



UNIVERSIDADE FEDERAL DO RIO GRANDE DO NORTE
CENTRO DE CIÊNCIAS EXATAS E DA TERRA
DEPARTAMENTO DE INFORMÁTICA E MATEMÁTICA APLICADA
BACHARELADO EM ENGENHARIA DE SOFTWARE



Sistema de Detecção Automatizada de Incidentes de Redes Decorrentes das Mudanças nas Configurações dos Switches

Silas Tiago de Medeiros

Natal-RN
Junho de 2017

Silas Tiago de Medeiros

Sistema de Detecção Automatizada de Incidentes de Redes Decorrentes das Mudanças nas Configurações dos Switches

Monografia de Graduação apresentada ao Departamento de Informática e Matemática Aplicada do Centro de Ciências Exatas e da Terra da Universidade Federal do Rio Grande do Norte como requisito parcial para a obtenção do grau de bacharel em Engenharia de Software.

Orientador

Prof.º Dr. Marcos César Madruga Alves Pinheiro

UNIVERSIDADE FEDERAL DO RIO GRANDE DO NORTE – UFRN
DEPARTAMENTO DE INFORMÁTICA E MATEMÁTICA APLICADA – DIMAP

Natal-RN
Junho de 2017

Catálogo da Publicação na Fonte. UFRN / SISBI / Biblioteca Setorial
Centro de Ciências Exatas e da Terra – CCET.

Medeiros, Silas Tiago de.

Sistema de detecção automatizada de incidentes de redes decorrentes das mudanças nas configurações dos switches / Silas Tiago de Medeiros. - Natal, 2017. 61f.: il.

Orientador: Prof. Dr. Marcos César Madruga Alves Pinheiro.

Monografia (Graduação) – Universidade Federal do Rio Grande do Norte. Centro de Ciências Exatas e da Terra. Departamento de Informática e Matemática Aplicada. Bacharelado em Engenharia de Software.

1. Incidentes de redes – Monografia. 2. Gerência de redes – Monografia. 3. ITIL – Monografia. 4. Gerência de configuração – Monografia. I. Pinheiro, Marcos César Madruga Alves. II. Título.

RN/UF/BSE-CCET

CDU: 004.77

Monografia de Graduação sob o título *Sistema de Detecção Automatizada de Incidentes de Redes Decorrentes das Mudanças nas Configurações dos Switches* apresentada por Silas Tiago de Medeiros aceita pelo Departamento de Informática e Matemática Aplicada do Centro de Ciências Exatas e da Terra da Universidade Federal do Rio Grande do Norte, sendo aprovada por todos os membros da banca examinadora abaixo especificada:

Prof.º Dr. Marcos César Madruga Alves Pinheiro
Orientador
DIMAp
UFRN

Prof. MSc. Aluizio Ferreira da Rocha Neto - Mat.
2149456
Convidado
IMD
UFRN

Prof. MSc. Carlos Gustavo Araújo da Rocha - Mat.
1379492
Convidado
IFRN

Natal-RN, 17 de junho de 2017.

A todas as pessoas que lutam por um mundo melhor.

Agradecimentos

Todo trabalho acadêmico exige muito esforço intelectual e físico, mas quando falamos do trabalho de conclusão de curso é preciso enfatizar o quanto esse trabalho exige do psicológico e de tempo investido. A longa jornada de desenvolvimento, que passa desde a fase de formulação e definição do tema e vai até a apresentação e avaliação da banca examinadora, é muito pesada para ser carregada sozinha.

No meu caso não foi diferente, apesar de todas as dificuldades eu consegui superar graças ao apoio de uma equipe de peso. Primeiro agradeço a Deus, pois ele com seu imenso amor sempre concede os nossos pedidos e nos orienta a seguir os caminhos corretos. Minha eterna gratidão a minha família, Francisca e Moacir que são meus pais, Gledson meu irmão, e Flavia Nanci (Flavinha) minha esposa e todos os tios, e primos. Todos eles são e sempre serão o meu pilar.

Gostaria de agradecer também aos amigos da SINFO, que na hora que precisei de alguma ajuda em relação ao meu trabalho, sempre estavam ali prontos a me ajudar.

Por fim, gostaria de evidenciar minha gratidão a todo corpo docente, administrativo e de apoio do DIMAp, em especial o meu orientador Marcos César Madruga Alves Pinheiro, que juntos possibilitaram a minha formação no curso de Engenharia de Software.

“Se você pensa que pode ou se pensa que não pode, de qualquer forma você está certo.”

Henry Ford

Sistema de Detecção Automatizada de Incidentes de Redes Decorrentes das Mudanças nas Configurações dos Switches

Autor: Silas Tiago de Medeiros

Orientador : Prof.º Dr. Marcos César Madruga Alves Pinheiro

RESUMO

A detecção de incidentes em redes não é uma tarefa simples, principalmente em redes de médio e grande porte, de modo que o papel de detectar e solucionar problemas fica a cargo de funcionários especializados. Idealmente, a detecção e o tratamento dos incidentes devem ocorrer no menor intervalo de tempo possível, conforme pré-estabelecido pelos Acordos de Nível de Serviço (ANS). Entre os diversos fatores que colaboram para o atendimento dos tempos definidos em um ANS, como o volume total de chamados em atendimento, e a existência de procedimentos bem definidos, um dos que merecem destaque é a garantia de que a configuração real dos ativos está de acordo com o especificado na documentação mantida pela equipe de TIC. Tal garantia é necessária para todos os tipos de ativos, e em especial para os switches, pois embora tipicamente não sofram alterações tão frequentemente, muitas vezes são gerenciados por pessoas/equipes de TIC diferentes. A proposta deste trabalho é a confecção de um sistema de monitoramento de configurações, que apoia as práticas e processos ITIL, ajudando na descoberta de incidentes decorrentes de configurações errôneas ou não autorizadas em ativos de redes. O desenvolvimento da solução é baseado em um modelo que faz uso de webservices RESTfull, bem como do protocolo SNMP.

Palavras-chave: Incidentes de Redes, Gerencia de Redes, ITIL, Gerência de configuração.

Automated Incident Detection System Networks due from Changes in the Configurations from Switch

Author: Silas Tiago de Medeiros

Advisor: Prof.º Dr. Marcos César Madruga Alves Pinheiro

ABSTRACT

Detection of network incidents is not a simple task, mostly in medium and large networks, So that the role of problem detection and resolution is handled by specialized. Ideally, the detection and treatment of incidents should occur in the shortest time possible, as defined by SLA. Among the several factors that collaborate to meet the times defined in a SLA, as the total volume of calls in service, and the existence of well-defined procedures, one of the most notable is the assurance that the actual configuration of the assets is as specified in the documentation maintained by the TIC team. This guarantee is required for all types of assets, and especially for switches, because they do not usually change frequently, are often managed by different TIC people/teams. The proposal of this work is the creation of a system of monitoring configurations, to support ITIL practices and processes, assisting in the discovery of incidents resulting from erroneous or unauthorized configurations in network assets. The solution development is based on a template that makes use of RESTfull webservices, as well as the SNMP protocol.

Keywords: Network Incidents, Network Management, ITIL, Configuration Management

Lista de ilustrações

Figura 1 – Ciclo de vida de um serviço segundo a ITIL	16
Figura 2 – Componentes da Arquitetura de Gerência de Redes	22
Figura 3 – Figura representando MIB II	23
Figura 4 – Figura representando MIB privada	24
Figura 5 – Representação do recurso	29
Figura 6 – Classe de recurso município	30
Figura 7 – Arquitetura Concreta dos componentes do sistema	31
Figura 8 – Diagrama de Sequência que representa a coleta das configurações . . .	34
Figura 9 – Diagrama de Sequência que representa o fluxo de aprovação das configurações	35
Figura 10 – Cenário de Coleta de Informação	36
Figura 11 – Informações coletadas da interface	39
Figura 12 – Vlans coletadas das interfaces	39
Figura 13 – Arquitetura do Banco de Dados do sistema	40
Figura 14 – Tela de Cadastro de Racks	42
Figura 15 – Tela de Cadastro de Ativos	44
Figura 16 – Tela de Aprova Configuração Pendente	44
Figura 17 – Tela de Aprova Configuração Pendente	45
Figura 18 – Tela de Listagem de Todas as vlans coletadas dos ativos	45
Figura 19 – Tela de Listagem de Todos os Switches que possuem vlan 1	46
Figura 20 – Metodo verificaInformacao	46
Figura 21 – Diagrama de Caso de Uso que representa as principais funções do sistema.	52

Lista de tabelas

Tabela 1 – Tabela de metodos do municipio	30
Tabela 2 – Tabela de objetos da MIB II utilizados no sistema	37
Tabela 3 – Tabela de detalhamento da resposta ifOperStatus	38
Tabela 4 – Tabela de objetos da MIB privada utilizados no sistema	38
Tabela 5 – Tabela de detalhamento da resposta do tipo porta	38
Tabela 6 – Tabela de metodos do switch	61
Tabela 7 – Tabela de metodos da vlan	62

Lista de abreviaturas e siglas

ANS	Acordo de Nível de Serviço
ASN.1	Abstract Syntax Notation One
AT	Address Translation
BDGC	Banco de dados do gerenciamento de configuração
CLI	Command Line Interface
CMIP	Common Management Information Protocol
CORBA	Common Object Request Broker Architecture
DIMAp	Departamento de Informática e Matemática Aplicada
EGP	Exterior Gateway Protocol
IC	Item de Configuração
ICMP	Internet Control Message Protocol
IP	Internet Protocol
ITIL	Information Technology Infrastructure Library
JAX-RS	API for RESTfulWeb Services
JAX-WS	Java API for XML-Based Services
JSF	Java Server Faces
MIB	Management Information Base
OID	Object Identifier
PDU	Protocol Data Unit
RDM	Requisição de Mudança
REST	Representational State Transfer
RFC	Request For Comment
RPCs	Remote Procedure Calls
SMI	Structure of Management Information

SNMP	Simple Network Management Protocol
SOAP	Simple Object Access Protocol
TCP	Transmission Control Protocol
TIC	Tecnologia de Informação e Comunicação
UFRN	Universidade Federal do Rio Grande do Norte
URIs	Uniform Resource Identifiers
WSDL	Web Services Description Language
W3C	World Wide Web Consortium

Sumário

1	INTRODUÇÃO	15
1.1	Motivação	16
1.2	Objetivos	17
1.2.1	Objetivo Geral	17
1.2.2	Objetivo Especifico	17
1.3	Metodologia	17
1.4	Organização do trabalho	18
2	EMBASAMENTO TEÓRICO	19
2.1	Gerenciamento de Redes	19
2.1.1	SNMP (Simple Network Management Protocol)	21
2.1.1.1	Arquitetura de Gerência de Redes	22
2.1.1.2	MIB (Management Information Base)	22
2.2	ITIL (Information Technology Infrastructure Library)	24
2.2.1	Gerenciamento de Configuração	26
2.3	Web Service	27
2.3.1	REST (Representational State Transfer)	28
3	SOLUÇÃO DE MONITORAMENTO	31
3.1	Arquitetura	31
3.1.1	Servidor	32
3.1.1.1	Coleta	32
3.1.1.2	Análise	32
3.1.1.3	Alerta	32
3.1.2	Cliente	33
3.1.3	Switch	33
3.1.4	Base de Dados das Configurações dos Ativos	33
3.1.5	MIB	33
3.2	Diagramas de Sequência	33
4	IMPLEMENTAÇÃO	36
4.1	Detalhamento dos componentes	36
4.1.1	Servidor	37
4.1.2	Cliente	37
4.1.3	MIB	37
4.1.4	Base de Dados das Configurações dos Ativos	38

4.2	Ambiente de Validação da Implementação da Arquitetura . . .	39
4.3	Diagrama do Banco de Dados	40
4.4	Interface do sistema	42
4.4.1	Local	42
4.4.2	Ativo	42
4.4.2.1	Cadastro de Ativo	43
4.4.2.2	Analisando as Configurações Pendentes	43
4.4.2.3	Listagem de Vlans	44
4.5	<i>Web Service</i> de verificação de alterações nos switches	45
5	CONSIDERAÇÕES FINAIS	48
5.1	Trabalhos Futuros	48
5.2	Limitações	49
	REFERÊNCIAS	50
	APÊNDICES	51
	APÊNDICE A – DIAGRAMAS DE CASO DE USO	52
	APÊNDICE B – PRINCIPAIS <i>WEB SERVICES</i>	60

1 Introdução

Aos longo dos anos, os ambientes de Tecnologia de Informação e Comunicação (TIC) das instituições vem aumentando bastante de tamanho e complexidade. Esses ambientes são compostos por tecnologias de comunicação de dados, processamento e armazenamento, e oferecem um conjunto tão grande e diversificado de serviços, que eles já se incorporaram aos fluxos de execução dos procesos das organizações.

