Réussir son déploiement Ubuntu en Environnement Active Directory

Benoît Métrot

Réussir son déploiement Ubuntu en environnement Active Directory

Benoit Métrot
Institut PPrime
11 Boulevard Marie et Pierre Curie
86 360 Chasseneuil du Poitou

Résumé

Face à l'installation d'un grand nombre de machines, une approche est le recours aux outils proposés par le monde DevOps, et plus particulièrement par la composante opérationnelle. C'est ainsi que, sur le campus du Futuroscope de l'Université Poitiers, au sein d'un parc de machines à vocation pédagogique, construit autour d'un domaine Active Directory, une solution de déploiement du système d'exploitation GNU/Linux a vu le jour.

Elle est la rencontre de la distribution Ubuntu avec l'outil Ansible. Ce dernier assure la configuration complète des ordinateurs du parc (installation de logiciels, configuration réseau, jonction au domaine Active Directory, synchronisation horaire, définition des comptes administrateurs, montage des volumes réseau…). Ansible met également en musique l'ensemble des composants serveur nécessaires au processus de déploiement (service TFTP pour l'installation du système via PXE, configuration réseau DHCP, distribution des fichiers d'installation).

Cela en fait une solution intégrée facile à mettre en œuvre. Elle utilise les capacités offertes par Ansible pour la rendre paramétrable et adaptable à des contextes variés. L'originalité du projet tient aussi dans la dimension de partage et d'échange qu'il porte. En effet, soutenu par le service formation de l'établissement, une offre de formation interne s'est construite et a permis de former des correspondants informatiques à l'usage de cette solution, afin qu'ils puissent, déployer le système Ubuntu dans leurs environnements.

Mots-clefs

Ubuntu, Ansible, Déploiement, Automatisation…

1 Introduction

Parmi les différents sites géographiques de l'Université de Poitiers, celui du Futuroscope est situé à une dizaine de kilomètres de Poitiers, sur la commune de Chasseneuil du Poitou. Sa création remonte à la construction du parc du Futuroscope, où une volonté politique forte visait à créer une technopole regroupant des entreprises de services et des établissements d'enseignement supérieur et de recherche scientifique. SP2MI est l'autre nom de ce campus. Derrière cet acronyme, qui signifie Sciences Physique, Mathématique, Mécanique et Informatique, se cachent plusieurs départements d'enseignement et laboratoires de recherche qui animent ce campus.

Dans cet article, nous nous intéresserons aux moyens informatiques mis à disposition des étudiants au sein de ces départements d'enseignements. Classiquement, il s'agit de salles de travaux pratiques composées d'ordinateurs fixes, répartis sur plusieurs bâtiments. Plus particulièrement, nous expliquerons comment s'est industrialisé le déploiement du système GNU/Linux (distribution

Ubuntu) et surtout comment cette démarche a été proposée aux autres composantes de l'établissement.

2 Contexte

2.1 Une organisation humaine avant tout

Les départements d'enseignement présents sur le campus du Futuroscope appartiennent tous à l'Unité de Formation et de Recherche (UFR) des Sciences Fondamentales et Appliquées (SFA). Il s'agit de :

- département Physique ;
- département Mécanique ;
- département Mathématique ;
- département Informatique ;
- département Électronique, Électrotechnique et Automatique.

Chaque département possède un parc informatique (salles de travaux pratiques), qui est placé sous la responsabilité d'un personnel technique. La plupart de ces personnes partagent leur temps de travail entre ces parcs et celui des laboratoires de recherches auxquels elles sont également affectés. À partir de 2010, un travail en équipe sans fondement hiérarchique s'est mis en place. La force du collectif nous a permis de construire un système d'information pédagogique mutualisé entre les cinq départements d'enseignement. Les différentes tâches se sont naturellement réparties en fonction des appétences et du savoir-faire de chacun. Néanmoins, les décisions structurantes se sont toujours prises collectivement afin de répondre au mieux aux besoins de chaque entité.

2.2 Un parc construit autour d'un ActiveDirectory

Dans ce parc pédagogique, toutes les machines fonctionnent sous un système d'exploitation Windows 10. Mais bon nombre d'entre elles possèdent un double amorçage avec un système GNU/Linux (distribution Ubuntu 22.04 au moment de l'écriture de l'article). Plusieurs raisons nous ont orienté vers le choix d'un annuaire Active Directory comme cœur du système.

