Serviço de controle de domínio utilizando Linux, Samba, LDAP e ClamAV

Leonardo Bruno Lopes*, Paulo E. M. Almeida*

**Departamento de Recursos em Informática*
Centro Federal de Educação Tecnológica de Minas Gerais
Av. Amazonas, 5253, Belo Horizonte, MG, Brasil.

Email: leonardo@dri.cefetmg.br, pema@dri.cefetmg.br

Resumo – Este artigo tem por objetivo descrever a solução adotada como serviço de controle de domínio no CEFET-MG. Um serviço desta natureza passou a ser interessante para a Instituição ante o crescimento do número de estações de trabalho conectadas à rede, e o conseqüente aumento da demanda por recursos compartilhados como impressoras e diretórios. Na implementação desta solução foram utilizados apenas softwares e ferramentas de código aberto, em virtude de possuírem ótima relação entre custo e benefício. O artigo relata também as dificuldades encontradas, problemas e soluções envolvidas no processo de implantação, além de uma análise sobre as vantagens, desvantagens e resultados obtidos.

Termos de Indexação – domínio, samba, antivírus, software livre

1. Introdução

Nos últimos anos, com os crescentes incentivos do Governo Federal, o Centro Federal de Educação Tecnológica de Minas Gerais (CEFET-MG) experimentou uma fase de intenso crescimento. Assim, foram ampliados o número de cursos ofertados e de alunos, além do quadro de professores e de servidores técnico-administrativos. Tal expansão, somada aos investimentos diretos em tecnologia da informação, resultou no aumento do número de estações de trabalho conectadas à rede. O CEFET-MG possui hoje 10 *campi* no estado e 4 unidades conveniadas, contando com aproximadamente mais de 1500 computadores, sendo que a maior parte destes está nos *campi* I e II em Belo Horizonte.

Tornou-se desejável utilizar um método de controle e autenticação de usuários centralizados para estas estações. A solução natural parecia ser a implantação de controladores de domínio baseados em Windows Server com Active Directory, visto que a grande maioria das estações do CEFET-MG executam algum sistema operacional da família Windows. Esta solução, no entanto, era contrária às diretrizes da Instituição que, a partir do seu departamento de TI, o Departamento de Recursos em Informática - DRI, visa aumentar a eficiência e a estabilidade dos serviços de informática com redução simultânea de custos. Decidiu-se, portanto, utilizar como alternativa GNU/

Linux, Samba e outras ferramentas de código aberto para implantar os controladores de domínio.

Nas seções 2, 3 e 4 deste trabalho são apresentadas as ferramentas utilizadas na construção da solução, bem como a metodologia utilizada na integração e configuração das mesmas de modo a obter os resultados esperados. Em seguida, nas seções 5 e 6, são analisados os benefícios obtidos para a Instituição e os desdobramentos para a comunidade de usuários e desenvolvedores de *Software* Livre.

2. Samba com LDAP

Samba é um *software* de código aberto lançado em 1992 e que provê serviços de autenticação, compartilhamento de arquivos e impressoras para qualquer computador cliente que suporte o protocolo SMB[1]/CIFS[2], como as diversas versões do sistema operacional Windows, sendo compatível até mesmo com as mais recentes [1].

Para funcionar corretamente, o Samba precisa dispor de uma base de dados com informações sobre as contas de usuários e de máquinas e isto pode ser feito utilizando-se um dos 4 modos (*backends*) disponíveis: (1) *smbpasswd*, baseado em arquivos semelhantes ao *passwd* UNIX, (2) *tdbsam*, que usa um banco de dados trivial (TDB), (3) *mysqlsam*, que usa um banco de dados MySQL, e (4) *ldapsam* que utiliza um diretório LDAP[3] [2] para armazenar as informações do domínio, sendo que os dois últimos são os mais utilizados. Apesar de não requerer nenhuma configuração especial, o *tdbsam* não é recomendado para implementações que evolvam a criação de mais de 250 objetos ou que dependa de replicação [3]. O *ldapsam* não possui esta limitação e ainda garante à solução maior escalabilidade e flexibilidade, motivo que levou o CEFET-MG a adotá-lo.

3. Administração de contas e grupos de usuários

Quando da implantação dos controladores de domínio Samba, o CEFET-MG já dispunha de um diretório LDAP que continha as contas dos usuários do serviço de correio eletrônico. Estas contas foram mantidas e acrescidas dos atributos relacionados ao Samba. Já era utilizado também o *software* GOsa[4], um poderoso sistema de gerenciamento de serviços baseados em LDAP, como Samba, e-mail, contas e grupos POSIX[5], Asterisk, etc [4] .

