Les
DOSSIERS
LeNetDevOps

Rencontres NetDevOps édition #1

SEPTEMBRE 2025

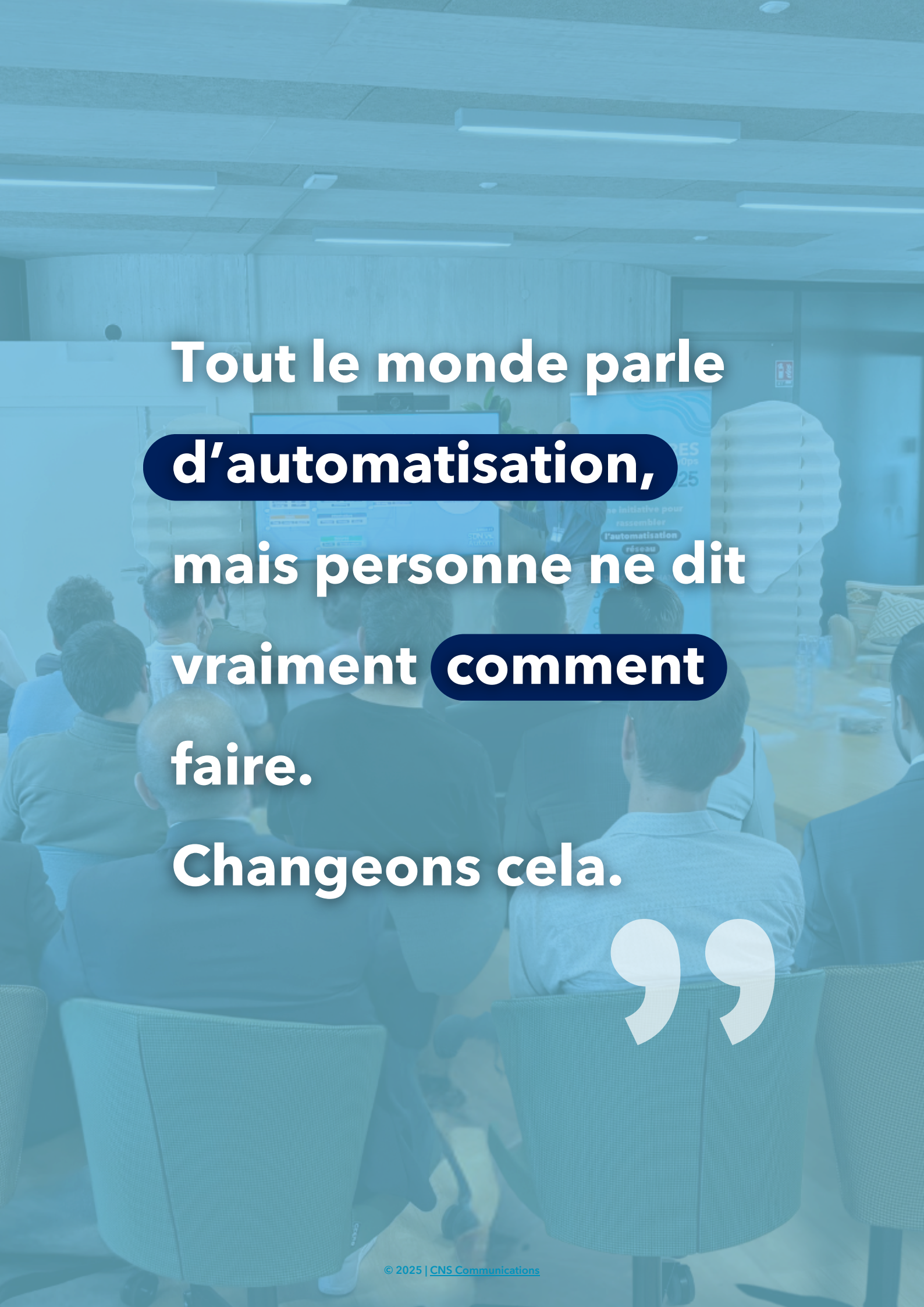
SDN versus Automatisation : Duo, ou Duel ?

Ce dossier enrichit la collection de
Dossiers Thématiques proposés par
la communauté LeNetDevOps.

Il s'agit de la deuxième publication
de la collection.

Sommaire

01. Édito



Édito

30 SEPT
2025
LYON

Pour la deuxième fois depuis le lancement de LeNetDevOps, plus de 50 participants se sont réunis le temps d'un après-midi, pour partager leurs conseils, expériences et pour mutualiser leurs approches.

Une nouvelle étape clé ! Le 30 septembre à Lyon, la communauté francophone de l'automatisation des réseaux s'est retrouvée à nouveau.

