SWMB, un logiciel libre et modulaire pour sécuriser votre parc Windows

Gabriel Moreau, Olivier De-Marchi

Laboratoire LEGI - UMR5519

7 décembre 2023 / Paris

Origine du projet

- Contexte de l'**ESR** (Enseignement Supérieur et la Recherche)
- Existence de scripts PowerShell au LEGI / GPO sur Active Directory à la DR11 CNRS
- Création d'un **Groupe de Travail** fin 2019 au sein de RESINFO
  (Réseau métiers des ASR du supérieur - Administrateurs Systèmes et Réseaux)
- Réalisation d'une première maquette de faisabilité en 2020
- Prise de contact avec l'ANSSI
- Laisser l'**ASR autonome à 100%** sur la politique de sécurité de son unité de recherche
- **Aider et mutualiser** le travail des ASR par une meilleure collaboration

Resinfo cnrs

SWMB (Secure Windows Mode Batch)

- Besoin de sécuriser Microsoft Windows 10 (et 11)
- **Outil modulaire** avec des **règles** et des **anté-règles** (pourvoir faire et défaire)
- Outil sans état « comme » un gestionnaire de configuration sous GNU/Linux (`cfengine`, `puppet`, `ansible...`)
- SWMB peut-être lancé plusieurs fois à l'identique (*idempotent*)
- **Outil en production** au LEGI sur tous les postes informatiques
- Pas d'interaction avec l'utilisateur, bien tester sur quelques postes avant de trop déployer
- Packaging pour **simplifier** son propre **déploiement** : `setup.exe` (NSIS), OCS, WAPT, PDQ Deploy

- Ne pas réinventer la roue, choix d'une **licence libre**
- Point de départ, le projet «Win10-Initial-Setup-Script» par Disassembler0 (licence MIT)

SWMB et Active Directory

- SWMB n'est pas incompatible avec l'Active Directory
- Il y a presque toujours des machines hors AD dans un parc machine (serveur de badge, automate GTC...). Comment gérez-vous ces machines au cours du temps ?
- Fichiers de **configuration au format texte** donc auto-documenté
- SWMB permet de garder dans une **arborescence Git** (GitLab) l'ensemble des configurations au cours du temps et quelle personne a poussé (validé) une modification
- Avec SWMB, on sait quelles actions sont lancées et quand
- Cet ensemble permet de répondre à un objectif de **qualité des règles** sur son parc au **cours du temps**

Vocabulaire - Trois concepts principaux

- Les **tweaks** sont des règles de base dans SWMB. En général, chaque tweak a son pendant. L'un fait, l'autre défait (`Enable` / `Disable` par exemple)
- Les **presets** sont des fichiers regroupant en leur sein un ensemble de tweaks. SWMB propose ainsi plusieurs jeux de preset, ceux-ci sont régulièrement mis à jour par la communauté
- Les **modules** sont les implémentations des tweaks en PowerShell. Chaque module regroupe en général le code source de plusieurs tweaks, classés par grande catégorie

Le code SWMB importe les modules « à chaud » avant de traiter les tweaks définis dans les presets un par un

Organisation du code - Les **modules**

- Le dossier `Modules` regroupe le module principal `SWMB.psm1` qui intègre les routines du cœur des algorithmes
- Les sous modules sont placés dans le sous-dossier `Modules\SWMB`
  - Le code concernant l'implémentation des **tweaks** de l'ordinateur (LocalMachine)
  - Le code concernant l'implémentation des **tweaks** de l'utilisateur courant (CurrentUser, extension ≪_CU≫)
- Exemple :
  - `Modules/SWMB/CurrentUser-Application.psm1`
  - `Modules/SWMB/CurrentUser-Privacy.psm1`
  - `Modules/SWMB/LocalMachine-Network.psm1`
  - `Modules/SWMB/LocalMachine-Privacy.psm1`
  - `Modules/SWMB/LocalMachine-Security.psm1`

Organisation du code - Les **tweaks**

- Les **tweaks** sont souvent implémentés avec 3 fonctions PowerShell
- Exemple avec ClearPageFile (nettoyer le fichier `PAGEFILE.SYS` lors de l'arrêt de la machine)
  - Fonction `TweakEnableClearPageFile` - tweak **EnableClearPageFile**
  - Fonction `TweakDisableClearPageFile` - tweak **DisableClearPageFile**
  - Fonction `TweakViewClearPageFile` - voir dans quel état nous sommes (débogage)
- Par précaution et sécurité, toutes les fonctions doivent commencer par le préfixe **Tweak**
- Ainsi, **SWMB n'exécute pas n'importe quel code PowerShell**

