



HAL
open science

Squest – Comment rendre les infrastructures facilement consommables

Gauthier Jonet, Olivier Le Monnier, Fantin Lepont

► **To cite this version:**

Gauthier Jonet, Olivier Le Monnier, Fantin Lepont. Squest – Comment rendre les infrastructures facilement consommables. JRES (Journées réseaux de l'enseignement et de la recherche) 2024, Renater, Dec 2024, Rennes, France. <hal-04893172v2>

HAL Id: hal-04893172

<https://hal.science/hal-04893172v2>

Submitted on 17 Jan 2025

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire HAL, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



Distributed under a Creative Commons CC BY-NC 4.0 - Attribution - Non-commercial use - International License

Squest – Comment rendre les infrastructures facilement consommables

Gauthier Jonet

Olivier Le Monnier

Fantin Lepont

Université de Caen Normandie
Direction du système d'information (DSI)
Pôle infrastructures
Esplanade de la Paix
14000 Caen

Résumé

Depuis 2010, notre DSI centralise la gestion des infrastructures et fait face à une demande croissante de serveurs virtuels. Alors, au pôle infrastructures, nous automatisons le déploiement des configurations de ces serveurs virtuels avec Puppet.

L'augmentation de la demande rend indispensable la gestion du cycle de vie. Une application existante est adaptée pour couvrir ce besoin. Toutefois, décorrélée de l'automatisation, elle s'avérera un facteur limitant.

Parallèlement, pour d'autres besoins, nous commençons à utiliser Ansible, AWX et Netbox. Cela nous permet d'étendre le périmètre couvert en automatisant le provisionnement des serveurs virtuels.

Cette nouvelle phase diminue encore le temps de livraison, ce qui participe à l'augmentation de la demande. Cependant, certaines étapes chronophages sont encore gérées manuellement. Finalement l'automatisation engendre un surplus de travail.

Nous avons donc besoin d'un outil pour gérer la chaîne de provisionnement, de la demande jusqu'à la livraison du serveur virtuel complet, prêt à l'emploi.

Squest, logiciel open source développé par l'équipe infrastructures de HPE pour ses propres besoins, offre une interface permettant à l'utilisateur de commander ses serveurs, de suivre sa demande en temps réel, d'interagir avec un administrateur et d'obtenir un retour lorsque son serveur est prêt. Il peut aussi commander des opérations « du deuxième jour » : gérer les snapshots et les ressources, demander un certificat TLS...

Squest offre une grande autonomie à nos utilisateurs et libère les administrateurs des tâches répétitives.

Mots-clés

infrastructures, self-service, automatisation

Table des matières

1 Contexte.....	3
2 Gestion du cycle de vie.....	3
3 Outils utilisés pour l’automatisation.....	4
4 Paradoxe de Jevons.....	5
5 Objectif : De la commande en ligne à la livraison.....	6
6 Squest.....	6
6.1 Qu’est-ce que c’est ?.....	6
6.2 Notre cas d’utilisation.....	7
6.3 Stratégie de déploiement.....	9
6.4 Gestion du projet.....	10
6.5 Courbe d’adoption.....	10
7 Perspectives.....	12

1 Contexte

Fondée en 1432, l'Université de Caen Normandie accueille aujourd'hui plus 30 000 étudiants sur sept campus Caennais et cinq autres sites répartis sur toute la Normandie occidentale.

La direction du système d'information centralise la gestion des infrastructures systèmes et réseau. Son pôle infrastructures, composé de quatre administrateurs systèmes et deux administrateurs réseau, exploite 150 serveurs physiques, 820 serveurs virtuels, 26 routeurs, 550 commutateurs et 830 bornes WiFi.

Depuis 2010, ces deux équipes sont engagées conjointement dans un processus d'automatisation de la configuration des infrastructures. La solution adoptée alors était fondée sur Puppet, avec une exécution périodique de l'agent installé sur les serveurs virtuels, appelant un serveur centralisant les manifestes (classes) et hébergeant l'outil d'inventaire Foreman chargé de la classification des nœuds.

Courant 2013, les manifestes sont versionnés dans un dépôt git ; en 2016 ce dépôt est migré sur une instance Gitlab locale.

2 Gestion du cycle de vie

Outre les infrastructures, la DSI centralise également la messagerie opérée sous Zimbra. Autour de 2013, un modèle de gestion des adresses fonctionnelles est mis en place permettant aux messages de suivre les fonctions plutôt que les utilisateurs. Un outil baptisé GeDemande est développé pour gérer le cycle de vie de ces adresses fonctionnelles selon un circuit de validation administratif et technique.

