Déploiement

Cette catégorie concerne toute la partie déploiement.Liens utiles :
Documentation WAPT
Générateur Schneegans

- Modification de la configuration (console)
- ISO
- Configuration

Modification de la configuration (console)

Ajout de la fonctionnalité de déploiement

Pour ajouter la fonctionnalité de déploiement de postes il faudra ajouter une image WinPE, cette capture provient de la console qui nous indique d'ajouter une image.

Prérequis

Paquets

Il faudra aller d'abord chercher les paquets `tis-wads-requirement-minimal-x64` et `tis-wads-requirement-minimal` trouvable dans le dépôt de TranquilIT. Voici une procédure pour l'ajout de paquet



NOTE : Il faut que le paquet soit installé soit sur un poste, soit sur la console dédiée. Au moins une fois.

Pilotes

Une fois démarrer sur le réseau l'ordinateur est lancé sur l'image WinPE créer. A cet image s'ajoute une surcouche WADS qui viendra configurer le système selon le fichier unattend.xml avec les pilotes, afin d'installer le système.

Pour exporter les pilotes nous avons plusieurs choix:

1. Possibilité d'utiliser des pilotes génériques dans ce cas, il est possible d'utiliser les pilotes de machine virtuelles (oui oui de VM). Puis de faire les maj, après dans les paramètres ou

via Dell Command Update

2. (plus fiable) Possibilité de démarrer un pc fraichement arriver, le configurer, faire les maj de pilotes et de les exporter via la commande :

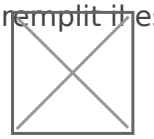
```
Export-WindowsDriver -Online -Destination FICHIER_DE_DESTINATION
```

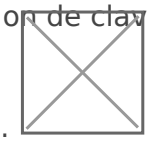
3. Voir avec le fabricant, pour qu'il fournisse les fichiers en .inf répartie dans des dossiers distinct. Exemple pour la carte réseau d'une VM :

```
netkvm.inf
|-> netkvm.cat
|-> netkvm.inf
|-> netkvm.pnf
|-> netkvm.sys
|-> netkvmp.exe
```

NOTE : Pour la création de l'image WinPE il sera nécessaire la première fois d'avoir un pack de pilotes.

Déploiement

Une fois les prérequis remplit il est alors possible de créer notre image WinPE qui servira pour le démarrage en réseau. 

- Une fois cliqué sur "oui" on arrive sur cet assistant de conception. On pourra cliquer sur la configuration de clavier qui nous interesse et choisir le dossier de pilote afin de confirmer la création. 

ISO

Récupération de l'Image Windows

Prérequis :

- Avoir obtenu l'image ISO de Windows.
- Être connecté à la Console WAPT

Intégration de l'ISO dans la Console WAPT

L'image ISO doit être mise à disposition du service WADS (WAPT Deployment Services) pour être exploitée lors du déploiement.

1. **Accéder à l'interface de Déploiement d'OS :**

- Sur la **Console WAPT**, naviguez vers l'onglet **Déploiement d'OS** (Déploiement de postes).

2. **Importer l'ISO d'installation :**

- Dans la section **Déploiement d'OS**, cherchez l'option ou l'icône **Importer des machines depuis un CSV** ou **Ajouter une ISO d'installation**.
- Suivez l'assistant :
- Pointez vers le chemin local où vous avez stocké votre fichier ISO Windows.
- La Console WAPT analyse l'image pour détecter les versions de Windows (WIM files) qu'elle contient.

3. **Sélectionner l'Architecture :**

- Choisissez l'architecture correspondante (ex. `x64`).

4. **Enregistrement de l'ISO :**

- La Console WAPT enregistre l'ISO, la rendant disponible pour être liée à un profil de déploiement.

Création de la Clé WinPE (Windows Preinstallation Environment)

Le WinPE est l'environnement minimal que le client télécharge via TFTP/HTTP pour exécuter l'Agent WAPT et lancer le processus d'installation.

1. **Naviguer vers l'outil WinPE :**

- Dans la section **Déploiement d'OS**, utilisez le bouton **Créer une clé WinPE**.