Exemple des tweaks ClearPageFile

```
# ClearPageFileAtShutdown
# https://deployadmin.com/2019/11/03/vider-le-fichier-dechange-a-chaque-arret-de-windows/
# Enable
Function TweakEnableClearPageFile { # RESINFO
  Write-Output "Clear PageFile.sys at shutdown..."
  Set-ItemProperty -Path "HKLM:\SYSTEM\CurrentControlSet\Control\Session Manager\Memory Management" -Name "
      ClearPageFileAtShutdown" -Type DWord -Value 1
}

# Disable
Function TweakDisableClearPageFile { # RESINFO
  Write-Output "Do not reset PageFile.sys at shutdown..."
  Set-ItemProperty -Path "HKLM:\SYSTEM\CurrentControlSet\Control\Session Manager\Memory Management" -Name "
      ClearPageFileAtShutdown" -Type DWord -Value 0
}

# View
Function TweakViewClearPageFile { # RESINFO
  Write-Output 'Clear PageFile.sys (0 nothing enable, 1 clear at shutdown)'
  $KeyPath = "HKLM:\SYSTEM\CurrentControlSet\Control\Session Manager\Memory Management"
  Get-ItemProperty -Path $KeyPath -Name "ClearPageFileAtShutdown"
}
```

Listing 1: clear-page-file.ps1

Type de **tweaks**

- `Enable / Disable`
- `Show / Hide`
- `Install / Uninstall`
- `Add / Remove`
- `Set / Unset`
- `SysMessage, SysRestart, SysRequireAdmin...`
- `View`

- Par défaut, les **tweaks n'ont aucun paramètre**
- Certains tweaks se configurent via une variable globale
  (Cela permet par exemple de faire passer en paramètre le serveur de temps)
- Les tweaks pour modifier la configuration de l'utilisateur courant finissent par « `_CU` »

Bilan du nombre de **tweaks**

Status	**Number of tweaks**			
Info	Number of RESINFO tweaks			132
Info	Number of Current User tweaks (_CU)			151
Info	Number of Enable and Disable tweaks	190	190	380
Warn	Number of Install and Uninstall tweaks	21	33	54
Warn	Number of Show and Hide tweaks	56	53	109
Info	Number of Add and Remove tweaks	3	3	6
Warn	Number of Set and Unset tweaks	41	10	51
Warn	Number of Pin and Unpin tweaks	0	2	2
Info	Number of total tweaks GPO			602
Info	Number of Sys tweaks (system)			9
Info	Number of View tweaks (debug)			32
Info	Number of Obsolete tweaks			3
Info	Number of total tweaks functions			646

Organisation du code - Les **presets**

- Les **presets** sont des fichiers de configuration à déployer
- Ce sont des listes de tweaks, un par ligne
- # est le caractère de commentaire
- Il y a des exemples de presets vérifiant le document de l'ANSSI du 6 juillet 2017
- Le tweak recommandé est en général écrit en premier sur une ligne
- L'anté-tweak, s'il existe, est toujours commenté à sa suite

Exemple de fichier de **presets**

```
SysRequireAdmin

### Privacy Tweaks ###
DisableTelemetry                # EnableTelemetry
DisableCortana                  # EnableCortana
DisableActivityHistory          # EnableActivityHistory         # ViewActivityHistory

### UWP Privacy Tweaks ###
DisableUWPBackgroundApps        # EnableUWPBackgroundApps
# DisableUWPVoiceActivation     # EnableUWPVoiceActivation

### Service Tweaks ###
# SetTargetRelease              # UnsetTargetRelease
DisableAppSuggestions           # EnableAppSuggestions          # ViewActivityHistory
# SetP2PUpdateLocal             # SetP2PUpdateInternet          # SetP2PUpdateDisable
DisableDiagTrack                # EnableDiagTrack               # ViewDiagTrack

### Security Tweaks ###
DisableAdminShares              # EnableAdminShares
EnableASLR                      # DisableASLR                   # ViewASLR
# EnableClearPageFile           # DisableClearPageFile          # ViewClearPageFile
```

Listing 2: LocalMachine-All.preset

Utilisation via les tâches planifiées

Lors de l'installation, SWMB propose de configurer 3 tâches planifiées :

