RECIT
Revista Eletrônica Científica Inovação e Tecnologia

Revista Eletrônica Científica Inovação e Tecnologia
ISSN: 2175-1846

https://periodicos.utfpr.edu.br/recit

REDES DEFINIDAS POR SOFTWARE: UM ESTUDO SISTÊMICO SOBRE AS NOVAS ABORDAGENS NO CONTEXTO DA IOT

Danielle Costa de Oliveira,
danielle.costa@ifmg.edu.br
http://orcid.org/0000-0002-2858-1324
IFMG-Instituto Federal de Minas Gerais, Formiga, MG - BRASIL.

Kimberly Lamounier Campos Ferreira,
kimberlylcf@gmail.com
http://orcid.org/0000-0001-8330-8104
IFMG-Instituto Federal de Minas Gerais, Formiga, MG - BRASIL.

Mario Luiz Rodrigues Oliveira
mario.luiz@ifmg.edu.br
http://orcid.org/0000-0002-3048-4451
IFMG-Instituto Federal de Minas Gerais, Formiga, MG - BRASIL.

RESUMO

As Redes Definidas por *Software* (Software Defined Networks, ou SDN) constituem um paradigma para o desenvolvimento de pesquisas em redes de computadores que vem ganhando a atenção de grande parte da comunidade acadêmica e da indústria. No entanto, essas redes ainda apresentam muitos desafios operacionais e de pesquisa. Diante desse cenário, o presente trabalho investiga os componentes de um sistema de rede definida por software com qual obteve como principal resultado, a apresentação de uma visão geral sobre novas abordagens e a identificação dos principais desafios de pesquisas em andamento que podem contribuir para o avanço das SDN.

PALAVRAS-CHAVE: Internet das Coisas. Computação na Nuvem. Redes Definidas por Software.

INTRODUÇÃO

O paradigma da Internet das Coisas (Internet of Things - IoT) durante a última década ganhou significativa atenção acadêmica assim como da indústria. Entre as principais razões por trás desse interesse está a capacidade da IoT pronunciar o desenvolvimento de objetos físicos que podem ser conectados à Internet e que se comunicam com o mínimo de intervenção humana possível (ATZORI et al., 2010).

Esses objetos, também chamados de objetos inteligentes, podem ser conectados com plataformas e aplicativos, trocar informações entre si e através da Internet**,** bem como tomar decisões automaticamente de acordo com sua programação. Por se aplicarem a várias funcionalidades, a quantidade desses objetos vem aumentado com o passar dos anos e o resultado, é uma enorme quantidade de dados sendo gerados de forma constante por eles (PERERA et al., 2013; FIRJAN, 2016; DAVE, 2011).

Para permitir o armazenamento virtual das informações que os dispositivos conectados produzem, a computação na nuvem vem sendo empregada como solução de forma que um objeto conectado possa ter acesso rápido às informações tanto geradas por ele, quanto por outros objetos vinculados a Internet, o que os possibilita trabalharem em modo colaborativo.

A computação na nuvem é um dos principais meios de serviço, infraestrutura, plataforma de *software* e análise de dados, conhecidos para a Internet das Coisas (OCAMPOS, 2015). É por meio dela que as aplicações podem acessar dados de forma compartilhada, sob demanda imediata e com o mínimo esforço de interação com o provedor de serviços. Tem como proposta uma alta confiabilidade, autonomia nos acessos a nuvem, escalabilidade e recursos mais dinâmicos, já visando sobretudo**,** o futuro da Internet das Coisas (MELL; GRANCE, 2011; VASHI et al., 2017).

As requisições de recursos em nuvem estão acontecendo de forma cada vez mais abundante e a arquitetura estática de rede de computadores atual é pouco flexível para a demanda. Diante desse cenário as Redes Definidas por *Software* (Software Defined Networks - SDN) surgem como uma proposta para ativar arquiteturas de nuvem, disponibilizando distribuição e mobilidade de aplicativos de forma automatizada, sob demanda e em grande escala. As SDN podem aprimorar os benefícios da virtualização de um *data center*, aumentar a flexibilidade, a utilização de recursos e reduzir custos e sobrecargas de infraestrutura (GUEDES et al., 2012).

Segundo Sakir (2013), uma rede que é definida por *software* permite uma ótima comunicação em todos os níveis da rede, desde a camada física até a aplicação para os usuários finais, oportunizando o desenvolvimento de novas aplicações através da melhor utilização de recursos oferecida pela rede. Como seu comportamento é amplo e centralizado, é mais simples que o modelo de rede de computadores tradicionais em que um fluxo, uma vez sido definido, a única maneira de realizar uma mudança é reconfigurando os dispositivos, o que torna o modelo restritivo e não escalável sob uma demanda. Em uma rede definida por *software* pode-se realizar modificações na configuração por meio de *software*, trazendo um conhecimento geral de todo o tráfego e os estados da topologia.

As organizações científicas, acadêmicas e industriais estão experimentando a tecnologia SDN e trabalhando para quantificar quando ela é aplicada ou implantada em diferentes contextos. Os benefícios vão desde menores despesas

de capital e operacionais até a criação de novas aplicações inovadoras, aproveitando a capacidade de programação da rede (MONGA, 2013).

O presente trabalho visa contribuir para o estado da arte com um referencial teórico sobre as novas abordagens para Redes Definidas por *Software* que estão sendo utilizadas para viabilizar soluções em nuvem, bem como levantar os principais desafios em aberto e que são fonte para novas pesquisas. Para tanto foram selecionadas as pesquisas conduzidas na última década. Pretende-se com isso sublinhar tecnologias, características e lacunas de pesquisas sobre SDN que podem ajudar a compreender os movimentos históricos e a direção em que a tecnologia está se movendo hoje.

Este artigo está organizado em seções da seguinte forma: na seção 2, subseção 2.1 e 2.2 é apresentada a teoria de base para a compreensão das abordagens tradicional estática e de rede definida por software. Na subseção 2.3 é apresentado um comparativo das abordagens. Na subseção 2.4 são identificadas as soluções que estão sendo utilizadas. E, finalmente, na subseção 2.5 são apresentados os principais desafios para a implantação das SDN.