Après le premier événement à Paris (ce mois de juin 2025), qui a posé les fondements de la communauté et a vu naître un engouement, à la fois pour la dynamique et pour le format, nous avons alimenté cet élan, en organisant une nouvelle édition des Rencontres. Nouveau thème, nouveaux intervenants et toujours le même format, une demi-journée pensée autour de retours terrain, de moments de débats et d'une soirée networking. Plus de cinquante personnes se sont réunies, de débutants à experts, pour partager leur méthodologie, confronter leurs visions et convictions, approfondir leurs pratiques et questionner les fondations d'une vision commune et d'un savoir-faire collectif.

Cette Rencontre l'a montré : les membres LeNetDevOps attendent ce lieu d'échange.

Cet événement offre un espace pour faire évoluer les connaissances en automatisation réseau, en questionnant chaque fois un nouvel angle. En septembre, nous nous sommes penchés sur la question :

SDN versus automatisation : duo, ou duel ?

Édito

Voici un nouveau Dossier thématique, pour enrichir les apprentissages et partager les expériences.

Cette série de Dossiers thématiques vise à agréger et documenter les temps forts de la communauté.

La présente et deuxième publication prolonge l'élan impulsé par le 30 septembre. Au fil de ces pages, vous êtes invité à

Découvrir les enseignements de l'édition #1 des Rencontres

➔ Nous reviendrons sur les cinq prises de parole des professionnels qui y sont intervenus (Capgemini, Cisco, CNS Communications, INSA Lyon et Red Hat) qui ont partagé des retours nés de leurs propres expériences terrain. Un panel large, comprenant un architecte réseau, un constructeur réseau, un ingénieur consultant, un chercheur universitaire et un éditeur de solutions d'automatisation, pour offrir des perspectives, éclairages et pratiques plurielles autour du sujet thématique.

➔ L'équipe LeNetDevOps vous propose également plusieurs points de réflexion complémentaires, pour élargir et poursuivre le débat.

Prenez part à la dynamique LeNetDevOps.

Ensemble, participons à fédérer les acteurs de la filière en offrant des espaces structurants pour les échanges : lors des Rencontres, en ligne et lors de moments plus informels

Objectifs de ce Dossier

Présenter et rappeler les concepts de SDN et leurs liens avec l'automatisation ;

Synthétiser les apports de chaque intervention ;

Offrir des clés de lecture additionnelles aux Rencontres, pour mieux aborder ce sujet.

Bonne lecture !

02. Le SDN

Le SDN & automatisation

Le Software-Defined Networking (SDN) est une approche qui consiste à séparer les fonctions du réseau en plusieurs plans logiques, pour en améliorer la flexibilité et la programmabilité.

Le SDN propose une gestion centralisée d'équipements réseau et des fonctions d'automatisation.

L'objectif porté par cette Rencontre #1 était d'évaluer les aspects de ce SDN qui entrent en résonance – ou en conflit – avec la démarche d'automatisation d'un réseau, afin de déterminer si ces concepts sont complémentaires, ou finalement, plutôt opposés.

Le plan de contrôle (Control plane) est chargé d'analyser l'état du réseau, de prendre les décisions dynamiques de routage ou encore, d'appliquer les politiques de configuration. Ces fonctions sont centralisées au sein d'un contrôleur, qui pilote les équipements du plan de données via des interfaces programmables.

Le plan de données (Data plane) assure l'acheminement effectif du trafic réseau selon les règles définies par le plan de contrôle. Il gère le traitement, la transmission et le filtrage des paquets entre les équipements.

Le plan d'administration (Management plane) est responsable de la supervision, de la configuration et du suivi global du réseau. Il permet aux administrateurs de déployer, surveiller et maintenir les équipements et services via des outils ou tableaux de bord, souvent intégrés au contrôleur central.

Né des travaux de recherche des années 2000, le concept s'est concrétisé avec l'apparition du protocole OpenFlow, avant d'être structuré par la création de l'Open Networking Foundation (ONF).

L'objectif du SDN est de rendre le réseau plus **agile** **automatisable** et **adaptable** aux besoins des infrastructures modernes.

...Pour des besoins tels que ceux des infrastructures liées au cloud et à la virtualisation.

Malgré ces travaux initiés, les efforts de standardisation ont été freinés par un contexte concurrentiel.

.

Un contexte qui pousse finalement les constructeurs à **développer leurs propres implémentations**, du sur-mesure, souvent basées sur des protocoles propriétaires, plutôt que d'utiliser des standards ouverts.

Une approche que l'on retrouve aujourd'hui dans différentes solutions commerciales → avec des protocoles privés, fermés, qui sont spécifiques à chaque marque.

Le concept se résume désormais au pilotage de différents périmètres du réseau

(tels que les commutateurs, bornes Wi-Fi, routeurs WAN, firewalls, etc.)

Cela au travers de contrôleurs comme Meraki Dashboard, Aruba Central, Arista CloudVision, FortiManager ou Panorama, par exemple.

Les mêmes contrôleurs intègrent désormais des fonctions avancées de gestion, d'orchestration et d'automatisation.