2. **Sélectionner les options :**

- L'outil va vous demander de sélectionner le modèle WinPE de base.
- Assurez-vous que l'option pour inclure l'Agent WAPT et les outils nécessaires est cochée.

3. **Génération et Enregistrement :**

- Lancez la génération de la clé WinPE. Le fichier généré sera stocké sur le Serveur WAPT (`SRVWAPT`) et utilisé par le service TFTP lors du démarrage PXE.

Prochaines Étapes Logiques

Une fois l'ISO et le WinPE intégrés, les étapes finales consisteront à :

1. **Exporter les pilotes** nécessaires pour vos postes clients et les intégrer dans des paquets WAPT (via l'onglet Pilotes).
2. **Créer une Configuration** (un profil de déploiement) dans l'onglet **Déploiement d'OS** pour lier l'ISO, les pilotes et les actions à effectuer (formatage, jointure de domaine, etc.).

Configuration

Cette partie est dédiée à documenter une configuration Windows 11 24H2+ type, fonctionnel.
NOTE : Il n'est pas obligatoire d'installer systématiquement l'agent WAPT ni même de joindre un domaine, pratique dans le cas de la vente de l'ordinateur ou de configuration pour des franchisés.

Description

Le script suivant :

- Configure la langue d'installation en **français**
- Installe la version **Pro** de Windows (index 6)
- Cible et formate la première partition disponible
- Injecte les pilotes
- Définis le nom de la machine comme le nom d'enregistrement auprès de la console
- Désactive la télémétrie
- Passe les écrans OOBE
- Joins le domaine de manière autonome

