**UNIVERSIDADE DO OESTE DE SANTA CATARINA – UNOESC**

**CURSO DECIÊNCIA DA COMPUTAÇÃO**

**FERRAMENTA PARA ANÁLISE DE LOGS DA PLATAFORMA MIKROTIK**

RAFAEL ADRIANI BARZOTTO

Videira – SC, 2014

**RAFAEL ADRIANI BARZOTTO**

**FERRAMENTA PARA ANÁLISE DE LOGS DA PLATAFORMA MIKROTIK**

Projeto de Monografia apresentada ao Curso de Ciência da Computação da Universidade do Oeste de Santa Catarina como requisito parcial para a disciplina de TCC II

Videira – SC, 2014

**SUMÁRIO**

[1 INTRODUÇÃO 5](#_Toc398908800)

[1.1 OBJETIVO GERAL 6](#_Toc398908801)

[1.2 OBJETIVOS ESPECÍFICOS 6](#_Toc398908802)

[1.3 JUSTIFICATIVA 6](#_Toc398908803)

[2 REVISÃO BIBLIOGRÁFICA 8](#_Toc398908804)

[2.1 SEGURANÇA DA INFORMAÇÃO 8](#_Toc398908805)

[2.2 GERENCIAMENTO DE REDE 8](#_Toc398908806)

[2.3 SISTEMAS DE REGISTROS DE EVENTOS 9](#_Toc398908807)

[2.4 SYSLOG 11](#_Toc398908808)

[2.5 MIKROTIK 12](#_Toc398908809)

[2.6 SENCHA EXT JS 13](#_Toc398908810)

[2.7 ARQUITETURA MVC 13](#_Toc398908811)

[2.7.1 MODEL – CAMADA DE ABSTRAÇÃO DE DADOS 14](#_Toc398908812)

[2.7.2 VIEW – CAMADA DE APRESENTAÇÃO 14](#_Toc398908813)

[2.7.3 CONTROLLER – CAMADA DE NEGÓCIOS 14](#_Toc398908814)

[3 CRONOGRAMA 15](#_Toc398908815)

[4 ORÇAMENTO 16](#_Toc398908816)

[5 CONCLUSÃO 17](#_Toc398908817)

[6 REFERENCIAS BIBLIOGRAFICAS 18](#_Toc398908818)

# introdução

Devido à utilização da Internet nos mais diversos ambientes, a responsabilidade do administrador de ter o controle sobre o que acontece aos recursos e informações da rede aumentou. Todas as questões relacionadas a usuários, garantir a segurança do sistema e evitar abusos está ligada diretamente ao monitoramento.

Desde modo, a segurança tem se tornado cada vez mais importante, isso porque as ferramentas de captura de trafego e exploração de vulnerabilidades estão cada vez mais avançadas. Como alternativa para manter a segurança do sistema são usadas ferramentas de proteção como *Firewall* e *Proxy*, mas aliado a essas ferramentas o administrador precisar manter-se atualizado de possíveis falhas e atividades suspeitas na rede, e isso é possível através dos *logs* de acesso.

*Logs* são arquivos no formato de texto onde a maioria dos serviços executados registra suas atividades. A capacidade de analisar *logs* possibilita o monitoramento de atividades e futuras observações. Assim, os *logs* também servem como meio de prevenção. Estes arquivos possuem um tamanho relativamente grande o que o torna inviável a análise de forma manual.

O objetivo desse trabalho é desenvolver uma ferramenta para auxiliar na análise de *logs* gerados pela plataforma Mikrotik, com a possibilidade de interagir com registros de serviços distintos e a centralização em um banco de dados. Através disso podem-se apresentar os dados de forma clara e prática.

## Objetivo Geral

Desenvolver uma aplicação capaz de ler e analisar os arquivos de *log* do sistema embarcado RouterOS da MikrotikLtd.

## Objetivos Específicos

* Planejar o armazenamento dos arquivos de *log* em um servidor com recursos computacionais eficientes;
* Planejar uma aplicação capaz de ler os arquivos de *log*;
* Desenvolver uma ferramenta de análise de *logs* capaz de filtrá-los de acordo com a necessidade requerida e mostrá-los;
* Demonstrar os resultados adquiridos;

## JUSTIFICATIVA

A análise de *logs* de acesso de um sistema permite identificar, para cada tema, o volume de acessos, perfil dos usuários, temas mais consultados, além de garantir a segurança do sistema, pois registram todas as informações e eventos que forem detectados. Os arquivos de *log* podem ser um meio eficiente de analisar incidentes e peculiaridades.

