Autenticação de Estações de Trabalho em Redes Definidas por Software com Utilização de Certificados Auto-Assinados

Osiel O. Souza[1], **Jeferson C. Nobre** [1]

[1]Instituto de Informática – Universidade do Vale do Rio dos Sinos (UNISINOS)
Caixa Postal 15.064 – 91.501-970 – São Leopoldo – RS – Brazil

osielolivera@gmail.com, jcnobre@unisinos.br

Abstract. *At the same time the architecture of the Software Defined Networking - SDN has to be promising, there are some security challenges to be overcome in the implementation of such technology. The need for authentication of network components becomes a key issue in SDN due to centralization of logic controllers. This paper proposes the implementation of a public key infrastructure combined with the use of self-signed certificates as a possible solution to the authentication problem of the origin in a SDN OpenFlow.*

Resumo. *A mesmo tempo em que a arquitetura das redes definidas por software (Software Defined Networking - SDN) apresenta-se promissora, existem alguns desafios de segurança a serem transpostos na implementação de tal tecnologia. A necessidade de autenticação dos componentes da rede torna-se uma questão fundamental em SDN devido a centralização lógica dos controladores. Este trabalho propõe a implementação de uma infraestrutura de chaves públicas aliada a utilização de certificados auto-assinados como uma possível solução ao problema de autenticação da origem em uma SDN OpenFlow.*

1. Introdução

A abstração de rede proporcionada pela arquitetura SDN aliada a programabilidade e visão centralizada chamaram a atenção de profissionais de TI e pesquisadores do mundo todo, movendo rapidamente o conceito para uma realidade [Nadeau and Gray 2013]. A divisão entre o plano de controle e plano de dados permite que a tomada de decisões antes realizada nos dispositivos, seja efetuada em outro ponto da rede através de software [Guedes et al. 2012]. A inteligência para tomada de decisões está concentrada no plano de controle, executada por uma entidade de rede chamada de controlador [Nadeau and Gray 2013]. Uma das primeiras abordagens que fundamentou o paradigma SDN foi a definição do protocolo OpenFlow [OpenFlow Specification-Version 2013]. O Openflow pode ser utilizado para prover diversas funcionalidades nas infraestruturas de rede, como por exemplo a implementação gradual de recursos e funcionalidades relacionadas a segurança.

As redes de computadores, tanto as tradicionais quanto as SDN, necessitam de mecanismos de segurança. Um dos mecanismos de segurança aplicado as rede de computadores é a autenticação de portas, a qual restringe o acesso não autorizado de dispositivos a uma rede local [Barros and Foltran Junior 2008]. As estações finais que se conectam a uma rede de computadores nem sempre serão confiáveis, podendo apresentar inúmeras vulnerabilidades. O protocolo OpenFlow, apesar de largamente utilizado em SDN, não possui um mecanismo nativo para autenticação segura da origem. Por padrão, as estações

finais se autenticam a uma rede OpenFlow através da validação de seus endereços MAC (*M*edia Access Control) ou IP (*I*nternet Protocol), com base na lógica de comutação de pacotes definida pelo administrador da topologia.

O presente artigo propõe a implemetanção de uma infraestrutura de chaves públicas aliada a utilização de certificados auto-assinados como possível solução ao problema de autenticação da origem em uma SDN OpenFlow.

2. O Paradigma das Redes Definidas por Software

O paradigma SDN desacopla o controle de encaminhamento dos dados e estreita a interação entre aplicações, dispositivos e serviços de rede, sejam estes reais ou virtualizados [Guedes et al. 2012]. Uma SDN possibilita a seu administrador o desenvolvimento de uma lógica centralizada, implementação gradual de recursos e visão global da rede, consolidando as ações em um único ponto de controle. A rede torna-se independente dos fabricantes de equipamentos pois os dispositivos são responsáveis apenas por executar as decisões previamentes definidas pelo controlador [Nadeau and Gray 2013]. O plano de controle é responsável por estabelecer um conjunto de dados utilizado para a criação de uma tabela de encaminhamento. Essa tabela é utilizada pelo plano de dados para encaminhar o tráfego no destino correto [Guedes et al. 2012]. O plano de dados é responsável pela comutação e repasse dos datagramas na rede. Tal plano opera em nível de link coletando os datagramas entrantes via inúmeros meios físicos, tais como fibra óptica, cabeado ou sem fio [Nadeau and Gray 2013].

