# Estudo de Vulnerabilidades em API e APP de Controladores SDN ONOS

Marcelo D. S. Leite, Luiz H. dos S. Souza, Waslon T. A. Lopes, Fabrício B. S. Carvalho e Iguatemi E. Fonseca

***Resumo*— No contexto de Redes Móveis de Quinta Geração (5G), a tecnologia de Redes Definidas por Software (SDN – *Software Defined Networks*) têm sido cada vez mais adotada pelo mercado. Por conta disto, um tema bastante atual é o estudo e impacto na cibersegurança nestas redes. O foco deste trabalho está nos possíveis impactos das vulnerabilidades no popular controlador SDN ONOS (*Open Network Operating System*) através das API REST expostas nesse controlador.**

***Palavras-Chave*— SDN, ONOS, Vulnerabilidades, REST API, 5G, Open RAN**

## I. Introdução

O paradigma de Redes Definidas por Software (SDN – *Software Defined Networks*) [1] permite o controle de uma rede por meio de aplicativos de software. O controlador ONOS [2] é uma instância de um popular controlador que pode atuar em uma SDN, gerenciando seus componentes e permitindo uma comunicação tanto externa quanto interna de alguns protocolos (como o REST API). A REST API é um vetor de ataque bastante explorada atualmente e com um impacto bastante profundo na rede. Neste contexto, este trabalho pretende apresentar um estudo das vulnerabilidades encontradas nos *endpoints* das API REST e seus impactos no controlador ONOS.

## II. Controlador ONOS

O controlador ONOS é um sistema operacional que controla e gerencia como os pacotes de dados são encaminhados através dos componentes de rede (como *switches* e *links*). Ele também executa programas e módulos, provendo assim serviços de comunicação entre hosts e redes de computadores. Este controlador possui uma infraestrutura de aplicações onde é possível realizar trabalhos customizados no roteamento de pacotes, monitoramento de serviços e gerenciamento.

Composto por múltiplos módulos e com objetivos específicos, o controlador ONOS possui fácil configuração, fronteiras bem definidas e bibliotecas de protocolos generalistas. A Figura 1 apresenta as camadas da arquitetura do ONOS, que possuem uma finalidade específica definida para alguns serviços primários: Protocolos e Elementos de rede, Provedores, *Core*, Aplicações, Interface Norte (*NorthBound*) API (*Consumer*), Interface Sul (*SouthBound*) API (*Provider*).

A interface Norte (*Northbound*) e interface Sul (*Southbound*) são duas camadas presentes na arquitetura do ONOS, responsáveis por serem uma interface de serviços da qual os aplicativos ou componentes de sua arquitetura podem ter acesso a todos (os outros componentes) presentes no ONOS. A interface Norte pode ser acessada por uma interface gráfica

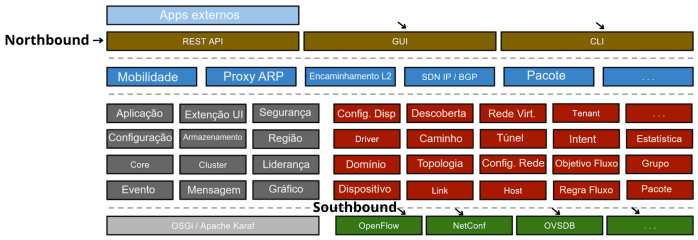


Fig. 1. Subsistemas do Controlador SDN ONOS. Adaptada de [2].

(GUI), linha de comando (CLI) ou por API REST. Por sua vez, a interface Sul define alguns protocolos como: OpenFlow, NETCONF, REST API e SNMP.

A utilização do protocolo REST API está cada vez mais popular e possui como ponto forte a interoperabilidade entre sistemas. Cada vez mais integrada entre os sistemas de software, muitos sistemas utilizam esse protocolo em sua arquitetura. Entretanto, os aspectos de segurança em sua utilização estão cada vez mais sendo ignorados, trazendo assim um vetor de ataque bastante impactante e acarretando assim um forte impacto na segurança desses sistemas. Por conta desta prática não segura da utilização das REST API nas arquiteturas de sistemas web, um projeto chamado OWASP TOP 10 API Security (*Open Web Application Security Project*) [3] foi criado para servir como um guia de boas práticas na utilização deste protocolo. As Top 10 REST API de vunerabilidade divulgada pela OWASP são:

1) Falha de autorização em nível de objeto;
2) Falha de autenticação de usuário;
3) Exposição de dados excessiva;
4) Falta de recursos e limitação de taxa;
5) Falha de autorização de nível de função;
6) Atribuição em massa;
7) Configuração incorreta de segurança;
8) Injeção;
9) Gerenciamento inadequado de ativos;
10) Registro e monitoramento insuficientes.

Desta forma, o foco deste trabalho está no estudo das possíveis vulnerabilidades encontradas nas REST API disponibilizadas pelo controlador ONOS.

## III. Exemplo de aplicação ONOS: Open RAN

Open RAN (*Open Radio Access Networks* – Rede de Acesso de Rádio Aberta) [5] é uma arquitetura de rede que desagrega os componentes tradicionalmente integrados em uma estação base de celular. Em uma infraestrutura de rede tradicional, os elementos de hardware e software são fornecidos por

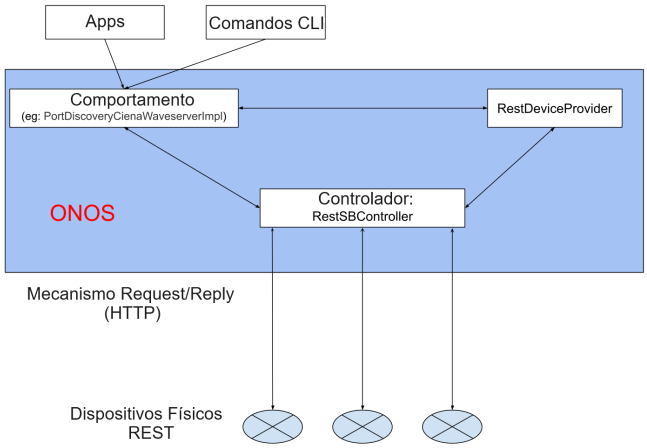


Fig. 2. Protocolo REST Interface Sul (*Southbound)*. Adaptado de [4].

um único fornecedor, tornando as redes caras de implantar, atualizar e manter. Com o Open RAN, esses componentes são desagregados e podem ser fornecidos por diferentes empresas.

