SSH refusé ? Le parcours rapide pour retrouver la bonne clé

Suivez ce parcours dans l'ordre pour identifier pourquoi le serveur refuse votre clé publique. Chaque étape isole une cause précise sans modifier la sécurité au hasard.

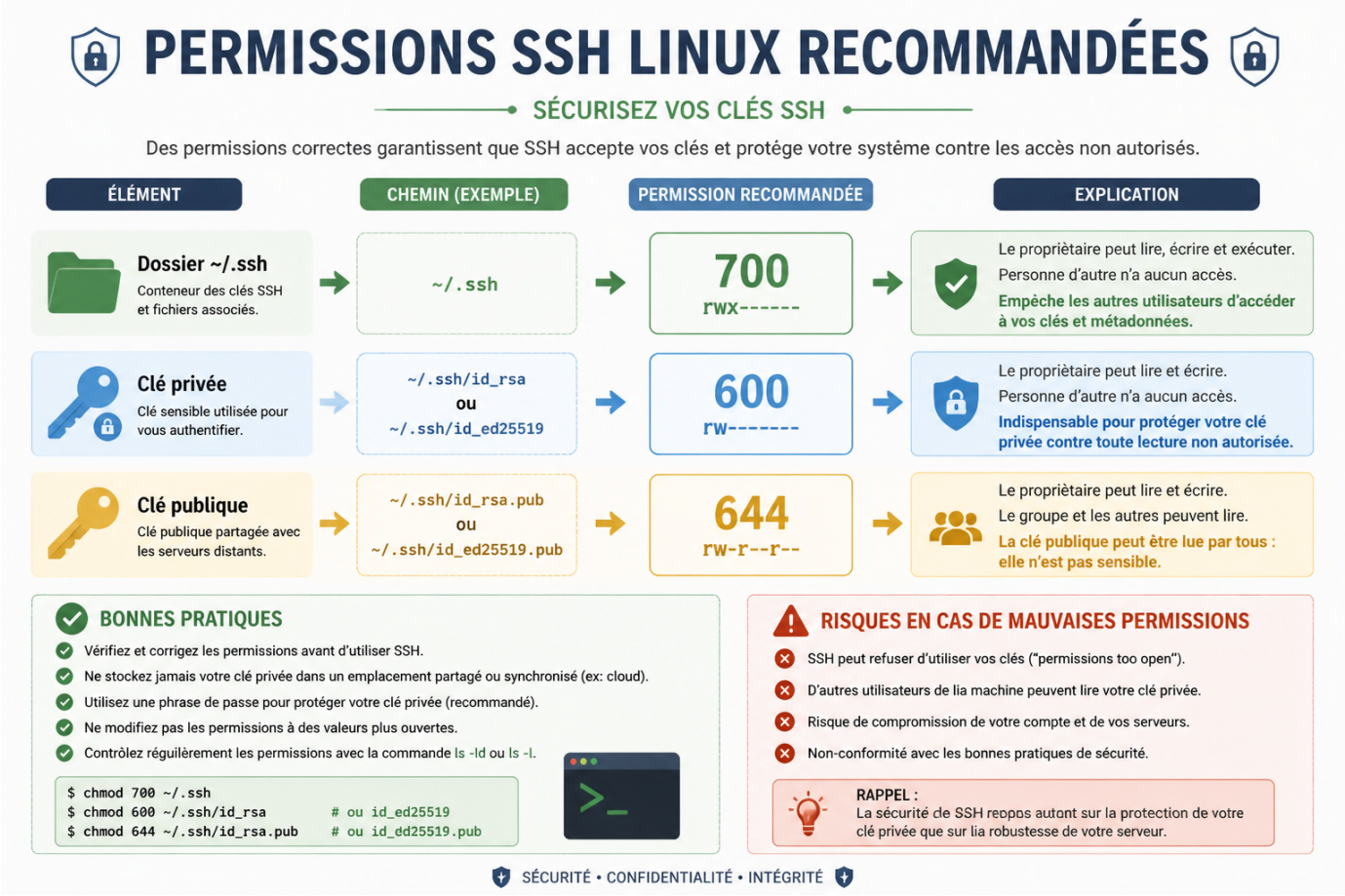
Au sommaire

Avant de commencer : ce que signifie l'erreur

« Permission denied (publickey) » est un refus d'authentification par le serveur, pas nécessairement un problème d'accès au réseau.

