

# Junos OSに時代が追いついてきた、これはクラウド時代のネットワークエンジニアのための、新たな必修科目だ

「クラウド時代にネットワークエンジニアは要らなくなるのでは」といわれることがある。真実は、複雑なコマンド体系を持つネットワーク OS をマスターしたことで、自己満足に陥っているネットワークエンジニアが、時代遅れになるかもしれないということだ。だが、自分が時代遅れにならないための選択肢は、まだある。

2015 年末には、ハードウェアとの分離提供も発表された、ジュニパーネットワークスのネットワーク OS である「Junos」。ジュニパーのネットワーク製品を採用するユーザーには、主な採用理由の一つとして Junos の存在を指摘する人が多い。

例えば GMO クリック証券の箱崎和男氏は、「Junos OS の操作は初めてですが、とても使いやすいですね」と話している。また、ミクシィの松本聡氏は、「他社製品に比べて、コンフィグの管理が格段にやりやすいと感じています。コマンドの入力、つまり作業内容がコンフィグの変更内容と一致する仕様は、バージョン管理システムとの相性が良いです。コミット方式が採られているため、設定コマンドの入力順序をあまり意識しなくてもよいですし、コミット前にコンフィグの差分をチェックできるところも気に入っています」と語っている。大塚商会の倉橋孝典氏は、オペレーションが統一でき、自動化が進めやすい点を高く評価している。

これら 3 社だけではない。「Junos OS を一度体験してしまうと、元に戻れない」という人たちが後を絶たない。そこまで言わせる Junos とは、いったいどんなネットワーク OS なのか。以下ではその魅力に迫る。

## JunosのCLIは、IOSライクなCLIと根本的に違う

Software Defined Networking (SDN) が登場したころ、これを推進する人々が最初に攻撃したのは、ネットワーク機器を CLI で設定することの不毛さだ。だがこれは、IOS ライクなコマンド体系を持つネットワーク OS には当てはまるものの、Junos には当てはまらない。Junos の CLI が「不毛」ではない理由は、以下の通りだ。

### 「コマンドライン群→commit」という操作体系

IOS ライクな CLI では、ファイルシステムのディレクトリを変更するのに似て、コマンド階層をたどり、コマンドを一つずつ投入しては抜けて、次のコマンド投入へ移る。そして、コマンドを一つ投入するたびに、これが有効になり、ネットワーク機器の動作に反映されてしまう。

一方 Junos では、複数のコンフィグを並べるように次々に記述し、その後に commit コマンドで、これらをまとめて投入するという手順だ。「Junos におけるコンフィグとは、設定ファイルを適用する作業」という表現もできる。この説明だけでも、作業効率にどれだけの違いがあるかは、容易にご理解いただけるはずだ。

Junos では、commit にいくつかのオプションがあり、作業ミスや時間の無駄を防げるようになっている。commit の前に、commit

### IOSライクなこれまでのネットワークOS

コンフィグ投入	コンフィグの削除
(router)# AAA	(router)# AAA
(router-AAA)#option BBB	(router-AAA)#option BBB
(router-AAA-BBB)#sub-option abc	(router-AAA-BBB)#no sub-option abc
(router-AAA-BBB)#exit	(router-AAA-BBB)#no option BBB
(router-AAA)#option CCC	(router-AAA)#option CCC
(router-AAA-CCC)#sub-option abc	(router-AAA-CCC)#no sub-option abc
(router-AAA-CCC)#exit	(router-AAA-CCC)#no option CCC
(router-AAA)#exit	(router-AAA)#no AAA
(router)#	(router)#

IOSライクなネットワークOSでは、コマンドを一つずつ投入し、コマンド階層を抜けてから次のコマンド投入に移る。コマンドを投入することにより、これがネットワーク機器の動作に反映されてしまうため、十分な検証が必要。入力したコンフィグを削除するには、入力時と同じ順序で一つずつ削除する必要がある

### Junos

コンフィグ投入	コンフィグの削除
(router)#set AAA Option BBB abc	(router)#rollback 1
(router)#set AAA Option CCC abc	(router)#commit check
(router)#commit check	configuration check succeeds
(router)#commit	(router)#commit

Junosでは、入力コマンドをまとめて記述したのち、「commit」によってこれを投入する。これにより初めてネットワーク機器の動作に反映される。commit前に、「configuration check」で文法チェックもできる。直前に投入したコンフィグの削除は、「rollback 1」と入力するだけで実行できる

check とすれば、文法が正しいかがチェックされる。これを確認してから commit コマンドで設定を反映させればよい。show | compare コマンドを使うと、自分が投入しようとしているコンフィグと、現時点で動作しているコンフィグの差分が表示される。これを見て設定内容に誤りがないかどうかを、投入の前に確認できる。

また、commit の代わりに commit confirmed とすると、設定が一定時間（デフォルトでは 10 分）だけ有効になり、その後自動的にロールバックされる。これにより、複雑な設定であっても「試しに」投入し、正常に動作するかどうかを実地で確認することができる。リモート設定時にパケットフィルタ設定を誤り、自身のアクセスがブロックされて困るといった事態も防げる。

### いったん投入した設定の削除は rollback 一発で済む

IOS ライクな CLI では、特にいったん投入したコンフィグを削除しようとしたときに、その難しさを痛切に感じた経験を持つ人が多い。正しくコマンドのナビゲーションをしながら、投入した順序と寸分たがわず間違えないように、一つずつ削除していかなければならない。この作業は時間的にも、精神的にも負担が大きい。何度もリハーサルを繰り返し、何時間、何十時間も掛けて、ようやく作業を終えるといったケースは稀ではない。

