

SR. EDUARDO BARASAL MORALES: Bom dia a todos. Sejam todos bem-vindos a mais uma live do Intra Rede. Então, essa live é a primeira do ano de 2021. A gente tem ali o assunto Programabilidade das Redes, você tem que ir muito além da CLI para não ficar obsoleto. Que é um tema que a gente quer trazer tanto para as pequenas redes, para as médias redes, para as grandes redes.

Para as pequenas, se você tem ali poucos elementos de rede, ainda você, talvez, consiga configurar tudo na mão. Mas, pensando em expansão, você já tem que começar a ver alguns processos de automação e melhoria aí do seu gerenciamento da sua rede.

Para aquelas médias redes, já devem ter ali uma equipe um pouquinho mais robusta e ter ali um pouquinho mais de equipamentos. Ainda deve estar dando para trabalhar tudo na mão, nas CLIs, mas... lembrando: vai crescer um dia, também tem que pensar em como crescer de maneira ordenada, de maneira fácil e que você consiga incluir novos participantes.

E aquelas grandes redes. Se por acaso você não faz nenhuma, ainda, automação muito provavelmente você está com uma equipe gigantesca e está tendo ali muito trabalho, e às vezes ali muitos errinhos devido ali a erro humano de algum administrador que digitou errado ou fez alguma informação errada.

Então essa é uma live que a gente tenta aí abranger todas as redes: as pequenas, as médias e as grandes. E, para isso, a gente trouxe vários especialistas da área, como o José Miguel Guzmán, da Whitestack, o Helio Barroso, da Cisco, Elizandro Pacheco, da NextHop Solutions, o Renato Almeida de Oliveira, do CITEx, né, o Exército brasileiro, o Leopoldo Mauricio, da Globo, e o Fabio Nunes, do NIC.br.

Mas antes da gente começar, eu queria fazer os agradecimentos aos patrocinadores. Então, a gente tem a Juni Link IP e Cloud Network by Giovaneli consultoria, WZTECH Networks, ICANN, Netfinders Brasil, Novatec editora, Eletronet, Globenet Telecom, Mundivox, 4Linux, Solintel, e o apoio de mídia da Revista RTI e Infra News Telecom.

Mais uma coisinha antes de a gente começar, são uns pequenos avisos, que eu acho que é de interesse do público. A gente vai ter dois sorteios, tá? Os sorteios da 4Linux, que vai sortear um curso à escolha do ganhador. Então, a gente está colando agora no chat do YouTube o link para você se inscrever. Então, no final da live, a gente deve dizer quem é o sorteado. E temos também o sorteio da Netfinders Brasil, que vai dar um curso de BGP Multivendor, no modo gravado, também, para o ganhador.

Então, são dois sorteios, os dois links devem estar colados agora no chat, e quem quiser pode se inscrever. Também teremos certificado desta live. Então, se você quiser um certificado de participação, de que você participou, de que você assistiu, a gente tem as inscrições até as

14 horas. Está sendo colocado um link aí no YouTube. E nesse link vai ser enviado um e-mail depois que você se inscrever. E aí você precisa confirmar que você estava assistindo. Confirmou que estava assistindo, depois lá das 2 horas, vai ser gerado o certificado, e você vai poder ali, depois, compartilhar com os seus colegas, colocar no LinkedIn, colocar nas suas redes sociais. Temos agora um vídeo do Cidadão na Rede, que é um projeto nosso que a gente tenta ali transmitir alguns conhecimentos de cidadania digital. Então, a gente vai rodar o videozinho agora.

[exibição de vídeo]

SR. EDUARDO BARASAL MORALES: Moreiras, quer continuar?

SR. ANTONIO MARCOS MOREIRAS: Sim, Eduardo. Muito bom dia, gente, novamente. Eu sou Antonio Moreiras, do NIC.br, para quem não me conhece ainda. E sejam todos muito, muito bem-vindos a essa nossa primeira live Intra Rede de 2021.

O Eduardo já fez uma introdução sobre o tema, e a gente tem expectativa, assim, de que essa live seja excelente. É um tema muito, muito importante. Não dá mais para o profissional de rede ficar desatualizado, saber lá só os comandinhos do Cisco, do Juniper ou usar o *inbox* sozinho, lá no Mikrotik, e fazer as coisas todas na mão. Isso não é escalável, isso é algo que já está obsoleto.

Então, bom, essa é a minha opinião, né? A gente trouxe um monte de especialistas na área que sabem bem mais do que eu sobre isso daí. E vamos ver se eles vão concordar comigo e se vão dar elementos para que vocês aí do outro lado da tela também se convençam disso e integrem esses processos de NetDevOps, de automação nas redes de vocês. Eu acho que vai ser um negócio muito legal, vocês vão começar a fazer essas integrações, automatizar a rede e vão dormir mais tranquilo, ter finais de semanas mais tranquilos e vai ser muito bom.

Esse começo da live é sempre um pouco, assim, atabalhado, um pouco bagunçado, a gente fica com medo de esquecer os avisos, de esquecer as coisas. Mas tem um que eu nunca esqueço aqui, que é o seguinte: a gente tem, por enquanto, umas trezentas e poucas pessoas aí acompanhando a live, e... Deixa eu dar uma olhada aqui como estão as coisas. Não, ó, 450 pessoas assistindo agora, muito bem! Então, primeira coisa, pessoal, deem aí, manda o link naquele grupo de WhatsApp, naquele grupo de Telegram dos colegas de trabalho, do pessoal de redes aí, lá no Facebook, no grupo do Facebook, a gente está transmitindo pelo YouTube e pela página do NIC.br no Facebook. Então, pode mandar um link ou outro link, tanto faz para o pessoal entrar, né? Tem um pessoal ainda que está meio dormindo aí, apesar da gente começar a live tarde para dar tempo de

todo mundo acordar, e no meio da semana, para dar tempo, que todo mundo já pegou no tranco ali, já está bem, mas eu acho que ainda tem gente que não está ligada que começou.

E a outra coisa: 450 pessoas, assistindo, eu tenho certeza que a maior parte desses 450 já acompanhou as outras lives nossas, então, por favor, aquele joinha logo agora no começo. Se você não gostar, depois da live, você tira o joinha, não tem problema, mas é importante isso para a gente desde já, porque ajuda a distribuição do conteúdo pelo YouTube. Então, o YouTube vai mostrar para mais gente. Mesma coisa no Facebook, ele vai mostrar para mais gente que está inscrita lá na página, inscrita no canal, o conteúdo, se você der o joinha, certo?

Então, sem mais delongas, vamos entrar no assunto. Vamos passar a palavra para os especialistas. E eu vou começar chamando o José Miguel Guzmán, da Whitestack. Então, José Miguel, no mundo--

SR. JOSÉ MIGUEL GUZMÁN: Olá, bom dia.

SR. ANTONIO MARCOS MOREIRAS: Bom dia. No mundo do software, do sistema, José Miguel, a gente fala daquele processo de DevOps, onde a gente tem uma integração entre o desenvolvimento de software, os testes, o *deployment*, para produção, como você vê os profissionais da rede, Internet, telecomunicações, tem um processo similar a esse DevOps dos sistemas para as redes? A automação e a programação entram nesse processo? Então, eu gostaria de chamar e pedir para, se você puder, falar um pouco como isso funciona, as principais tecnologias, ferramentas envolvidas. Então, a palavra é sua, José Miguel.

SR. JOSÉ MIGUEL GUZMÁN: Perfeito, Antonio, muito obrigado pelo convite. Bom, a pergunta é bem interessante. Qual a minha visão? O que está acontecendo hoje no mundo do telecom, que é onde as organizações têm ainda muitos processos que são manuais, é mesma coisa que aconteceu faz 15 anos, 20 anos na indústria do software. A única razão porque a indústria do software conseguiu um sucesso importante, ter uma agilidade e entregar funcionalidade muito rápida é porque eles tiveram uma mudança nas suas metodologias. Então, a indústria de telecom está mudando de um modo onde o mais importante era a confiabilidade, mais que agilidade, ao modelo onde agora a agilidade é importante porque, já, a confiabilidade é uma situação que já está, de alguma forma, absorvida. Então, a agilidade é o desafio agora. E o convite para os profissionais de telecom é de adotar aquelas metodologias que têm demonstrado ter sucesso no mundo do software. E que não significa todas as metodologias, tem algumas delas. Então, a gente, de alguma forma, chama tudo isso de Network DevOps, que é o que eu quisesses(F) falar agora.

Eu não vou falar muito de ferramentas, tem muitas ferramentas. Porque vou deixar os outros colegas falarem mais das coisas específicas. Mas em relação ao desafio organizacional, o desafio mais

para as pessoas, eu diria o seguinte. Não sei se vocês conseguem ver minha tela, não?

Vou assumir que sim. Então, qual é o grande desafio que têm as redes hoje? Primeiro, Opex elevado, o custo de operação elevado. Elas têm também um custo de investimento que é elevado. Elas têm, claramente, muito pouca agilidade. E você ainda tem, quer fazer muitas operações na rede de forma manual [ininteligível]. Nesta época, é muito estranho, porque no mundo do software, você tem tudo automatizado. Mas a rede continua sendo muito manual.

Você tem uma metodologia de provisionamento que também é manual, botar serviços na rede é uma coisa que toma vários dias, não é uma coisa instantânea, falta de agilidade. Você tem também uma situação que têm muitos equipamentos que são específicos, você tem caixa para *switching*(F), você tem caixa para *routing*, você tem caixa para seguridade, para segurança. E para cada uma delas tem um especialista diferente, tem comandos diferentes e metodologias diferentes. A metodologia para o Firewall é diferente da metodologia para um switch ou para um roteador.

E, ao final, o que você tem hoje é um mundo muito complexo. Não esqueça que já temos quase 9 mil RFCs, que são todas normativas dos padrões técnicos que são utilizados nas redes. Então, aí a gente tem um desafio importante.

Então, o que a gente precisa fazer para atingir um ponto de eficiência onde a gente tem... onde ir ao mesmo passo, à mesma velocidade que a indústria de software, que a indústria do Cloud? O que a gente precisa fazer? Primeira coisa, temos que vencer as dificuldades dentro do Data Center. O Data Center, se você olha aqui a Google, a Facebook, as grandes fornecedoras de conteúdo estão fazendo, é tudo automatizado, muito rápido. Você tem que reduzir a dependência dos fabricantes e isso é [ininteligível] agilidade, não é uma questão de gostar ou não gostar de uma marca. É uma coisa de agilidade. Eu posso pegar o melhor fornecedor em um determinado momento. Hoje isso é difícil.

Você tem também uma pressão que é muito importante dos OTTs, talvez o seu chefe fala para você: "Como é que a Netflix consegue? Como é que a Amazon consegue? A Google consegue, e você não consegue fazer?". Claro, eles estão usando outras metodologias, que são diferentes das metodologias da indústria de telecom, que procura confiabilidade e não agilidade. A Google, o Facebook procura agilidade.

Tem também uma situação onde estamos vendo muito pouca inovação nos serviços, a rede e os serviços de rede são mais ou menos estáveis, não tem muita diferença em termos de serviço, isso também é uma consequência da pouca agilidade, porque se a gente tivesse mais agilidade, você poderia liberar mais serviços. Temos uns desafios

de espaço e energia, que é bem claro. Está cada vez mais difícil colocar, botar mais hardware no Data Center. E finalmente, a gente, todavia, uma utilização de enlaces que é muito ineficiente. Infelizmente, o protocolo IP tem algumas limitações. Uma delas é que você só envia tráfego pela melhor rota.

Bom, então, aí aparece DevOps. DevOps é uma metodologia que foi criada na indústria do software e que procura um ciclo permanente e constante de inovação. Botar serviços novos na rede não é um projeto, é um processo. Está acontecendo sempre. E essa é a grande diferença. A gente não se prepara para fazer implantação de novos serviços. A gente está sempre fazendo implantação de novos serviços. É uma coisa rotineira. Então, para fazer isso, obviamente, não é fácil e merece uma mudança de paradigma.

O que aconteceu na indústria do software? Na indústria do software foi criado esse conceito de Network DevOps, porque a gente precisava de uma coisa que estivesse na intersecção do desenvolvimento do software, do *quality assurance* e da facilidade de operação. Para quê? Para acelerar o processo de desenvolvimento de aplicativos, aumentar a frequência de liberação. Eu posso liberar software, por exemplo, todos os meses. Eu não preciso esperar muitas *features* prontas, eu libero todos os meses. Os lançamentos, então, são mais estáveis e muito mais alinhado com os nossos objetivos, da empresa. Cada vez que a empresa tem uma condição de mercado, libera software que acompanha nesse desafio de mercado.

Isso no mundo de software. Como é que a gente mapeia a mesma coisa no mundo de telecom, mundo de redes? Bom, seria uma técnica, o Network DevOps, seria uma técnica na intersecção da criação de serviços, da *quality assurance* e das operações, simplicidade de operação. Se a gente consegue uma coisa, uma tecnologia que me permita fazer liberação rápida de serviços, mantendo a qualidade, estabilidade da rede e com uma facilidade de operação, já temos agilidade. Então, temos melhora do *time-to-market*, temos serviços mais estáveis. E alinhado também com o objetivo da operadora, um novo serviço, uma nova região da rede, uma... agora suporte para Gpon(F), para 5G, tudo isso muito mais fácil de conseguir.

Como é que a indústria do software conseguiu fazer isso? Com duas coisas: automação, ferramenta de software, não é com pessoas, é com ferramentas de software, scripts, e com monitoramento. A Google fala: "Você não pode melhorar se não monitorar, se não tem monitoramento". A única forma de melhorar é ter números, data dura. E hoje ainda é um desafio monitorar. Muitas vezes acontece, você tem uma janela de manutenção, faz todo um trabalho complexo, termina a janela de manutenção, faz teste dos serviços e detecta que tem um serviço que não está funcionando, e tem que fazer *rollback*. E depois você descobre que o serviço não estava funcionando antes da janela e

que o problema não tinha nenhuma relação com o que você fez. Bom, esse tipo de problemas é porque você não tem monitoramento extremo a extremo, ponta a ponta. Você tem que ter isso como parte do processo constante. E isso vai um pouquinho mais longe do que somente ter monitoramento de SNMP, tem telemetria e algumas outras coisas que fornecem muito mais informação.

Então, as vantagens do Network DevOps: desde a nossa perspectiva, tem um benefício que é cultural, o time de operação é muito mais produtivo, muito mais feliz, porque tem menos problemas, eles operam, eles conseguem manter a estabilidade. Portanto, o compromisso dos profissionais é muito maior. E também dá muito mais, aos profissionais, oportunidades de crescimento profissional. Porque agora o pessoal de operações não está resolvendo falha todo dia, ele está fazendo script, fazendo automação, fazendo monitoramento e fazendo estratégias de operação. E tem, obviamente, um benefício também para o negócio, melhor *time-to-market*, comunicação dentro da organização que é muito mais fácil. Tem mais ferramentas para fazer as comunicações e evitar todo o tipo de confusões, que normalmente existem quando você faz tudo manual. E, portanto, muito mais tempo para fazer inovação, que é importante. A inovação, afinal, tem benefício concreto no negócio. Não é só uma coisa que é bonita, é uma coisa que tem rentabilidade.

Agora uma pergunta que sempre a gente faz: qual é a relação entre SDN e DevOps? Porque você tem hoje um conjunto de tecnologias que vão para o mundo da SDN. SDN desde a perspectiva de OpenFlow, P4 ou SDN da perspectiva de, por exemplo, uma rede com *segment routing* e um PCEP para MPLS. Ou, inclusive, uma rede com SDN baseada em scripts, também é possível. Qual é a relação? Bom, na nossa visão, DevOps é uma prática das organizações, das pessoas. SDN, ou programabilidade da rede, é uma tecnologia da rede. É uma questão técnica. Então, o DevOps precisa da SDN, mas não é a mesma coisa. As pessoas não são SDN, as pessoas fazem, mas têm metodologias, e utilizam os benefícios técnicos da SDN.

E um dos seus benefícios técnicos, importante, é a agilidade. Eu não posso fazer agilidade na organização se quando eu faço interação(F) com a rede, é tudo manual. É impossível. Eu preciso ter uma rede que responde rápido. E o benefício, portanto, da SDN é um benefício desde o ponto de vista da rede, poupança de custos, melhor engenharia de tráfego, maior eficiência da rede. E os benefícios do DevOps são benefícios a nível da organização, poupança de custo também, [ininteligível] é menor, mais ferramentas, talvez menos processos manuais, melhor engenharia de operação e também maior eficiência. Mas se você enxerga aí, tem dois planos. O SDN está no plano da rede, camada 1, 2, 3; o Network DevOps está na camada 8, das pessoas, para falar de uma forma.

Bom, temos ferramentas para fazer SDN, e muitas delas eu sei que os colegas da apresentação vão falar delas, eu não vou falar de nela delas, tem muita coisa hoje disponível, muita coisa que é *opensource* e que tem custo baixo e que permite fazer uma série de... implantar os princípios básicos de Network DevOps. Por exemplo, iterações incrementais, eu estou todo dia fazendo um processo de inovação, de teste, de análises e de liberação de serviço. Eu termino de liberar, eu começo novamente um novo ciclo, sempre. Todos os meses, todas as semanas. Portanto, liberar serviços, para mim, não é uma aceitação, é uma coisa que é rotineira.

Continuidade, estou fazendo sempre, contínuo. Nunca termino, é processo, não é um projeto, é um processo. Automação, eu preciso de automação. Para ter um processo contínuo, eu preciso configurar a rede em segundos, não em minutos, não em horas, em segundos. Preciso ter automação. Felizmente temos muitas ferramentas hoje para fazer automação a um nível de *opensource*. Autosserviço, muito importante, *self-service*. Eu não preciso pedir para uma outra pessoa fazer uma coisa na rede, eu posso fazer sozinho. Porque tenho um *framework* de ferramentas que me fornecem a possibilidade de fazer mudanças na rede com as autorizações devidas. Mas eu não preciso falar com o meu colega, para um acesso, uma autorização. Eu posso fazer sozinho, porque tenho um *framework* que me dá essa possibilidade.