No sistema RouterOS operante nos produtos Mikrotik existe a ferramenta de geração de log, porém as placas chamadas RouterBoards (que são o hardware do sistema da Mikrotik) não possuem memória suficiente para armazenar muito conteúdo, sendo assim a segurança e a auditoria ficam comprometidas.

Encontra-se também dificuldade em ler os arquivos de *log* gerados pelo sistema RouterOS. Não existe nenhuma aplicação especifica que auxilie na análise de dados gerados pelo sistema em interesse.

Por tanto, justifica-se o presente trabalho, dada a importância dos arquivos de *log*.

# REVISÃO BIBLIOGRÁFICA

Nessa seção será abordado toda a teoria necessária para o entendimento sobre segurança da informação, sistemas de registros de eventos, também detalhes de seu funcionamento e sua importância dentro da tecnologia da informação.

## SISTEMAS DE INFORMAÇÃO

É um conjunto de componentes, dados ou pessoas que estão inter-relacionados e são capazes de processar, armazenar e disseminar informações a todo tipo de ativo, essas informações são usadas para coordenação, controle e até fornecer apoio à tomada de decisões dentro de uma organização.

Sistemas de Informação são todos os sistemas que produzem ou geram informações, que são dados trabalhados (ou com valor atribuído ou agregado a eles) para execução de ações e para auxiliar processos de tomada de decisões. (REZENDE, 2005, p. 21)

Um sistema de informação podem conter dados referentes a qualquer tipo de coisa, podem ser dados de pessoas, locais, documentos, etc. Um exemplo são os dados armazenados em um sistema bancário, nele serão encontrado informações pessoais, endereços, número de contas, senhas e até o saldo.

Levando em conta que toda a informação é crucial para a organização e hoje em dia o mundo está totalmente ligado a tecnologia dá-se um grande enfoque na segurança da informação.

## segurança da informação

A área de segurança de rede atua de modo em que “vilões” e possíveis ataques sejam defendidos ou evitados.

Como relata Ferreira (2003, p. 1) “A informação é um ativo que, como qualquer outro ativo importante, tem um valor para a organização e, consequentemente, necessita ser adequadamente protegido”.

Como a frequência e a variedade de ameaças são incontáveis a segurança e gerenciamento da rede torna-se atualmente um assunto muito focado e importante.

### Objetivos da Segurança da Informação

A informação é de fato um dos maiores bens de um negócio, é a partir dela que se sabe tudo o que acontece e poderá acontecer, no caso ao usar processos de apoio à decisão baseado em informação. Por ser um recurso crucial para a sobrevivência de organizações, todos os ativos que trabalham com informações devem ser mantidos seguros.

## REDEs

## GERENCIAMENTO DE REDE

Usinas de energia elétrica possuem uma sala de controle onde existem diversos mostradores, medidores, *leds*, tudo isso para que o administrador possa monitorar o seu funcionamento. Em uma rede também partimos desse princípio, porém, para descobrir algum problema iminente o operador deve analisar os dados gerados a partir de seus equipamentos.

O objetivo da Gerência de Redes é monitorar e controlar os elementos da rede (sejam eles físicos ou lógicos), assegurando certo nível de qualidade de serviço. Para realizar esta tarefa, os gerentes de redes são geralmente auxiliados por um sistema de gerência de redes. Um sistema de gerência de rede pode ser definido como uma coleção de ferramentas integradas para a monitoração e controle da rede. Este sistema oferece uma interface única, com informações sobre a rede e pode oferecer também um conjunto poderoso e amigável de comandos que são usados para executar quase todas as tarefas da gerência da rede. (STALLINGS, 1998 apud SUAVÉ; LOPES; NICOLLETTI, 2003, p. 25).

No início das redes de computadores, onde não era utilizado por milhares ou milhões de pessoas, o gerenciamento da rede era prático, com realizações de alguns testes ou o reinicio dos equipamentos. Hoje com a tecnologia em alta e com grandes estruturas de redes, viu-se necessário gerenciar todos os equipamentos de modo eficiente.

Dentro deste contexto, COSTA (2008, p. 6) afirma:

Ter um ambiente mapeado e monitorado é fundamental para o processo de crescimento de uma empresa, já está mais do que comprovado que com um ambiente de T.I bem planejado seu negócio tem chances de dar certo, mesmo para as empresas em que o principal foco seja T.I, pois todos dependem hoje da Internet e dos serviços que ela disponibiliza.