一方、Junos では、削除作業が rollback 1 というコマンド 1 つで

終わってしまう。この差はあまりにも大きい。この場合、1 というのは一つ前の世代のこと。Junos では、50 世代を保持できる。従って、49 回のロールバックが実行できることになる。こうしたバージョン管理の仕組みを搭載していることは、作業の正確性と作業負荷の軽減に大きく寄与する。他社の間でも、Junos のロールバック機能を真似る動きが見られるが、一世代のみの対応にとどまるなど、機能が限定的である場合がほとんどだ。

ネットワーク運用者は、コンフィグの投入や、その切り戻しといった作業自体に、時間や気力を奪われるべきではない。例えば正確を期すためにネットワーク設定で時間が食われ、新サービスの提供開始が遅れたとしたら、このサービスのオーナーはどう思うだろうか。これはもはや、ネットワーク運用担当者が我慢してやっていたらいいといった問題ではない。

## 自動化の推進で、さらにスピードアップ

Junos では、さらに「JUNOScript」というスクリプティング環境が用意されている。これを活用して、運用の自動化を積極的に進めることができる。多様な側面からの自動化が容易に実現できる点が、非常に興味深い。

JUNOScript は、システムが起動するものと、運用者が起動するものの 2 つに大別できる。

まず、システムが起動するスクリプト（「Event Policy」「Event Script」と呼ばれる）については、スクリプトに指定した日時、あるいは何らかのイベントをトリガーとして、起動される。これを使って、例えば特定時刻におけるファイアウォールのセッション数と機器の CPU 使用率を自動的に取得し、外部の FTP サーバへアップロードするプロセスを自動化できる。また、Router1 経由の外部接続に障害が発生した場合、Router2 をデフォルトゲートウェイとするよう自動的に設定を変更するといったことができる。

運用担当者が起動するスクリプトは、commit コマンド発行時に自動実行される「Commit Script」と、担当者がスクリプト名を入力することで起動される「Op Script」がある。

Commit Script では、commit 時に、あらかじめマクロとして記述した補完的な設定を自動適用できる。また、ユーザー組織独自のポリシーに基づき、警告メッセージを表示して commit を中止する、あるいはコンフィグに自動修正を加えるといったことができる。一方、Op Script では、CLI にスクリプト名を入力するだけで、例えばトラブル時に必要となる情報を、一括して取得し、必要な情報のみを抽出して、外部サーバに自動保存するなどができる。優秀なエンジニアの知見を、スクリプトという形で表現しておけば、運用の属人化をいくらかでも軽減できる。

## 「スクリプティングなど今さらイヤ」でも大丈夫

JUNOScript では、上記のように豊富な自動化が実現可能だが、スクリプティングスキルが要求されることもたしかだ。だが心配しないでほしい。「スクリプティングはしきいが高い」というネットワーク

運用担当者でも、Junos コマンドを知ってさえいれば使える自動化機能として、「Event Option」が用意されている。

Event Options は、自動実行されるべきアクションを、Junos コマンドで記述できる機能。Junos デバイスに対して直接設定するので、外部ツールなどとの連携を考慮することなく、活用できる。

イベントあるいはタイマーに基づいて実行されるアクションとしては、各種コマンドの他、コンフィグの書き換え、ファイルのアップロード、トラップの送付などがある。JUNOScript を起動することもでき、高度な条件分岐を実現したい場合などにも応用可能だ。

例えば、障害発生時の Syslog メッセージに応じて、ネットワーク機器の設定を自動的に変更するようにあらかじめ用意しておくことができる。

## Junosはすぐに学べて明日からでもタダで使える

これまで、IOS ライクなネットワーク OS に対する Junos の利点を、多数紹介してきた。それでも、懸命に習得した IOS の世界を捨てたくないという人はいるだろう。ただ、ここには IOS とは大きく異なる、新しい世界が開けていることを知ってほしい。

ジュニパーネットワークスは通信事業者向けルータの開発・提供からスタートした企業であり、通信事業者の世界では、Junos と IOS が両方使えて一人前とされている。その後ジュニパーは、データセンタースイッチ、キャンパスネットワークスイッチ、セキュリティ機器など、ネットワーク製品の総合ベンダとなった。これら全ての製品が、Junos という単一の OS で動いている。

昔から、Junos の使いやすさは、使った人には知られていたが、「良さそうだけど、製品が高価すぎて触れない」という声も多かった。だが、今は違う。Junos を搭載した安価なスイッチ製品が登場している。

さらにすごいのは、無料で触りまくれる環境が整っていることだ。これは「vSRX」というもの。次世代ファイアウォール、侵入検知・防御、アンチウイルス、アンチスパム、Web フィルタリングなど豊富な機能を持つ総合セキュリティルータである「SRX」の仮想アプライアンス版で、本番利用でなければ、全く制限なしに、無償で使える。ジュニパー製品のユーザーにも、vSRX で新たな設定を試すという人はたくさんいる。

繰り返しになるが、この製品の OS は、ジュニパーの他のネットワーク製品と、完全に同一だ。Junos の上に、各種のセキュリティ機能がデーモンとして動いている。

vSRX で Junos の世界を体験してみれば、この OS はネットワークエンジニアの思考回路に寄り添う操作体系を持っていて、CLI の習得もある程度自然にでき、思っていたよりも楽だと分かるだろう。

そして何よりも、現在のネットワーク機器で、設定変更の準備と検証に、膨大な時間を掛けている人なら、「これがネットワークエンジニアの宿命」とあきらめる必要はないのだということに気付くはずだ。クラウド時代における、ネットワークのイノベーターを目指すのは、今からでも遅くない。