Parallèlement, des outils tels qu'Ansible, Terraform, Python ou les API REST poussent encore plus loin cette logique, en traitant le réseau comme un composant logiciel à part entière (Infrastructure as Code).

C'est précisément à cette intersection se pose la question que nous traiterons dans ce présent dossier :

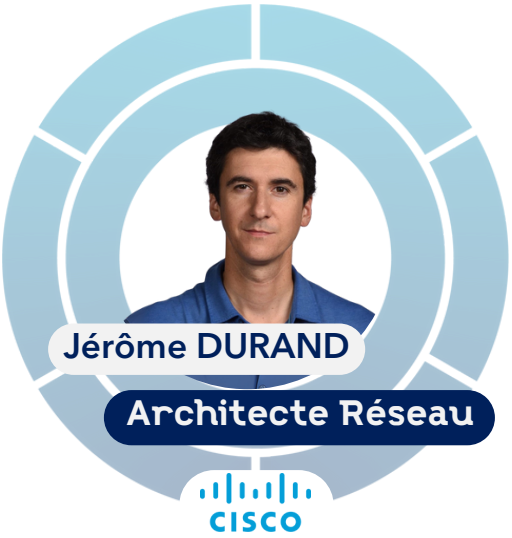
Dans la modernisation des infrastructures,

Le SDN et l'automatisation réseau poursuivent-ils le même objectif, ou s'agit-il d'approches distinctes – voire redondantes ou concurrentes ?

03. Ils l'ont vécu.

Des **interventions** de professionnels, sur des cas vécus : des **exemples concrets** pour creuser le sujet SDN versus Automatisation.

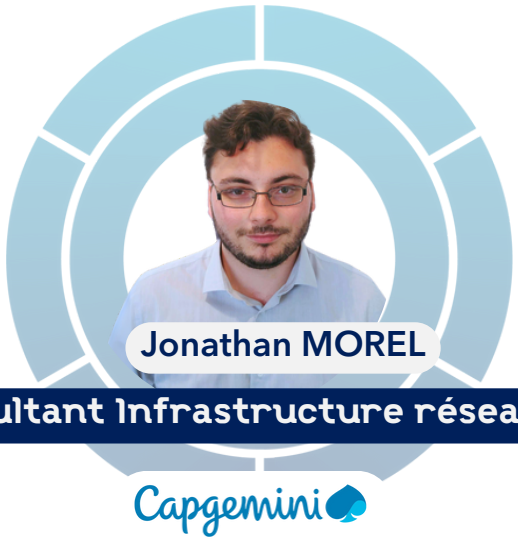
5 Speakers



Jérôme DURAND

Architecte Réseau

cisco



Jonathan MOREL

Consultant Infrastructure réseau

Capgemini

Guillaume MAULE

Responsable Cellule Automatisation

CNS

Pierre FRANÇOIS

Maître de Conférence

INSA INSTITUT NATIONAL DES SCIENCES APPLIQUÉES LYON

Yann MORTIER

Spécialiste en automatisation

Red Hat

Pourquoi intégrer la sécurité dès la conception ?

Sécuriser l'automatisation : un impératif, pas une option.

Par Jonathan Morel, Consultant Infrastructure Réseau chez Capgemini.

Automatiser, c'est gagner en efficacité… et donc en surface d'attaque. Jonathan propose des exemples concrets et des bonnes pratiques à mettre en œuvre pour intégrer la sécurité dès la conception des scripts et des pipelines.

Selon lui, le SDN est un vecteur facilitant l'automatisation, à condition qu'il soit correctement sécurisé.

Les outils SDN n'appellent qu'à être automatisés car ils présentent des API, même si elles sont plus ou moins faciles à utiliser.

L'automatisation donne par ailleurs lieu à des pistes d'optimisation.

Dans un cas client, qui consistait à l'origine dans le déploiement d'environ 120 sites du 802.1X (norme de sécurité) pour sécuriser l'accès au réseau, l'opportunité de standardiser d'autres aspects de l'outil SDN a été saisie. En exemple ? La nomenclature des tags pour identifier la fonction des ports a été homogénéisée, grâce à l'automatisation.

De plus, l'automatisation permet de faciliter les rollbacks et d'en fiabiliser le processus.

Parmi les cas présentés, celui d'un site où les prérequis au fonctionnement du 802.1X n'étaient pas remplis. Le rollback a été facilité et accéléré et le redéploiement mené après la replanification de l'opération a été aussi simple que la première itération, car toutes les actions étaient scriptées.

Jonathan met en perspective les limites de l'automatisation, et notamment de ce qui vaut le coup d'être automatisé :

➜ *Ce que le contrôleur fait déjà très bien*

Il n'est pas toujours indispensable d'automatiser le redémarrage d'un équipement, notamment quand le contrôleur SDN s'en charge déjà efficacement.

