

---

Aurelio Grott, Gabriel Dominico, Victor Lucas de M. Mafra

*Análise e solução de vulnerabilidades em ambiente LAMP  
baseada em experimentação com Kali Linux*

---

Joinville  
2016

**UNIVERSIDADE DO ESTADO DE SANTA CATARINA**  
**BACHARELADO EM CIÊNCIA DA COMPUTAÇÃO**

**Aurelio Grott, Gabriel Dominico, Victor Lucas de M. Mafra**

**ANÁLISE E SOLUÇÃO DE VULNERABILIDADES EM**  
**AMBIENTE LAMP BASEADA EM EXPERIMENTAÇÃO**  
**COM KALI LINUX**

Trabalho de conclusão de curso submetido à Universidade do Estado de Santa Catarina  
como parte dos requisitos para a obtenção do grau de Bacharel em Ciência da Computação

**Charles Christian Miers**  
**Orientador**

Joinville, Junho de 2016

# **ANÁLISE E SOLUÇÃO DE VULNERABILIDADES EM AMBIENTE LAMP BASEADA EM EXPERIMENTAÇÃO COM KALI LINUX**

Aurelio Grott, Gabriel Dominico, Victor Lucas de M. Mafra

Este Trabalho de Conclusão de Curso foi julgado adequado para a obtenção do título de Bacharel em Ciência da Computação e aprovado em sua forma final pelo Curso de Ciência da Computação Integral do CCT/UDESC.

Banca Examinadora

---

Charles Christian Miers - Doutor (orientador)

---

Charles Christian Miers - Doutor

---

Charles Christian Miers - Doutor

## Agradecimientos

*“We must know - we will know!”*

*- David Hilbert*

## Resumo

O **SDN!** (**SDN!**) é uma tecnologia recente que permite ao administrador de redes um maior controle sobre uma rede. Tal controle é obtido através da separação entre o *Control Plane* e *Data Plane*, o que caracteriza uma **SDN!**. Neste trabalho são conceituados diversos pontos-chaves relativos ao assunto, tais como *OpenFlow* e os planos do **SDN!**. Em seguida é descrita a ferramenta de simulação de redes *Mininet* e no fim do trabalho é descrito dois *benchmarks* com o objetivo de coletar dados para uma análise de desempenho. Tendo em vista que os *hardwares* que suportam **SDN!** são relativamente recentes, o seu custo é muitas vezes proibitivo para um pequeno grupo de pesquisa ou um pesquisador independente, de tal forma que o uso de simuladores se torna indispensável para o desenvolvimento científico e tecnológico na área. Este trabalho tem como objetivo realizar um comparativo entre a transferência de dados de tamanho médio dentro de um ambiente simulado no *Mininet* e um ambiente utilizando o modelo tradicional de rede (*Data Plane* e *Control Plane* acoplados).

Palavras-chaves: SDN, Openflow, Mininet, software livre, transferência

# Abstract

**SDN!** is a new technology which gives the network administrator greater power over his network. Such control is given through the separation between the control plane and the data plane, which characterizes an **SDN!**. In this paper it is conceptualized several key points relative to **SDN!**, such as *OpenFlow* and the **SDN!** planes. In the following section it is described the networking simulation tool *Mininet* following by the description of two benchmarks that will be used with the objective of data collection for later analysis. Since the hardware that supports **SDN!** are relatively new, their costs are very often prohibitively high for a small research group or an independent researcher, so that the use of simulators becomes indispensable for the technological and scientific development in the field. This paper has as its major objective to do a comparative between the transferring of medium sized data in a Mininet simulated environment and a traditional networking model (i.e. coupled Data Plane and Control plane).

