

EAGLEVIEW – SISTEMA DE GERENCIAMENTO REMOTO DE SERVIDORES

**Nelson Monteiro da Silva Neto¹, Thiago de Paiva Silva²,
Alberto Correia³**

¹⁻³Universidade do Vale do Paraíba – Faculdade de Ciência da Computação
Av. Shishima Hifumi, 2911 CEP 12244-000 São José dos Campos – SP – Brasil,
nelsonnetto@yahoo.com.br¹, thiagooes@terra.com.br², alberto@univap.br³

Resumo - Com a crescente evolução das redes, o aumento da complexidade e da quantidade de dados trafegados e a necessidade de se obter mais segurança sobre os dados, tornou-se imperativo a implementação de ferramentas que gerenciassem estes serviços internamente. Neste cenário o software EagleView, desenvolvido totalmente com softwares e padrões open-source, como JAVA e a API JAVA JXTA, através de seus módulos cliente e servidor realiza a análise em tempo real dos principais serviços de um servidor, exibindo ao administrador informações visuais e logs em formato texto que o auxiliam nas decisões em relação ao servidor.

Palavras-chave: Administração de Redes, Monitoramento, Java, JMX, JXTA, P2P, HSQLDB
Área do Conhecimento: Ciência da Computação

Introdução

Em virtude do aumento da dependência das empresas por serviços de TI (Tecnologia da Informação) em seus ramos de atuação; sejam eles lógicos ou físicos; existe a contrapartida e o aumento da competitividade entre as empresas que gerenciam estes serviços. Neste contexto as demandas de competitividade das equipes de TI, exigem racionalização no uso dos recursos, inclusive humanos, de gerenciamento de rede.

Conseqüentemente, a capacidade de gerenciamento remoto de servidores, equipamentos de rede e dispositivos de automação, torna-se um aspecto crítico na administração da rede. Uma infra-estrutura de gerenciamento *out-of-band* eficaz permite a operação de *data-centers* ou CPDs (Centro de Processamento de Dados) desassistidos, com gerenciamento de rede centralizado e mais automatizado [1][2]. Os resultados são custos de manutenção reduzidos, resolução mais rápida de problemas e maior disponibilidade da rede.

Ferramentas de gerenciamento baseadas no protocolo SNMP (*Simple Network Management Protocol*) dependem da rede de dados para transportar as informações de gerenciamento. Isto é chamado de gerenciamento de rede *in-band*. Este é um modo conveniente para executar as atividades comuns de gerenciamento, monitorar a rede e atuar sobre uma falha de aplicação [3][4].

Mas se ocorre uma falha na rede ou se a camada de protocolos de rede de um determinado servidor deixa de responder, a conectividade com os nós da rede é perdida e as ferramentas de gerenciamento *in-band* não são capazes de diagnosticar ou recuperar o sistema. É por isso

que gerenciar redes de missão crítica requer caminhos de gerenciamento alternativos ou gerenciamento *out-of-band*. Os nós gerenciados podem então ser alcançados quando perde a conectividade com a rede ou ainda quando a rede está inoperante.

Em *data-center* ou CPD típicos, o gerenciamento *in-band* é eficaz, para prevenir ou atuar, em menos da metade das situações que podem causar interrupções não planejadas. Uma parte significativa destas interrupções é causada por falhas de infra-estrutura (falhas de hardware, travamento do sistema operacional, problemas de rede, falhas de energia, etc...) e somente podem ser investigadas por ferramentas de gerenciamento *out-of-band*.

Combinando mecanismos *in-band* e *out-of-band* e automatizando rotinas de gerenciamento, o gerenciamento de TI pode garantir que os procedimentos necessários serão seguidos para manter a rede de dados em pleno funcionamento, minimizando e ou prevenindo falhas de aplicações e de infra-estrutura, assim como erros humanos de operação.

Materiais e Métodos

Para o desenvolvimento deste sistema utilizamos os seguintes conceitos:

- Hardware
 - Um computador pessoal que terá acesso a Internet;
 - Um equipamento capaz de executar a aplicação que servirá de “hospedeiro” da versão cliente do sistema;
 - Um ou mais servidores que serão monitorados. Para este trabalho, entende-se servidor como um sistema de computação (hardware) que fornece serviços a uma rede de computadores. Esses serviços podem ser de diversa natureza, por exemplo, arquivos e correio eletrônico ;
- Software

Para o desenvolvimento da ferramenta de monitoramento, foi escolhida a linguagem de programação JAVA, tanto para o desenvolvimento da versão cliente quanto para o desenvolvimento da versão servidor de monitoramento, tendo em vista que essa é uma linguagem orientada a objetos, multiplataforma e que possui uma larga escala de utilização e aceitação no mercado, sendo gratuita [5]. O desenvolvimento contou também com a utilização da API JMX (Java Management Extensions) é uma tecnologia Java que provê ferramentas para administrar e monitorar aplicações, objetos de sistemas, dispositivos (como impressoras) e serviços de redes; e ainda com a API JAVA JXTA, que é uma especificação aberta para comunicação entre dispositivos, sem considerar sua localização física e tecnologia de rede no qual se encontra instalado [6][7]. JXTA é considerada a plataforma P2P mais madura atualmente [8].