En 2018, face à l'explosion du nombre de serveurs virtuels et au constat que certains tombent en désuétude, GeDemande est adapté pour gérer leur péremption après une année, sauf renouvellement explicite. Le circuit de validation administratif, inadapté à la particularité technique des serveurs virtuels est vite abandonné. Les administrateurs système sont notifiés par mail des demandes saisies par les utilisateurs et doivent valider techniquement puis acquitter le provisionnement une fois celui-ci réalisé. La plupart du temps, un dialogue avec le demandeur est requis pour préciser la demande et un ticket est alors créé dans la plateforme d'assistance.

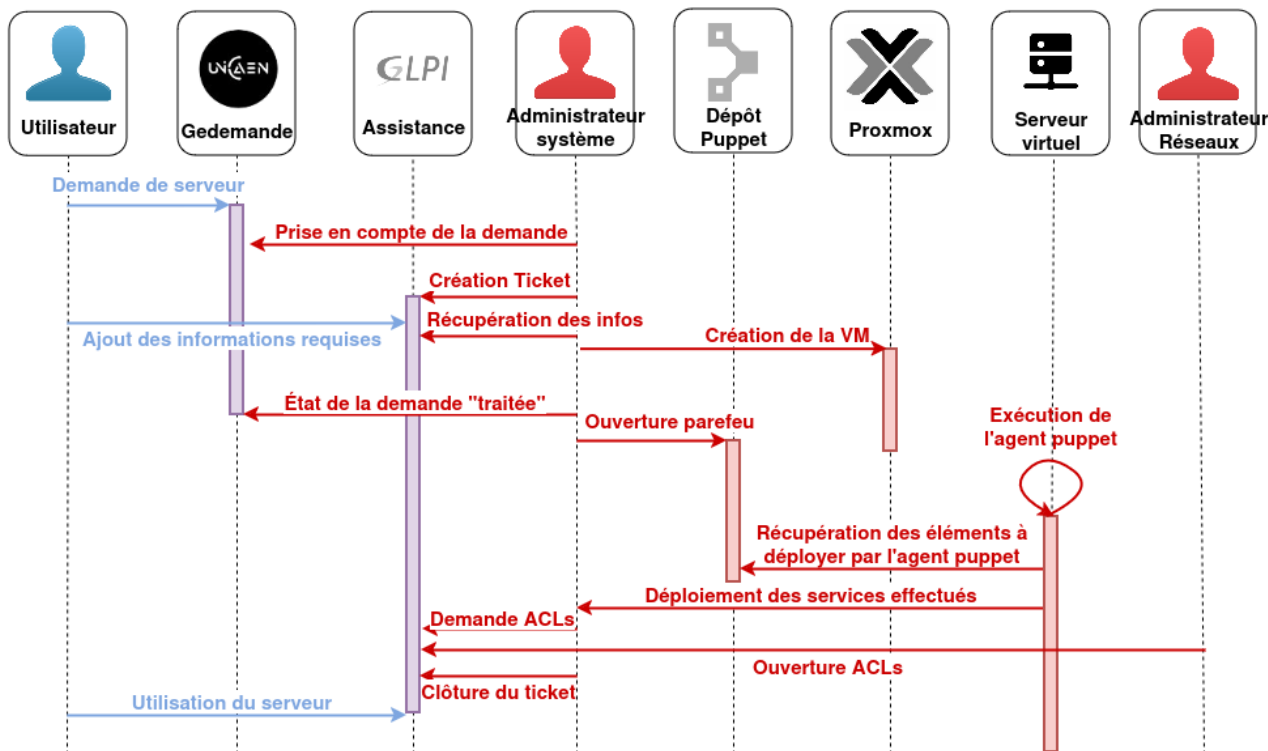


Figure 1: Workflow manuel de traitement des demandes de serveurs virtuels

Cette volonté de gérer le cycle de vie en utilisant une application décorrélée des besoins et de l'automatisation existante s'avère finalement un frein dans le traitement des demandes de provisionnement de serveurs virtuels.

3 Outils utilisés pour l'automatisation

Lors de l'été 2020, le réseau du centre de données est réformé. Une fabric IP fondée sur des actifs Cumulus Linux est mise en place. L'outil fourni par l'éditeur pour le provisionnement s'appuie sur des *playbooks* Ansible. Deux nouveaux outils sont alors mis en place : Netbox et AWX.

Netbox référence l'ensemble des informations techniques utiles au déploiement réalisé par Ansible. AWX centralise l'exécution des *playbooks*, permet notamment d'accéder aux journaux et de déclencher ces exécutions via des appels HTTP (*webhook*).