Em uma arquitetura SDN, interfaces *Southbound* são usadas para a comunicação entre o controlador e o plano de dados, possiblitando alterações na rede em tempo real. Tais interfaces podem ser de código aberto ou proprietárias e constituem uma camada de abstração para o controle e gerenciamento da topologia. Umas das formas de implementar interfaces *S*outhbound é através do protocolo OpenFlow. Os elementos chave do protocolo OpenFlow tornaram-se parte da definição comum de SDN. Tais elementos são a separação do plano de controle e plano de dados. O OpenFlow estabelece um protocolo padrão, que age como mediador entre o controlador e os equipamentos da rede, por meio de uma API moderna e extensível [OpenFlow Specification-Version 2013]. É importante salientar que o OpenFlow é um conjunto de protocolos e uma API. Não devemos categorizar o OpenFlow como um produto por si só, ou uma única característica de um produto, pois o controlador não executa nada sem uma API para determinar as regras como os pacotes devem ser comutados [Nadeau and Gray 2013].

3. Desafios de Segurança em SDN

Ao mesmo tempo em que a arquitetura SDN apresenta-se promissora, existem alguns desafios de segurança a serem transpostos na implementação de tal tecnologia. A centralização do plano de controle traz inúmeras vantagens, como programabilidade, lógica centralizada e visão global da rede [Nadeau and Gray 2013]. Tais atributos representam significativos benefícios, porém aumentam a exposição do controlador e suas aplicações a ataques de DOS e interceptação de fluxos. Em uma rede OpenFlow, todo pacote é analisado, estando um pacote com o cabeçalho não associado aos fluxos existentes, é enviado para inspeção do controlador. Caso algum comutador da topologia envie uma quantidade incomum de novos cabeçalhos de pacotes para o controlador, tal entidade poderia ter seus recursos de processamento esgotados[Kreutz et al. 2013].

A necessidade de autenticação dos elementos de rede não é exclusividade de uma SDN, tão pouco um novo paradigma, redes convencionais sem separação do plano de dados e controle apresentam o mesmo tipo de problema [Kreutz et al. 2013]. A questão peculiar em uma SDN é o aumento da criticidade desse fato, pois estando um nó da rede comprometido, alvos como o controlador e estação de gerenciamento podem ser alcançados, tornando a rede vulnerável [Kreutz et al. 2013]. O protocolo OpenFlow provê programabilidade a rede, porém, não possui um mecanismo nativo para autenticação de portas [OpenFlow Specification-Version 2013]. Por padrão, as estações finais se autenticam a uma rede OpenFlow através da validação de seus endereços MAC (*M*edia Access Control) ou IP (*I*nternet Protocol), com base na lógica de comutação de pacotes definida pelo administrador da topologia. Um atacante poderia obter o endereço MAC ou IP de uma estação legítima e configurar esses dados em uma estação invasora. Caso o acesso fosse bem sucedido, a estação maliciosa poderia explorar aplicações vulneráveis no plano de controle ou disparar um ataque de DOS contra o controlador, consumir recursos de link, memória e processamento dos comutadores, gerando indisponbilidade de serviços na topologia.

4. Proposta para Autenticação de Portas em SDN

Em um ambiente de rede onde o meio físico é compartilhado ou aberto, como nas redes sem fio e redes cabeadas, a necessidade de confiança nas estações finais torna-se um aspecto fundamental na topologia [Barros and Foltran Junior 2008]. Uma forma encontrada para resolução desse problema foi o desenvolvimento de protocolos para autenticação de portas, provendo controle de acesso a uma rede computacional. O padrão 802.1X é amplamente utilizado em mecanismos para autenticação de portas, tal padrão provê autenticação entre clientes de rede e os equipamento nos quais estão conectados.

O padrão IEEE 802.1X possibilita o acesso autenticado em redes Ethernet, Token Ring e redes sem fio padrão 802.11, também oferece suporte ao protocolo RADIUS *(remote authentication dial in user service support)*. Até que o cliente esteja devidamente autenticado, o controle de acesso 802.1X habilita somente o tráfego do protocolo EAP *Estensible Authentication Protocol* na porta onde a estação estiver conectada. O EAP apresenta uma alternativa para interligação de redes devido a sua capacidade de adaptação a novos mecanismos de autenticação e pode por exemplo ser utilizado em conjunto com o protocolo TLS para implementações onde sejam utilizados certificados digitais [Barros and Foltran Junior 2008]. O EAP-TLS usa certificados padrão X.509 para verificar a identidade do usuário, aplicação ou estação de trabalho, supurtando ainda autenticação mútua.