Script

```
<?xml version="1.0" encoding="utf-8"?>
<unattend xmlns="urn:schemas-microsoft-com:unattend">
<settings pass="windowsPE">
<component name="Microsoft-Windows-International-Core-WinPE"
processorArchitecture="{{architecture}}" publicKeyToken="31bf3856ad364e35" language="neutral"
versionScope="nonSxS" xmlns:wcm="http://schemas.microsoft.com/WMIConfig/2002/State"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
<SetupUILanguage>
<UILanguage>fr-FR</UILanguage>
</SetupUILanguage>
<InputLocale>040c:0000040c</InputLocale>
<SystemLocale>fr-FR</SystemLocale>
<UILanguage>fr-FR</UILanguage>
<UILanguageFallback>fr-FR</UILanguageFallback>
```

```
□□□<UserLocale>fr-FR</UserLocale>
□□</component>
□□<component name="Microsoft-Windows-Setup" processorArchitecture="{{architecture}}"
publicKeyToken="31bf3856ad364e35" language="neutral" versionScope="nonSxS"
xmlns:wcm="http://schemas.microsoft.com/WMIConfig/2002/State"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
□□□<ImageInstall>
□□□□<OSImage>
□□□□□<InstallToAvailablePartition>true</InstallToAvailablePartition>
□□□□□<WillShowUI>OnError</WillShowUI>
□□□□□<InstallFrom>
□□□□□□<MetaData>
□□□□□□□<Key>/IMAGE/INDEX</Key>
□□□□□□□<Value>6</Value>
□□□□□□</MetaData>
□□□□□</InstallFrom>
□□□□</OSImage>
□□□</ImageInstall>
□□□<UserData>
□□□□<ProductKey>
□□□□□<Key/>
□□□□</ProductKey>
□□□□<AcceptEula>true</AcceptEula>
□□□</UserData>
                        <UseConfigurationSet>false</UseConfigurationSet>
□□</component>
□</settings>
□<settings pass="offlineServicing">
□□<component name="Microsoft-Windows-PnpCustomizationsNonWinPE"
processorArchitecture="{{architecture}}" publicKeyToken="31bf3856ad364e35" language="neutral"
versionScope="nonSxS" xmlns:wcm="http://schemas.microsoft.com/WMIConfig/2002/State"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
□□□<DriverPaths>
□□□□<PathAndCredentials wcm:keyValue="1" wcm:action="add">
□□□□□<Path>c:\installdir\drivers</Path>
□□□□</PathAndCredentials>
□□□</DriverPaths>
□□</component>
□</settings>
```

```
<settings pass="generalize">
<component name="Microsoft-Windows-Security-SPP" processorArchitecture="{{architecture}}"
publicKeyToken="31bf3856ad364e35" language="neutral" versionScope="nonSxS"
xmlns:wcm="http://schemas.microsoft.com/WMIConfig/2002/State"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
<SkipRearm>1</SkipRearm>
</component>
</settings>
<settings pass="specialize">
<component name="Microsoft-Windows-UnattendedJoin" processorArchitecture="{{architecture}}"
publicKeyToken="31bf3856ad364e35" language="neutral" versionScope="nonSxS"
xmlns:wcm="http://schemas.microsoft.com/WMIConfig/2002/State"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
<Identification>
<Credentials>
<Domain>adwapt.local</Domain>
<Password>pass</Password>
<Username>wadsjoin@adwapt.local</Username>
</Credentials>
<JoinDomain>adwapt.local</JoinDomain>
                        </Identification>
</component>
<component name="Microsoft-Windows-International-Core"
processorArchitecture="{{architecture}}" publicKeyToken="31bf3856ad364e35" language="neutral"
versionScope="nonSxS" xmlns:wcm="http://schemas.microsoft.com/WMIConfig/2002/State"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
<InputLocale>040c:0000040c</InputLocale>
<SystemLocale>fr-FR</SystemLocale>
<UILanguage>fr-FR</UILanguage>
<UILanguageFallback>fr-FR</UILanguageFallback>
<UserLocale>fr-FR</UserLocale>
</component>
<component name="Microsoft-Windows-Security-SPP-UX" processorArchitecture="{{architecture}}"
publicKeyToken="31bf3856ad364e35" language="neutral" versionScope="nonSxS"
xmlns:wcm="http://schemas.microsoft.com/WMIConfig/2002/State"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
<SkipAutoActivation>true</SkipAutoActivation>
</component>
<component name="Microsoft-Windows-SQMApi" processorArchitecture="{{architecture}}"
```

```
publicKeyToken="31bf3856ad364e35" language="neutral" versionScope="nonSxS"
xmlns:wcm="http://schemas.microsoft.com/WMIConfig/2002/State"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
□□□<CEIPEnabled>0</CEIPEnabled>
□□</component>
□□<component name="Microsoft-Windows-Shell-Setup" processorArchitecture="{{architecture}}"
publicKeyToken="31bf3856ad364e35" language="neutral" versionScope="nonSxS"
xmlns:wcm="http://schemas.microsoft.com/WMIConfig/2002/State"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
□□□<ComputerName>{{hostname}}</ComputerName>
□□□<ProductKey>VK7JG-NPHTM-C97JM-9MPGT-3V66T</ProductKey>
□□</component>
        <component name="Microsoft-Windows-Deployment"
processorArchitecture="{{architecture}}" publicKeyToken="31bf3856ad364e35" language="neutral"
versionScope="nonSxS" xmlns:wcm="http://schemas.microsoft.com/WMIConfig/2002/State"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
            <RunSynchronous>
                <RunSynchronousCommand wcm:action="add">
                    <Path>"c:\installdir\wads.exe" --setup-startup-
script="c:\installdir\SetupComplete.cmd"</Path>
                    <Order>2</Order>
                    <Description>postinstall</Description>
                </RunSynchronousCommand>
            </RunSynchronous>
        </component>   □□
□</settings>
□<settings pass="oobeSystem">
□□<component name="Microsoft-Windows-Shell-Setup" processorArchitecture="{{architecture}}"
publicKeyToken="31bf3856ad364e35" language="neutral" versionScope="nonSxS"
xmlns:wcm="http://schemas.microsoft.com/WMIConfig/2002/State"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
□□□<OOBE>
□□□□<HideEULAPage>true</HideEULAPage>
□□□□<HideOEMRegistrationScreen>true</HideOEMRegistrationScreen>
□□□□<HideOnlineAccountScreens>true</HideOnlineAccountScreens>
□□□□<HideWirelessSetupInOOBE>true</HideWirelessSetupInOOBE>
□□□□<NetworkLocation>Work</NetworkLocation>
□□□□<SkipUserOOBE>true</SkipUserOOBE>
□□□□<SkipMachineOOBE>true</SkipMachineOOBE>
```

```
□□□□<ProtectYourPC>1</ProtectYourPC>
□□□</OOBE>
                        <AutoLogon>
                                    <Enabled>true</Enabled>
                                    <LogonCount>1</LogonCount>
                                    <Domain>adwapt.local</Domain>
                                    <Username>Administrateur@adwapt.local</Username>
                                    <Password>
                                              <value>pass</value>
                                              <PlainText>true</PlainText>
                                    </Password>
                        </AutoLogon>
□□□<RegisteredOrganization>CHCOM</RegisteredOrganization>
□□□<RegisteredOwner>CHCOM</RegisteredOwner>
□□□<DisableAutoDaylightTimeSet>false</DisableAutoDaylightTimeSet>
□□□<TimeZone>Romance Standard Time</TimeZone>
□□</component>
□</settings>
</unattend>
```