➜ *Le traitement des cas marginaux*

Le traitement des cas marginaux est à évaluer. Par exemple, s'il y a une ou deux exceptions sur un nombre important d'équipements, le temps passé au développement d'un script pourra s'avérer plus long que d'une réalisation manuelle (seulement si l'action est simple et peu sujette à erreurs).

Il met en exergue l'accès sur la sécurité de la plateforme d'automatisation.

Celle-ci est amenée à avoir accès à tout le réseau, ce qui pourra impacter fortement la sécurité en cas de compromission. Il revient sur les fondamentaux :

➜ *L'intégrité*

Via un contrôleur de version (CVS)

➜ *Le runtime*

Si la plateforme doit gérer un failover par exemple, il faut qu'elle soit disponible en permanence.

➜ *La confidentialité des secrets*

Respecter des pratiques, comme ne pas indiquer dans les scripts les tokens API, les mots de passe ou les éléments de connexion…

➜ *La traçabilité des exécutions*

Pour remonter et pouvoir identifier "qui a fait quoi" et "quand".

Intervention complète sur YouTube

Intervention de Jonathan Morel

Construisez votre stratégie gagnante !

Automatisation : Clé en main, sur-mesure, ou hybride ?

Par Jérôme Durand, Architecte Réseau chez Cisco.

La question de l'automatisation est de plus en plus présente, surtout aujourd'hui, lorsque la multiplication des applications et des équipements surcharge les équipes IT.

Vaut-il mieux choisir un produit fini ou développer soi-même son automatisation ?

Lorsqu'un client échange avec un constructeur, il doit se demander s'il est réellement approprié de choisir un "produit fini" proposé par ce constructeur, ou s'il devrait plutôt développer lui-même l'automatisation qu'il utilisera.

Jérôme précise qu'entre les deux, il n'y a pas de mauvaise solution. Chaque stratégie présente ses avantages et ses défis pour une transformation digitale progressive et sécurisée. En outre, on trouvera des profils de clients correspondant aux deux cas extrêmes.

Il présente le cas des API comme un bon moyen de conciliation. En effet, elles sont présentes sur le produit clé-en-main et sont également tout à fait intégrables dans une automatisation plus globale. Il aborde ainsi les pistes de réflexion à envisager selon le besoin type de l'entreprise.

➜ *Quel problème adresser ?*

La réflexion se porte sur le besoin réel : est-il lié à la technique ou au business, couvre-t-il un seul ou plusieurs domaines ? Comment orchestrer la problématique ?

➜ *Le marché est-il mature ?*

Si la solution existante est mûre sur un besoin, pourquoi serait-il judicieux de réaliser soi-même de l'automatisation ? Car si le matériel le permet et que la solution est éprouvée, il est plus facile de relier les besoins à des outils existants et prédéveloppés. Mais cet argument ne s'appliquera pas si les besoins ne sont pas entièrement couverts par l'outil. Dans ces situations, il deviendra finalement nécessaire de concentrer les efforts de développement sur ces cas d'usage, qui sont spécifiques.

➜ *Quelle complexité / simplicité ?*

Comment appréhender le nombre d'outils, qui lorsqu'ils se multiplient deviennent difficiles à gérer ?
Car certaines entreprises vont multiplier leurs outils afin de répondre à leurs différents usages, mais elles oublient la partie gestion associée. Jérôme prend l'exemple du SD-WAN, pour lequel le développement de contrôleurs (programmant des routeurs) a toujours été complexe pour les constructeurs.

➜ *Opérer un réseau est différent de l'automatiser*

Opérer un réseau consiste en fait à comprendre ce qui s'y passe, à le superviser et à assurer la remédiation en cas d'incident. Mais un automate, lui, ne pourra jamais accomplir pleinement l'ensemble de ces actions. Dans la logique d'automatisation, il est essentiel de garder cela à l'esprit et de continuer à travailler sur l'ensemble du réseau, qui reste tout aussi important. Par exemple, il faut comprendre ce qui se passe dans l'infrastructure – notamment grâce à l'observabilité – afin de pouvoir automatiser des actions en réaction.

.

➜ *AIOPS et data*

Ce point est une réflexion en lien avec les promesses de l'IA. Quand il s'agit de choisir une solution d'automatisation, ce choix futur doit-il porter sur une solution d'automatisation dite "classique", ou plutôt sur une solution "de data" effectuant des automatisations associées à l'intelligence artificielle. Si le second choix est retenu, il devra s'appuyer sur des données finement récoltées, afin d'alimenter les LLMs et les intelligences... d'où la réflexion cruciale sur la data.

➜ *Capacité et support*

Les équipes réseau en place seront-elles prêtes à opérer leur réseau d'une nouvelle façon ? Le support à fournir, et la gestion des problèmes qui apparaîtront, devront eux aussi être repensés. Une réflexion devra donc être menée sur ces aspects.

Pour conclure, Jérôme se penche sur un dernier point :

La différence entre orchestration et automatisation.