Début 2021, les premiers *playbooks* pour le provisionnement de serveurs entrent en service. Ils permettent notamment de déployer les hyperviseurs VMware Horizon offrant une solution de bureaux virtuels massivement utilisée lors du deuxième confinement qui aura lieu l'automne suivant.

À la rentrée 2022, le provisionnement automatique de serveurs virtuels est implémenté. Il s'appuie sur la source de vérité Netbox. Un formulaire intégré à Netbox déclenche l'exécution du *playbook* sur AWX via un *webhook*.

Les demandes formulées dans l'application GeDemande sont toujours prises en charge manuellement par les administrateurs système qui remplissent et valident le formulaire Netbox.

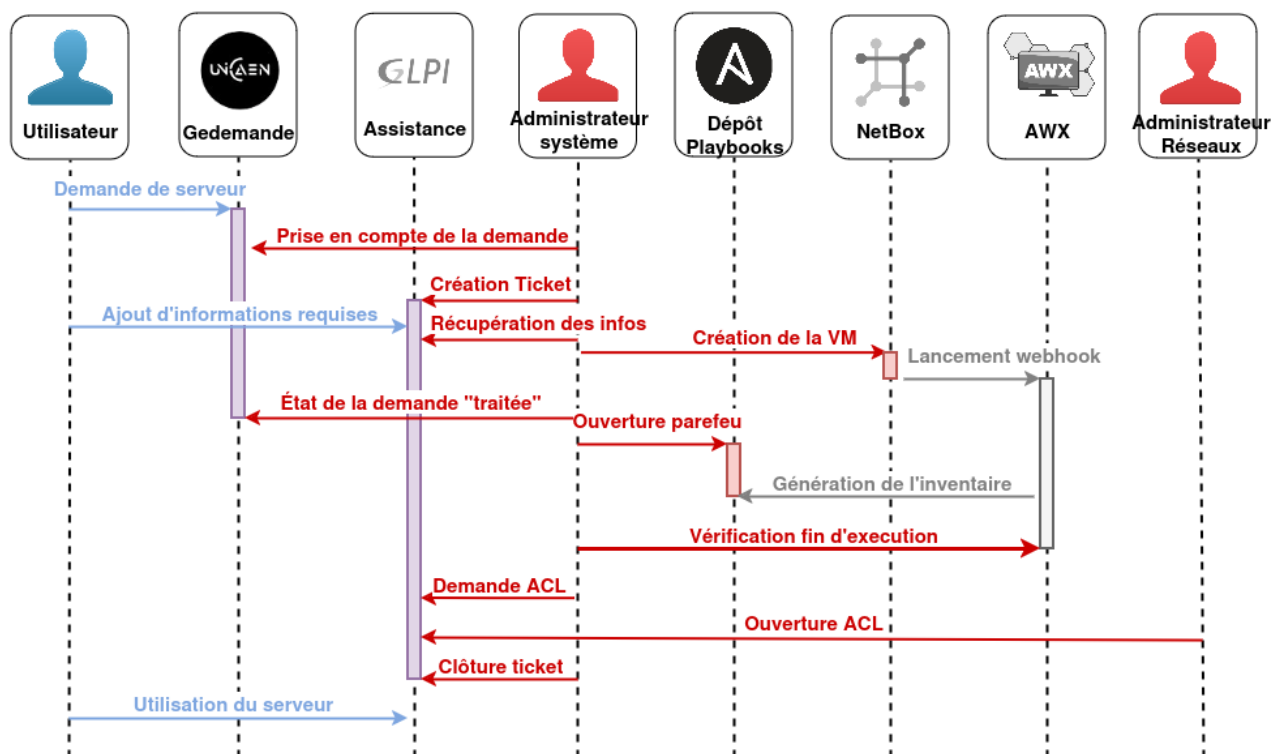


Figure 2: Workflow de traitement des demandes de serveurs virtuels avec Netbox

Cependant certaines tâches restent manuelles : validation et acquittement dans GeDemande, création et prise en charge d'un ticket dans le Helpdesk le cas échéant, adaptation de la configuration du pare-feu local, mise en place des ACL sur les pare-feux périmétriques par un administrateur réseau via un nouveau ticket Helpdesk...

4 Paradoxe de Jevons

L'automatisation, même partielle, du provisionnement des serveurs virtuels raccourcit le délai de livraison de manière notable. Les utilisateurs en demandent donc plus facilement et en plus grand nombre : serveurs de tests, de préproduction, de développement, répartition des applications hébergées sur un serveur mutualisé vers des serveurs dédiés...