ESTADO DA ARTE

Esta seção apresenta o sistema de uma rede definida por *software* considerando suas principais características. Antes, a abordagem tradicional é caracterizada. Em seguida, as soluções usadas para a integração da rede IoT com SDN e os desafios para a implantação no cenário tecnológico atual também são identificados.

2.1 Abordagem tradicional

As redes de computadores convencionais são tipicamente formadas por grande número de dispositivos físicos, como roteadores e *switches*, e por vários protocolos que gerenciam o tráfego para transportar, distribuir e encaminhar informações por todo o mundo (KREUTZ et al., 2015).

As políticas que abrangem eventos e aplicativos de rede são configuradas pelos operadores de rede que necessitam transformar políticas de alto nível orientadas ao negócio, manualmente, através de comandos de configuração de baixo nível orientadas a tecnologia, enquanto se adaptam as mudanças das condições da rede. Essa é uma implementação vertical que agrupa o plano de controle com o plano de dados (NUNES et al., 2014).

O plano de controle fornece as funcionalidades de gerenciamento, como configurações e monitoramento, com implementação em *software*, e o plano de dados oferece as funcionalidades necessárias para processar pacotes, suas entradas e saídas, seu mecanismo de encaminhamento, incluindo o *hardware* da interface de rede no processo e tratamento em tempo real (COMER, 2016).

Para adicionar à problemática, nas redes que usam IP (Internet Protocol) o processo de configurar a rede de acordo com políticas pré-definidas e reconfigurar quando elas apresentam falhas, é complexo, uma vez que não existe nenhum indicador apontando para onde a falha possa estar, ou mudanças podem ocorrer.

A tarefa de transformação de políticas se torna ainda mais complexa de se realizar, pois as ferramentas disponíveis são limitadas, já que um fabricante pode fornecer características em algumas interfaces e não em outras a fim de diferenciar

seus produtos, permitindo a compatibilidade entre seus dispositivos, mas entretanto, dificultando a comunicação quando o dispositivo é fabricado por um concorrente. O resultado disso é um complexo gerenciamento que dificulta o monitoramento e acompanhamento do funcionamento da rede que inclui ainda a perda de conexões, disponibilidade de serviços, além de deixar o ajuste de desempenho propenso a erros (NUNES et al. 2014; COMER, 2016).

Também na arquitetura das redes convencionais, a abstração se realiza em camadas, lidando apenas com os planos de dados e não há fraca abstração no plano de controle, sendo esse o que fornece funcionalidades de gerenciamento, como configurações e monitoramento, com implementação em *software*, que se tornam lentos em comparação ao *hardware* de interface de rede, que são otimizados (PINHEIRO, 2013; COMER, 2016).

Segundo Passos, *et al.* (2016), a arquitetura das redes convencionais não suporta as necessidades de computação e armazenamento dinâmicas que exigem uma gestão de tráfego mais flexível, devido ao acesso sob demanda nas nuvens que normalmente são distribuídas geograficamente em vários países, requerendo um processamento paralelo massivo e com uma conectividade constante.

2.2 Abordagem SDN

Os equipamentos usados nas redes de computadores, como *switches* e roteadores, tornaram-se "caixas pretas" devido as implementações dos dispositivos serem baseadas em *softwares* fechados sobre o *hardware* proprietário, impossibilitando a reprogramação deles. Isso leva a arquitetura da rede convencional para um alto nível de complexidade que tem impacto direto no seu funcionamento correto (MOREIRA et al., 2009).

A infraestrutura da rede, assim como seus protocolos, como o IPv4, e seu desempenho de velocidade, de latência, de taxa de erros, de tempo de resposta, são dificuldades para o atendimento da demanda crescente dos dispositivos conectados à rede e seus novos desafios que são cada vez mais emergentes (ETSI, 2014).

As Redes Definida por *Software* então surgem como uma proposta para permitir a evolução da rede diante das limitações expostas. A ideia desse novo paradigma é desacoplar a lógica de controle de rede dos dispositivos que compõem a arquitetura e com isso promover a centralização do plano de controle e possibilitar o desenvolvimento de análises detalhadas chegando a decisões sobre como o sistema, como um todo, deve operar sendo baseado em *software* (NISHTHA; SOOD, 2014).

Promove também mais flexibilidade através de sua implementação em *hardware* de comutação, com velocidade maior que o encaminhamento convencional, e a gerência da condução do tráfego pela rede, que por uma interface programável permite que o controlador possa desenvolver aplicações que otimizam a rede e seu desempenho para características específicas. Ao introduzir novas abstrações, minimiza os problemas com controle de rede, simplificando-a e gerenciando-a (GUEDES, 2012; COMER, 2016; KREUTZ et al., 2015).

Ser programável significa permitir várias adaptações necessárias para cada uso específico de requerimento de informação, adequando-se até mesmo a alocação da largura de banda de forma dinâmica. Esse dinamismo é importante para abranger novas tecnologias e suas funções com maior eficiência e agilidade.

As funções, necessárias para computação na nuvem, tem de ser elásticas, de alta disponibilidade de armazenamento, processamento e personalização da entrega de dados. (OCAMPOS, 2015).

Owens (2010) descreve a elasticidade como essencial para o conceito de computação em nuvem. Da perspectiva do fornecedor, a elasticidade garante melhor uso dos recursos computacionais, proporcionando economias de escala e permitindo que vários usuários sejam atendidos simultaneamente. A partir de perspectiva do usuário, a elasticidade tem sido usada principalmente para evitar a provisão inadequada de recursos e, consequentemente, a degradação do desempenho do sistema. Além disso, alguns estudos apresentam o uso da elasticidade para outros fins, como, aumentando a capacidade de recursos locais, redução de custo e economia de energia. A última característica, refere-se aos métodos empregados na implementação de soluções de elasticidade (CALHEIROS et al., 2011; FIT'O; PRESA; FERNANDEZ, 2010; MARSHALL; KEAHEY; FREEMAN, 2010; SHARMA; et al., 2011; SHEN; SUBBIAH; GU; WILKES, 2011; BNDES, 2017; AMAZON, 2018).