Tout d'abord, les possibilités d'administration système et de configuration centralisées offertes par les stratégies de groupes (GPO) apportent de réelles facilités dans le travail au quotidien pour les systèmes Windows. S'en priver aurait été contre-productif.

Ensuite, la DSI de notre établissement développe et maintient un connecteur capable d'injecter des utilisateurs Active Directory à partir du système d'information RH de l'établissement. Il s'agit d'un service Windows, qui s'installe sur un contrôleur de domaine. Grâce au mécanisme de synchronisation des mots de passe intégré à l'outil, les étudiants et les enseignants retrouvent, sur les postes de travail des salles de travaux pratiques, le même identifiant et mot de passe que sur leur messagerie ou leur environnement numérique de travail.

L'interopérabilité avec les systèmes Linux a tout d'abord été assurée avec une authentification Kerberos utilisant le module *krb5* de la pile PAM (Pluggable Authentifcation Module) couplé au

module NSS (Name Service Switch) pour obtenir les informations d'identifications des utilisateurs (identifiant, uid, gid, répertoire d'accueil). Toutes ces informations sont d'ailleurs stockées au sein d'Active Directory dans les attributs UNIX prévus à cet effet. C'est le connecteur développé par notre DSI qui se charge de peupler ces attributs, afin de garantir une cohérence globale. Avec le déploiement de la distribution Ubuntu 18.04, nous sommes passés à l'utilisation de la couche logicielle SSSD développée par RedHat. Là où le module *pam_krb5* faisait uniquement de la vérification de mot de passe avec le protocole Kerberos, avec SSSD, chaque machine fait partie intégrante du royaume Kerberos sous-jacent à Active Directory. Chaque machine est authentifiée dans le domaine (elle possède un compte d'ordinateur dans Active Directory), ce qui ouvre des possibilités de SSO, notamment pour les montages de volumes de fichiers réseau.

Avec le recul, ce choix d'Active Directory comme cœur du système informatique était pertinent et continue de l'être aujourd'hui. Il assure par exemple une continuité de la sécurité en propageant les désactivations de comptes depuis le système d'information de l'établissement, lorsqu'une compromission (souvent de l'hameçonnage via messagerie électronique) est détectée par le RSSI de l'établissement...

2.3 Et les données utilisateurs dans tout cela ?

Un des besoins primordiaux exprimés par l'ensemble des départements était la rationalisation de l'occupation des salles. Un étudiant du département d'informatique doit pouvoir se rendre dans une salle du département de mathématiques, ouvrir sa session (Windows ou GNU/Linux) sur les machines et retrouver ses données. L'objectif ici est de faciliter la construction des emplois du temps en organisant une séance de travaux pratiques dans une salle d'un autre département, lorsque les salles habituelles du département sont toutes occupées. À terme, cela maximise le taux d'occupation des salles et limite leur multiplication. Ce qui est plutôt vertueux dans le contexte de développement durable actuel.

Côté Windows, la réponse à cette problématique passe par le recours aux profils itinérants. Les données de chaque utilisateur sont chargées depuis un serveur de fichiers central, lors de l'ouverture de session. Au moment de la fermeture de session, les données sont renvoyées vers le serveur. Pour soulager l'usage du réseau, notamment en fin de séance quand tous les profils remontent vers le serveur central, nous avons essayé d'utiliser un temps la redirection de dossiers proposée par Windows. Ce mécanisme, qui consiste à faire pointer plusieurs dossiers du profil utilisateur (mes documents, bureau, application data...) vers des dossiers d'un volume réseau, nous posait plusieurs problèmes. Le premier était la sensibilisation des utilisateurs. Malgré les explications, de nombreux utilisateurs continuaient à déposer (et donc à perdre !) leur travail dans des endroits non redirigés (racine du profil par exemple). Le second était, contre toute attente, lié à la performance. Un logiciel tel que Solidworks semble accéder constamment à plusieurs fichiers de son dossier dédié dans *application data* du profil Windows. La multitude d'accès réseau engendrés pénalisait énormément la fluidité de son interface graphique.