Toutefois, l'automatisation n'étant toujours pas complète, une fois de plus, certaines tâches, les plus fastidieuses et chronophages, se trouvent démultipliées. Finalement, le travail des administrateurs système chargés de gérer les demandes de provisionnement de serveurs virtuels est rendu plus répétitif.

5 Objectif : De la commande en ligne à la livraison

Notre besoin correspond au cahier des charges suivant, dressé collectivement lors de la campagne des entretiens professionnels 2023 :

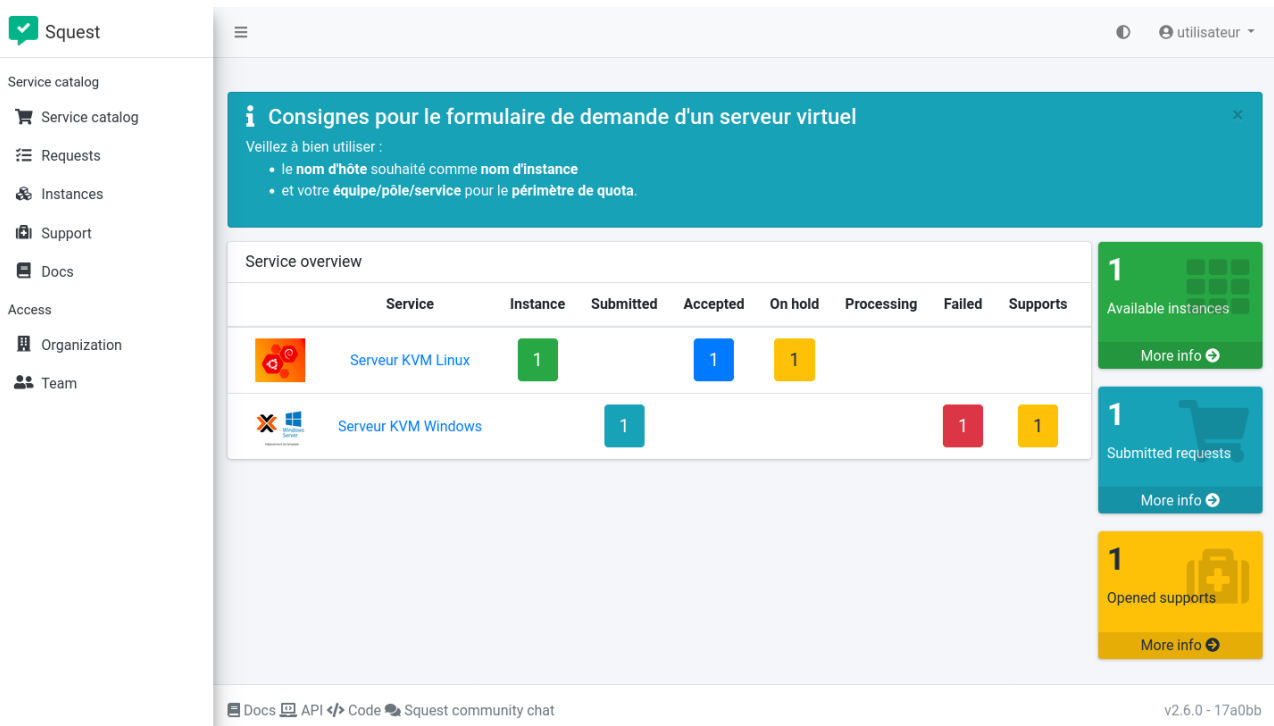
« Permettre la création de serveurs virtuels à la demande :

- systèmes d'exploitation Linux et Windows ;
- profiler les machines (RAM/CPU, applis) ;
- demandes exclusivement par les supports infos qui sont au contact des utilisateurs demandeurs ;
- mise en place de règles d'utilisation à définir, comme par exemple : cycle de vie impliquant un renouvellement régulier, *renouvellement interdit pour des versions d'OS dépassées, transfert de référent en cas de départ du demandeur initial.* »

Nous avons donc besoin d'un outil pour gérer la chaîne de provisionnement, de la demande jusqu'à la livraison du serveur virtuel complet, prêt à l'emploi.

6 Squest

6.1 Qu'est-ce que c'est ?



The screenshot displays the Squest user interface. On the left is a navigation menu with items like 'Service catalog', 'Requests', 'Instances', 'Support', 'Docs', 'Access', 'Organization', and 'Team'. The main content area features a 'Service overview' table and three summary cards on the right.