Colaboração, quando eu posso fazer sozinho as coisas, eu já foco a minha conversa em como fazemos com que o teu serviço fale melhor com o meu serviço e que todo mundo funcione... tudo funcione muito melhor. Já não estou na coordenação da janela, de quem fez essa mudança, é uma conversação em um nível diferente. Visão holística. Visão extremo a extremo, ponta a ponta. Já não estou preocupado da caixa, do comando, da lista de acesso, estou procurando(F) do serviço geral, porque configurar uma lista de acesso, agora, na rede, é uma coisa que me toma um segundo. E se eu faço um erro, o *rollback* também é um segundo. Então, eu posso ter agora uma visão que é mais holística. E uma série de práticas que não precisamos ir aos detalhes, agora, eu vou compartilhar depois a apresentação, práticas que são bem concretas e que prometem fazer todos esses princípios que eu mencionei.

Agora, já para ir fechando a minha apresentação, e como é sempre concreto, o mundo do software tem os conceitos de CI/CD, *continuous integration and continuous delivery*, que são práticas da indústria do software para em todo momento estar fazendo teste do software. Cada vez que eu toco o software, eu inicio um teste. Então, se eu quebrei uma coisa no código, eu vou saber, *self-service*, não preciso perguntar para ninguém. Toco o código, quebrei, eu sei, corrijo, já não está quebrado? Perfeito. E tem *continuous delivery*, que uma vez que o código está pronto para ser liberado, o processo de liberação

nos servidores da Amazon, por exemplo, é automatizado. Não é à mão, é automático.

Como que a gente implementa uma coisa similar no mundo da rede? Mesmas ferramentas, por exemplo aí temos o Git, que é um repositório de código para código fonte de aplicações, mas o que acontece?? É que as configurações dos roteadores também é texto, e pode ser levado para um repositório Git. E você tem também ferramentas como Jenkins, Jenkins é um motor de *continuous integration* e *continuous delivery* que faz uma automação de diferentes processos. Por exemplo, pega uma configuração nova, proposta por um engenheiro, faz uma verificação. Como é que faz a verificação? Faz o *deployment* automatizado na máquina de laboratório. Tudo bem? Perfeito. Passei essa etapa. Vou para a seguinte fase agora: inserir parâmetros. Coloco todos os parâmetros específicos da rede, preparando a configuração para a implantação final. Poderia fazer uma aprovação, poderia criar um e-mail automatizado para o chefe para que pegue(F) uma olhada, as configurações, e faça uma aprovação. Ele responde no e-mail: ok. Quando o Jenkins consegue o ok do chefe no processo, ele libera as configurações, faz esse SH(F) através de Ansible para os roteadores e pronto, a configuração já está em produção. E se eu preciso fazer *rollback*, a mesma coisa, muito rápido, muito mais seguro e é uma [ininteligível] de agilidade. A gente tem feito projetos desse tipo com bastante sucesso e utilizando principalmente ferramentas de *opensource*. Então, isso é o que eu queria contar. A minha apresentação só procura ser uma introdução, porque temos aqui nesse painel profissionais com muita experiência das ferramentas, que eles vão explicar em muito melhor detalhe. Obrigado.

SR. EDUARDO BARASAL MORALES: Obrigado, Guzmán, realmente, muito interessante a sua apresentação. Até porque você já falou de conceitos como NetDevOps, que é, assim, prosseguimento da carreira de alguém que é da área de redes, é uma coisa que ele já pode pensar em como se especializar.

Pessoal, se tiverem alguma dúvida para o Guzmán ou para qualquer outro palestrante, fiquem à vontade para escrever no chat do YouTube. Então, já postem ali as perguntas, a gente vai coletando essas suas perguntas e depois a gente vai fazer as perguntas aos palestrantes. Então, nessa primeira rodada inicial, a gente vai ouvir todos os palestrantes e depois a gente abrir para perguntas. Mas já podem ir escrevendo, não precisa guardar ela para o último minuto. Já escreve. A gente já está capturando elas, já estamos armazenando e depois a gente vai fazer elas para os palestrantes.

Outra coisa, também, se vocês quiserem interagir no chat, colocar o nome da empresa, a cidade, o estado, até para a gente acompanhar de onde é que a gente está conseguindo alcançar com

essa live, fica à vontade aí para participar e interagir com a gente. Às vezes a gente manda algumas mensagenzinhas no chat, tá? A gente tem a nossa equipe, às vezes eu e o Moreiras, a gente consegue já ir respondendo e dando um toque aí para vocês.

Mas, seguindo aí a nossa rodada de palestras, eu vou chamar agora o Helio Barroso, da Cisco, para fazer a sua apresentação. Então, Helio, a automação de redes não nasceu ontem, né? Existe aí uma cronologia do desenvolvimento de soluções. Poderia nos contar um pouquinho aí sobre a evolução dessas soluções? Qual a visão da Cisco sobre a progressão dessas tecnologias? Até porque a Cisco se envolve muito nisso. Desde ali o IETF, como também fazendo novas soluções baseadas nos clientes e apresentando coisas proprietárias. Quais foram as dificuldades encontradas e como foram superadas? Conhecer essas soluções é essencial, hoje, para o reconhecimento profissional na área de redes e infraestrutura? Então, o palco é seu, Helio, fica à vontade.

SR. HELIO BARROSO: Bom dia, pessoal. Primeiro, grato pelo convite, né? Sua pergunta é bem abrangente, eu tentei aqui consolidar o máximo do tópico, porque o tópico é muito abrangente, né? Então, focando na área de *service providers*, a gente vê, sim... Tem passado por uma experiência larga aí nos últimos... praticamente, nos últimos 7, 8 anos, né, enfrentando aí com muito vigor esse desafio da automação das redes, dos *service providers* e também lidar com conceitos como programabilidade de redes.

A Cisco evoluiu muito nesse tempo para atender às demandas colocadas pelos negócios dos *service providers*. A história surgiu, na verdade, ela é bem antiga. Porque se você for em toda operadora aqui no Brasil, você... de grande e médio porte, ao menos, né, que é mais o nosso foco, você vai ver que tem sempre algum nível de automação, que seja rudimentar, baseada em script, baseada em SMP, mas ele existe, né? Porque sempre foi impossível lidar com toda demanda. Mas a CLI ainda prevalece. E todas as suas dificuldades também prevalecem, embora haja um investimento muito claro. E está hoje no plano de negócio de toda operadora aqui no Brasil atingir o máximo nível de automação.

Principalmente motivado... Opa. Principalmente motivado... Desculpa, pessoal, eu compartilhei a apresentação errada aqui. Principalmente motivado pela digitalização, né? A digitalização que trouxe uma massiva utilização, uma massiva customização de serviços, necessidade de respostas rápidas, e aproveitamento dos gastos, agilidade no negócio, né? Ela tem um impulso muito alto agora nas operadoras. Mas a discussão não começou hoje. A discussão, ela vem de tempo. Tem iniciativas aí, o PCEP, por exemplo, é um protocolo que está entre a gente há coisa de 15 anos e ele é um item de programabilidade hoje, nas redes de *service provider*, ainda é. A Cisco está propondo simplificação.

E tem também... Mas [ininteligível] com a história do SDN, né, que já era um movimento que já vinha de algum tempo, mas tomou, a partir lá do Martin casado, em Stanford, tal, criou a sua primeira concepção, e aí vem um debate muito forte e que se tornou viável, na verdade, por conta da evolução do hardware, da capacidade do hardware e da virtualização sobre a simplificação das redes. A nossa visão, a visão da Cisco de que a questão não é mais uma decisão por um protocolo ou outro protocolo, simplesmente por uma arquitetura ou outra arquitetura. O foco dos negócios, para atingir o nível demandado por digitalização é simplificação, automação e virtualização. Virtualização para escalar o máximo possível. Automação para tornar a operação de rede rápida, provisionamento rápido. A detecção de problemas rápida, a predição de problemas, viabilizar a predição de problemas. E a simplificação, porque as redes, elas vêm com um histórico aí de mais de 30 anos, de padrões, de protocolos que são seguidos, que hoje já não atendem mais às necessidades dos negócios. Ela é complexa e ela tem que ser... e ela está sendo paulatinamente simplificada. Não é mais uma discussão de protocolos, o *drive* aqui, a orientação aqui é simplificar, tá, nesse caso.

Falando um pouco de cada um desses... simplificação, automação e virtualização, eu não vou tocar muito aqui pela falta de tempo, tá? O primeiro tópico, de virtualização. A gente tem coisa de quase dez protocolos hoje na rede. Tem mais do que essa lista aqui, tá, em uma rede de *service provider*. O fato é que traz uma imensa complexidade disso, dificuldade, falta de agilidade e complexidade. A palavra complexidade já diz tudo. A proposta da Cisco tem sido, e de outros *vendors* também, já é quase uma unanimidade nesse mercado, a simplificação da rede. Como? No transporte, usar o IGP, SPF, ou ISS(F), o que for aí do seu gosto, com o *segment routing*, que trouxe uma capacidade de programabilidade para as redes, escalável, sem precedentes, que permite não só a implementação de roteamento em único domínio, em um único *backbone*, mas estender isso até o acesso mais remoto, tá? Uma tecnologia única, unificadora de domínios.

O BGP para serviços, BGP já é antigo, consolidado e que tem se mostrado o protocolo mais adequado, principalmente quando trabalha na camada de serviços. Isso tudo com o apoio de controles para unificar e simplificar o roteamento, de orquestradores de rede para automatizar o provisionamento e configuração de redes, reação a falhas. E também é uma imensa quantidade de dados trabalhados hoje com Inteligência Artificial e *analytics* avançado, que te permite ajudar muito na manutenção da rede, no suporte da rede, tornando ele preditivo... preemptivo, com técnicas como a análise de causa-raiz e tudo o mais que desenvolvem na operação dia a dia da rede.

Essa simplificação, ela tem muito a ver também, no *service providers*, com os domínios, né? Porque hoje uma rede metro, uma rede core e um Data Center, que são componentes necessários para

os serviços oferecidos em um *service provider*, além do acesso, são domínios separados, com administração separada, com carga pesada de gerenciamento separado. Inclusive, as formas de implementar IP na metro, no Data Center e no core vêm sendo ultimamente diferentes.

O esforço da Cisco e da maioria dos *vendors*, hoje, é simplificar isso através do *segment routing*, trabalhando com [ininteligível] e trabalhando... e o EVPN na camada de serviços. Com duas linhas, de protocolos e arquiteturas, a gente consegue, assim, entregar, não só entregar os serviços com mais agilidade e mais facilidade, mas consegue, pela primeira vez, olhar para a rede como se fosse uma única coisa, só, através do IP. Que é uma coisa que vai ter [ininteligível], começa agora, engatinhando, com o advento do *segment routing* versão 6, baseado em IPv6, onde a gente torna desnecessária aquela a camada do MPLS, aqueles *lables* que 'tagueavam' os serviços, que 'tagueavam' os Nexthops para despacho de pacotes, eles passam a ser concentrados, hoje, no cabeçalho, no *header* do IPv6, que já tem essa concepção, já traz no seu *header* a capacidade de endereçar serviços, de endereçar o que eram antes *lables* para implementar o roteamento de uma rede, substituindo definitivamente o MPLS. E indo além, indo não só nos domínios do Data Center ao Metro, indo também para agregação, indo para os serviços, que é um benefício que é muito demandado, principalmente, para as expansões que virão pelo 5G, né?

Nosso modelo de automação, tá? O nosso modelo de automação é declarativo, baseado em modelos, apoiado por *machine learning*, a inteligência artificial e muito baseada em dados, né? Para implementar isso, a gente conta com vários componentes, né? As ferramentas Cisco? Sim, temos ferramenta Cisco. Mas a gente... várias ferramentas Cisco. O *opensource* é muito importante para a gente. Por quê? Ele complementa essas soluções e ele acelera a entrega das nossas soluções. Hoje, uma ferramenta da Cisco, ela não parte de uma aplicação monolítica escrita do zero, né? Ela conta com componentes já existentes, que estão atendendo bem no mercado, onde a Cisco adiciona ao seu *set*... adiciona, coloca dentro das suas ferramentas e suporta os clientes, né, também nessa utilização aí de *opensource*. Ela aumenta, ele melhora as suas ferramentas, torna o desenvolvimento mais rápido para entregar as soluções de automação.

Assim como conceitos já expostos, não vou repassar isso, no CI/CD, NetDevOps, e os vários sabores, porque a gente... Uma coisa que é importante ressaltar: não existe uma única solução que caiba para todo modelo. Uma coisa é uma OTT, que tem investimento gigante, recursos gigantes, capital intelectual gigante. Outra coisa são empresas intermediárias, operadoras no Brasil, ou mesmo a maioria das operadoras na Europa e nos Estados Unidos e na Ásia, tá? E tem também os negócios menores, *enterprise*, tal. Para cada um tem um modelo de automação, tem o que se adequa melhor. É impossível você

achar um modelo que seja único de negócios, de relacionamento que atenda todas essas demandas. Estamos falando de *service providers*, estamos falando de *service providers* no Brasil e América Latina, então, é para isso que eu estou direcionando essa apresentação.

O nosso modo de automação, ele é baseado em modelos, ele é declarativo. Você declara a sua intenção: o que você quer implementado, baseado nos serviços, tá, que você quer ver na rede. O como isso é feito, que é o que impera na CLI e que é o que impera, importante frisar, em quem automatiza redes baseados em scripts que são scripts que hoje interagem, na prática, com o CLI, é o modelo imperativo, onde o objetivo principal, que é o que tem que se entregar, se perde um pouco, se perde muito nos detalhes de configuração *device, device a device, e, principalmente, no como fazer*. A nossa ideia é abstrair isso. A proposta do nosso modelo, do modelo que a Cisco defende, e praticamente é uma unanimidade hoje, é declarativo. Você tem que ter um sistema de automação onde você declara a sua intenção e as ferramentas, e a camada de ferramentas abaixo faz o seu serviço.

Ferramentas e modelos. Que para implementar isso você precisa tornar os *devices* modelos, ser baseados em modelos, através de modelos como o Yang, por exemplo, que é uma linguagem, que é um padrão que permite você a unificar a linguagem para a interação com redes, com os *devices*, tá? Superando as diferenças entre *vendors*, superando o problema de... em muito problema dos *skills* necessários para tocar uma rede.

Desculpa aqui, nela você tem, aí, ele é baseado em orquestradores que trabalham com os *controllers* para fazer o provisionamento e configuração de rede. O Yang mais protocolos Netconf e a opção(F) mais HTTP do Restconf são aí as estrelas desse modelo. Assim, todo roteador da Cisco hoje tem suporte a esses protocolos, tem suporte a Yang, que possibilita a você colocar os roteadores da Cisco com outros roteadores de outros *vendors* e tratar isso como algo unificado, tá?

Agora, a grande vantagem do modelo, já frisei isso, do modelo declarativo, é você... é agilidade, habilidade de você fazer *updates* parciais, quando você trabalha com modelo baseado em script, alguns outros protocolos aí, com o OpenFlow, você tem dificuldades para *rollback*, você tem dificuldades para fazer *update* parcial, para lidar com mudança, para fazer *rollback* e tudo. Isso está tudo suportado dentro do declarativo módulo, dentro do modelo de... da ideia de orientar a configuração de rede, o provisionamento de rede e a operação de rede a modelos, né? Que te permite *rollback*, ele te permite uma série de recursos que lidam com muito da complexidade do como fazer, tá?

Não vou insistir nesse tópico, ele já foi muito bem abordado pela Whitestack, né? É importante destacar que é o seguinte: não basta você programar, você ter scripts e programar, ter automação, você tem que programar correto, e para você programar correto em um processo de automação é fundamental. A cultura de desenvolvimento de software, que vem com DevOps, que te permite pré-testes, pós *deployment* testes, *continuous integration*, *continuous delivery* e todos os conceitos [ininteligível] que te permitem não só provisionar, mas provisionar rápido, configurar rápido e também provisionar e configurar certo e corrigir erros, né? Erro é sempre uma preocupação em toda operadora.

Falando em erro, vamos falar dos primeiros, dos *use cases* mais comuns que a gente tem implementado através da América Latina, tá? O continente, ele está, falando sobre andamento, uma das dificuldades que a gente vê em operadoras são duas, basicamente, a primeira é que a gente passa por um longo período de baixo investimento, né, dificuldades, isso tem retardado a automação. Porque demanda um investimento inicial, né, para você parar de gastar tanto, você tem que trabalhar algum investimento. Quando eu falo investimento, não é só investimento Capex, é Opex também, é o pessoal, é envolvimento do pessoal.

E também tem alguma resistência cultural. O nosso continente, normalmente, ele demora mais a tomar algumas decisões. Ele prefere que, principalmente operadores, preferem que as coisas aconteçam lá fora, madurem lá fora para depois passar a adotar. Mas há progressos. Há progressos no Brasil, sim, a gente tem, nesses sete anos implementado, em vários lugares, a automação. No México, na Colômbia, por toda a América Latina, a gente consegue ver esse movimento. E os primeiros *use cases* de automação são esses que a gente está listando aí, são os principais, você atua, por exemplo, em janelas de manutenção, para simplificar, as janelas de manutenção, janelas automáticas, através da execução de MOPs automáticos, né, [ininteligível] automáticos. Ela tem, a Cisco tem aplicações para *upgrade* de IOS que faz *end-to-end*, com pouca necessidade de [ininteligível] pelo operador, inclusive, com *rollbacks*, [ininteligível] o *upgrade* de IOS que em uma operadora grande em *router* [ininteligível] é sempre um parto, né? Aquela janela de seis horas para fazer, e a gente conseguir diminuir muito significativamente o tempo disso, reduzir aí o esforço a 15, entre 10 e 15% do tempo.

