

Daniel Darlen Corrêa Ribeiro

**Software Livre na Administração Pública.
Estudo de caso sobre adoção do SAMBA na Auditoria Geral
do Estado de Minas Gerais**

Monografia apresentada ao curso de Administração em Redes Linux (ARL) da Universidade Federal de Lavras como parte das exigências da disciplina Monografia para obtenção do título de Especialista em Administração em Redes Linux.

Orientador
Prof. Msc. Joaquim Quinteiro Uchôa

Lavras
Minas Gerais - Brasil
2004

Daniel Darlen Corrêa Ribeiro

**Software Livre na Administração Pública.
Estudo de caso sobre adoção do SAMBA na Auditoria Geral
do Estado de Minas Gerais**

Monografia apresentada ao curso de Administração em Redes Linux (ARL) da Universidade Federal de Lavras como parte das exigências da disciplina Monografia para obtenção do título de Especialista em Administração em Redes Linux.

Aprovada em *29 de fevereiro de 2004*

Prof. Msc. Fernando Cortez Sica

Profª. Msc. Kátia Cilene Amaral Uchôa

Prof. Msc. Joaquim Quinteiro Uchôa
(Orientador)

Lavras
Minas Gerais - Brasil

Sumário

1	Introdução	1
2	Contextualização e problematização	3
2.1	Auditoria Geral do Estado	3
2.1.1	Estrutura organizacional	3
2.1.2	Foco da análise	5
2.2	Contextualização do ambiente	6
2.2.1	Aspectos operacionais	6
2.2.2	Aspectos legais – princípios constitucionais	7
2.3	Segurança das informações	9
2.4	Definição do problema	9
2.5	Considerações finais	10
3	Software Livre	13
3.1	Conceitos fundamentais	13
3.2	A filosofia do <i>Software</i> Livre	14
3.2.1	Free Software Foundation e o projeto GNU	14
3.2.2	O princípio básico da colaboração	17
3.3	Licenças de <i>Software</i>	19
3.3.1	A licença GPL	20
3.4	<i>Software</i> Livre em ambientes corporativos	21
3.5	<i>Software</i> Livre na Administração Pública	23
3.6	Considerações finais	25
4	SAMBA	27
4.1	Histórico e características	29
4.2	Instalação	30
4.3	Configuração	31
4.3.1	Edição do arquivo <code>smb.conf</code>	31
4.3.2	Configuração via <i>SWAT</i>	34
4.4	Considerações finais	36

5	Análise evolutiva	37
5.1	Considerações iniciais	37
5.1.1	O ambiente	37
5.1.2	Riscos imediatos	38
5.1.3	O ponto de partida	39
5.2	A era da incerteza	39
5.2.1	Proposta e implementação	39
5.2.2	Resultados	40
5.2.3	Comentários	41
5.3	A era do compartilhamento	42
5.3.1	Proposta e implementação	42
5.3.2	Resultados	43
5.3.3	Comentários	44
5.4	A era do controle	44
5.4.1	Proposta e implementação	45
5.4.2	Resultados	48
5.4.3	Comentários	48
5.5	Considerações finais	49
5.5.1	Análise dos resultados	49
5.5.2	Novas perspectivas	50
6	Conclusão	53
A	Lei Delegada 92, de 29/01/2003	63
B	Tradução da Licença GPL para português	71
B.1	Introdução	72
B.2	Licença Pública Geral GNU	73
C	Rotinas de monitoramento do SAMBA	81
C.1	Variáveis utilizadas nas rotinas	82
C.2	Rotina <i>smbcontrol</i>	82
C.3	Rotina <i>smbmaqlog</i>	84
D	Distribuições GNU/Linux disponíveis no mercado.	85

Lista de Figuras

2.1	Composição da Auditoria Geral.	4
2.2	Organograma da Auditoria Geral.	5
4.1	<i>Web Site</i> do SAMBA	27
4.2	Compilação e instalação do SAMBA	30
4.3	Ajustes na instalação do SAMBA	31
4.4	Linhas a descomentar no arquivo <i>/etc/services</i>	31
4.5	Arquivo <i>smb.conf</i> – seção global.	33
4.6	Arquivo <i>smb.conf</i> – compartilhamentos.	34
4.7	Tela de <i>login</i> da interface <i>SWAT</i>	36
4.8	Administração do SAMBA via <i>SWAT</i>	36
5.1	Equipes 4 e 6 utilizadas como servidores de arquivos.	40
5.2	SAMBA utilizando diretiva <i>security = share</i>	43
5.3	Diretivas para utilização do SAMBA como PDC.	45
5.4	Definindo domínio nas estações Windows.	46
5.5	Exemplo de um compartilhamento para trabalho de auditoria.	46
5.6	<i>Script</i> de inicialização de um usuário.	47
5.7	Compartilhamentos mapeados em clientes Windows.	47
5.8	Tela de <i>login</i> das estações Windows.	48
C.1	<i>Scripts</i> executados para os trabalhos de auditoria.	81
C.2	<i>Script</i> executado no compartilhamento <i>netlogon</i>	81
C.3	Rotina <i>smbcontrol</i>	83
C.4	Trecho do <i>log</i> gerado pelo <i>script smbcontrol</i> para usuários.	83
C.5	<i>Log</i> gerado pelo <i>script smbcontrol</i> para compartilhamentos.	83
C.6	Rotina <i>smbmaqlog</i>	84
C.7	Trecho de <i>log</i> gerado pelo <i>script smbmaqlog</i>	84

Lista de Tabelas

3.1	Comparação entre os modelos Cathedral e Bazar.	18
4.1	Seções do arquivo <code>smb.conf</code>	32
4.2	Opções de compartilhamento do SAMBA.	35
C.1	Variáveis utilizadas nas rotinas da SCAO.	82

*Aos homens e mulheres de boa-vontade,
que acreditam na liberdade de expressão
e na difusão de suas idéias.
A luta continua!*

Agradecimentos

A Deus, pela graça a todos concedida;
aos meus pais, Rachel e Roberto,
que sempre se esforçaram para que eu estudasse;
à minha esposa Silânia,
pelo tempo de convivência furtado e pelo seu apoio incondicional;
ao Prof. Msc. Joaquim pela incentivadora orientação;
aos amigos que sentiram falta das conversas e das violas;
a todos que colaboraram com livros, referências,
debates, críticas e sugestões, em especial:
José Márcio, Vanessa, Renilda, Fabrício, Marco Patrício, Célia,
Carlos Antão, Henrique, Inês, Fernando Sette,
Alberto Viotti, Charles, Gizelda,
Cláudia Corrêa, Dieimes Donato, Eduardo Lúcio,
Samuel, Kátia Cilene, Domingues, Marconi.
O meu muito obrigado!