Côté Linux, le concept de profil itinérant n'existe pas. C'est un serveur de fichiers NFS central qui distribue les répertoires d'accueil aux postes clients. Une tentative d'utilisation de serveur NFS sous Windows (afin de mutualiser profil Windows et répertoire d'accueil Linux) s'est avérée être un échec. La prise en charge de la casse dans le nommage des fichiers ainsi que des attributs Unix

étaient mal supportés dans l'implémentation du service NFS par Microsoft. Cela causait des dysfonctionnements dans les logiciels côté Linux notamment sur les fichiers cachés de paramétrage. Voulant rester le plus proche du monde Unix, notamment pour les TP de système d'exploitation qui utilisent les mécanismes de verrouillage de fichiers Posix ou les ACL étendus, nous avons préféré un système à base de NFS plutôt que de passer sur d'autres technologies tels que CIFS.

Aujourd'hui les profils utilisateurs Windows sont donc stockés sur un serveur de fichiers tournant sous Windows Server. Les répertoires d'accueil Linux sont déposés sur un serveur de fichiers tournant sous distribution GNU/Linux Debian. Néanmoins, l'itinérance des données utilisateurs demande à ce que les versions de logiciels soient autant que faire se peut dans des versions comparables, notamment pour éviter des problèmes de format de fichiers de paramétrage (profil Firefox).

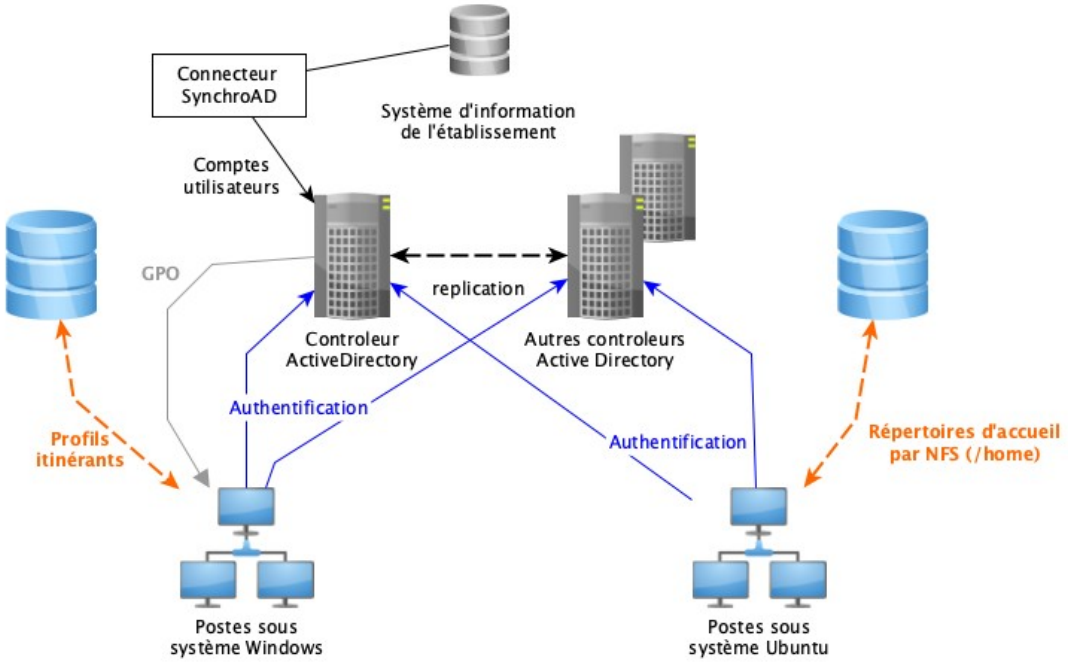

Figure 1: Vue d'ensemble du système d'information pédagogique

3 Une industrialisation progressive

3.1 Les postes Windows

Dans les premières années, les systèmes d'exploitation Windows étaient déployés avec des outils tiers tels que Norton Ghost ou Clonezilla. C'est avec l'arrivée de Windows 7 que nous nous sommes tournés vers les outils de déploiement proposés par Microsoft.

Aujourd'hui les postes Windows sont déployés avec le couple WDS et MDT. Le principe est de construire un poste modèle pour chaque département, qui embarque l'ensemble des logiciels demandés par la formation. Ce modèle est généralisé (sysprep) puis capturé avec les outils WDS pour générer une image. Elle est alors prête à être déployée sur une ou plusieurs salles. Le recours à MDT permet d'assurer, au moment du déploiement de l'image, des opérations particulières comme

l'intégration automatique au domaine Active Directory ou le partitionnement du disque. Ainsi les machines Windows sont déployées dans un mode *zéro touch* (sans intervention clavier).

