Liberté • Égalité • Fraternité
RÉPUBLIQUE FRANÇAISE

PREMIER MINISTRE

Secrétariat général
de la défense
et de la sécurité nationale

*Agence nationale de la sécurité
des systèmes d'information*

Paris, le 13 janvier 2017

N° DAT-NT-13/ANSSI/SDE/NP

Nombre de pages du document
(y compris cette page) : 22

Note technique

Recommandations pour la mise en œuvre d'une politique de restrictions logicielles sous Windows

Public visé :

Développeur	
Administrateur	✓
RSSI	✓
DSI	✓
Utilisateur	

Informations

Avertissement

Ce document rédigé par l'ANSSI présente les **« Recommandations pour la mise en œuvre d'une politique de restrictions logicielles sous Windows »**. Il est téléchargeable sur le site www.ssi.gouv.fr. Il constitue une production originale de l'ANSSI. Il est à ce titre placé sous le régime de la « Licence ouverte » publiée par la mission Etalab (www.etalab.gouv.fr). Il est par conséquent diffusable sans restriction.

Ces recommandations sont livrées en l'état et adaptées aux menaces au jour de leur publication. Au regard de la diversité des systèmes d'information, l'ANSSI ne peut garantir que ces informations puissent être reprises sans adaptation sur les systèmes d'information cibles. Dans tous les cas, la pertinence de l'implémentation des éléments proposés par l'ANSSI doit être soumise, au préalable, à la validation de l'administrateur du système et/ou des personnes en charge de la sécurité des systèmes d'information.

Personnes ayant contribué à la rédaction de ce document :

Contributeurs	Rédigé par	Approuvé par	Date
BAS, BAI, BSS	BAS, BSS	SDE, Comité éditorial	13 janvier 2017

Évolutions du document :

Version	Date	Nature des modifications
1.0	6 décembre 2013	Version initiale
2.0	13 janvier 2017	Compléments concernant les applications universelles et les considérations de sécurité

Pour toute question :

Contact	Adresse	@mél
Bureau Communication de l'ANSSI	51 bd de La Tour-Maubourg 75700 Paris Cedex 07 SP	conseil.technique@ssi.gouv.fr

Table des matières

1 Préambule

L'intérêt principal des mécanismes de restrictions logicielles réside dans la possibilité de limiter l'exécution des programmes à une liste de programmes dûment autorisés (liste blanche). Le principe d'une liste blanche procure plusieurs avantages :

- une meilleure protection contre les programmes malveillants : en empêchant systématiquement l'exécution des programmes non répertoriés dans la liste, ceux susceptibles de contenir un code malveillant sont également bloqués, que ce dernier soit connu ou non des bases de signature de l'antivirus ;
- un blocage de l'installation ou de l'utilisation de logiciels indésirables, en particulier ceux qui sont susceptibles de porter atteinte aux performances ou de rendre instables les systèmes, et qui dans tous les cas augmentent la surface d'attaque de celles-ci ;
- un blocage de l'installation ou de l'utilisation de logiciels sans licence.

Sur un système à jour de ses correctifs de sécurité et respectant le principe de séparation des privilèges, l'activation des mécanismes de restrictions logicielles augmente la maîtrise du système. Enfin, en règle générale, l'activation des mécanismes de restrictions logicielles n'engendre pas d'altération des performances du système, cette mesure est totalement transparente pour l'utilisateur.

Bien que les mécanismes de restriction logicielle puissent s'appliquer sur des systèmes serveurs, c'est essentiellement sur les postes de travail utilisateurs qu'ils présentent un intérêt. Il s'agit du périmètre de déploiement abordé dans ce document.

Recommandations

2 Les mécanismes SRP et AppLocker

2.1 Différences entre SRP et AppLocker

À partir de Windows XP et Windows Server 2003, Microsoft introduit les SRP (*Software Restriction Policies*) permettant la mise en œuvre d'une politique de restrictions logicielles. Grâce aux SRP, il devient possible de restreindre l'exécution de programmes sur le poste de l'utilisateur en définissant une liste blanche à l'aide de règles particulières.

Bien configurées, les SRP offrent une protection efficace. Néanmoins, leur configuration peut être assez lourde à maintenir dans un environnement dynamique, en particulier sur un grand parc de postes. À cet égard, AppLocker qui est une évolution des SRP, apporte de nettes améliorations.

Une différence fondamentale entre SRP et AppLocker est le champ d'application des règles. Avec SRP, tous les utilisateurs d'une machine sont affectés indifféremment par les règles. Avec AppLocker, il est possible de cibler un utilisateur précis d'une machine ou d'un domaine.

Pour pouvoir bénéficier d'AppLocker sur les postes utilisateur, il est impératif de disposer de Windows 7 Entreprise ou Intégrale ou de Windows 8, 8.1 ou 10 Entreprise. Les éditions professionnelles de Windows 7, 8, 8.1 et 10 permettent de configurer des règles AppLocker, mais ces dernières sont inopérantes. Ainsi, AppLocker peut être activé sur les systèmes suivants :

- Windows 7 éditions Entreprise et Intégrale ;
- Windows 8, 8.1 et 10 édition Entreprise ;
- Windows Server 2008 R2 éditions Standard, Entreprise, Datacenter et pour les systèmes Itanium ;
- Windows Server 2012 et 2012 R2 éditions Standard et Datacenter.

Windows 7, 8, 8.1 et 10 supportent toujours les SRP, mais si une stratégie configure simultanément des règles SRP et AppLocker, seules les règles AppLocker seront prises en compte.

R1 - Mettre en œuvre une politique de restrictions logicielles fine avec AppLocker

Pour mettre en œuvre une politique de restrictions logicielles fine, il est préférable d'utiliser Applocker plutôt que SRP.

Dans la suite de ce document, seul l'usage d'AppLocker est abordé.

2.2 Fonctionnement d'AppLocker

Avant de détailler les différents mécanismes de restrictions logicielles d'AppLocker, il est nécessaire d'en comprendre le fonctionnement, d'identifier les avantages et inconvénients des différentes règles mais aussi la manière dont elles interagissent entre elles. Ce paragraphe est une synthèse des éléments fournis par Microsoft sur le site `technet.microsoft.com` que le lecteur est invité à consulter pour tout complément d'informations ou toute précision sur le fonctionnement d'AppLocker.

AppLocker permet d'« AUTORISER » ou de « REFUSER » à un utilisateur (ou un groupe d'utilisateurs) le lancement de programmes sous différentes formes :

– exécutables : fichiers `.exe` et `.com` ;
– Windows Installer : fichiers `.msi` et `.mst` ;
– scripts : fichiers `.ps1` (powershell), `.bat`, `.cmd`, `.vbs`, `.js` ;
– bibliothèques : fichiers `.dll` et `.ocx` ;
– applications universelles `.appx`.

Pour déterminer si un programme est autorisé à s'exécuter ou non, AppLocker évalue d'abord les règles du type « REFUSER » puis celles du type « AUTORISER ». Il est possible de combiner des règles « AUTORISER » et « REFUSER ». Pour une règle donnée, peuvent être définies des exceptions. Lorsqu'un programme ne fait l'objet d'aucune règle du type « AUTORISER », il est automatiquement bloqué (refus implicite).

Lorsqu'un programme est bloqué par Applocker, l'utilisateur en est informé par un message d'erreur comme *« Ce programme a été bloqué par une stratégie de groupe, veuillez contacter votre administrateur »*. Ce message peut être personnalisé en modifiant un paramètre d'une GPO [1] [2], un lien « *Plus d'informations* » dont le contenu est paramétrable apparaît. Cette possibilité peut être utilisée par exemple pour rediriger l'utilisateur vers l'ouverture d'un ticket d'incident.

R2 - Privilégier les règles AppLocker du type « AUTORISER »

> Pour une meilleure lisibilité du comportement d'AppLocker, il est préférable de n'utiliser que des règles du type « AUTORISER » avec si nécessaire des exceptions.