Resumo

Mais do que simples redução de custos com licenças de utilização, a adoção de *Software* Livre possibilita desenvolvimento tecnológico, diferenciação de produtos ou serviços, independência de fornecedor único e maior segurança na execução de processos. Outro fator relevante, refere-se à socialização do conhecimento, favorecido pela publicação e distribuição de códigos fonte dos programas. Essas vantagens aplicadas à Administração Pública servem para promover aprimoramento das ações de governo com conseqüente melhoria no atendimento às demandas sociais. Baseado nessa visão, este trabalho propõe-se a realizar uma análise evolutiva dos mecanismos de autenticação e controle de acesso da rede de computadores da Superintendência Central de Auditoria Operacional, por meio da utilização da ferramenta SAMBA, na intenção de contribuir com um exemplo elucidativo sobre os benefícios alcançados com a adoção de *Software* Livre.

Capítulo 1

Introdução

A utilização de ferramentas de Tecnologia da Informação permitem, cada vez mais, que as instituições desenvolvam mecanismos de controle de acesso e armazenamento de dados, visando dotar de segurança o ambiente computacional. Segundo [McGEE; PRUSAK (1995)], essa preocupação justifica-se pelo caráter estratégico que a informação¹ adquire no âmbito corporativo, tornando-se um ativo que precisa ser administrado de forma especial².

De maneira geral, o Estado encontra-se no rol das instituições que precisam garantir fidedignidade, controles de acesso e registros de alterações de seus dados armazenados. Esse cuidado deve, obrigatoriamente, transcender as estruturas fechadas dos bancos de dados, que já possuem mecanismos naturais de controle, e alcançar as mais elementares situações de criação e manipulação de dados: os arquivos de trabalho³ dos usuários.

Existem alternativas para autenticação de usuários em redes corporativas e controle de acesso a arquivos/diretórios que permitem desenvolver mecanismos de proteção. Dentre as soluções disponíveis no mercado, merecem destaque aquelas desenvolvidas sob licença GPL⁴, caracterizadas *Software Livre*⁵ que permitem economia de preço, utilização plena, liberdade para instalação, execução e modificação, entre outros aspectos.

¹ [STAIR (1998)] define *informação* como um conjunto de fatos (dados) organizados de tal forma que adquirem valor adicional além do fato (dado) em si.

² [DAVENPORT (2003)] e [NONAKA; TAKEUCHI (1997)] realizam estudo detalhado sobre gerenciamento estratégico da informação, juntamente com as já citadas referências [McGEE; PRUSAK (1995)] e [STAIR (1998)].

³ No contexto apresentado, o termo *arquivos de trabalho* refere-se a relatórios, planilhas eletrônicas, apresentações multimídia, arquivos em texto puro, imagens digitalizadas, etc., onde o usuário possui autonomia para criação, edição ou alteração segundo sua vontade.

⁴ *General Public License* ou *Licença Pública Geral*. O Apêndice B apresenta a tradução da GPL para o português.

⁵ *Software livre* se refere à liberdade dos usuários executarem, copiarem, distribuírem, estudarem, modificarem e aperfeiçoarem o *software* [FSF (2003)]. Esse conceito será tratado no Capítulo 3.

Entre as soluções *Software* Livre existentes destaca-se a ferramenta SAMBA⁶, que permite o estabelecimento de autenticação de usuários e controle de acesso a arquivos, além de integrar vários sistemas operacionais e suas variantes.

O estudo cuidadoso da utilização do SAMBA, aliado à política informacional das instituições, poderá apontar mecanismos eficientes de controle e disseminação de informações para a Administração Pública.

Nesse contexto, o presente trabalho objetiva realizar avaliação sobre adoção de *Software* Livre na Administração Pública, por meio de um estudo de caso de utilização do SAMBA como solução para autenticação de usuários e controle de acesso aos arquivos e relatórios da Auditoria Geral do Estado de Minas Gerais (AUGE), referentes à Superintendência Central de Auditoria de Gestão (SCAG) e Superintendência Central de Auditoria Operacional (SCAO).

A presente monografia encontra-se organizada da seguinte forma: o Capítulo 2 trata da contextualização e problematização motivadora deste trabalho. Apresenta ainda a estrutura organizacional da Auditoria Geral do Estado, seus aspectos operacionais e legais que delimitam o escopo da segurança das informações armazenadas, definindo o problema motivador deste trabalho. O Capítulo 3 aborda a conceituação do *Software* Livre, sua forma colaborativa de produção, e realiza uma breve discussão sobre as licenças de *software* GPL e EULA. É apresentada uma análise do *Software* Livre em ambientes corporativos, em especial, sua utilização no setor público. No Capítulo 4, é descrita a ferramenta de autenticação de usuários e compartilhamento de arquivos e impressoras, SAMBA, cuja aplicação e resultados obtidos servirão para as conclusões do estudo de caso. Aliado a isso, é apresentada uma descrição da ferramenta, sua instalação e configuração. O Capítulo 5 trata da análise evolutiva, considerando a rede de computadores da SCAO, a adoção do SAMBA e os resultados obtidos. Por fim, no Capítulo 6, são apresentadas as conclusões deste trabalho.

⁶ Página oficial <http://www.samba.org>.

Capítulo 2

Contextualização e problematização

2.1 Auditoria Geral do Estado

2.1.1 Estrutura organizacional

Conforme [MINAS GERAIS (2003)]¹, a Auditoria Geral do Estado é o órgão responsável pelo Controle Interno no Estado de Minas Gerais, sendo composta por:

- Gabinete;
- Assessoria Técnica;
- Superintendência Central de Auditoria de Gestão;
- Superintendência Central de Auditoria Operacional;
- Superintendência Central de Correição Administrativa;
- Superintendência de Planejamento, Gestão e Finanças.

Essa estrutura orgânica é bastante recente. Até a promulgação da Lei Delegada 92 [MINAS GERAIS (2003)], as atividades de Controle Interno e Correição Administrativa encontravam-se assim distribuídas entre as entidades do Poder Executivo estadual:

- a Auditoria Geral do Estado contemplava exclusivamente a figura do seu dirigente máximo, o Auditor-Geral do Estado. Para desempenho de suas funções contava com apoio técnico e administrativo da então Secretaria de Estado da Casa Civil [MINAS GERAIS (1985)], [MINAS GERAIS (1995)];

¹ O anexo A apresenta o texto original da Lei Delegada 92.

- a Superintendência Central de Correição Administrativa encontrava-se vinculada à então Secretaria de Estado de Recursos Humanos e Administração [MINAS GERAIS (1996)];
- a Superintendência Central de Auditoria Operacional encontrava-se vinculada à Secretaria de Estado da Fazenda [MINAS GERAIS (1998)].

Dessa forma, as funções de Auditoria, ou Controle Interno, eram efetivamente desempenhadas pela Superintendência Central de Auditoria Operacional; os processos administrativos, envolvendo servidores públicos, eram de responsabilidade da Superintendência Central de Correição Administrativa; enquanto a Auditoria Geral mantinha a função de assessoria direta ao Governador do Estado.

Para a formação da estrutura atual da Auditoria Geral do Estado, que será tratada neste trabalho por AUGÉ, foram incorporados os acervos técnicos (incluindo Recursos Humanos) e patrimoniais oriundos da Superintendência Central de Auditoria Operacional e da Superintendência Central de Correição Administrativa, àqueles já disponibilizados pela Secretaria de Estado da Casa Civil. A Figura 2.1 apresenta a formação da AUGÉ após a publicação da Lei Delegada 92, em janeiro de 2003.