Service	Instance	Submitted	Accepted	On hold	Processing	Failed	Supports
Serveur KVM Linux	1		1	1			
Serveur KVM Windows		1				1	1

Summary cards on the right:

- 1 Available instances
- 1 Submitted requests
- 1 Opened supports

Footer: Docs API Code Squest community chat v2.6.0 - 17a0bb

Figure 3: Page d'accueil Squest du point de vue de l'utilisateur

Squest est un logiciel open source développé par l'équipe infrastructure de HPE pour répondre à ses propres besoins. C'est une interface web fondée sur Django et couplée à une instance AWX. Elle permet aux utilisateurs de commander des *services*. Chaque service présente au moins une *opération* de création. Chaque opération est associée à un modèle de tâche sur AWX. L'opération de création d'un service crée une *instance*. Les opérations réalisables une fois les instances créées, sont dites « *opérations de jour 2* » qui seront définies plus tard dans cet article.

6.2 Notre cas d'utilisation

Dans notre cas, les services seront des serveurs virtuels.

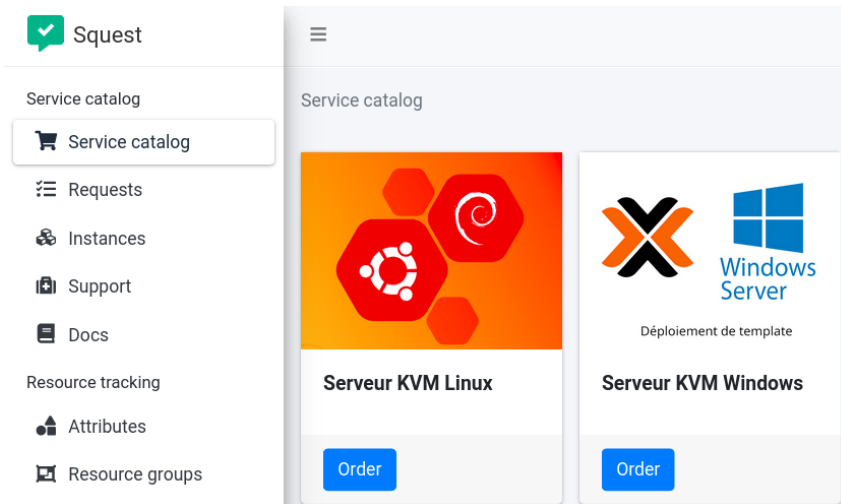


Figure 4: Catalogue de services d'infrastructures actuel

Plusieurs opérations de création seront proposées, correspondant à des modèles préétablis selon différents niveaux de ressource.