Assembler toutes les pièces pour répondre à un besoin – comme dans l'analogie de la maison – consistera, dans chaque domaine, à s'appuyer plutôt sur des solutions qui réalisent déjà 70 à 80 % du travail, puis à compléter les besoins encore restants grâce à l'automatisation.
En revanche, l'orchestration est beaucoup plus personnelle : en ce sens, il est difficile de proposer une orchestration clé-en-main qui réponde exactement aux besoins d'une entreprise et à la façon dont elle souhaite les mettre en œuvre.

Intervention complète sur YouTube

Intervention de Jérôme Durand

Quels rôles le SDN peut-il assumer dans une plateforme d'automation ?

La place du contrôleur SDN.

Par Guillaume Maule, Responsable Automatisation chez CNS Communications.

Pour automatiser, il faut d'abord se rappeler... ce que l'on automatise : c'est souvent une infrastructure réseau constituée de constructeurs différents. En France, le SDN est généralement rapporté à du management, notamment sur la partie du réseau gérée de manière centralisée sur un contrôleur.

Situer le contrôleur au sein d'une plateforme d'automation permet d'identifier les rôles qu'il pourra y assumer.

Dans son intervention, Guillaume présente les différents éléments présents dans une plateforme et la capacité d'un contrôleur SDN à remplir certaines des fonctions qui la composent.

➜ Le moteur d'automatisation

C'est ce moteur qui interagit avec l'infrastructure, afin de lui pousser des instructions, de collecter de la donnée - et éventuellement - de réaliser de la compilation. Ce moteur effectue simplement l'action. Pour la réaliser, il doit récupérer ailleurs les informations nécessaires.

Sujet de la prochaine Rencontre #2 →

➜ *La source d'intention*

Quant à elle, elle stocke de manière interrogeable l'état désiré du réseau, en matière d'inventaire, mais aussi de configuration : les consignes, les standards, ou encore, les exceptions.

➜ *L'état opérationnel*

C'est le côté réalité de la source d'intention. En effet, il regroupe toutes les informations du terrain, qui pourront en fait être différentes de ce qui était désiré. On parlera dans ce cas des écarts, ou encore, des "drifts" *

**Sujet traité dans la Rencontre #0*

➜ *Le dépôt*

Cette partie assure le stockage du code (les scripts), mais aussi les configurations, les rapports, etc.

➜ *L'orchestration*

Elle permet de prendre de la hauteur, ceci en gérant les droits, en pilotant les environnements d'exécution, ou encore en programmant les jobs pour tracer et superviser ce qui "est fait" dans la plateforme.

➜ *La présentation*

Elle assure l'interfaçage externe (les API, les emails, etc.) et l'interaction des utilisateurs (GUI, formulaires, etc.)

➜ *La sécurité*

Dernier point de cette liste. Elle assure les bonnes pratiques et consiste par exemple en un gestionnaire de secrets.
Le contrôleur SDN a toute sa place aujourd'hui dans chacune des briques énumérées. Ses fonctions "de base" lui permettent par exemple d'effectuer facilement des mises à jour d'équipements, ou encore de paramétrer des templates de configuration.

Le contrôleur gère aussi les droits selon les utilisateurs, pour leur permettre de créer des workflows, afin de déployer ensuite ces changements dans le réseau.

Dans le cadre de l'utilisation d'un contrôleur SDN, il est nécessaire de se poser plusieurs questions, pour :

Cerner son besoin, choisir et définir son automatisation de manière réfléchie.

➜ *La séparation entre l'intention et le réel*

En séparant intention et réel, on pourra stocker le design et l'implémentation de manière différente et spécifique. C'est crucial pour pouvoir faire de la conformité, par exemple.

➜ *Mono-vendeur / multi-vendeurs*

Elle se pose quant aux disparités au sein même des fonctions, ou des applications (LAN, WAN, Firewalling, etc.)

➜ *La dépendance*

La question de la dépendance vis-à-vis de l'investissement, c'est-à-dire du risque soulevé par le fait de "tout" miser sur un contrôleur, et sur une solution d'un constructeur. Est-ce que ce risque est (ou non) acceptable pour l'entreprise ?

En parcourant toutes ces questions, on réalise qu'il est nécessaire de déterminer une balance viable et alignée avec les objectifs et les moyens de l'entreprise et de son équipe réseau.

C'est cette approche qui permet de faire des choix de stratégie et de design d'automatisation.

Entre l'utilisation de solutions personnalisées et agnostiques, ou alors, de solutions plus génériques et propriétaires.

Intervention complète sur YouTube

Intervention de Guillaume MAULE

Les challenges liés à l'automatisation des réseaux.

Standardisation IETF, industrie et recherche.

Par Pierre François, Maître de Conférence à l'INSA Lyon.

L'intervention de Pierre s'est concentrée sur la standardisation de l'automatisation réseau à l'IETF (Internet Engineering Task Force), une organisation de normalisation pour les standards de l'internet. Il propose un débat sur les problèmes posés aux opérateurs réseau, aux vendeurs d'équipements et aux acteurs de la standardisation à l'IETF.