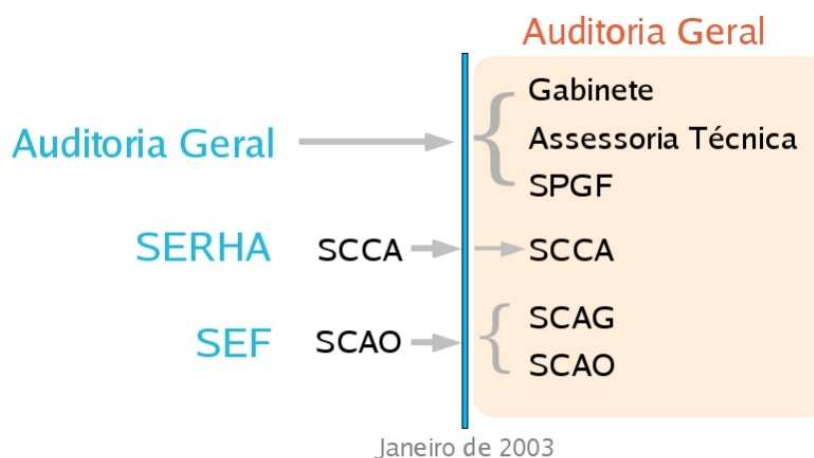


Figura 2.1: Composição da Auditoria Geral.

Cabe ressaltar, que nessa época, a então Superintendência Central de Auditoria Operacional (SCAO), foi desmembrada em duas superintendências centrais: uma manteve o nome de origem e a outra passou a se chamar Superintendência Central de Auditoria de Gestão (SCAG). O autor desta monografia fazia parte da equipe de trabalho da SCAO e passou a ter sua lotação na SCAG com a promulgação da Lei Delegada 92 [MINAS GERAIS (2003)].

A Figura 2.2 apresenta o organograma da AUGE em janeiro de 2003.

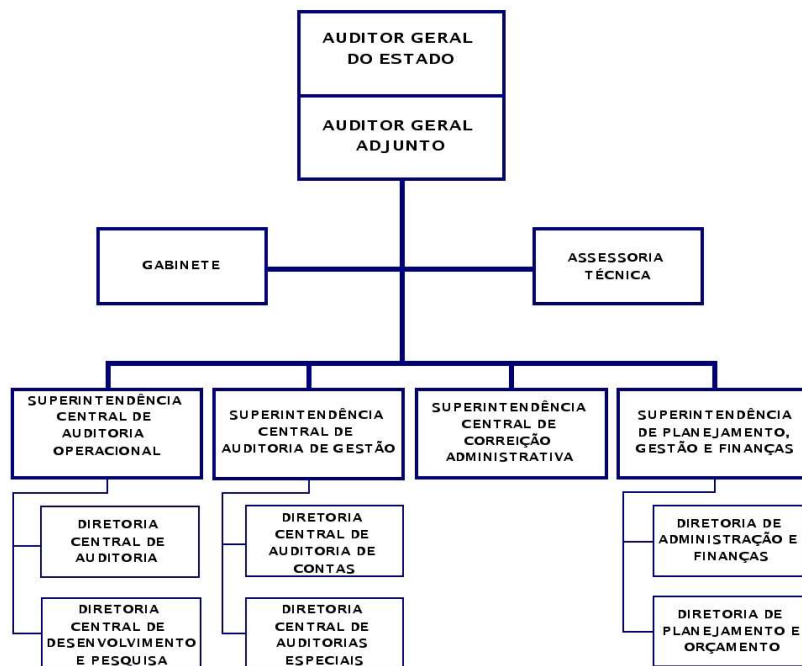


Figura 2.2: Organograma da Auditoria Geral.

2.1.2 Foco da análise

Conforme apresentado, a estrutura atual da AUGE advém da incorporação de duas superintendências centrais e de seus eventuais desmembramentos.

Enquanto a Auditoria Geral dependia do apoio da Secretaria da Casa Civil, seu parque computacional estava restrito a um servidor de arquivos utilizando Windows NT e poucas estações conectadas²; a Superintendência Central de Correição Administrativa não operava em rede e; a Superintendência Central de Auditoria Operacional contava com um servidor de arquivos rodando SAMBA para, aproximadamente, 15 estações de trabalho³.

O foco de análise desta monografia será a rede de computadores da antiga SCAO⁴, que adotou o SAMBA como solução para o servidor de arquivos em 1999.

²Não foi possível precisar o número exato. Um valor provável seria 6 estações.

³Esse era o número de estações em utilização quando da incorporação da SCAO à AUGE. Houve períodos em que o número de estações era superior a 20.

⁴Atualmente essa rede de computadores atende às Superintendências Centrais de Auditoria de Gestão (SCAG) e Operacional (SCAO).

2.2 Contextualização do ambiente

2.2.1 Aspectos operacionais

Desde 1998, a SCAO necessitava desenvolver ou adquirir uma solução para controlar a criação e manipulação de arquivos em meio magnético. Uma vez que os computadores operavam em rede, a idéia de se utilizar um servidor para autenticação de usuários e armazenamento de arquivos vinha ao encontro das necessidades da superintendência, principalmente, com a chegada de uma máquina capaz de executar essa função, em dezembro daquele ano. Esse equipamento⁵ utilizava o sistema operacional Solaris⁶, versão 2.5. As demais máquinas, que seriam utilizadas como “estações de trabalho”, utilizavam plataforma Windows⁷.

Não foi cogitada a mudança dos sistemas operacionais das estações por envolver treinamentos para os funcionários, os quais operavam com tranquilidade o sistema adotado. Outro complicador, refere-se ao *custo psicológico* associado a uma proposta de inovação:

Ainda que este custo não seja facilmente contabilizado, é interessante observar que o custo psicológico pode ser uma das maiores barreiras à mudança. Por maiores que sejam os problemas causados por um sistema, estes podem ser minimizados inconscientemente pelo custo psicológico de uma eventual mudança. [...] O medo pode surgir como resposta, caso haja alguma percepção de que a novidade possa não ser tão boa quanto parece, o que é estimulado por campanhas de medo, incerteza e dúvidas... [FERRAZ (2002)]

Um terceiro complicador para novos direcionamentos deve ser destacado. Em função do seu caráter legalista (que será discutido na Subseção 2.2.2), a Administração Pública não possui característica pioneira frente ao mercado, permanecendo à reboque das inovações tecnológicas e metodologias aplicadas no setor privado⁸. Uma proposta de inovação, geralmente, só é aceita no setor público quando exemplos de sucesso são registrados por outras corporações. Antes de mensurar o custo psicológico de uma mudança no setor público, torna-se necessário que tal proposta de mudança tenha ocorrido, normalmente no setor privado, e apresentados resultados satisfatórios, amplamente divulgados, para ingressar na “ordem do dia”.