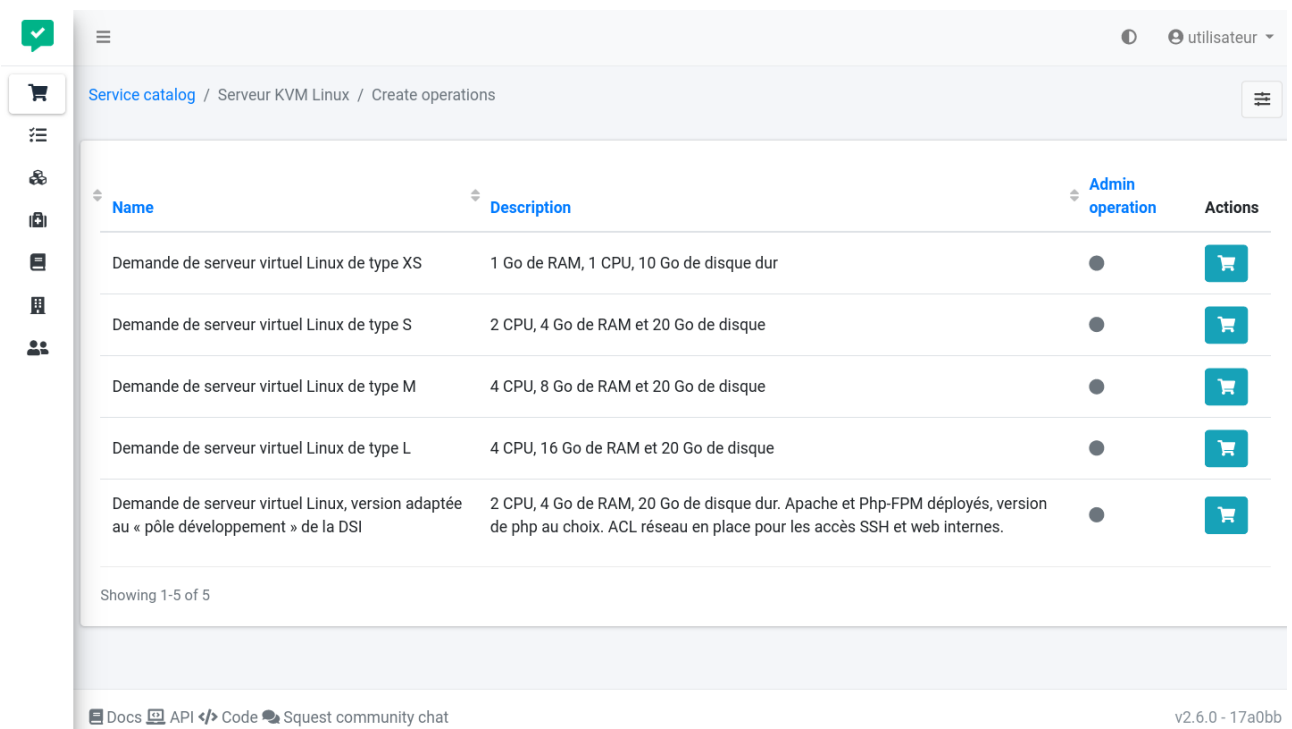


Figure 5: Opérations de création/Modèles disponibles

Les utilisateurs pourront suivre le traitement de leur demande en temps réel. Ils pourront interagir avec un administrateur au travers des fonctionnalités de support intégrées.

The screenshot displays a request tracking page for request #463. The interface is divided into several sections:

- Request #463 Summary:**
 - Instance: [ktest-fl-suppr](#)
 - Service: Serveur KVM Linux
 - Operation: Demande de serveur virtuel Linux de type XS
 - Operation type: **CREATE**
 - Quota scope: Direction du Système d'Information (DSI) - Pôle Infrastructures
 - Request state: **COMPLETE**
 - Instance state: **DELETED**
 - User: [utilisateur](#)
 - Date submitted: 03 Jun, 2024 14:17
 - Date complete: 03 Jun, 2024 14:26
 - Accepted by: [lepont211](#)
 - Processed by: [lepont211](#)
- Requested by:** [utilisateur](#) (03 Jun, 2024 14:17)
 - platform: Debian Bookworm
 - vlan: SERVEUR
 - comments: Test suppression
 - webaccess: non
 - web: non
 - Spec:


```
{
            "FQDN": "ktest-fl-suppr.unicaen.fr",
            "IP": "10.14.130.230",
            "CPU": 1,
            "RAM": 1,
            "Disk": "10 Go",
            "VLAN": "SERVEUR (34)",
            "ProxmoxID": "330",
            "NetboxID": "1111",
            "End_of_life": "2025-06-03"
          }
```
- Comments:** **COMPLETE**
 - [lepont211](#) (03 Jun, 2024 14:17): Test suppression
- Submitted:**
 - Admin review:
 - Survey:
 - Accepted by [lepont211](#)
- Accepted:**
 - Process the request:
 - Sent for processing by [lepont211](#)
- Processing:**
 - Processing (03 Jun, 2024 14:26):
 - Request processed successfully.
- Complete:**

Figure 6: Page de suivi des différents phases d'une opération

Ils seront notifiés lorsque leurs instances seront disponibles. Des opérations de jour 2 leur permettront : de gérer les snapshots et les ressources, de demander un certificat TLS, de rendre disponible leur service web au travers des mandataires inverses...

The screenshot displays a cloud management dashboard for an instance named 'kdocenstock' (ID: 241). The instance is running 'Serveur KVM Linux' and is in an 'AVAILABLE' state. The 'Operations' tab is active, showing a list of tasks:

Name	Description	Type
Suppression du serveur virtuel	Supprimer le serveur virtuel, les ACL sur les parefeux et le certificat	DELETE
Prolongation du serveur virtuel	Prolonger la durée de vie d'un serveur d'un an - Disponible 3 mois avant la fin de validité de votre serveur.	UPDATE
Prise d'un snapshot	Prendre un snapshot. Il sera supprimé automatiquement dans 14 jours.	UPDATE
Ajout d'un disque dur local	Ajouter un disque local sur le stockage CEPH Data	UPDATE
Fourniture d'un certificat SSL	Générer et copier un certificat SSL dans /etc/certificats	UPDATE
Restauration d'un snapshot	Restaure toutes les données depuis la prise du snapshot.	UPDATE
Ouverture de l'accès web depuis l'extérieur	Configuration automatique des Reverse Proxy pour l'accès depuis Internet.	UPDATE
Archiver le serveur	Archive le serveur	UPDATE
Reboot de la VM	Arrêt forcé et redémarrage complet de la VM.	UPDATE

Figure 7: Opération dites « de jour 2 »

6.3 Stratégie de déploiement

Comme vu précédemment, au fil des années et des outils utilisés, la DSI a utilisé différentes méthodes pour déployer ses serveurs virtuels (Puppet puis Ansible). Afin de simplifier l'exploitation et d'harmoniser les configurations de ces serveurs, nous avons décidé de profiter de la mise en place de Squest pour accélérer la mise à niveau de notre parc.