Ce projet, financé par SPIE, Huawei et Swisscom, vise à réaliser de la détection automatique de disruption de service. Un déploiement est actuellement en cours chez Swisscom (télécommunications), sur leurs services VPN notamment.

Pierre nous expose comment la standardisation est approchée à l'IETF, alors que d'autres standards existent déjà sur le marché, tels que NetConf, Yang, OpenConfig, des modèles propriétaires par exemple.

À l'IETF, il existe des processus bien plus longs avant de publier un standard.

Mais, ce standard aura l'avantage d'être robuste une fois sur le marché, car il est soumis à la critique des différents membres en interne et suscite des débats lors de sa phase de conception.

L'un des plus gros défis pour définir un standard est de trouver le plus grand dénominateur commun entre les différents constructeurs (Huawei, HPE, Cisco, Nokia, etc.). Une étape difficile car chaque constructeur présente une implémentation singulière. Le but est d'être spécifique et de se différencier sur le marché. Dans la réalité de marché que nous connaissons, la solution réside finalement dans la mise en place de correspondances entre un standard, et les implémentations des constructeurs pour permettre un langage commun entre tous les acteurs.

À ce niveau, le SDN devient finalement une simple question de terminologie. Il revient alors à du management de réseau, comme évoqué précédemment, étant donné que tous les acteurs opèrent sur des devices finaux.

L'information collectée n'est alors pas standard. En effet, chaque constructeur jouit de sa propre implémentation, qui est pensée pour son écosystème.

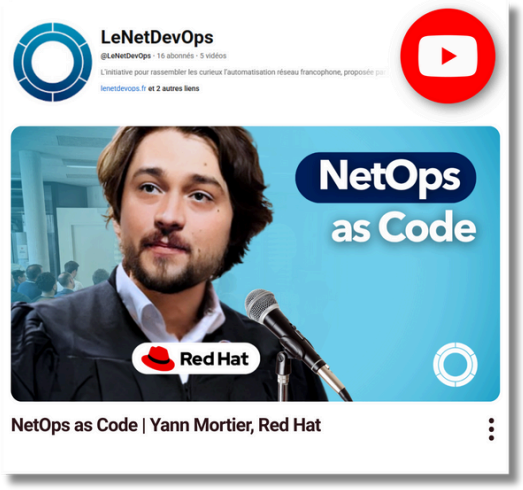
Intervention complète sur YouTube

Intervention de Pierre François

Unifier et automatiser la gestion d'infrastructure.

NetOps As Code

Par Yann Mortier, Spécialiste automatisation chez Red Hat.



La question est de savoir comment unifier et automatiser la gestion d'infrastructures réseau qui sont hétérogènes, pour réussir à innover, simplifier les opérations et gagner en efficacité ?

Yann précise qu'il existe trois grandes couches de complexité, liées à l'automatisation des réseaux.

➜ *Première couche*

La forte volumétrie d'équipements et de constructeurs (différents !) pour un même type de service

➜ *Deuxième couche*

Elle accompagne la précédente, l'arrivée massive des contrôleurs qui entendent contrôler le réseau pour gérer les différents plans de son parc, multiplie en fait les interfaces... ce qui entraîne une perte de productivité et d'efficacité opérationnelle.

➜ *La troisième couche*

Elle concerne le vieillissement et l'obsolescence des équipements en place. La plupart d'entre eux manquent de vecteurs qui facilitent l'automatisation.

Face à ces couches de complexité, l'enjeu est de réussir à retrouver un excellent niveau de sécurité et d'efficacité opérationnelle, en standardisant et en industrialisant la gestion de son parc.

Yann recommande en ce sens une approche NetOps.

Celle-ci permet de déclencher des workflows via du code. Celui-ci est stocké dans un dépôt Git (GitLab, GitHub, etc.). Ansible Automation Platform (AAP) permet d'orchestrer via différents déclencheurs, parmi lesquelles sont envisageables des actions manuelles ou encore, des actions planifiées, par exemple.

En employant cette démarche, Yann précise qu'un outil SDN sera interrogeable par Ansible.

Il mobilise l'exemple d'un cas client, pour mettre en lumière le fait que l'automatisation permet de gagner du temps en industrialisant et en normalisant le réseau, tout en fiabilisant les déploiements et la maintenance.

Le but est de permettre de bénéficier des points positifs des automatisations proposées par le constructeur.

(Telles que le templating, les opérations de masse, etc.).

Cette démarche offre un seul point d'entrée sur le réseau, ce qui facilitera la communication entre l'orchestrateur et le reste des équipements.

Intervention complète sur YouTube

Intervention de Yann Mortier

04. Analyse en 3 points

Point 1

Les automatismes intégrés au SDN : efficacité et simplicité.

Les différentes solutions SDN intègrent nativement des fonctions d'automatisation, prêtes à l'emploi.

