

Agilizando a detecção, análise e correção de erros em um sistema de banco de dados replicados de forma assíncrona multi-master

Leonardo Moerschberger, Miguel Dresch
Curso de Tecnologia em Redes de Computadores
Sociedade Educacional Três de Maio (SETREM)
Av. Santa Rosa, 2405 Três de Maio/RS Brasil
{leosted,miguel_dresch}@hotmail.com

Claudio Schepke
Instituto de Informática
Universidade Federal do Rio Grande do Sul
Caixa Postal 15.064 - 91501-970 Porto Alegre/RS
schepke@gmail.com

Resumo—Este artigo tem como objetivo estruturar um ambiente para a detecção, análise e correção de erros envolvidos em um sistema de banco de dados replicados de forma assíncrona multi-master. Para viabilizar a estrutura proposta utilizou-se da ferramenta Zabbix, responsável pelo monitoramento e gerenciamento dos servidores com o banco de dados e o OpenVPN para possibilitar uma comunicação segura entre o Zabbix server e os Zabbix Agentes que estão instalados nos servidores remotos. Através da estrutura proposta é possível detectar o surgimento de um problema e agilizar a manutenção dos servidores.

I. INTRODUÇÃO

No mercado existem as mais variadas soluções de comunicação para interligar um ponto distante do outro, dentre as quais destacam-se a Internet, a qual é utilizada pela grande maioria das empresas como solução, transformando a mesma na maior rede de comunicação atual.

Com foco na comunicação, deve-se levar em consideração a característica da disponibilidade, pois existem inúmeros fatores externos que podem influenciar na mesma, visto que nenhuma solução de comunicação garante 100% de disponibilidade.

Com o crescimento gradual de algumas empresas surge a necessidade da expansão das mesmas, estando estas presentes em vários pontos geograficamente distintos, surgindo à necessidade de integração da base de dados.

Uma solução que viabiliza este cenário é a utilização de replicação de bancos de dados.

Diante da situação em que os meios de comunicação não podem garantir 100% de disponibilidade e os custos envolvidos em redundância da comunicação são altos, a replicação assíncrona se torna ideal para uma estrutura de sites onde a comunicação não é permanente.

No entanto, com a utilização da replicação assíncrona há a possibilidade de conflitos e a dificuldade de manutenção da semântica transacional, já que, a replicação assíncrona possibilita a atualização de qualquer site.

Considerando essa possibilidade faz-se necessário a detecção desses conflitos, que devem ser tratados a nível global.

Este artigo propõe uma abordagem prática utilizando-se de um conjunto de ferramentas visando facilitar a detecção, análise e correção dos possíveis problemas encontrados no modelo de replicação assíncrona do SGDB

(Sistema de Gerenciamento de Banco de Dados) Sybase Anyware no modelo multi-master.

Este artigo está dividido em V seções. A Seção II apresenta os trabalhos relacionados. Na Seção III esta a análise da situação. Na Seção IV destaca-se os componentes e funcionamento das ferramentas envolvidas. Na Seção V apresenta-se os resultados obtidos. Por fim, a Seção VI apresenta a conclusão.

II. TRABALHOS RELACIONADOS

DATE (2003) [1] comenta na replicação assíncrona que a partir do momento em que são realizadas as transações entre as "réplicas" elas são propagadas com retardo, ou seja, são transmitidas momentos depois da transação, tendo como consequência um tempo de latência em que as réplicas não são idênticas. A maioria das soluções utiliza replicação assíncrona pela leitura de logs de transações ou de uma fila estável de atualizações que precisam ser propagadas.

DATE (2003) [1] salienta a desvantagem da replicação assíncrona: a existência de inconsistência de dados, sendo uma tarefa difícil de evitar e corrigir, podendo haver conflitos relacionados à ordem em que as atualizações são propagadas.

BURETTA (1997) [2] demonstra dois modelos para definir o detentor do direito de propriedade sobre os dados para aplicações que utilizam replicação: modelo mestre/escravo, podendo atualizar somente uma base de dados, e o modelo multi-master, onde se pode atualizar qualquer base de dados do site.

No modelo mestre/escravo, existe uma base de dados, chamada mestre, que pode ser armazenada de forma centralizada ou distribuída. Essa base mestre é a única passível de atualização. As demais réplicas, denominadas escravos, permitem apenas operações de leitura e são atualizadas por meio de replicação assíncrona.