Migração de *devices* constantes dentro das operadoras. Validação de configuração, provisionamento, virtualização, que até vai para outras camadas, né? Um dos nossos principais *use cases* hoje de virtualização, de automação de virtualização está no [ininteligível], aquela parafernália toda de *mobility* que passou nas operadoras, tecnicamente(F) passaram do hardware, mais consolidado isso no

Brasil e na América Latina para o software, né? Para o *openstack* ou gradualmente para containers.

Falando de um caso legal aqui, só para dar ideia do que a gente está fazendo, já estou entrando nos *use cases* de automação. *Ingress Peering*, *egress Peering*, controle de Peering por latência, por custo, *load-share*, baseado em telemetria, com resiliência, todos os *use cases*, que se você for olhar para as operadoras, ainda, hoje, aqui no Brasil, elas são indispensáveis, nem cogita porque não... acha que com BGP você consegue fazer isso. Mas suportado por telemetria, suportado por BGP, pelo *segment routing* e também suportado, principalmente pelo Netconf e Yang, a gente consegue implementar, tem casos gigantes, por exemplo, no Japão, onde a gente consegue deixar isso tudo automático. Inclusive com tempo de resposta excelente, até fazendo um off-road de alguns protocolos de roteamento. É um *use case* muito interessante de se mostrar, um dia de ter só uma live só sobre esse assunto.

Fundamental para a gente: não basta só programar a rede, configurar, provisionar de forma automática, a gente tem que resolver os problemas rápido, a gente tem que antecipar os problemas através de predição. A gente tem uma enorme quantidade de dados dentro da rede, tá, que a gente pode coletar. E hoje, com a telemetria, a gente conseguiu uma grande mudança. Um grande impulso nisso. Porque agora a gente tem mais dados e um tempo menor e de forma mais escalável. Por exemplo, em uma escala que o SMP nunca atingiria.

Então, a partir desse câmbio para telemetria, e através da consolidação dos dados e aplicação de *analytics* avançado, aplicação de *machine learning* e outras tecnologias(F), e outros... inteligência artificial, a gente tem conseguido avançar muito na parte de manutenção da saúde da rede, no *troubleshooting* e *remediation* cada vez mais rápidos. Em aplicações *bandwidth on demand*, em manutenção de SLAs, de *performance tuning* e controle, *control plane*. Eu tenho alguns *use cases*, eu imagino que eu estou encerrando aqui, já, que a gente já está no final do meu tempo aqui. O tempo já se esgotou. Tem *use cases*, aqui, por exemplo, identificar um problema com BGP, que é uma agulha no palheiro, você consegue através de monitoramento de BGP [ininteligível] e suportado por *dashboards*.

Você também consegue, por exemplo, detectar tráfego assimétrico, tá? Você consegue reduzir o tempo entre falhas através de técnicas de inteligência artificial, você tem ferramentas para isso, para determinar a causa-raiz. Engenharia de tráfego tem aí já dezenas de *use cases*, baseada em telemetria, baseadas em *playbooks* aplicados via Ansible, baseado em *segment routing*, acompanhado do PCE.

Bom, isso aqui, passar muito rapidamente, só sobre alguns casos da tecnologia. Desculpa por ter passado do tempo aí. Agradeço e

vamos seguir no debate aí, que tem muito mais para falar sobre o assunto.

SR. ANTONIO MARCOS MOREIRAS: Muito, muito obrigado, Helio. Foi uma excelente apresentação aí. E o pessoal está interagindo no chat, fazendo perguntas de um jeito que eu nunca vi aqui nas nossas lives, viu? Eu estou muito, muito feliz com esse público.

Espero que a gente consiga juntar as perguntas e colocar todas. Mas teve até uma solução para a gente levar esse tema também para um podcast. E olha que pela quantidade de interesse e de perguntas que estão sendo feitas, talvez a gente tenha que fazer isso. Eu estou duvidando que a gente vá conseguir responder todas hoje. Talvez, a gente leve esse tema para outras instâncias aí. Volte a trabalhar nisso. Até porque é um tema muito, muito importante e interessante, né?

E eu vi aí o Demi, que é nosso diretor, acompanhando, a gente, fazendo comentários. O Danton, está sempre com a gente, é sempre participativo lá no GTR e também estava por aí fazendo comentário, então, abraço.

A Silvana Corrêa, ela respondeu, eu fiz uma pergunta lá no chat, se tinha alguém fora do Brasil e a Silvana Corrêa disse que está direto de Porto, Portugal. Eu não sei se ela é portuguesa ou não, mas... E me disseram aí que um Lucas Angola, que deve ser da Angola, eu não tenho certeza. Mas se tiver mais alguém assistindo, de fora do Brasil aí, dá um oi para a gente lá no chat, para a gente ter uma ideia. Eu vou dar um oi também para o Wagner do Salto Soluções Corporativas, que pediu um salve. Nunca mandei um salve antes, vai ser o meu primeiro salve ao vivo aqui no YouTube para o Wagner.

O Diogo Fenner (sic) fez uma pergunta, eu sei que não é a hora das perguntas, mas parece que ele adivinhou já, né, o que a gente ia perguntar para o Elizandro Pacheco e perguntou quase a mesma coisa. Então, ele falou: qual é o modelo de automação aí para pessoal dos provedores menores, com *vendors* diferentes? E a gente sabe que Elizandro, que está sempre aqui nos nossos eventos, trabalha com consultoria, trabalha com ensino, ele tem uma grande experiência com os provedores regionais, com os provedores menores, né?

Então, Elizandro, a gente sabe que tem aí um mercado enorme de fabricantes de equipamentos de rede, a gente sabe que os provedores, normalmente, usam equipamentos aí de vários fabricantes diferentes, conforme suas necessidades e os seus bolsos, né? E as redes são muito heterogêneas, *multivendor*, principalmente no caso dos provedores regionais. Então, o que você pode falar para a gente, Elizandro, usando a programação, a gente tem como ajudar a criar soluções aí integrando e automatizando essas redes de uma forma que o trabalho dos provedores também, dos provedores regionais, dos provedores menores aí, fique mais fácil, né? Você acha que os profissionais têm que conhecer as soluções desse tipo para continuar

no mercado, continuar trabalhando bem? O que você pode falar para a gente, Elizandro? A palavra é tua.

SR. ELIZANDRO PACHECO: Bom dia, Moreiras. Bom dia a todo mundo. Sem dúvida, né? Eu, inclusive uma curiosidade, antes de eu trabalhar com redes, né, e ingressar, lá em 2004, com provedores, eu trabalhava profissionalmente com desenvolvimento, né? Então, eu vou trazer um pouco o que o pessoal já apresentou, falando mais na linguagem dos provedores regionais e onde que isso, de fato, pode se aplicar, né, nas nossas redes, que geralmente é tão diferente do que o pessoal está acostumado aí nas operadoras, enfim, né? Cada vez se aproximando mais, mas, enfim, ainda muito diferente. Então, deixa eu compartilhar a minha tela aqui. Está minha tela certinha aí? Tá, né? Beleza, então.

Então, quando a gente fala de programabilidade de redes, olhando para o ISP, né, basicamente a gente está levantando questões e problemas que muitas vezes os provedores regionais, eles sequer notam que existe, né? Mas, como também já foi falado anteriormente, conforme o provedor cresce e a rede cresce, inevitavelmente, ele vai ter que lidar com problemas como: a centralização da gestão da rede, a escalabilidade e a praticidade, né?

Então, isso aqui basicamente resume muito do que o provedor vai encontrar conforme a rede cresce. Afinal de contas, a gente já não está mais na época em que a gente tinha a visão de que um provedor regional, com mil assinantes, era um provedor grande, né? Então isso escalou muito, e hoje a gente tem uma rede aí, nos provedores, *multivendor*, né? É fato. Não tem hoje rede em provedor, praticamente, né, que utilize 100% de um único *vendor*. Então, a gente tem um roteador de uma marca, o firewall de outro, CGNAT de outra, o switch de outra, a OLT de outra, e assim por diante, né?

E, na grande maioria das vezes, a gente é obrigado a lidar com softwares que possuem requisitos diferentes. E também problemas diferentes, obviamente. Então, um só roda em Windows, o outro precisa de uma versão específica de sistema operacional, o outro precisa de um hardware específico, e assim por diante, né? Além, claro, de a gente ter o grande problema de que geralmente não tem nenhum tipo de integração entre eles.

Então, o nosso trabalho manual fica de um tamanho que, conforme a rede cresce, isso vira uma bola de neve que se a gente não prestar atenção agora, amanhã ou depois essa bola de neve vai atropelar a gente, tá? Então, a soma de todos esses fatores acaba resultando em desperdício de recurso, e conseqüentemente... E eu não estou falando só de recurso de hardware, eu estou falando também de pessoal e financeiro, né? Um ambiente de produção, né, complexo e de difícil manutenção. E muitas vezes tem um cara que mexe na OLT, um cara que mexe no roteador, um cara que mexe no BNG, e isso, é

difícil integrar tudo isso, né? E isso é justamente as ilhas isoladas de gestão, né? E também a dificuldade na manutenção de suporte. Quando a gente precisa de algo ali da nossa rede, às vezes de um fabricante, enfim, né, uma série de outros problemas, tá?

Então, a gente precisa notar e começar a estudar o que já tem no mercado, né? Porque o pouco que tem disponível hoje para os provedores regionais nem sempre estão de acordo com o que se faz necessário para escalabilidade de rede, né? Com isso, acabam surgindo alguns sistemas que facilitam tarefas simplesmente, né? Ele é o cara... Que é o 'scriptzinho', que vai lá, ou, enfim, né? Mas utilizam tecnologias que foram adaptadas para essa função, né? Como o Telnet, aí, acontece muito, né? Inclusive tem sistemas grandes aí no mercado que automatizam e são muito bons, diga-se de passagem, mas que utilizam tecnologias que não foram feitas especificamente para isso. Então, utilizando dessa forma que até então, muitas vezes é só o que a gente tinha disponível, também, nos equipamentos disponíveis nos provedores regionais, a automatização de tarefas utilizando esses métodos, hoje ela pode parecer uma solução, mas ali na frente, ela pode criar outro problema, né? Então, os sistemas que se utilizam de soluções desse tipo, eles colocam em risco diversos outros fatores e estão sujeitos à quebra muito facilmente. E aí é que começa a entrar a história das apresentações anteriores, né? Então, por exemplo, em sistemas que utilizam lá um Telnet, por exemplo, para realizar uma tarefa e te retornar um dado, tem muita necessidade de a gente utilizar lá *parsers* de textos, formatar isso, né, em um padrão e padronizar esses retornos, tá?

Mas, veja bem, no momento que eu faço isso, uma simples alteração de tabulação, por exemplo, né, por parte de um fabricante em uma atualização, quebra todo o sistema. E isso não pode acontecer nas nossas redes. Então, é nesse ponto que se faz necessária a padronização. E como a gente já viu e devemos acreditar, ela existe, né?

Então, a gente tem aí protocolos de configuração de rede e padrões de modelagem de dados. Esse é o caminho que a gente tem que seguir. E o primeiro ponto que os desenvolvedores e administradores de rede devem olhar é Netconf e Yang. Eu estou falando, pessoal, do que a gente deve começar a olhar para depois juntar tudo isso. Então, eu trouxe Docker aí, nos anos passados, através do lançamento do livro e tal, e agora a gente está vendo isso aí ser aplicado em outros lados, e Netconf e Yang, sem dúvidas, é a duplinha do bem que vai nos ajudar muito, tá?

Então, quando a gente utiliza esses padrões, tanto de protocolo como de modelagem nos nossos sistemas, a gente aumenta muito a confiabilidade, além de prover uma capacidade de escalabilidade e compatibilidade com múltiplos *vendors*. Então, quando eu utilizo

Netconf e Yang, eu utilizo com Huawei, ok. Aquilo ali está legal, está funcionando. Se eu precisar, agora, botar um dispositivo Cisco, 'cara', isso aí... a implementação disso, como já tem um padrão, um protocolo definido e uma modelagem de dados definida, a implementação disso é rápida. Eu não preciso reescrever todo um sistema, ver como é que ele retorna, fazer gambiarra lá nos retornos do Telnet, existe um padrão. Então, isso torna a rede *multivendor* melhor gerenciável. E essa gerência automatizada, com uma confiabilidade muito grande. Isso sem contar que eu posso pegar o Git, posso fazer versionamento de versão e etc.

Esses medos que a gente geralmente tem de: vai aplicar uma configuração, e se 'deu pau', a gente está 'ferrado', tu acha que eu vou confiar isso em um sistema? Isso tem que ser quebrado. E Netconf e Yang vem aí para ser a ferramenta que vai nos ajudar nisso. E muita atenção aos frameworks, tá? Eu arrisco a dizer que eles vão... que redes vai tomar um caminho bem semelhante, como já dito também ao que aconteceu com desenvolvimento nos últimos anos, né? Então, antes era meio uma zona, todo mundo desenvolvia todo mundo junto. Aí depois começaram a separar cada coisa no seu lugar, né? Um modelo, começando lá pelo MVC e aí, depois começaram a surgir os frameworks aí, que hoje são uma mão na roda e, 'cara', é praticamente impossível trabalhar. E já existem frameworks para isso, né, para automação de redes. Então, prestem atenção nisso. E eu fico realmente muito contente que esses dois mundos se integrem de novo, né? Principalmente porque eu adorava programação, mas eu programava a parte contábil de uma URP. Então, agora poder programar para redes é uma coisa que sempre me motivou, tá?

Então, lá em provedores regionais, esse processo todo, ele já vem acontecendo, né? E, em determinado momento, quando descobriram lá que era preciso abandonar o WinBox e ir para CLI, foi um primeiro choque. Então, se criou aquela ideia de que o cara que manja de CLI, que decora os comandos, que é uma bíblia de comando, que ele é o cara indispensável, na verdade, ou que ele é o cara, assim, nossa, né? E, na verdade, isso foi só o primeiro processo, né, pessoal? Tu largar o WinBox lá e ir para o Shell foi só o primeiro passo, né, porque isso foi uma lacuna de tempo. Quem começou antes lá, já estava no CLI, né? E agora volta isso para poder automatizar. Então, isso foi o primeiro passo. Foi importante? Está sendo importante? Obviamente que sim. Mas é suficiente? Obviamente que não, tá?

Então, com a migração aí dos provedores regionais para equipamentos como Huawei, Cisco, Juniper, agora Netconf, Yang e etc. são acessíveis também ao nosso mercado, né? Então, a gente passa a ter suporte nos nossos equipamentos, que está ali no nosso *rack* a tudo isso que a galera está falando aqui, entendeu? Então, a gente vai para outra realidade e entende que CLI não basta, tá? Então, justamente a falta da bibliotecas, frameworks e compatibilidade com padrões de

mercado e *vendors* do passado deixaram tanto os administradores quanto os empresários cegos para essa tecnologia, até porque muita gente nem sabia que isso existia. E aí é que começa a surgir o sistema para resolver tarefas isoladas e outro monstro se cria.

Então, Netconf e Yang, a gente tem padrão de protocolo, de modelagem de dados e a gente consegue ter múltiplos *vendors* falando a mesma língua com o mesmo software, tá? Ainda que a gente não entre diretamente na utilização de framework para essa centralização, é importante que a gente saiba que a grande maioria dos fabricantes, além de suportar isso, alguns ainda disponibilizam seus próprios SDKs para desenvolvimento.

E claro, que a gente não pode esquecer de containers. O Docker, por exemplo, que eu já trouxe há alguns anos para cá, para o mundo dos provedores regionais, é uma ferramenta que além de economizar recursos de hardware, não ver esses absurdos que a gente vê nos provedores, economiza pessoal e tempo e, conseqüentemente, dinheiro, além de trazer a flexibilização da escalabilidade, porque ela se torna praticamente infinita, né? E aumenta muito a nossa confiança no nosso ambiente de produção, beleza? Então, hoje em dia manter uma máquina virtual para cada serviço, para integração com *vendors* diversos e etc., como a gente geralmente faz nos provedores, é um desperdício sem fim, né? Além dessas integrações padronizadas através de software, com protocolo de redes e etc., isso se torna cada vez mais indispensável para o crescimento, tá?

Então, anote aí, provedor regional, o que vocês têm que aprender na sequência: Netconf, Yang, Python, tá? Linguagem para a rede, vocês vão ver que tudo ou muito, né, se usa em cima disso. O pessoal da Globo pode até falar melhor do que eu. E Docker, né, e não só Docker. Eu digo Docker, porque é o que a gente já tem conteúdo focado na gente e etc., tá? Mas, depois de tudo isso, vocês já vão ter um bom começo e depois vem a parte de juntar tudo isso e, claro, juntar o desenvolvimento, né? E, claro, seguir o sucesso, os exemplos de sucesso de quem já fez, tá?

Eu, particularmente, quando vi, comecei a ver Netconf e Yang dessa forma, junto com meu *brother*, Diorges Rocha, desenvolvemos um projeto de conceito, para resolver problemas simples, quando o 'cara' migrou lá para Huawei, por exemplo, ele precisava, simplesmente, ver os dados do cliente PPPoE dele. E ok, fazer isso via SNMP(F), ou com a 'gambiarrinha' do Telnet, beleza. Agora, a gente queria fazer isso com Netconf integrado. O Diorges aí, grande parceiro meu, programador *web*, Full Stack também. E com ele a gente fez, a gente integrou através de Netconf e Yang dispositivos Huawei para diminuir os impactos de quem vem de RouterOS, por exemplo.