---

1 *Server Message Block*, o nome original do protocolo "falado" pelo Samba, desenvolvido na década de 80 pela IBM e adotado pela Microsoft.
2 *Common Internet File System*, o novo nome para o SMB, desde a década de 90.
3 *Lightweight Directory Access Protocol*, protocolo de acesso a diretórios de propósito geral, baseados no padrão X.500
4 *Gonicus System Administrator*, *software* de gerenciamento desenvolvido em PHP pela Gonicus Labs.
5 *Portable Operating System Interface*, conjunto de normas definidas pelo IEEE com o objetivo garantir a portabilidade do código-fonte de um programa entre sistemas operacionais que atendam as normas POSIX.

O GOsa passou então a ser utilizado como o sistema oficial de administração de contas de usuários e grupos dos domínios. Como ainda não existia nenhum trabalho de tradução para o Português, foi necessário traduzir ao menos as partes mais utilizadas no nosso caso particular. Tal esforço resultou na tradução de aproximadamente 30% de toda a *interface*, e este trabalho já está compartilhado com a comunidade, devendo inclusive estar disponível nas próximas versões oficiais do GOsa[6].

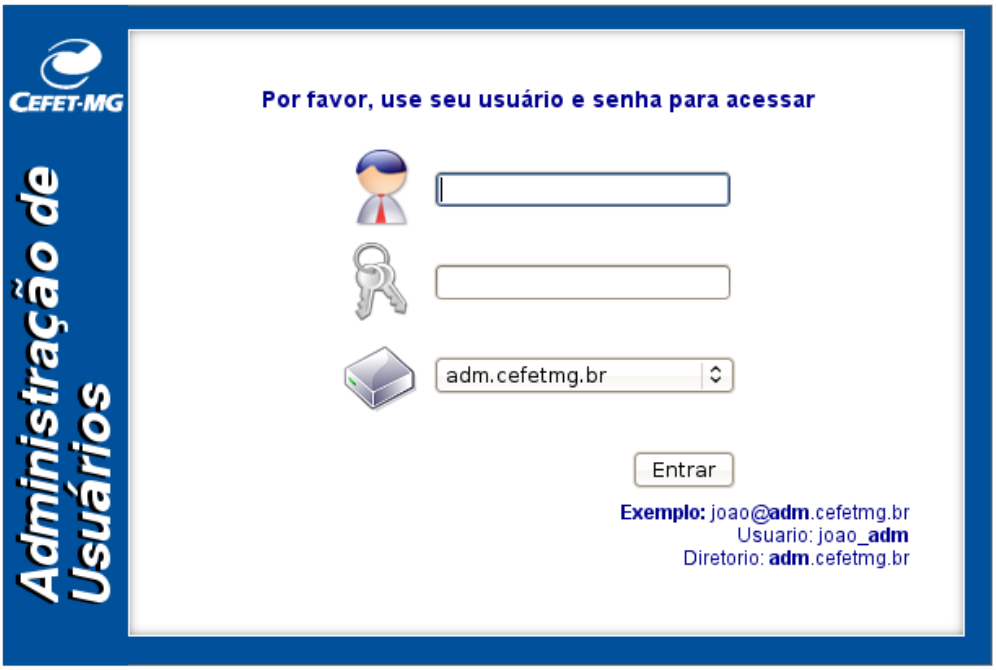

Fig. 1 – Detalhe da tela de autenticação do GOsa, já com as personalizações feitas pelo CEFET-MG

Além da tradução e de algumas modificações na aparência, foi necessário integrar ao GOsa alguns *shell scripts* que pudessem interagir com o sistema, executando tarefas como criação e remoção de diretórios pessoais *(home folder)* do usuário, bem como definição da quota de disco que este poderá utilizar. Visto que os papéis dentro da solução proposta são desempenhados por vários servidores, foi necessário dotar estes *scripts* da capacidade de disparar uma tarefa em outro servidor, diferente do que recebeu o comando. Para tanto, foram definidas chaves públicas para que a conexão SSH[7] pudesse ser efetuada sem a interação do usuário. Também foram criados usuários específicos com privilégios limitados para executar somente as tarefas relacionadas ao *script* em questão, sem qualquer prejuízo para a segurança. Assim, ao concluir a criação de uma conta POSIX, por exemplo, o GOsa invoca localmente um *script* que abre uma conexão SSH com o servidor de arquivos e dispara neste o processo de criação do diretório pessoal do usuário.