Na arquitetura Open RAN, a RAN pode ser vista em quatro blocos principais: Unidade de Rádio (RU), Unidade Distribuída (DU), Unidade Centralizada (CU) e Controlador de Inteligência RAN (RIC – *RAN Intelligent Controller*).

Dentre os componentes que a Open RAN utiliza, há softwares como xApps e rApps para controle do sistema, mas também são utilizados controladores SDN. O componente RIC é nativo da nuvem e se baseia em várias plataformas bem estabelecidas da ONF (*Open Networking Foundation*), incluindo o controlador SDN ONOS. A identificação de vulnerabiliades nos controladores ONOS, por meio do vetor de ataque as REST API, podem impactar profudamente a arquitetura OpenRAN já que existe uma grande dependência desta arquitura aos controlados SDN ONOS.

## IV. Estudo de vulnerabilidade e impacto na arquitetura ONOS

A lista apresentada pelo OWASP TOP 10 *Security* REST API demonstra alguns aspectos das vulnerabilidades relacionadas à utilização de API REST mais comuns na atualidade. As vulnerabilidades listadas possuem um alto grau de impacto na arquitetura e compromentimento do sistema, como por exemplo uma falha na autenticação de um usuário ou um ataque de injeção. A Figura 2 apresenta um exemplo de acesso ao protocolo REST na interface Sul (*southbound*) na arquitetura do ONOS. Neste exemplo é possivel verificar que, por meio dessa comunicação, obtêm-se informações e é possível ter um controle de configuração de dispositivos por meio das API REST expostas nessa camada.

De acordo com as API REST expostas na arquitetura ONOS, é possível ter acesso a um conjunto de categorias de componentes como: Dispositivos (*Device*), Link, *Host*, Topologia, Caminho, Fluxo, Grupo, Aplicações e Configuração de Componentes. Um atacante pode comprometer tokens de autenticações ou explorar falhas na implementção dos *endpoints* para assumir a identidade de outros usuários explorando assim uma vulnerabilidade de autenticação. Esta falha pode gerar um forte impacto também nos aplicativos, comprometendo assim a sua criação, modificação ou remoção. Um ataque deste porte pode comprometer a funcionalidade e disponiblidade dos serviços no sistema como um todo. A lista de *endpoints* do grupo *Application* é mostrada em seguida:

**Application**

GET /applications
GET /applications/app-name
POST /applications/
DELETE /applications/app-name
POST /applications/app-name/active
DELETE /applications/app-name/active
GET /applications/ids/entry
GET /applications/ids/

Como precedente das vulnerabilidades nas REST API dos controladores ONOS, existem 2 CVEs recentementes cadastradas (*Common Vulnerabilities and Exposures*), CVE-2023-30093 [6] e CVE-2023-24279 [7], que são relacionadas às vulnerabilidades do tipo XSS (*cross-site scripting*). Estes tipos de vulnerabilidades são referentes às possibilidades de ataques permitindo que o atacante pode executar um script arbitráreo via um payload customizado por meio de um parâmetro em uma URL na documentação da REST API. Com isso, o ataque pode ter uma severidade bastante alta pois o impacto deste ataque é propocional a quanto malicioso pode ser este script, como o comprometimento de um usuário com altos privilégios permitindo ao atacante controle total da aplicação.

## V. Conclusões

Este estudo teve como objetivo discutir algumas vulnerabilidades e impactos de um atacante, explorando tanto falhas das API REST fornecidas na comunicação dos componentes internos da arquitetura do ONOS, quanto dos aplicativos e agentes externos que consomem os serviços de rede do ONOS. Como trabalhos futuros, pretende-se expandir esse estudo de vulnerabilidades nos *endpoints* das API REST em controladores ONOS utilizados como base na arquitetura de Redes Móveis de Quinta Geração (5G) que utilizam o Open RAN como base tecnológica.

## Referências

[1] A. Kanwal, M. Nizamuddin, W. Iqbal, W. Aman, Y. Abbas and S. Mussiraliyeva, "Exploring Security Dynamics in SDN Controller Architectures: Threat Landscape and Implications,"in IEEE Access, vol. 12, pp. 56517-56553, 2024, doi: 10.1109/ACCESS.2024.3390968.
[2] https://opennetworking.org/onos/ (Acessado em Junho 2024)
[3] https://owasp.org/www-project-api-security/ (Acessado em Junho 2024)
[4] https://wiki.onosproject.org/display/ONOS/Appendix+B%3A+REST+API/ (Acessado em Junho 2024)
[5] M. Polese, L. Bonati, S. D'Oro, S. Basagni and T. Melodia, "Understanding O-RAN: Architecture, Interfaces, Algorithms, Security, and Research Challenges", in IEEE Communications Surveys & Tutorials, vol. 25, no. 2, pp. 1376-1411, Second quarter 2023, doi: 10.1109/COMST.2023.3239220.,
[6] https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2023-30093 (Acessado em Julho 2024)
[7] https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2023-24279 (Acessado em Julho 2024)