Après avoir dressé un état des lieux des caractéristiques des serveurs virtuels à sélectionner : propriétaire, système d'exploitation, méthode de déploiement utilisée..., nous avons choisi quels serveurs virtuels seraient importés dans Squest (en tant qu'instance).

Cette intégration, réalisée le 29 février 2024, a permis aux propriétaires des serveurs concernés d'avoir immédiatement accès aux opérations de « jour 2 ».

Les serveurs virtuels écartés devront être réinstallés, soit parce que non conformes d'un point de vue technique (nous avons souhaité ne pas réintégrer les containers LXC), ou parce qu'ils ont été déployés avec un outil jugé obsolète (nous devons donc adapter les modules Puppet en playbook Ansible).

6.4 Gestion du projet

Squest est déployé par un *playbook* sur un serveur virtuel dédié (lui-même intégré à Squest !) sous la forme d'une composition d'images Docker. Une documentation simple permet de sauvegarder, mettre à jour et restaurer la plateforme en quelques commandes.

Nous avons, dans un premier temps, installé une maquette afin de découvrir l'outil et ses fonctions. À l'issue de cet essai, nous avons adapté les *playbooks* qui étaient jusqu'à lors exécutés manuellement ou au travers de Netbox. Cette maquette de préproduction a été ouverte progressivement de quelques administrateurs système au départ, puis à l'ensemble du pôle infrastructures, jusqu'à l'ensemble des informaticiens membres ou non de la DSI, nous permettant à chaque phase d'améliorer le service rendu.

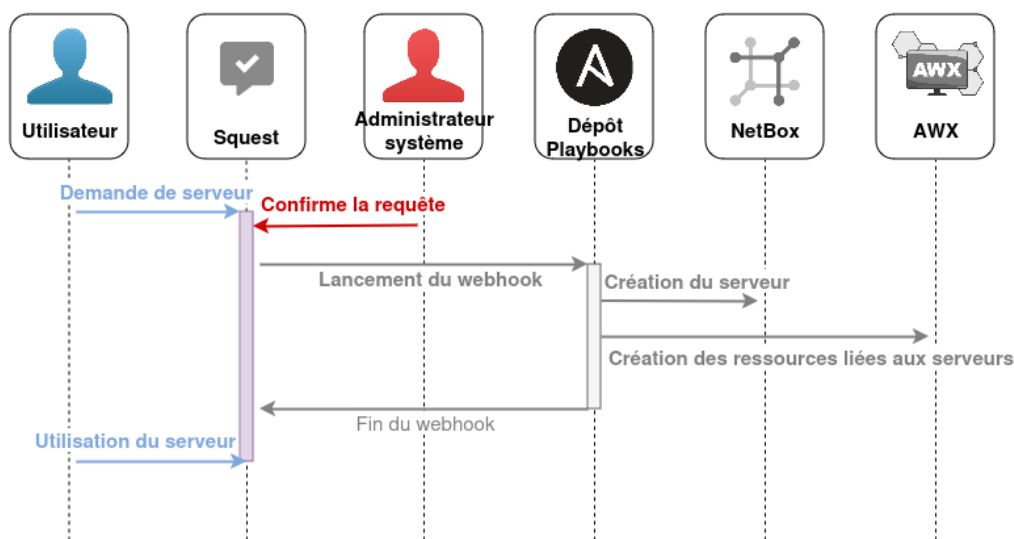


Figure 8: Workflow de traitement des demandes de serveurs virtuels avec Squest

6.5 Courbe d'adoption

Depuis la présentation à l'ensemble des utilisateurs (administrateurs d'application, développeurs de la DSI, informaticiens de proximité en composantes...) réalisée le 12 avril 2024, le service est en production. Les premières semaines, l'utilisation de l'outil était assez modérée, bien que le petit nombre d'utilisateurs ait déployé un nombre conséquent de serveurs virtuels¹.

Le premier import des serveurs virtuels compatibles a eu lieu en 03/2024, s'en est suivi une période de tests avec une équipe restreinte, puis une mise en production officielle le 12/04/2024.

¹ La DSI a fait le choix de n'appliquer aucun quota pour ses utilisateurs, bien que Squest le permette.