Comptons parmi elles l'exécution de changements de configuration, le déploiement de politiques réseau, les mises à jour logicielles ou encore la collecte d'informations en temps réel.

Ces fonctions visent à industrialiser certaines opérations, tout en réduisant le risque d'erreur humaine.

➜ *Par exemple, Cisco Meraki Dashboard permet de déployer simultanément des VLANs, des ACL ou des politiques de QoS sur des centaines d'équipements.*

➜ *De même, Aruba Central ou encore FortiManager offre la possibilité d'orchestrer automatiquement les mises à jour de Firmwares, ou la propagation de règles de sécurité sur l'ensemble d'un parc.*

Cette intégration native présente un avantage majeur : la rapidité de mise en œuvre.

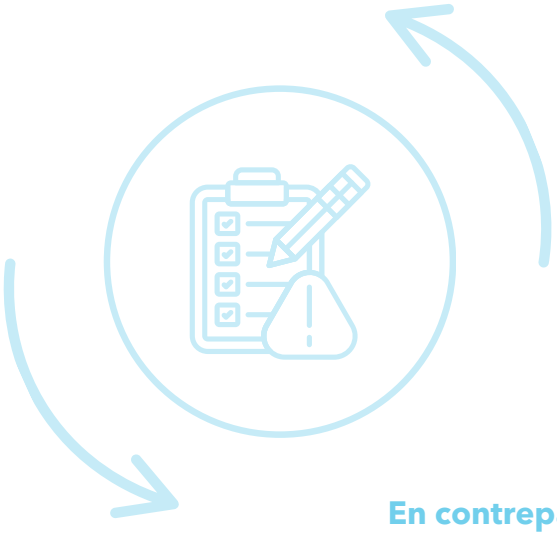
Le constructeur assure lui-même la compatibilité entre le matériel, le firmware et le contrôleur, ce qui simplifie grandement le maintien en conditions opérationnelles.

Les administrateurs bénéficient d'une interface prête à l'emploi pour superviser et pour piloter l'ensemble du périmètre rattaché à ce contrôleur, avec des fonctions de monitoring et d'alerting intégrées.

Le contrôleur SDN connaît également les particularités de ses équipements – comme les protocoles propriétaires, les capacités spécifiques, les particularités ou les contraintes d'implémentation.

Cela permet d'éviter les incompatibilités ou encore les erreurs de configuration,

Que l'on pourrait rencontrer avec des outils génériques. Par exemple, nous pourrions citer le contrôleur Arista CloudVision, qui permet de gérer de façon automatique la synchronisation des configurations EVPN / VXLAN. C'est une tâche qui serait complexe, sujette à erreurs humaines, si elle devait être scriptée manuellement.



En contrepartie, cette approche lie étroitement l'administrateur dans l'écosystème du fournisseur.

La dépendance à des API propriétaires, à des formats de configurations "non standards", ou à des contrôleurs fermés (plateforme propriétaire, fournie par un éditeur ou un constructeur), peut limiter la portabilité et la souplesse souhaitée des processus d'automatisation... surtout dans des environnements multi-constructeurs.

Point 2

Automatiser en-dehors du SDN : flexibilité et indépendance

Une automatisation indépendante des contrôleurs SDN permet de garder le contrôle sur la logique d'exploitation tout en assurant une cohérence dans des environnements multi-constructeurs ou hybrides.

Des outils comme Ansible, Terraform, ou encore Python (via HTTPs / API REST ou SSH / CLI historiques) permettent de piloter simultanément plusieurs domaines technologiques : un Catalyst Center pour le LAN, un FortiManager pour le WAN, un Panorama pour les pare-feux, ou directement des équipements non-SDN.

Là où les interfaces graphiques centralisées sont intuitives mais peu adaptées aux déploiements massifs ou à la gestion des versions de configuration, intervient l'approche Infrastructure as Code (IaC). Elle permet d'automatiser ces actions de manière reproductible, traçable et versionnée, à l'image d'un code logiciel.

Par exemple, un playbook Ansible peut déployer des configurations VLAN cohérentes sur plusieurs sites hétérogènes, ou un script Terraform peut gérer la création synchronisée d'instances réseau dans différents environnements cloud et on-premise.

Répondre à la limite du cliquodrome

Cette approche répond aussi à une limite fréquente du modèle SDN : le "cliquodrome".

L'enjeu de la source d'intention.