6 *Ticket* n° 611 aberto no *site* do projeto. Disponível em: https://oss.gonicus.de/labs/gosa/ticket/611. Acesso em: 18 mar 2009

7 *Secure Shell*, protocolo de rede que permite a conexão entre dois computadores em rede usando um canal seguro.

4. Compartilhamento de arquivos e impressoras

Uma questão problemática no CEFET-MG era a falta de controle no uso de diretórios compartilhados. Era comum quando um usuário, no intuito de facilitar o acesso dos colegas aos seus arquivos de trabalho, compartilhava a sua unidade *C:* sem nenhum tipo de restrição. Isto gerava diversos transtornos como disseminação de vírus e a perda de documentos importantes em virtude de não existir nenhuma política de *backup* ou segurança associada a estes compartilhamentos.

Com a aquisição de unidades de armazenamento externo (*storages*) de grande capacidade, foi possível contornar este problema com grandes vantagens. O *storage*, fisicamente conectado a um servidor da rede executando GNU/Linux, fica responsável por armazenar os diretórios pessoais dos usuários, bem como as pastas institucionais, que são compartilhadas com os usuários do mesmo departamento ou setor. Este servidor exporta via NFS[8] para cada controlador de domínio uma árvore de diretórios correspondente ao seu domínio. Quando um usuário efetua autenticação no domínio a partir de uma estação Windows, o controlador Samba enviará para o cliente um *script* de *logon*, responsável por mapear uma unidade de rede (*U:*) para o diretório pessoal do usuário e outra (*Z:*) para a pasta institucional. Este *script* também realiza outras tarefas como sincronizar o relógio da estação com o controlador e definir variáveis de ambiente relacionadas ao um *software* específico utilizado por algum setor. Convencionou-se que o grupo primário (atributo da conta POSIX) de cada usuário seria o grupo POSIX associado ao seu setor de trabalho. Assim, o Samba foi configurado para executar, a cada *logon*, um *script* com o mesmo nome do grupo primário. Por exemplo, os usuários que trabalham na biblioteca têm como grupo primário o grupo **biblioteca**, e executam, ao efetuar *logon*, o *script* **biblioteca.bat** que mapeia em *Z:* o compartilhamento **\\srv-dominio\biblioteca**.

O controle de acesso aos compartilhamentos é feito em dois níveis: sistema de arquivos e configurações específicas do Samba. O controle pelo sistema de arquivos é feito integrando-se à base de usuários e grupos locais os usuários e grupos do LDAP, o que pode ser feito por meio da instalação da biblioteca *libnsswitch* e da configuração do arquivo de seleção de serviços de rede (/etc/nsswitch.conf). A partir daí, o sistema de arquivos se torna capaz de reconhecer *uids* e *gids* das contas armazenadas no LDAP e gerenciar corretamente as permissões. Do ponto de vista do Samba, a definição do compartilhamento no arquivo de configuração também especifica quais usuários e grupos terão acesso ao recurso.

O compartilhamento de impressoras é possível para qualquer impressora compatível com CUPS[9] instalada no controlador de domínio. A definição do compartilhamento é feita no arquivo de

8 *Network File System*, protocolo de sistema de arquivos de rede desenvolvido pela Sun Microsystems em 1984.
9 *Common UNIX Printing System*, sistema de impressão de código aberto desenvolvido pela Apple Inc.

configuração do Samba, bem como a especificação das permissões de impressão. Também é possível armazenar versões de *drivers* para diversas plataformas Windows, possibilitando a instalação automática da impressora na estação, com o recurso "Adicionar nova impressora".