Outro conceito importante relacionado as funções da nuvem é a orquestração. Com orquestração é possível programar a automatização de comportamentos, coordenando elementos que o hardware e software da rede necessitam para realizar atualizações, gerenciamento e provisionamento de recursos para serviços e aplicativos através de um controlador próprio. Essa característica centralizada e extensível torna-se ideal para encaminhamento de rede, segundo o modelo OSI (Open System Interconnection), nos níveis físico, enlace e rede (SDXCENTRAL; 2015).

Nas subseções a seguir são abordadas as aplicações de Virtualização e *OpenFlo*w as quais estão relacionadas a SDN e que contribuem para sua implantação e otimização.

2.2.1 Virtualização das Funções de Rede

A Virtualização das Funções da Rede (do inglês Network Functions Virtualization - NFV) foram desenvolvidas para solucionar o problema da ossificação da rede, onde as aplicações baseadas em *hardware* tem sua vida útil reduzida com a aceleração de inovações, o que limita sua capacidade de avanço, e dificulta a integração entre tecnologias diferentes, reduzindo assim o retorno dos investimentos em novos serviços voltados a rede. As funções sendo implantadas em *software* podem ser executadas em vários padrões de mercado, tornando os equipamentos genéricos, melhorando a eficiência de processos operacionais, com a possibilidade de realocação em locais distintos, sendo flexível e sem necessitar de instalações de novos *hardwares* através da migração dinâmica de cargas para a nuvem - o que reduz o consumo de energia elétrica (ETSI, 2014; BEDOR et al., 2015; ROSA, et al., 2014; CISCO, 2018a).

As NFV compartilham elementos com a SDN pois ambas trabalham com a ideia da virtualização de *softwares* e apesar de serem tecnologias diferentes, podem combinar suas funcionalidades, como tarefas de gerenciamento e orquestração que são programadas através da customização de funções de rede pelo plano de dados, para então obter a padronização das interfaces. Algumas funções se relacionam para obter implementações de IoT, oferecendo um tráfego mais ágil, direção e otimização conjunta de funções e recursos de rede (BIZANIS; KUIPERS, 2015; LI, 2015; BEDOR et al., 2015, CISCO, 2018b).

2.2.2 OpenFlow

As redes que são definidas por *software* possuem dois tipos de interface, para norte (northbound) e para sul (southbound). A aplicação de interface programável (Aplication Programming Interface - API) *northbound* realiza a comunicação entre o plano de gerenciamento, traduzindo requisitos das aplicações desse plano em instruções de baixo nível para os dispositivos, com o plano de controle, transmitindo as estatísticas de rede processadas pelo controlador e fornecidas pelos dispositivos da rede. Já a API *southbound* realiza o vínculo dos elementos de controle e encaminhamento de dados, comunicando os requisitos das aplicações de rede e reprogramando os equipamentos para que possam desempenhar várias funções, como roteamento, controle de fluxo e comutação. É nesse ambiente que o protocolo *OpenFlow* se encontra (GOMES et al., 2015).

O *OpenFlow* é um protocolo aberto para programação de tabela de fluxos para dispositivos da rede, como comutadores, *switches,* roteadores, pontos de acesso sem fio, determinando ações de encaminhamento de pacotes e particionando o tráfego de acordo com o interesse dos operadores. Realiza uma conexão de planos de dados, através de equipamentos, aos planos de controle, através de um controlador que pode ser implementado em um servidor comum (ROTHENBERG, 2011).

Um fluxo é uma combinação de campos de cabeçalho das camadas transporte, rede ou enlace a ser processado pelo dispositivo da rede. Cada tabela de fluxo se comunica com o controlador que é responsável pelas regras e ações instaladas no *hardware*, configurando assim as entradas de fluxo com base nos pacotes de dados de entrada e mantendo estatísticas de pacotes, como número de cada fluxo e seu tempo para promover refinamentos e decisões de manipulação dos pacotes (NISHTHA, SOOD, 2014; ROTHENBERG, 2011; HU, 2014).

Dessa forma é promovida a generalização dos planos de dados, separando as políticas de rede, sua implementação na comutação de *hardware*, encaminhamento de tráfego, consequentemente gerando flexibilidade. Ao possibilitar a criação e reconfiguração das abstrações em redes e facilitando inovação e evolução, esse protocolo vem ganhando atenção das indústrias e fornecedores oferecendo suporte de API *OpenFlow* para seus equipamentos (KIRKPATRICK, 2013; KREUTZ et al., 2015).

2.3 Comparativo das Abordagens

Nessa seção são comparadas as características das redes tradicionais e das redes definidas por *software* visando expor as semelhanças e diferenças dessas abordagens. A forma utilizada para tabular as características, conforme mostrado na Tabela 1 a seguir, foram baseadas nos trabalhos de Rezende (2015), Rubens (2017), Bobba *et al.*(2014), Rothenberg (2011), Oliveira (2015), Pinheiro (2013) onde se buscou a identificação da computação relacionada ao contexto de recursos e funcionalidades que são exigidos para a nuvem.

Tabela 1 – Comparação entre as abordagens Tradicional e SDN

Características	Tradicional	SDN

Tipo de abstração	*Hardware* (plano de dados)	*Software* (plano de controle)
Infraestrutura	Plano de controle e plano de dados integrados no mesmo sistema físico	Plano de controle e dados desacoplados
Arquitetura em camadas	Aplicação, transporte, rede, enlace e física	Aplicação, controle e planos de dados
Plano de controle	Troca de informações de controle entre os equipamentos, responsável pelo cálculo das rotas dos dispositivos	Interface aberta para plano de controle, coordena todas as ações e decisões dos dispositivos
Plano de dados	Não há acesso direto ao plano de dados	Acesso ao plano de dados, controlando e executando maneira que os encaminhamentos acontecem
Inteligência da rede	Distribuída	Centralizada
Configurável	Não configurável, *software*s e hardwares específicos, com configuração única vinculada ao fabricante, tornando-a fechada	Configurável, contendo *software* genérico e independente do fabricante
Implementação de políticas e decisões	Feita de maneira local devido aos equipamentos serem fechados ao proprietário	Feita de maneira central, realizadas em um único plano de controle, obtendo visão global dos equipamentos
Flexibilidade	Não é flexível, sua arquitetura não pode ser alterada ou modificada	Flexível, permite a programação do plano de controle
Fluxo	Tende-se a seguir rotas fixas determinadas pelos protocolos de roteamento e comutação	Unifica comportamento de diferentes tipos de dispositivos com dinamismo sobre o fluxo por meio do plano de controle
Elasticidade	Não possui garantia de adquirir recursos de modo dinâmico	Possui elasticidade, permite que vários usuários sejam atendidos e aumentando a capacidade de recursos