⁵ Essa máquina, que ainda pertence ao patrimônio da AUGÉ, consiste em um servidor *Sun Spark Ultra 450*, com capacidade de armazenamento da ordem de 25 Gigabytes (expansível), 2 processadores de 250MHz que perfazem uma frequência final de operação da ordem de 450MHz.

⁶ O Solaris é um sistema operacional, variante do Unix, distribuído pela Sun Microsystems. Página oficial: <http://www.sun.com/software/solaris>.

⁷ Sistema operacional Windows 95 ou Windows 98.

⁸ Com excessão das instituições de pesquisa que são estruturadas para aprimorar o conhecimento científico e promover inovação.

Além do contexto apresentado, a realidade orçamentária da superintendência e as restrições financeiras impediam a aquisição de soluções proprietárias (pagas), como a adoção de clientes NFS para Windows.

Cabe ressaltar que a SCAO não contava com equipe de profissionais na área de Tecnologia da Informação: seu corpo funcional era composto por auditores, inspetores e auxiliares administrativos. A Secretaria de Estado da Fazenda, à qual a superintendência estava vinculada, direcionava a atenção de sua equipe de desenvolvimento de *software* para as questões relativas à arrecadação e controle financeiro estadual, não atuando em outras áreas.

2.2.2 Aspectos legais – princípios constitucionais

Por fazer parte do Poder Executivo Estadual, a SCAO, assim como toda Administração Pública⁹, sujeita-se às peculiaridades relativas aos Órgãos do Estado determinados por um extenso conjunto de instrumentos legais. Dessa forma, qualquer tomada de decisão precisa, necessariamente, balizar-se pelas normas e regimentos vigentes, em especial, aqueles constantes na Magna Carta, na Constituição Estadual, para o caso Estados, e na Lei Orgânica, para o caso dos municípios.

A Constituição da República Federativa do Brasil prevê¹⁰ que a “administração pública direta, indireta ou fundacional, de qualquer dos Poderes da União, dos Estados, do Distrito Federal e dos Municípios obedecerá aos princípios¹¹ de legalidade, impessoalidade, moralidade, publicidade e eficiência [BRASIL (1988)]. De forma complementar, a Constituição do Estado de Minas Gerais incorpora o princípio da razoabilidade¹² [MINAS GERAIS (1989)].

Nesse contexto, a Administração Pública do Estado de Minas Gerais tem como elementos norteadores das ações de governo, os seis princípios constitucionais destacados:

Legalidade: Exige que toda e qualquer atividade administrativa deve ser autorizada por lei, diferentemente do tratamento dispensado à administração particular:

Na Administração Pública não há liberdade nem vontade pessoal. Enquanto na administração particular é lícito fazer tudo que a lei não proíbe, na Administração Pública só é permitido fazer o que a lei autoriza. A lei para o particular significa “pode fazer assim”; para o administrador público significa “deve fazer assim” [MEIRELLES (1996)].

⁹ “A Administração Pública é o principal meio através do qual o Estado procura a realização dos objetivos de uma sociedade” [MOTTA (1999)].

¹⁰ Art. 37 da Constituição da República Federativa do Brasil.

¹¹ “Princípios Administrativos são os postulados fundamentais que inspiram todo o modo de agir da Administração Pública” [CARVALHO FILHO (2003)].

¹² Art. 13 da Constituição do Estado de Minas Gerais.

Como consequência desse princípio, “a Administração Pública não pode, por simples ato administrativo, conceder direitos de qualquer espécie, criar obrigações ou impor vedações aos administrados; para tanto, ela depende de lei” [DI PIETRO (2001)]

Impessoalidade: Princípio que visa a igualdade de tratamento dispensada aos administrados, impedindo favorecimento de alguns indivíduos em detrimento de outros que se encontrem em mesma situação jurídica. Dessa forma, a Administração deve voltar-se, exclusivamente, para o interesse público, e não o privado.

Esse princípio também deve ser aplicado para excluir a promoção pessoal de autoridades ou servidores públicos sobre suas realizações administrativas [MEIRELLES (1996)].

Moralidade: Exige observância de preceitos éticos na conduta do administrador público. Reflete preocupação com a possibilidade de que maus administradores pratiquem ações na busca de seus próprios interesses, ou de terceiros.

Dessa forma, a atuação do administrador deve seguir padrões éticos de probidade, decoro e boa-fé.

Publicidade: Os atos da Administração devem merecer a mais ampla divulgação possível. Esse princípio busca assegurar os efeitos das decisões tomadas e propiciar seu conhecimento e controle por parte dos interessados, bem como orientar a conduta interna dos administradores envolvidos.

Eficiência: Sinaliza que os serviços prestados devem ser da mais alta qualidade, buscando atingir os melhores resultados.

Além da qualidade do serviço prestado, esse princípio também pode ser relacionado à forma pela qual os meios são geridos: procurando maximizar resultados e minimizar custos, com a utilização racional de métodos, técnicas e normas.

Razoabilidade: Prevê que as tomadas de decisões devem contribuir efetivamente para satisfatório atendimento dos interesses públicos, buscando coerência lógica nas medidas administrativas.

Uma argumentação para esse princípio diz que a lei, por si só, não consegue estabelecer uma conduta específica para todos os atos da Administração. Assim, o administrador, diante da diversidade de ações enfrentadas, deve analisar cada caso e encontrar a melhor solução, agindo como a sociedade esperaria que agisse um homem ético.

O princípio da razoabilidade não encontra-se literalmente expresso na Constituição da República, tendo sua fundamentação implícita. Em algumas

Constituições Estaduais, entretanto, já possui referência direta, a exemplo de Minas Gerais [MOTTA (1999)].

Os princípios constitucionais são as diretrizes norteadoras, “aplicáveis em qualquer situação com fins de coordenar, direcionar, condicionar ou integrar a criação, interpretação ou aplicação das demais normas” [MOTTA (1999)]. Dessa maneira, as leis, decretos, portarias, resoluções afetas à Administração Pública, têm sua fundamentação baseada nos princípios destacados.

As diretrizes gerenciais, da mesma maneira que as normas vigentes, devem manter congruência com os princípios constitucionais da Administração Pública. Nesse contexto, destaca-se a importância de qualquer tomada de decisão dentro do ambiente público, em particular, a “simples” escolha de um *software*.

2.3 Segurança das informações

Conforme apresentado na Subseção 2.1.1, a Superintendência Central de Auditoria Operacional, que será o universo analisado neste trabalho, desempenhava diretamente a função de Auditoria e Controle Interno no Estado de Minas Gerais.

A partir de 1995, a informática passou a ser efetivamente utilizada como instrumento de auxílio no trabalho do auditor; logo todos os relatórios emitidos passaram ser armazenados nos computadores da SCAO. Com relação aos anos anteriores, todo material foi microfilmado, tendo seu conteúdo catalogado e controlado por um *software* específico.

Além do armazenamento dos relatórios finais, a Superintendência mantinha guarda de todos os “papéis de trabalho¹³” em meio magnético, os quais poderiam variar entre correspondências eletrônicas (*e-mails*), documentos texto, planilhas ou bases de dados complexas. Percebe-se que toda a memória do Controle Interno do Poder Executivo do Estado de Minas Gerais encontrava-se armazenada na SCAO¹⁴, sendo impossível quantificar seu valor agregado, dada sua relevância. O ambiente computacional passou a ser fator crítico de sucesso para o desempenho da instituição com destaque para a segurança das informações.