Keywords: SDN, Openflow, Mininet, open source, trasnference

## Lista de Figuras



## Lista de Tabelas

## Lista de Siglas e Abreviaturas

**BD** Banco de dados

**CGI** Common Gateway Interface

**DNS** Domain Name System

**GNU** Gnu Not Unix

**GPL** General Public License

**HTTP** HyperText Transfer Protocol

**LAMP** Linux Apache MySQL PHP

**PHP** Hypertext Preprocessor

**SMTP** Simple Mail Transfer Protocol

**SQL** Structured Query Language

**UDESC** Universidade do Estado de Santa Catarina

# Sumário

<b>Lista de Figuras</b>	<b>5</b>
<b>Lista de Tabelas</b>	<b>6</b>
<b>Lista de Siglas e Abreviaturas</b>	<b>7</b>
<b>1 Introdução</b>	<b>9</b>
<b>2 Conceitos</b>	<b>10</b>
2.1 LAMP . . . . .	10
2.2 HISTÓRICO . . . . .	10
2.3 FUNCIONAMENTO E COMPONENTES BÁSICOS . . . . .	10
2.3.1 Linux . . . . .	10
2.3.2 Apache . . . . .	11
2.3.3 MySQL . . . . .	11
2.3.4 PHP . . . . .	12
2.4 APLICABILIDADE . . . . .	13
<b>3 Conclusão</b>	<b>14</b>
<b>Referências Bibliográficas</b>	<b>15</b>



## 2 Conceitos

### 2.1 LAMP

### 2.2 HISTÓRICO

### 2.3 FUNCIONAMENTO E COMPONENTES BÁSICOS

#### 2.3.1 Linux

Linux poderia ser descrito como um sistema operacional similar a qualquer outro, como Windows e OS X. Porém tem algo que o destaca, desde sua origem em 1991, e que é o motivo do sistema ter crescido e ganhado uma grande força na computação, atualmente presente em lugares desde a bolsa de valores de Nova York e supercomputadores à telefones celulares e computadores pessoais, o Linux é um software livre desenvolvido de maneira colaborativa (PROFFITT, 2009). Mais de 1.000 desenvolvedores de pelo menos 100 diferentes companhias, contribuíram para cada versão do kernel sob a licença General Public License (GPL) que é baseada em quatro liberdades (FSF, 2016):

- A liberdade de executar o programa como quiser, para qualquer propósito;
- A liberdade para estudar como o programa funciona, e alterá-lo para que ele execute como você queira. Ter acesso ao código fonte é necessário para tal;
- A liberdade para redistribuir cópias para ajudar o próximo; e
- A liberdade para distribuir cópias de suas versões modificadas para outros. Fazendo isso você concede à comunidade a chance de se beneficiarem de suas alterações. Ter acesso ao código fonte é necessário para tal.

Por esses motivos o Linux tem sido bem-sucedido, particularmente como plataforma de servidor: até mesmo em organizações que confiam veemente em sistemas operacionais comerciais como Microsoft Windows, o Linux aparece frequentemente em

papeis infraestruturais, como em *gateways* de Simple Mail Transfer Protocol (SMTP) e servidores Domain Name System (DNS) devido a sua confiança, segurança, baixo custo e a qualidade excepcional das aplicações do servidor (BAUER, 2005).

### 2.3.2 Apache

### 2.3.3 MySQL

O Banco de dados (BD) MySQL, foi projetado com base no mSQL, o qual tinha muitos problemas, como não ser rápido e flexível o suficiente para o uso dos usuários, com isso a necessidade de um novo BD foi aumentando e com base nesse conceito foi desenvolvido o que hoje conhecemos como MySQL.

Um BD pode ser definido como uma coleção de dados. Porém para conseguir acessar os dados armazenados nesse sistema, teve-se a necessidade de criar algum tipo de gerenciador, sendo o MySQL um dos mais usados. Algumas características (MYSQL, 2013a) desse sistema podem ser vistas abaixo:

- **Banco de dados relacional:** a principal diferença desse tipo de BD para os outros é que os dados são guardados em pequenas tabelas de uma forma que seu acesso seja da forma mais eficiente o possível.
- **Open Source:** esse termo corresponde que qualquer pessoa pode modificar o *software* do jeito que preferir, podendo ajustá-lo conforme a sua necessidade.
- **Rápido, confiável, escalável e fácil de usar:** como foi criado para atender a grandes quantidades de dados de uma forma mais rápida que seus concorrentes, foi apenas lógico que se tornasse um dos mais rápidos BD. Portanto começou a ser utilizado em grande escala, consequentemente a segurança foi aumentando juntamente com sua escalabilidade para atender a demanda de usuários.