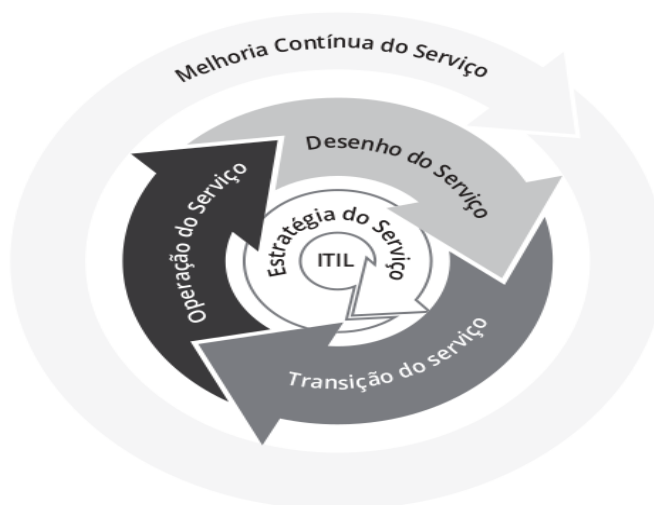
Como consequência dessa forte integração, a maior parte das instituições atualmente possui uma grande dependência da TIC, de modo que para que a instituição consiga prestar o serviço que é realmente sua atividade fim, é necessário que os recursos de TIC estejam funcionando conforme o esperado. Naturalmente, a medida que as tecnologias se tornam mais complexas e as redes precisam conectar cada vez mais pessoas e dispositivos dentro da instituição, bem como com entidades externas, tornando-se maiores, a ocorrência de problemas se torna mais frequente e sua solução exige maior esforço.

Como os problemas precisam ser solucionados em até um certo limite de tempo, para que não comprometam as atividades de uma forma muito impactante, normalmente se define esse limite máximo para cada possível tipo de incidente previsto, e a equipe de TIC se compromete respeitá-lo. Isso é o que se chama um Acordo de Nível de Serviço (ANS).

Evitar ou reduzir a ocorrência de incidentes, e agir rapidamente para solucionar os problemas dentro do tempo estipulado no ANS, não é tarefa simples, e requer uma definição muito clara dos processos TIC, bem como uma forte disciplina para executá-los conforme planejado. A resposta encontrada pelas instituições para essa questão, tem sido uma forte utilização da Information Technology Infrastructure Library (ITIL), que se constitui de melhores práticas para o gerenciamento de serviços de TIC. A ITIL foi criada no final da década de 1980 pelo governo britânico para permitir uma melhor avaliação dos prestadores de serviço, uma vez que a terceirização estava sendo fortemente utilizada. Na década de 1990 a ITIL passou a ser adotada por diversas instituições europeias e atualmente é utilizada em todo o mundo, tanto em instituições públicas como privadas.

A ITIL define o ciclo de vida de um serviço como sendo formado pelos cinco elementos mostrados na Figura 1. Enquanto a *Estratégia do Serviço* se preocupa em garantir que os serviços oferecidos sejam os adequados para ajudar a organização a atingir seus objetivos, e o *Desenho do Serviço* com o projeto do serviço em si, as fases de *Transição do Serviço* e *Operação do Serviço* estão relacionadas a implantação e utilização do serviço. Finalmente, a *Melhoria Contínua do Serviço* se refere a constante avaliação e aprimoramento do serviço.

Figura 1 – Ciclo de vida de um serviço segundo a ITIL



Fonte: (FILHO, 2012)

1.1 Motivação

Um dos pilares de toda a gestão de serviços na ITIL é o processo de *Gerenciamento de Configuração*. Apesar desse processo fazer parte da fase de *transição do serviço* do ciclo de vida de um processo, ele é utilizado por diversos outros processos ITIL, como, por exemplo, no processo de *Gestão de Incidentes* da fase de *Operação do Serviço*. Isso é natural, pois o *Gerenciamento de Configuração* fornece informações sobre todos os ativos, bem como dos relacionamentos entre eles.

Entretanto, apesar da importância dessas informações, em muitas organizações elas são mantidas de forma manual, ou seja, alimentadas estaticamente pelas pessoas que gerenciam os ativos, tipicamente registrando-as em Wikis ou algo similar. Essa abordagem é propensa a erros, pois com o tamanho, diversidade e dinamismo dos ambientes de TIC, é muito comum que as informações salvas nessas Wikis fiquem desatualizadas e não reflitam de fato a configuração real dos ativos. Com isso, não só o tempo para atendimento aos incidentes e solicitações diversas será comprometido, podendo inviabilizar o cumprimento dos ANS, como erros podem ser acarretados por assumir configurações errôneas como válidas.

Além disso, embora os processos da fase de *Transição do Serviço* cuidem das alterações realizadas em serviços já existentes, como é o caso do processo *Gerenciamento de Mudança*, algumas situações problemáticas podem surgir na prática. Por exemplo, eventualmente alguém pode não seguir o o fluxo do processo conforme definido e realizar uma alteração por conta própria em algum ativo. Além disso, devido a complexidade da interface de configuração de alguns ativos, em especial quando se utiliza a interface de linha de comando (CLI), podem ocorrer erros ao informar os parâmetros nos comandos. Problemas de segurança também podem acontecer, como alguém obter uma senha de

acesso indevidamente e realizar alterações nas configurações.

As duas considerações acima, ou seja, a necessidade de manter uma base de configuração atualizada e de detectar alterações indesejadas nas configurações, se aplicam a qualquer tipo de ativo, mas em grandes redes são especialmente importantes para os switches. Em redes de grandes organizações formadas por centenas de switches é muito difícil manter as informações completamente atualizadas, a menos que exista uma forte cultura de utilização da ITIL, principalmente porque o gerenciamento desses equipamentos acaba, muitas vezes, sendo distribuído entre diferentes equipes de TIC.

A proposta deste trabalho é a confecção de um sistema de monitoramento de configurações na camada de enlace em switches, que ajudará na descoberta de incidentes de rede provenientes das más configurações ou configuração indevida.

1.2 Objetivos

Esta seção apresenta o objetivo geral e os objetivos específicos desse trabalho.

1.2.1 Objetivo Geral

O objetivo deste trabalho é a criação de um sistema de detecção de incidentes de redes gerados por alterações não autorizadas nas configurações de switches garantindo também uma visão atualizada da configuração dos mesmos.

1.2.2 Objetivo Especifico

Os objetivos específicos deste trabalho foram:

- Implementar um sistema de captura de informações de configuração dos switches.
- Criar uma Base de Configuração dos switches.
- Integrar o sistema com outros sistemas de monitoramento.

1.3 Metodologia

A metodologia utilizada para o desenvolvimento deste trabalho seguiu as seguintes etapas:

1. Escolha do tema: Nesta fase, foi identificada uma necessidade dentro do contexto das redes de computadores e o escopo foi delimitado.
2. Revisão bibliográfica: Nesta fase, foi feita uma pesquisa sobre as diversas abordagens utilizadas para lidar com o tema escolhido.

3. Coleta de informações: Nesta fase, foram investigadas tecnologias/protocolos para obtenção das informações de configuração de um switch.
4. Definição da arquitetura do sistema: Nesta fase, se deu o planejamento da arquitetura do sistema, onde foram definidos seus componentes, a forma de comunicação entre eles, bem como as informações necessárias com respeito aos ativos.
5. Projeto e Desenvolvimento: Nesta fase, foi desenvolvido o sistema proposto a partir da arquitetura.
6. Escrita do trabalho: Nesta fase, que ocorreu em paralelo com as demais, foi realizada a escrita desse texto.

1.4 Organização do trabalho

Este capítulo apresentou uma introdução ao trabalho destacando os objetivos, uma motivação para sua realização e a metodologia utilizada.

O restante do trabalho está organizado em capítulos, dos quais o capítulo 2 apresenta o embasamento teórico, onde são explicados todos os conceitos utilizados no trabalho.

Em seguida, no capítulo 3, é apresentada a arquitetura do sistema e alguns diagramas de sequência.

Já no capítulo 4 será abordado o desenvolvimento do sistema, mostrando algumas telas do sistemas, a modelagem do banco de dados e a classe principal da solução.

Em seguida, nas considerações, é feita uma breve avaliação do trabalho realizado, incluindo as limitações encontradas durante o desenvolvimento, e finalizando com as possibilidades de trabalhos futuros.

No Apêndice são descritos os casos de uso, bem como o principal método do sistema que é a coleta de informação de um switch.

2 Embasamento Teórico

Este capítulo tem como objetivo apresentar a fundamentação teórica referente a tópicos utilizados para o desenvolvimento deste projeto.

2.1 Gerenciamento de Redes

Gerenciamento de redes segundo (KUROSE et al., 2010) "inclui o oferecimento, a integração, e a coordenação de elementos de hardware, software e humanos, para monitorar, testar, consultar, configurar, analisar, avaliar e controlar os recursos da rede, e de elementos, para satisfazer às exigências operacionais, de desempenho e de qualidade de serviço em tempo real a um custo razoável."

As informações transmitidas nas redes de computadores tem se tornado cada vez mais importantes. Seu monitoramento é essencial, visto que na ausência deste mecanismo, a detecção de problemas não se dá de modo pró-ativo, e alguns problemas, tais como, congestionamento da rede, recursos mal utilizados ou sobrecarregados, e problemas com segurança, dificilmente são detectados.

A gerência de rede controla e monitora o uso dos recursos em uma rede. Tendo como tarefas fundamentais, coletar informações da rede, tratar as informações coletadas, e encaminhar as soluções dos problemas. Para desenvolver tais tarefas existem uma série de atividades que ajudam em sua execução, tais como: Verificar Acordo de Nível de Serviço(ANS), conhecer e controlar alterações nos equipamentos, estabelecer critérios para disparos de alarmes entre outros. (SANTOS, 2015)

Com o desenvolvimento das arquiteturas de redes tanto o modelo OSI quanto TCP/IP, o gerenciamento de redes foi dividido em 5 tipos, de acordo com sua função. Essa divisão ficou conhecida pela sigla FCAPS, formada a partir das iniciais de cada área de gerenciamento. (SANTOS, 2015)

- *Fail*(Gerência de Falhas)
- *Configuration*(Gerência de Configuração)
- *Accounting*(Gerência de Contabilização)
- *Performance*(Gerência de Desempenho)
- *Security*(Gerência de Segurança)

Gerenciamento de Falhas: Tem como objetivo Detectar, isolar, e corrigir as falhas numa rede. Para isso o gerenciamento de falhas inclui as seguintes etapas:

- Detecção da falha.
- Isolamento da falha.
- Conserto da falha.
- Teste do conserto.
- Registro da falha e da solução.

Gerenciamento de Configuração: Tem como objetivo de conceder ao administrador conhecimento dos dispositivos que fazem parte da rede, e suas respectivas configurações de hardware e software. Para isso conta com alguns itens que fazem parte desse gerenciamento tais como:

- Inclusão de novos componentes na rede.
- Alteração de configurações nos componentes.
- Atualização de dispositivo.
- Outros

Gerenciamento de Contabilização: Tem como objetivo de estimar qual o custo do uso dos recursos da rede, e dependendo do uso pode-se contingenciar o uso do recurso. Para isso conta com algumas funções:

- Controlar o recurso.
- Manter Limites de consumo.
- Outros.

Gerenciamento de Desempenho: Tem como objetivo de mensurar e controlar o desempenho da rede, com isso garantir o tráfego mínimo na rede. Para isso conta com algumas funções:

- Avaliar as comunicações na rede.
- Monitorar as operações da rede diariamente.
- Localizar pontos críticos.
- Outros.

Gerenciamento de Segurança: Tem como objetivo controlar o uso da rede para que esta seja usada de modo apropriado, restringindo o acesso para os usuários que não autorizados. Para isso conta com algumas funções:

- Proteção das informações.
- Controle de acesso ao sistema.
- Manter registros de eventos relativos à segurança.
- Outros.

Com o crescimento das redes de computadores e a necessidade cada vez mais por gerenciamento, houve a necessidade de criação de padrões para ferramentas. Com isso o modelo OSI e o TCP/IP criaram seus próprios padrões:

- CMIP, protocolo derivado da arquitetura OSI, por sua complexidade derivada do modelo e pela demora em sua padronização não ficou popular.
- SNMP, protocolo derivado da arquitetura TCP/IP, por sua simplicidade foi adotado como padrão para redes TCP/IP, está em sua terceira versão e é o protocolo mais utilizado no mundo de gerência de redes.

2.1.1 SNMP (Simple Network Management Protocol)

É o protocolo mais utilizado na Internet para gerenciar uma rede, e mesmo apesar de suas limitações em versões iniciais, conseguiu superar o antigo CMIP e se tornar um padrão de fato.

As informações dos objetos gerenciados são armazenados na MIB, que é uma base de dados contendo todas as informações necessárias para o gerenciamento do objeto. A SMI, utiliza a ASN.1 para descrever as informações que são armazenados na MIB.

Além disso, são definidas mensagens PDU, para serem trocadas durante uma comunicação entre o gerente e o agente. Estas mensagens estão apresentadas a seguir.

- `getRequest`: Mensagem enviada pelo gerente ao agente solicitando o valor de uma variável;
- `getNextRequest`: Mensagem enviada pelo gerente ao agente solicitando o valor da próxima instância de objeto MIB;
- `getBulkRequest`: Mensagem enviada pelo gerente ao agente solicitando um bloco de dados;
- `getResponse`: Mensagem enviada pelo agente ao gerente, informando o valor de uma variável que lhe foi solicitado;
- `setRequest`: Mensagem enviada pelo gerente ao agente para solicitar que seja alterado o valor de uma variável.
- `trap`: Mensagem enviada pelo agente ao gerente, informando um evento ocorrido.

2.1.1.1 Arquitetura de Gerência de Redes

Uma abordagem da arquitetura dos sistemas de gerência de redes apresenta três componentes: (KUROSE et al., 2010).

- Agente.
- Estação de gerenciamento.
- Protocolo de Gerência.

A estação de gerenciamento é um sistema que coleta as informações e que um administrador gerencie as atividades de gerenciamento.