3.2 Premiers pas avec Ubuntu 16.04

Parallèlement à la mise en production des outils de déploiement Microsoft, nous avons également cherché à mettre en place un processus d'installation automatisé pour les systèmes Linux, en lieu et place des outils de clonage utilisés. Nous avons commencé avec une installation minimaliste réalisée par le réseau à l'aide d'un serveur PXE (protocoles DHCP et TFTP). L'installeur Ubuntu 16.04 supportait alors un fichier de réponse (technologie *preseed*) automatisant la totalité du processus d'installation.

Une fois le système installé, une série de scripts Shell, largement inspirée du projet PLACO [1], était chargée de configurer l'ensemble des composants nécessaires au fonctionnement (authentification, impression, répertoire d'accueil, etc.) et d'y installer les logiciels demandés, qu'ils figurent ou non parmi les paquets de la distribution.

Cette solution artisanale donnait des résultats satisfaisants et nous a démontré qu'il était possible, à partir d'une machine vierge, d'installer Windows avec WDS/MDT (en réservant une partition sur le disque), puis d'installer la distribution Ubuntu de façon automatisée.

3.3 Ansible débarque avec Ubuntu 18.04 LTS

Forts de cette première expérience d'automatisation, nous avons cherché à professionnaliser les choses en utilisant des outils de configuration bien connus. J'ai découvert l'outil Puppet lorsque je faisais partie de l'équipe PLMteam du réseau Mathrice (http://www.mathrice.fr). Son modèle d'exécution avec un agent ne correspondait pas au besoin. Le principe d'Ansible avec une machine centrale qui agit par SSH sur un ensemble de machines cibles collait tout à fait avec les scripts Shell utilisés jusqu'ici. La transition s'est donc faite toute en douceur vers ce nouvel outil [2]. Il a été adopté facilement par le reste de l'équipe.

Le système Windows (désormais en version 10) est déployé avec le couple WDS/MDT, qui, pour les machines en double amorçage, laisse une partie du disque inoccupée. Cet espace libre est utilisé pour effectuer une installation minimale de la distribution Ubuntu à travers le réseau. À cette époque, la distribution est encore installable à partir du Debian-Installer qui supporte une automatisation complète via un fichier de préconfiguration *preseed*. Cette phase ne nécessite pas d'intervention au clavier de l'opérateur. Ansible prend le relais pour effectuer la configuration du poste ainsi que les installations de logiciels, à l'aide d'un ensemble de *playbooks* écrits spécifiquement. Ils effectuent toutes les opérations de configuration des machines qui avaient déjà été identifiées dans la version précédente (16.04).

En complément de l'installation initiale, la maintenance du parc était prise en charge par Ansible. Qu'il s'agisse d'ajouter de nouveaux logiciels, d'effectuer les mises à jour de sécurité, l'outil a été d'une aide précieuse pendant toute la période d'exploitation de cette distribution.

4 Partage au sein de l’établissement

Après le déploiement de la distribution Ubuntu 18.04 LTS sur le parc informatique des départements d’enseignement du campus du Futuroscope, d’autres collègues gestionnaires de parc au sein de l’Université de Poitiers m’ont questionné sur la méthode utilisée. Plusieurs personnes ont alors manifesté un intérêt pour cette solution. L’idée de partager cette expérience avec d’autres personnes venait de germer.

4.1 Création d’un kit de déploiement

Le printemps 2022 vient de se terminer, les machines GNU/Linux du campus du Futuroscope tournent encore avec la distribution Ubuntu 18.04. Les versions de logiciels commencent à être obsolètes, il devient nécessaire d’envisager une migration vers une distribution plus récente. Après échange avec les responsables des départements, il est décidé de prolonger l’aventure avec la distribution Ubuntu mais dans sa version 22.04, qui vient de sortir.

Plutôt que de simplement remettre au goût du jour l’ensemble de *playbooks* utilisé jusqu’ici, je décide de construire un kit de déploiement complet qui pourra être facilement partagé avec d’autres gestionnaires de parc intéressés. Une solution de déploiement se compose généralement d’un ensemble de briques qu’il faut intégrer les unes avec les autres. Plutôt que d’écrire un guide qui expliquerait la marche à suivre pour installer et configurer ces briques, je m’oriente vers un kit dont l’objectif est triple :

- configurer un poste de contrôle qui concentre les services nécessaires à la réalisation des deux autres objectifs avec notamment
  - une configuration IP automatique par DHCP si aucun serveur n’est disponible sur le réseau ;
  - amorçage réseau du programme d’installation Ubuntu ;
  - distribution des composants du programme d’installation Ubuntu et des logiciels à installer ;
  - application des recettes (*playbooks*) Ansible fournies dans le kit pour la configuration des postes clients.
- déployer une installation minimale de la distribution Ubuntu 22.04 sur un parc de machines ;
- installer les logiciels voulus (qu’ils soient inclus dans les paquets de la distribution ou non) et configurer les machines pour qu’elles s’intègrent au système d’information (un domaine Active Directory dans notre cas) ;

La dernière partie de l’article détaille les aspects techniques de la solution et explique comment est construit ce kit.