Escalabilidade	Ineficaz para atender os novos padrões de tráfego necessitando de uma rede dinâmica e virtualizada	Limitada pelo aumento de latência de acordo com o distanciamento entre os comutadores e o controlador, número de conexões suportadas pelo controlador e número de entrada na tabela de fluxos
Orquestração	Não possui orquestração, não permite personalizar funções e nem modificar estados da rede	Possui, sendo um sistema operacional da rede que possibilita a implementação modular para modificar o estado de rede e coordenar suas interações

Fonte: Autores (2020).

As redes tradicionais são integradas verticalmente, isto é, os planos de controle e de dados são agrupados. Essa concentração impõe dificuldades no gerenciamento da rede. A proposta das SDN é quebrar essa integração vertical, separando a lógica de controle da rede dos roteadores e comutadores subjacentes.

Além disso, as redes definidas por *software* introduzem uma habilidade essencial para contornar alguns dos problemas anteriormente descritos, elas são programáveis. Dessa forma, novas tecnologias podem ser implementadas na rede de forma gradual.

O fato das SDN serem programáveis lhes confere versatilidade e juntamente às VNFs (Virtualized Network Functions) permitem garantir qualidade de serviço, o que uma rede tradicional não é capaz. Isso se dá devido ao fato que ela foi construída sob dois paradigmas: FIFO (First in First Out) e a lei do menor esforço, o que impede a diferenciação de serviços, pois não existe nenhuma forma de priorizar pacotes a serem enviados, uma vez que todos ficam na mesma fila em que o primeiro pacote a chegar é o enviado primeiro.

A capacidade para programar simultaneamente (ou orquestrar) grande número de dispositivos de rede faz das SDN um forte recurso para futuros operadores de rede. No entanto, as estratégias de transição devem ser consideradas e aplicações amigáveis ao usuário devem ser desenvolvidas.

2.4 Soluções em SDN

O paradigma SDN está sendo utilizado como facilitador da implantação da IoT, no entanto, ainda requer maio abstração para os níveis necessários de segurança e serviço a serem atingidos para a integração de componentes e principalmente para o aumento da infraestrutura de IoT.

As redes de sensores sem fio estão utilizando estruturas de IoT mais amplas, realizando interconexão através da SDN e da virtualização, aproveitando a possibilidade de injeção lógica de roteamento que a SDN oferece. A IoT implantada

em ambientes urbanos deve permitir que o conjunto de nós sensores suportem múltiplas aplicações de vários desenvolvedores em todo o meio físico com sua infraestrutura sendo apenas via *software.* A SDN pode ajudar na orquestração da rede e possibilitar a exploração das APIs. Porém ser centralizada, não se adequa as diferentes redes de acesso que são esperadas para a escala urbana, bem como a mobilidade para ir de um ponto de acesso a outro, sendo proposto um esquema distribuído para essa implantação (COX et al., 2017).

As Veicular Ad Hoc Networks (VANETs) são redes veiculares onde os veículos se comunicam entre si em um ambiente *ad hoc* (configuração de rede de dispositivos, chamados também de nós, não havendo um nó central em que as informações transmitidas dos outros nós convergem eliminando a necessidade de um roteador central para realizar a comunicação dessa rede com outros destinos) com sua infraestrutura fixa com transceptores de beira de estrada ou estações rádio base de telefonia celular (MORAES; XAUD; XAUD, 2009). Sua configuração possui um controlador SDN que executa ações de roteamento enquanto os veículos e transceptores executam de comutadores SDN (CHAHAL et al., 2017).

A prática de *Bring Your Own Device* (BYOD) ou "traga seu próprio dispositivo" vem se enquadrando no contexto IoT e servindo de driver para adoção das SDN. Tem como proposta uma rede refinada de segurança para gerenciamento de rede e aplicação de políticas para aplicativos e dispositivos móveis em redes corporativas, estendendo os recursos de SDN para o host final (MORRISON et al., 2018; KANG et al., 2019).

A segurança em IoT é algo muito discutido, desafiando os operadores para poder garantir que não haja propósitos maliciosos na rede dos dispositivos. Existem trabalhos que apontam diversas formas de aumentar a segurança de uma rede, principalmente as SDN que dispõem de mecanismos mais dinâmicos e permitem o monitoramento dos fluxos de rede, detectando anomalias e impedindo atividade maliciosa em redes organizacionais. Alguns artigos, são focados em alterar o protocolo *OpenFlow*, outros em adicionar *middleboxes* na rede (SPOONER; ZHU, 2016; HA; KIM, 2016). No entanto, é importante ressaltar que ainda existem muitos problemas de segurança para que a SDN seja confiável e robusta em produção (PRATHIMA et al., 2019).

Minimizar o consumo de energia, sem afetar desempenho e segurança, também é uma consideração importante para transmissão de dados em uma rede IoT. Em geral, equilibrar a necessidade de segurança e minimizar consumo de energia para IoT em camadas físicas, camadas de rede e aplicativos em execução na IoT, requer uma arquitetura com tais capacidades. Nesse contexto, a SDN tem sido explorada juntamente com novas tecnologias emergentes (YAZDINEJAD et al., 2020).

2.5 Desafios para implantação das SDN

A plataforma SDN é considerada como uma solução promissora para Internet das Coisas porque pode oferecer melhorias expressivas na rede, no entanto, ela traz alguns desafios a serem resolvidos nesse contexto e que dificultam sua implantação, como: garantir a segurança dos controladores, encontrar formas de satisfazer as diferentes métricas de Qualidade de Serviço (QoS), desenvolver sistemas operacionais para gerenciar eficientemente a integração entre IoT e SDN, e preencher a lacuna de pesquisas quando a Internet das Coisas é integrada aproveitando os benefícios da SDN (HAKIRI et al., 2015).