No modelo multi-master não existe o conceito de mestre. A replicação de dados pode ocorrer de forma síncrona, na qual as réplicas são atualizadas imediatamente, ou assíncrona, onde as réplicas podem ser atualizadas após um tempo determinado, ou após a ocorrência de algum evento específico.

Segundo KEMME & ALONSO (2000) [3], é frequente deixar a tarefa de manter a consistência num modelo de replicação assíncrona a cargo da aplicação usuária.

III. ANÁLISE DA SITUAÇÃO

Alguns aspectos importantes que foram decisivos na escolha da replicação assíncrona multi-master foram: disponibilidade, desempenho, autonomia, flexibilidade, escalabilidade e baixo custo em comunicação. Sabendo dos problemas envolvidos na replicação de modelo assíncrono multi-master, foi realizada uma avaliação nas regras de negócio da aplicação, para saber se as mesmas são pertinentes quanto a sua utilização.

Estudos sobre como aplicar a replicação assíncrona multi-master no *SGDB Sybase Anywhere* foram necessários, bem como a análise de logs, caso erros no processo de replicação ocorressem.

Um estudo sobre a real necessidade de controle e monitoramento que a empresa pretendia sobre a estrutura de bancos replicados também foram realizados.

IV. COMPONENTES E FUNCIONAMENTO

Analisando os problemas e necessidades deste projeto, foram elencadas as ferramentas Zabbix e OpenVPN, para atuarem no ambiente, tendo ciência de que todo ambiente projetado é baseado no Sistema Operacional Linux.

Para um melhor entendimento da proposta desse artigo serão descritos a seguir todos os componentes envolvidos e, após, os procedimentos envolvidos para o funcionamento da solução.

Banco de dados: para DATE (2003) [1] um banco de dados é uma coleção de dados persistentes, usada pelos sistemas de aplicação de uma determinada empresa.

Sistema de gerencia de banco de dados: é uma camada que fica entre o banco de dados armazenado fisicamente e os usuários do sistema. O *SGDB* isola os usuários do banco de dados dos detalhes de nível de hardware.

Replicação de Banco de Dados: a replicação de banco de dados consiste em propagar registros de um determinado banco de dados ou o próprio banco de dados como replica para sites distintos.

Zabbix Server: responsável por coletar informações da monitoração de todos hosts envolvidos na replicação, e pelo envio de alertas.

Zabbix Agent: responsável por coletar e enviar as informações monitoradas do host, onde se encontra instalado, para o Zabbix Server.

Zabbix Interface WEB: responsável por apresentar de forma visual, através de uma interface web, os dados coletados por Zabbix Server. Nele, também são configuradas as condições e alertas mediante os dados que serão coletados.

Open VPN: fez-se necessário a utilização de VPN para haver uma comunicação fim a fim entre o Zabbix Agent e o Zabbix Server, o que possibilita a execução de script no host cliente através da interface WEB do Zabbix. VPN pode ser utilizada para acesso remoto aos hosts clientes quando houver necessidade, estando ativa a comunicação com o host.

Script de manutenção: dentre todos os componentes esse, por sua vez, é o que exerce as operações no host do cliente, sendo responsável por iniciar, parar, analisar logs do banco de dados e pela replicação. A partir dessas

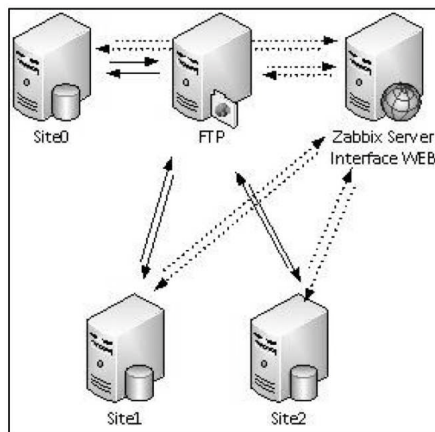


Figura 1. Estrutura do ambiente.

operações, o script informa ao Zabbix Agent através de logs específicos, como essas operações foram efetuadas, se ocorreram erros, se foram bem sucedidas ou se houve algo que possa ter impedido a operação. O script de manutenção é agendado no próprio sistema operacional para efetuar o processo de replicação frequentemente, de acordo com um tempo pré-estabelecido.

FTP: responsável por receber todas as mensagens de operações de dados de todos os bancos de dados da replicação.

Funcionamento: o funcionamento da replicação é feito através de troca de mensagens criptografadas utilizando o protocolo FTP, com mensagem de confirmação de recebimento.

O cenário de funcionamento da solução como um todo está ilustrado na Figura 1.

