_ed25519

Host proxmox-ms
    HostName 10.151.151.239
    User root
    IdentityFile ~/.ssh/id_ed25519

Host squall
    HostName 10.151.151.245
    User nancre
    IdentityFile ~/.ssh/id_ed25519

Host linoa
    HostName 10.151.151.246
    User nancre
    IdentityFile ~/.ssh/id_ed25519

Host quistis
    HostName 10.151.151.247
    User nancre
    IdentityFile ~/.ssh/id_ed25519

Host selphie
    HostName 10.151.151.240
    User nancre
    IdentityFile ~/.ssh/id_ed25519

Host network
```

```
    HostName 10.151.151.40
    User root
    IdentityFile ~/.ssh/id_ed25519
    StrictHostKeyChecking no

Host webserver
    HostName 10.151.151.100
    User root
    IdentityFile ~/.ssh/id_ed25519
    StrictHostKeyChecking no

Host bitwarden
    HostName 10.151.151.231
    User root
    IdentityFile ~/.ssh/id_ed25519
    StrictHostKeyChecking no

Host bookstack
    HostName 10.151.151.220
    User root
    IdentityFile ~/.ssh/id_ed25519
    StrictHostKeyChecking no

Host bastion
    HostName 10.151.151.244
    User nancre
    IdentityFile ~/.ssh/id_ed25519
    StrictHostKeyChecking no
    UserKnownHostsFile /dev/null

Host plex
    HostName 10.151.151.105
    User nancre
    IdentityFile ~/.ssh/awx_ansible_key
EOF

chmod 600 "$CONFIG_FILE"

echo
echo "[5/6] Déploiement de la clé sur les hôtes root"
```

```
read -s -p "Mot de passe root : " ROOTPASS
echo
export SSHPASS="$ROOTPASS"

for host in "${ROOT_HOSTS[@]}"; do
  echo "  - root@$host"
  sshpass -e ssh-copy-id \
    -o StrictHostKeyChecking=no \
    -i "$KEY_FILE.pub" \
    root@"$host"
done

unset SSHPASS
unset ROOTPASS

echo
echo "[6/6] Déploiement de la clé sur les hôtes nancre"

read -s -p "Mot de passe nancre : " NANCREPASS
echo
export SSHPASS="$NANCREPASS"

for host in "${NANCRE_HOSTS[@]}"; do
  echo "  - nancre@$host"
  sshpass -e ssh-copy-id \
    -o StrictHostKeyChecking=no \
    -i "$KEY_FILE.pub" \
    nancre@"$host"
done

unset SSHPASS
unset NANCREPASS

echo
echo "=== Vérification ==="
ssh -G proxmox > /dev/null
ssh -G squall > /dev/null
ssh -G bastion > /dev/null
```

```
echo
echo "Bootstrap terminé avec succès."
```

Installation du script

```
nano bootstrap-fedora.sh
chmod +x bootstrap-fedora.sh
```

Exécution

```
./bootstrap-fedora.sh
```

Le script va demander deux mots de passe distincts :

- le mot de passe de `root` pour les machines administrées en root ;
- le mot de passe de `nancre` pour les machines administrées avec ce compte.

Structure des accès

Accès en root

- `10.151.151.230`
- `10.151.151.100`
- `10.151.151.220`
- `10.151.151.231`
- `10.151.151.40`
- `10.151.151.239`
- `10.151.151.249`

Accès avec nancre

- `10.151.151.244`
- `10.151.151.245`
- `10.151.151.246`
- `10.151.151.247`

- `10.151.151.240`

Tests manuels

Après exécution, vérifier les alias SSH :

```
ssh proxmox
ssh proxmox-ms
ssh squall
ssh linoa
ssh quistis
ssh selphie
ssh network
ssh webserver
ssh bitwarden
ssh bookstack
ssh bastion
ssh nas
```

Si tout est correct, aucune demande de mot de passe ne doit apparaître pour les hôtes déjà provisionnés.

Notes

- Le compte `plex` utilise ici une clé dédiée `~/.ssh/awx_ansible_key`.
- Si tu veux uniformiser complètement le poste, tu peux aussi basculer Plex sur la clé `id_ed25519`.
- La configuration est pensée pour être reproductible sur un nouveau portable Fedora.

Revision #2
Created 2026-06-19 14:07:52 UTC by Nico là
Updated 2026-06-19 14:09:24 UTC by Nico là