2.4 Definição do problema

Segundo [UCHÔA (2003b)], merecem destaque os seguintes elementos de um ambiente computacional, sob a ótica da segurança:

¹³ “Os papéis de trabalho são o conjunto de documentos e apontamentos com informações e provas coligidas pelo auditor, preparados de forma manual, por meios eletrônicos ou por outros meios, que constituem a evidência do trabalho executado e o fundamento de sua opinião” [CFC (1997)].

¹⁴ Atualmente a guarda dessa informação é de competência da AUGÉ, sendo compartilhada entre SCAG e SCAO.

1. **Confiança:** é possível confiar na disponibilidade do sistema? os dados armazenados vão estar acessíveis quando forem necessários? os mecanismos de backups são suficientes para garantir que as informações armazenadas possam ser recuperadas com facilidade em caso de problemas?
2. **Integridade:** os dados recuperados são confiáveis? como garantir que as informações não foram alteradas na fonte ou no tráfego de dados? como garantir que o que foi acessado é idêntico ao que foi armazenado?
3. **Confidencialidade:** como certificar que os dados só podem ser acessados por quem de direito? como garantir a privacidade dos usuários e dos dados? como impedir a espionagem de informações?

Em abordagem semelhante, [RUSSEL; GANGEMI (1991)] destacam quatro atributos para segurança de sistemas computacionais:

1. **Autenticidade:** entidades, como usuários e processos, devem ter sua identidade devidamente certificada a fim de possibilitar o emprego de controles de acesso eficientes aos recursos de um sistema computacional, bem como permitir a realização de auditorias;
2. **Confidencialidade:** um sistema computacional seguro deve evitar que qualquer informação seja revelada para entidades que não possuam autorização para acessá-la;
3. **Integridade:** o sistema deve impedir que as informações nele contidas sofram modificações não autorizadas, sejam essas acidentais ou intencionais;
4. **Disponibilidade:** o sistema deve manter as informações disponíveis para os seus usuários legítimos.

Dessa forma, temos a definição do problema motivador desta monografia: *Considerando a natureza pública da SCAO e a importância das informações armazenadas em seus computadores, como estabelecer, no âmbito de sua rede interna, mecanismos de controle de acesso aos arquivos que garantam confiabilidade, confidencialidade e integridade dos dados, além de permitir realização de auditorias?*

2.5 Considerações finais

Para o estabelecimento de mecanismos de controle de acesso que garantissem a segurança das informações armazenadas na SCAO seria necessário o desenvolvimento ou aquisição de um *software* específico para essa finalidade. Como não

havia possibilidade de desenvolver tal ferramenta, a opção restante seria adquirir uma solução disponível no mercado. Entretanto, alguns fatores influenciariam diretamente nessa estratégia:

1. Todos os equipamentos deveriam manter os sistemas operacionais já instalados – o que implicava que a solução deveria ser capaz de fazer a comunicação entre as estações *Windows* e o servidor *Solaris*, no caso da utilização desse último.
2. Não havia recursos orçamentários e financeiros disponíveis para compra de *software*.
3. Os princípios constitucionais deveriam ser respeitados – o que descartava o uso de *software* não licenciado, por ferirem diretamente os princípios da legalidade, moralidade, razoabilidade e, dependendo da qualidade do *software*, eficiência.

Nesse contexto, a escolha de uma solução que conjugasse as considerações apresentadas com as necessidades operacionais recaiu estrategicamente sobre uma ferramenta *Livre!*¹⁵: o SAMBA.

Nos capítulos posteriores será apresentada uma discussão sobre *Software Livre*, as vantagens estratégicas de sua utilização, e sobre o SAMBA, antes de realizar a análise evolutiva da adoção da ferramenta pela SCAO.

¹⁵ A expressão *Livre!* é amplamente utilizada para reforçar a idéia de liberdade. Em termos práticos é equivalente à afirmação *Software Livre*.

Capítulo 3

Software Livre

3.1 Conceitos fundamentais

Software Livre é o *software* disponibilizado, de forma gratuita ou não, com as premissas de liberdade de instalação; plena utilização; possibilidade de modificações/aperfeiçoamentos para necessidades específicas; distribuição da forma original ou modificada, com ou sem custos. Contrapõe-se ao modelo *software proprietário* onde o usuário de *software* não tem permissão para redistribuí-lo nem alterar seu funcionamento para ajustar-se às suas necessidades – via de regra, os códigos fonte¹ não acompanham uma distribuição proprietária.

A manutenção do modelo proprietário está fundamentada na lógica que os usuários finais não irão conseguir produtos (programas e documentações) de qualidade sem incentivo financeiro para o programador². Como consequência, a estratégia adotada para manutenção do modelo proprietário defende o sigilo na divulgação do código fonte, bem como criação de mecanismos restritivos para utilização dos executáveis³, sob forma de licenças de utilização. Cabe ressaltar que ainda que parte do código fonte seja disponibilizado, o *software* permanece proprietário se forem mantidas restrições de controle de uso, distribuição e modificação.

Além das questões mencionadas, o modelo proprietário pode ser analisado como uma resposta às pressões do mercado, conforme destaca [HEXSEL (2002)]:

Software proprietário (não-livre) geralmente é produzido com a finalidade de obtenção de lucro e portanto está sujeito à três tipos principais de pressões de mercado: [1] inclusão de funcionalidades “im-

¹ “O código fonte de um programa é a versão daquele programa produzida diretamente pelo autor, e que descreve o comportamento, ou função do programa. Código fonte é geralmente escrito em uma linguagem de alto nível como C, Java ou Perl” [HEXSEL (2002)].

² Essa visão é representada por Bill Gates em sua *Carta aberta aos hobbistas* [GATES (1976)].

³ “O código executável, ou a versão em ‘binário’, de um programa é aquela que resulta do processo de tradução automática do código fonte para uma versão que será executada diretamente pelo computador” [HEXSEL (2002)].

prescindíveis” (e freqüentemente inúteis), [2] obsolescência programada para possibilitar a venda de novas versões, e [3] prazos de desenvolvimento e testes muito curtos para atender às pressões já mencionadas. Na tentativa de atender ao mercado, e especialmente ao departamento de marketing, produtos importantes de *software* são vendidos antes de terem passado por testes suficientes, e estarem portanto estáveis e livres da maioria dos erros de programação.

Uma questão relevante refere-se aos formatos de arquivos. Um dos maiores problemas associados ao *software* proprietário é a utilização de formatos exclusivos (proprietários) para codificação da informação tratada/formatada pelos aplicativos:

Considere-se, a título de exemplo, o uso difundido em uma organização de um editor de textos que emprega um formato proprietário, que chamaremos de formato X_1 . Após um período inicial de acomodação de toda a organização ao novo editor, uma parcela significativa de toda a documentação produzida passa a ser gerada e armazenada com a formatação daquele editor, no formato X_1 .