Foi utilizado como repositório de dados, o HSQLDB, escrito totalmente na linguagem JAVA, sendo uma solução simples, que utiliza poucos recursos e que possui bom desempenho, largamente utilizado em aplicações que necessitam interagir com uma camada de persistência, através da linguagem SQL [9][10].

- Sistema Operacional

Seu desenvolvimento e utilização deu-se em sistemas operacionais Linux (distribuições Red Hat e Slackware) e Microsoft Windows.

- Arquitetura do sistema:

Na concepção do sistema foi necessário classificá-lo em duas versões:

- Servidor: que tem a finalidade de analisar a situação do serviço monitorado do servidor, de tempo em tempo, e que deverá reportar a versão cliente caso haja alguma anomalia (serviço parado), uma segunda função dessa versão será de tratar o comando enviado pela versão cliente e traduzi-la conforme o sistema operacional disponível no equipamento que controla essa versão.

- Cliente: que tem a finalidade de receber a informação da versão servidora e disponibilizá-la de uma maneira amigável e transparente para o usuário (administrador), uma segunda função dessa versão será de disponibilizar as funcionalidades de início, re-início e parada dos serviços do servidor monitorado.

A comunicação mencionada dar-se-á utilizando as técnicas da tecnologia JXTA, usando protocolo P2P e um meio de comunicação (Internet).

A Figura 1 ilustra a estrutura do EagleView.

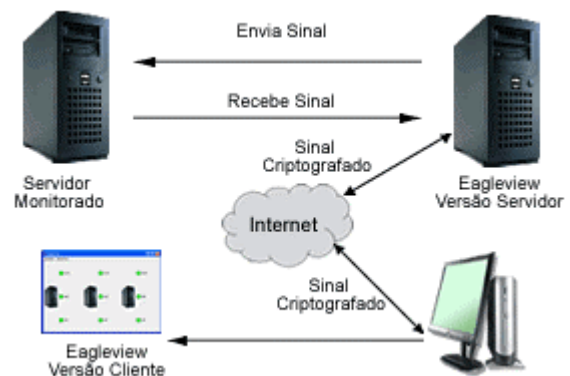


Fig.1 – Estrutura do EagleView. O EagleView de tempo em tempo envia e recebe sinais do servidor monitorado com o intuito de analisar esses sinais. Caso haja alguma anomalia (serviço parado), o sistema criptografa e envia um sinal de alerta via internet, usando protocolo P2P, para uma estação cliente que tratará o sinal recebido e apresentará a situação atual ao usuário do sistema cliente (administrador do servidor monitorado) e se for o caso, o sistema cliente enviará um comando de resposta, conforme a vontade do usuário, de volta ao servidor monitorado.

Resultados e Discussão

Os resultados obtidos com o protótipo do EagleView, revelam pleno funcionamento da interface gráfica apresentando os dados obtidos pelo sistema de análise de serviço de uma forma amigável ao usuário e implementa de uma maneira simples e transparente a funcionalidade de início, re-início e parada de serviços.

A figura 2 ilustra o software durante o monitoramento dos serviços.

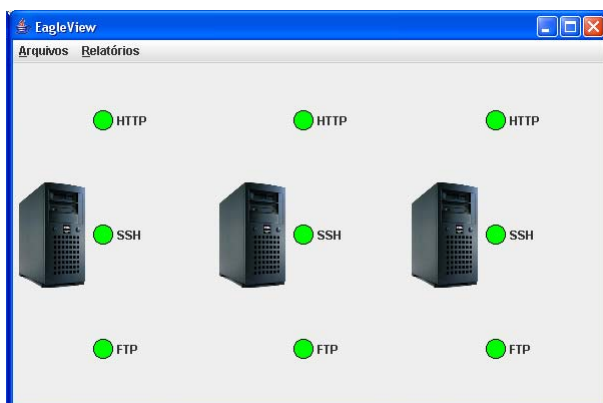


Fig.2 – Interface gráfica do sistema EagleView. Ao centro temos a ilustração dos servidores analisados. O sistema apresenta a situação atual dos serviços monitorados através de LEDs de duas cores (Verde – ligado e Vermelho – Desligado). Na barra de menu, na parte superior do sistema, temos a opção de relatórios.

Após o desenvolvimento completo do projeto, teremos a seguinte estrutura do software:

Monitoramento

O módulo cliente do EagleView estará enviando e recebendo sinais sistematicamente à ao módulo servidor do EagleView, verificando se os servidores monitorados e seus serviços estão no ar e armazenando as informações recebidas em repositório de dados, com o intuito de gerar relatórios ao usuário. Durante essa troca de informações o aplicativo criptografará todos os dados trocados mantendo o sigilo destas informações.