- **LocalMachine-Boot.ps1** - Tâche se lançant au démarrage de la machine
  Problème, beaucoup d'utilisateur reboot peu souvent leur machine
- **LocalMachine-PostInstall.ps1** - Tâche se lançant en asynchrone après l'installation
  Permet de forcer des réglages de suite
- **CurrentUser-Logon.ps1** - Tâche se lançant à l'ouverture de la session utilisateur
- En pratique, il y a rarement besoin de lancer SWMB manuellement sur un poste

Utilisation via une interface graphique minimaliste

- Lancer le script interactif de chiffrement de tous les lecteurs avec BitLocker
- Suspendre ou Reprendre BitLocker
- Exécuter immédiatement une des tâches programmées
- Indiquer la présence d'une mise à jour disponible de SWMB
- Lister l'ensemble des logiciels installés dans les ruches HKLM, HKCU et HKU

Activer le chiffrement BitLocker

- Avant le chiffrement, BitLocker est configuré via
  - Les clefs de registre `HKLM:\SOFTWARE\Policies\Microsoft\FVE`
  - Forcer l'algorithme de chiffrement XtsAes256, `EncryptionMethodWithXtsOs = 7`
  - Interdire à l'utilisateur de modifier le code PIN, `DisallowStandardUserPINReset = 1`
  - ...
- Clefs de chiffrement / déchiffrement stockés sur le disque système
- Des droits particuliers sont appliqués sur les fichiers contenant ces clefs
  (lecture impossible, copie par un compte administrateur)
- À charge à chacun de sauver ces clefs dans un coffre-fort centralisé
  (container VeraCryt par exemple)

SWLN : Secure Windows Local Network

- Comment utiliser SWMB dans son unité, sur son site ?
- Déployer SWMB tel quel en appliquant le jeu de preset par défaut du GT RESINFO (règles de l'ANSSI plus quelques autres)
- Étendre SWMB
  - SWMB est un framework qui exécute des fonctions PowerShell
  - Écrire son code spécifique sous forme de fonctions TweakEnable, TweakSet. . . dans un module

SWLN au LEGI

- `CurrentUser-Logon.preset`
- `LocalMachine-Boot.preset`
- `LocalMachine-PostInstall.preset`
- `Local-Addon.psm1`
- `Custom-VarOverload.psm1` - paramètre global pour quelques tweaks
- `install.bat`
- `post-install.ps1`
- `uninstall.bat`
- `Makefile` - créer le Zip qui va bien pour OCS Inventory
- `print` - dossier avec les drivers des photocopieurs

Exemple de paramètres globaux

```
# NTP
$Global:SWMB_Custom.NTP_ManualPeerList = 'XXX.XXX.XXX.XXX'

# Target Release
$Global:SWMB_Custom.ProductVersion           = 'Windows 10'
$Global:SWMB_Custom.TargetReleaseVersionInfo = '21H2'
```

Listing 3: Custom-VarOverload.psm1

Ajouter ses propres règles

- En général, un tweak (ou une GPO) revient à modifier la valeur d'une clef de registre
- On trouve presque toujours la solution sur internet et sinon...
  la clef de registre est capturée avec ProcessMonitor de Sysinternals !
- Ces sites internet proposent une vue similaire à celle de « gpedit »
- Ils sont très complets et permettent de rechercher via de nombreux filtres
  - Group Policy Search (`gpsearch`)
  - Group Policy Administrative Templates Catalog (`admx.help`)
  - Tableur Excel de Microsoft contenant toutes les GPO des systèmes d'exploitation

Conclusion - **SWMB**

- **Programme libre**, **modulaire** et **collaboratif**
- Chaque ASR l'adapte à son contexte et ne pousse que les tweaks qu'il souhaite (**autonomie**, rien n'est obligatoire)
- **Fonctionne** en production
- Il est facile de modifier ses propres scripts PowerShell pour les intégrer dans cet environnement
- Le projet à besoin des ASR pour progresser et intégrer des nouvelles fonctionnalités
- Prenez **votre sécurité en main** en **partageant** aussi votre **savoir faire**

- SWMB n'a pas les mêmes objectifs que tous les programmes graphiques qui vous proposent des tweaks

Licence

Merci à toutes les personnes et entités
nous ayant aidés ou ayant participé depuis le début

Cette présentation est sous : Licence Art Libre

http://artlibre.org/