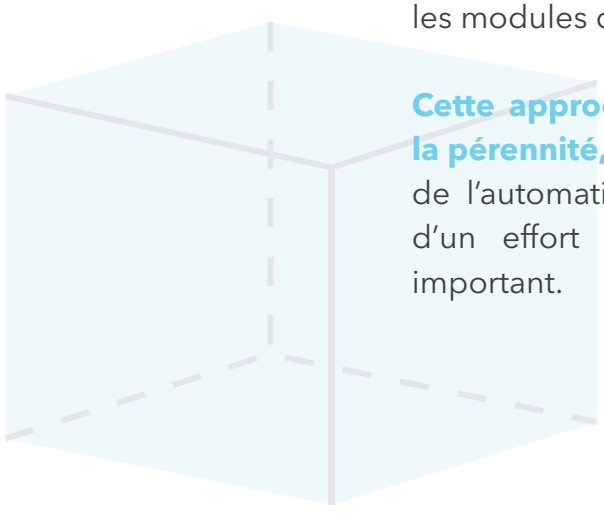
Un autre enjeu réside dans la notion de source d'intention ("source de vérité").

Lorsqu'un contrôleur SDN devient le seul point de déclaration des politiques réseau, il devient difficile de vérifier la conformité entre l'intention (design) et la réalité (implémentation).

En externalisant cette source d'intention (via un logiciel spécialisé, un dépôt Git, une CMDB, etc.), on bénéficie d'une vision transverse, capable de vérifier la cohérence entre plusieurs domaines technologiques.

Meilleure pérennité de l'automatisation.

Enfin, cette indépendance apporte une meilleure pérennité : si l'automatisation est construite autour de standards ouverts (API REST, NetConf, YANG, etc.), le changement de fournisseur ou de technologie n'impose pas de repartir de zéro – il suffit d'adapter les modules d'intégration.



Cette approche favorise à la fois la pérennité, et la transversalité de l'automatisation réseau, au prix d'un effort qui sera certes plus important.

Point 3

Complémentarité : un équilibre à trouver.

Chaque périmètre technologique demande de trouver le bon équilibre, entre l'utilisation des automatisations natives du SDN, et le développement de mécanismes externes.

Ce choix est similaire - entre écrire un script Python sur-mesure et utiliser un module Ansible, ou un provider Terraform maintenu par le constructeur.

Tout dépend du degré de personnalisation, de contrôle et de maintenance souhaité.

Dans un environnement SDWAN Velocloud par exemple, il peut être pertinent de s'appuyer sur les templates de configuration proposés par le contrôleur pour déployer les politiques de répartition et priorisation des flux applicatifs.

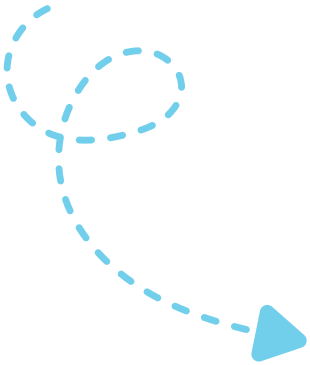
Mais cela n'empêche pas de gérer la configuration de ces templates et leur association avec les équipements, au travers de playbooks Ansible.

Le SDN agit alors comme un catalyseur de l'automatisation.

Il fournit une couche d'abstraction centralisée et programmable, offrant des API et des modèles de données exploitables.

L'automatisation elle, fournit les processus et outils

Ces derniers peuvent exploiter ces capacités à grande échelle, en y incluant des principes issus du développement logiciel (CI/CD, GitOps, versioning, ou encore des tests automatisés).



En d'autres termes : le SDN est un outil permettant à la démarche d'automatisation de gagner en efficacité et maintenabilité.

Pour finir

Le SDN et l'automatisation réseau ne s'excluent pas.

Mais ils répondent à des besoins complémentaires.

Le SDN transforme la structure du réseau. Il apporte en ce sens de la programmabilité, mais aussi de la centralisation.

L'automatisation optimise quant à elle l'exploitation du réseau et favorise la transversalité, l'ouverture et la standardisation des pratiques.

Le premier s'inscrit dans une logique verticale et packagée, spécifique à chaque constructeur...

...quand la seconde adopte une approche horizontale et plus personnalisable, ce qui permet de couvrir l'ensemble de l'écosystème réseau.

Dans la pratique, les organisations tendent à combiner ces deux approches.

Ceci en exploitant les capacités d'abstraction offertes par les contrôleurs SDN, tout en s'appuyant sur des outils d'automatisation ouverts pour orchestrer et piloter l'ensemble du réseau.

SDN et automatisation participent à une même transformation

Celle d'un réseau devenu programmable, adaptable et intelligent, où la maîtrise du logiciel est désormais au cœur de la performance et de la résilience des infrastructures.

Remerciements

L'équipe LeNetDevOps remercie l'ensemble des intervenants pour leur temps et leur expertise, ainsi que l'ensemble des participants pour leur mobilisation et la qualité des débats. Ensemble, nous poursuivons la dynamique de notre communauté francophone.

Participez

Aux prochaines Rencontres

Événement gratuit, places limitées.

Suivez le lien, et remplissez le formulaire.

Prenez part à la communauté, ouverte à tous, rejoignez le Slack

lenetdevops.fr


lenetdevops@cns-com.com


lenetdevops.fr


youtube.com/@LeNetDevOps


linkedin.com/company/lenetdevops