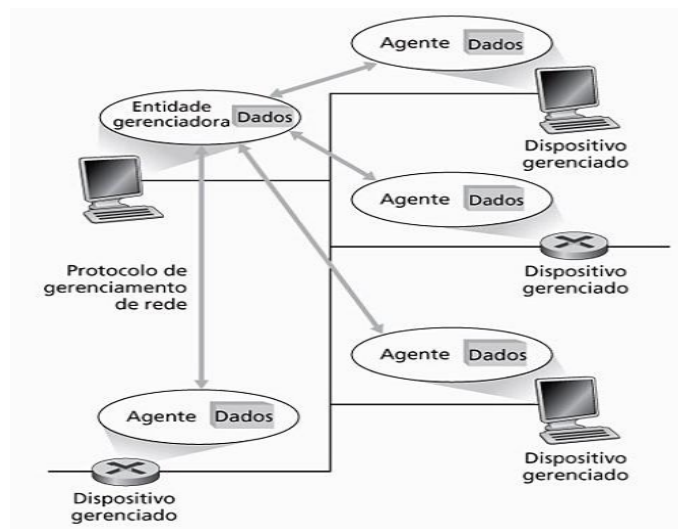
O agente de gerenciamento responde às solicitações da estação de gerenciamento.

Os recursos a serem gerenciados são representados como objetos, e a coleção de objetos são descritos como MIB.

O protocolo de gerência utilizado atualmente é o snmp, ele quem faz a comunicação entre o agente e a estação de gerenciamento.

A figura a seguir mostra como quais são os componentes da arquitetura de gerência de redes.

Figura 2 – Componentes da Arquitetura de Gerência de Redes



Fonte: (KUROSE et al., 2010)

2.1.1.2 MIB (Management Information Base)

MIB é uma coleção de objetos gerenciados, que contém todas as informações necessárias para a gerência da rede. Os objetos são dispostos em forma hierárquica e reconhecidos por um OID.

A primeira versão da MIB se deu com a RFC 1066. Sua evolução ocorreu quando a RFC 1213 propôs uma segunda MIB, a MIB II.

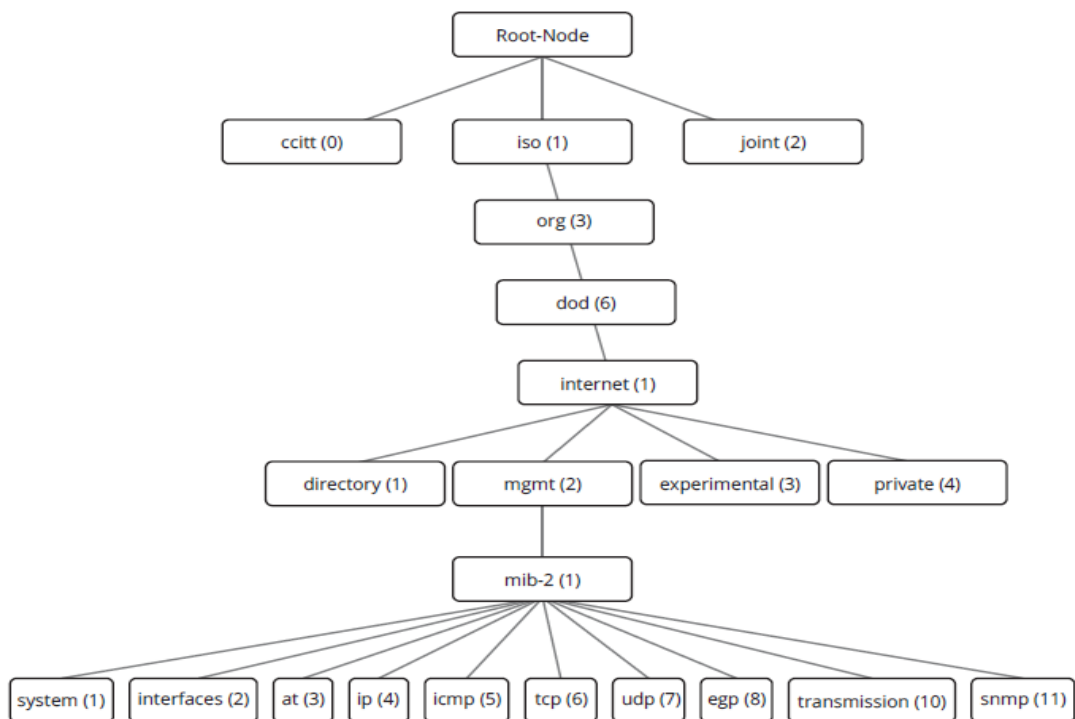
As MIBs existentes são: MIB I, MIB II, MIB experimental, MIB privada.

A MIB II, que é a evolução da MIB I, a MIB experimental é aquela que seus componentes estão e fase de teste, a MIB privada são aquelas que possuem informações específicas do fabricante do dispositivo e que este é responsável por especificar seus produtos.

Um OID é formado por uma sequência de inteiros baseada em nós da árvore, separada por pontos (.), O caminho da informação se inicia com o nó principal da árvore e vai descendo em profundidade até se chegar na informação que deseja. .

A figura abaixo mostra o posicionamento da mib II na árvore.

Figura 3 – Figura representando MIB II



Fonte: (SANTOS, 2015).

Logo abaixo esta descrito alguns grupos da Mib II:

- System - Define informações gerais do sistema.
- Interfaces - Define informações sobre as interfaces de rede.
- AT - Relação entre endereço físico e endereço de rede.
- IP - Define informações do protocolo IP.
- ICMP - Define estatísticas do protocolo ICMP.
- TCP - Define informações do protocolo TCP e tabela de conexões.

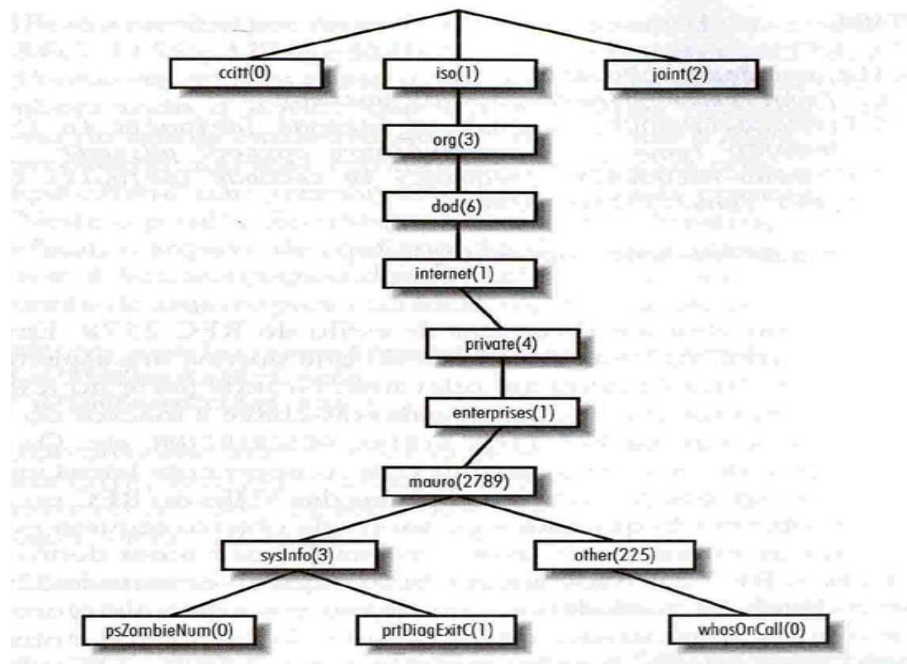
- UDP - Define informações do protocolo UDP.

A seguir, são listados alguns exemplos de objetos de alguns grupos.

- Grupo System.
 - * sysDescr - Define a descrição dos equipamentos.
 - * sysObjectID - Identifica o objeto fabricante.
- Grupo Interfaces
 - * ifIndex - Número da interface.
 - * ifDescr - Descrição da interface.
 - * ifType - Tipo da interface.

A figura abaixo mostra o posicionamento de uma MIB privada na árvore.

Figura 4 – Figura representando MIB privada



Fonte: MIB ().

2.2 ITIL (Information Technology Infrastructure Library)

Embora a maioria das instituições tenha iniciado a organização do gerenciamento de rede com base no modelo FCAPS (ISO/OSI), mesmo no uso da arquitetura de gerenciamento para a internet, um crescente número de instituições está começando a utilizar o modelo ITIL. Esse modelo funcional mais abrangente começou a receber atenções e tem sido adotado como base para a organização, não apenas do gerenciamento de rede, mas de toda a infra-estrutura de TI das organizações. (SANTOS, 2015)

A ITIL busca promover a gestão com foco no cliente e na qualidade dos serviços de TI. A ITIL endereça estruturas de processos para a gestão de uma organização de TI apresentando um conjunto abrangente de processos e procedimentos gerenciais. (SANTOS, 2015)

Criada em 1980 pelo governo britânico para permitir uma melhor avaliação dos prestadores de serviço, uma vez que a terceirização estava sendo fortemente utilizada. Na década de 1990 a ITIL passou a ser adotada por diversas instituições europeias e atualmente é utilizada em todo o mundo, tanto em instituições públicas como privadas.

Na sua primeira versão foi constituído de 40 livros. Ela sofreu uma reformulação em sua segunda versão, sendo reduzidos os volumes de livros para 8. E chegando em sua terceira versão os 5 volumes realizam parte do ciclo de vida do serviço. Seus 5 volumes são:

- **Estratégia de Serviço:** Direciona de como projetar, desenvolver, implementar o gerenciamento de serviço. Principais processos:
 - Gerenciamento de demanda
 - Gerenciamento de Portifólio
 - Gerenciamento financeiro
- **Projeto de Serviço:** Direciona para o projeto, desenvolvimento e implementação do gerenciamento de serviço. Principais processos:
 - Gerenciamento de Nível de Serviço
 - Gerenciamento de Catálogo de Serviço
 - Gerenciamento de Disponibilidade
 - Gerenciamento de Fornecedores
 - Gerenciamento de Capacidade
 - Gerenciamento de Continuidade dos Serviços de TI
- **Transição de Serviço:** Mostra como desenvolver e melhorar as capacidades de transferir serviços novos ou alterados. Serviço entregues para a produção e com riscos controlados. Principais processos:
 - Gerenciamento de Mudanças.
 - Gerenciamento de Ativos de Serviços e Configurações.
 - Gerenciamento de Conhecimento.
- **Operação de Serviço:** Desempenha as atividades garantindo que as entregas dos serviços estão em níveis acordados para os usuários de negócio. Principais processos:
 - Gerenciamento de Eventos.

- Gerenciamento de Incidentes.
 - Execução de Requisição.
 - Gerenciamento de Acesso.
- **Melhoria Contínua de Serviço:** Garante a criação e manutenção de serviços, através do projeto, transição e Operação de Serviços. Principais processos:
 - Medição do Serviço.
 - Processo de Melhoria de 7 Passos.

Como visto na Figura 1, o modelo ITIL utiliza a estratégia de serviço como o núcleo do ciclo de vida do serviço o projeto, transição e operação de serviço como formação de ciclo de vida orbitando o núcleo, sendo este conjunto apoiado pela melhoria contínua do serviço.

2.2.1 Gerenciamento de Configuração

Com o armazenamento e a gerência das informações, este processo oferece um controle dos serviços e equipamentos de TI. Ele também previne as mudanças que são feitas sem autorização via gestão de mudança.

Item de configuração (IC) é um ativo, serviço, que está sob o controle deste processo.

As atividades do processo de gerência de configuração são:

- **Planejamento.** Definir objetivos, escopo e políticas.
- **Identificação.** Esta atividade envolve a coleta das informações do IC, essa coleta pode ser feita tanto manual como automaticamente com o auxílio de ferramentas.
- **Controle.** Atividade que garante que as mudanças no BDGC só são realizadas com autorização e adota procedimentos para que as mudanças sejam documentadas.
- **Acompanhamento de Status.** Atividade que registra o estado dos ICs tornando-os rastreáveis.
- **Verificação e auditoria.** Verifica se a configuração do ativo ou serviço está conforme registrado no BDGC.

Os objetivos deste gerenciamento são:

- Fornecer gerenciamento de TI com maior controle dos itens de configuração.
- Manter os dados atualizados dos itens de configuração.

- Assegurar mudanças controladas pelos procedimentos do controle de mudanças.
- Facilitar a adição ou remoção ICs.
- Identificar automaticamente outros itens de configuração quando qualquer IC é objeto de incidente.
- Outros.

2.3 Web Service

Com o crescimento da internet, como meio de comunicação e o avanço das linguagens de programação, a integração de sistemas se torna algo que inevitável.

Com este pensamento programadores e desenvolvedores, sempre procuraram maneiras de construir softwares e sistemas de forma distribuída, que seja reutilizável, escalável e livre para qualquer plataforma.

Seguindo esta razão surge os Web Services, sistemas interoperáveis que podem ser implementados em um servidor qualquer e que podem ser acessados através da internet.

Existe vários tipos de webservices e cada um implementa sua lógica de aplicação, estas implementações são definidas por padrões tais como: CORBA, SOAP, REST, entre outros.