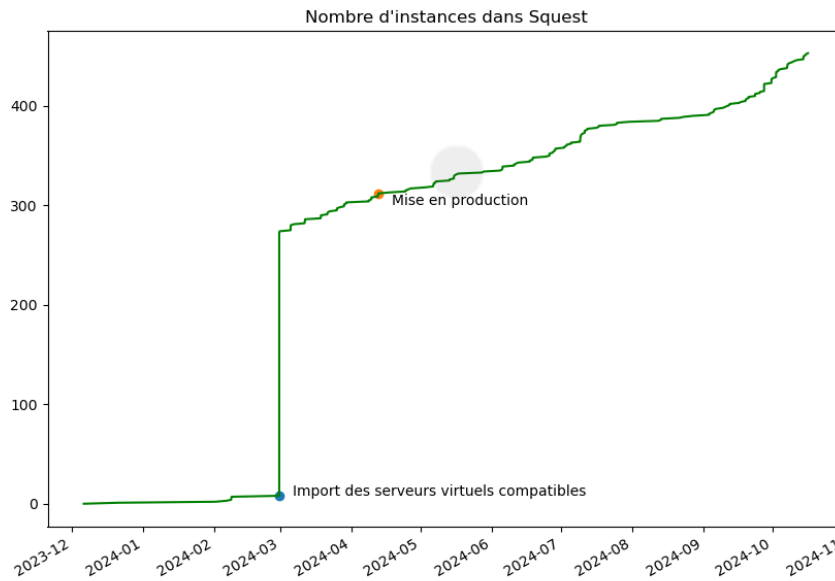


Figure 9: Évolution du nombre d'instances actives gérées depuis Squest

Depuis sa mise en production, nous observons une augmentation régulière d'instances créées depuis Squest sur notre infrastructure de virtualisation.

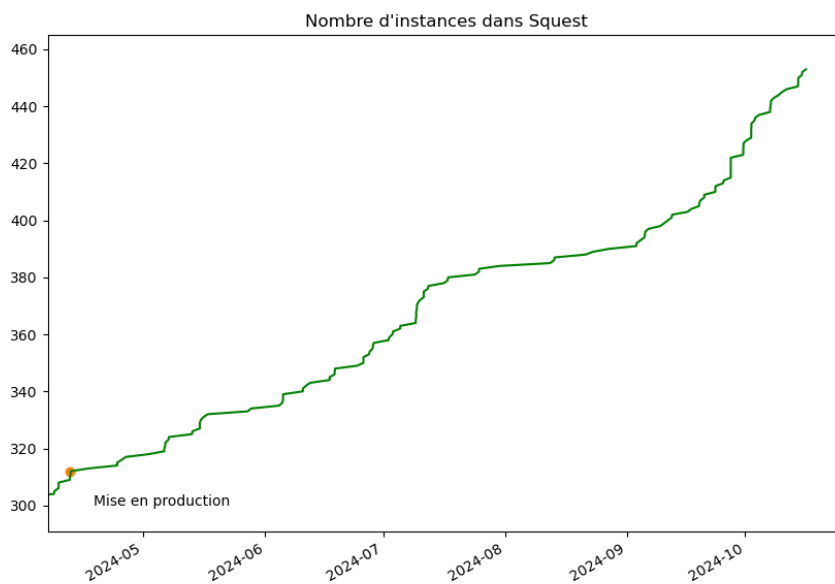


Figure 10: Évolution depuis la mise en production

L'outil occasionne régulièrement des retours toujours très positifs, le nombre et la diversité des usagers augmentent régulièrement, et il recense une part toujours plus importante de l'ensemble des

serveurs virtuels. Près de 50 % du parc de serveurs virtuels toutes technologies confondues est aujourd'hui présent dans Squest². Perspectives

Convaincus de l'intérêt de Squest, de nombreuses perspectives s'offrent à nous afin de mieux répondre aux besoins des utilisateurs : en offrant de nouveaux services (autres technologies de virtualisation telle que VMware, notamment pour la partie VDI, espaces de stockage, ACL réseau) et en permettant de nouvelles opérations sur les instances déjà déployées.

Bibliographie

- [1] Sispheor. Squest passe en v2 + retour d'expérience sur la pile Ansible/RHAAP/Squest ; <https://linuxfr.org/news/squest-passe-en-v2-retour-d-experience-sur-la-pile-ansible-rhaap-squest>
- [2] Documentation ; <https://hewlettpackard.github.io/squest/2.6.0/>
- [3] Code source ; <https://github.com/HewlettPackard/squest>
- [4] Chat room ; https://app.gitter.im/?updated=1.11.61#/room/#HewlettPackard_squest:gitter.im

² Aucun système de facturation interne n'est mis en œuvre aujourd'hui.