L'autorisation ou le refus d'exécution d'un programme est conditionné à la vérification de règles pour lesquelles trois types différents existent :
– les règles basées sur le chemin d'accès, qui permettent d'autoriser ou de refuser l'exécution de fichiers se trouvant dans le répertoire et les sous-répertoires du chemin. Pour désigner les répertoires classiques du système de fichiers, AppLocker utilise des variables qui sont différentes des variables d'environnement de Windows (voir le tableau de correspondance dans l'annexe A). Ces règles seront appelées de type *Chemin d'accès* dans la suite ;
– les règles reposant sur une signature électronique, qui permettent d'autoriser seulement les fichiers signés par un éditeur donné, et répondant éventuellement à d'autres critères comme le nom du produit, le nom du fichier et sa version. Ces règles seront appelées de type *Éditeur* dans la suite ;
– les règles basées sur l'empreinte cryptographique (`sha256`) d'un fichier, qui n'autorisent que le fichier correspondant à l'empreinte. Ces règles seront dénomées de type *Empreinte* dans la suite.

Les règles basées sur le chemin d'accès offrent une grande souplesse, mais exigent en contre-partie la maîtrise dans le temps du contenu et des autorisations des répertoires associés afin de s'assurer que seuls des programmes légitimes peuvent s'y trouver. En général, la mise à jour d'un logiciel n'oblige pas à modifier les règles existantes.

Les règles basées sur une signature électronique obtenue à l'aide de certificats de confiance offrent quant à elles plus de sécurité et, selon leur configuration, une souplesse à géométrie variable. Les mises à jour des programmes sont en général transparentes.

Les règles basées sur des empreintes offrent le meilleur niveau de sécurité car elles n'autorisent que les fichiers correspondant à l'empreinte cryptographique. Par contre, lors de la mise à jour d'un logiciel, les règles doivent la plupart du temps être modifiées.

1. *Group Policy Object* ou stratégie de groupe. Pour plus d'informations sur les GPO, lire l'article « Stratégie de groupe pour les débutants » sur le site de Microsoft à l'adresse https://technet.microsoft.com/fr-fr/library/hh147307(v=ws.10).aspx

2. Le paramètre est accessible dans l'arborescence suivante : *Configuration ordinateur* → *Stratégies* → *Modèles d'administration* → *Composants Windows* → *Explorateur Windows* → *Définir le lien d'une page web de support.*

De façon analogue aux règles, les exceptions peuvent s'appliquer à un répertoire, à une signature électronique ou à une empreinte de fichier, et ce quel que soit le type de la règle à laquelle elles sont rattachées.

R3 - Privilégier les règles basées sur la signature électronique

> Lorsque cela est possible, il convient d'utiliser des règles basées sur la signature électronique pour autoriser ou refuser l'exécution d'un programme en s'étant assuré au préalable que les certificats et les autorités de certification sont de confiance.

3 Mise en œuvre d'une stratégie de restrictions logicielles avec AppLocker

Les principales étapes pour le déploiement d'une stratégie de restrictions logicielles sont les suivantes :

- réaliser un inventaire des applications utilisées et autorisées sur l'ensemble des machines du domaine Windows ;
- créer les règles pour les applications autorisées à s'exécuter ;
- créer les règles pour les scripts et les installeurs ;
- créer des règles pour les bibliothèques (optionnel) ;
- créer des règles pour les applications universelles ;
- tester les règles mises en place et affiner leur configuration si nécessaire.

3.1 Démarche préalable

3.1.1 Réaliser un inventaire des applications utilisées

Cette étape consiste à répertorier l'ensemble des applications nécessaires à la réalisation des missions d'une entité. Pour cela, il convient d'établir une liste des logiciels autorisés sur la base de critères bien précis (fonctionnalités, robustesse, mises à jour de sécurité régulières, etc.), qui permettra de définir une (ou plusieurs) configuration(s) de référence. Des logiciels de gestion de parc informatique peuvent être employés pour connaître les applications utilisées au sein d'un organisme.

AppLocker possède également une fonctionnalité d'audit qui permet de simuler l'application d'une politique de restrictions logicielles sans bloquer l'exécution des programmes. Lorsque cette fonctionnalité est activée, les règles ne sont pas appliquées mais simplement évaluées, et tous les évènements générés sont écrits dans le journal d'AppLocker. Ce dernier peut être consulté dans l'observateur d'évènements de Windows en parcourant l'arborescence de la manière suivante : *Journaux des applications et services* → *Microsoft* → *Windows* → *AppLocker*.

R4 - Inventorier les applications à l'aide de la fonctionnalité d'audit

> Lorsqu'il n'existe pas d'inventaire exhaustif des applications utilisées dans une organisation, la fonctionnalité d'audit d'AppLocker peut être utilisée pour identifier les applications inconnues.

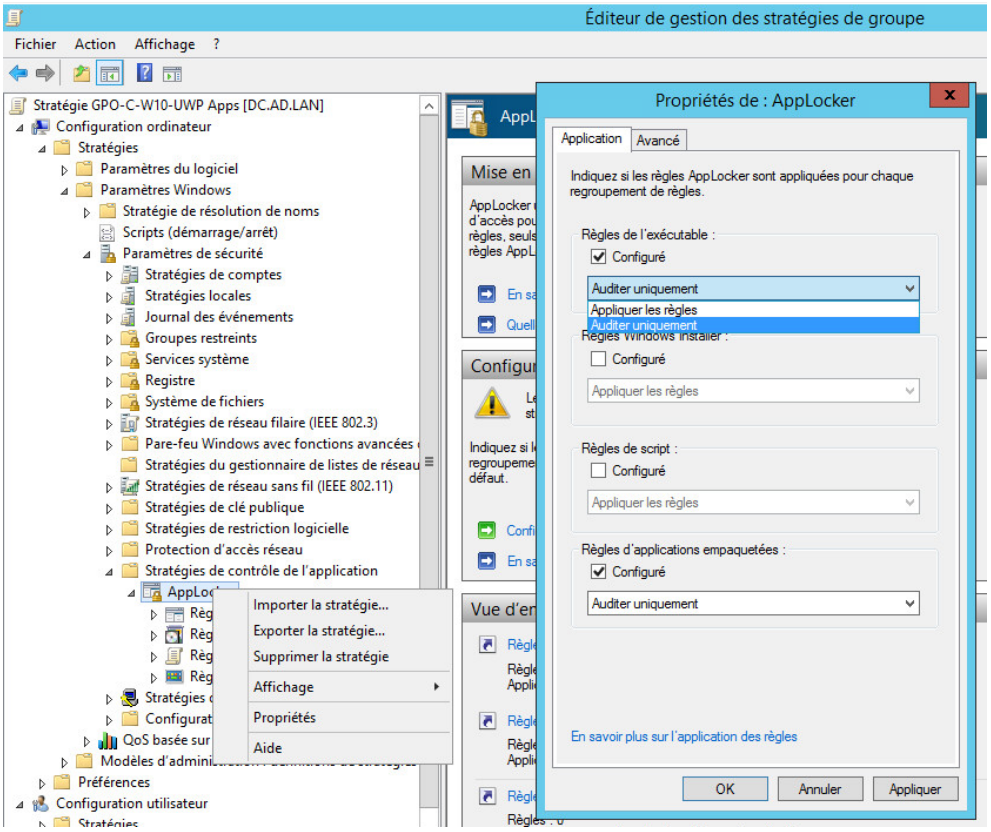

FIGURE 1 – Configuration d'AppLocker en mode audit

3.1.2 Activer et configurer AppLocker sur les systèmes

Une fois l'inventaire des applications réalisé, il reste à activer le mécanisme et à créer les règles correspondantes sur la configuration de référence. AppLocker ne fonctionne que si le service *Identité de l'application* est démarré.

R5 - Démarrer automatiquement le service `AppIdSvc` sur les équipements à protéger

Pour activer AppLocker sur un système, le service `AppIdSvc` - *Identité de l'application* - doit être configuré pour démarrer automatiquement au lancement du système.