Segundo Scott-Hayward *et al.* (2013) existem duas linhas de pensamento relacionados aos problemas de segurança de uma SDN: a primeira mostra que os problemas de segurança podem ser resolvidos por programação, devido a visão centralizada da rede; a segunda linha diz que a mesma centralização e programação da rede trazem novos problemas de segurança. Em uma rede centralizada, fica mais simples fazer bloqueios pontuais em *switches* mais próximos a usuários, algo que não é possível em redes tradicionais, pois em geral, os *switches* não têm essa capacidade. Por outro lado, um atacante tem uma boa chance que seus ataques passem pelo controlador, abrindo um grande leque de opções. Alguns exemplos são injeção de código nos pacotes para análise ou execução no controlador, geração de pacotes mal formados para sobrecarregar o controlador, dentre outros. KHAN *et al.* (2016) realizam uma previsão dos possíveis problemas de segurança que podem ocorrer até 2020 em decorrência de vulnerabilidades existentes.

Ao separar o plano de controle do plano de dados, também surgem novos desafios de segurança para as arquiteturas SDN. Isso inclui problemas com as APIs do controlador SDN, memória e vários outros. As redes baseadas no *OpenFlow*, por exemplo, são suscetíveis a uma variedade de problemas que incluem adulteração, repúdio, divulgação de informações, *spoofing,* escalonamento de privilégios e negação de serviço. Como muitos controladores utilizam o *OpenFlow,* eles também têm problemas de segurança e resiliência comparáveis (KLOTI; KOTRONIS; SMITH, 2013; SEZER, 2013; PORRAS et al., 2012; CAMP, 2013).

A heterogeneidade de aplicações e requisitos mostram como os comportamentos de aplicativos IoT são dinâmicos, desafiando as tecnologias baseadas em definição de *software* a encontrar formas de satisfazer as diferentes métricas de QoS, uma vez que cada necessidade tem sua política mais apropriada para sua funcionalidade, honrando parâmetros estabelecidos entre provedores de serviço e usuários finais, especificados em Acordos de Nível de Serviço (Service Level Agreements - SLA) (ANDRIOLI; RIGHI; AUBIN, 2017;REZENDE, 2016).

Os dispositivos IoT, em geral, usam várias tecnologias para intercomunicação e embora a IoT use vários *middlewares* para reduzir, a diferença entre a passagem de mensagens de aplicativos e dispositivos, a interoperabilidade ainda é um problema para aprimorar o desempenho e aumentar a reutilização da rede IoT (KIANI, 2018). Para lidar com essa interoperabilidade, o Sistema Operacional de Rede (Network Operation System - NOS) desempenha um papel importante. Os NOS atuam como controladores em uma rede definida por *software* e são empregados para realizar a interoperabilidade em sistemas heterogêneos, porém se tornam inflexíveis e estáticos, quando se trata das redes IoT baseadas em sensores sem fio (Wireless Sensor Network - WSN). Esses controladores não se adequam, pois não são capazes lidar com interoperabilidade em larga escala e conversão de fluxo (BERA et al.2016; TAYYABA et al., 2016).

Outro desafio é a viabilização de negócios para estimular o desenvolvimento e o interesse das empresas nesse setor de tecnologia e assim aumentar as pesquisas e acelerar os resultados para implantação da plataforma SDN, acompanhando o crescimento de dispositivos e informações. Esse estímulo acompanha a QoS, reafirmando questões como leis de proteção de dados pessoais que integram uma nova legislação aplicada no país (GROSSMANN, 2017).

CONSIDERAÇÕES FINAIS

Este trabalho apresentou um estudo sobre as principais características de uma rede definido por *software* e os desafios atuais de pesquisa que ampliam as formulações sobre o tema e que podem orientar futuras investigações. O estudo fez referência às pesquisas recentes, onde buscou-se identificar alguns dos principais impulsionadores e preocupações que impactam o avanço das SDN.

Tendo identificadas as principais características de uma rede definida por *software*, observou-se as oportunidades para se avançar com o plano de controle e o plano de dados, a interoperabilidade com tecnologias IoT, impactos na segurança, e outras tecnologias emergentes.

No que diz respeito ao futuro das SDN, fatores comuns entre todas as áreas de pesquisa, incluindo latência, taxa de fluxo, redundância, confiabilidade, segurança, custo e disponibilidade ainda devem ser mais bem explorados.

O estudo também serve como um ponto de referência para orientar os futuros pesquisadores e profissionais da indústria que procuram contribuir com o avanço da SDN, especificamente, no que diz respeito à inúmeras aplicações, atuais e imaginadas, dentro o contexto da IoT.

SOFTWARE DEFINED NETWORKS: A SYSTEMIC STUDY ON NEW APPROACHES IN THE CONTEXT OF IOT

ABSTRACT

Software Defined Networks (Software Defined Networks, or SDN) constitute a paradigm for the development of research on computer networks that has been gaining the attention of a large part of the academic community and industry. However, these networks still have many challenges operational and research. Given this scenario, the present work investigates the components of a software defined network system with which it obtained as its main result, the presentation of an overview of new approaches and the identification of the main challenges of ongoing research that may contribute to the progress of SDN.

Keywords: Internet of Things. Computing in the Cloud. Networks Defined by Software.

REFERÊNCIAS

ANDRIOLI, L.; RIGHI, R. da R.; AUBIN, M. R. **Analisando métodos e oportunidades em redes definidas por software (SDN) para otimizações de tráfego de dados**. Revista Brasileira de Computação Aplicada, v. 9, n. 4, p.2, 13 dez. 2017.

AMAZON. **Amazon Spot Instances.** Disponível em: <http://aws.amazon.com/ec2/spot-instances/>**.** Acesso em: 24 de fev. 2018.

ATZORI, L.; IERA, A.; MORABITO, G. **The internet of things: A survey. Computer networks**, v. 54, n. 15, p. 2787-2805, 2010.