Seu módulo servidor estará analisando o servidor em tempo real, e caso ocorra alguma anomalia ele enviará um sinal para a solução para o módulo cliente com as informações necessárias, sendo estas:

- Tempo decorrido desde a última reinicialização do gerenciamento do sistema na rede;
- Serviços Monitorados;
- Estado dos Serviços Monitorados;
- Observações;

Essas informações serão vitais para as tomadas de decisão que se darão à detecção de quaisquer anomalias.

A figura 3 ilustra o software durante a exibição do relatório de um dos servidores monitorados.

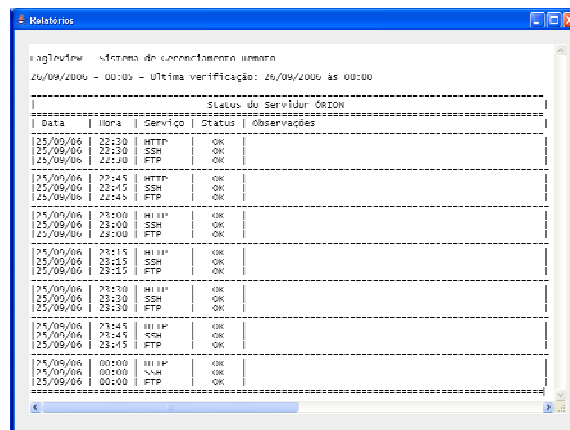
The image shows a window titled 'Relatórios' with a menu bar 'Arquivos' and 'Relatórios'. The main content is a table with the following columns: 'Data', 'Hora', 'Serviço', 'Status', and 'Observações'. The table contains multiple rows of data, all showing 'OK' status for services like HTTP, SSH, and FTP at various times on 25/09/06. The window title bar also indicates 'Sistema de Monitoramento EagleView' and 'Última verificação: 26/09/2006 às 00:00'.

Fig.3 – Tela de exibição de relatórios sobre o estado em tempo real dos serviços dos servidores monitorado, onde consta data e hora da última visualização, nome do servidor monitorado, serviços, Status e Observações.

Ações

Com as informações fornecidas pelo monitoramento o administrador acionará o EagleView e através do mesmo analisará a melhor ação a ser tomada, bem como se for o caso, parar e iniciar os serviços disponíveis no servidor e reiniciar o serviço gerenciado.

Pretensões Futuras

Existe a pretensão de criar uma interface que implemente as mesmas funcionalidades da versão cliente para sistemas móveis, tais como, celulares, Palmtops e equipamentos do gênero, visto que essa é uma área que tem crescido substancialmente nos últimos anos e por ser uma outra facilidade que poderá ser um diferencial marcante para o EagleView.

Conclusão

Com esta aplicação será possível reduzir custos com mão de obra dedicada para o controle dos servidores, realização de testes e análise de como os servidores se comportam em tempo real, permitindo ao administrador um controle cada vez maior de seus equipamentos e consiga assim obter melhores resultados dos mesmos.

Esta aplicação e seu desenvolvimento provam ainda as funcionalidades e implementações conceituais e técnicas comentadas neste artigo.

Referências

- [1] Definições de Conceitos out-of-band. Disponível em: <http://en.wikipedia.org/wiki/Out-of-band>. Acesso em 25 mai.2006.
- [2] Definições de Conceitos data-centers. Disponível em: http://en.wikipedia.org/wiki/Data_center. Acesso em 25 mai.2006
- [3] Definições de Conceitos protocolo SNMP. Disponível em: <http://en.wikipedia.org/wiki/Snmp>. Acesso em 15 jun.2006
- [4] Definições de Conceitos in-band. Disponível em: <http://en.wikipedia.org/wiki/In-band>. Acesso em 25 mai.2006
- [5] Definições de Conceitos JAVA. Disponível em: <http://www.java.sun.com>. Acesso em 30 mai.2006
- [6] Definições de Conceitos JMX. Disponível em: <http://en.wikipedia.org/wiki/JMX>. Acesso em 20 abr.2006
- [7] Definições de Conceitos JXTA. Disponível em: <http://www.jxta.org/>. Acesso em 20 abr.2006
- [8] Definições de Conceitos P2P. Disponível em: http://pt.wikipedia.org/wiki/Peer_to_peer. Acesso em 20 abr.2006
- [9] Definições de Conceitos HSQLDB. Disponível em: <http://pt.wikipedia.org/wiki/HSQLDB>. Acesso em 20 abr.2006
- [10] Definições de Conceitos SQL. Disponível em: <http://pt.wikipedia.org/wiki/SQL>. Acesso em 20 abr.2006