[...] o fornecedor do editor lança uma nova versão do produto, com uma série de características adicionais, com um formato X_2 que é ligeiramente incompatível com o formato X_1 . Em geral, as novas versões são compatíveis com as (imediatamente) anteriores, exceto pelas novas características. Assim que alguma das novidades passe a ser usada regularmente por parte relevante da organização (central de pagamentos, por exemplo), as cópias do editor devem ser atualizadas para a última versão, para que todos os usuários possam se beneficiar da ‘novidade’. [...] se a ‘novidade’ for apenas cosmética, paga-se um preço talvez elevado por algo de utilidade discutível [HEXSEL (2002)].

Existem, ainda, problemas na incompatibilidade do formato proprietário com os formatos de produtos similares. Tomando ainda o formato X_1 como exemplo, uma proposta de adoção de outro produto encontrará resistências em função de cultura estabelecida e custos (psicológicos e operacionais) associados à conversão de documentos para o novo formato.

3.2 A filosofia do *Software* Livre

3.2.1 Free Software Foundation e o projeto GNU

Movido pela idéia de combater as licenças de utilização, que paulatinamente restringiam a liberdade dos usuários, Richard Stallman fundou, em 1984, a *Free Software Foundation*, com objetivo de estabelecer uma nova filosofia de produção de *software*: a filosofia do *Software* Livre, destacada em [FSF (2003)]:

“*Software* Livre” é uma questão de liberdade, não de preço. Para entender o conceito, você deve pensar em “liberdade de expressão”, não em “cerveja grátis” [*sic!*].

“*Software* livre” se refere à liberdade dos usuários executarem, copiarem, distribuírem, estudarem, modificarem e aperfeiçoarem o *software*. Mais precisamente, ele se refere a quatro tipos de liberdade, para os usuários do *software*:

- a liberdade de executar o programa, para qualquer propósito (liberdade no. 0);
- a liberdade de estudar como o programa funciona, e adaptá-lo para as suas necessidades (liberdade no. 1) – acesso ao código fonte é um pré-requisito para essa liberdade;
- a liberdade de redistribuir cópias de modo que você possa ajudar ao seu próximo (liberdade no. 2);
- a liberdade de aperfeiçoar o programa, e liberar os seus aperfeiçoamentos, de modo que toda a comunidade se beneficie (liberdade no. 3) – acesso ao código fonte é um pré-requisito para essa liberdade.

Dessa forma, o movimento *Software* Livre, surgiu para reaver uma modalidade de conduta entre programadores: a ação colaborativa. Apesar do movimento ser contemporâneo à explosão da internet, seus princípios básicos remontam ao início da história da computação:

O conceito do *Software* Livre é antigo. Quando os primeiros computadores chegaram às universidades, eram ferramentas de pesquisa. O *software* era distribuído livremente, e os programadores eram pagos pela função de programar, não pelos seus programas. Somente mais tarde, quando os computadores alcançaram o mundo dos negócios, os programadores passaram a se sustentar restringindo direitos a seu *software* e cobrando taxas por cópia. A idéia (política) do *Software* Livre foi popularizada por Richard Stallman desde 1984, quando ele fundou a *Free Software Foundation* e o projeto GNU [PERENS (1999)].

De fato, no início da computação (décadas de 50 e 60), o aspecto mais relevante de qualquer sistema era a máquina (*hardware*) – programas eram distribuídos gratuitamente e constantemente aprimorados. Não era concebida a idéia da venda de *software*, que só ocorreu em 1975, quando Bill Gates comercializou um interpretador BASIC para um micro da época, o Altair. Com o passar do tempo, o *software* ganhou relevância e acabou por se tornar mais importante que o *hardware*. Assim, fabricantes apoiaram a idéia de Gates, e passaram a vender

licenças de utilização de aplicativos, sem fornecer o código fonte, com restrições para (re)distribuição.

A *Free Software Foundation* iniciou, em 1984, o projeto GNU⁴, em contraposição às restrições impostas pelo modelo proprietário. A idéia do projeto era criar um sistema operacional e uma série de utilitários e aplicativos oriundos da filosofia *Software Livre*, tendo como ambiente de desenvolvimento o Unix⁵. Stallman desenvolveu o editor de textos *emacs* e o compilador *gcc*, e no início da década de 90 já haviam sido criadas várias ferramentas, restando o núcleo (*kernel*) do sistema operacional.

Essa necessidade foi atendida por uma vitoriosa experiência de ação colaborativa comandada por Linus Torvalds. Ele havia desenvolvido o núcleo do sistema operacional Linux⁶, e disponibilizou o seu código na internet para que outros programadores ajudassem na tarefa de criação de um sistema estável e seguro. A resposta da comunidade foi positiva e em pouco tempo o Linux passou a integrar o projeto GNU:

Quando Linus Torvalds escreveu o Linux, ele completou a última grande lacuna. Pessoas puderam então colocar o Linux junto com o sistema GNU para compor um sistema livre completo: um sistema GNU baseado em Linux (ou sistema GNU/Linux, para simplificar) [STALLMAN (2003)].

[FERRAZ (2002)] conclui sobre a proposta de Stallman e o período marcado pela utilização de *software* proprietário:

Olhando para o passado, nós vemos que as idéias defendidas por Stallman nada mais são do que direitos que por muito tempo havíamos esquecido, e que durante esse período a indústria de *software* conseguiu prosperar erguendo barreiras artificiais, dividindo e conquistando usuários, duplicando esforços e criando produtos de baixa qualidade, que causam perdas de bilhões de dólares todos os anos⁷.

Portanto, o movimento *Software Livre* pode ser visto como um retorno ao espírito de cooperação, característico dos primeiros programadores, com objetivo de oferecer liberdade de escolha e qualidade dos programas.

⁴ GNU é uma sigla recursiva que significa *GNU is Not Unix*. “Curiosamente, essa sigla não explica o que é GNU, mas somente o que não é” [FERRAZ (2002)].

⁵ “O ambiente de desenvolvimento do Projeto GNU era Unix porque este era o sistema tecnicamente mais avançado, estava disponível para várias plataformas, era distribuído mais livremente que os outros sistemas, sendo por essas razões o mais popular na academia e em setores da indústria de computação” [HEXSEL (2002)].

⁶ <http://www.linux.org>.

⁷ Em [FERRAZ (2002)] foram tratados custos da baixa qualidade de *software*.

3.2.2 O princípio básico da colaboração

Eric Raymond, em [RAYMOND (1998)], faz uma análise interessante a respeito da construção de *software*. Segundo sua visão, as formas de produção de *software* podem ser comparadas ao modelo “catedral” ou ao modelo “bazar”. O primeiro modelo sugere o enclausuramento como forma de produção, enquanto o modelo “bazar” remete a um grande turbilhão de opiniões, conforme assinala:

[...] Eu acreditava que os programas mais importantes (sistemas operacionais e ferramentas realmente grandes como Emacs) necessitavam ser construídos como as catedrais, habilmente criados com cuidado por mágicos ou pequenos grupos de magos trabalhando em esplêndido isolamento, com nenhum beta para ser liberado antes de seu tempo. [...] a comunidade Linux pareceu assemelhar-se a um grande e barulhento bazar de diferentes agendas e aproximações [...] de onde um sistema coerente e estável poderia aparentemente emergir somente por uma sucessão de milagres [RAYMOND (1998)].