BEDOR, A. A. B.; ALVES, C. V. de S.; PEREIRA, G. O.; SOUZA, P. M. R.; MELO, M. A. **NFV - Network Function Virtualization.** Rio de Janeiro, RJ. In: Universidade Federal do Rio de Janeiro, 2015. Disponível em: <https://www.gta.ufrj.br/ensino/eel879/trabalhos_vf_2015_2/NFV/introducao.html>. Acesso em: 10 mai. 2018.

BERA, S.; MISRA, S.; ROY, S.; OBAIDAT, M. (2016). **Soft-WSN: Software-Defined WSN Management System for IoT Applications.** IEEE Systems Journal.

BIZANIS, N.; KUIPERS, F. A. **SDN and Virtualization Solutions for the Internet of Things: A Survey**. In: IEEE Access, v. 4, p. 5591–5606, 2016.

BNDES, Banco Nacional de Desenvolvimento Econômico e Social. **Relatório do Plano de Ação – Iniciativas e Projetos Mobilizadores**. Versão 1.1, nov. 2017. Disponível em: <https://www.bndes.gov.br/wps/wcm/connect/site/269bc780-8cdb-4b9b-a297-53955103d4c5/relatorio-final-plano-de-acao-produto-8-alterado.pdf?MOD=AJPERES&CVID=m0jDUok&CVID=lXysvoX&CVID=lXysvoX&CVID=lXysvoX&CVID=lXysvoX&CVID=lXysvoX&CVID=lXysvoX>. Acesso em: 25 nov. 2017.

BOBBA, R.; BORRIES, D. R.; HILBURN, R.; SANDERS, J.; HADLEY, M.; SMITH, R. **Software-Defined Networking Addresses - Control System Requirements**, 2014. Disponível em: <https://pdfs.semanticscholar.org/4a97/2628cf32b0abb87e4bcafc69c22707917b4b.pdf>. Acesso em: 16 mai 2018.

CALHEIROS, R. N.;VECCHIOLA, C.; KARUNAMOORTHY, D.; BUYYA, R. **The aneka platform and qos-driven resource provisioning for elastic applications on hybrid clouds.** In: Future Generation Computer Systems, vol. 28, n. 6, p. 861–870, jun. 2011.

CAMP, L. J.SMALL, C. **OpenFlow vulnerability assessment,** in Proc. 2nd ACM SIGCOMM Workshop Hot Topics Softw. Defined Netw., 2013, pp. 151–152.

CISCO. **Software-Defined Networking**. Disponível em: <https://www.cisco.com/c/en/us/solutions/software-defined-networking/overview.html#~stickynav=4>. Acesso em: 19 jul. 2018.

CISCO. **Cisco Network Functions Virtualization (NFV)**. Disponível em: <https://www.cisco.com/c/en/us/solutions/service-provider/network-functions-virtualization-nfv/index.htmlAcesso> em: 19 ago. 2018

CHAHAL, M.; HARIT, S.; MISHRA, K. K; SANGAIAH, A. K.; ZHENG, Z. A Survey on software-defined networking in vehicular ad hoc networks: Challenges, applications and use cases, Sustainable Cities and Society, vol. 35, p. 830-840, nov. 2017.

COMER, D. E. **Redes de Computadores e Internet**, 6. ed. Rio Grande do Sul: Bookman, p. 482-485, 2016.

COX, J. H.; CHUNG, J.; DONOVAN, S.; IVEY, J.; CLARK R. J.; RILEY, G.; OWEN, H. L. **Advancing Software-Defined Networks: A Survey,**. In: IEEE Access, v. 5, p. 25487-25526, 2017.

DAVE, E. **The Internet of Things: How the Next Evolution of the Internet Is Changing Everything.** In: Cisco Internet Business Solutions Group (IBSG), Cisco Systems, Inc., San Jose, CA, USA, White Paper, 2011.

ETSI, European Telecommunications Standards Institute. **Network Functions Virtualization.** França, 2014. Disponível em: <http://www.etsi.org/images/files/ETSITechnologyLeaflets/NetworkFunctionsVirtualization.pdf>. Acesso em: 10 mai. 2018

FIRJAN. Indústria 4.0: **Internet das Coisas.** In: Cadernos SENAI de Inovação, Rio de Janeiro, RJ, jul. 2016. Disponível em: <http://www.firjan.com.br/publicacoes/publicacoes-de-inovacao/industria-4-0.htm>. Acesso em: 20 mar. 2018.

FIT'O J. O.; PRESA, I. G.; FERNANDEZ J. G. **Sla-driven elastic cloud hosting provider.** In: Proceedings of the 18th Euromicro Conference on Parallel, Distributed and Network-based Processing, ser. PDP 2010. IEEE, p. 111–118, 2010.

GOMES, L. C.; ARAUJO, M. S. de A.; TABAK, P.; EUSÉBIO, P. S.; CAMPOS, V. S. **Redes Definidas por Software (SDN)**. Rio de Janeiro, RJ. In: Universidade Federal do Rio

de Janeiro, 2015. Disponível em: <https://www.gta.ufrj.br/ensino/eel879/trabalhos_vf_2015_2/SDN/index.html>. Acesso em: 11 mai 2018.

GROSSMANN, L. O. **Viabilizar Negócios em Internet das Coisas é o Desafio de Agora**. Futurecom. 20. ed. São Paulo, 04 out. 2017. Disponível em: <http://www.convergenciadigital.com.br/cgi/cgilua.exe/sys/start.htm?UserActiveTemplate=site&infoid=46410&sid=153>. Acesso em: 01 ago. 2018.

GUEDES, D.; VIEIRA, L.; VIEIRA, M.; RODRIGUES, H.; NUNES, R. **Redes Definidas por Software: uma abordagem sistemica para o desenvolvimento de pesquisas em Redes de Computadores.** In: Minicursos do Simposio Brasileiro de Redes de Computadores-SBRC 2012, p. 160–210, 2012.

HA, T.; KIM, S. **Suspicius traffic sampling for intrusion detection in software-defined netwoks.** Computer Netwoks, v.109, p.172-186, 2016.

HAKIRI, A.; BERTHOU, P.; GOKHALE, A.; ABDELLATIF, S. **Publish/subscribe-enabled software defined networking for efficient and scalable IoT communications.** In: IEEE Communications Magazine, v. 53, n. 9, p. 48-54, set. 2015.