Contudo, mesmo com medidas de segurança sendo tomadas, precisamos ainda tomar algumas atitudes para dificultar que seu BD seja acessado por pessoas não autorizadas, alguns métodos básicos que ajudam a proteger são descritas abaixo (MYSQL, 2013b):

- Não prover acesso a ninguém para a tabela usuário do BD MySQL.

- Não guardar senhas sem algum tipo de função *hash* (algoritmo usado para transformar sua senha para uma *string* ilegível).
- Crie senhas aleatórias, porém de fácil memorização.
- Invista em um *firewall*, protegem pelo menos 50% dos ataques feitos contra seu *software*.
- Sempre criptografe os dados que precisam ser enviados pela internet.

### 2.3.4 PHP

O Hypertext Preprocessor (PHP) foi criado em 1994 por Rasmus Lerdorf, o projeto inicial era um simples conjunto de Common Gateway Interface (CGI)s binários escritos na linguagem de programação C, usados para rastrear as visitas ao seu *site*. Com o tempo, otimizações foram sendo feitas e funcionalidades adicionadas. Sendo lançado em 1998, o PHP 3.0 foi a primeira versão que contém traços do PHP de hoje em dia, incluindo o suporte a programação orientada a objeto. Porém essa versão tinha muita dificuldade em processar aplicações complexas, foi com base nessa premissa que foram lançadas as versões 4.0 e 5.0 (Julho de 2004), principalmente para melhorar seu antecessor e acrescentar dezenas de novos recursos.

Usado principalmente para desenvolvimento *web*, é um *script open source* de uso geral. Podemos especificar em quais áreas os *scripts* PHP são mais utilizados (PHP, 2016), como:

- **Scripts no lado do servidor.** Podendo acessar os resultados do seu programa com um navegador web.
- **Scripts de linha de comando.** Executar os scripts sem um servidor ou navegador, apenas necessita de um interpretador PHP.
- **Escrever aplicações desktop.** Não é a melhor linguagem para se desenvolver aplicações desktop, porém para um programador experiente o PHP tem alguns recursos avançados que permitem escrever esse sistema.

Uma característica é a escalabilidade que o PHP possui, podendo ser utilizado na maioria dos sistemas operacionais e servidores *web*. Com isso ele vem sendo aplicado

---

cada vez mais, por suas várias extensões que facilitam a conectividade com diversos banco de dados.

## 2.4 APLICABILIDADE





## Referências Bibliográficas

BAUER, M. *Linux server security*. Sebastapol, CA Cambridge: O'Reilly, 2005. ISBN 978-0-596-00670-9.

FSF. *What is free software? - GNU Project - Free Software Foundation*. January 2016. <http://www.gnu.org/philosophy/free-sw.en.html>. (Accessed on 04/08/2016).

MYSQL. *MySQL :: MySQL 5.7 Reference Manual :: 1.3.1 What is MySQL?* April 2013. <https://dev.mysql.com/doc/refman/5.7/en/what-is-mysql.html>. (Accessed on 04/09/2016).

MYSQL. *MySQL :: MySQL 5.7 Reference Manual :: 6.1.1 Security Guidelines*. April 2013. <http://dev.mysql.com/doc/refman/5.7/en/security-guidelines.html>. (Accessed on 04/09/2016).

PHP. *PHP: O que o PHP pode fazer? - Manual*. March 2016. [http://php.net/manual/pt\\_BR/intro-whatcando.php](http://php.net/manual/pt_BR/intro-whatcando.php). (Accessed on 04/09/2016).

PROFFITT, B. *What Is Linux: An Overview of the Linux Operating System — Linux.com — The source for Linux information*. April 2009. <https://www.linux.com/learn/what-linux-overview-linux-operating-system>. (Accessed on 04/09/2016).