Domino API – Rotation automatique des tokens via Service Account & HashiCorp Vault

Version: 1.0 | Auteur: Équipe DevOps / Datalab | Dernière mise à jour: aujourd'hui

Contexte: Sécuriser l’accès à l’API Domino par rotation périodique des tokens de Service Account et dépôt sécurisé du secret dans HashiCorp Vault.

# 1. Objectifs

- Éliminer les tokens statiques et réduire la fenêtre d’exposition.

- Automatiser la création/rotation du token API Domino pour un Service Account.

- Pousser le nouveau token dans HashiCorp Vault (KV) pour consommation par les jobs/applications.

- Auditer, tracer et simplifier l’exploitation.

# 2. Portée

Cette procédure couvre : Domino Datalab (Jobs planifiés), l’API Domino v4 (Service Accounts & Tokens), et HashiCorp Vault (auth cert mTLS + KV v2/v1).

# 3. Architecture & flux

+---------------------------+ +----------------------+ +-------------------------+  
| Domino Scheduled Job | calls | Domino API v4 | stores | HashiCorp Vault (KV) |  
| rotate.py (Python) +--------->| /serviceAccounts/\* +---------->| secret/domino-api/... |  
| - crée/rotate token | token | /tokens | token | (versioning automatique)|  
+---------------------------+ +----------------------+ +-------------------------+

# 4. Prérequis

- Accès Domino avec une clé API autorisée à gérer les Service Accounts et leurs tokens.

- Service Account Domino existant (ou créé par le script).

- Accès HashiCorp Vault (Enterprise ou OSS) avec méthode d’authentification cert (mTLS) opérationnelle.

- Rôle/policy Vault autorisant l’écriture sur le chemin KV cible.

- Chaîne de confiance TLS (CA) disponible pour vérifier Vault.

# 5. Variables d’environnement utilisées

# Domino  
DOMINO\_URL="https://<domino-host>/v4/serviceAccounts"  
DOMINO\_API\_KEY="<domino\_api\_key>"  
SA\_ID="<uuid\_du\_service\_account>" # ou SA\_NAME/SA\_EMAIL si création/lookup  
  
# Vault (mTLS + Namespace)  
VAULT\_ADDR="https://hvault.staging.echonet:8200"  
VAULT\_NAMESPACE="UPM\_CARDIF/BF05/EC002I002860" # si Enterprise avec namespaces  
VAULT\_CLIENT\_CERT="/etc/vault/certs/client.crt"  
VAULT\_CLIENT\_KEY="/etc/vault/certs/client.key"  
VAULT\_CACERT="/etc/ssl/certs/ca\_chain.crt"  
VAULT\_KV\_MOUNT="secret"  
VAULT\_SECRET\_PATH="domino-api/api-domino-run-jobs"

# 6. Script de rotation – résumé

Le script Python exécute les étapes suivantes :

1) Génére un nouveau token pour le Service Account Domino (POST /v4/serviceAccounts/{id}/tokens).

2) Authentifie sur Vault via auth/cert/login pour obtenir un client\_token.

3) Écrit le secret dans KV v2 (ou v1 en fallback) au chemin configuré, en mettant à jour la clé api\_domino\_run\_jobs.

4) (Optionnel) Invalide l’ancien token si un nom constant est utilisé côté Domino.

## 7. Exemple minimal d’écriture dans Vault (KV v2)

# login cert -> récupère client\_token  
POST $VAULT\_ADDR/v1/auth/cert/login  
Headers: X-Vault-Request: true, X-Vault-Namespace: $VAULT\_NAMESPACE (si applicable)  
mTLS: --cert $VAULT\_CLIENT\_CERT --key $VAULT\_CLIENT\_KEY  
-> retourne .auth.client\_token  
  