A gerência de rede tem o apoio dos registros de atividade gerados pelos seus equipamentos, esses registros chamados *logs*, dão a possibilidade de tomar medidas preventivas e corretivas para o bom andamento da rede.

## sistemas de registros de eventos

Segundo Neto (2003) um evento é algum fato que possa gerar sensação e, por isso, ser motivo de notícia. Descrevendo mais a fundo, um fato é algo que acontece, tem datas e horários de início e fim.

Sistema de registros de eventos também chamados de *logs* tem como objetivo proporcionar uma “radiografia” completa do funcionamento de um processo, programa, hardware, entre outros.

Os logs de acesso e atividade são importantes para a administração da segurança de redes e sistema, pois registram informações sobre seu funcionamento e sobre eventos por eles detectados. Muitas vezes, os logs são o único recurso que um administrador possui para descobrir as causas de um problema. (FERREIRA, 2003, p. 56)

Os *logs* podem ficar armazenados em arquivos, em base de dados ou na memória do computador. Normalmente os logs são armazenados na mesma máquina em que são gerados, porém, isso possui alguns riscos. Um desses riscos é que em caso de invasão os *logs* podem ser destruídos. Um meio de contornar isso é a criação de um *loghost*, que é um sistema dedicado a coleta e armazenamento de *logs* de outros equipamentos e sistemas da rede.

Segundo o Centro de Estudos, Resposta e Tratamentos de Incidentes de Segurança no Brasil (2012):

A partir da análise de *logs* pode ser capaz de:

* Detectar o uso indevido de um computador, de um usuário tentando acessar arquivos de outro usuário do sistema;
* Detectar um ataque, como de força bruta ou a exploração de vulnerabilidade;
* Rastrear (auditar) as ações executadas por um usuário em um computador, como programas utilizados, comandos executados e tempo de uso do sistema;
* Detectar problemas de hardware ou nos programas e serviços instalados no computador.

Baseado nessas informações o administrador da rede pode tomar medidas preventivas para evitar que um futuro problema ocorra, como:

* Se um atacante estiver tentando explorar alguma vulnerabilidade, pode-se verificar se as medidas preventivas já foram aplicadas e tentar evitar que o ataque ocorra;
* Se não for possível evitar um ataque, os *logs* podem permitir que as ações executadas pelo atacante sejam rastreadas, como arquivos alterados e informações acessadas.

*Logs* são essenciais para notificações de incidentes, pois permitem que diversas informações importantes sejam detectadas, como por exemplo: a data e o horário em que uma determinada atividade ocorreu, o fuso horário do *log*, o endereço IP de origem da atividade, as portas envolvidas, o protocolo utilizado no ataque (TCP, UDP, ICMP, etc.), os dados completos que foram enviados para o computador ou rede e o resultado da atividade (se ela ocorreu com sucesso ou não).(CERT.BR, 2012)

## SYSLOG

Syslog é um padrão criado pela comunidade internacional IETF (*Internet EngineeringTask Force)* para transmissão de logs via rede. Seu funcionamento é simples, o equipamento configurado com suporte ao Syslog envia uma mensagem de texto (com menos de 1024 bytes) para um servidor remoto (syslogd) que fará o armazenamento do texto.

O Syslog é muito utilizado para o gerenciamento da rede e também para auditorias de segurança. Suportado por diversas plataformas e sistemas, assim como no Linux e seus derivados.

Desenvolvido na década de 1980 por Eric Allman, inicialmente era um parte do projeto do *SendMail*, porém viu-se que era um aplicação de tamanha importância que mais tarde acabou se tornando a solução de registro padrão do Unix, ainda também houve adições do Syslog em roteadores e outros equipamentos de rede.

Visto que de fato o Syslog era uma ferramenta ótima o IETF criou como sendo um protocolo na documentação descritiva de padrões de protocolos RFC 3164, logo mais tarde com algumas alterações surgiu o RFC 5424.

## MIKROTIK

Mikrotik é uma empresa da Letônia que desenvolve softwares e hardwares para Gerenciamento de Redes e para Provedores de Internet. Seu principal produto é o sistema operacional baseado em Linux chamado MikrotikRouterOSEVERYBODYou apenas RouterOS.

Seu Sistema Operacional pode ser instalados nos *hadwares* proprietários chamados de RouterBoards ou em um computador de arquitetura x86.