A experiência com um projeto de código livre, o Fetchmail⁸, possibilitou a Raymond concluir que o *software* produzido no modelo “bazar” apresenta melhoria significativa para a qualidade do produto final, uma vez que o processo de depuração do código tende a ser feito com mais rapidez e mais qualidade. Ao se trabalhar em comunidade, naturalmente o erro será um fenômeno comum, pois uma versão do programa será analisada por vários desenvolvedores, e a possibilidade de solução ser trivial para alguém é muito grande. Assim, reduz-se o lapso entre a detecção e a correção de imperfeições, ao contrário do modelo “catedral” onde o exame de código é feito por um grupo restrito, demandando tempo considerável, com a possibilidade de que nem todos os erros sejam detectados.

A regra do modelo “bazar” foi tratada em [RAYMOND (1998)] como “Lei de Linus⁹”: “dada uma base grande o suficiente de beta-testes e co-desenvolvedores, praticamente todo problema será caracterizado rapidamente e a solução será óbvia para alguém”; ou em sua forma favorita, “dados olhos suficientes, todos os erros são triviais”.

O modelo “catedral”, empregado na elaboração de *software* proprietário, reflete bem a pressão do mercado sobre a indústria de *software*. O projeto fica a cargo de um grupo restrito de projetistas, enquanto o trabalho é executado por um grupo de programadores. A gerência do projeto, atendendo às pressões do mercado, estabelece metodologias e prazos que devem ser cumpridos à risca. Esse modelo, segundo [HEXSEL (2002)] dificulta formação de massa crítica de usuários e desenvolvedores nas fases iniciais do desenvolvimento de um sistema. Como a fase de testes fica a cargo de um grupo restrito, existe certa probabilidade de

⁸ Página oficial: <http://www.catb.org/~esr/fetchmail>.

⁹ Uma referência direta à Linus Torvalds e seu estilo de desenvolvimento bazar.

um erro não ser detectado. Quando isto acontece, o produto é disponibilizado e comercializado sem atingir a estabilidade necessária. “São conhecidos inúmeros exemplos de produtos que foram disponibilizados e vendidos antes de atingirem a estabilidade ou a maturidade simplesmente porque o prazo de ‘desenvolvimento’ se esgotou” [HEXSEL (2002)].

Dois componentes são fundamentais para a alta qualidade dos programas desenvolvidos no modelo “bazar”:

1. publicação do código fonte do programa.
2. número de desenvolvedores e usuários envolvidos no processo;

Dessa forma, com a publicação do código fonte, as portas para depuração, descoberta e correção de erros estão abertas para quem se interessar¹⁰. Por consequência, a massa crítica é facilmente atingida com a atuação da comunidade, cada vez maior, de desenvolvedores e usuários que se incumbem da tarefa de realizar testes e correções. É interessante notar que não existem prazos para serem cumpridos nem metas a serem alcançadas. O *software* evolui à medida que melhorias vão sendo implementadas. A Tabela 3.1 mostra os principais pontos dos modelos apresentados.

Tabela 3.1: Comparação entre os modelos Catedral e Bazar.

Modelo Catedral	Modelo Bazar
- trabalho controlado por um grupo de projetistas;	- não há hierarquia entre os participantes;
- estabelecimento de metodologias, tarefas e prazos;	- projetos informalmente organizados ao redor de uma proposta inicial;
- demora para lançamento de atualizações;	- participação voluntária de interessados;
- dificuldade para atingir <i>massa crítica</i> ;	- atualizações constantes;
- dificuldade para atingir qualidade esperada;	- <i>massa crítica</i> atingida rapidamente;

¹⁰ Cabe ressaltar que a relevância (popularidade e/ou importância) de um projeto são fundamentais para despertar o interesse da comunidade.

Como exemplo de sucesso de modelos “bazar”, além dos já mencionados GNU/Linux e Fetchmail, merecem destaque o SAMBA¹¹; o sistema de processamento de textos T_EX¹²; o servidor *www* Apache¹³; o manipulador de imagens Gimp¹⁴; os ambientes de trabalho Window Maker¹⁵ e Gnome¹⁶; o gerenciador de finanças pessoais GnuCash¹⁷; o pacote de escritório OpenOffice¹⁸; entre outros. Um amplo catálogo de *Software* Livre é mantido por [FSF/UNESCO (2004)], atualmente com mais de 2.700 programas¹⁹.

Percebe-se que o modelo “bazar”, utilizado na criação de *Software* Livre, tem como resultado programas eficientes, de alta qualidade, capazes de desenvolver vantagens competitivas, como será apresentado na seção 3.4.

3.3 Licenças de *Software*

“Para entender o conceito de *Software* Livre deve-se pensar em ‘liberdade de expressão’, e não em ‘cerveja grátis’” [*sic!*] [FSF (2003)]. Essa afirmação serve para tentar dissipar a dificuldade de entendimento entre o que é *Software* Livre e o que é *software* grátis.

Tal preocupação é recorrente da polêmica emergida, principalmente, devido ao significado do termo *free* na “língua pátria” do movimento GNU. Em inglês, o termo *free* possui dois significados: grátis e livre, sendo esse último o defendido por Richard Stallman e pela Free Software Foundation para designar *Software* Livre. Entretanto, como o *software* está intimamente ligado aos avanços tecnológicos e ao mercado de produção, é natural que se associe o termo *free* com gratuidade, custo zero.

Na intenção de evitar essa ambigüidade, que felizmente não ocorre na língua portuguesa, a categoria de programas *Software* Livre também é denominada *Open Source* (*código aberto*²⁰) – terminologia preferida por Eric Raymond. Mas ainda surgem diferenças entre *código aberto* e *Software* Livre em função da existência de programas cujo código fonte é aberto para investigação, mas não para alteração, caracterizando-se, em última instância, código proprietário.

Dessa forma, destacam-se alguns incômodos das duas posições:

¹¹ <http://www.samba.org>.

¹² Este trabalho foi preparado utilizando o L^AT_EX 2_ε, um sistema derivado do T_EX. O maior referencial sobre T_EX encontra-se em <http://www.ctan.org>.

¹³ <http://www.apache.org>.

¹⁴ <http://www.gimp.org>.

¹⁵ <http://www.windowmaker.org>.

¹⁶ <http://www.gnome.org>.

¹⁷ <http://www.gnucash.org>.

¹⁸ <http://www.openoffice.org>.

¹⁹ O número de aplicativos catalogados em 14/01/2004 era 2.775.

²⁰ Página oficial do movimento: <http://www.opensource.org>.

1. O termo *Free Software* pode ser usado em programas gratuitos, porém proprietários.
2. O termo *Open Source* pode se referir a programas onde o usuário tem acesso ao código, mas não pode alterá-lo ou redistribuí-lo.