Por serem interoperáveis, ou seja não dependem de plataformas específicas para ser implementados, os web services são bem flexíveis e extensivos podendo ser seus serviços consumidos por qualquer aplicação, não importando qual a linguagem de programação utilizada. (CERAMI, 2002)

Para que essa tecnologia seja interoperável ela tem que possuir algumas características: (CHAPPELL; JEWELL, 2002)

- Padronização para representações de dados
- Baixo Acoplamento
- Capacidade de ser síncrona ou assíncrona
- Suportar chamadas de procedimento remoto RPCs.
- Suportar troca de documento

Cada plataforma oferece os recursos necessários para que os desenvolvedores possam disponibilizar ou acessar web services. A organização W3C define alguns padrões para definir o funcionamento de um web service. Em geral, as plataformas de maior uso comercial implementam a arquitetura definida pelos padrões da W3C.

Na plataforma Java, que é a linguagem a qual vamos utilizar, existem especificações que definem a implementação java dos padrões estabelecidos pelo W3C. Estas especificações são:

- A especificação JAX-WS: Especificação para criação de webservice SOAP.
- A especificação JAX-RS: Especificação para criação de webservice REST.

2.3.1 REST (Representational State Transfer)

Estilo arquitetural proposto por (FIELDING, 2000), em que buscou as melhores práticas nos estilos de arquiteturas existentes para criar um novo que agregasse em apenas um estilo.

- Cliente/Servidor. Este estilo define a separação das responsabilidades entre cliente e servidor, principalmente separar a interface do usuário do armazenamento de dados, facilitando assim as operações do servidor. Neste estilo o servidor contém os serviços são invocados por aplicações clientes independente de plataforma. (FIELDING, 2000).
- Stateless. Este estilo deriva do cliente/servidor e com a restrições de que o servidor não aceita nenhum estado de sessão, a cada nova requisição, o cliente deve enviar todas as informações necessárias junto à requisição para que a mesma possa ser processada pelo servidor. (FIELDING, 2000).
- Cache. Este estilo permite fazer uma cópia da informação de uma requisição para que não seja preciso fazer uma próxima consulta num segundo request. (FIELDING, 2000)
- Interface Uniforme. Estilo que define alguns princípios fundamentais para que seja feito um contrato de comunicação entre cliente e servidor. Esses princípios são:
 - Identificação do Recurso: Todo recurso deve possuir uma URI única e coesa para facilitar o seu acesso.
Ex: HTTP/1.1 GET http://localhost:8080/snmp/rest/consulta/vlans/switch/1, neste exemplo com o metodo get se traz todas as vlans do switch de id 1.
 - Representação do Recurso: Formato de retorno do recurso, existe vários formatos na web tais como: JSON, XML, TXT, HTML entre outros, hoje em dia muitas aplicações utilizam o JSON pois ser bem mais leve que os outros formatos e é bem intuitivo. Na figura 6 mostra a representação do recurso retornado pelo item anterior.
 - Resposta auto-descritiva: Meta informações importantes como código http, nome do host acessado entre outros.

- Hipermedia: Retorno de de links ou links do recurso no corpo da resposta ajudando clientes se guiarem a outros recursos.

Figura 5 – Representação do recurso

```
[
  {
    "numeroVlan": 1
  },
  {
    "numeroVlan": 202
  },
  {
    "numeroVlan": 255
  }
],
{
  "numeroVlan": 2204
}
]
```

Fonte: Próprio autor

- Sistema de Camadas. Este estilo permite uma arquitetura ser composta de camadas hierárquicas, cada camada fornecendo serviço para a camada superior e usando as camadas abaixo dele. (FIELDING, 2000)
- Code-on-Demand. Este estilo permite que código do servidor seja migrado para o cliente. (FIELDING, 2000)

Todas as informações são disponíveis em forma de recurso. Um cadastro de produto, uma foto, um documento entre outros, são exemplos de recursos (FIELDING, 2000).

O protocolo HTTP possui uma pequena e fixada série de operações e de interface simplificada que asseguram a comunicação entre clientes e serviços. Essa interface é definida por quatro métodos principais (GET, PUT, DELETE, POST). (BURKE et al., 2013)

A tabela 1 mostra os metodos do sistema para as operações da entidade municipio.

Os métodos HTTP podem ser mapeados para métodos Java de uma classe de recurso. As anotações @GET, @PUT, @POST, @DELETE e @HEAD são utilizadas para realizar esse mapeamento.

Na Figura 6 mostra a URI que é definida parcialmente pela anotação @Path, e também pelo Media type representado pela anotação @produces e também o metodo que estou utilizando neste caso o GET.

Tabela 1 – Tabela de metodos do municipio

Recurso	Método HTTP	Semântica	Descrição do Parametro
/rest/municipio/	GET	Lista todos os municipios	
/rest/municipio/{id}	GET	Lista o municipio por seu id	id correspondente ao municipio
/rest/municipio/	POST	Adiciona um municipio	
/rest/municipio/{id}	PUT	Atualiza um municipio	id correspondente ao municipio
/rest/municipio/{id}	DELETE	Remove um municipio	id correspondente ao municipio

Figura 6 – Classe de recurso municipio

```

@Path("/municipio")
public class MunicipioRecurso {
    private IMunicipio daomunicipio = new DaoMunicipio();

    @GET
    @RolesAllowed("ADMIN")
    @Produces({ MediaType.APPLICATION_JSON })
    public List<Municipio> verMunicipios() {
        return daomunicipio.listarMunicipios();
    }
}

```

Fonte: Próprio autor

3 Solução de Monitoramento

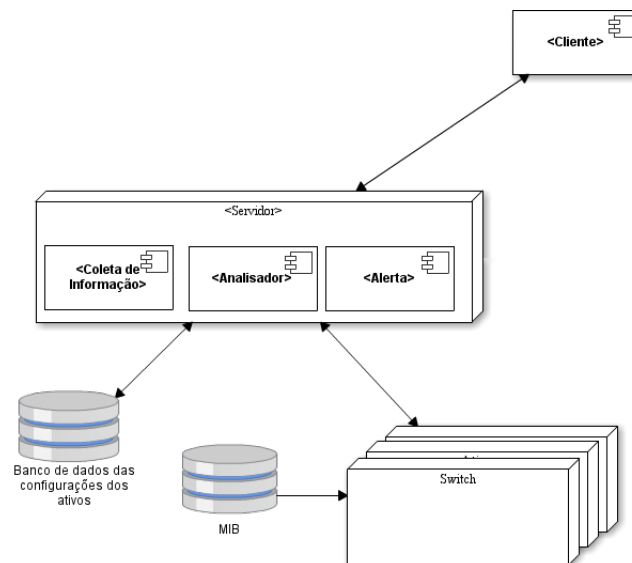
A proposta deste trabalho consiste na criação de um sistema de monitoramento de configurações de switches que apoie os processos ITIL. O principal requisito da solução consiste na capacidade de detectar em poucos minutos qualquer alteração na configuração dos switches, emitindo um alerta para a equipe responsável pelo referido ativo. Desse modo, a equipe poderá analisar se o evento consiste da execução de uma ação já planejada ou é uma operação não autorizada.

Um benefício direto dessa solução é a disponibilização para a equipe de TIC de informações referentes as configurações dos switches realmente atualizadas, sendo possível monitorar switches de diversos fabricantes diferentes.

Conforme visto na seção 2.2.1, *Gerenciamento de Configuração*, para que se tenha uma boa gerência dos ativos, a ITIL recomenda o uso do BDGC, que consiste de um repositório de dados onde ficam armazenados os itens de configuração(IC), e deve ser mantido sempre atualizado para facilitar a identificação dos possíveis incidentes na rede.

3.1 Arquitetura

A Figura 7 apresenta a visão da arquitetura da solução proposta por este trabalho, mostrando seus componentes, bem como as ligações entre os mesmos. Na referida figura pode-se identificar os seguintes componentes: Servidor, Cliente, Switch, MIB, Base de Dados das Configurações dos Ativos.



Fonte: Próprio autor

3.1.1 Servidor

Sistema que tem as funções de coletar, analisar e alertar o usuário sobre as alterações de configurações dos ativos. De fato, cada uma dessas atividades é realizada por um componente separado, mas que se comunicam. Desse modo, o servidor registra as configurações atuais no BDGC e emite um alerta quando alterações são detectadas. O processo de monitoramento é realizado da seguinte maneira, primeiro o servidor faz a coleta das configurações do switch, e após essa operação as informações coletadas são analisadas e persistidas no BDGC. O sistema faz esse monitoramento periodicamente, e embora o intervalo padrão tenha sido definido como cinco minutos, esse tempo pode ser configurado pelo usuário.

A seguir uma breve explicação dos módulos do servidor. No Capítulo 4 será abordado como suas funções são realizadas.

3.1.1.1 Coleta

Módulo que captura as informações do switch tais como: velocidade da porta, se a porta está ligada ou não, tipo da porta, e as vlans para as quais a porta está atribuída. Outras informações também poderiam ser coletadas, entretanto, para esse trabalho foram utilizadas apenas as informações citadas anteriormente.

Cada informação monitorada, é obtida de uma MIB, onde existe um OID que representa a referida informação. Algumas informações foram obtidas de MIBs privadas, gerando uma complexidade adicional a essa tarefa, pois, nesses casos, devido a falta de padronização, é necessário descobrir qual o OID e o tipo de dados do retornado.

3.1.1.2 Analise

Módulo que analisa as informações do switch durante a coleta da informação. O servidor compara a informação do ativo com a configuração atualizada da base de configuração, e caso essa configuração seja igual, o sistema atualiza o timestamp da consulta. Caso contrário, o sistema persistirá a configuração no banco com o status de configuração pendente, e chamará o módulo de alerta para informar o usuário sobre o ocorrido.

3.1.1.3 Alerta

Módulo que gera alertas para os usuários. Este módulo é configurado pelo usuário para que o sistema gere os tipos de alertas desejados quando alterações nas configurações forem detectadas.

3.1.2 Cliente

Um sistema que permite aos usuários analisarem as configurações dos ativos, e para as alterações detectadas, decidirem se as aceitam ou não. Ser rejeitada significa, por exemplo, que a equipe identificou algum erro na configuração ou que ela foi realizada por alguém não autorizado. Vale ressaltar que na versão atual, o sistema não consegue restaurar a configuração anterior do ativo para o equipamento, de modo que essa operação precisa ser feita manualmente.

3.1.3 Switch

Equipamento que terá suas configurações monitoradas. Embora atualmente o sistema realize apenas o monitoramento de switches, o modelo proposto pode ser utilizado para qualquer tipo ativo de rede que suporte o protocolo SNMP, como roteadores e pontos de acesso, por exemplo.

3.1.4 Base de Dados das Configurações dos Ativos

Consiste num repositório utilizado para o armazenamento das configurações dos equipamentos. Essa base, armazena tanto as informações atualizada dos ativos, quanto todo o histórico de configurações, indicando, para cada uma delas, se ela foi aprovada ou rejeitada pela equipe de TIC.

O Capítulo 4 apresentará maiores detalhes sobre essa base de configurações.

3.1.5 MIB

Como explicado na Seção 2.1.1.2, a MIB é um Base de informação mantida pelos ativos que contém suas configurações. Entre as diversas informações disponíveis sobre os switches na MIB e que a solução proposta utiliza encontram-se: vlans as quais as portas estão atribuídas, status da porta (ativa ou não), auto-negociação, entre outros.

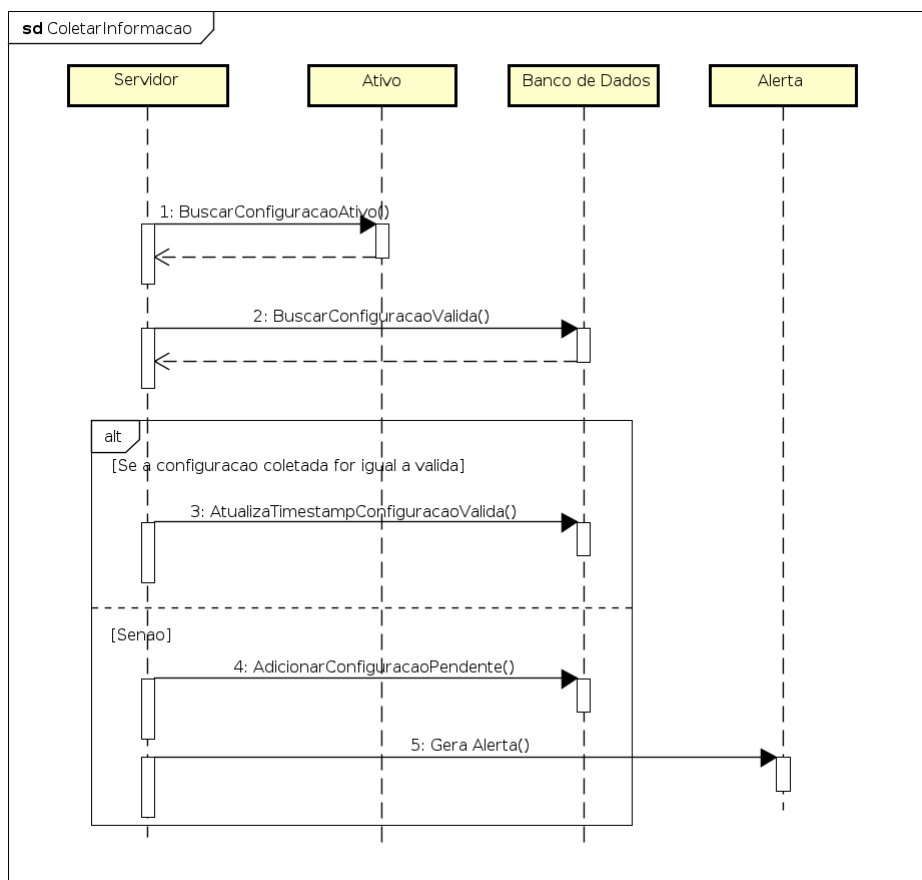
No Capítulo 4 será abordado o detalhamento das MIBs utilizadas no trabalho.