4.2 Deux sessions de formations pour apprendre

Même avec le meilleur des manuels, il est souvent difficile de prendre en main un outil, lorsque l’on se retrouve seul face à son écran. C’est pourquoi, dès le début, je me suis associé avec le service de

formation permanente de l'établissement, dans le but de proposer un stage de prise en main du kit de déploiement.

Une première session de formation est organisée dès le mois de juin 2022. Elle rassemble une dizaine de gestionnaires de parc, tous en postes dans des composantes et services de l'Université de Poitiers. Intitulée « Réussir son déploiement Ubuntu en environnement Active Directory », cette formation d'une durée de trois jours s'adresse aux gestionnaires de parc désireux d'automatiser le déploiement de la distribution GNU/Linux Ubuntu sur un parc de machines, dans un contexte Active Directory. Peu importe qu'il s'agisse de machines en double amorçage, l'authentification des utilisateurs s'effectue sur les contrôleurs Active Directory.

Très axée sur la pratique, elle se déroule sur machine où chaque stagiaire dispose de trois ordinateurs : un pour simuler le poste de contrôle, deux autres comme postes clients sur lesquels sera déployé le système Ubuntu. L'idée est de se mettre en situation avec un mini parc informatique. Un domaine Active Directory de test est également disponible sur le réseau afin de fournir un annuaire d'identification et d'authentification. Elle s'organise en trois temps.

1. *Configuration du poste de contrôle.* Après quelques notions théoriques sur les composants qui entrent en jeu lors de l'installation automatisée du système via PXE, le participant est invité à déployer par lui-même l'ensemble des composants nécessaires. Il est aidé dans cette tâche par l'utilisation de *playbooks* Ansible prêts à l'emploi qui automatisent toutes les opérations : service TFTP, service DHCP, récupération des images d'installation, etc.
2. *Déploiement de systèmes Ubuntu.* Dès le milieu de la première demi-journée, les stagiaires disposent d'un poste de contrôle totalement fonctionnel. Ils commencent alors leur premier déploiement de système à partir du poste de contrôle qu'ils ont configuré. Jusqu'au milieu du deuxième jour, ils vont apprendre comment appliquer les *playbooks* Ansible proposés pour installer des logiciels et configurer les machines de façon à ce qu'elles puissent s'authentifier sur le domaine Active Directory.
3. *Adaptation du kit aux besoins de chacun.* Le reste du temps est consacré aux mécanismes de personnalisation du kit afin de l'adapter à son contexte. Qu'il s'agisse de définir l'adresse du proxy web, d'installer un ensemble de paquets de la distribution ou de déployer des logiciels spécifiques hors distribution, tout cela est prétexte à découvrir le fonctionnement d'Ansible pour amener la personne vers l'écriture de ses propres *playbooks*.

Bien évidemment, trois jours ne sont pas suffisants pour devenir un spécialiste Ansible, mais cela est suffisant pour comprendre son fonctionnement et la structure du kit de déploiement.

À l'issue de la première session, les retours furent très positifs. Plusieurs personnes se sont lancées dans la mise en production du kit au sein de leurs parcs respectifs. Elles ont ainsi réussi à réutiliser le kit dans des contextes différents de celui pour lequel il avait été conçu.

Pour bénéficier de conditions d'organisation satisfaisantes, le nombre de places avait été limité à dix, lors de la première session. Après cette première expérience réussie, il restait des collègues intéressés. C'est pourquoi, toujours avec le soutien du service de formation permanente de l'établissement, une nouvelle session de formation a vu le jour en juin 2023. C'est à nouveau une dizaine de personnes qui ont suivi le stage. Le contenu de cette nouvelle session était très similaire à

la première. Fort des retours des premiers participants, le programme a été amélioré sur quelques points, comme l'adaptation du kit aux besoins de chacun. En effet, c'est un des aspects essentiels, et il était nécessaire de proposer davantage d'explications et d'exercices pour améliorer la compréhension.