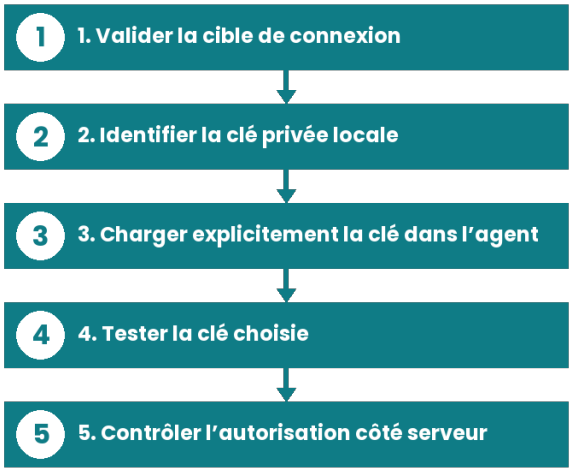
Les quatre contrôles prioritaires sont : compte distant, clé privée proposée, clé publique dans authorized_keys et permissions.

La clé privée reste exclusivement sur votre poste ; seule la clé publique est autorisée sur le serveur.



INFOGRAPHIE *Schéma des permissions SSH Linux pour le dossier .ssh et les clés publiques ou privées.*

Parcours de correction en 5 étapes



1 **1. Valider la cible de connexion**
Vérifiez l'utilisateur distant, le nom du serveur et, si nécessaire, le port communiqué. Dans « ssh utilisateur@nom-du-serveur », l'utilisateur est le compte Linux du serveur, pas votre nom local.

2 **2. Identifier la clé privée locale**
Listez les fichiers SSH avec « ls -la ~/.ssh ». Le fichier sans suffixe .pub est la clé privée ; le fichier .pub correspondant est la clé publique. Ne générez pas une nouvelle paire avant d'avoir vérifié les clés existantes.

3 **3. Charger explicitement la clé dans l'agent**
Démarrez l'agent avec « eval "$(ssh-agent -s)" », puis ajoutez la clé avec « ssh-add ~/.ssh/id_ed25519 ». Contrôlez les identités chargées avec « ssh-add -l -E sha256 ».

4 **4. Tester la clé choisie**
Forcez temporairement la clé à utiliser : « ssh -i ~/.ssh/id_ed25519 utilisateur@nom-du-serveur ». Si ce test réussit alors que la commande simple échoue, la sélection automatique de clé ou l'agent est probablement en cause.

5 **5. Contrôler l'autorisation côté serveur**
Avec un accès d'administration alternatif, comparez la clé publique locale affichée par « cat ~/.ssh/id_ed25519.pub » avec une ligne du fichier « ~/.ssh/authorized_keys » du compte distant visé. La ligne doit être complète et rester sur une seule ligne.