3.2 Diagramas de Sequência

O diagrama de sequência da Figura 8 representa como ocorre a captura da informação de configuração de um ativo. Vale ressaltar que o servidor executa as verificações periodicamente, e as faz para cada ativo sendo monitorado. Esta periodicidade é configurada pelo próprio usuário através do sistema, deixando as configurações gravadas no BDGC.

Inicialmente, o servidor obtém as informações de configuração do ativo, e concomitante busca a configuração válida que está persistida no BDGC. Após ter buscado as informações, o servidor faz uma comparação das informações, e caso a configuração

Figura 8 – Diagrama de Sequência que representa a coleta das configurações



Fonte: Próprio autor

coletada seja igual a válida o sistema atualiza a hora de verificação da configuração. Caso sejam diferentes, o sistema cria uma nova configuração pendente associada às informações do switch.

Pode-se observar também que quando é detectada certa alteração na configuração, o módulo de alerta é acionado para que a equipe de TIC tome ciência da situação. Ter um módulo separado para realizar essa função fornece uma grande flexibilidade ao sistema, pois diversos mecanismos de alerta podem ser implementados, como por exemplo, envio de emails, mensagens instantâneas de texto (SMS), Whatsapp ou Telegram, e integração com softwares de monitoramento de redes comumente utilizados, como é o caso do Nagios e Zabbix.

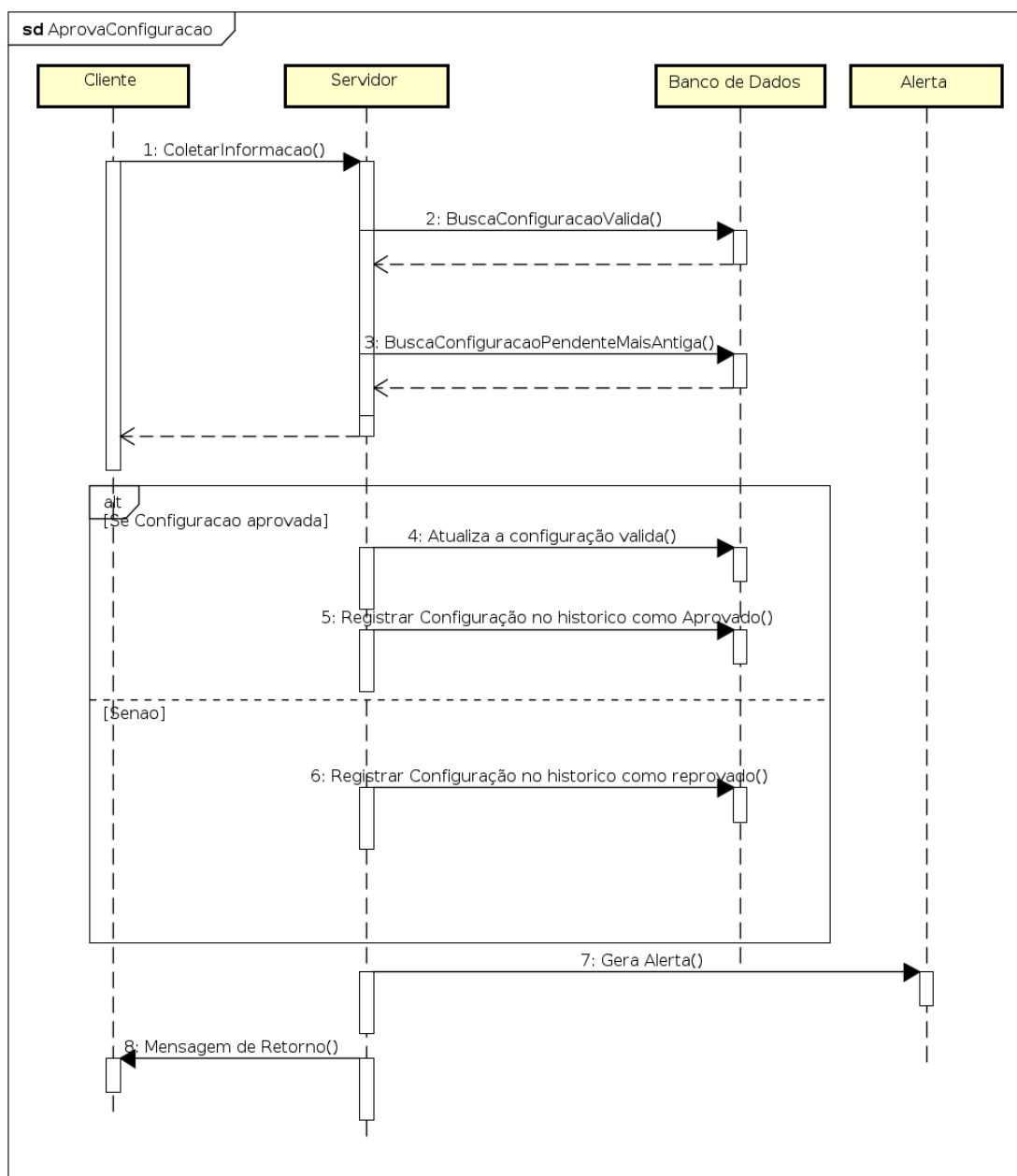
Pode-se analisar que esse procedimento pode, e de fato é, utilizado também para dar uma carga inicial no BDGC com as configurações dos ativos. Isso evita a necessidade de fazer a entrada manual dessas informações.

Antes de analisar o fluxo de aprovação das configurações, é importante observar que podem existir várias configurações pendentes. Isso acontece porque cada vez que o servidor executa o procedimento de checagem das configurações e existe uma divergência entre a configuração registrada no banco de dados como sendo a válida e configuração real

do ativo no momento, essa configuração real é registrada como pendente. A avaliação das configurações pendentes, ou seja, sua aprovação ou rejeição, deve ser feita da mais antiga para a mais nova.

O diagrama de sequência da figura 9 mostra o fluxo de aprovação de cada configuração do ativo. Inicialmente o sistema cliente faz uma requisição ao servidor para buscar a configuração válida e a pendente mais antiga. Caso a configuração pendente seja aprovada, ela passa a ser a configuração válida com status de aprovada e o campo de configuracao valida é atualizado para true, e o de reprovada para false. Caso a configuração seja reprovada apenas o campo reprovada é atualizado para true.

Figura 9 – Diagrama de Sequência que representa o fluxo de aprovação das configurações



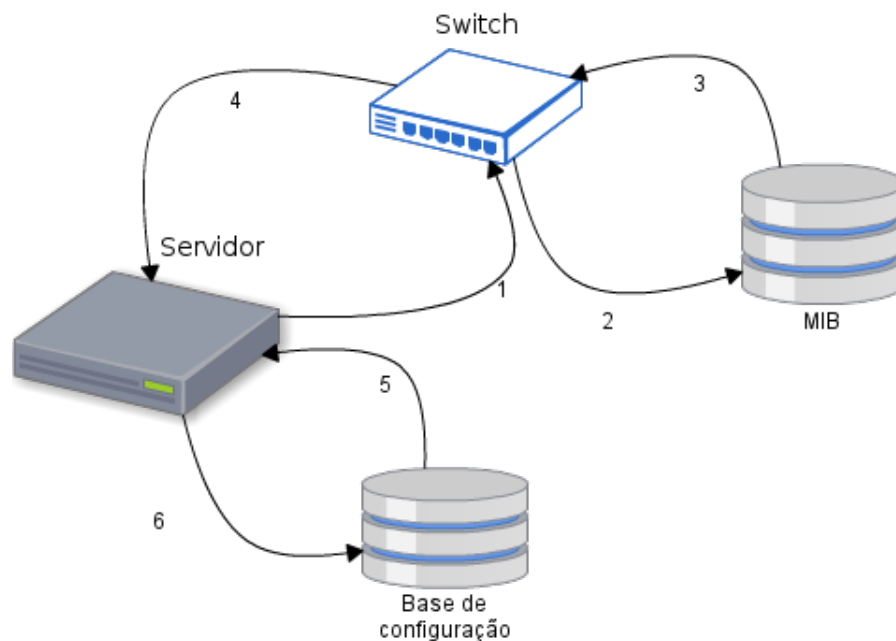
Fonte: Próprio autor

4 Implementação

Este capítulo apresenta detalhes sobre a implementação da solução proposta. Após uma breve explicação sobre quais foram as tecnologias e protocolos utilizados para realizar a comunicação entre os componentes do sistema, são mostrados detalhes sobre a modelagem do banco de dados. Na sequência, algumas telas do sistema são apresentadas para exemplificar as informações disponibilizadas. Finalmente, alguns detalhes sobre a implementação em Java são discutidos.

Na Figura 10 podemos ver como se dá o fluxo de comunicação entre os diversos elementos envolvidos no monitoramento da configuração de um ativo. Os passos 1 e 4 consistem, respectivamente, do envio pelo servidor que envia uma solicitação SNMP para obter a configuração do switch, e da resposta. A solicitação utiliza o método *bulkrequest*. Os passos 2 e 3 são realizados internamente ao switch. Finalmente, os passos 5 e 6 se referem, respectivamente, a solicitação e obtenção da configuração válida que está salva no banco de dados SQL.

Figura 10 – Cenário de Coleta de Informação



Fonte: Próprio autor

4.1 Detalhamento dos componentes

Conforme já mencionado no Capítulo 3, o sistema de monitoramento proposto é composto pelos seguintes componentes: Servidor, Cliente, MIB, Base de Dados das

Configurações dos Ativos. A seguir, serão detalhadas as tecnologias utilizadas em cada componente.

4.1.1 Servidor

Sistema que tem as funções de coletar, analisar e alertar o usuário sobre as configurações dos ativos. A coleta das configurações dos ativos é realizada através do protocolo SNMP com a biblioteca Java SNMP4J, e após essa operação as informações são persistidas no BDGC que é um banco de dados relacional PostgreSQL. Para suportar o monitoramento periódico dos ativos é utilizado o framework *Quartz Scheduler*. O servidor também faz uso do framework jersey para implementação de web services REST, para que os dados da base sejam consumidos via API REST.

4.1.2 Cliente

Este módulo pode ser implementado como um aplicativo mobile, um sistema Web, ou desktop, visto que a arquitetura do servidor permite o consumo dos dados via REST. Para este trabalho optou-se por implementar aplicação web em Java. Este sistema permite aos usuários analisarem as configurações, e para as alterações detectadas, decidirem se as aceitam ou não. Este módulo é composto de uma aplicação JSF, Primefaces e Bootstrap que são frameworks de componentes visuais. Também foi utilizada a biblioteca jersey para consumir os dados do servidor via REST.

4.1.3 MIB

Base de informação mantida pelos ativos que contém suas configurações. Entre as diversas informações disponíveis sobre os switches na MIB e que a solução proposta utiliza encontram-se: vlans as quais as portas estão atribuídas, status da porta (ativa ou não), auto-negociação, entre outros.

Na Tabela 2 são representados os grupos da MIB II que foram utilizados para a coleta da informação.

Tabela 2 – Tabela de objetos da MIB II utilizados no sistema

Grupo da MIB	Objetos	OID	Valor
Interfaces	ifDescr	1.3.6.1.2.1.2.2.1.2.1	GigabitEthernet1/0/1
Interfaces	ifSpeed	1.3.6.1.2.1.2.2.1.5.1	100000000
Interfaces	ifOperStatus	1.3.6.1.2.1.2.2.1.8.1	1

Na Tabela 3 são detalhados os tipos de respostas para o grupo ifOperStatus.

Tabela 3 – Tabela de detalhamento da resposta ifOperStatus

OID	Valor	
1.3.6.1.2.1.2.2.1.8	1	No-Shutdown
1.3.6.1.2.1.2.2.1.8	2	Shutdown

Algumas informações imprescindíveis só foram encontradas em MIBs privadas, como foi o caso do tipo da porta e do mapeamento de vlan-porta. A Tabela 4 mostra as informações coletadas do switch HPE 5120-24G EI, onde a primeira linha da tabela representa a informação do tipo da porta.

Na segunda linha da Tabela 4, o OID referente ao mapeamento vlan-porta é o 1.3.6.1.4.1.25506.8.35.2.1.1.1.19, que vem acrescido do numero da vlan. Como pode ser visto no exemplo da tabela é o numero 1. O valor do atributo é uma string contendo todas as portas, separadas por virgula.

Tabela 4 – Tabela de objetos da MIB privada utilizados no sistema

Descrição	OID	Valor
Tipo da porta	1.3.6.1.4.1.25506.8.35.1.1.1.5.1	2
Mapeamento Porta-vlan	1.3.6.1.4.1.25506.8.35.2.1.1.1.19.1	25,26,27

Na Tabela 5 são detalhadas as possíveis respostas que podem ser retornadas para uma consulta referente ao tipo da porta. A resposta à consulta vem como um numero inteiro, sendo que cada número representa um tipo de porta, conforme exposto na tabela.