5 Chef ! Donnez-moi la recette !

Techniquement, le kit de déploiement proposé aux participants de la formation est une collection de *playbooks* et de rôles Ansible. Tout est disponible sur la forge du CNRS au sein d'un projet dédié[1].

5.1 Le poste de contrôle

Le poste de contrôle est la machine centrale dans la mise en œuvre du kit. Il s'agit d'une machine sur laquelle sont installés les services réseau nécessaires au processus (TFTP, DHCP, partage des ISOs Ubuntu). C'est également à partir de cette machine que seront appliqués les *playbooks* Ansible. Le dimensionnement de cette machine ne dépend pas de la taille du parc, mais du nombre de machines que l'on souhaite traiter simultanément. Une station de travail dotée d'un processeur type i7 ou i9 avec 16Go ou 32Go de mémoire vive sera capable d'adresser plus d'une cinquantaine de machines simultanément. Le téléchargement des paquets logiciels Ubuntu s'effectue directement sur les dépôts officiels, le stockage local et le réseau seront uniquement utilisés pour le déploiement de logiciels hors distribution et cela est externalisable sur un serveur dédié, le cas échéant.

Dans bien des cas, une station de travail standard mais dédiée à cette fonction pourra donc faire office de poste de contrôle. Dans le cadre d'une administration mutualisée (un parc administré par plusieurs personnes), une machine virtuelle accessible par SSH est aussi une bonne solution. Il est même envisageable d'utiliser plusieurs postes de contrôle pour répartir la charge entre plusieurs gestionnaires.

Concrètement, la mise en place du poste de contrôle se résume à quatre opérations :

- installer une distribution GNU/Linux Ubuntu (en version 22.04 au moment de l'écriture de ces lignes) en suivant les recommandations inscrites dans les supports de formations ;
- installer Ansible à partir du dépôt ppa proposé par l'éditeur, afin d'obtenir une version récente ;
- construire un couple de clés privée/publique SSH qui servira de sésame d'authentification à Ansible pour s'authentifier sur l'ensemble des machines du parc ;
- Obtenir le kit de déploiement depuis le dépôt GIT, puis appliquer le *playbook* nommé playbook-ubuntutoolkit.yml contenu dans le kit.

Voici donc les quelques commandes liées aux actions citées ci-dessus :

```
ssh-keygen -a 100 -t ed25519 -f ~/.ssh/id_ansible
git clone https://src.koda.cnrs.fr/benoit.metrot.2/ubuntu-toolkit.git
cd ubuntu-toolkit
ansible-playbook -i inventaire/ --ask-become playbook-ubuntutoolkit.yml
```

1 https://src.koda.cnrs.fr/benoit.metrot.2/ubuntu-toolkit

Le terme kit de déploiement désigne en réalité l'ensemble des fichiers contenus dans le dossier ubuntu-toolkit.

5.2 Inventaire

Afin de pouvoir adresser les machines sur lesquelles il doit intervenir, Ansible a besoin de connaître l'inventaire des machines du parc. Il s'effectue par le biais d'un fichier texte sur deux colonnes (le caractère tabulation sépare les colonnes). Chaque ligne qui ne commence pas par le caractère # ou [désigne une machine. La première colonne contient le nom de la machine et la seconde un ensemble de variables tel que son adresse IP. Lorsque le fichier contient des sections à la manière d'un fichier ".ini " (une section commence par une ligne où le nom de section est encadré par les caractères [et]), chaque section correspond à un groupe de machines.

```
# Fichier inventaire avec groupe
[dualboot]
dualboot1-l       ansible_host=172.16.203.140
dualboot2-l       ansible_host=172.16.203.141

[linux]
ubuntu1           ansible_host=172.16.203.142
ubuntu2           ansible_host=172.16.203.143

[ubuntu:children]
dualboot
linux
```

Dans l'exemple de fichier d'inventaire reproduit ci-dessus, 4 machines et trois groupes sont définis :

- Les machines dualboot1-l, dualboot2-l, ubuntu1 et ubtunu2 ayant respectivement comme adresses IP 172.16.203.140, 172.16.203.141, 172.16.203.142 et 172.16.203.143 ;
- Un groupe de machines nommé dualboot contenant les machines dualboot1-l et dualboot2-l ;
- Un groupe de machines nommé linux contenant les deux machines ubuntu1 et ubuntu2 ;
- Un groupe de machines nommé ubuntu qui rassemble toutes les machines contenues dans les groupes dualboot et Linux (:children est un élément syntaxique pour indiquer qu'il s'agit d'un groupe de groupe), soit les 4 machines au total.