Plus de détails

Variables

Ces variables servent tout du long du fichier de configuration :

Paramètre générique	Description
{{architecture}}	Variable WAPT/WADS remplacée par l'architecture du système cible (ex: amd64).
{{hostname}}	Variable WAPT/WADS remplacée par le nom d'hôte de la machine à déployer.

Composants

Composant	Paramètres clés	Rôle / Justification
windowsPE		

Composant	Paramètres clés	Rôle / Justification
Microsoft-Windows-International-Core-WinPE	UILanguage, InputLocale, SystemLocale, UserLocale	Configuration linguistique initiale : Définit toutes les langues et formats régionaux sur Français (France) (fr-FR, code 040c).
Microsoft-Windows-Setup	InstallToAvailablePartition: true	Ciblage de l'installation : Indique au programme d'installation de formater et d'utiliser la première partition disponible pour l'OS.
Microsoft-Windows-Setup	Key: /IMAGE/INDEX Value: 6	Sélection de l'image (WIM) : Spécifie que l'image à installer est la 6ème image trouvée dans le fichier WIM de l'ISO. (Généralement, pour Windows Server, l'index 6 correspond à une édition spécifique, ex: Datacenter/Standard avec expérience utilisateur.)
Microsoft-Windows-Setup	AcceptEula: true	Contrat de Licence : Accepte automatiquement le contrat de licence utilisateur final (EULA).
Microsoft-Windows-Setup	ProductKey: <Key/>	Clé produit : Le champ est laissé vide ici, car la clé de produit sera généralement insérée plus tard dans la passe specialize (ou dans la phase de configuration WAPT).
offlineServicing		
Microsoft-Windows-PnpCustomizationsNonWinPE	Path: c:\installdir\drivers	Injection de Pilotes : Spécifie le chemin d'accès aux pilotes qui doivent être installés dans l'image Windows. WADS s'assure que les pilotes nécessaires sont bien placés dans ce répertoire temporaire (c:\installdir\drivers) avant l'exécution de cette passe.
generalize		
Microsoft-Windows-Security-SPP	SkipRearm: 1	Activation de la Licence : Empêche le processus Sysprep de "réarmer" l'état d'activation de la licence Windows.
specialize		
Microsoft-Windows-Shell-Setup	ComputerName: {{hostname}}	Nom de la machine : Le nom de l'ordinateur est défini en utilisant la variable fournie par WADS ({{hostname}}), garantissant que chaque machine déployée aura un nom unique et cohérent.