Então, só para vocês terem uma ideia, rapidamente, esse aplicativo aqui, ele está utilizando Netconf e Yang para, por exemplo,

eu ver dados de um usuário, aqui, consultar um usuário PPPoE, sei lá, em qualquer, e ter uma informação em tempo real do que está trafegando, dados da sessão PPPoE, obviamente IPv4 e IPv6 ali, está estabelecido. Mas simplesmente isso aqui, pessoal, que foi tão rápido eu pegar esses dados de um assinante, isso aqui está desenvolvido utilizando [ininteligível], um framework PHP, mas utilizando o Netconf e Yang em cima de roteadores Huawei fazendo BNG, tá? Então, se eu precisar migrar isso aqui rapidamente para Juniper, para Cisco ou para qualquer outro, esse sistema, apesar de parecer que ele está fazendo uma função simples, ele está utilizando todo o *back end* de Netconf e Yang. Então, migrar isso para qualquer outra plataforma é rápido. E eu posso gerenciar 200, 300 mil equipamentos aqui, centralizado, e essa é a grande sacada da coisa, tá, pessoal?

Então, a gente tem várias facilidades, capacitação de Suporte N1, obtenção de informação sobre dispositivos, segurança na comunicação, né, para não deixar 'telnetzinho' exposto, centralização da gerência, velocidade de *troubleshooting* e escalabilidade.

Então, pessoal, vamos em frente e sair da CLI e ir além dela, da CLI, para a gente não ficar obsoleto.

Espero ter conseguido trazer o que esses gigantes estão falando um pouco mais na linguagem do provedor regional. Vocês saibam que a gente, óbvio, já trabalha em cima disso e esse é o caminho, tá pessoal? Então, estudem, prestem atenção nisso aí, exijam mais dos desenvolvedores, esse tipo de integração, porque o futuro está aí. Se a rede de vocês quer crescer, vocês querem crescer de forma organizada e confiável esse é o caminho.

Então, agradeço demais o convite do pessoal do NIC, para mim é sempre um prazer e uma honra estar junto com vocês. Espero ter conseguido trazer um pouquinho aí desse conteúdo também para provedores regionais e me coloco à disposição de todos vocês. Depois, se for para o podcast também, tenho o maior prazer, caso eu venha a ser convidado também, tá bom, pessoal? Então, um forte abraço e a gente se vê mais ao final na rodadinha de dúvidas, beleza?

SR. EDUARDO BARASAL MORALES: Bom, obrigado, Elizandro. Realmente foi muito interessante toda a sua apresentação. O Elizandro é um cara que realmente já participa com a gente em várias frentes aí de eventos e lives, mas ele é um cara bem polêmico, né? Ele simplesmente mostrou um negócio aí do SBT, sendo que a gente tem um cara da Globo apresentando, gosta de polêmica, né? Botar um fogo no parquinho.

Bom, mas piadas à parte, realmente, foi muito interessante. É legal de você ter comentado que o parque tecnológico de muitos provedores mistura aí múltiplos *vendors* e, às vezes, a gente fica pensando que tem que ter ali um desenvolvedor, um programador, um administrador que saiba todas as linguagens, conversar em todas as

máquinas e que isso daí é muito complicado. E, é claro, usando aí essas ferramentas já padronizadas, como Netconf e Yang, a gente consegue integrar tudo de maneira simples.

Bom, seguindo aí as nossas apresentações, eu vou chamar agora o Renato Almeida de Oliveira, do CITEx, do Exército brasileiro, para fazer a próxima apresentação.

Então, Renato, muitas organizações têm um certo receio e uma resistência ao uso de soluções de automatização para configuração das redes. Como o usuário, e também desenvolvedor, que a gente já conhece, e que você já participa dessas comunidades, dessas soluções, que recomendações você daria para os profissionais da área começarem a estudar o uso dessa tecnologia nas redes? Quais os ganhos que eles vão ter usando essas tecnologias? E, já puxando aí algumas perguntas que vieram no chat, se tem soluções gratuitas? Então, o palco é seu, fica à vontade. Renato, você está sem som.

SR. RENATO ALMEIDA DE OLIVEIRA: Pronto, consegue ouvir? Pronto. Obrigado. Queria agradecer o convite para a participação na live. E em relação à pergunta, quando você for na Internet procurar por programabilidade, vai aparecer um monte de coisa, um monte de tecnologias distintas, cada uma atuando em uma área um pouco diferente do seu equipamento. E aí a minha ideia aqui é trazer um pouco de: o que a gente tem disponível e onde cada uma dessas ferramentas atuam melhor dentro do seu equipamento. Então, por exemplo... Só trazendo uma pesquisa rápida, você consegue ver diversas ferramentas, como por exemplo: Ansible, Napalm, PyATS, Netmiko, Nornir.

E bom, qual delas é melhor usar? Qual delas a gente prefere não usar? E aqui é importante sempre pensar em qual nível de abstração(F) que você deseja dentro do que você está fazendo. Então, por exemplo, quando a gente está trabalhando com automação de equipamento existem várias etapas que a gente busca atingir. Então, primeiro, a gente precisa se conectar ao equipamento. Então, vamos supor que a gente vai se conectar via Telnet, ou se conectar via SSH ou, por exemplo, Netconf e Restconf, dependendo(F) do que você quer trabalhar. Então, se a sua preocupação, se o que você precisa fazer no momento é: quero me conectar, e a partir daí eu já tenho alguns tipos de [ininteligível], já tenho algumas [ininteligível] prontas, você pode buscar soluções como, por exemplo, Paramiko. Paramiko é uma solução que a finalidade dela é acessar o seu equipamento via SSH. E isso não é alguma coisa relativamente simples e fácil, você simplesmente sai implementando o código como um todo, porque manter uma sessão SSH depende de você ser capaz de se conectar ao equipamento, identificar qual o *prompt* que o equipamento te retorna, como por exemplo, se você tiver trabalhando com IOS, Cisco, o *prompt* é de uma forma, se estiver trabalhando com [ininteligível], o *prompt* é

de outra forma. O RouterOS, em Mikrotik, é de outra forma. Então, compreender cada uma dessas formas e equipamentos está(F) te respondendo ao protocolo SSH tem uma dificuldade própria. Então, com isso você, com o Paramiko, você consegue estabelecer a sessão.

Então, uma vez estabelecida a sessão, cada tipo de equipamento tem uma CLI(F) própria. Então, se você não tem uma forma de receber essa CLI(F), receber a linguagem que essa CLI(F) está te entregando, processar ela e conseguir trabalhar com ela, você vai ter uma dificuldade muito grande. Então, para isso a gente tem, por exemplo, o PyEZ para a linha Junos, então, é um framework próprio da SDK da Junos para conseguir conectar no seu equipamento Juniper e extrair as configurações. Em seguida, a gente tem o Netmiko.

Percebe que está de forma sequencial, então o PyEZ, ele utiliza dentro do seu código Paramiko. Então, essas ferramentas, elas estão uma utilizando as outras de uma forma direta ou indireta.

Em seguida nós temos o Netmiko. Com o Netmiko, a gente consegue ter um *range* maior de [ininteligível]. Então, com o Netmiko, eu consigo acessar um [ininteligível], eu consigo acessar um Cisco, consigo alguma coisa com o Juniper. E mais aqui à direita, a gente já tem um framework, esse já está acompanhando a linha da Cisco, que é o Genie, em que ele se propõe justamente a isso, ele se conecta a diversos tipos de *devices*, como por exemplo, Cisco, recentemente eu fiz a contribuição para o [ininteligível], Juniper e diversos outros.

E, bom, consegui me conectar ao equipamento. Agora, eu quero fazer mudanças de alterações, mudanças de configurações nele. E com isso, a gente tem outro protocolo, que é... outro framework, que é muito legal, que é o Napalm. Napalm, ele usa um conceito que é muito interessante que ele não se baseia em se conectar ao equipamento e ir jogando diversas linhas de comando, uma sequencialmente a outra. Ele usa uma técnica que alguns operadores já atualizam dentro das suas infraestruturas só que de forma manual, que é montar o arquivo de configuração internamente, dentro de um bloco de notas ou qualquer coisa. Pegar esse arquivo de configuração, ele manda para o dispositivo e faz uma operação de *configure replace*. A partir dessa configuração de *configure replace*, o próprio equipamento já faz toda a troca da configuração.

E esse é um conceito que ele é muito interessante, depois a gente pode não necessariamente utilizar o Napalm, mas trazer conceitos de uma [ininteligível] de framework para utilização do seu *script* sem necessariamente utilizar ele. Então, se você estiver começando a programar e já souber algo de programação, é sempre interessante que você vá até o repositório de cada um desses frameworks, e dê uma estudada no código individual deles, porque mesmo que você não queira utilizar o framework em si. Quando você utiliza conceitos de um

conceito de um framework para a sua aplicação, você consegue ter algo muito mais amplo e muito mais flexível.

Então, subindo mais o nível, eu não estou preocupado aqui, quando eu estou trabalhando com automação, em configurar um único equipamento. Eu quero configurar dezenas, centenas de equipamentos de forma paralela. E aí, a gente já começa a ter outros dois frameworks, que é o Nornir, que recentemente até saiu uma certa disputa entre Ansible e Nornir. O Ansible, ele, ainda estamos em nível de código, então, até esse momento eu estou falando em códigos de Python, você escreve o código, um *script* para execução das tarefas e o Nornir, ele é extremamente eficiente na serialização das suas conexões. Então, você cria um procedimento para ele executar, ele vai receber o código, vai serializar e distribuir ele na quantidade de equipamentos que você quiser.

O Ansible, da mesma forma, só que é capaz de fazer essa serialização, só que o Ansible, ele abstrai a linguagem de programação para você. Então, se você não sabe Python, não quer, por algum motivo, aprender Python ou algum tipo de código de programação(F), o Ansible, ele entrega para você uma linguagem de domínio, uma linguagem específica de domínio. Então, você vai escrever um arquivo de código, YAML, em você vai juntar vários bloquinhos, vários módulos, cada módulo vai fazer uma tarefa, um pedacinho da tarefa. Juntando cada uma dessas tarefas, você tem a execução da sua automação.

E aqui é interessante que o Ansible, como uma solução como um todo, cada módulo dele utiliza quem está embaixo. Então, por exemplo, você tem o módulo de Ansible que utiliza framework da Cisco Genie. A gente tem módulos de Ansible que usam Napalm, módulos de Ansible que utilizam Netmiko. Então, apesar de você não estar vendo diretamente que você está utilizando, provavelmente, a maioria desses frameworks que estão disponíveis, as ferramentas que você utiliza, utilizam eles de forma indireta. E mais em paralelo a isso, já com PyATS, que acompanha o framework da Genie da Cisco, que é a capacidade de executar testes. Isso eu vou falar um pouco mais para a frente.

Bom, mas até agora a gente ainda está em CLI, eu ainda estou falando em acessar um terminal, escrever o código e digitar o comando [ininteligível] Python executa o meu *script* ou então Ansible *playbook execute* ou o *playbook* que eu definir. E aí ele vai executar todas as tarefas que eu defini, que existe(F) no meu código ou através do meu *playbook* Ansible. Subindo mais um nível, a gente já tem uma interface de usuário, então, a gente já tem uma [ininteligível] em que usuário consegue acessar terminal web, uma página web, executar a sua automação e receber respostas a respeito dela. Então, para isso que a gente tem o AWX, que é versão *opensource* e o Ansible Tower, que seria a versão proprietária, já seguindo na linha do Ansible. E na linha

da Cisco, o PyATS, a gente tem o Xpresso, que foi lançado acho que até recentemente, que está disponível publicamente.

E agora as coisas começam a ficar interessante porque você começa a pensar mais como ecossistema. Você tem uma aplicação web que expõe uma API Rest(F), ou expõe algum tipo de API, que você pode colocar para se comunicar com outros tipos de funcionalidade de monitoramento que você tem, como por exemplo, vamos supor que você tem monitoramento implementado(F) em Zabbix e que quando corre determinado alerta configurado dentro da sua infraestrutura, você faz com que o Ansible ou o Zabbix, por exemplo, tome uma ação para executar um determinado um *script* seu. Então, você consegue colocar a sua infraestrutura reagindo de forma muito mais rápida quando você passa a integrar as suas diversas funcionalidades de automação.

Dando mais um passinho. Agora vem essa aí, quando conecta a um equipamento, quando a gente fala em CLI, a gente vê dados não estruturados. Então, a gente recebe uma linha de texto, um arquivo de texto onde a gente recebe um texto gigantesco e tem que tentar processar de alguma forma. E isso é muito vinculado à sintaxe de cada equipamento. Então, se a gente estiver falando com Cisco, nós falamos com... essa sintaxe é de um [ininteligível]; se nós estivermos falando, por exemplo, com [ininteligível] Huawei, [ininteligível] HP, a gente está falando de outro tipo de sintaxe. Se a gente falar de Huawei, é outro tipo; Mikrotik é outro tipo. E isso acaba prendendo e limitando até o próprio profissional, porque ele acaba se acostumando com um tipo específico de sintaxe, ele estuda aquilo e o conceito de rede, o conceito de infraestrutura dele é baseado em *syntax*.

Quando a gente começa a trabalhar com programabilidade, um dos passos é tentar abandonar a adesão orientada a *syntax*, a gente começa a tentar pensar em uma visão baseada em modelo de dados. E aí modelo de dados existem diversos tipos. A gente pode pensar, quando fala em Netconf e Yang, Yang é, de fato, só uma separação, Netconf e Restconf são protocolos que transportam modelos de dados. Esses modelos de dados são definidos através de Yang, e aí ele tem uma RFC própria e não existe uma definição, "Ah, o modelo de dado é esse". Cada fabricante, às vezes, tem o seu próprio modelo de dados, seu próprio modelo Yang. E esse modelo, ele varia, pode variar de fabricante para fabricante, inclusive de versões. Então, por exemplo, se a gente está trabalhando com Cisco, dependendo da versão do IOS, você tem uma mudança do próprio modelo de dados Yang. E aí existem versões abertas, como por exemplo a OpenConf, que é um modelo de dados liderado por uma equipe *multivendor* que busca uma padronização. Existe outro tipo de modelo de dados que é IETF, que também busca uma padronização. E todos esses modelos de dados, eles buscam descrever o equipamento como ele é, de forma agnóstica ao fabricante.

E aí as coisas começam a ficar mais interessantes, porque você começa a pensar na sua rede pela rede, pela técnica de rede, pelos protocolos e como ela deveria funcionar e se desvincula da visão de: "Ah, não, eu estou trabalhando com o Cisco, então, eu tenho que utilizar o comando X ou Y", ou então: "Estou trabalhando com HP e tenho que utilizar um comando ou outro". Quando você passa a pensar em forma de modelo de dados, você passa a ver a tua rede como ela de fato é, como os seus equipamentos têm que se comunicar um com o outro, quais são os *timers* que um equipamento tem que falar com o outro, e você passa a acessar isso diretamente, sem ter uma complexidade agregada que é a sintaxe proprietária do fabricante.

E aqui são dois passos. A gente, primeiro, precisa conseguir transportar da CLI que o dispositivo entrega, para o modelo de dados e, em seguida, transportar do seu modelo de dados para uma CLI. E aqui um framework, um módulo(F) que é muito utilizado para redes, não necessariamente é o melhor que existe, existem muitas críticas à utilização de Jinja, mas é uma linguagem de *templates*. Então, eu tenho, como o modo com o Jinja, eu consigo, com a mesma estrutura de dados, conseguir escrever uma CLI para utilizar, por exemplo, com Comware, com HP, uma CLI para utilizar com IOS ou um código de CLI para utilizar, por exemplo, com Mikrotik. E isso vai passar e te dar uma flexibilidade, porque agora você tem um código que é capaz de atuar na sua rede de forma transversal. Independe se você está trabalhando com Cisco ou se você está trabalhando com HP, ou com o que quer que seja. Inclusive, você ainda pode utilizar Jinja, apesar de talvez não ser necessariamente ser a melhor técnica, para escrever... para a utilização com o Yang.

Então, por exemplo, usualmente, quando a gente fala de... quando a gente trabalha com Yang, são dois protocolos básicos, a gente tem o RestConf, que a gente busca criar o Json, a partir desse Json, a gente envia para o equipamento, o equipamento faz a mudança da configuração dele. Quando a gente trabalha com o Netconf, a gente cria o XML e esse XML é inserido junto com openload, que é acessado via SSH para ir para o equipamento. E aqui, isso vem um ponto importante, mesmo que a gente esteja trabalhando com o Netconf, com Netconf, Restconf, ou qualquer outra coisa, a *control plane* e *data plane* ainda está nos equipamentos. Os equipamentos, eles ainda têm os seus protocolos de roteamento funcionamento da mesma forma. O que está se fazendo aqui, é: eu não(F) estou tirando essa inteligência que o equipamento já tem nativamente e inserindo nele. Eu estou simplesmente automatizando a forma de criar a configuração para injetar no equipamento. Isso ainda nos traz muita flexibilidade, até como... você pode descer, às vezes, o nível de protocolo para ter algo um pouco mais inteligente.

Então, vamos para o próximo passo. De todos esses protocolos que eu comentei, eu gostaria de dar um destaque especial ao PyATS,

que é um protocolo Cisco. Ah, outro detalhe, tudo o que eu falei até agora é *opensource*. Nada do que... nenhum dos protocolos que eu falei, nenhuma das coisas que eu falei, você vai precisar ter investimento na solução para poder utilizá-los. Você vai precisar de ter investimento de desenvolvedores, desenvolvendo código, mas a solução, em si, não.

E o que o PyATS traz e que é uma solução muito madura é a capacidade de testes. Ele é, de todos frameworks que eu apresentei, ele é o principal framework orientado a teste. E quando você está, mesmo como um profissional, fazendo modificação do equipamento, sempre que você faz a modificação, você testa, você nunca deixa de fazer, você nunca faz uma configuração do equipamento de rede e deixa de testar se o resultado dessa modificação foi efetivo ou não. E se você não... se você está passando para a automação e não pensa em testar aquilo que você está configurando no equipamento, você está criando um risco um enorme. Porque antes era uma pessoa configurando um equipamento por vez. Agora eu estou falando de um software que vai configurar os seus 300 equipamentos de forma simultânea. Então, o erro de configuração em um equipamento, você pode quebrar a sua rede toda. E nesse sentido, o PyATS, ele se apresenta como muito maduro, nessa forma de te dar framework próprio para realizar testes em cima da configuração que você implementou.