Tabela 5 – Tabela de detalhamento da resposta do tipo porta

OID	Resposta	Correspondente
1.3.6.1.4.1.25506.8.35.1.1.1.5	1	Trunk
1.3.6.1.4.1.25506.8.35.1.1.1.5	2	Access
1.3.6.1.4.1.25506.8.35.1.1.1.5	3	Hybrid

4.1.4 Base de Dados das Configurações dos Ativos

Consiste do banco de dados postgresSQL, utilizado para o armazenamento das configurações dos equipamentos. Esta base armazena tanto as informações atualizadas dos ativos, quanto todo o histórico de configurações, indicando, para cada uma delas, se ela foi aprovada ou rejeitada pela equipe de TIC.

A Figura 11, representa as informações coletadas de cada interface do switch tais como: o nome da interface representado pelo padrao_interface, tipo da porta representado

pelo tipo_vlan, velocidade, se a porta está ligada ou não, e tipo do conector. Já a Figura 12, representa as vlans existentes em cada interface do referido ativo.

Figura 11 – Informações coletadas da interface

id_porta	padrao_interface	tipo_vlan	velocidade	ligada	tipo_conector
1	GigabitEthernet1/0/1	trunk	1000	No-Shutdown	RJ45
2	GigabitEthernet1/0/2	trunk	1000	No-Shutdown	RJ45
3	GigabitEthernet1/0/3	access	1000	No-Shutdown	RJ45
4	GigabitEthernet1/0/4	trunk	1000	No-Shutdown	RJ45
5	GigabitEthernet1/0/5	trunk	1000	No-Shutdown	RJ45
6	GigabitEthernet1/0/6	trunk	1000	No-Shutdown	RJ45
7	GigabitEthernet1/0/7	trunk	1000	No-Shutdown	RJ45
8	GigabitEthernet1/0/8	trunk	1000	No-Shutdown	RJ45
9	GigabitEthernet1/0/9	trunk	1000	No-Shutdown	RJ45
10	GigabitEthernet1/0/10	trunk	1000	No-Shutdown	RJ45
11	GigabitEthernet1/0/11	trunk	1000	No-Shutdown	RJ45
12	GigabitEthernet1/0/12	access	100	No-Shutdown	RJ45
13	GigabitEthernet1/0/13	access	1000	No-Shutdown	RJ45
14	GigabitEthernet1/0/14	access	1000	No-Shutdown	RJ45
15	GigabitEthernet1/0/15	access	1000	No-Shutdown	RJ45
16	GigabitEthernet1/0/16	access	1000	No-Shutdown	RJ45
17	GigabitEthernet1/0/17	hybrid	1000	No-Shutdown	RJ45
18	GigabitEthernet1/0/18	hybrid	1000	No-Shutdown	RJ45
19	GigabitEthernet1/0/19	hybrid	100	No-Shutdown	RJ45
20	GigabitEthernet1/0/20	hybrid	100	No-Shutdown	RJ45
21	GigabitEthernet1/0/21	hybrid	100	No-Shutdown	RJ45
22	GigabitEthernet1/0/22	hybrid	1000	Shutdown	RJ45
23	GigabitEthernet1/0/23	hybrid	1000	No-Shutdown	RJ45
24	GigabitEthernet1/0/24	hybrid	100	No-Shutdown	RJ45
25	Ten-GigabitEthernet1/0/25	access	10000	No-Shutdown	SFP
26	Ten-GigabitEthernet1/0/26	access	10000	No-Shutdown	SFP
27	Ten-GigabitEthernet1/0/27	access	10000	No-Shutdown	SFP
28	Ten-GigabitEthernet1/0/28	trunk	1000	No-Shutdown	SFP

Fonte: Próprio autor

Figura 12 – Vlans coletadas das interfaces

id_porta	numero_vlan
1	610
1	611
1	612
1	613
1	614
2	610
2	613
3	610
3	613
4	610
4	614
5	610
5	611
6	610
6	611
7	610
7	612
8	610
8	612
9	610
9	612
10	610
10	611
11	610
11	611

Fonte: Próprio autor

4.2 Ambiente de Validação da Implementação da Arquitetura

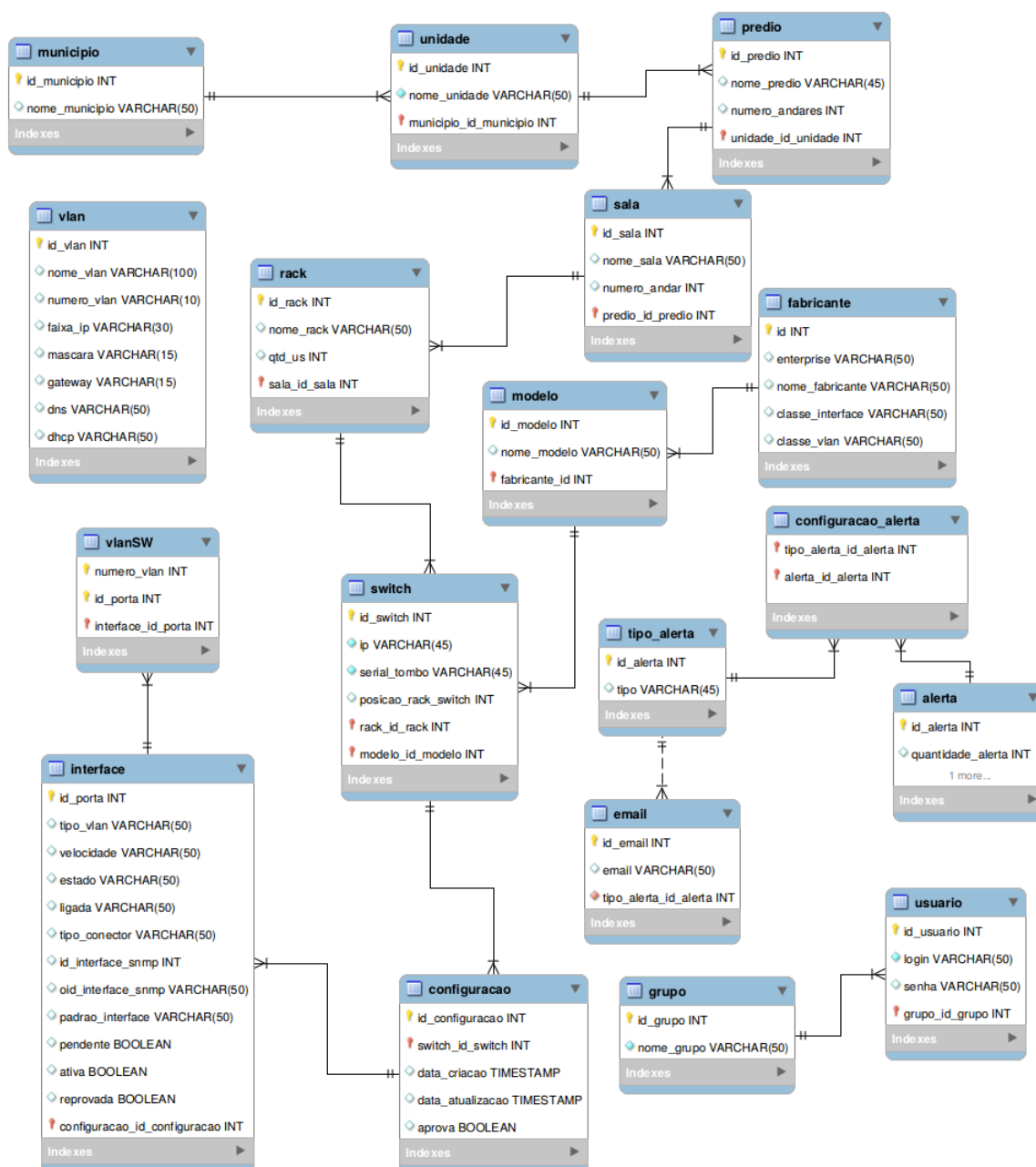
Para validar a arquitetura, uma infraestrutura foi configurada para dar suporte a implementação da solução proposta. O sistema desenvolvido foi executado em uma máquina

virtual com 4 Gigabytes de memória RAM, onde foi instalado o sistema operacional centOS 7, Java versão 1.7.0_111 e tomcat em sua versão 8.

4.3 Diagrama do Banco de Dados

A Figura 13 apresenta a modelagem do banco de dados utilizado na solução proposta. Nessa figura pode-se observar a existência das seguintes tabelas:

Figura 13 – Arquitetura do Banco de Dados do sistema



Fonte: Próprio autor

- municipio - Tabela onde são cadastrados os municípios.

- unidade - Tabela onde são cadastradas as unidades de cada município.
- predio - Tabela onde serão cadastrados os prédios de cada unidade.
- sala - Tabela onde são cadastradas as salas dos prédios onde contém os equipamentos a serem monitorados.
- rack - Tabela onde são cadastrados os racks das salas.
- switch - Tabela onde são cadastrados todos os switches da empresa.
- modelo - Tabela onde são cadastrados todos os modelos dos switches existentes na empresa.
- fabricante - Tabela onde são cadastrados todos os fabricantes dos switches existentes na empresa.
- configuração - Tabela onde são cadastrados todos os id das configurações dos switches.
- interface - Tabela onde são cadastradas as configurações validas referentes ao switch.
- vlansw - Tabela onde são cadastradas as vlans dos switches com as respectivas portas.
- vlan - Tabela onde são cadastradas manualmente as configurações das vlans que existem na organização.
- email - Tabela que conterá todos os emails utilizados para receberem os alertas do sistema.
- usuario - Tabela que contém os usuários que podem acessar o sistema.
- grupo - Tabela que contém os grupos que têm as permissão de acessar cada recurso.
- alerta - Tabela que contém a configuração do alerta, como: quantidade de alertas e horários.
- tipo_alerta - Tabela que contém os tipos dos alertas que podem ser utilizados, , como: Email, SMS, Telegram entre outros.
- configuracao_alerta - Tabela onde se mantém a associação da tabela alerta com a tipo_alerta para criar a configuração do alerta desejado pelo usuário.

4.4 Interface do sistema

A interface de utilização do sistema está dividida em três partes principais:

- Local. Menu onde o administrador irá cadastrar onde os ativos estão localizados, incluindo: município, unidade, prédio, sala e rack.
- Usuário. Menu onde o administrador irá cadastrar os usuários do sistema, grupos e também os emails para envio dos alertas.
- Ativo. Menu onde o administrador interage com os ativos.

4.4.1 Local

Nesta tela mostrada na figura 14 o usuário poderá acessar os menus de cadastro dos locais onde os ativos estão localizados. Aqui são cadastrados os racks onde os switches estarão localizados. Para cada rack são mantidas informações de onde ele se encontra, partindo de um nível mais genérico que é o município, até o mais específico que é a sala. Também é informado a quantidade de Us do rack para que no cadastro dos switches possa ser registrada sua posição dentro do rack.

Figura 14 – Tela de Cadastro de Racks

The image shows a web application interface for 'CADASTRO DE RACK'. On the left is a dark sidebar menu with icons and text: 'Início' (home icon), 'Voltar' (back arrow icon), 'Operações com Município', 'Operações com Unidade', 'Operações com Predio' (grid icon), 'Operações com Sala', and 'Operações com Rack'. The main content area is titled 'CADASTRO DE RACK' and contains several form fields: 'Município' (dropdown menu showing 'Natal'), 'Unidade' (dropdown menu showing 'Central'), 'Predio' (dropdown menu showing 'SINFO'), 'Sala' (dropdown menu showing 'DATACENTER'), 'Nome do Rack' (text input field containing 'Redes'), and 'Quantidade de Us do Rack' (text input field containing '50'). At the bottom left of the form area is a green rounded button labeled 'Salvar'.

Fonte: Próprio autor

4.4.2 Ativo

Nesta parte do sistema usuário terá acesso aos seguintes menus:

- Operações com Ativos. Menu onde o administrador realizará a gestão dos ativos.
 - Cadastro de Fabricante.
 - Cadastro de Modelo.
 - Cadastro de Ativo.
 - Avaliar Configuração Pendente.
- Operações com Vlans. Menu onde o administrador irá cadastrar os grupos.
 - Listar Todas as vlans Coletadas dos Ativos.
 - Cadastrar Vlans.

Nas seções a seguir são apresentados mais detalhes de alguns desses menus.

4.4.2.1 Cadastro de Ativo

Assumindo que o administrador já tenha cadastrado previamente as informações referentes aos locais, racks, fabricantes e modelos, ele poderá acessar esse menu para realizar o cadastro do ativo. A figura 15 mostra as informações que devem ser fornecidas durante o cadastro.

Embora parte dessas informações não estejam diretamente relacionadas a verificação de alterações nas configurações dos ativos, elas poderão ser utilizadas futuramente para a geração de relatórios importantes, como por exemplo, os locais onde ocorrem mais incidentes. Também são úteis caso se deseje fazer uma verificação do ambiente físico onde o ativo se encontra, para analisar as condições de controle de acesso a sala do rack, por exemplo.

4.4.2.2 Analisando as Configurações Pendentes

Para avaliar as configurações pendentes deve-se seguir os passos mencionados a seguir:

1. Selecionar o ativo para o qual se vai avaliar a configuração pendente.
2. Selecionar a primeira configuração do ativo.
3. Clicar no botão listar configurações.
4. Clicar no botão *Aprova* ou no botão *Reprova* dependendo do que o administrador deseje fazer com a configuração.