Le démarrage automatique de ce service peut être configuré par GPO, comme le montre la figure 3.

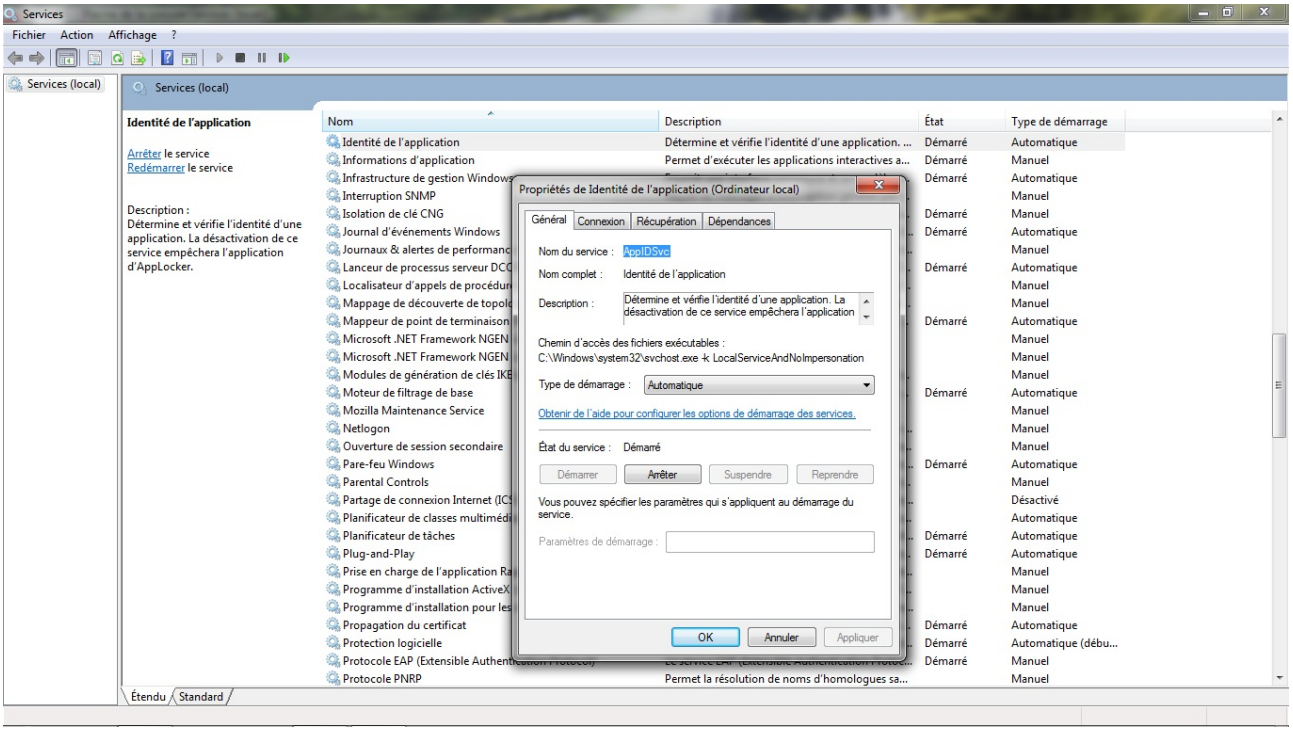

FIGURE 2 – Configuration du service *Identité de l'application* en démarrage automatique

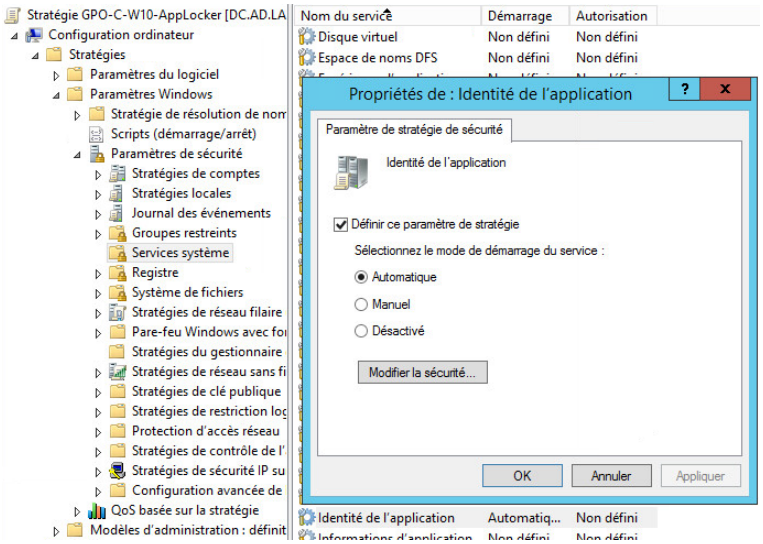

FIGURE 3 – Configuration du service *Identité de l'application* en démarrage automatique par GPO

R6 - Utiliser des comptes utilisateurs non privilégiés

Les systèmes sur lesquels AppLocker est déployé doivent être utilisés avec un compte utilisateur standard. Les utilisateurs ne doivent en aucun cas disposer de privilèges d'administration (locaux ou distants).

AppLocker peut être configuré au travers d'une stratégie locale, applicable à une machine donnée, ou bien d'une stratégie de groupe lorsque les machines sont rattachées à un domaine. Dans ce cas, il

est conseillé de tester préalablement la stratégie en mode « audit » sur un échantillon représentatif du parc informatique, puis de l'appliquer progressivement sur l'ensemble du domaine.

3.2 Configuration des règles

3.2.1 Créer les règles pour les exécutables

Cette étape consiste à créer et à tester les règles Applocker permettant aux utilisateurs d'exécuter uniquement les logiciels autorisés dans une organisation.

> ⚠ Si Applocker est activé et aucune règle n'est définie, l'exécution de tous les programmes sera bloquée, y compris les exécutables système. En pratique cela pourrait se traduire par l'impossibilité d'ouvrir une session sur la machine.

La fonction *Créer des règles par défaut* crée automatiquement trois règles de type *Chemin d'accès* qui sont :

- une règle autorisant tous les utilisateurs à exécuter les programmes situés dans `C:\Windows` et ses sous-répertoires ;
- une règle autorisant tous les utilisateurs à exécuter les programmes situés dans `C:\Program Files`, ses équivalents `C:\Program Files (x86)` et `C:\Programmes` et leurs sous-répertoires ;
- une règle autorisant les administrateurs à exécuter les fichiers depuis tous les emplacements.

Ces trois règles permettent d'activer AppLocker avec une configuration qui, dans la plupart des cas, sera immédiatement opérationnelle. Cette configuration par défaut requiert toutefois quelques modifications, que la suite de ce document s'attache à décrire, afin d'élever le niveau de sécurité.