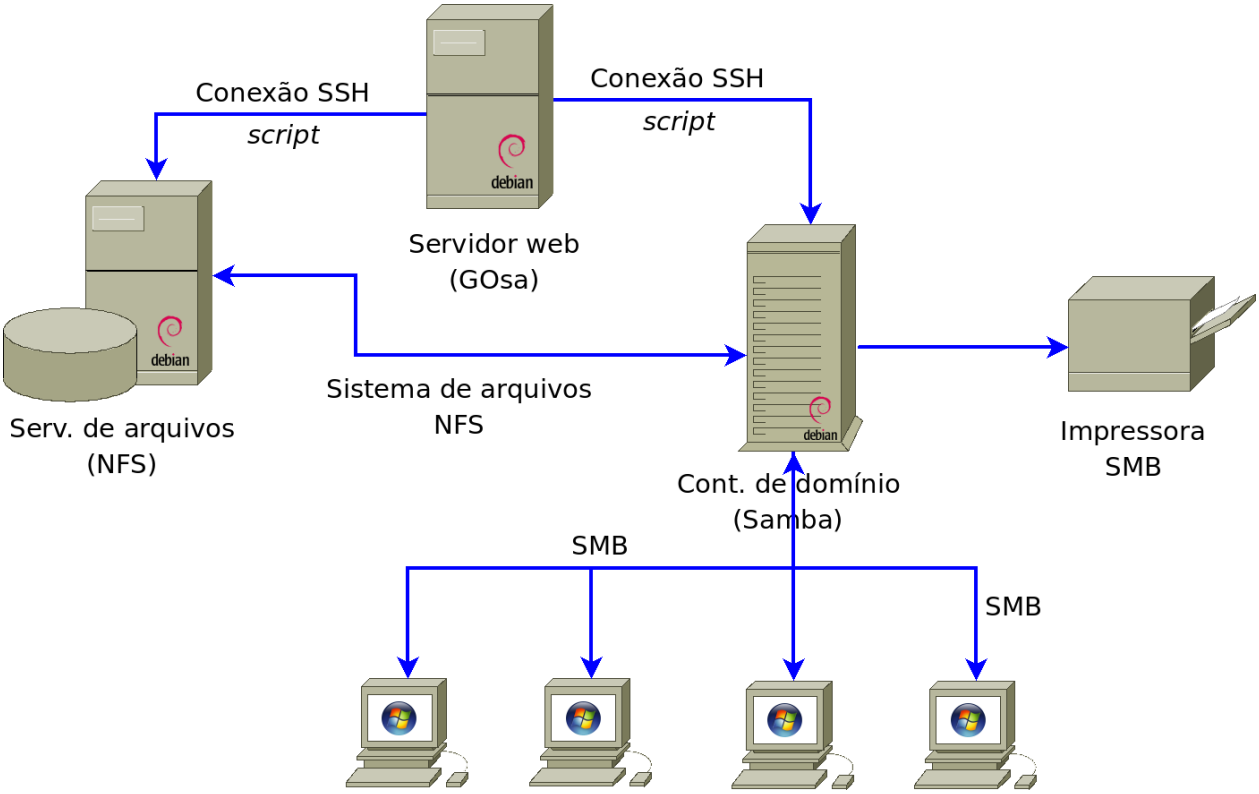

Fig. 2 – Diagrama esquemático com a interação entre os servidores, estações e impressoras

O sistema de arquivos escolhido para os volumes que seriam usados com o Samba foi o XFS[10], por apresentar características adequadas ao projeto. destacam-se aqui duas das principais: a capacidade de expansão *on-line* do tamanho do volume e o sistema de quotas de disco baseadas em projetos [5]. Aqui, um projeto estabelece uma quota associada a uma árvore de diretórios. Ao gravar um arquivo sob esta árvore, um usuário não terá sua quota pessoal decrementada, mas a quota do projeto. Esta funcionalidade foi de extrema importância para que a implantação das pastas departamentais compartilhadas pudesse ser efetivada.

Todos os compartilhamentos disponibilizados pelo controlador são protegidos contra vírus pelo ClamAV[11] [6]. Para realizar a verificação no momento do acesso ao arquivo, foi necessário utilizar o módulo *vscan-clamav*. Esta biblioteca compartilhada (*shared object*) é desenvolvida e mantida pelo projeto OpenAnti-virus [7] e foi criada para utilizar o recurso de sistema de arquivos virtual (VFS) do Samba [8]. Ao tentar acessar um arquivo, o *vscan-clamav* invoca o ClamAV que realiza a verificação. Caso o arquivo esteja infectado, o usuário será alertado e impedido de prosseguir. Existe também, dentro de uma sessão, um histórico de acesso e modificação para cada

10 Sistema de arquivos de 64 bits desenvolvido pela SGI.
11 Anti-vírus de código aberto, desenvolvido para sistemas UNIX/Linux.

arquivo, o que evita a verificação desnecessária e a conseqüente perda de desempenho. A definição das permissões de impressão é feita apenas na definição do compartilhamento no arquivo de configuração do Samba.