# write KV v2 (nouvelle version)  
POST $VAULT\_ADDR/v1/secret/data/domino-api/api-domino-run-jobs  
Headers: X-Vault-Token: <client\_token>, Content-Type: application/json, X-Vault-Namespace: ...  
Body:  
{  
 "data": {  
 "api\_domino\_run\_jobs": "<NOUVEAU\_TOKEN\_DOMINO>"  
 }  
}

# 8. Policy Vault (KV v2 – exemple)

# Mount: secret (KV v2)  
path "secret/data/domino-api/\*" {  
 capabilities = ["create", "update", "read", "list"]  
}  
path "secret/metadata/domino-api/\*" {  
 capabilities = ["read", "list"]  
}

# 9. Configuration du Job planifié Domino

- Projet : Datalab / <votre-projet>

- Script : rotate.py (dans le repo/projet Domino)

- Hardware Tier : minimal suffisant (ex. 1 CPU / 1–2 Go RAM).

- Fréquence : recommandation tous les 80 jours (TTL tokens Domino ≈ 90 jours).

- Variables d’environnement : cf. section 5 (définies dans le Job/Workspace ou via Vault).

- Logs : conserver la sortie standard; activer l’alerte en cas d’échec (HTTP != 2xx).

# 10. Procédure d’exploitation

• Forcer une rotation manuelle : lancer le Job planifié immédiatement.

• Vérifier la version du secret : via l’UI Vault (onglet Version History) ou API GET /v1/secret/metadata/... (KV v2).

• Consommateurs : s’authentifier auprès de Vault et lire la clé api\_domino\_run\_jobs. Mettre à jour le cache applicatif si nécessaire.

# 11. Tests de validation (exemples)

# 1) Login cert  
curl --cert $VAULT\_CLIENT\_CERT --key $VAULT\_CLIENT\_KEY --cacert $VAULT\_CACERT \  
 -H "X-Vault-Request: true" -H "X-Vault-Namespace: $VAULT\_NAMESPACE" \  
 -X POST "$VAULT\_ADDR/v1/auth/cert/login"  
  
# 2) Lire KV v2  
curl --cacert $VAULT\_CACERT --cert $VAULT\_CLIENT\_CERT --key $VAULT\_CLIENT\_KEY \  
 -H "X-Vault-Token: $CLIENT\_TOKEN" -H "X-Vault-Namespace: $VAULT\_NAMESPACE" \  
 "$VAULT\_ADDR/v1/secret/data/domino-api/api-domino-run-jobs"  
  
# 3) Appel Domino avec le nouveau token (exemple générique)  
curl -H "X-Domino-Api-Key: <api\_domino\_run\_jobs>" \  
 "https://<domino-host>/v4/whoami"

# 12. Dépannage (FAQ)

403 Forbidden sur Vault :

- Vérifier que le login cert renvoie bien un client\_token (namespace correct).

- Vérifier la policy pour secret/data/domino-api/\* (KV v2).

- Vérifier la chaîne TLS (CA) et que le mTLS est bien transmis sur toutes les requêtes (login + write).

400/404 sur /data/… : le mount n’est peut‑être pas en KV v2 → utiliser le fallback KV v1 /v1/secret/<path> sans enveloppe {"data": ...}.

401/403 Domino : clé API invalide ou rôle du Service Account insuffisant pour créer des tokens.

# 13. Sécurité & conformité

- Rotation périodique < 90 jours (TTL).

- Pas d’exposition du secret en clair dans les logs (masquage recommandé).

- Stockage chiffré au repos (Vault).

- RBAC Domino/Vault revu régulièrement.

- Gestion des certificats mTLS (renouvellement, révocation).

# 14. Journalisation & Audit

- Logs Domino Jobs : succès/échec, statut HTTP, version Vault écrite.

- Audit Vault : auth/cert/login, writes sur secret/data/....

- Conserver l’historique des versions (KV v2) pour rollback si nécessaire.

# 15. Change Log

v1.0 - Mise en place de la rotation automatique, écriture KV v2, job planifié 80j.