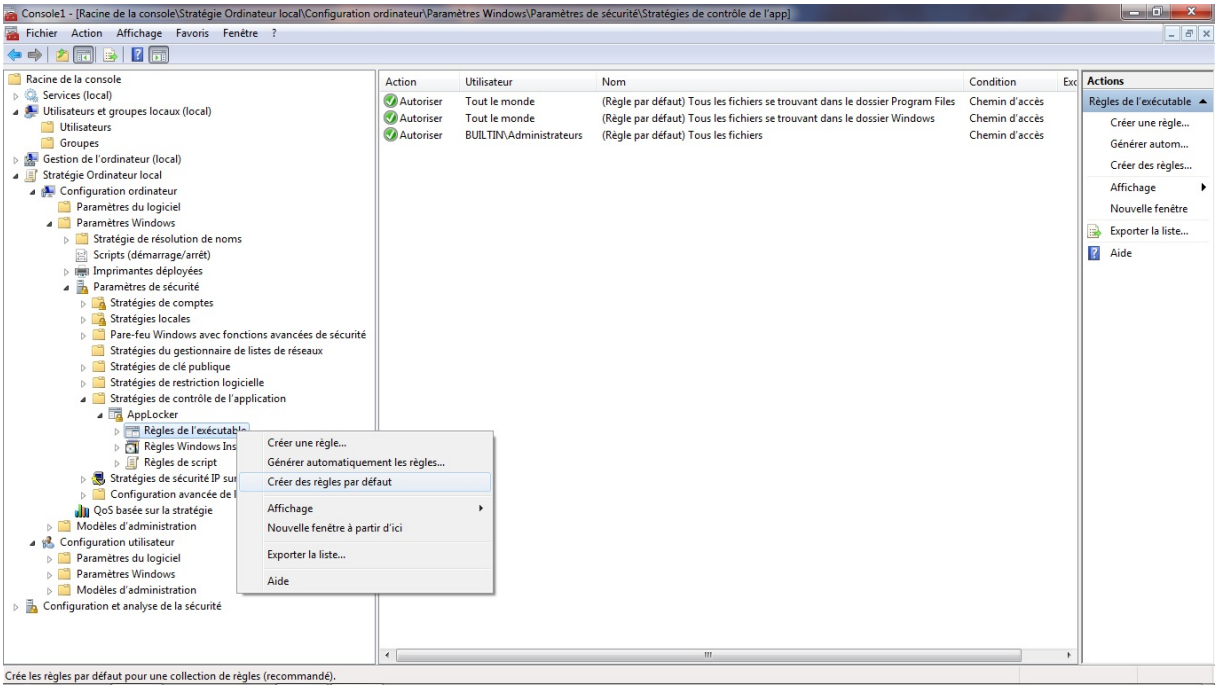

FIGURE 4 – Règles créées par défaut pour les exécutables

Il est à noter que la règle créée par défaut pour `C:\Windows` autorise de fait

l'exécution de programmes depuis des emplacements sur lesquels un utilisateur standard dispose des droits d'écriture, par exemple `C:\Windows\temp`, `C:\Windows\tasks` ou encore `C:\Windows\system32\spool\drivers\color`. Ainsi, tout utilisateur standard peut contourner cette règle de façon triviale, en créant ou en copiant un programme dans ces emplacements. Pour remédier à ce problème, il est recommandé de lister l'ensemble des répertoires accessibles en écriture aux utilisateurs standard et d'ajouter des exceptions de type *Chemin d'accès* à la règle par défaut. Un logiciel comme *AccessEnum* pourra être avantageusement utilisé pour identifier rapidement les répertoires en question.

R7 - Prêter une attention particulière aux dossiers accessibles en écriture

Lorsqu'une règle de type *Chemin d'accès* est utilisée, le dossier spécifié ainsi que ses sous-dossiers ne doivent être accessibles en écriture qu'aux administrateurs et aux entités `SYSTEM`.

Il suffit d'un seul dossier autorisé par AppLocker et accessible en écriture aux comptes utilisateurs non privilégiés pour rendre inefficace la stratégie AppLocker.

R8 - Compléter la règle par défaut créée pour `C:\Windows`

La règle par défaut créée pour `C:\Windows` doit être complétée par l'ajout d'exceptions pour les sous-répertoires accessibles en écriture par les utilisateurs.

Une méthode alternative conseillée consiste à remplacer la règle par défaut par une règle de type *Éditeur* n'autorisant que les composants signés par Microsoft.

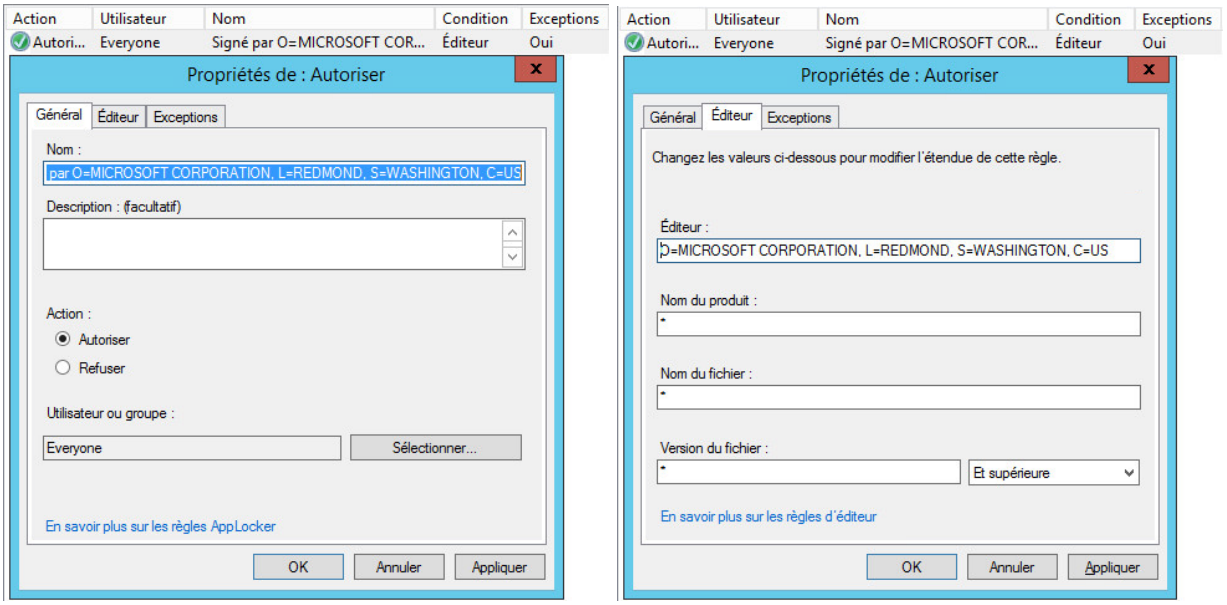

FIGURE 5 – Règle d'autorisation des exécutables de l'éditeur Microsoft

De façon analogue, la règle par défaut créée pour `C:\Program Files` (ou répertoires équivalents) autorise tous les programmes de ce dossier à s'exécuter. Il convient donc de s'assurer que toutes les applications présentes à cet emplacement sont autorisées. Il convient aussi de s'assurer qu'aucune application n'y a créé de répertoire accessible aux utilisateurs en écriture.

Néanmoins, sur un parc informatique existant, il est peu probable que toutes les systèmes soient homogènes. Aussi, pour une meilleure maîtrise des applications autorisées à s'exécuter et une meilleure lisibilité des règles, il est préférable de supprimer la règle par défaut au profit de règles autorisant explicitement les applications stockées à cet emplacement.

R9 - Remplacer la règle créée par défaut pour `C:\Program Files`

Il est recommandé de supprimer la règle par défaut concernant `C:\Program Files` et de la remplacer par des règles autorisant explicitement chaque application de cet emplacement à s'exécuter.

Une méthode alternative consiste à supprimer la règle par défaut au profit de règles basées sur les signatures numériques des applications ou, lorsque possible, sur les empreintes.

Un assistant permet de générer automatiquement les règles des exécutables en scannant les dossiers dans lesquels ils se trouvent. L'assistant permet de choisir entre deux algorithmes :

- la création de règles s'appuyant sur des signatures numériques le cas échéant, dans le cas contraire l'assistant propose au choix de créer une règle reposant sur l'empreinte ou sur le chemin d'accès ;
- la création de règles reposant sur les empreintes des fichiers.

Bien que cet assistant facilite la tâche de création des règles, il convient de s'assurer que les règles ainsi générées ne sont pas trop permissives.