Inicialmente seus produtos eram projetados para provedores de pequeno porte, hoje já oferece equipamentos para grande escala e produtividade.

Todas essas funções a seguir podem ser encontradas em produtos Mikrotik:

* Roteador dedicado;
* Bridge com filtros em layer2;
* Firewall com layer7 e diversos filtros;
* Controle de velocidade, garantia de banda, burst, hierarquia e disciplinas de filas;
* Ponto de Acesso Wireless em modo 802.11 e proprietário, cliente wireless, WDS, NSTREME, NSTREME Dual;
* Concentrado PPPoE, PPtP, IPSeC, L2TP, etc;
* Roteador de borda;
* Hotspot com gerenciamento de usuários;
* Recursos de Bonding, VRRP, etc;
* Linguagem avançada de scripts;
* Roteamento com OSPF, MPLS, BGP, etc;
* Ferramentas: watchdog, badwidth test, toch, syslog.

(MEDIAWIKI, 2014)

## SENCHA EXT JS

Sencha EXT JS é um conjunto de bibliotecas *JavaScript* para desenvolver aplicativos para internet. Ele combina recursos de HTML5, *JavaScript* e CSS3 para permitir a criação de aplicações web ricas e com uma interface similar a sistemas desktop, o que facilita o seu uso.

Criado em 2006 por Jack Slocum evoluiu rapidamente devido ào apoio da comunidade de colaboradores.

O EXT JS hoje mantido pela empresa Sencha é encontrado em duas versões, sendo a versão com licença para uso comercial e a versão com licença *open-source* (GPL v3).

Com esse *framework* é possível fazer o desenvolvimento *front-end* da aplicação e de forma prática integrar o *back-end* (servidor), que pode ser qualquer um que trabalhe com requisições Ajax, como por exemplo: PHP, Strunts, Spring, v-raptor, Rails, Python, ASP.NET.

## ARQUITETURA DE SOFtware

Para trabalhar com o tamanho e a complexidade de sistemas os engenheiros de softwares usam princípios e métodos elaborados para manter a disciplina e obter resultados de qualidade dentro do desenvolvimento de softwares. Esses métodos são parte da Arquitetura de Software

Com base em discussões realizadas no Software Engineering Institute da Carnegie Mellon University, David Garlan e Dawayne Perry (Garlan, 1995) definiram arquitetura de software como:

“A estrutura dos componentes de um programas/sistema, seus inter-relacionamentos, princípios e diretrizes guiando o projeto e evolução ao longo do tempo.”

E completando, Pressmann (2006, p. 443) descreve a arquitetura de uma aplicação Web como uma infraestrutura que possibilite o sistema atingir objetivos de negócio.

Existem várias arquiteturas que podem ser usadas, no entanto, o autor acima citado sugere que seja usada uma arquitetura em três camadas nomeada MVC (Modelo, Visão e Controlador).

### Arquitetura MVC

O MVC, *Model*, *View* e *Controller* (Modelo, Visão e Controlador) separa a aplicação em três segmentos lógicos, onde cada um tem uma responsabilidade. O núcleo do *framework* é responsável por fazer a comunicação entre estes segmentos.

A arquitetura MVC pode ser utilizada no projeto e na implementação de sistemas iterativos oferecendo a vantagem da modularidade, o que possibilita tanto ajudar o desenvolvimento conceitual das aplicações quanto permitir que partes já desenvolvidas para uma aplicação possam ser reutilizadas em uma nova aplicação. (MENDES, 2003, p. 134).

#### Model – Camada de Abstração de Dados

A camada denominada *Model* (Modelo) gerencia o comportamento e os dados da aplicação. Responde a pedidos de informação (geralmente das *views*) e responde às instruções para a mudança de estados (geralmente dos *controllers*).

#### View – Camada de Apresentação

A camada *View* (Visão) é responsável por renderizar (formatar e gerar interfaces) as informações recebida dos controladores e modelos e apresentar o resultado (HTML) ao visitante do site. São as *Views* que fornecem formulários, tabelas, campos, botões e todos os demais componentes visuais.

É importante ressaltar que as telas devem ser livres de toda logica de negócio/programação (nela deve ter apenas a parta básica para impressão de dados nas marcações HTML). Qualquer interação com a camada Modelo (*Model*) deve ser delegada para a camada do *Controller*, esta que vai invocar e repassar os dados para a *View* adequada, mantendo a parte visual desacoplada da lógica da aplicação.