Um modo de prover autenticação, integridade e confidencialidade em uma SDN é através da implementação de uma infraestrutura de chaves públicas. Tal infraestrutura é uma composição de segurança cujos serviços são executados e entregues utilizando conceitos e técnicas de criptografia assimétrica [Adams and Lloyd 2003]. Isso significa que para a encriptação de uma mensagem serão necessárias uma chave pública e uma chave privada. As infraestruturas de chaves públicas são responsáveis pela distribuição e gerenciamento de tais chaves. Podemos dizer ainda que tais infraestruturas foram desenvolvidas para autenticar e identificar usuários e serviços, garantindo que as informações trocadas estejam disponíveis apenas as entidades autorizadas, assegurando que se uma entidade realizar uma ação, não poderá negar que a realizou [Adams and Lloyd 2003].

O presente artigo propõe o desenvolvimento de um mecanismo de autenticação para estações finais em uma SDN OpenFlow através da implementação de uma infraestrutura de chaves públicas. O mecanismo proposto irá adotar o padrão 802.1x para realizar a troca de mensagens com as estações de trabalho na camada de enlace. As estações deverão enviar um pacote 802.1x com o método de autenticação EAP-TLS ao controlador. O mecanismo de autenticação deverá realizar a troca de mensagens 802.1x com a estação e receber um pacote com conteúdo EAP proveniente da estação suplicante. Posteriormente o mecanismo de autenticação deverá encaminhar a requisição para o servidor RADIUS, que por sua vez, irá consultar a raiz de certificação e identificar se a estação possui um certificado válido. A raiz de certificação é responsável pela emissão, gerenciamento e revogação de certificados para a SDN. A assinatura dos certificados será realizada pela autoridade certificadora, que atesta a autenticidade do certificado através de sua chave privada. Caso as credenciais estejam corretas, a estação de trabalho recebe uma mensagem de sucesso e o processo de autenticação é concluído. Por fim, as tabelas de encaminhamento serão criadas por um mecanismo de controle de fluxo e repassadas ao comutador que passará a tratar os pacotes oriundos dos dispositivos previamente autenticados.

5. Conclusão

Pela observação dos aspectos analisados nesse artigo, entende-se que a segurança de uma SDN necessita de mecanismos eficientes para autenticação e controle de acesso. A centralização lógica dos controladores e a criticidade de suas aplicações nos fazem entender quão importante é a autenticação da origem em uma arquitetura SDN. A falta de um mecanismo seguro para autenticação dos dispositivos finais poderia permitir que uma estação maliciosa disparasse ataques contra o controlador e provocasse indisponibilidade de recursos na rede. O desenvolvimento de um mecanismo de autenticação com base na utilização de certificados auto-assinados poderia garantir autenticação da origem em uma SDN, dificultando o acesso de estações maliciosas na topologia.

Referências

Adams, C. and Lloyd, S. (2003). *Understanding PKI: concepts, standards, and deployment considerations*. Addison-Wesley Professional.

Barros, L. G. and Foltran Junior, D. C. (2008). Autenticação ieee 802.1 x em redes de computadores utilizando tls e eap.

Guedes, D., Vieira, L., Vieira, M., Rodrigues, H., and Nunes, R. (2012). Redes definidas por software: uma abordagem sistêmica para o desenvolvimento de pesquisas em redes de computadores. *Minicursos do Simpósio Brasileiro de Redes de Computadores-SBRC 2012*, 30(4):160–210.

Kreutz, D., Ramos, F., and Verissimo, P. (2013). Towards secure and dependable software-defined networks. In *Proceedings of the second ACM SIGCOMM workshop on Hot topics in software defined networking*, pages 55–60. ACM.

Nadeau, T. D. and Gray, K. (2013). *SDN: Software Defined Networks*. "O'Reilly Media, Inc.".

OpenFlow Specification-Version, O. S. (2013). 1.4. 0.