Então, só mostrando, por exemplo, um caso de uso que você pode utilizar, seria, por exemplo, o que o PyATS te permite? Você é capaz de se conectar ao equipamento, realizar um *backup* de todos os seus equipamentos, só que, bom, você terá trabalhando com automação, você pode ter uma falha na conexão do equipamento e não se foi capaz de realizar o *backup*. Então, você é capaz de verificar se o seu *backup*, ele foi, de fato, executado ou não. Se o seu *backup* não foi executado, ele interrompe a execução e te alerta: olha, eu não consegui realizar o *backup* de um equipamento ou outro; olha, esses níveis de testes que você implementou antes executar não passaram, então eu não vou continuar a minha implementação.

Em seguida a gente pode... a gente começa a aplicar a configuração. Então, a gente vai, aplica a configuração em todos equipamentos e, novamente, vamos fazer testes. A configuração que eu apliquei no equipamento, ela produziu o resultado que eu esperava? Ela foi, de fato, aplicada? Tive alguma falha de conexão no equipamento [ininteligível] que eu não consegui aplicar a configuração? Não, a minha configuração está correta. Então, a gente vai para o próximo passo. Será que os resultados da minha configuração foram os resultados que eu esperava? Então, vamos supor que a gente está falando em configuração de filtros BGP(F) para fazer anúncio de rota. Será que todas as rotas que eu estou anunciando são adequadas em relação àquilo que eu espero? Você consegue fazer

esse tipo de teste. E, caso você tenha falhas em cada um dos seus estados, com o próprio framework, ele já te realiza o *rollback* de todos os seus equipamentos e, em seguida, a desconexão e te informa sobre tudo isso que aconteceu.

Então, acho que esse era o meu último. E só para finalizar, eu gostaria de fazer destaque a mais duas outras ferramentas que são maravilhosas dentro desse contexto que é o NetBox. O NetBox é uma ferramenta de modelagem de equipamentos em um IPAM em que permite que você modele os seus equipamentos, configure os seus equipamentos nele e ele vai servir somente como fonte de verdade para as suas automações. Então, quais são os equipamentos que eu tenho na rede e como eles se relacionam entre si? Quais são as credenciais e como eles funcionam? Isso é extraído através do NetBox na minha relação de equipamentos.

E por fim, que é uma parte consideravelmente crítica disso é como é o meu acesso? Quais as credenciais que eu preciso para acessar meus equipamentos? Como eu vou fazer a gestão dessas credenciais? E nisso a gente tem junto a ferramenta da HashiCorp Vault. Ambas são *opensource*, então, vocês podem utilizar [ininteligível] toda essa infraestrutura. E elas permitem, uma, fazer um gerenciamento da sua documentação, e a outra fazer um gerenciamento de nível de acesso com credenciais, provendo tokens e revogando tokens de acesso para os seus usuários. E acho que a minha apresentação termina aqui, obrigado.

SR. ANTONIO MARCOS MOREIRAS: A gente é que agradece, Renato. Muito obrigado pela apresentação. Foi, assim, extremamente didática a tua apresentação. Muito boa mesmo.

Aliás, gente, eu não sei o que vocês estão achando da live agora, até vou pedir para colocarem no chat aí do YouTube, no Facebook, o que é que vocês estão achando. Mas é assim, a gente que está organizando, eu, o Eduardo, toda a equipe, a gente convida os palestrantes, a gente tenta coordenar o tema, né? Mas eu estou achando aqui que os palestrantes se conversaram escondidos da gente aí, e coordenaram num nível super, ultra, plus, porque, assim, parece que cada palestra aqui está complementando a outra de uma forma que, nossa, as coisas vão se encaixando e vão respondendo as dúvidas.

Eu estou vendo que muitas das dúvidas que vão aparecendo no chat o próximo palestrante vai respondendo, o próximo palestrante vai respondendo. Parece que vocês também que estão acompanhando aí a gente vão adivinhando, né, qual que é o próximo passo lógico disso daqui e vai indo para a frente.

Eu só estou chateado com uma coisa, temos 720 pessoas assistindo e só 430 likes no YouTube. Então, vocês não estão gostando, gente? Se está excelente assim, tem bastante gente dizendo que está muito boa, então, vamos dar os likes lá no YouTube. Porque o NIC.br

difícilmente paga impulsionamento de vídeos lá no YouTube. Então, a gente depende da distribuição, ali automática, que a plataforma faz. Se não tiver o like de vocês, a plataforma não indica esse vídeo, mesmo que o cara esteja lá inscrito no canal, com o sininho ativado e tudo mais. O que, aliás, se vocês não estão ainda, também podem fazer. Se inscrevam no canal, ativem o sininho para receber a notificação dos nossos vídeos, entrem também na nossa página do Facebook, deem o like na página. Temos também presença em outras redes sociais, no LinkedIn. O pessoal há pouco estava perguntando, há pouco não, já faz um tempão, no começo da live estavam perguntando: Ah, pessoal, passa os contatos do LinkedIn dos palestrantes. Olha, na página do NIC.br, do LinkedIn, quando a gente fez a propaganda da live, a gente falou: "Ó, vai ter a live no dia tal, dia tal", todos os palestrantes estavam 'lincados' lá, os perfis do LinkedIn dos palestrantes, inclusive os nossos, dos moderadores também. Então, se vocês entrarem lá na página do NIC.br no LinkedIn, vocês vão achar os perfis lá de todos os palestrantes, enfim. Podem pedir para adicionar e tudo mais.

E vamos dar continuidade aqui no assunto, já falei demais. E o próximo painalista aqui é Leopoldo Mauricio, da Globo. A Globo tem apoiado a gente em diversas iniciativas, está com a gente, por exemplo, no OpenCDN em Salvador, em Manaus. Bom, Leopoldo, sem sombra de dúvidas aí, a gente sabe que a rede, o *backbone*, a infraestrutura da Globo é gigantesca, com muitos equipamentos, muitos *racks*, enfim. A escalabilidade a gente imagina que seja bastante importante para vocês e que não dê, na prática, para ficar configurando equipamento por equipamento lá com a CLI. Então, a gente pede para você contar um pouquinho para a gente como que são, nas práticas, as soluções que vocês adotam de programação, de automação, se vocês usam SDN? Como que vocês criaram essa infraestrutura que vocês têm hoje, que funciona tão bem que suporta as operações da Globo? E quais foram as dificuldades encontradas nesse processo todo, como é o *case* da Globo, Leopoldo? Então, a palavra é tua, fica à vontade.

SR. LEOPOLDO MAURICIO: Bem, gente, bom dia. Eu sou Leopoldo Mauricio, como Moreira falou, eu trabalho na Globo. Eu vou falar um pouquinho da empresa, do que a gente implementou na empresa como solução de SDN, falar um pouquinho disso. Eu estou aproveitando uma palestra que eu fiz com Gustavo Pantuza há uns anos atrás a respeito desse serviço, dessa implementação que a gente fez. Eu adaptei ela para cá, obviamente.

Então, de cara, a gente começa sempre dizendo que nós somos uma empresa carioca, né? Vocês estão vendo que tem uma foto aí bonita do Rio de Janeiro, para tentar vender a ideia de que a Globo é uma empresa, pelo menos a *globo.com*, é uma empresa sediada no Rio de Janeiro, principalmente, onde nós trabalhávamos. Hoje a gente tem uma marca nova, todo mundo conhece, que é essa marca da

Globo. E todo mundo sabe que a gente é o maior grupo de mídia da América Latina, né? A Globo faz uns anúncios dizendo que nós somos cem milhões de uns, né? A gente tem mais de cem milhões de usuários consumindo os nossos produtos. E desde 2017 quase 80% de alcance na Internet brasileira. A gente tem vários produtos aí que o pessoal conhece, que são produtos líderes na área de vídeos, de esporte, na parte de notícias, né? G1 é líder, na área de entretenimento também. Todo mundo conhece o Big Brother, né? Todo mundo já votou lá no Big Brother, enfim.

A gente tem uma série de produtos suportados pelo nosso data center hoje. A gente tem um data center físico, tá? Tem aí um desenhinho tentando representar ele. E, dentro desse data center, que é um prédio hoje, a gente tem um montão de servidores, né? Mais de 5 mil servidores. E a gente tem esse data center, os servidores em uma rede e esse data center interligado a diversos outros ASNs, além de estarmos ligados a eles via PTT, a gente tem Peering. Enfim, a gente tem mais de 3 teras de banda hoje de ligação com os ASNs.

Se a gente entrar um pouquinho dentro do data center para olhar a arquitetura que a gente tem hoje suportando esses servidores e para que entregam(F) esses conteúdos de entretenimento, enfim, de notícias, de vídeo, a gente vai encontrar uma arquitetura *spine-leaf* que a gente usa hoje. Então, tem um montão de *rack*, tem um montão de roteadores spines, tem alguns roteadores *core* e a gente tem um monte de *racks* ligados a esses spines, e spines ligados a esse *core*, tem muito de BGP. E se você olha um pouco mais para cada *rack*, você encontra cada *rack* com equipamentos de topo de *rack*, além disso, você encontra dentro do *rack* múltiplas máquinas, a maioria delas suportando, a maioria das máquinas que suporta as aplicações são VMs, VMs ou containers, dentro de um *bare-metal*, dentro de um servidor físico. Então, tem um montão de servidor físico no *rack*, esses servidores estão ligados através do ToR para o mundo exterior ao *rack* e múltiplas redes existem dentro desse *rack*. Tem lá equipamento que está, seja VM, seja *bare-metal*, ou seja container, na rede vermelhinha, na rede laranja, na rede azul.

A gente tem então, aí um desafio, um desafio de gerenciamento de uma nuvem privada que é significativo, que é grande. Então, eu tenho centenas de *racks* ou dezenas de *racks* e eu preciso gerenciar dentro dessa dezena de *racks* um montão de servidores, que tem um montão de VMs dentro deles, que tem containers dentro das VMs. Enfim, é um mundão de coisas para gerenciar. Hoje, a gente tem gerenciadores de nuvens, a gente tem sistemas operacionais de nuvens tipo OpenStack. A Globo implementou o data center dela à nuvem privada, o data center dela usando o full CloudStack, está o símbolo do CloudStack aí, fazendo só um apanhado geral. E a gente tem *clusters* que vêm de XCP, além de KVM, dentro do nosso data center.

Se a gente der uma olhada rápida, está meio acelerado porque o tempo é curto, tá? Mas espero conseguir ser o mais claro possível. Se der uma olhada rápida na arquitetura dos nossos clusters de Xen, ou XCP, que é o que a gente tem como plataformas de virtualização mais usadas dentro da nossa infra hoje, a gente encontra, então, esses dispositivos de topo de *rack*, que estão lá em cima, ligados a um montão de servidores. Como a gente falou, o servidor está aí representado como *host*, na figura. E aí dentro do *host*, então, tem uma plataforma de virtualização e tem o hypervisor dela. Então, tem lá, por exemplo, o hypervisor do Xen, que está em azul e dentro desse hypervisor, ele gerencia, como plataforma de virtualização, uma série de VMs, ok? O hypervisor oferece também um Switch virtual, que aí está representado pelo OvS, verdinho. Esse Switch virtual permite, então, que as VMs consigam falar com o mundo exterior através das interfaces de rede da máquina física que enxergam, que estão ligadas, geralmente(F), lá em cima no equipamento de topo de *rack*.

Como eu falei antes, dentro da VM, também, a gente tem um montão de containers dentro dela e esses containers, eles estão em múltiplas redes, né? Rede vermelhinha, rede quadradinha, ali, azul, rede verdinha. Enfim, precisa de segurança para que esses múltiplos produtos, múltiplas VMs e múltiplos containers consigam funcionar dentro do desse data center de forma que eu não tenha brechas e falhas de segurança que permitam que alguém ataque a empresa e tire do ar os serviços que a gente tem. Então, a gente precisa pensar nisso.

A primeira maneira de organizar, a primeira forma que a gente pensou em organizar esse parque gigantesco foi a de segregar os dispositivos em diferentes ambientes. Então, tenho clusters Xen de XCP, de ambiente verdinho, ambiente amarelinho ali, ambiente vermelhinho. Então, segregando em ambientes diferentes, a gente consegue ter clusters isolados, do ponto de vista de roteamento. Então, eu tenho um cluster, por exemplo, em um ambiente de *backend*, tenho outros em um ambiente de *frontend*. Quem está no *backend*, todo mundo conhece bastante o *backend* e *frontend*. Então, o *backend* geralmente são as VMs ou máquinas que estão aqui com o [ininteligível] do código, executando alguma coisa que precisa ser processado, e o *frontend*, geralmente, é um servidor que está recebendo comunicação do mundo, né? O [ininteligível] que está sendo feito pelo 'cara' que está vindo do mundo, está batendo no *frontend*. Então, a gente organizou em ambientes de BE, de FE, entre outros ambientes de roteamento.

E aí, depois dessa organização, a gente olhou e disse: Ok, agora eu tenho ACLs, políticas, aqui eu estou chamando de lista de controle de acesso, que eu vou precisar criar para controlar quem fala com quem dentro do ambiente de BE ou dentro do ambiente de FE. Inicialmente, as listas de controle de acesso eram sempre implementadas em [ininteligível] de rede, né, antes do topo de *rack*,

geralmente, você consegue colocar ACLs, mas não dá para colocar mais de 150 mil ACLs dentro de um equipamento de rede para ficar gerenciando isso, dentro de um ToR, porque é caro demais e não vai escalar. Para começar a brincar, tem desenvolvido dentro da Globo uma API de gerenciamento de política, que a gente chama de ACL API. É baseado nessa segmentação de ACL API, então, ela trabalha baseada na segmentação que a gente criou de *backend* e de *frontend*, tá? Tem uma série API gerenciando BE e uma série API gerenciando coisas de FE, por exemplo.

Como eu disse, é caro colocar isso no equipamento de rede, porque a Tcam, que iria processar todas essas políticas é cara para caramba, então a gente partiu para software. Como partir para software? Fazendo SDN. Na hora que a gente pensou em fazer SDN, a gente integrou, então, a gente olhou, deu uma olhada geral e viu: Ah, tem muita rede, tem muito dispositivo, tem muita VLAN, tem muito ambiente, o que é que a gente faz? A gente quer, então, fazer o quê? Tirar, com SDN, a política que estava no topo do *rack* e descer ela para o sítio(F) virtual. Esse foi nosso objetivo, né? A gente programou a Network API para traduzir as séries para um modelo OpenFlow. A gente usa o protocolo OpenFlow. E aí, a network API, a gente já escrevia documento de topo de *rack* de diferentes fabricantes, Cisco, Juniper, Dell, *wherever*, e depois da nossa construção, ela passou a escrever também as séries OpenFlow nos OvSs.

A gente tem um controlador SDN. Na época do doutorado, eu pesquisei lá a [ininteligível], entre outros, e o que eu escolhi na época do doutorado ainda foi o OpenDaylight para ser controlador SDN. O OvS é o switch virtual do Xen, por padrão, que é Open Switch. Então, a Network API fala com quem? Fala com... ela está programada para falar com os meus controladores, e os meus controladores para falar com os meus Switches virtuais, beleza?

A gente traduz ACLs, modelos qualquer que seja, para um Json, que vai ser, obviamente, processado para que as ACLs sejam escritas nos equipamentos. Estou terminando aqui. Então, a gente tem as ACL API(F) que fala com a Network, a vermelhinha, falando com a Network API, então, enviando as ACLs. A Network API escreve as ACLs em uma fila. Aí eu tenho uma série de consumidores que pegam esses dados Json, em formato Json dessa fila, escreve em um controlador SDN, que é o OpenDaylight, e o OpenDaylight escreve essas ACLs nos diversos Switches virtuais, onde as VMs estão efetivamente ligadas.

Nossa infra hoje tem 40 *workers*, tá, consumindo essa fila e eu tenho ambientes, hoje, com cerca de 80 servidores por ambiente, por exemplo, rodando. Oitenta servidores, dá mais ou menos dez clusters de Xen. O nosso tempo de... a gente tem um desempenho bom, essas CLs são escritas rapidamente. A gente tem o nosso controlador otimizado, hoje, para responder 62.5 *requests* por segundo, tá? O

nosso controlador é uma VM. Tudo aqui é *opensource*, né? E a gente... a nossa infra é tão otimizada que a gente consegue escrever políticas nos Switches virtualizados mais rápido do que nos equipamentos físicos, né? Então, a nossa API de rede, que é Network API, gasta mais de 20 quase 25 segundos para escrever políticas em cem ACLs, por exemplo, em dispositivos de rede, físico, e eu consigo escrever em Switches virtualizados em tempo muito menor. O ambiente é resiliente, a gente guarda as ACLs nos Switches virtuais, mesmo que eu tenha queda na Network API, eu não perco a ACL; se o controlador cair, eu não perco ACL. Se cair o Switch virtual, porque a máquina deu ruim, a máquina física, eu tenho as ACLs guardadas no nosso banco da ACL API, as políticas estão lá, elas são reescritas quando a máquina nova sobe, tá?

Então, tudo isso é transparente para o usuário final. O DevOps... Eu cheguei no final, né? O DevOps só pede que a app(F) A precisa falar com o banco de dados B, ou precisa falar com o *file system* C. Ele não precisa ter visibilidade de onde ele está escrevendo. Ele pede isso através de uma interface web e não importa se essa política tem que ser escrita em 40 [ininteligível] Switches virtuais, não importa se dez controladores SDNs serão chamados, isso é transparente para o DevOp. Ele só interage com essa interface web, pede o que deseja e tudo é escrito para ele de maneira transparente, em uma automática.

Passei dois minutinhos, mas tentei passar a visão geral.