#### Controller – Camada de Negócios

Os controladores, como o próprio nome sugere, controlam a aplicação. As regras de negócio são definidas principalmente nos controladores. Eles processam os pedidos, manipulam os modelos ou até mesmo outro controlador.

O *Controller* trabalha com as requisições web, dados de formulário, pedido de informações ao *Model* e reenviando ao *Model* todas as informações a serem salvas no banco de dados por exemplo.

Na arquitetura MVC, o componente *controller* comunica-se diretamente com os componentes da *view* e *model*. Uma ilustração da arquitetura MVC é dada na Figura 1.

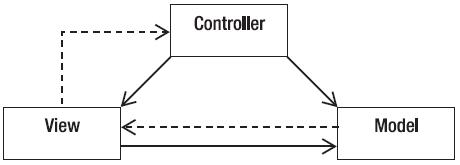


Figura 1

## BANCO DE DADOS

Bando de dados é definido por um conjunto de dados gravados em uma mídia apropriada, organizados em coleções que se relacionam entre si e representam algo que tenha sentido. Quem permite manusear estes dados é o Sistema Gerenciador de Banco de Dados (SGBD).

O SGBD permite construir e manter um banco de dados, possibilitando ao profissional responsável por essa área a alteração da estrutura existe, a criação de novas estruturas, execução de teste, back-up e restauração de dados, analise de performance, entre outras ações. Alguns exemplos de SGBDs disponíveis no mercado: Interbase, SQL Server, MS Access, MySQL, FireBird, Oracle.

### MySQL

O MySQL surgiu da necessidade de controlar a conexão e as tabelas criadas com a linguagem SQL (*Structured Query Language*), pois a ferramenta que era utilizada não era eficiente.

Por volta de 1979 Michael Widenius na companhia suíça TcX criou um banco de dados chamado UNIREG, porém ao decorrer do projeto percebeu que seu banco não era tão eficiente e recorreu a outro sistema, o mSQL, que por sua vez era ainda inferior ao UNIREG. Foi nesse ponto que surgiu a ideia de unir os dois bancos de dados. Utilizando a estrutura e API´s de ambos os bancos nascia a primeira versão do MySQL no ano de 1995.

O MySQL utiliza a linguagem SQL, que é a linguagem padrão para acessar SGBD relacionais. A SQL é utilizada para armazenar e recuperar dados a partir do bando de dados.

O MySQL além de ser um banco de dados é um SGBD relacional multiencadeado, de código fonte aberto e de nível corporativo.

Os principais motivos para o uso do MySQL na elaboração do sistema em foco são:

* Alto Desempenho;
* Baixo Custo;
* Facilidade de Uso;
* Portabilidade;

## PHP

# 

# CONCLUSÃO

Os estudos realizados em cima de deste trabalho, oferecem vasto conhecimento sobre tais ferramentas, que notavelmente são de grande importância para a programação de computadores.

Por esse motivo conhecer essa base dará mais flexibilidade e auxilio para o futuro, tanto para as próximas partes a serem estudadas quanto no desenvolvimento deste projeto.

# referencias

CERT.BR, **Cartilha de Segurança para Internet**: Mecanismos de Segurança. Brasília, DF, 2012

COSTA, F..**Ambiente de redes monitorados com nagios e cacti**. Rio de Janeiro: Ciência Moderna, 2008.

COSTALES, Bryan; SPAFFORD, Gene. **SendMail**. 2. Ed. San Francisco: California. O’Reilly & Associates, Inc. 1996.

FERREIRA, Fernando Nicolau Freit. **Segurança da Informação**. Rio de Janeiro: Ciência Moderna, 2003.

IERF, **O Protocolo Syslog – RFC 5424**. 2009.

MEDIAWIKI, **Mikrotik Manual**. Riga, Letônia, 2014

MENDES, Antônio. **Arquitetura de Software – Desenvolvimento orientado para arquitetura**. Rio de Janeiro: Campus, 2002.

RNP**, Os Logs como Ferramenta de Detecção de Intrusão**. Brasília, 2014.

SUAVÉ, J.P; LOPES, R.V; NICOLLETTI, P.S. **Melhores práticas para a gerência de redes de computadores**. 1. Ed. Rio de Janeiro: Campus.

WEINBERG, Joshua. **Centralized Logging with Syslog.** The Journal for UNIX Administrators, Vol. 7, Nº 10. 1998