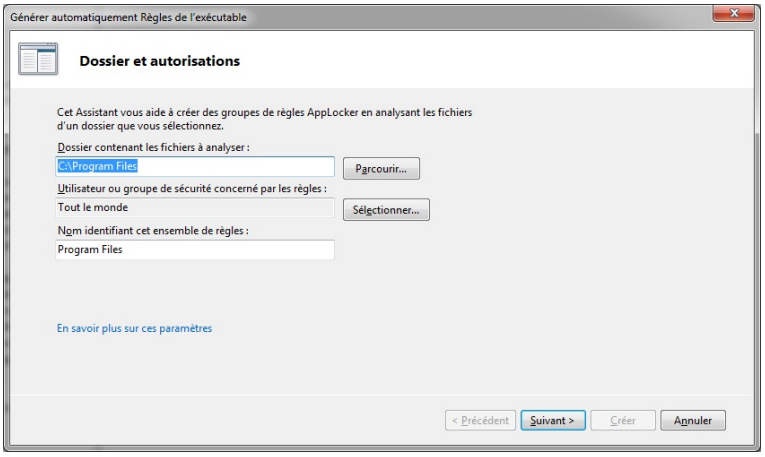

FIGURE 6 – Démarrage de l'assistant de génération automatique de règles

3.2.2 Créer les règles pour les scripts et les installeurs

Les règles pour les scripts et les installeurs doivent également être configurées. S'agissant des installeurs, les règles créées par défaut n'autorisent que les fichiers signés ou se trouvant dans `C:\Windows\installer`, non accessible en écriture aux utilisateurs. Sur un réseau bien administré,

ces règles ne posent pas de problème particulier. En ce qui concerne les scripts, les règles par défaut n'autorisent leur exécution que depuis `C:\Windows`, ce qui peut être contourné, comme présenté à la section 3.2.1.

R10 - Refuser l'exécution de scripts depuis `C:\Windows`

Supprimer les règles autorisant l'exécution des scripts depuis `C:\Windows`. N'autoriser que des scripts signés numériquement ou désignés explicitement par leur empreinte.

3.2.3 Créer les règles pour les bibliothèques (optionnel)

Par défaut, les règles concernant les bibliothèques ne sont pas appliquées. Un message d'avertissement met en garde l'administrateur concernant un possible impact sur les performances ainsi que des comportements inattendus si les règles ne sont pas correctement définies (voir figure 7).

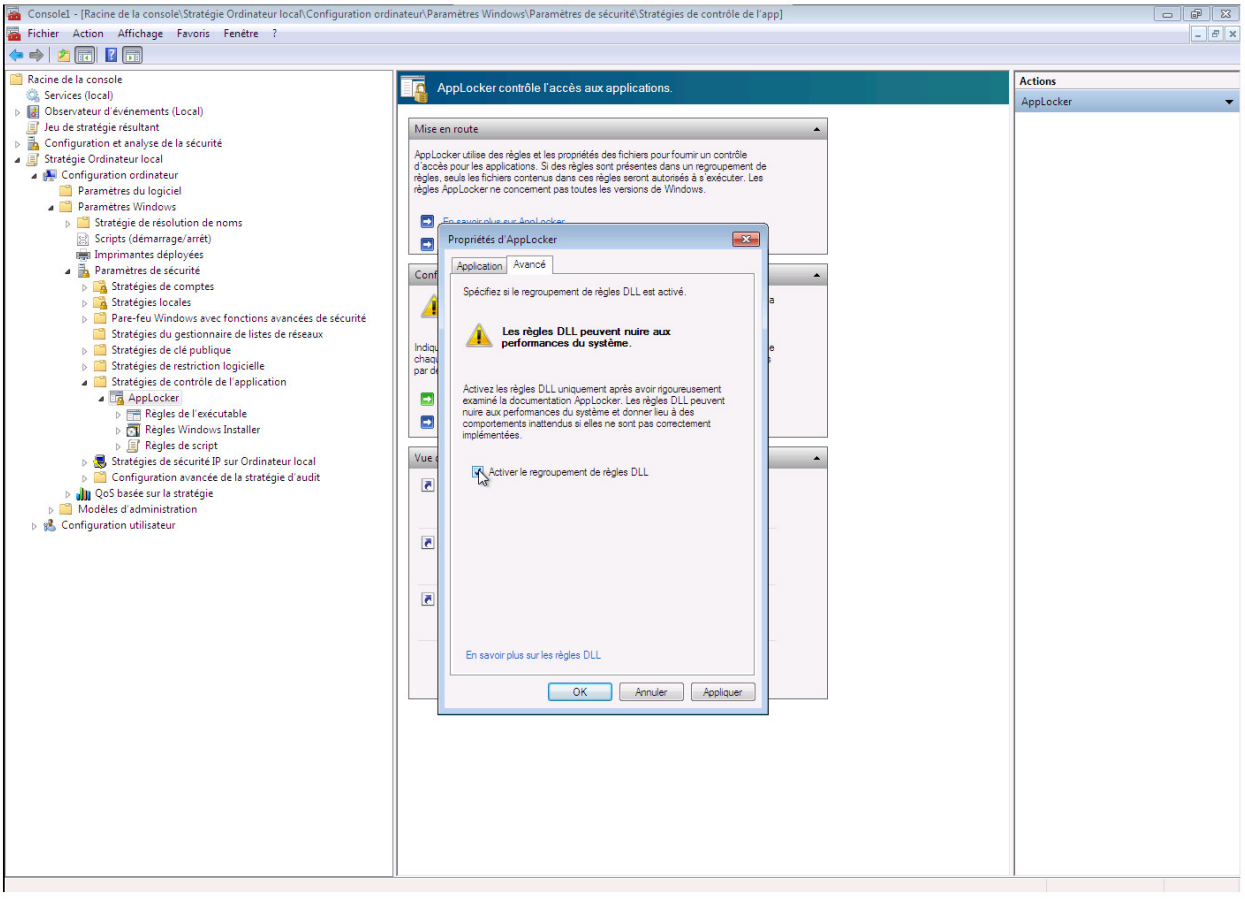

FIGURE 7 – Activation des règles DLL

En fonction des configurations logicielles, l'activation des règles DLL peut effectivement nécessiter l'ajout plus ou moins conséquent de règles spécifiques afin de garantir le bon fonctionnement des applications. Quoi qu'il en soit, la possibilité d'activer des règles DLL doit toujours être prise en considération. Plusieurs méthodes de contournement connues d'AppLocker s'appuient sur le chargement d'une bibliothèque contenant un code malveillant (via le binaire `rundll32.exe` par

exemple).

R11 - Activer les règles concernant les DLL

Dans la mesure du possible, activer les règles concernant les DLL en cochant la case dans l'onglet avancé de la fenêtre de configuration de mise en application des règles.

R12 - Identifier les dossiers accessibles en écriture (règles applicables aux DLL)

À l'instar des règles relatives aux exécutables, les règles par défaut doivent être complétées par l'ajout d'exceptions portant sur les sous-répertoires des chemins d'accès qui restent accessibles en écriture aux utilisateurs.

3.2.4 Créer les règles pour les applications universelles

Cette étape consiste à créer les règles permettant aux utilisateurs d'exécuter les applications universelles (également appelées *Packaged Apps*, applications empaquetées, *Windows Apps*, *Metro style apps*, applications immersives ou bien encore *Modern Apps*) autorisées dans une organisation. Il est en premier lieu important de distinguer les applications universelles sous windows 10, les applications universelles sous windows 8 et les applications de bureau classiques.

Les applications classiques de bureau sont les applications historiques, généralement installées et composées d'un fichier exécutable (avec une extension `.exe`) ainsi que d'un ensemble d'autres fichiers de ressources. Il s'agit des applications habituellement utilisées depuis les toutes premières versions de Windows. L'exécution de ces applications est couverte par les règles créées dans la section 3.2.1.

De manière synthétique, les applications universelles Windows 10 sont développées pour la plateforme *Universal Windows Platform (UWP)* tandis que les applications universelles Windows 8 utilisent spécifiquement l'interface de programmation WinRT (*Windows Runtime*). Ces deux types d'applications portent le même nom mais sont pourtant bien différentes. Une application universelle pour Windows 8 ne s'exécutera pas sur Windows 10 et inversement. En revanche, la plateforme UWP est fortement inspirée et reprend une partie de son prédécesseur, ce qui facilite le portage des applications universelles Windows 8 vers Windows 10.