5. Resultados

A implantação da solução exposta trouxe diversos benefícios, tanto para os usuários quanto para os administradores da rede. Atualmente o CEFET-MG possui quatro controladores de domínio em funcionamento, sendo que três deles possuem entre si relações de confiança. Isto permite a um usuário de qualquer um desses três domínios se autenticar em qualquer estação que esteja afiliada a um domínio participante da rede de confiança. Testes com relação de confiança entre domínios Samba e Windows NT também já foram realizados com sucesso.

Dentre os benefícios, destacamos os seguintes:

- solução imediata dos problemas de indisponibilidade de arquivos e recursos devido aos "vírus de compartilhamento";
- aumento da qualidade do atendimento ao usuário, em virtude da administração centralizada;
- melhora significativa do nível de segurança, com a política de *backup* centralizado para todas as pastas pessoais e institucionais;
- possibilidade de auditoria de acesso aos arquivos e pastas disponíveis para mais de um usuário;
- possibilidade de controle de impressão baseado em quotas, uma vez que o serviço é centralizado, e
- no caso específico do CEFET-MG, a utilização de uma senha única para acessar os diversos serviços da instituição, decorrente do "aproveitamento" das contas de *e-mail* pré-existentes.

Nota-se que os objetivos anteriormente definidos foram alcançados, o que denota a eficiência da solução.

6. Conclusão

As dificuldades e desafios encontrados na implantação de uma solução baseada em ferramentas de código aberto acabam por gerar mais contribuições para os *softwares* e sistemas envolvidos, alimentando os seus ciclos de evolução e ampliando o conhecimento disponível aos interessados naquela área específica. São, portanto, maiores as chances de sucesso para aqueles que decidirem adotar a solução. Como exemplo concreto desta afirmação, as modificações realizadas no

código fonte do *vscan-clamav* por técnicos do DRI. Para torná-lo compatível com a versão 3.2.5 do Samba, disponível no Debian 5, foi necessário analisar o código e rastrear as mensagens de erro por centenas de vezes até que a compilação fosse realizada com sucesso. O nosso êxito, no entanto, dependeu também dos resultados obtidos anteriormente por outros desenvolvedores [9].

Uma solução como a apresentada demonstra que é possível substituir as soluções comerciais por outras totalmente baseadas em *software* livre de igual ou melhor qualidade, sem que haja redução da qualidade do serviço ou aumento de custos.

Referências

[1] VERNOOJ, Jelmer R.; TERPSTRA, John H.; CARTER, Gerald. **The Official Samba 3.2.x HOWTO and Reference Guide**. 2008. Disponível em http://us1.samba.org/samba/docs/man /Samba-HOWTO-Collection/IntroSMB.html#id2543643. Acesso em: 18 mar 2009.

[2] **OpenLDAP**. Disponível em http://www.openldap.org. Acesso em: 20 mar 2009

[3] VERNOOJ, Jelmer R.; TERPSTRA, John H.; CARTER, Gerald. **The Official Samba 3.2.x HOWTO and Reference Guide**. 2008. Disponível em http://us1.samba.org/samba/docs/man/ Samba-HOWTO-Collection/passdb.html#id2585525. Acesso em: 18 mar 2009.

[4] **GOsa**. Disponível em: https://oss.gonicus.de/labs/gosa. Acesso em: 23 mar 2009.

[5] **SGI**. http://oss.sgi.com/projects/xfs. Acesso em: 20 mar 2009.

[6] KOJM, Tomasz. **Clam AntiVirus 0.95 User Manual**. Sourcefire, Inc. 2009. Disponível em http://www.openantivirus.org/index.php. Acesso em: 20 mar 2009.

[7] **OpenAntiVirus Project**. http://www.openantivirus.org/index.php. Acesso em: 20 mar. 2009.

[8] VERNOOJ, Jelmer R.; TERPSTRA, John H.; CARTER, Gerald. **The Official Samba 3.2.x HOWTO and Reference Guide**. 2008. Disponível em http://us1.samba.org/samba/docs/man/ Samba-HOWTO-Collection/VFS.html. Acesso em: 19 mar 2009.

[9] **SourceForge.net**. Disponível em: http://sourceforge.net/tracker/?func=detail&atid=310590& aid=2521012&group_id=10590. Acesso em: 23 mar 2009.