Considerando um primeiro evento no *Site1*, como uma inserção de dados no banco de dados, o script de manutenção será executado pelo agendador de tarefas do sistema com a função de iniciar o processo de replicação caso haja conexão estável via Internet. Com o processo de replicação iniciado, o *Site1* verifica se existe(m) mensagem(s) para si. Caso haja, essas mensagens são aplicadas no banco de dados em questão (*Site1*). Após o término dessa operação, o *Site1* envia a mensagem de inserção para o servidor FTP. Feito isso o processo é finalizado.

Todos os procedimentos de receber, enviar mensagens e aplicar as mensagens recebidas, podem ser auditados por um log. Após o término desse ciclo de replicação, o script de manutenção verificará se houve algum erro no ciclo ou não. O script gera um log onde o Zabbix Agent está configurado para capturar o valor desse log. Se houver erro, o mesmo gera um valor específico. Caso contrário, outro valor é armazenado.

O Zabbix Server contata o Zabbix Agent requerendo a

informação daquele log. Dependendo do valor que Zabbix Server receber, este o interpreta como um problema ou não, enviando um alerta via e-mail para um contato ou grupo de contatos, caso um problema seja detectado.

O *Site0* contém o banco de dados que é responsável por controlar as replicações e também é o único detentor de todas as mensagens envolvidas. Em um segundo momento, o *Site0* invoca o script de manutenção para iniciar o processo de replicação via agendador de tarefas.

Os procedimentos do *Site0* são os mesmos do *Site1*, porém com uma diferença. Sendo o único detentor de todas as mensagens envolvidas fica a cargo do mesmo aplicar-se a alteração do *Site1*, caso não haja erro(s), e repassar ao *Site2* a mensagem do *Site1*. Lembrando que todas essas mensagens são direcionadas a um único servidor de FTP. Após o ciclo de replicação ter terminado, o script de manutenção gera o log informando a situação. Assim, o processo de geração de log se iguala ao do *Site1*.

E por fim, o *Site2* faz os procedimentos conforme *Site1*.

A manutenção e a análise pode ser feita através da interface WEB do Zabbix, invocando funções no host desejado. Dentre algumas características, destacam-se:

- Apresentação de logs gerados pelo processo de replicação, incluindo erros.
- Execução Backup do banco de dados.
- Inicialização do processo de replicação.
- Desativação do processo de replicação.
- Inicialização ou paralização do banco de dados.

Aproveitando as funcionalidades da ferramenta Zabbix, foram monitorados vários outros itens de importância elevada, como: Espaço em disco, tempo de uptime, latência de ping do host até o servidor de FTP, número de mensagens acumuladas, data da última replicação, dentre outros.

V. RESULTADOS OBTIDOS

Com a nova estrutura instalada a manutenção dos servidores se torna mais ágil e com maior precisão, pelo fato do Zabbix Server alertar quando ocorrer um problema e qual a sua causa, sem a necessidade do usuário final informar que esta tendo problemas, já que o monitoramento permite a execução de ações e notificações, informando aos responsáveis sempre que uma anormalidade for detectada.

Outra forma de agilizar a manutenção é utilizando-se da rede VPN que permite o acesso remoto, utilizando-se do protocolo SSH, permitindo assim a manutenção remotamente quando o problema esta relacionado aos softwares instalados nos servidores, só não será possível fazer a manutenção quando o problema for nível físico.

VI. CONCLUSÃO

Neste trabalho foi apresentada uma proposta da implementação de monitoração utilizando a ferramenta Zabbix, através de redes seguras com VPN, para agilizar a detecção de erros em um ambiente de replicação assíncrona multi-master, bem como facilitar a análise e correção dos mesmos.

Tal solução apresentou-se viável, garantindo a segurança do fluxo das informações. No momento a solução está ativa em uma empresa onde são monitorados mais de 60 servidores.

REFERÊNCIAS

- [1] C. J. DATE, *Introdução a sistemas de banco de dados*, 4th ed. Rio de Janeiro: Editora Campus, 2003.
- [2] M. BURRETA, M. *Data Replication: tools and techniques for managing distributed information*. EUA: John Wiley e Sons, Inc, 1997.
- [3] K. Bettina and G. Alonso, "Don't be lazy, be consistent: Postgres-r, a new way to implement database replication," in *Proceedings of the 26th International Conference on Very Large Data Bases*, ser. VLDB '00. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 2000, pp. 134–143. [Online]. Available: <http://dl.acm.org/citation.cfm?id=645926.671855>