HU, F.; HAO, Q.; BAO, K. **A Survey on Software-Defined Network and OpenFlow: From Concept to Implementation.** In: IEEE Communications Surveys & Tutorials, v. 16, n. 4, p. 2181-2206, 2014.

KANG, Q., XUE, L., MORRISON, A., TANG, Y., CHEN, A., & LUO, X. **Programmable In-Network Security for Context-aware BYOD Policies**. arXiv preprint arXiv:1908.01405, 2019.

KIRKPATRICK, K. **Software-defined networking.** In: Communications of the ACM, v. 56, n. 9, p. 16-19, 9 set. 2013.

KHAN, S.; SHAH, M.; SHER, N.; ASIM, Y.; NAEEM, W.; KAMRAN, M. (2016). **Software-Defined Networks (SDNs) and Internet of Things (IoTs): A Qualitative Prediction for 2020.** International Journal of Advanced Computer Science and Applications.

KREUTZ, D.; RAMOS, F. M. V.; VERÍSSIMO, P. E.; ROTHENBERG, C. E.; AZODOLMOLKY, S.; UHLIG, S. **Software-Defined Networking: A Comprehensive Survey.** In: *Proceedings of the IEEE*, vol. 103, n. 1, p. 14-76, jan. 2015.

KIANI, F. **A Survey on Management Frameworks and Open Challenges in IoT**. Wireless Communications and Mobile Computing 2018 (2018).

KLOTI, R; KOTRONIS,V; SMITH,P. **OpenFlow: A security analysis**, in Proc. 21st IEEE Int. Conf. Netw. Protocols (ICNP), Oct. 2013, pp. 1–6.

LI, Y.; CHEN, M. **Software-Defined Network Function Virtualization:** A Survey. In: IEEE Access, v. 3, p. 2542-2553, 2015.

MARSHALL, P.; KEAHEY, K.; FREEMAN, T. Elastic site: Using clouds to elastically extend site resources. In: Proceedings of the 10th Intl. Conference on Cluster, Cloud and Grid Computing, ser. CCGRID 2010. IEEE, p. 43–52, 2010.

MELL, P. M.; GRANCE, T. **The NIST Definition of Cloud Computing,** 2011. Special Publication 800-145, National Institute of Standards and Technology, Gaithersburg, MD.

MONGA, I et al**. Operationalization of Software-Defined Networks (SDN)** . The 3rd International Symposium on Network Virtualization, September 6th, 2013.

MORAES, A. L. D.; XAUD, A. F. dos S.; XAUD, M. F. dos S. **Redes Ad Hoc**. Rio de Janeiro, RJ. In: Universidade Federal do Rio de Janeiro, 2009. Disponível em: <https://www.gta.ufrj.br/grad/09_1/versao-final/adhoc/index.html>. Acesso em: 25 jul. 2018

MOREIRA, M. D. D.; FERNANDES, N. C.; COSTA, L. H. M. K; DUARTE, O. C. M. B. **Internet do Futuro: Um Novo Horizonte.** In: Minicursos do Simposio Brasileiro de Redes de Computadores (SBRC). Sociedade Brasileira de Computação (SBC), cap. 1, p. 52, 2009.

MORRISON, A., XUE, L., CHEN, A., & LUO, X. **Enforcing Context-Aware {BYOD} Policies with In-Network Security**. In: 10th {USENIX} Workshop on Hot Topics in Cloud Computing (HotCloud 18). 2018.

NISHTHA; SOOD, M. **Software defined network** – Architectures, *2014. In: International Conference on Parallel, Distributed and Grid Computing*, Solan, 2014, p. 451-456.

NUNES, B. A. A.; MENDONCA, M.; NGUYEN, X.; OBRACZKA, K.; TURLETTI, T. A **Survey of Software-Defined Networking: Past, Present, and Future of Programmable Networks.** In *IEEE Communications Surveys & Tutorials*, vol. 16, n. 3, p. 1617-1634, 2014.

OCAMPOS, T. **Internet das Coisas nas Nuvens**. Computação Brasil - Revista da Sociedade Brasileira de Computação, Porto Alegre, RS, n. 29, p. 19-22, abr. 2015. Disponível em:

<http://www.sbc.org.br/images/flippingbook/computacaobrasil/computa_29_pdf/comp_brasil_2015_4.pdf>. Acesso: 20 mar. 2018.

OLIVEIRA, I. C. **Aprimorando a Elasticidade de Aplicações de Banco de Dados utilizando Virtualização em Nível de Sistema Operacional.** Porto Alegre: PUC-RS, p. 29-45, 2015. Disponível em: <http://tede2.pucrs.br/tede2/bitstream/tede/6433/2/476714%20-%20Texto%20Completo.pdf>. Acesso em: 16 mai 2018.

OWENS, D. **Securing elasticity in the cloud**. Queue, vol. 8, n. 5, p. 10, 2010. Disponível em: <https://queue.acm.org/detail.cfm?id=1794516>. Acesso em: 10 abr. 2018.

PASSOS, A.P. R. S.; BARBOSA, A. de S.; FIGUEIREDO, E. R.; FILHO, M. A. C. de S.; MAIA, T. L. A.; CARVALHO, Y. M. L. **Software Defined Networks.** Rio de Janeiro, RJ. In: Universidade Federal do Rio de Janeiro, 2016. Disponível em: <https://www.gta.ufrj.br/grad/16_2/2016SDN/conceitos.html>. Acesso em: 03 abr. 2018.

PERERA, C.; ZASLAVSKY, A.; CHRISTEN, P.; GEORGAKOPOULOS, D. Context Aware **Computing for The Internet of Things: A Survey**. In: *IEEE Communications Surveys & Tutorials*, vol. 16, n. 1, p. 414-454, 2013.

PINHEIRO, A. J. **Escalabilidade do Plano de Controle em Redes Overflow**. Quixadá, CE, Universidade Federal do Ceará, 2013. Disponível em: <http://www.repositoriobib.ufc.br/000012/000012bd.pdf>. Acesso em: 03 abr. 2018.