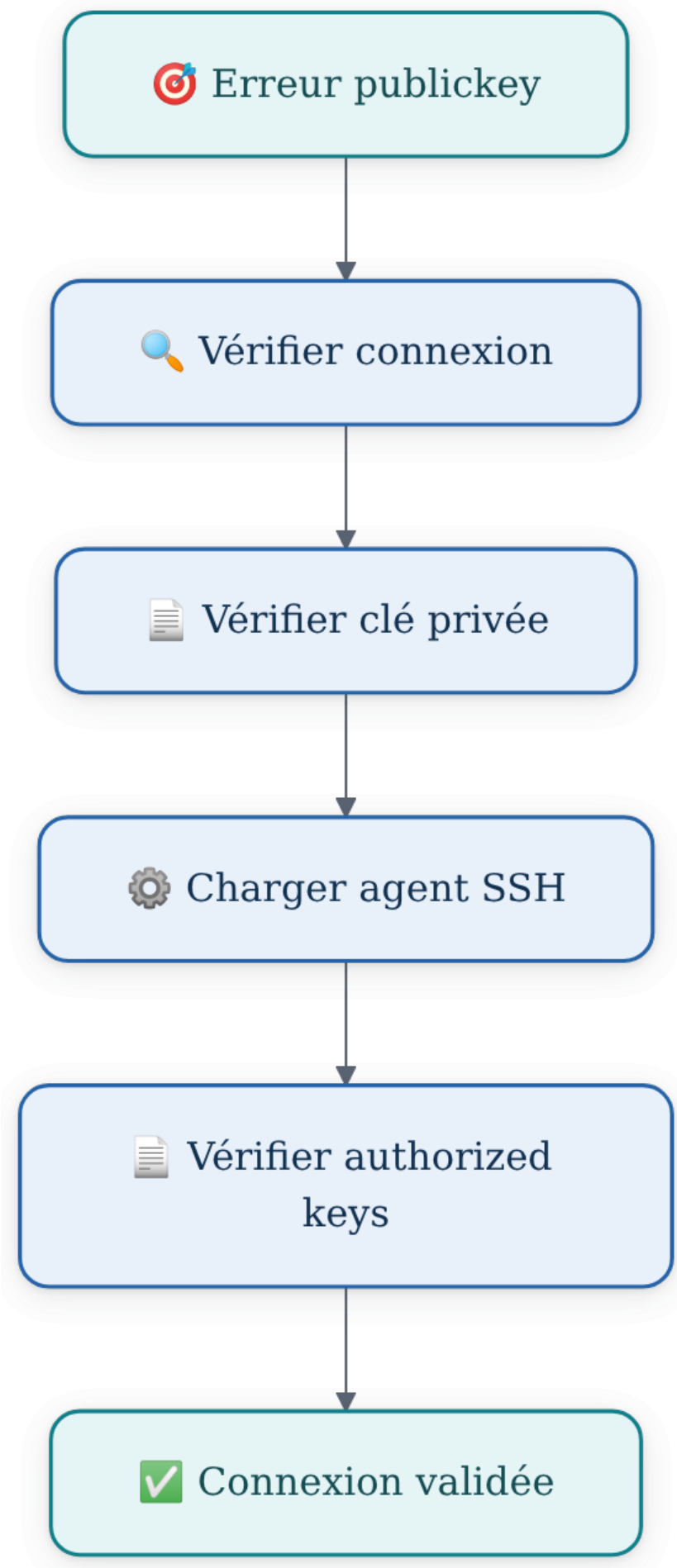

🎯 Erreur publickey
🔍 Vérifier connexion
📄 Vérifier clé privée
⚙️ Charger agent SSH
📄 Vérifier authorized keys
✅ Connexion validée

SCHÉMA *Chemin de diagnostic d'un refus d'authentification SSH par clé publique.*

Repères de permissions à vérifier

Élément	Droit recommandé	Commande de vérification
Dossier ~/.ssh	700	ls -ld ~/.ssh
Clé privée	600	ls -l ~/.ssh/id_ed25519
Clé publique	644	ls -l ~/.ssh/id_ed25519.pub

Checklist finale avant de réessayer

- [ ] **Le compte indiqué avant @ est bien le compte Linux distant attendu.**
- [ ] **Le port SSH est précisé avec -p si le serveur n'utilise pas le port habituel.**
- [ ] **La clé privée ciblée existe localement et n'a pas de suffixe .pub.**
- [ ] **La bonne clé est chargée dans ssh-agent ou testée avec l'option -i.**
- [ ] **La clé publique correspondante est présente dans authorized_keys du bon compte distant.**
- [ ] **Le dossier .ssh, les clés et leurs propriétaires respectent les droits attendus.**
- [ ] **Un test détaillé est lancé avec « ssh -v utilisateur@nom-du-serveur » pour voir la clé réellement proposée.**

> ! **Ne contournez pas le problème en ouvrant les droits**
> N'élargissez pas les permissions du dossier SSH ou des clés pour faire disparaître l'erreur : OpenSSH peut précisément refuser des fichiers trop exposés. Modifiez uniquement les fichiers appartenant à votre compte et ne communiquez jamais votre clé privée.

Cas à ne pas confondre

Un avertissement concernant l'empreinte ou l'identité du serveur ne correspond pas à « Permission denied (publickey) » : il intervient avant l'authentification. Pour GitHub, utilisez l'utilisateur distant « git » et testez la connexion avec « ssh -T git@github.com ».

Avant toute modification côté serveur, utilisez une voie d'administration autorisée et vérifiez que vous intervenez sur le bon compte distant.