SR. EDUARDO BARASAL MORALES: Muito obrigado, Leopoldo. Realmente é muito interessante a visão que você mostrou, né, de aplicação de SDN. E já até cutucando as coisas que você falou, né? Que você falou: tinha ali mais de 150 mil ACLs. É o que a gente estava falando, uma rede grande, não dá para a gente ficar pensando em querer configurar tudo na mão, manualmente, entrar na CLI e tentar fazer isso. Temos que pensar aí em processos de automação. E o caso de vocês, da Globo, é muito interessante, porque mostra aí o SDN funcionando.

Bom, seguindo aí a nossa rodada de apresentações iniciais, eu vou chamar agora o Fabio Nunes, do IX. Fabio, a gente sabe que hoje o IX, ele está presente em, se não me engano, 33 localidades espalhadas pelo país, com número expressivo de Sistemas Autônomos participantes, né? A gente até fala que o IX de São Paulo é o maior do mundo em quantidade de participantes, mais de 2 mil participantes, com picos de mais de 11 tera de tráfego, sendo ali um dos maiores. Depende do dia, ele é o maior do mundo, em tráfego, depende, no outro dia, às vezes ele fica um pouquinho abaixo, mas sempre está ali entre os maiores. Para manter toda essa infraestrutura funcionando de forma eficiente, escalável e segura, foi muito importante vocês mudarem e automatizaram certos processos nos últimos anos. Poderia nos contar um pouco como foram essas mudanças, quais foram as

dificuldades encontradas, como essas mudanças trouxeram melhorias operacionais para toda a rede do IX? Então, fica à vontade, Fabio.

SR. FABIO PESSOA NUNES: Bom dia, pessoal, obrigado pelo convite, eu sou o Fabio do IX.br, da equipe de engenharia. Na verdade, realmente, a gente tem tido um crescimento expressivo nos últimos anos, Eduardo. E realmente não dá para pensar na rede do IX sem um, no futuro, sem realmente investir em automação, né? É muito importante para, realmente, a gente conseguir garantir uma qualidade e para os nossos participantes, tá? Então, a ideia aqui, na verdade, é mostrar alguns processos que, na verdade, até... mais práticos mesmo que até os provedores que estão iniciando com processo de programabilidade, que possam até aplicar nas suas redes, né, um processo inicial, assim, para você sentir e pegar gosto, digamos assim, pela coisa e realmente mudar a cultura da sua rede, realmente, para esse cenário, tá? Vamos lá.

Bom, o IX.br tem algumas particularidades, algumas delas até compartilhada com muitos provedores, né? Nossa rede, nossas redes têm até muitas similaridades com ISPs. Mas, hoje, nós somos localizados em 33 localidades, que são redes totalmente independentes entre elas, ok? Sendo que a soma de tráfego total de todas essas localidades está atingindo acima de 15 terabits por segundo. E cada um desses sites, cada uma dessas redes têm vários sites nessas localidades e cada um desses sites têm vários elementos de redes desenvolvidos(F) para atender nossos participantes.

No nosso cenário, a gente tem vários tipos de equipamentos, vários fabricantes para vender o mesmo serviço. Então, a gente nunca foi preso a um *venders* específico, a um software específico. Além de que alguns equipamentos já são um pouco mais legados, que dificultam, às vezes, um processo de automação mais simples e mais amigável para a gente, operador, certo? Com toda essa expansão é frequente a gente precisar fazer expansões de capacidade na rede, aumentar a capacidade, trocar equipamentos para fornecer mais capacidade para os nossos participantes. Então, é muito frequente esse trabalho que a gente tem, que acho que pessoal até vê pelo tanto de janelas de manutenções que a gente faz aí nas nossas redes, ok? E, além disso, a gente fornece vários tipos de conexões para nossos participantes. Então, a gente não consegue ter uma padronização exata de como a conexão do nosso cliente, participante está sendo feita no nosso equipamento. Então, isso dificulta também uma padronização geral da rede, ok?

Bom, com toda essa diversidade, é essencial que a gente tenha uma fonte de verdade da rede, né? Até o Renato já comentou, a ferramenta, [ininteligível] NetBox aqui, que deixo até como solução, como sugestão para quem está começando, ou quem está em um *lab*(F) e queira fazer, começar esse processo, mas é uma ferramenta

muito completa, tem muito a agregar aí. Uma API muito boa que dá realmente para começar muito bem esse processo de automação, tá?

Mas essa... no nosso caso, ter uma base de dados confiável é essencial, porque como eu vou saber que o equipamento, na localidade tal, no site X tem um equipamento da marca tal, com um modelo tal, que tem essas conexões com o IP tal, é muita informação para se colocar tudo no *script*. É muita coisa. Então, a existência dessa base de dados acaba sendo uma peça-chave nesse processo, pelo menos nesse início de uma automação como um todo, certo?

Bom, tendo essa base de dados confiável, de onde tirar essas informações, eu preciso saber como, que dados acessar do equipamento, como acessar esses dados e como trocar essas informações com os *devices*, certo? Então, na verdade, aí que entra o modelo Yang, que ele já foi citado por vários outros palestrantes, mas que são realmente tipos de contratos especificando: ó, para trocar esse dado, para fazer tal ação em um equipamento, nesse equipamento específico, você precisa mandar a informação nessa estrutura. Então, o Yang, ele não define a forma, não define se é um XML, se é Json, como você está mandando isso, mas ele define a estrutura desses dados. O campo, o nome da interface tem que ser string(F), o nome de tal tamanho. A VLAN, ela só pode ser inteiro de 1 a 4.094. Então, o Yang realmente modela esses dados para que você saiba como trocar exatamente a informação correta com o equipamento, ok?

E ele não serve apenas para configuração dos equipamentos, ele serve também para informações e dados de estado do equipamento. Então, eu preciso de uma informação do estado de uma interface de erros [ininteligível] que seja, ele vai me definir que a informação que eu vou receber na caixa, da caixa, tem uma estrutura daquele formato, ok? Facilita muito para, realmente, eu poder mapear o que eu vou receber, certo?

Como já foi dito anteriormente, esses modelos, eles geralmente são disponibilizados pelos fabricantes, tá, eles são disponibilizados pelos fabricantes, mas não são apenas disponibilizados pelos fabricantes, tem consórcios *opensources*, [ininteligível] que especificam esses modelos e também organizações que como o IETF, IEEE e ITU-T, tá? Bom, tendo esses modelos definidos, ou sabendo como tem que fazer essa troca de informação com os *devices*, a gente vai seguir, como comunicar com um *device*, que protocolo... Só complementando, aqui é um... eu vou deixar como exemplo um modelinho, como é um modelo Yang, né? Vocês podem encontrar facilmente na Internet. A apresentação está pública e depois vocês podem dar uma olhada.

Mas a gente precisa de uma forma de realmente comunicar com o *device*, um protocolo que faça essa comunicação, que mande esses dados que eu já defini no meu modelo de dados, certo? Então, no nosso

caso, a gente resolveu utilizar o Netconf, né? O editor, o mais ideal para nosso caso, que realmente faz essa interface de gerenciamento com o *device*. O Netconf, ele manda a informação para o *device*, ela faz essa troca de informação com o *device* via XML, mas eu, como usuário, eu não preciso ter essa informação. Eu não preciso realmente entender o XML e ir lá e criar o XML para mandar. Tendo o modelo Yang, já é facilmente mapeado entre um modelo de dados e o XML que o Netconf vai utilizar para mandar esses dados para o equipamento, para trocar esses dados com o equipamento, certo? Então, no nosso caso, aqui é um exemplo até que vocês podem buscar, né, a gente começou utilizando o Yang Development Kit, que é uma ferramenta *opensource* desenvolvida pela Cisco, mas funciona com qualquer tipo de equipamento que suporte Netconf ou Restconf e tenha as definições de Yang, tá? Então, facilmente utilizada. E basicamente o YDK, o que ele faz? Ele tem um módulo, que eu forneço para ele o modelo Yang, né? Que eu peguei lá do meu *vendor*. E ele transforma esse modelo de dados em uma biblioteca para o seu código, no seu software lá, para você poder usar como for necessário, ok?

Então, eu vou transformar isso nessa biblioteca, na hora de criar, por exemplo, fazer a configuração de uma interface, a única coisa que eu preciso fazer é distanciar lá um objeto de interface e especificar os campos dentro desse objeto, falando: Ah, o nome da interface é Gigabit Internet 1, ela está *shutdown*. Então, ele já pegou automaticamente do Yang e transformou isso em uma biblioteca ali, que vai só permitir para mim os valores que o Yang tinha definido. Então, já fez a mágica aí, para mim, eu não preciso me preocupar com o formato que isso vai ser enviado, tá? Assim, é específico... o *device*, eu peguei essa informação do fabricante, é só um modelo lá. Então, eu não preciso ficar me preocupando mais com formato de... fazer *parser* de configuração de *output* de comando, de uma linha de comando, que eu entrei lá com *expects*, coisa do tipo, sabe? Realmente, o Yang me deu esse poder de mapear tudo em um objeto ali e facilmente fazer essa troca de informação com o *device*, tá?

E isso me ajuda muito na agilidade de configuração. Então, eu recebo um equipamento novo, tendo o modelo Yang eu já... só jogar no YDK ali, ele já me gerou a biblioteca, eu já tenho a informação de como os dados devem ser trocados, tá? E isso me ajuda... mudanças em massa na rede, eu consigo fazer com muito mais facilidade, né? Uma migração em massa ou verificação geral na rede de todas as interfaces, facilmente eu consigo fazer essa brincadeira utilizando o YDK, tá?

Bom, aqui eu vou... assim, para ser um pouco mais prático mesmo, para vocês verem a simplicidade da coisa, é realmente um exemplinho usando o YDK, de configuração de uma interface *bundle* e agregando uma interface nesse *bundle*, utilizando um IOS XR, tá? Então, vocês podem ver que é pouco código aqui nesse... que você cria

a interface, distancia um objeto mesmo, da interface, atribui os dados que você quiser, que são permitidos, referentes àquela interface. Eu mando a configuração para uma *datastore* candidata, dentro da caixa, né? Depois, eu posso gerar outro tipo de configuração, que no caso é da interface física mesmo, agregando ela, habilitando, colocando descrição. De novo, eu mando ela para uma configuração candidata na caixa. Então, está vendo? O YDK, ele está fazendo isso automático para mim, ele faz essa comunicação com o Netconf e com a caixa baseado no próprio objeto ali, que eu criei, tá? Então, no final, ele faz o *commit* dessa configuração.

Se você olhar aqui no lado da tela, você vai ver, realmente, é uma mensagem que é trocada, a parte das mensagens que são trocadas entre o *device* e o Netconf. Você vai ver que é um XML ali, mas eu, como programador, não preciso me preocupar com isso. O próprio YDK fez essa transformação porque ele tem a informação do Yang, né, que ele gerou na minha biblioteca. Então, ele formatou certinho os dados como a caixa, o equipamento de rede está esperando receber para aquela configuração em específico, tá? E no final, ele também dá um comando lá, XML com o *commit* [ininteligível], tá?

Bom, essa foi uma configuração, mas eu quero pegar um estalo da rede, né? E não especificamente Cisco, eu tenho várias caixas na rede, de fabricantes distintos. Então, no caso, aqui a gente tem uma SLX da *Brocade*, que eu estou querendo saber todos os níveis LLDP que estão... dessa caixa em específico, tá? Então, de novo, eu distanciei meu objeto lá do [ininteligível] LLDP, mandou a requisição para a caixa, ele já me respondeu. A gente pode até ver aqui na parte do *debug(F)* mesmo, que ele me respondeu um XML já com toda a informação bonitinha ali. Mas eu não preciso formatar esse XML, ele vai lá e coloca tudo no objeto para mim, que depois eu vou poder trabalhar esse objeto para fazer o que eu quiser, tá? Bem simples e bem poderoso, tá?

Aí, uma outra coisa que a gente tem usado muito e pode ajudar também, muitos provedores, é o uso do ZTP, o Zero Touch Provisioning, tá? O ZTP, ele ajuda na automação inicial e configuração inicial dos equipamentos de rede, tá? O primeiro, o dia zero dele, da rede lá, né? Então, a ideia é que, quando ele não tem... quando ele dá o *boot* e não tem *startup-config*, ele entra nesse modo ZTP, ele fica solicitando um IP via DHCP. E, ao receber o IP via DHCP com a opção 67 'setada' lá no pacote, contendo o caminho de um *script*, a caixa faz o download desse *script* e executa toda a configuração que eu mandar ele executar, tá? Então, isso reduz muito o meu trabalho manual e toda vez eu tenho que indicar(F) o mesmo *template* em todas as caixas que eu vou ativar. E reduz erros também, porque toda vez a mesma configuração, eu tenho que mudar partes específicas do meu 'templatezinho' lá, é bem ruim, a gente sabe que erros acontecem. Então, isso elimina.

Então, a sequência dessas atividades no nosso ambiente lá é a seguinte, tá, só para exemplificar. Então, eu vou lá, eu configuro na minha base de dados, o equipamento, né, represento ela com interface, IPs, conexões, o *serial number* que ele tem. Essa é a única parte que eu preciso ir lá e realmente pôr a mão para realmente especificar a configuração do equipamento na minha base de dados. No momento que eu configurarei na base de dados, o DHCP server é configurado automaticamente para disponibilizar aquele IP fixo para aquele *serial number* com aquele caminho de *script* que eu 'setei'. Bom, a gente decidiu fazer um *script* genérico para todas as caixas, que vai acionar alguma outra automação externa que vai fazer essa configuração, tá? Isso daí eu falo em sequência.

Então, ok, está lá pronto para receber o IP. A nossa equipe de infra vai lá em campo, liga o equipamento na rede de gerência, está lá, certinho. O equipamento 'bootou', ele vai receber o IP e o caminho do *script* via DHCP, ele faz o download desse *script*, faz a atualização de software automática, e o *script* que a gente fez, ele aciona um Ansible externo, e esse Ansible vai pegar toda a informação que tem na nossa base de dados para configurar todas as informações iniciais da caixa que são necessárias. Então, minha interação com o equipamento é muito pouca, é zero, na verdade. Eu só preciso colocar na base de dados lá e vai aparecer o equipamento lá para mim, configurado já com o meu *template* lá no Ansible, né? Então, é uma atividade que realmente vale a pena aí, para vocês brincarem e pensarem em colocar como uma *feature* na rede de vocês.

Bom, é isso. Eu só quis dar alguns exemplozinhos aqui de alguns usos, casos de usos que a gente tem utilizado e vocês podem... os provedores podem, com facilidade, implementar também. Isso ajuda bastante na redução de erros, né, de atividades repetitivas, do dia a dia. E a gente está em um processo grande de automação, ultimamente, conversando muito sobre isso, que a gente busca que realmente no futuro aí, realmente não tenha interação humana nossa com os equipamentos do IX, realmente o pessoal possa entrar lá no nosso portal e pedir um serviço e esse serviço já ser automaticamente configurado para o participante, ok? Bom, é isso. Obrigado, pessoal.

SR. EDUARDO BARASAL MORALES: Obrigado, Fabio. Realmente, muito interessante tudo que você apresentou aí do ZTP.

Bom, sem mais delongas, porque a gente já está um pouquinho atrasado para a parte de perguntas. Eu vou dar só alguns avisos que são importantes para o nosso público, e depois a gente já vai selecionar algumas perguntinhas e já vai colocar para os nossos palestrantes. Já peço, de antemão, desculpa, porque a gente não vai conseguir postar todas as perguntas para os palestrantes, a gente vai selecionar algumas delas, porque a gente já está com o tempo um pouco

estourado. Mas vamos tentar, ali, abranger algumas perguntas que são mais recorrentes no chat.

Bom, primeiro aviso, a gente vai colocar agora, o formulário de avaliação. Então, essa live, a gente vai ter um formulário de avaliação, que são duas perguntinhas simples, tá, para a gente saber o que vocês estão achando da live. Se vocês gostaram, se o tópico está interessante e o que a gente pode melhorar. Então, está colocando, aí, o QR Code. Respondam esse formulário de avaliação, duas perguntinhas simples, tá? Só para a gente ter noção do que a gente pode fazer para os próximos, o que a gente pode melhorar e como que a gente deve continuar.

Depois disso, eu vou pedir também, para o pessoal colocar no chat do YouTube, o link de inscrição para ganhar o certificado, tá? São duas coisas distintas. Então, se você quer ganhar o certificado de participação dessa live precisa se inscrever, e depois vai vir um e-mail na sua caixa de e-mail, e você precisa clicar no link. Clicou no link, você vai poder ter o certificado da live. E lembrando que esse processo vai até às 2 horas da tarde. Depois disso a gente fecha as inscrições, aí não dá para se inscrever mais.

Outros avisos que eu tenho que dar: a questão dos sorteios. O pessoal já está me alertando. Então, novamente, a gente tem dois sorteios ocorrendo aqui durante essa live. O primeiro da 4Linux, tá? Que é um curso EAD à escolha do ganhador. Então, está sendo colocado o link aí no chat. Então, quem quiser se inscreva. Daqui a pouco a gente já deve sortear a pessoa. E do Netfinders Brasil, que é um curso de BGP Multivendor em modo gravado. Então, novamente, está sendo colocado o link, precisa se inscrever.

Então, são dois sorteios, temos o formulário de avaliação e o certificado. Então, são esses avisos. Vamos aí, para a parte das perguntas. Moreiras, quer fazer a primeira pergunta?

