

Documentation – Scripts Azure Update Manager

Objectif

Ces scripts permettent d'automatiser la configuration de **Windows Update Manager (In-Guest Patching)** pour une VM Linux dans Azure.

L'idée est de définir une **fenêtre de maintenance hebdomadaire** (ici chaque dimanche à 02h00 UTC) où seuls les patchs de sécurité et critiques seront appliqués, sans redémarrage automatique.

Un script complémentaire permet de supprimer facilement cette configuration.

Scripts

1. `azure-update-manager-setup.sh`

 Configure et assigne une fenêtre de maintenance à la VM.

Étapes principales

1. Création / mise à jour d'une Maintenance Configuration
2. Mise à jour de la VM pour patching par la plateforme
3. Assignation de la configuration à la VM
4. Vérifications de la configuration et de l'assignation

Commandes utiles

- Vérifier la politique active :

```
az vm show -g rg-viona -n viona-mvp-gpu --query  
osProfile.linuxConfiguration.patchSettings -o jsonc
```

- Évaluer les patchs disponibles :


```
az vm assess-patches -g rg-viona -n viona-mvp-gpu -o jsonc
```

- Lancer un patch manuel (hors fenêtre) :

```
az vm install-patches -g rg-viona -n viona-mvp-gpu  
--maximum-duration PT2H  
--reboot-setting Never  
--linux-parameters '{"classificationsToInclude":["Security","Critical"]}'
```

 **À exécuter une seule fois dans la VM Ubuntu :**

```
sudo systemctl disable --now apt-daily.timer
sudo systemctl disable --now apt-daily-upgrade.timer
sudo systemctl disable --now unattended-upgrades
sudo apt remove -y unattended-upgrades
```

 **Hold** : Geler les drivers sensibles (NVIDIA, CUDA, kernel) :

```
sudo apt-mark hold 'nvidia-*' 'cuda-*' 'nvidia-dkms-*' 'linux-image-*'
'linux-headers-*
```

2. `azure-update-manager-cleanup.sh`

➡ Supprime l'assignation et la configuration créées.

Étapes principales

1. Suppression de l'assignation
2. Suppression de la Maintenance Configuration

Commandes du script `azure-update-manager-cleanup.sh`

- **Suppression de l'assignation** (retire le lien entre la VM et la Maintenance Configuration) :

```
az rest --method delete
--url "https://management.azure.com${ASSIGN_ID}?api-version=${API_ASSIGN}"
```

- **Suppression de la Maintenance Configuration** (supprime l'objet de configuration) :

```
az rest --method delete
--url "https://management.azure.com${MC_ID}?api-version=${API_CFG}"
```

 **À savoir** :

- `ASSIGN_ID` pointe vers `/virtualMachines/<VM>/providers/Microsoft.Maintenance/configurationAssignments/<nom-assignation>`.
- `MC_ID` pointe vers `/resourceGroups/<RG>/providers/Microsoft.Maintenance/maintenanceConfigurations/<nom-config>`.
- Les variables `API_ASSIGN` et `API_CFG` définissent les versions d'API Azure utilisées.

Exemple d'utilisation

 Avant exécution, donner les droits d'exécution aux scripts :

```
chmod +x azure-update-manager-setup.sh azure-update-manager-cleanup.sh
```

1. Déployer et configurer la fenêtre de patching :

```
./azure-update-manager-setup.sh
```

1. Vérifier que la configuration et l'assignation sont bien créées.
2. Si besoin de tout nettoyer (rollback) :

```
./azure-update-manager-cleanup.sh
```