Les applications universelles, empaquetées en un seul et unique fichier au format `.AppX`, sont généralement moins complexes que les applications de bureau classiques. De par leur cadre d'exécution contrôlé, elles présentent bien moins de risques de sécurité pour le système que les applications de bureau classiques. Les applications universelles peuvent d'ailleurs être installées avec un simple compte utilisateur non privilégié et sont publiables et téléchargeables via un magasin d'applications (*Windows Store* ou magasin privé déployé en interne). Enfin, elles sont utilisables sur une large gamme d'équipements (tablettes, ordiphones, ordinateurs, etc.). L'exécution des applications universelles n'est pas couverte par les règles créées à la section 3.2.1.

R13 - Créer des règles AppLocker pour les applications universelles

Dès lors qu'une stratégie AppLocker a été mise en œuvre efficacement, il peut être pertinent de la compléter par un contrôle des applications universelles. Ce contrôle passe entre autres par la mise en œuvre de règles de restrictions logicielles spécifiques aux applications universelles.

Si la restriction des applications universelles est activée mais qu'aucune règle n'est définie, l'exécution de toutes les applications universelles sera bloquée. AppLocker ne supporte que des règles éditeur pour les applications universelles, et ces dernières sont obligatoirement signées.

La fonction *Créer des règles par défaut* crée automatiquement une règle de type *Éditeur* permettant d'autoriser toutes les applications universelles.

R14 - Autoriser explicitement l'exécution de chaque application universelle

Il est recommandé d'autoriser explicitement l'exécution de chaque application universelle au sein d'une organisation.

Notons toutefois que certaines fonctionnalités du système sont fournies par des applications universelles de Microsoft, parmi lesquelles :

- le panneau de configuration des paramètres du système *Windows.ImmersiveControlPanel* ;
- le menu démarrer (entre autres) *Microsoft.Windows.ShellExperienceHost* ;
- la gestion des vignettes *Microsoft.Windows.SecondaryTileExperience* ;
- la boîte de dialogue relative aux tâches d'impression : *Windows.PrintDialog* ;
- la gestion du passage du mode clavier et souris au mode tablette tactile et inversement pour le matériel hybride type 2 en 1 *Windows.DevicesFlow* ;
- le composant de recherche *Microsoft.Windows.Cortana* ;
- la calculatrice *Microsoft.WindowsCalculator* ;
- l'horloge et les alarmes *Microsoft.WindowsAlarms* ;
- la visionneuse d'images *Microsoft.Windows.Photos*.

Concernant les applications universelles de Microsoft, il est conseillé de les autoriser par deux règles de type *Éditeur* (une pour `CN=Microsoft Corporation` et une pour `CN=Microsoft Operating System`) puis d'ajouter des exclusions afin de bloquer spécifiquement les applications universelles de Microsoft qui ne sont pas souhaitées. De cette manière, si de nouvelles fonctionnalités du système sont fournies par des applications universelles suite à des mises à jour de Windows, elles ne seront pas bloquées.

Les règles AppLocker recommandées pour les applications universelles de Microsoft sont illustrées dans l'annexe C. Il appartient à chaque organisation d'inventorier les applications à autoriser en fonction de ses besoins et des risques de sécurité identifiés, puis de créer les règles AppLocker correspondantes. Des commandes PowerShell facilitent la récupération des signatures d'applications universelles et leur inventaire sur un poste de travail. Ces commandes sont également présentées dans l'annexe C.

3.3 Tester les règles mises en place et affiner leur configuration si nécessaire

Lorsque les règles ont été créées et testées avec succès sur la configuration de référence, elles doivent être testées en conditions réelles. Dans un premier temps, afin d'éviter de bloquer les utilisateurs, la configuration en mode *audit* pourra s'avérer utile. Pour faciliter l'analyse de l'ensemble des évènements générés sur les machines, il est possible de les centraliser grâce à la fonction de collecte des évènements disponible depuis Windows Server 2003 R2, Windows Vista SP1 et Windows Server 2008.

La collecte des évènements peut se faire en créant un abonnement sur la machine prévue à cet

effet, ce qui permet en outre de sélectionner seulement les évènements intéressants grâce à un filtre de requête. Une liste non exhaustive des évènements générés par AppLocker est fournie dans l'annexe B.

Il est également possible d'obtenir des statistiques sur les fichiers qui sont (ou seraient) bloqués par une stratégie AppLocker grâce à la *cmdlet* Powershell `GetAppLockerFileInformation`. Par exemple, la commande suivante permet de connaître le nombre de fois où un fichier aurait été bloqué si les règles avaient été appliquées.

```
Get-AppLockerFileInformation -EventLog -Logname C:\...\fichier-log -EventType Audited
  -Statistics
```

Dans un deuxième temps, AppLocker pourra être déployé sur une partie du parc informatique, de préférence sur un échantillon représentatif. Si les tests sont concluants il pourra alors être étendu à l'ensemble du domaine.

3.4 Maintenir les règles à jour au gré des évolutions des systèmes

L'installation de nouveaux logiciels, de mises à jour ou correctifs sur le parc informatique doit toujours être précédée d'une phase de qualification sur des systèmes de référence. En effet ces actions pourraient être à l'origine de blocages ou de comportements anormaux pour les utilisateurs, nécessitant la modification ou l'ajout de règles.

Le choix des règles doit être adapté aux spécificités des logiciels. Il s'agit de trouver le meilleur compromis entre niveau de sécurité et facilité de maintenance. Même lorsque les exécutables sont signés, il n'est pas toujours possible d'écrire des règles adaptatives (il arrive en effet que le numéro de version du logiciel soit inclus dans le champ *Nom du produit*, ce qui empêche d'écrire une règle pérenne).

4 Considérations de sécurité fondamentales

Lors de la configuration d'AppLocker, il faut être concient de certaines limites et possibilités de contournement.

L'exécutable `regsvr32.exe` peut être détourné pour exécuter du code arbitraire local ou distant par l'intermédiaire d'astuces bien connues des attaquants et largement décrites sur Internet. Si cet exécutable est autorisé par AppLocker, il est ainsi possible de contourner trivialement une stratégie AppLocker et d'exécuter du code malveillant (établissement d'un canal de contrôle à distance, opérations de rançonnage, exfiltration de données, tentatives d'élévations de privilèges, etc.). En sus de la règle AppLocker autorisant les exécutables signés par Microsoft, il est donc primordial de bloquer explicitement `regsvr32.exe` dans la mesure où un utilisateur non privilégié n'a pas de raison de l'utiliser. En complément, il serait également pertinent de bloquer les flux sortants crées à l'initiative de cet exécutable au niveau du pare-feu Windows.

R15 - Créer une exception pour l'exécutable `regsvr32.exe`

> Pour éviter le contournement trivial d'AppLocker, il est recommandé de créer une exception de type *Éditeur* pour interdire l'exécution du programme `regsvr32.exe` par des utilisateurs non privilégiés.

L'exécutable `InstallUtil.exe` du *Microsoft Framework 4.0* peut également être détourné pour contourner des stratégies AppLocker. Dans la mesure où l'utilisation de cet exécutable n'est pas nécessaire par un utilisateur non privilégié, il est recommandé de le bloquer.

R16 - Créer une exception pour l'exécutable `InstallUtil.exe`

Pour éviter le contournement trivial d'AppLocker, il est recommandé de créer une exception de type *Éditeur* pour interdire l'exécution du programme `InstallUtil.exe` du *Microsoft Framework 4.0* par un utilisateur non privilégié.

Ces deux exceptions (`regsvr32.exe` et `InstallUtil.exe`) doivent être ajoutées à la règle qui autorise les exécutables de l'éditeur Microsoft (figure 8), ou celle qui autorise l'emplacement `%WINDIR%` (figure 9).