SR. ANTONIO MARCOS MOREIRAS: Vou fazer, sim, Eduardo. E só reforçando o que o Eduardo já falou, é certeza que não vai dar tempo de fazer, assim, nem uma pequena parte das perguntas que vocês colocaram no chat, né? Porque a live vai até 12h30 e já são 12h13. Então, mas o que a gente pode falar para vocês é o seguinte, a gente vai voltar nesse assunto e as perguntas de vocês vão ser consideradas. Então, a gente está pensando, por exemplo, em fazer um 'websódio' do Camada8, que é nosso podcast, sobre esse assunto. Quem não conhece o Camada8, procura aí na Internet, procura na página do NIC.br, vocês vão ver um podcast muito legal. Então, a gente deve voltar com um 'websódio' e colocar esse assunto. E a gente deve produzir algum outro tipo de material. Não sei se vai ser artigo, se vai ser um workshop, se vai ser alguma coisa, mas a gente está vendo que tem bastante interesse, tem bastante perguntas que vão ficar sem responder. Então, não fiquem chateados, tenham certeza de que a

gente vai pegar todas essas dúvidas e vai produzir material para responder elas. Se não for tudo hoje, e a gente vai contar, inclusive, com a ajuda, espero eu, dos painelistas que estão aqui hoje, dos palestrantes que estão aqui hoje, para ajudar a gente a responder uma parte disso daí e colocar em um material organizado.

Então, eu vou fazer uma pergunta e vou chamar todos os painelistas na mesma ordem que eles falaram. Vou pedir para a resposta ser bastante curta. Eu estou juntando aqui a pergunta do Marcio William Ebuchi com a do Danton, com a do David Alain, e mais ou menos a ideia é a seguinte: são só os equipamentos modernos com suporte a SDN, com suporte a Netconf e Yang que podem ser usados nesse processo de automação de NetDevOps? Tem alguma alternativa para a gente usar equipamentos mais antigos ou que não tenham essas *features*? Sei lá, pega um roteador antigão ou pega um Mikrotik mesmo, tem alguma alternativa? E esses equipamentos que têm essas *features*, esses equipamentos mais modernos, como que é a penetração deles no mercado? Todos são assim hoje? Não? Tem só alguns? Estão disponíveis? Eles são mais caros do que equipamentos que não têm essas *features*? Então, José Miguel, você pode responder para a gente?

SR. JOSÉ MIGUEL GUZMÁN: Sim, poderia tentar. Não, na minha visão, tem aqui dois planos diferentes, no plano de controle, que é onde a SDN, talvez, seja mais conhecida. Aí, tem uma diversidade de equipamentos, não tanta penetração de equipamentos propriamente SDN, mas equipamento bem tradicionais, camada 2, camada 3 e MPLS. E no plano de gestão, que é onde a automação pode ter uma maior utilidade, aí, as tecnologias não têm mudado muito. Aí, você tem uma opção de fazer com Netconf, nos equipamentos de mais alta gama, você tem suporte de Netconf. Mas algumas vezes, inclusive, o Netconf não suporta todas as funcionalidades. Então, na experiência da gente, a gente continua fazendo muita automação com *command-line*. Aí, você tem ferramentas como Ansible, como Python, que tem muita livreria para interagir com o equipamento usando *command-line*, e também temos a opção de Netconf. Temos as duas. Tendo Netconf e tendo suporte de *command-line*, você consegue cobrir 100% da base instalada de equipamento novo e equipamento mais antigo.

SR. ANTONIO MARCOS MOREIRAS: Obrigado, José Miguel. Helio, o que você pode falar para a gente a respeito disso?

SR. HELIO BARROSO: Pois não. Bom, então, sim. Olha só, a gente tem, principalmente na América Latina, né? Eu conheço razoavelmente projetos em todo o continente, de automação, eu tenho participado. Na América Latina, os provedores médios e provedores grandes, eles têm muito *device* legado, né? O ritmo que se troca equipamento aqui na América Latina não é o mesmo de outras geografias, tá? Não tem aquele dinamismo dos americanos, ou mesmo

dos europeus, ou asiáticos. Então, a gente encontra muito, e por limitações, às vezes, não só de custo, até de logística ou cliente. Por exemplo, tem roteador que está em *backbone* aí, há coisa de 20 anos, por quê? Porque teve um cliente que comprou uma interface com HSI, que é uma interface que quase ninguém usa hoje. Ninguém usa hoje, tá? E o cara não quer migrar e está lá, o roteador modelo antigo, não tem mais suporte, não tem nada. Então, você vai se deparar nisso, você vai se deparar com muita frequência aqui.

Solução para isso, é o que mais ou menos o que o José falou, né? Você tem o modelo Yang, é o ideal que se busca, né, tem uma penetração grande. Por exemplo, depois eu vou falar em particular sobre a penetração, mas você pode complementar a sua automação, sim, com ferramentas como Ansible, como o Netmiko ou outras que você trabalha. Ferramenta da Cisco, que faz orquestração, ela tem também, um recurso de produzir *drivers* de forma automática e que até o próprio cliente usuário da ferramenta pode produzir *drivers* que compatibilizam o *netflow*... perdão, o Yang, o modelo Yang, modelado na ferramenta, com o *device* que não tenha o modelo Yang também. É uma opção que a gente oferece para quem adquire essa ferramenta aí, mais grandes e médios provedores.

Em relação à penetração, o que acontece? Eu acho que, assim, apesar de todos esses *devices* legados, os *devices* novos, todos que têm entrado, têm entrado com o suporte a Netconf e Yang, já há coisa de cinco, seis anos, então, a renovação é natural. E está se tomando... às vezes, você tem problema que você tem que fazer um *upgrade* do sistema operacional, mas não tem problema de conflito por hardware, vai ter que fazer o *upgrade* e você está com ele suportado.

Então, assim, tem uma penetração grande. E tem uma particularidade, né? Mesmo para *devices* pequenos, de fabricantes pequenos, tal, a concepção do Yang e do Netconf, são protocolos de modelo de dados bem *light weight*, tá? Ele não tem problemas de escalabilidade, ele não tem problema de ser implementado em *devices* de pequeno porte com baixa capacidade. Então, é um padrão que não opõe obstáculo à adoção de hardwares mais simples e de menor custo.

SR. ANTONIO MARCOS MOREIRAS: Obrigado, Helio. Elizandro, por favor.

SR. ELIZANDRO PACHECO: Opa. Bom, já nos provedores regionais, né, a gente tem, basicamente, muito Mikrotik, né? E, felizmente, hoje, bastante Huawei, bastante ZTE, bastante Cisco, né? E esses, né, Netconf e Yang, ok. E na questão do Mikrotik, que é uma maioria, eu diria assim, ainda hoje, né? Apesar de muitas vezes eu criticar alguns casos, ele tem uma coisa muito bacana que é a própria API, né? Então, esse problema, como eu mencionei anteriormente, dos *parsers* e tal, de texto, na API deles, isso praticamente zera esse problema, né? Porque ela tem um padrão bem definido de retorno, é

uma coisa que eles não mexem tanto. Então, esse tipo de integração facilita muito no Mikrotik por API. É bom os fabricantes nacionais, né, darem uma olhada melhor aí, Netconf e Yang, porque a galera está bem atrasada. Fora isso, a gente tem ZTE, Huawei, até mesmo lá no acesso. A ZTE, se não me engano, a partir da C600, ela já tem suporte também, a Netconf e Yang. Então, nos provedores, a gente tem uma penetração bastante elevada, né, de equipamentos com suporte a Netconf e Yang. E os que não têm, na grande maioria são Mikrotik que tem uma API própria, né? Que facilita muito a vida do desenvolvedor, né? Basicamente é isso.

SR. ANTONIO MARCOS MOREIRAS: Que legal, Elizandro. Foi uma resposta que me surpreendeu um pouco aqui. Eu não esperava um elogio ao API lá do Mikrotik. Mas é bom saber que, então, o pessoal dos provedores pequenos não tem muita desculpa aí, tem por onde conseguir fazer a automação, e com um suporte até adequado da caixa, embora seja proprietário, né? Renato, por favor, nos dê sua visão sobre o tema.

SR. RENATO ALMEIDA DE OLIVEIRA: Então, sobre utilização de equipamentos legado ou alguma coisa, a maioria dos frameworks APIs que existem, elas são *opensource*. E ser *opensource* já traz a facilidade de você ter toda uma comunidade envolvida desenvolvendo código. Então, lá quando eu havia dito em questão de abstração, no pior cenário, você tem, por exemplo, Paramiko, que você consegue se conectar via SSH e já fazer algumas trocas de comandos. Então, basicamente, qualquer equipamento que você tenha uma célula(F) exposta, que você tenha um acesso SSH ou um acesso Telnet, você consegue, em algum nível, fazer automação. Agora, quando a gente soube um pouquinho para começar a falar em termos de modelo de dados e em fazer *parsers* de configuração, existe, de fato... ou, no caso, por exemplo, do Genie, ele, existe a possibilidade de você contribuir ou de outras pessoas contribuírem para adicionar novos equipamentos a essa base. Então, apesar de você, talvez, não ter um framework ideal, maravilhoso, para você trabalhar hoje, talvez, com o passar do tempo, você ainda consiga receber um suporte CLI(F) para o seu equipamento. Então, isso não é algo que está estático. E os equipamentos legados, eles não vão deixar de existir, então, existe a demanda para automação de outros tipos de equipamento.

Então, a parte em relação à penetração de equipamento, eu não posso falar muito, como trabalho com administração pública, a gente percebe aquilo que a gente comprou por licitação. Então, isso é, de fato, uma dificuldade nossa, de a gente ter que saber trabalhar e conseguir trabalhar com qualquer tipo de equipamento que um fornecedor que receba uma licitação entregue. Então, a gente acaba tendo esse um cenário *multivendor*.

SR. ANTONIO MARCOS MOREIRAS: Muito bom, Renato. Muito obrigado. Você, inclusive, respondeu outras perguntas que tinham sido feitas no chat, especificamente sobre acesso SSH e outro tipo de coisa com a tua resposta. Leopoldo, por favor, você pode nos dar aí, a sua visão, sua experiência nisso? Olha o áudio, Leopoldo. Você está sem áudio.

SR. LEOPOLDO MAURICIO: Tirei do mudo, desculpa. Estão me ouvindo? Alô? Ouvindo?

SR. ANTONIO MARCOS MOREIRAS: Sim, sim, agora está legal, pode falar.

SR. LEOPOLDO MAURICIO: Ok. É, você, independente do equipamento, você pode, sim, fazer automação de configuração dele, não necessariamente usando Netconf e Yang, seja ele suportando ou não, você tem como fazer isso, como fazer a automação usando ferramentas como as que foram mostradas pelo menino, eu não me lembro o nome, do Exército brasileiro. Eu estou colando aqui, olha, no chat do evento lá no YouTube, o nosso link no GitHub, que tem, por exemplo, a Network API. A Network API que orquestra todos os equipamentos de rede da globo.com, a antiga globo.com, né, do nosso data center. Ela é código aberto, tá? E se você for lá no GitHub, você verá que a gente tem orquestração de diferentes tipos de equipamentos, se você mergulhar um pouco no código, diferentes tipos de fabricantes sendo providos pela Network API, porque ela abstrai isso. A gente tem uma API de gerenciamento de rede que abstrai o fabricante e o tipo de tecnologia que o fabricante suporta.

Então, a gente, tem desde Juniper, Cisco, Dell, HP, qualquer que seja o equipamento, a Network API fala com esses equipamentos, 'deploya' eles. Então, a gente tem um livro de orquestração que permite que você só ligue o equipamento na tomada, ponha um cabo de rede nele, preenche um formulário, aperta um botão, ele é instalado automaticamente, configurado automaticamente para nascer em um ambiente, por exemplo. E aí se você mergulha no código, você vê que não existe necessariamente essa obrigatoriedade do equipamento ter suporte a Netconf e Yang. É bem-vindo, mas, como o menino do exército mostrou, você tem, na sua API, diferentes possibilidades de interface com o equipamento. E eu convido todo mundo a ir lá no GitHub da Globo, eu coloquei o link lá. O GitHub da Globo tem muito exemplo de código que pode ser útil para quem está começando, para quem tem alguma dúvida relacionada a isso.

SR. ANTONIO MARCOS MOREIRAS: Obrigado, Leopoldo. Leopoldo, cola para a gente no chat interno o link, porque eu acho que o YouTube bloqueia quando não é o, vamos dizer assim, o dono da sala. Daí, a nossa equipe vai colar lá o link do GitHub. Então, cola aqui para a gente.

E, Fabio, aí, encerrando essa primeira rodada de perguntas, espero eu com chave de ouro, né? Qual é a experiência aqui no IX.br? Todos os equipamentos têm Netconf, Yang e SDN? Como que é isso? Por favor.

SR. FABIO PESSOA NUNES: Não, na verdade, assim, a gente tem bastante equipamento legado ainda, que não suportam Netconf e Yang. Mas, assim, acho que a maioria, boa parte do equipamento, acho que, no nosso caso, todos os equipamentos, pelo menos Ansible a gente consegue utilizar. Então, acho que é o básico aí, para a gente conseguir fazer algum tipo de automação. Mas eu acho que até naqueles casos que realmente não têm nenhum módulo para o Ansible, acho que sempre sobra a gente conseguir fazer algum tipo de automação, algum tipo de verificação utilizando um *expect*, por exemplo, né? Então, eu acho que alguma coisa a gente consegue fazer. Mas eu acho que, assim, eu nunca... apesar de a gente ter que pensar na rede como um todo, eu volto ao ponto que o mínimo que a gente conseguir fazer de automação já vai ajudar muito a gente no nosso dia a dia. Então, comece, pelo menos, por aqueles pontos na rede que realmente você tem uma facilidade maior. Se você está em um processo inicial de automação, começa naqueles pontos mais simples que têm uma ferramenta já disponível, que facilita o seu dia a dia e o seu processo de aprendizagem.

SR. EDUARDO BARASAL MORALES: Obrigado, Fabio. Vamos para a próxima pergunta, também, a gente juntou de diversas pessoas que falaram coisas muito parecidas. Então, Silvio, Teodoro(F), João Paulo. Como se tornar um profissional habilitado a atuar nessa área? Que tipos de cursos procurar? Onde procurar mais informações? Vocês têm recomendações de livros, materiais ou cursos específicos? Agora eu vou fazer essa pergunta para a mesa, mas eu queria que todo mundo fosse breve nessa resposta, porque é coisa simples, né? Se tem alguma recomendação de livro, um material, curso, para as pessoas poderem se especializar. E eu também vou chamar na ordem os palestrantes, então, primeiro, José, você gostaria de comentar? José Guzmán.

SR. JOSÉ MIGUEL GUZMÁN: Estou com problema de áudio, você poderia repetir para mim, por favor?

SR. EDUARDO BARASAL MORALES: Claro, José. Pergunta ali: Que recomendação de livro, material ou curso para as pessoas se especializarem nessas automações, SDN, Netconf e Yang, você tem alguma recomendação para um profissional começar essa carreira?

SR. JOSÉ MIGUEL GUZMÁN: Sim. Mais que livro, eu acho que há dois conceitos mais básicos que qualquer engenheiro de rede deveria tentar aprender, são duas coisas bem básicas: Linux. Se você não sabe Linux, se você conhece *command-line* dos equipamentos, mas não conhece ainda Linux, esse é um bom investimento. Você pode

aprender Linux on-line, tem muito... inclusive em português, tem muito treinamento aí disponível, e também tem classes formais e tem livros. Mas o melhor é a experiência, é jogar com o Linux. Segunda coisa: Python. Quando você conhece Linux e Python, qualquer das outras ferramentas já fica muito mais perto. Daí, você pode pular para Ansible, para Chef, para Puppet, para qualquer outra. Se você não consegue ter um bom conhecimento de Linux e de Python, fica um pouco mais difícil. Então, eu diria que Linux e Python é um bom ponto de partida.

SR. EDUARDO BARASAL MORALES: Obrigado, José. Hélio, gostaria de complementar?

SR. HELIO BARROSO: Opa. Seguinte, primeira coisa. Desculpa aí. Tem muita coisa nova para aprender, né? Mas não pare de prestar atenção na evolução das redes, da tecnologia de rede. Protocolos e tudo. Elas vão ficar... elas não estão sendo substituídas, elas estão sendo renovadas. Muita coisa vai ser abandonada, mas isso é normal. A simples história da rede, mais de 30 anos, sempre foi assim, né? O número de protocolos que foram abandonados, o número de protocolos que foram surgindo é grande.

Então, assim, você continua tendo que ser um especialista de rede. Isso é uma recomendação para quem vai entrar e é um toque para quem está. Porque, às vezes, muita gente acha que está mudando para a carreira de desenvolvimento, e não é isso, não é? Você está somando *skills*, conhecimentos e habilidades de desenvolvedores para aplicar em rede, ou extensivamente em Cloud, tá? Agora, para se familiarizar, a partir daí, da base que você tem que ter de rede, para você estender para esse mundo de SDN, programabilidade e automação, o José já citou Python e Linux, são fundamentais. Eu acrescentaria Cloud, tá? Você tem que ter conhecimento de Cloud também, porque tem muita integração, principalmente em ambiente corporativo e também *service provider*. Tem muita interação, Cloud híbrida, integrações e tal, que você vai ter que lidar, tá? Isso é importantíssimo, tá?

Ferramentas, outras ferramentas de framework, assim como Ansible, Terraform, também são necessárias. Fonte para... E também, eu destaco assim, eu citei aqui, e parece futurismo, né? Futurismo, não, é cenário de curto prazo: inteligência artificial e *analytics* vão começar a entrar cada vez mais forte, já estão entrando. Tem operadoras na América Latina que estão baseando a sua operação para correção de problemas, antecipação de problemas, planejamento, estão se apoiando pesadamente nesse tipo de conhecimento. Você não precisa virar um engenheiro de Inteligência Artificial ou um *data scientist*, mas é bom ir adquirindo alguns conceitos que você vai aplicar no dia a dia.

Como fonte de conhecimento, eu vou... tem muito livro, tem os livros da O'Reilly que têm muito conteúdo sobre rede, e os livros da Cisco. Eu posso postar as recomendações. Eu queria destacar a nossa contribuição aberta para a comunidade, que é a DevNet, né? O site é DevNet, Cisco DevNet, tem aí, cobre várias áreas. Eu passo o link para vocês. Tem muito conteúdo gratuito, treinamentos, tem laboratório Sandbox para você praticar. Esse aí, ele é muito utilizado pela comunidade Cisco e também pela comunidade *opensource*. Você pode, não só praticar em cima de produtos e ferramentas Cisco, mas também com *opensource*, tá? Todo esse conteúdo que a gente abordou aqui, com certeza você vai encontrar lá gratuitamente.