Composant	Paramètres clés	Rôle / Justification
Microsoft-Windows-Shell-Setup	ProductKey: VK7JG-NPHTM-C97JM-9MPGT-3V66T	Clé de licence : Insère une clé de produit. (Ceci est souvent une clé générique KMS/GVLK pour l'édition Standard ou Datacenter et nécessite un serveur KMS pour l'activation finale.)
Microsoft-Windows-UnattendedJoin	JoinDomain: adwapt.local	Jointure de Domaine : Joint l'ordinateur au domaine adwapt.local.
Microsoft-Windows-UnattendedJoin	Credentials	Authentification de la jointure : Fournit les identifiants pour effectuer la jointure au domaine (wadsjoin@adwapt.local / pass).
Microsoft-Windows-International-Core	UILanguage, InputLocale, etc.	Configuration linguistique de l'OS : Répète la configuration française pour l'environnement utilisateur final de l'OS.
Microsoft-Windows-Security-SPP-UX	SkipAutoActivation: true	Activation de la Licence : Ignore la tentative d'activation automatique de Windows.
Microsoft-Windows-SQMApi	CEIPEnabled: 0	Télémétrie : Désactive la participation au programme d'amélioration de l'expérience client (CEIP).
Microsoft-Windows-Deployment	RunSynchronousCommand	Script de post-installation : Exécute le script wads.exe qui met en place le script de démarrage final (c:\installdir\SetupComplete.cmd). Ce script est essentiel pour le déploiement WAPT/WADS car il est généralement responsable de l'installation finale de l'Agent WAPT et du lancement des paquets d'installation.
oobeSystem		
Microsoft-Windows-Shell-Setup	HideEULAPage: true, SkipUserOOBE: true, SkipMachineOOBE: true	Passer OOBE : Désactive tous les écrans d'expérience utilisateur initiale (questions de compte, paramètres Wi-Fi, EULA, etc.) pour rendre l'installation entièrement silencieuse.
Microsoft-Windows-Shell-Setup	TimeZone: Romance Standard Time	Fuseau Horaire : Définit le fuseau horaire, généralement sur l'heure d'Europe centrale (CET/CEST), cohérent avec la France.
Microsoft-Windows-Shell-Setup	RegisteredOrganization, RegisteredOwner	Informations d'enregistrement : Enregistre la machine sous l'organisation CHCOM.

Composant	Paramètres clés	Rôle / Justification
Microsoft-Windows-Shell-Setup	AutoLogon	Connexion automatique post-installation : Configure le système pour se connecter automatiquement une fois (LogonCount: 1) après le OOBE, en utilisant les identifiants de l'administrateur de domaine. Ceci permet au script final (SetupComplete.cmd) d'exécuter des tâches nécessitant une session utilisateur complète avant que l'utilisateur final ne voie le bureau.

Fonctionnalités

Phase du Déploiement	Composant	Fonctionnalité Clé du Script
windowsPE (Pré-Installation)	`Microsoft-Windows-International-Core-WinPE`	Configure la langue d'installation et toutes les locales initiales sur **Français (France)** (`fr-FR`)
	`Microsoft-Windows-Setup`	Installe le système d'exploitation en utilisant la **6ème image (index 6)** trouvée dans le fichier WIM de l'ISO
	`Microsoft-Windows-Setup`	Accepte automatiquement le contrat de licence utilisateur final (EULA)
	`Microsoft-Windows-Setup`	Précise au programme d'installation de cibler et formater la **première partition disponible** (`InstallToAvailablePartition: true`)
`offlineServicing` (Hors Ligne)	`Microsoft-Windows-PnpCustomizationsNonWinPE`	**Injecte des pilotes :** Ajoute le chemin `c:\installdir\drivers` comme source pour les pilotes à intégrer dans l'image Windows
`specialize` (Premier Démarrage)	`Microsoft-Windows-Shell-Setup`	Définit le **Nom de la machine** (`ComputerName`) en utilisant la variable WADS/WAPT `{{hostname}}` pour garantir un nom unique
	`Microsoft-Windows-UnattendedJoin`	**Jointure au Domaine :** Joint l'ordinateur au domaine `adwapt.local` en utilisant des identifiants spécifiques (`wadsjoin@adwapt.local` / `pass`)
	`Microsoft-Windows-Deployment`	**Post-installation WADS :** Exécute la commande `wads.exe` pour mettre en place le script de démarrage final (`SetupComplete.cmd`), crucial pour le lancement de l'Agent WAPT

Phase du Déploiement	Composant	Fonctionnalité Clé du Script
`oobeSystem` (Post-Configuration)	`Microsoft-Windows-Shell-Setup`	**Passer OOBE :** Contourne tous les écrans d'expérience utilisateur (EULA, comptes, Wi-Fi) pour une installation totalement silencieuse
	`Microsoft-Windows-Shell-Setup`	Configure la **Connexion automatique** au domaine (`adwapt.local`) en utilisant le compte administrateur une seule fois (`LogonCount: 1`), permettant l'exécution de tâches finales
	`Microsoft-Windows-Shell-Setup`	Définit le fuseau horaire sur **Romance Standard Time** (Heure d'Europe centrale/France)
	`Microsoft-Windows-Shell-Setup`	Définit l'organisation enregistrée sur **CHCOM**