Os passos 1 e 2 são mostrados na figura 16 enquanto os passos 3 e 4 na figura 17. Naturalmente, antes de clicar no botão para aprovar ou rejeitar a configuração, deve-se

Figura 15 – Tela de Cadastro de Ativos

The screenshot shows a web interface for registering a switch. On the left is a dark sidebar with navigation options: 'Início' (Home), 'Voltar' (Back), 'Operações com Switch', and 'Operações com Vlans'. The main content area is titled 'CADASTRO DE SWITCH' and contains the following fields:

- Município: Natal
- Unidade: Central
- Predio: SINFO
- Sala: DATACENTER
- Rack: Redes
- IP: x.x.x.x
- Tombo: 123
- Posicao no Rack: 20
- Modelo: H3C
- Usuario SNMP: UsuarioSNMP
- Senha do Usuario: [masked]

A green 'Salvar' button is located at the bottom left of the form area.

Fonte: Próprio autor

Figura 16 – Tela de Aprova Configuração Pendente

The screenshot shows a web interface for pending configurations. On the left is a dark sidebar with navigation options: 'Início' (Home), 'Voltar' (Back), 'Operações com Switch', and 'Operações com Vlans'. The main content area is titled 'CONFIGURAÇÕES PENDENTES' and contains the following fields:

- Ativo: 192.0.2.3
- Configuracao Pendente: 1

A green 'Listar Configurações' button is located at the bottom left of the form area.

Fonte: Próprio autor

analisar os campos que foram alterados. Para simplificar essa tarefa, esses campos são coloridos com a cor vermelha de modo a melhorar sua visualização. Um exemplo desse recurso é mostrado na Figura 17 para a interface *GigabitEthernet\1\0\3*. O fato do valor (*trunk*) na coluna *Tipo de Vlan* está em vermelho indica que o valor registrado no banco para a configuração tida como válida era outro.

4.4.2.3 Listagem de Vlans

Após a coleta das configurações dos ativos, o administrador já pode listar todas as vlans que já foram coletadas pelo sistema. Isso é importante pois o administrador pode inspecionar as vlans existentes em sua organização. A Figura 18 mostra uma lista com todas as vlans coletadas dos switches.

Figura 17 – Tela de Aprova Configuração Pendente

Padrao Interface	Velocidade	Estado Porta	Ligada	Tipo Conector	Tipo Porta	Vlan
GigabitEthernet1/0/1	1000	Up	No-Shutdown	RJ45	trunk	610 611 612 613 614
GigabitEthernet1/0/2	1000	Up	No-Shutdown	RJ45	trunk	610 613
GigabitEthernet1/0/3	1000	Up	No-Shutdown	RJ45	trunk	610 613

Fonte: Próprio autor

Outra funcionalidade disponibilizada é a descoberta de quais ativos possuem determinada vlan. Para isso, ainda na tela mostrada na Figura 18 basta clicar no ícone do computador referente a vlan desejada. A Figura 19, por exemplo, mostra todos os switches que possuem a Vlan 1. Essa informação é muito útil para identificar problemas nas redes de acesso, verificando se uma determinada Vlan existe em todos os equipamentos no caminho entre dois determinados switches, por exemplo, entre um switch de borda e um de *core/backbone*.

Figura 18 – Tela de Listagem de Todas as vlans coletadas dos ativos

Vlan	Visualizar a Configuração da Vlan	Visualizar quais Switches possuem esta vlan
1	🔍	💻
202	🔍	💻
255	🔍	💻
600	🔍	💻
610	🔍	💻
611	🔍	💻
612	🔍	💻
613	🔍	💻
614	🔍	💻

Fonte: Próprio autor

4.5 Web Service de verificação de alterações nos switches

Nesta seção será apresentada parte da implementação da principal operação realizada pela solução proposta que é a verificação das configurações de um ativo para

Figura 19 – Tela de Listagem de Todos os Switches que possuem vlan 1

IP do Switch	Tombo do Switch	Posicao no Rack	Modelo	Nome do Rack
192.0.2.3	1	1	H3C	DIMAP
192.0.2.50	2	2	H3C	Distribuidor

Fonte: Próprio autor

identificar se houve alguma alteração. A implementação foi feita em Java e utiliza o modelo de *Web Services*, conforme pode ser observado na Figura 20. A utilização de Web Service REST facilita a integração com outros sistemas.

Foram utilizadas as anotações da API JAX-RS como `@RoleAllowed`, `@Path` e `@Consumes`. A seguir temos a descrição do comportamento dessas anotações:

- `@Path` - Indica ao servidor onde se encontra o recurso desta classe. As requisições são feitas via http para esta URI que usa esse caminho para ter acesso aos serviços fornecido por esta classe.
- `@RoleAllowed` - Indica ao servidor os grupos que têm permissão para acessar esse recurso.
- `@Consumes` - Determina qual o formato de dados que o servidor aceita.

Como vimos nas descrições acima, o sistema está protegido por autenticação e autorização, e apenas grupos com permissão, são autorizados a realizar operações no sistema.

Figura 20 – Metodo verificaInformacao

```

@POST
@RolesAllowed("ADMIN")
@Path("/coleta")
@Consumes({ MediaType.APPLICATION_JSON })
public void verificarInformacao(Switch comutador){

    Switch ativo = new Switch();
    Switch ativoBanco = new Switch();

    ativo = buscaConfiguracaoSwitch(comutador.getModelo.getfabricante(), comutador.getIp());
    ativobanco = buscaConfiguracaoBanco(comutador.getId());

    if(comparaSwitch(ativo, ativoBanco) ){
        atualizaTimestampConfiguracao(ativo);
    }else{
        adicionaConfiguracaoPendente(ativo);
        geraAlerta(ativo);
    }
}

```

Fonte: Próprio autor

Analisando o código contido na Figura 20 pode-se verificar que ele implementa as operações conforme definido pelo Diagrama de Sequência exposto na Figura 8 da seção 3.2.

No Apêndice B, está descrito as URLs utilizadas no trabalho.

5 Considerações finais

O objetivo geral desse trabalho foi desenvolver um sistema que permita extrair informações de switches, e detectar possíveis alterações nas suas configurações, com o requisito de utilizar uma arquitetura web que possa ser integrada com outros sistemas. Nesse sentido, foi realizada uma revisão bibliográfica nas áreas de monitoramento de ativos de redes e arquitetura de serviços web com o propósito de suportar a proposição do trabalho.

Com respeito a implementação, na base da arquitetura foi utilizado o modelo de *Web Services*, que é uma solução adotada na integração de sistemas e na comunicação entre aplicações diferentes. Com esta tecnologia é possível que novas aplicações possam interagir com aquelas que já existem e que sistemas desenvolvidos em plataformas diferentes sejam compatíveis. Com respeito a obtenção das configurações dos ativos foi utilizado o protocolo SNMP, uma vez que é a solução utilizada no gerenciamento de dispositivos em redes IP e garante a independência de fabricante.

Os objetivos propostos foram alcançados, uma vez que com o uso do sistema os administradores dos ativos poderão tomar as ações necessárias quando alterações indevidas forem realizadas. Outro ponto importante foi a criação da base de configuração, onde se tem todos os dados dos switches e informações pertinentes as sua configurações, garantindo uma documentação atualizada para suportar as operações de gestão de serviços segundo a ITIL.

5.1 Trabalhos Futuros

Ao longo da realização deste trabalho percebeu-se que ele serve como uma base sobre a qual diversas outras funcionalidades podem ser construídas, entre as quais encontram-se:

- Gerar relatórios estatísticos a respeito das alterações, indicando por exemplo, o setor e/ou switch, onde elas mais ocorreram.
- Reverter a configuração do ativo para os valores anteriores considerados válidos quando uma alteração pendente for rejeitada.
- Identificar o usuário que realizou cada alteração nas configurações do ativo.
- Gerar a topologia da rede, indicando quais portas interligam cada par de switches. De posse dessa informação, várias funcionalidades interessantes poderiam ser oferecidas, como por exemplo: i) indicar todos os switches que conseguem se comunicar através de uma dada Vlan; ii) Verificar se os parâmetros (ex: taxa transmissão) nas portas

que interconectam dois switches estão coerentes; iii) verificar se algum protocolo para lidar com loops físicos está configurado nos switches onde isso for necessário.

- Coletar configurações da camada de rede, tais como: tabela de roteamento, dados de protocolos de roteamento dinâmico, entre outros.

5.2 Limitações

A principal limitação encontrada durante o trabalho foi a dificuldade para definir um procedimento de busca genérico que retornasse as configurações dos switches independente de seu fabricante. Embora as MIBs do SNMP tenham essa finalidade, algumas das informações necessárias pelo sistema aqui proposto não fazem parte de MIBs padronizadas, como a MIB-II. Quando esse é o caso, ou a informação de fato não existe para aquele dispositivo, ou ela estará em uma MIB proprietária, e portanto os OIDs serão diferentes, mesmo se tratando da mesma informação. Isso aconteceu, por exemplo, com as informações referentes ao mapeamento de vlan-porta. A solução adota neste trabalho foi permitir o cadastro no banco de dados das MIB privadas de cada fabricante e de uma classe customizada que implementa a coleta da informação.

Referências

- BEZERRA, E. *Princípios de Análise e Projeto de Sistema com UML*. [S.l.]: Elsevier Brasil, 2015. v. 3.
- BURKE, B. et al. *Restful Java with Jax-RS 2.0*. [S.l.]: "O'Reilly Media, Inc.", 2013.
- CERAMI, E. *Web services essentials: distributed applications with XML-RPC, SOAP, UDDI & WSDL*. [S.l.]: "O'Reilly Media, Inc.", 2002.
- CHAPPELL, D. A.; JEWELL, T. *Java web services*. [S.l.]: Tecniche Nuove, 2002.
- FIELDING, R. T. *Architectural styles and the design of network-based software architectures*. Tese (Doutorado) — University of California, Irvine, 2000.
- FILHO, F. C. *ITIL v3 Fundamentos*. [S.l.]: Escola Superior de Redes, 2012.
- KUROSE, J. F. et al. *Redes de Computadores ea Internet uma abordagem top-down*. [S.l.]: Pearson, 2010.
- MIB. <https://www.gta.ufrj.br/grad/04_1/snmp/mib.htm>. Acessado: 2017-05-18.
- SANTOS, M. T. *Gerencia de Redes de Computadores*. [S.l.]: Escola Superior de Redes, 2015.

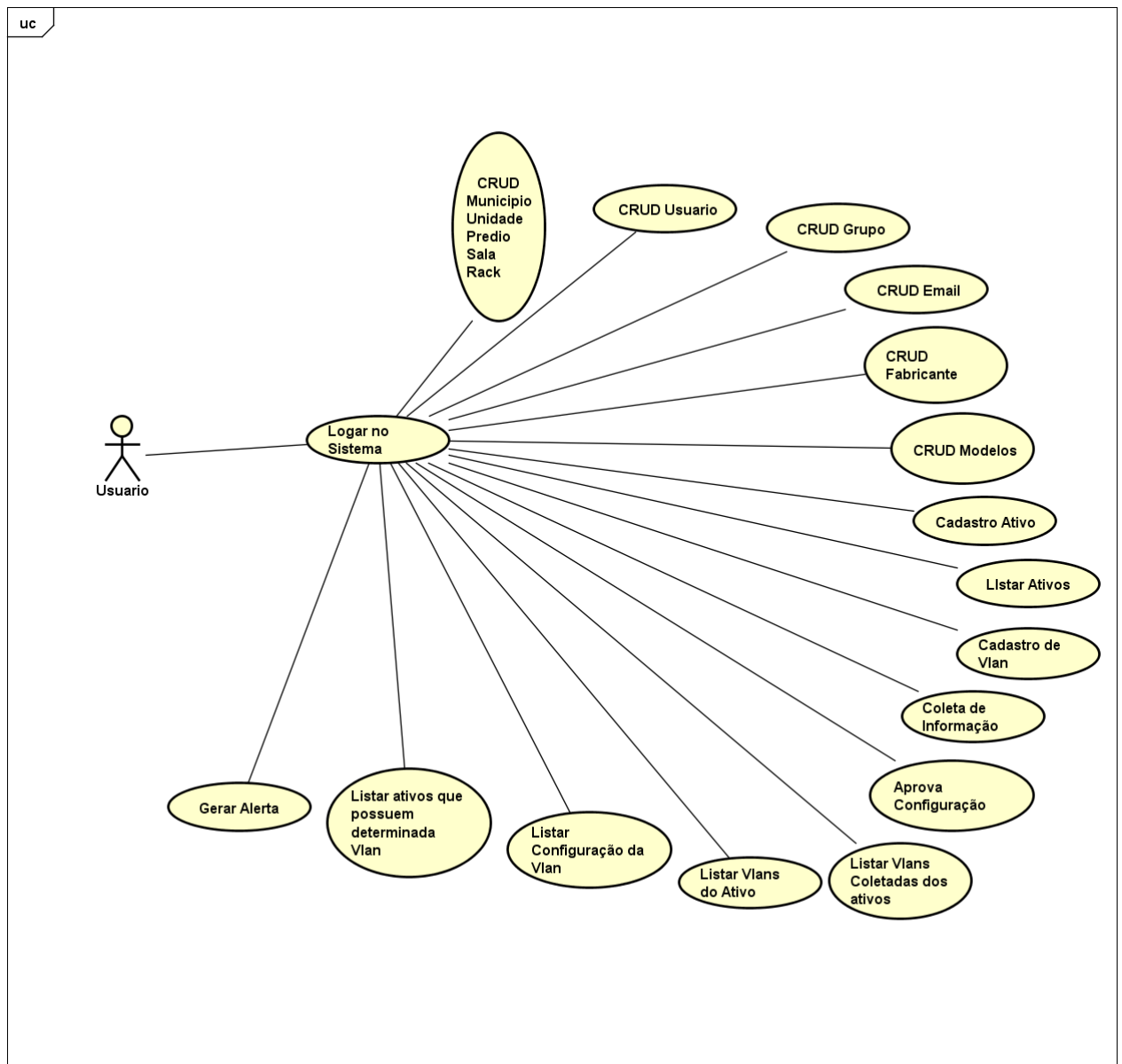
Apêndices

APÊNDICE A – Diagramas de Caso de Uso

Uso

Um caso de uso é a especificação de uma sequência completa de interações entre um sistema e um ou mais agentes externos a esse sistema (BEZERRA, 2015). As características do caso de uso são: sempre iniciados por um ator; sempre retornam um resultado ao autor; especifica uma funcionalidade completa. A seguir serão apresentados alguns casos de uso do sistema.