A Cisco também tem cursos da Cisco, da parte de *learning*, né, de treinamento, que também te orienta. E tem a formação DevNet Associate, Professional, tal, parecida com as certificações da Cisco. Vai vir um tipo de CCE(F) de DevNet agora também, aí, à frente, que vocês poderão, em tempo breve, estar utilizando. Hoje já é realidade, tá? É isso.

SR. EDUARDO BARASAL MORALES: Obrigado, Helio. Então, seguindo aí, vamos chamar agora o Elizandro. Vamos, Elizandro, pode falar do seu livro, pode falar dos seus cursos.

SR. ELIZANDRO PACHECO: Beleza. Então, bom, não tem como não recomendar a Network Education, né? Que é a nossa escola focada em provedores regionais, né? Desde 2015 aí, atuando especificamente com esse ramo. A gente tem, inclusive, né, treinamento de Docker, que é o que eu acrescentaria à parte do Linux, obviamente, é a base, como já foi falado, e do Python, até para que a gente comece já a organizar e arquitetar o nosso ambiente de desenvolvimento de maneira certa, né? Então, acho que o Docker vem aí para somar nesse ambiente, porque é um dos grandes problemas, né? Saber programar é uma parte da coisa, né? A arquitetura do software em si, geralmente, ela é o que faz os softwares, ali na frente, sucumbir, né? Então, além de Linux e Python, eu começaria, em um próximo passo, por Docker também, em função do ambiente de desenvolvimento, né? Para depois partir, aí, para outros tópicos como os que eu coloquei lá no slide.

Sobre desenvolvimento, eu, particularmente, quando vou testar alguma coisa, eu procuro a documentação oficial, geralmente tem lá o Comece Aqui, e aquilo ali te dá um bom embasamento, ao menos para ti entender os conceitos e entender a própria documentação. A partir daí, tem muito material na Internet hoje em dia, né? Não tem fonte maior de informação, certo?

SR. EDUARDO BARASAL MORALES: Obrigado, Elizandro. Bom, seguindo aí, Renato, gostaria de complementar?

SR. RENATO ALMEIDA DE OLIVEIRA: Sim, só trazer o ponto que sempre quando a gente fala em desenvolvimento, hoje, desenvolvimento, ele é muito fortemente levado pela comunidade.

Então, estar envolvido com alguma comunidade que troca informação de software, por exemplo, o Twitter tem vários desenvolvedores que estão trocando diversos códigos, informações no Twitter. Tem comunidades dentro do próprio Webex da Cisco, que eles disponibilizam, por exemplo, a comunidade do PyATS. A comunidade do PyATS, ela tem próprios os desenvolvedores envolvidos na própria comunidade. Tem a comunidade do DevNet, tanto dentro da linha Cisco. Existem diversos canais em Discord, que você pode acessar e conseguir trocar informação. Vou publicar uns dois aqui no chat, uns dois links, onde um dos desenvolvedores do Twitter, que ele fez uma consolidação de URLs e links que você consegue acessar diversos conteúdos de diversos... pessoas que estão postando, publicando conteúdo na área de desenvolvimento para rede.

E, de novo, reforçar que Python, hoje, a parte de automação [ininteligível] está muito fortemente para Python. Então, saber Python é muito importante. E, além de saber Python, também é interessante você compreender um pouco de como funciona algumas arquiteturas de HTTP, então, por exemplo, compreender como funciona o protocolo Rest, entender como funciona Soap. Porque esse tipo de arquitetura, ela é aproveitada em praticamente todas APIs e tudo. Então, você conhecer isso vai te dar uma flexibilidade de você estar trabalhando com diversos tipos de aplicação diferente. Reforçando o que eu havia dito, hoje você trabalha com ecossistema, você não tem apenas a sua aplicação trabalhando. Você tem a sua aplicação, você tem NetBox, você tem uma aplicação de SDN, você tem diversas aplicações trabalhando junto, provavelmente todas utilizam protocolos Rest ou Soap para comunicar entre si. Então, isso é bem importante de se conhecer e estudar.

SR. EDUARDO BARASAL MORALES: Obrigado, Renato. Vamos agora a seguir. Leopoldo, gostaria de complementar? Leopoldo, você está no *mute*.

SR. LEOPOLDO MAURICIO: Opa, saí do *mute*. Minha vez? É, eu acredito que hoje... eu vou pegar um gancho na primeira fala. Sim, existe um mercado de *network engineer*, o cara que vai ser o analista de redes, mas o analista de redes é um pouquinho diferente. Precisa ser, na minha opinião, um pouquinho diferente do analista de redes de 10, 15 anos atrás. Ru acho que o analista de redes hoje precisa, além de estudar questões relacionadas à rede, como certificações que são muito famosas aí, pessoas que estudam certificações de Cisco, Juniper, entre outras. Eu acho que ele precisa desenvolver. E aí, a linguagem hoje, que muita gente... que é uma das mais famosas hoje para desenvolvimento, é o Python. Por isso muita gente falou do Python aqui. Mas tem o GO, né? Tem outras linguagens que você também pode mergulhar para ser capaz de programar dispositivos de rede, automatizar a configuração dos dispositivos de rede. Se você não conhecer os protocolos, como foi dito antes, não adianta você só ter o

viés de desenvolvimento, né? Porque você vai programar e configurar o que se você não conhece os protocolos que você está configurando, né? Não sei nada de BGP, nada de OSPF, não sei nada dos protocolos de rede especificamente, como é que eu vou comandar e controlar o meu equipamento? Então, precisa, sim, desse viés de *networking*, de conhecimento de protocolos de rede, enfim. E aí, sim, e junto a isso, se eu vou trabalhar com redes, eu preciso ser também, minimamente, um desenvolvedor. Hoje, ser um analista de rede sem ter um viés de desenvolvimento, eu acho que você fica com *gap* muito grande, sabe? Eu acho que é uma ideia muito ruim.

SR. EDUARDO BARASAL MORALES: Obrigado, Leopoldo. Fabio, você, para fechar a nossa discussão dessa pergunta. Fique à vontade.

SR. FABIO PESSOA NUNES: Bom, na verdade, não tem muito o que acrescentar, o que o pessoal falou é realmente o meu ponto de vista. Todos esses... acho que esses são os novos focos mesmo do profissional de rede. Então, é muito importante que a gente, todos esses... que a gente foque um pouco em aprender um pouco todos esses tópicos, né? Então, assim, eu vou deixar mais uma recomendação de livro, na verdade. Não sei nem se dá para ver. Eu vou postar para o pessoal, depois, para mandar lá no chat, tá? Mas esse livro aqui é um livro muito bom para entender do começo mesmo, ele dá alguns exemplos práticos de como usar mesmo Yang, Netconf, tá? E esse aqui também, que dá um *overview* muito bom de toda a parte de programabilidade e automação, tá? Então, vou passar o link lá para o pessoal pôr no chat.

SR. ANTONIO MARCOS MOREIRAS: Legal, muito obrigado. Eu gostaria de agradecer a todos os painelistas, né, pela participação. Agradecer também todo mundo que participou aí, acompanhando, comentando no *chat* e tudo mais. O Eduardo daqui a pouquinho volta com o resultado dos sorteios.

E eu tenho só alguns avisos. A gente está nos finalmente aqui, temos 530 pessoas ainda, assistindo ao vivo agora, um pouquinho mais do que 530 pessoas. Corajosos. Praticamente fazendo protocolos de jejum intermitente aí, segurando um pouquinho a fome do almoço para terminar a live. E estamos já no finalzinho, gente, é bem rápido.

Então, o primeiro aviso, reforçando, temos a avaliação da live, o formulário de avaliação da live. Gostaria de pedir para o pessoal responsável pelo chat, que cole novamente o formulário de avaliação. Temos 540 pessoas assistindo agora, eu preciso de pelo menos 540 respostas nesse formulário. Se possível, um pouquinho mais, para quem for ver o vídeo depois, né? É muito importante para a gente. É um negócio que vocês não vão gastar um minuto para responder. É uma nota de zero a dez que vocês vão dar, do nível de satisfação de

vocês, e um comentário sobre o que a gente tem que melhorar para a próxima live, tá? É só isso. Então, por favor, respondam para a gente.

Eu gostaria de falar também, sobre os projetos, sobre cursos e projetos que a gente tem no Ceptro. A gente tem o curso de boas práticas operacionais para Sistemas Autônomos. Temos duas turmas com inscrições abertas, uma fecha hoje. A próxima turma vai ser de 22 a 26 de março, e a turma seguinte vai ser de 5 a 9 de abril. Ambas as turmas estão com inscrições abertas. A turma de março fecha as inscrições hoje. Então, quem tiver interessado em fazer o curso, vê qual é a melhor data, se inscreve, preenche direitinho o formulário de inscrição. Coloca as informações lá, o minicurrículo, com em que área da empresa trabalha, porque tem uma espécie de um pequeno processo de seleção, tem vários critérios que a gente usa, né? Normalmente, a gente tem muito mais inscrições do que vagas, embora cada turma dessas tenha hoje 80 vagas. E, se por acaso, cara, você se inscrever, for selecionado e não puder fazer o curso, desista lá, assim, formalmente. Tem um link para você cancelar a inscrição, porque outra pessoa é chamada no lugar. Então, não é porque é um curso gratuito que você vai se inscrever e ficar lá, e não fazer simplesmente, né? E, por incrível que pareça, às vezes, tem gente que faz isso daí e a gente acaba tendo menos gente na turma do que a gente poderia atender bem. E isso é uma tristeza para a gente, porque é, de uma certa forma, um desperdício de recursos e tem muita gente querendo fazer o curso, né? Enfim, as inscrições estão abertas, façam lá as suas inscrições. A gente também tem o IPv6 EAD, o IPv6 a distância. Esse é um curso autoinstrucional, você vai fazer lá no seu ritmo, sozinho, com as perguntas no fórum. E esse, as inscrições estão abertas sempre. É um curso bastante bom, ele usa o livro Laboratório de IPv6, tem experimentos práticos, dá para você aprender muita coisa fazendo esse curso de IPv6 básico à distância.

Eu já citei várias vezes o Camada8, que é o nosso podcast. Semana que vem sai um 'websódio' novo, vai ser imperdível. É essa semana, Eduardo? O Eduardo está me falando que é essa semana aqui, eu estou perdido. Depois o Eduardo me corrige aqui, na hora que ele tomar a palavra, ele fala direitinho do próximo 'webisódio' do Camada8. Então, é essa semana, vai sair o 'webisódio' do Camada8 e eu estava achando que era só semana que vem.

E vamos ter... essa é a primeira live do Intra Rede, temos N lives programadas, várias lives do Intra Rede programadas até o final do ano. Inclusive, temos oportunidades de patrocínio para essas lives. O patrocínio não é muito caro. E vejam aqui, tivemos picos aqui de 700, mais de 700 pessoas assistindo ao vivo hoje. É um evento que tem sido... a gente tem caprichado na produção desses eventos, a nossa equipe, o pessoal de comunicação do NIC.br também, sempre nos ajudando. E vocês veem que tem sido eventos de qualidade, onde a gente traz muito conteúdo interessante, e essas oportunidades

também, de sorteios, são bastante interessantes para os patrocinadores. Então, o pessoal que é de empresas, de fabricantes, de fornecedores aí, que tem uma atuação mais nacional, dá uma olhada lá com carinho também, no nosso site, nas propostas de patrocínio, porque foi só uma live, ainda tem muitas pela frente. As propostas de patrocínio também abrangem os cursos BCOP, e também tem uma proposta de patrocínio para a Semana de Capacitação, que é uma iniciativa irmã dessa daqui, vamos dizer assim.

Por fim, eu quero falar dos vídeos do projeto Cidadão na Rede. Nós já tocamos um pequeno... demos *play* aí em um pequeno videozinho de 15 segundos no começo da live. E eu vou pedir para o Pedro dar um *play* em um segundo vídeo agora, no finalzinho da live.

Então, Pedro, quando você puder, por favor, solta o videozinho aí.

[exibição de vídeo]

SR. ANTONIO MARCOS MOREIRAS: Bom, então, vocês puderam ver o vídeo. No começo da live vocês viram o vídeo falando que rede cabeada é boa opção de conexão para o usuário doméstico, talvez uma conexão preferível sobre outros tipos de rede, porque não tem interferência, porque tem uma velocidade maior. E agora tem o vídeo de: "Ah, minha Internet parou, e agora o que eu faço?", com algumas dicas ali, né? Então, tem vários tipos de dicas, esses vídeos. Algumas dicas sobre infraestrutura, algumas dicas sobre segurança, algumas dicas sobre comportamento, direitos e deveres, vamos dizer assim, na Internet. São vídeos sobre cidadania digital, sobre como usar a Internet de forma plena. E a gente acha que muitos desses vídeos aí, podem ser úteis para os usuários de vocês que são provedores de acesso, para os usuários de vocês que são administradores de rede, administradores de sistema e estão cuidando de uma rede corporativa.

Talvez alguns desses vídeos possam tirar dúvidas dos usuários de vocês, e tirar a pressão aí, do callcenter, tirar a pressão do suporte. Então, a gente quer que vocês olhem o site do Cidadão na Rede, cidadonarede.nic.br, com bastante carinho, e vejam se não é interessante para vocês distribuírem esses vídeos para os seus usuários. Na verdade, a gente está contando com essa parceria de vocês. Tanto que a gente criou uma forma de vocês customizarem o vídeo com o seu logotipo. Então, é possível você assinar on-line, e isso está funcionando já a partir de hoje, teve bastante gente que já ficou interessada, mandou e-mail para a gente. Hoje o site já tem uma área onde você consegue cadastrar o seu usuário, cadastrar o CNPJ da sua empresa, subir o logotipo com fundo transparente, o logotipo todo em branco. E ele vai ficar naquela barrinha que aparece na parte de baixo do vídeo, como apoiador. Então, tem um processo semiautomatizado, você vai assinar um contrato on-line lá, tem algumas condições que se

aplicam. O pessoal da equipe depois vai olhar se o logo está dentro do esperado e vai dar um ok manual lá. Mas toda a customização dos vídeos, produção dos vídeos, você pode baixar esses vídeos customizados e subir no seu canal no YouTube, no seu LinkedIn, na sua página, mandar pela sua lista de WhatsApp ou individualmente para os seus clientes, da forma que você quiser, nas suas redes sociais, nas suas mídias. Então, você vai poder baixar o vídeo com o seu logotipo e divulgar ele nas suas mídias. E isso é bastante interessante, eu acredito, e espero que vocês concordem comigo. Deem uma olhada lá. Já temos mais de 20 vídeos produzidos e temos o compromisso de produzir um vídeo novo por semana. Então, terá bastante material aí, para divulgação, até o final do ano.

Bom, se tem mais algum aviso aqui, eu vou pedir para o Eduardo fazer, se eu esqueci de alguma coisa. Eduardo, você assume aí, agora, e dá o resultado dos sorteios, e segue com os avisos e encerramento da live?

SR. EDUARDO BARASAL MORALES: Sim, Moreiras. Bom, acho que uma das coisas que faltou você comentar, né, do Camada8, é que o episódio, não é que ele sai essa semana só, ele sai hoje. Na verdade, já está no ar, tá? Então, se vocês quiserem, vão lá ver o novo episódio do Camada8, que a gente faz uma discussão com uma pessoa que vai trazer uma visão de como que ele aprendeu redes e como ele desenvolveu tudo isso para melhorar a instituição dele. E aos pontos que ele está chegando com a implementação de redes e todas as melhorias que ele aprendeu. Então, é um episódio, assim, bem interessante.

Também o Intra Rede, tá? O Intra Rede é falar do próximo episódio do Intra Rede, né? A gente vai falar sobre a camada física. Então, não percam essa próxima, vai ser no dia 31/3. Então, ou seja, nesse mês ainda, no último dia do mês, que a gente vai discutir sobre 5G, cabos submarinos, cabos fluviais, conexões via rádio, conexões via satélite, fibras óticas. Então, a gente quer dar um panorama de como que está o andamento da camada física, né, atualmente. Como que elas estão evoluindo e como que vocês podem ficar atualizados nesse assunto. Então, última live aí, desse mês, que vai ser no dia 31, não percam.

Por fim, falar do sorteio, né? Já temos aí, os ganhadores. Então, o ganhador da 4Linux é o Anderson. Vai ser enviado um e-mail. Ele simplesmente só cadastrou como Anderson. Então, vai ser enviado um e-mail para você, e aí, você vai poder escolher o curso da 4Linux. E do Netfinders Brasil é o Fabrício de Souza(F). Então, vai ganhar lá o curso BGP Multivendor. Então, esses são os resultados do nosso sorteio.

Por fim, eu queria agradecer os patrocinadores, Juni Link IP and Cloud Network by Giovaneli consultoria, WZTech Networks, ICANN, Netfinders Brasil, Novatec Editora, Eletronet, Globenet Telecom,

Mundivox, 4Linux, Solintel e apoio de mídia da revista RTI e Infra News Telecom. Então, aí, já fica os nossos agradecimentos aos patrocinadores.

Mais os agradecimentos aos palestrantes, que foram ótimos aí, o José Miguel Guzmán, da Whitestack, o Helio Barroso, da Cisco, o Elizandro Pacheco, da NextHop Solutions, Renato Almeida de Oliveira, do CITEx, do Exército, o Leopoldo Mauricio, da Globo, e o Fabio Pessoa Nunes, do NIC.br. Então, nossos agradecimentos aos palestrantes, aos nossos ouvintes, aos nossos telespectadores aí, e a nossa equipe do NIC.br que nos serviu de apoio. Então, agradecimentos a todos. Nos vemos aí nos próximos Intra Redes. Então, até mais, pessoal.