Erreurs courantes

Il est défendable devant un service cybersécurité de dire que **mettre LAPS et BitLocker “dans l’unattend.xml” n’est ni essentiel, ni souhaitable**, car cela augmente la surface d’attaque sans apporter de bénéfice par rapport à un pilotage par GPO/MDM, qui est la pratique recommandée.[1][2]

LAPS : pourquoi éviter l’unattend.xml

- Les mots de passe locaux définis via des réponses d’installation (unattend.xml) sont stockés dans un fichier lisible sur le support d’installation et souvent laissé sur le disque après déploiement ; on trouve de nombreux retours de mots de passe admin en clair ou simplement encodés en Base64 dans des unattend.xml, ce qui est trivial à récupérer en post-compromission.[3][4]
- Des ressources de tests d’intrusion et de pentest listent explicitement les fichiers unattended comme cibles pour la recherche de mots de passe locaux afin d’escalader les privilèges, ce qui montre que ce vecteur est maintenant standard dans la kill-chain offensive.[4][5]
- Microsoft LAPS et Windows LAPS, au contraire, sont conçus pour être pilotés par stratégie (GPO/MDM) : on y configure centralement la rotation automatique, la complexité, l’intervalle de changement et le stockage sécurisé du mot de passe dans AD/Entra, sans jamais exposer ce mot de passe dans un fichier de réponse ou un script.[2][6]
- Des guides d’implémentation récents recommandent explicitement de créer une GPO dédiée (ex. C_LAPS) et de l’appliquer aux OU cibles, ce qui garantit une configuration homogène et auditables, tout en réduisant les secrets “cachés” dans l’infra de déploiement.[2]

En bref, inclure la logique LAPS dans l'unattend.xml reviendrait à réintroduire des mots de passe locaux statiques ou prévisibles dans un fichier facilement récupérable, alors que la bonne pratique actuelle est d'externaliser complètement cette gestion vers LAPS via GPO/MDM, avec rotation et stockage centralisé protégés.

BitLocker : pourquoi le laisser aux GPO/MDM

- Les bonnes pratiques BitLocker en entreprise recommandent d'utiliser des stratégies centralisées (GPO/Intune/MDM) pour : imposer l'algorithme de chiffrement, les exigences TPM/PIN, la sauvegarde des clés de récupération et l'activation automatique sur les machines cibles.[7][8][1]
- L'utilisation de Group Policy pour gérer BitLocker est explicitement citée comme une "best practice" afin de garantir homogénéité, traçabilité et gestion sécurisée des clés sur l'ensemble du parc, plutôt que de disperser la logique dans des scripts ou fichiers d'installation.[1][7]
- Unattend.xml permet bien de modifier certains comportements (par exemple désactiver le chiffrement automatique OEM via la clé PreventDeviceEncryption), mais ce fichier n'est ni conçu, ni recommandé comme point unique de contrôle de la politique de chiffrement, et son contenu peut être lu par un attaquant ayant accès au support ou au système.[9][8]
- Des retours d'attaque montrent que la recherche de fichiers XML d'installation (dont unattend.xml) fait partie des techniques standard pour découvrir des informations sensibles (clés, paramètres sécurité, éventuelles commandes BitLocker), ce qui va à l'encontre du principe de minimisation des secrets dans les artefacts de déploiement.[5][4]

En bref, l'activation et la configuration de BitLocker dans l'unattend.xml ne sont pas indispensables et peuvent même fragiliser la chaîne de déploiement si des paramètres sensibles y sont intégrés. L'approche recommandée est de déclencher et de gouverner BitLocker via des GPO/MDM dédiées, qui centralisent les clés, permettent l'audit et évitent l'exposition d'informations critiques dans des fichiers d'installation.

Sources :

1 2 3 4 5 6 7 8 9 10 11 12