Figura 21 – Diagrama de Caso de Uso que representa as principais funções do sistema.



Fonte: Próprio autor

[UC 01]: Logar no Sistema.

Descrição: O usuário deverá entrar com seus dados de login e senha. O Sistema deverá permitir o acesso ao conteúdo do sistema somente se os dados estiverem corretos.

Atores envolvidos: Usuário do sistema.

Pré-Condição: O usuário já deve estar cadastrado no sistema.

Cenário Principal de Sucesso:

1. O caso de uso é iniciado com o usuário inserindo o login e senha.
2. O usuário deverá entrar com seus dados.
3. O sistema busca no banco de dados se os dados do usuários estão corretos.
4. O sistema libera o acesso para o usuário.

Cenário Secundário de Falha:

- a) O sistema retorna a mensagem de erro Usuário ou senha errado.

Os exemplos de Cruds que existem no sistema são todos parecidos, diferindo apenas com respeito a entidade. Assim sendo, será explicado apenas um caso de uso referido a um crud, e a partir dele será possível entender os demais que existem no diagrama da Figura 21.

[Crud Rack]:

[UC 02]: Cadastrar Rack.

Descrição: O usuário deverá entrar com os dados referentes a cada rack.

Atores envolvidos: Usuário do sistema.

Pré-Condição: O usuário deve estar logado no sistema e ter permissão para fazer a operação, além de não ter o rack cadastrado na base de dados.

Pós-Condições: Retorno mensagem de rack cadastrado.

Cenário Principal de Sucesso:

1. O usuário deve ir no menu Local.
2. O usuário deve ir no menu operações com Rack.
3. O usuário deve ir em cadastro.
4. O usuário deverá informar os dados do rack.
5. O usuário submete os dados necessários para armazenamento no banco de dados.
6. O sistema validará os dados e retornará mensagem "Cadastro com sucesso".

Cenário Secundário de Falha:

- a) O sistema aborta a validação dos dados e retorna a mensagem de erro, "Cadastro não realizado".

[UC 03]: Editar Rack.

Descrição: O usuário deverá entrar com os dados referentes a cada rack.

Atores envolvidos: Usuário do sistema.

Pré-Condição: O usuário deve estar logado no sistema e ter permissão para fazer a operação.

Pós-Condições: Retorno mensagem de rack alterado com sucesso.

Cenário Principal de Sucesso:

1. O usuário deve ir no menu Local.
2. O usuário deve ir no menu operações com Rack.
3. O usuário deve ir em Listar.
4. O usuário deverá buscar pelo rack.
5. O usuário deve clicar no botão Editar para redirecionar para a pagina de edição.
6. O usuário terá os dados do rack, podendo alterá-los conforme sua necessidade, exceto seu código identificador.
7. O usuário submete a alteração e o sistema validará os dados.
8. O sistema retornará mensagem "Rack Alterado com sucesso".

Cenário Secundário de Falha:

- a) O sistema aborta a validação dos dados e retorna a mensagem de erro, "Edição não realizada".

[UC 04]: Excluir Rack.

Descrição: O usuário deverá entrar com os dados referentes a cada rack.

Atores envolvidos: Usuário do sistema.

Pré-Condição: O usuário deve estar logado no sistema e ter permissão para fazer a operação.

Pós-Condições: Retorno mensagem de rack excluído com sucesso.

Cenário Principal de Sucesso:

1. O usuário deve ir no menu Local.
2. O usuário deve ir no menu operações com Rack.
3. O usuário deve ir em Listar.

4. O usuário deverá buscar pelo rack.
5. O usuário terá os dados do rack, podendo excluí-lo conforme sua necessidade.
6. O usuário submete a exclusão do rack clicando no link Excluir.
7. O sistema retornará mensagem "Rack excluído com sucesso".

Cenário Secundário de Falha:

- a) O sistema aborta a validação dos dados e retorna a mensagem de erro, "Exclusão não realizada".

[UC 05]: Cadastrar Ativo.

Descrição: O usuário deverá entrar com os dados referentes a cada ativo.

Atores envolvidos: Usuário do sistema.

Pré-Condição: O usuário deve estar logado no sistema e ter permissão para fazer a operação, além de não ter o ativo cadastrado na base de dados.

Pós-Condições: Retorno mensagem de ativo cadastrado.

Cenário Principal de Sucesso:

1. O usuário deve ir no menu Switch.
2. O usuário deve ir no menu operações com Switch.
3. O usuário deve ir em cadastro Switch.
4. O usuário deverá informar os dados do ativo.
5. O usuário submete os dados necessários para armazenamento no banco de dados.
6. O sistema validará os dados e retornará mensagem "Cadastro com sucesso".

Cenário Secundário de Falha:

- a) O sistema aborta a validação dos dados e retorna a mensagem de erro, "Cadastro não realizado".

[UC 06]: Cadastrar Vlan.

Descrição: O usuário deverá entrar com os dados referentes a cada vlan.

Atores envolvidos: Usuário do sistema.

Pré-Condição: O usuário deve estar logado no sistema e ter permissão para fazer a operação, além de não ter a vlan cadastrada na base de dados.

Pós-Condições: Retorno mensagem de vlan cadastrada.

Cenário Principal de Sucesso:

1. O usuário deve ir no menu Switch.

2. O usuário deve ir no menu operações com Vlans.
3. O usuário deve ir ao cadastro de vlan.
4. O usuário deverá informar os dados da vlan.
5. O usuário submete os dados necessários para armazenamento no banco de dados.
6. O sistema validará os dados e retornará mensagem "Cadastro com sucesso".

Cenário Secundário de Falha:

- a) O sistema aborta a validação dos dados e retorna a mensagem de erro, "Cadastro não realizado".

[UC 07]: Coleta de Informação.

Descrição: O usuário deverá entrar com os dados referentes a cada ativo para coleta da informação.

Atores envolvidos: Usuário do sistema.

Pré-Condição: O usuário deve estar logado no sistema e ter permissão para fazer a operação, além de o ativo deve estar cadastrado na base de dados.

Cenário Principal de Sucesso:

1. O usuário deve ir no menu Switch.
2. O usuário deve ir no menu operações com Switch.
3. O usuário deve ir ao cadastro Switch.
4. O usuário deverá informar os dados do ativo.
5. O usuário submete os dados necessários para a coleta das informações.
6. O sistema fará a coleta da informação e persistirá as informações no banco.

[UC 08]: Aprova Configurações.

Descrição: O usuário deverá aprovar ou reprovar cada configuração.

Atores envolvidos: Usuário do sistema.

Pré-Condição: O usuário deve estar logado no sistema e ter permissão para fazer a operação, além de o ativo deve ter configurações pendente cadastrada na base de dados.

Pós-Condições: Retorno mensagem de configuração aprovada ou reprovada.

Cenário Principal de Sucesso:

1. O usuário deve ir no menu Switch.
2. O usuário deve ir no menu operações com Switch.

3. O usuário deve ir em listar configurações pendentes.
4. O usuário deverá informar o ativo.
5. O usuário deverá informar a configuração a ser aprovada.
6. O usuário deverá clicar no botão Listar Configurações para verificar as alterações da configuração.
7. O usuário deverá clicar no botão Aprova ou Reprova.
8. O sistema validará os dados e persistirá a informação no banco.

Cenário Secundário de Falha:

- a) Caso o usuário aprove ou reprove uma configuração superior a mais antiga o sistema retorna uma mensagem de erro, "Existe uma configuração anterior para ser aprovada ou reprovada".

[UC 09]: Listar Vlans Coletadas dos ativos.

Descrição: O usuário listará todas as vlans que já foram coletadas dos ativos.

Atores envolvidos: Usuário do sistema.

Pré-Condição: O usuário deve estar logado no sistema e ter permissão para fazer a operação.

Pós-Condições: Retorno as vlans coletadas.

Cenário Principal de Sucesso:

1. O usuário deve ir no menu Switch.
2. O usuário deve ir no menu operações com Vlans.
3. O usuário deve ir em Listar Todas as Vlans Coletadas dos Switchs.
4. O sistema retornará uma lista das vlans já coletadas.

[UC 10]: Listar Vlans do Ativo.

Descrição: O usuário listará todas as vlans do ativo.

Atores envolvidos: Usuário do sistema.

Pré-Condição: O usuário deve estar logado no sistema e ter permissão para fazer a operação.

Pós-Condições: Retorno as vlans do ativo.

Cenário Principal de Sucesso:

1. O usuário deve ir no menu Switch.
2. O usuário deve ir em operações com Switch.

3. O usuário deve ir em Listar Switchs.
4. O usuário deve buscar qual ativo quer listar as vlans.
5. O usuário deve ir clicar no ultimo icone da linha que corresponde as informações do ativo.
6. O sistema retornará as vlans já do ativo.

[UC 11]: Listar Configurações das Vlans.

Descrição: O usuário listará a configurações das vlans.

Atores envolvidos: Usuário do sistema.

Pré-Condição: O usuário deve estar logado no sistema e ter permissão para fazer a operação.

Pós-Condições: Retorno uma lista das configurações das vlans cadastradas.

Cenário Principal de Sucesso:

1. O usuário deve ir no menu Switch.
2. O usuário deve ir no menu operações com Vlans.
3. O usuário deve ir em Listar Configurações das Vlans.
4. O sistema retornará uma lista com as configurações das vlans.

[UC 12]: Listar Ativos que possuem determinada Vlan.

Descrição: O usuário listará os ativos que possuem determinada vlan.

Atores envolvidos: Usuário do sistema.

Pré-Condição: O usuário deve estar logado no sistema e ter permissão para fazer a operação.

Pós-Condições: Retorno uma lista de ativos que possuem determinada vlan.

Cenário Principal de Sucesso:

1. O usuário deve ir no menu Switch.
2. O usuário deve ir no menu operações com Vlans.
3. O usuário deve ir em Listar Todas as Vlans Coletadas dos Switchs.
4. O usuário deve clicar no ultimo icone de um computador, da linha que corresponde a vlan que o usuário quer verificar.
5. O sistema retornará uma lista com os ativos que possuem esta vlan.

[UC 13]: Gerar alerta.

Descrição: O sistema gerará um alerta para o administrador cadastrado.

Atores envolvidos: Usuário do sistema.

Pré-Condição: Existirá uma conta de usuário apenas para esta função, pois ele deverá ficar logado sempre no sistema para envio dos alertas.

Pós-Condições: Geração de alerta ao administrador.

Cenário Principal de Sucesso:

1. O sistema percorre no banco se existe configuração pendente..
2. O sistema gera um alerta para o administrador, que existe configuração pendente a ser aprovada.

APÊNDICE B – Principais *Web Services*

A seguir, nas Tabelas 6 e 7 são descritos os principais recursos do sistema que são disponibilizados através de *Web Services*.

Tabela 6 – Tabela de metodos do switch

Recurso	Método HTTP	Semântica	Descrição do Parametro
/rest/switch/todos/	GET	Lista todos os switches	
/rest/switch/todos/pendentes/	GET	Lista todos os switches com configurações pendentes	
/rest/switch/interface_p/{id}	GET	Listar configuração pendente mais antiga do switch	id do switch
/rest/switch/configuracoes/{id}	GET	Lista todas as configurações do switch	id do switch
/rest/switch/switch/rack/{id}	GET	Lista todos os Switches de um Rack	id do rack
/rest/switch/vlan/{numeroVlan}	GET	Listar todos os switches que possuem determinada vlan	numeroVlan da vlan especificada
/rest/switch/{id}	GET	Listar a configuração valida do switch	id correspondente ao switch especificado
/rest/switch/	POST	Adiciona um switch	
/rest/switch/verificaInformacao	POST	Faz a coleta da informação	
/rest/switch/{id}	PUT	atualiza a configuração do switch	id do switch a ser atualizado
/rest/switch/{id}	DELETE	Remove o switch	id do switch que deseja deletar

Tabela 7 – Tabela de metodos da vlan

Recurso	Método HTTP	Semântica	Descrição do Parametro
/rest/switch/vlans/	GET	Lista todas as vlans coletadas	
/rest/switch/vlans/id/{id}	GET	Lista todas as vlans de um switch pelo id do ativo	id do switch especificado
rest/switch/vlans/tombo/{tombo}	GET	Lista todas as vlans de um switch pelo tombo do ativo	tombo do ativo especificado
/rest/vlan	POST	Adiciona uma vlan	
/rest/vlan/	GET	Lista todas as vlans cadastradas	
/rest/vlan{id}	GET	Busca a vlan por seu id	id da vlan especificada
/rest/vlan/nome/{id}	GET	Busca a vlan por seu nome	nome da vlan especificada
/rest/vlan/{id}	PUT	Atualiza a vlan	id da vlan especificada
/rest/vlan/{id}	DELETE	Remove a vlan	id da vlan especificada