Avec une syntaxe relativement simple et la possibilité pour une machine d'appartenir à plusieurs groupes (via les groupes de groupes), les machines peuvent être classées selon plusieurs critères (site géographique, nom de bâtiment, numéro de salle, caractéristique matérielle spécifique, double amorçage, etc.). Par convention, l'inventaire du parc est constitué par un ou plusieurs fichiers textes (sans extension) déposés dans le répertoire d'inventaire situé à la racine du kit de déploiement.

5.3 Variables de groupes et variables machines

Les variables machines et les variables de groupe proposées par Ansible sont au cœur du mécanisme d'adaptation proposé dans le kit. En effet, un *playbook* tel que celui reproduit ci-dessous, qui est utilisé pour la configuration des postes clients, fait appel à un ou plusieurs rôles.

```
---
#
# Playbook de post-installation de l'ensemble des machines du parc
#
# Exemples d'usage du playbook :
# ansible-playbook -i inventaire playbook-postinstall.yml
# ansible-playbook -i inventaire -l machine1,machine2 playbook-postinstall.yml
#

- name: Postinstallation des machines du parc
  hosts: all:!localhost
  vars:
  roles:
    - role: roles/ubuntu
    - role: roles/logitheque
```

Un rôle est une structure logique Ansible qui regroupe un ensemble d'opérations à effectuer sur une machine, dans le but de configurer un ou plusieurs composants du système. D'une certaine manière cela organise le code Ansible de façon à le rendre modulaire et à faciliter sa maintenance. Tous les fichiers inhérents à rôle sont regroupés dans un dossier dédié. Ils sont tous rangés dans le dossier « roles », que l'on retrouve à la racine du kit. Il contient la collection de tous les rôles Ansibles proposés, parmi lesquels :

- *audt* : configure les services utilisés dans le processus de déploiement sur le poste de contrôle (TFTP, DHCP, distribution du programme d'installation…) ;
- *ubuntu :* installe et configure les postes clients (définition du nom d'hôte, configuration réseau, synchronisation horaire, routage des emails systèmes, rotation des journaux…) ;
- *logitheque :* déploie les logiciels nécessaires au bon fonctionnement du système qu'ils soient inclus dans les dépôts Ubuntu ou non (archive tar, binaire seul) ;
- *admember :* active l'authentification sur le domaine Active Directory en utilisant le composant système sssd.

En fonction de l'environnement où la machine est installée, il est nécessaire de spécifier un certain nombre de paramètres tels que : le nom du serveur DNS, le nom du domaine Active Directory, l'adresse d'un serveur proxy web… Plutôt que de modifier les fichiers de configuration contenus dans les rôles, le kit propose un certain nombre de variables, qui agissent directement sur leurs fonctionnements. Par exemple, un des paramètres du rôle "ubuntu" est *ubuntu_param_time_server.* Il spécifie le nom du serveur NTP à utiliser pour synchroniser l'horloge des postes clients.

La puissance du mécanisme de variable proposé par Ansible réside dans la possibilité de définir une variable pour une machine spécifique ou pour un groupe de machines. Les groupes de machines sont ceux définis dans le(s) fichier(s) d'inventaire(s). Il est alors possible avec une définition de

variable (une ligne d'un fichier texte), d'impacter la configuration de plusieurs dizaines de machines.

Ces définitions de variables s'effectuent dans deux sous-dossiers du dossier inventaire que l'on trouve à la racine du kit :

- *host_vars* : chaque fichier YAML de ce dossier porte le nom d'une machine du parc et contient les définitions de variables spécifiques à la machine. Ainsi le fichier inventaire/host_vars/dualboot1-l.yml contient les définitions de variables qui seront appliquées à la machine nommée dualboot1-l.
- *group_vars* :chaque fichier YAML de ce dossier contient les définitions de variables appliquées à toutes les machines du groupe dont le nom correspond au nom de fichier. Ainsi, les variables définies dans le fichier inventaire/group_vars/dualboot.yml seront appliquées à toutes les machines du groupe dualboot. Un exemple de contenu de ce fichier est présenté ci-dessous.

```
---
#
# Définition des variables pour les machines du groupe dualboot
#

# Paramétrage DNS pour utiliser le DNS ActiveDirectory en priorité
ubuntu_param_network_use_dnsdhcp: false
ubuntu_param_network_dnslist:
 - 172.16.203.1
 - 172.16.203.2

# Nom du domaine Windows à rejoindre
admember_param_domain_name: RESTAURANT.PRIVE

# Machines en dual boot
admember_param_dual_boot: true

# Paquets supplémentaires de la distribution à installer
logitheque_package_list_salle:
 - vlc
 - audacity
```