PORRAS, P.; SHIN, S.; YEGNESWARAN, V.; FONG, M.; TYSON, M.; GU, G. 2012. **A security enforcement kernel for OpenFlow networks.** In Proceedings of the first workshop on Hot topics in software defined networks (HotSDN '12). Association for Computing Machinery, New York, NY, USA, 121–126.

PRATHIMA M.J.;VANI K.A.; RAMA M.B.K.N. (2019) **SDN Security: Challenges and Solutions**. In: Sridhar V., Padma M., Rao K. (eds) Emerging Research in Electronics, Computer Science and Technology. Lecture Notes in Electrical Engineering, vol 545. Springer, Singapore.

REZENDE, J. F. **Rede Definida por Software (SDN).** Rio de Janeiro, RJ. In: Semana de Programa de Engenharia de Sistemas e Computação – UFRJ, 2015. Disponível em: <http://www.cos.ufrj.br/semana/2015/slides/slides_rezende.pdf>. Acesso em: 11 mai 2018.

REZENDE, P. H. A. **Extensões na Arquitetura SDN para o Provisionamento de QoS através do monitoramento e uso de Múltiplos Caminhos**. Uberlância: UFU, p. 44-46, 2016. Disponível em: <https://repositorio.ufu.br/bitstream/123456789/17550/1/ExtensoesArquiteturaSDN.pdf>. Acesso em: 31 jul. 2018.

ROSA, R. V.; SIQUEIRA, M. A.; ROTHENBERG, C. E.; BAREA, E.; MARCONDES, C. A. C. **Network Function Virtualization: Perspectivas, Realidades e Desafios.** São Paulo, 2014. Disponível em: <https://intrig.dca.fee.unicamp.br/wp-content/plugins/papercite/pdf/rosa2014network.pdf>. Acesso em: 10 mai. 2018.

ROTHENBERG, C. E.; NASCIMENTO, M. R.; SALVADOR, M. R.; MAGALHÃES, M. F. **OpenFlow e redes definidas por software: um novo paradigma de controle e inovação em redes de pacotes.** In: Cad. CPqD Tecnologia, Campinas, SP, v. 7, n.1, p. 65-76, jul. 2010/jun. 2011.

RUBENS, M. **Uma Análise sobre a viabilidade de Implantação da Arquitetura de Redes Definidas por Software,** 27 fev. 2017. Disponível em: <https://pt.linkedin.com/pulse/uma-an%C3%A1lise-sobre-viabilidade-de-implanta%C3%A7%C3%A3o-da-redes-maython-rubens>. Acesso em: 16 mai 2018.

SAKIR, S.; SCOTT-HAYWARD, S.; PUSHPINDER, K. C.; FRASER, B.; LAKE, D.; FINNEGAN, J.; VILJOEN, N.; MILLER, M.; RAO, N. **Are we ready for SDN? Implementation challenges for software-defined networks.** In: *IEEE Communications Magazine*, vol. 51, n. 7, p. 36-43, jul 2013.

SDXCENTRAL. **What is SDN Orchestration (SDN Policy Orchestration)?**.2015. Disponível em: <https://www.sdxcentral.com/sdn/definitions/what-is-sdn-orchestration/>. Acesso em: 19 jul. 2018.

SEZER, S. et al., **Are we ready for SDN? Implementation challenges for software-defined networks**, IEEE Commun. Mag., vol. 51, no. 7, p. 36–43, Jul. 2013.

SHARMA, U.; SHENOY, P.; SAHU, S.; SHAIKH A. **A cost-aware elasticity provisioning system for the cloud.** In: Proceedings of the 31st Intl. Conference on Distributed Computing Systems, ser. ICDCS 2011. IEEE, p. 559–570, 2011.

SHEN, Z.; SUBBIAH, S.; GU, X.; WILKES, J. **Cloudscale: elastic resource scaling for multi-tenant cloud systems.** In: Proceedings of the 2nd Symposium on Cloud Computing, ser. SOCC 2011. ACM, p. 5:1–5:14, 2011.

SPOONER, J.; ZHU, S. Y. **A Review of Solutions for SDN-Exclusive Security Issues.** (IJACSA) International Journal of Advanced Computer Science and Applications,vol. 7, n.8, 2016.

SCOTT-HAYWARD, S.; O'CALLAGHAN, G.; SEZER, S. **Sdn Security: A Survey,** 2013 IEEE SDN for Future Networks and Services (SDN4FNS), Trento, 2013, pp. 1-7.

TAYYABA, S. K.; SHAH, M. A.; KHAN, N. S. A.; ASIM, Y.; NAEEM, W.; KAMRAN, M. **Software-Defined Networks (SDNs) and Internet of Things (IoTs): A Qualitative Prediction for 2020**. In: International Journal of Advanced Computer Science and Applications (IJACSA), v. 7, n. 11, 2016.

VASHI, S.; RAM, J.; MODI, J.; VERMA, S.; PRAKASH, C. **Internet of Things (IoT): A vision, architectural elements, and security issues.** In: International Conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud) (I-SMAC), *P*alladam, p. 492-496, 2017.

YAZDINEJAD, A.; PARIZI, R.; DEHGHANTANHA, A.; ZHANG, Qi; CHOO, K. R. (2020). **An Energy-efficient SDN Controller Architecture for IoT Networks with Blockchain-based Security.** IEEE Transactions on Services Computing. PP. 1-1. 10.1109/TSC.2020.2966970.

Recebido: 2020-05/20
Aprovado: 2021-06-08
DOI: 103895/recit. V12n30.12415.
Como citar: OLIVEIRA D. C, FERREIRA, K L C, CORREIO M. L. R. O. REDES DEFINIDAS POR SOFTWARE: UM ESTUDO SISTÊMICO SOBRE AS NOVAS ABORDAGENS NO CONTEXTO DA IOT R. Eletr. Cient. Inov. Tecnol, Medianeira, v. 12. n. 30, p. 48- 67, jul/set, 2021 Disponível em: <https://periodicos.utfpr.edu.br/recit>. Acesso em: XXX.
Correspondência:
Danielle Costa de Oliveira,
IFMG-Instituto Federal de Minas Gerais, Formiga, MG - R. São Luiz Gonzaga, s/n - Bairro São Luiz, Formiga - MG, 35577-010 - BRASIL.