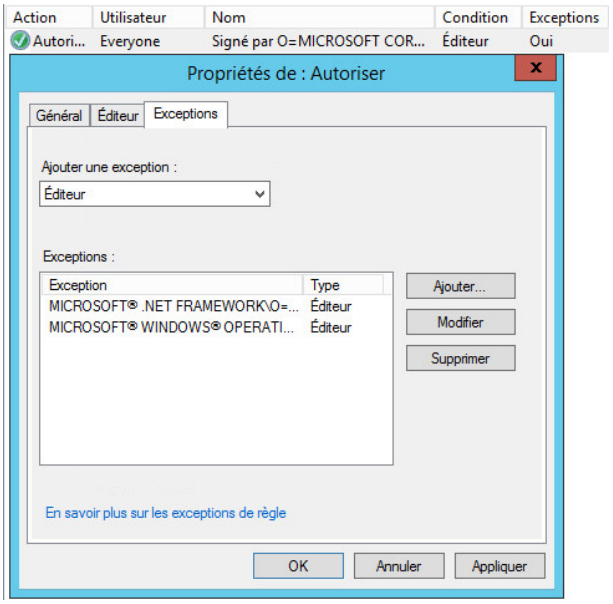

FIGURE 8 – Exceptions à la règle d'autorisation des exécutables Microsoft

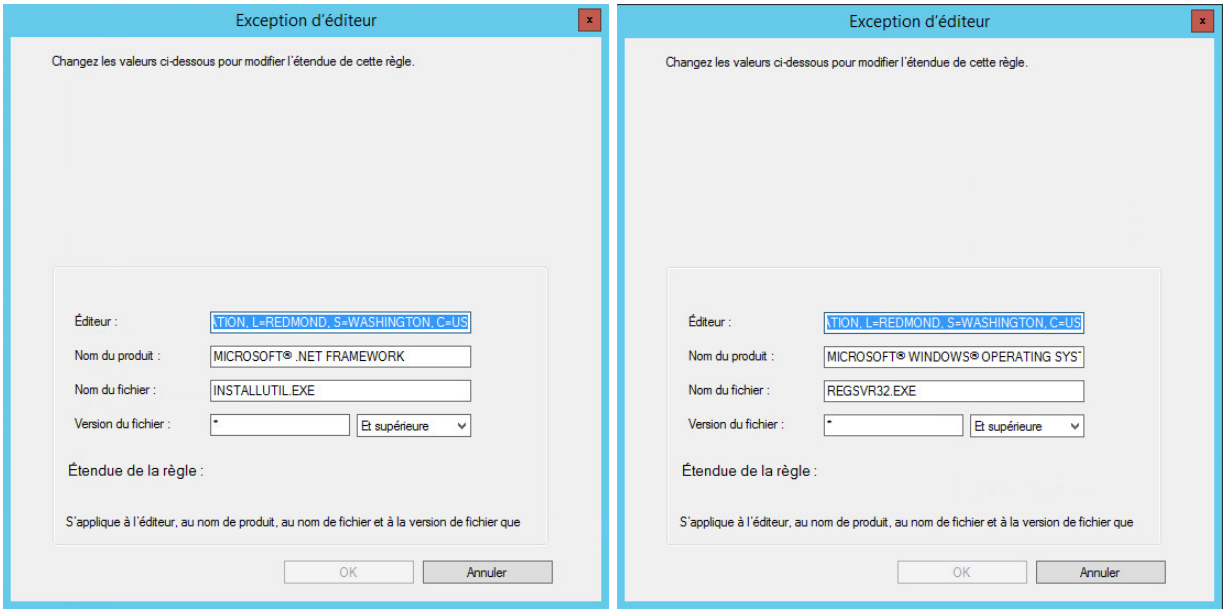

FIGURE 9 – Exceptions `InstallUtil.exe` et `regsvr32.exe` à la règle d'autorisation des exécutables Microsoft

Les composants ActiveX peuvent permettre d'instancier certains programmes dont l'exécution devrait être bloquée par AppLocker. Il est par exemple possible de lancer une session RDP en insérant le contrôle Microsoft RDP client dans un document Word. Seules des règles sur les bibliothèques permettent alors de bloquer ces moyens de contournement.

R17 - Bloquer les bibliothèques qui permettraient d'instancier des programmes indésirables

> Pour les programmes dont l'exécution est bloquée par AppLocker, si d'autres composants logiciels permettent de les instancier, ils doivent aussi être bloqués par des règles sur les bibliothèques associées.

Les règles AppLocker autorisent ou empêchent le lancement des programmes, mais n'exercent aucun contrôle sur le comportement des programmes une fois lancés. En utilisant des fonctions avec des paramètres spéciaux [3], un programme peut lancer des exécutables ou des bibliothèques sans que les règles ne s'appliquent. En particulier, les modules d'extension (greffons, plugins, etc.) de certains logiciels peuvent être exploités par un attaquant pour contourner les restrictions d'Applocker.

R18 - Évaluer la sécurité des programmes autorisés

> Les programmes autorisés à s'exécuter doivent avoir fait l'objet de vérifications pour s'assurer qu'ils ne comportent pas de fonctionnalités susceptibles de permettre un contournement d'Applocker. Les utilisateurs ne doivent pas être autorisés à installer des modules d'extension.

AppLocker ne contrôle pas l'exécution de tous les codes interprétés existants, comme par exemple les scripts Perl ou les macros. Des mesures de sécurité supplémentaires doivent être prises

3. Lorsque la fonction `LoadLibraryEx()` est appelée avec le drapeau `LOAD_IGNORE_CODE_AUTHZ_LEVEL`, les règles AppLocker sont ignorées (voir la page *Security Considerations for AppLocker* sur le site `technet.microsoft.com`).

afin de restreindre les possibilités en la matière.

R19 - Restreindre les possibilités d'exécution des codes interprétés par les programmes autorisés

Les logiciels offrant des possibilités de *scripting* doivent être paramétrés de façon à restreindre les possibilités d'exécution des codes interprétés. Dans le cas d'une suite bureautique, il est recommandé de restreindre l'exécution aux seules macros signées et de confiance, voire d'empêcher l'utilisation de l'éditeur de macros pour les utilisateurs qui n'en ont pas l'utilité.

Les applications 16 bits ne sont pas contrôlées par AppLocker car celles-ci sont exécutées dans une machine virtuelle instanciée par le processus *NTVDM - NT Virtual DOS Machine* - (`C:\Windows\system32\NTVDM.exe`).

R20 - Désactiver la fonction NTVDM

Si aucune application 16 bits n'est utilisée, désactiver la fonction *NTVDM* avec l'éditeur de stratégie de groupe [4].

4. La fonction peut être désactivée en se rendant dans l'arborescence suivante : *Configuration ordinateur → Modèles d'administration → Composants Windows → Compatibilité des applications 16 bits → Empêcher l'accès aux applications 16 bits.*

Annexes

A Variables de chemin d'accès utilisées par AppLocker

Le tableau ci-dessous indique les variables utilisées par AppLocker pour désigner les principaux chemins d'accès de Windows. La correspondance avec les variables d'environnement couramment utilisées est également indiquée.

Chemin d'accès	Variable d'environnement Windows	Variable AppLocker
Windows	`%SystemRoot%`	`%WINDIR%`
System32 et SysWOW64	-	`%SYSTEM32%`
Lecteur d'installation de Windows	`%SystemDrive%`	`%OSDRIVE%`
Répertoire des programmes	`%ProgramFiles%` `%ProgramFiles(x86)%`	`%PROGRAMFILES%`
Media amovibles	-	`%REMOVABLE%`
Périphériques de stockage amovibles	-	`%HOT%`

B Liste non exhaustive des évènements générés par AppLocker

Le tableau ci-dessous recense les identifiants et descriptifs des principaux évènements générés par Applocker.