Toutes les variables proposées dans le kit possèdent des valeurs par défaut. Il n'est donc pas nécessaire de définir de façon exhaustive l'ensemble des variables proposées. De même, il n'est pas nécessaire de créer des fichiers de définitions de variables pour toutes les machines et pour tous les groupes. Ansible définit un certain nombre de règles de priorité. Ainsi, lorsqu'une variable est définie à la fois dans un groupe et pour une machine spécifique, c'est la valeur définie pour la machine (dans le dossier host_vars) qui est retenue.

5.4 Processus de déploiement d'une machine

La succession d'opérations nécessaires au déploiement du système Ubuntu sur un poste du parc est la suivante.

1. S’il s’agit d’une machine dotée d’un double amorçage, commencer par installer le système Windows (avec les outils WDS et MDT si besoin), tout en laissant un espace libre suffisant pour Ubuntu sur le disque dur ;
2. Déclarer la machine dans le serveur DHCP intégré au poste de contrôle s’il a été configuré ou dans le serveur du réseau. Les champs *next-server* et *filename* indiqueront à la machine qu’elle doit s’amorcer sur l’installeur Ubuntu diffusé via TFTP par le poste de contrôle ;
3. Allumer la machine et la faire démarrer sur le réseau. L’installeur Ubuntu se lance et installe un système minimal dans l’espace disponible sur le disque. L’installeur est configuré pour poser le minimum de questions. Les paramètres de clavier, de langue et autres sont spécifiés via les variables du kit vues ci-dessus. L’opérateur garde néanmoins la possibilité de définir un partitionnement particulier ou choisir le disque d’installation lorsque la machine en possède plusieurs. Lorsque le programme d’installation se termine, la machine redémarre sous le système Ubuntu nouvellement installé. En ayant installé Ubuntu après Windows, c’est GRUB qui prend en charge la séquence d’amorçage de l’ordinateur. À ce titre, il propose un menu permettant de choisir le système d’exploitation à utiliser (Ubuntu ou Windows) ;
4. À partir du poste de contrôle, appliquer successivement les *playbooks* playbook-postinstall.yml et playbook-activedirectrory.yml. La machine est alors configurée automatiquement par Ansible. Les paquets et logiciels supplémentaires sont installés. La machine fait maintenant partie du domaine Active Directory et peut l’utiliser pour authentifier les utilisateurs…

6 Perspectives

Dans cet article, je vous ai présenté mon retour d’expérience sur la construction d’un kit de déploiement pour la distribution GNU/Linux Ubuntu. Depuis la première version, ce kit a gagné en fonctionnalité en intégrant de nouveaux rôles Ansible, à la demande de collègues ayant des besoins d’intégration spécifiques. Ce concept de kit est une chose à faire évoluer et à partager afin que chacun puisse l’adapter à son contexte sans repartir de zéro.

2024 est l’année des JRES mais également celle de la sortie d’une nouvelle version LTS de la distribution Ubuntu. Un travail de mise à niveau pour supporter les évolutions a commencé et devrait être achevé pour le jour de la conférence.

La réalisation de formations auprès d’autres gestionnaires de parc a été une expérience très enrichissante. Ce serait formidable que la publication de cet article soit le point de départ de nouvelles collaborations avec d’autres établissements qui souhaiteraient faire bénéficier leurs personnels d’une telle formation.

Bibliographie

[1] Mathrice, PLACO un générateur de plate-forme collaborative au service des communautés scientifiques. Poster au congrès JRES2009, Nantes, décembre 2009 ; https://math.univ-angers.fr/perso/jaclin/home/_attachments/2009/placo_poster.pdf.

[2] Métrot B. Retour d'expérience comparatif Ansible & Puppet, Rencontre Mathrice de Toulon, Octobre 2019 ; https://indico.math.cnrs.fr/event/4598/contributions/4061/attachments/2493/3085/bm.jm-oct2019-ansible-puppet.pdf.