Id	Type	Description	Catégorie
8001	Information	La stratégie AppLocker a été correctement appliquée	-
8002	Information	<Fichier> a été autorisé à s'exécuter.	`EXE` ou `DLL`
8003	Avertissement	<Fichier> a été autorisé à s'exécuter mais aurait été bloqué si les règles étaient appliquées.	
8004	Erreur	<Fichier> n'a pas été autorisé à s'exécuter.	
8005	Information	<Fichier> a été autorisé à s'éxécuter.	Script ou Installeur
8006	Avertissement	<Fichier> a été autorisé à s'exécuter mais aurait été bloqué si les règles étaient appliquées.	
8007	Erreur	<Fichier> n'a pas été autorisé à s'exécuter.	
8020	Information	L'exécution de <nom> a été autorisée.	Applications universelles
8021	Avertissement	L'exécution de <nom> a été autorisée mais aurait été empêchée si les règles étaient appliquées.	
8022	Erreur	L'exécution de <nom> a été empêchée.	
8023	Information	L'installation de <nom> a été autorisée.	Installation d'applications universelles
8024	Avertissement	L'installation de <nom> a été autorisée mais aurait été empêchée si les règles étaient appliquées.	
8025	Erreur	L'installation de <nom> a été empêchée.	
8027	Avertissement	Aucune règle d'applications empaquetées configurée	Applications universelles

C Création de règles AppLocker d'applications universelles

C.1 Règles AppLocker pour les applications universelles de Microsoft

Les applications universelles de Microsoft pré-installées peuvent être signées par deux éditeurs distincts, différenciés par leur *Common Name* (`CN`) :

- `CN=Microsoft Corporation...` ;
- `CN=Microsoft Windows....`

Il est donc nécessaire de créer deux règles AppLocker pour autoriser ces deux éditeurs, comme illustré en figure 10.

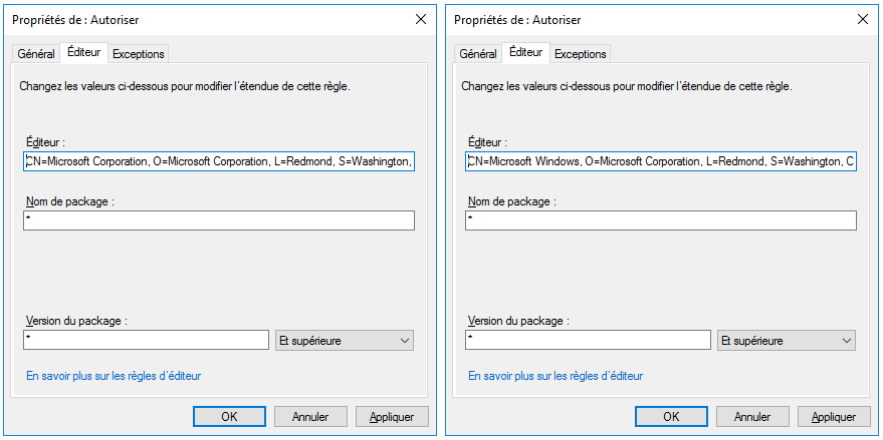

FIGURE 10 – Règles AppLocker d'éditeur pour applications universelles Microsoft

Pour bloquer les applications universelles de Microsoft non souhaitées, des exceptions à ces règles doivent être ajoutées, comme illustré en figure 11.

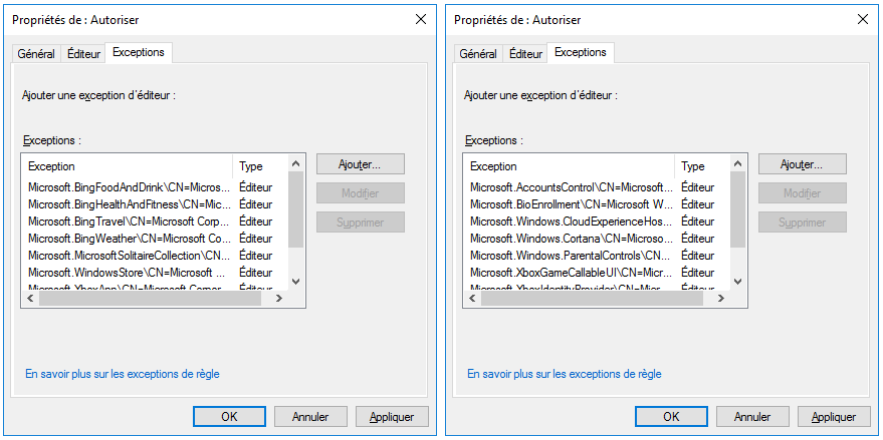

FIGURE 11 – Exceptions aux règles AppLocker d'éditeur pour applications universelles Microsoft

Ces exceptions devraient porter sur l'éditeur et le nom d'application (nom du *package*) quelle que soit sa version, comme illustré en figure 12.

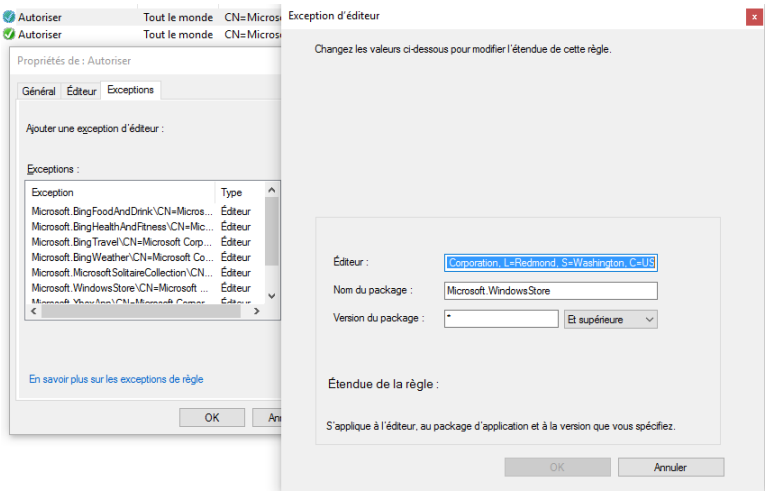

FIGURE 12 – Détail d'exception aux règles AppLocker d'éditeur pour applications universelles Microsoft

C.2 Commandes PowerShell utiles

La commande Powershell suivante affiche le nom de toutes les applications universelles présentes sur l'ordinateur où elle est exécutée.

```
Get-AppxPackage -AllUsers | Select Name
```

La commande Powershell suivante affiche les informations de l'application *ExempleApp*

```
Get-AppxPackage -Name ExempleApp
```

Parmi les informations de l'application, trois sont utiles pour la création des règles AppLocker :
– l'éditeur (*Publisher*) présenté sous forme de *Distinguished Name* (`CN=...`) ;
– le nom de l'application (*Name*) ;
– la version.

Ces informations se retrouvent au niveau de l'interface de création d'une règle pour application universelle, comme indiqué figure 13.

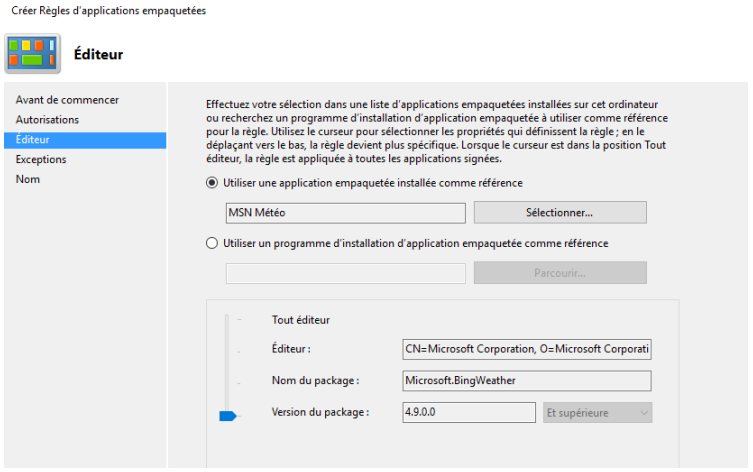

FIGURE 13 – Règle AppLocker pour application universelle