Em síntese, na língua inglesa os dois termos são ambíguos. Como a língua portuguesa descarta essa dúbia interpretação, este trabalho utilizará a terminologia *Software Livre*.

“Assim, mais do que o termo, é importante conhecer a licença de uso de um programa em questão. Essa licença é que vai dizer, em geral, o que pode e o que não pode ser feito com esse aplicativo” [UCHÔA (2003a)]:

Licença é o termo de outorga de direitos em que o autor define qual o grau de liberdade que terceiros possuem para modificar e/ou redistribuir um programa e/ou seus trabalhos derivados. Geralmente, a licença restringe a liberdade de uso [HEXSEL (2002)].

Ainda hoje, a grande maioria dos programas é distribuída sob as restrições impostas pelas licenças de uso, ou EULA's (*End User Licence Agreements*). Dessa maneira, o usuário não adquire o *software* no sentido *lato*, na verdade ele o “licencia” para utilização, sob certas circunstâncias listadas na EULA²¹.

3.3.1 A licença GPL

A licença criada e difundida pela Free Software Foundation, a *General Public License* (GPL) não restringe o uso do *software*. A GPL visa garantir, para qualquer usuário, os direitos de utilização do programa, assim como permite realização de cópias e modificações, que pressupõem estudo do código fonte, conforme [FSF (2003a)]²²:

Quando nos referimos a *software* livre, estamos nos referindo a liberdade e não a preço. Nossa Licença Pública Geral foi desenvolvida para garantir que você tenha a liberdade de distribuir cópias de *software* livre (e cobrar por isso, se quiser); que você receba o código fonte ou tenha acesso a ele, se quiser; que você possa mudar o *software* ou utilizar partes dele em novos programas livres e gratuitos; e que você saiba que pode fazer tudo isso.

Para proteger seus direitos, precisamos fazer restrições que impeçam a qualquer um negar estes direitos ou solicitar que você deles abdique.

²¹ [FERRAZ (2002)] salienta que algumas EULA's chegam a proibir a engenharia reversa, o que significa proibir qualquer tentativa para aprender como funciona o programa.

²² O Apêndice B apresenta a tradução completa da GPL para português do Brasil.

Estas restrições traduzem-se em certas responsabilidades para você, se você for distribuir cópias do *software* ou modificá-lo.

Por exemplo, se você distribuir cópias de um programa, gratuitamente ou por alguma quantia, você tem que fornecer aos recebedores todos os direitos que você possui. Você tem que garantir que eles também recebam ou possam obter o código fonte. E você tem que mostrar-lhes estes termos para que eles possam conhecer seus direitos.

Paradoxalmente, a GPL se vale de restrições. Entretanto, essas restrições servem para garantir ao usuário as quatro liberdades básicas da filosofia *Software Livre*²³. A GPL permite que o autor distribua livremente o seu código e, também, que outra pessoa simplesmente modifique-o conforme suas necessidades. Caso essa pessoa deseje redistribuir o código inicial com suas modificações, as restrições (da GPL) determinam que o “novo” código também esteja disponível sob GPL. Portanto, “qualquer um que redistribui o software (*Livre!*), com ou sem modificações, tem que passar adiante as liberdades de fazer novas cópias e modificá-las” [FSF (2001a)]. O método legal de tornar um programa em *Software Livre* e exigir que todas as versões modificadas e extendidas do programa também sejam livres é conhecido por *copyleft*²⁴.

Para que se tenha uma idéia clara de como funciona o *copyleft*, basta imaginarmos o texto legal mais restritivo do ponto de vista do *copyright*, em que se proíbe tudo aquilo que for possível com relação à cópia, distribuição, alteração etc., e ao final do texto colocarmos um único parágrafo dizendo: “Concedo todos esses direitos a terceiros, desde que esses direitos sejam mantidos em relação aos demais usuários” [VALOIS (2003)].

Portanto, os termos do *copyleft* garantem o reconhecimento do autor. Entretanto permitem intervenções, alterações e redistribuições de terceiros. Com isso o “produto” modificado sempre deverá ser mantido livre conforme sua versão original.

3.4 *Software Livre em ambientes corporativos*

Cada vez mais, a Tecnologia da Informação se concebe como fator decisivo para desenvolver competitividade. [McFARLAN (1998)] destaca que a evolução da tecnologia de sistemas de informação possibilitou que “os sistemas de computadores se transferissem de sistemas de suporte às atividades de escritório, para sistemas que oferecem vantagens competitivas significativas”. A Tecnologia permite

²³ Existem outros tipos de licenças de *Software Livre* que não serão tratadas neste trabalho. [FSF (2001b)] apresenta a lista completa de todas as licenças *Software Livre*, compatíveis ou não com a GPL.

²⁴ Maiores detalhes sobre *copyleft* podem ser obtidas em [FSF (2001a)] e [VALOIS (2003)].

que as organizações desenvolvam potencial para produzir “novas ferramentas afiadas que possam trazer ganhos duradouros em termos de participação no mercado”. Nesse cenário, o *software* passa a ter grande impacto na produtividade, tornando-se elemento fundamental de planejamento estratégico.

Para que uma organização seja competitiva, ela precisa desenvolver vantagens estratégicas. Segundo Michael Porter, vantagem estratégica é toda aquela vantagem que possa reduzir o poder de um fornecedor, reduzir custos, diferenciar seus produtos ou serviços em relação à concorrência, ou oferecer maior segurança e confiabilidade na execução de processos²⁵. Uma análise cuidadosa dessas premissas, demonstra como o *Software Livre* pode refletir um salto qualitativo para estabelecimento de vantagens estratégicas.

- **Redução do poder de um fornecedor:** “O erro estratégico mais grave que uma empresa pode cometer é adotar como padrão as soluções proprietárias de um único fornecedor. Na prática, isto pode significar tornar-se refém desse fornecedor” [FERRAZ (2002)]. De fato, o poder de negociação dos fornecedores influenciam diretamente na indústria. Fornecedores fortes são capazes de elevar preços de produtos finais ou determinar a qualidade dos mesmos – e a área da Tecnologia da Informação não é exceção à essa regra. Entretanto, a adoção de padrões abertos reduz tal poder, chegando a descartá-lo em alguns casos. Isso se deve à possibilidade de autonomia por parte das corporações. O *Software Livre* permite, pelo acesso ao código fonte, que as tecnologias sejam desenvolvidas internamente, ou pelo menos resguarda o conhecimento dentro da corporação. Caso essa tecnologia esteja a cargo de um fornecedor específico, uma eventual troca por um concorrente não torna-se problemática, uma vez que o código fonte é conhecido. Por essa razão, as corporações podem negociar de forma igualitária com fornecedores, sem risco de tornarem-se reféns²⁶.
- **Redução de custos:** Além da redução de custos com pagamento de licenças, o *Software Livre* não exige atualização freqüente do parque computacional (*hardware*). Essa característica não é encontrada no *software* proprietário. [HEXSEL (2002)] argumenta que quando o fornecedor decide comercializar uma nova versão dos aplicativos, o equipamento que os executa também deve ser atualizado ou mesmo substituído, ainda que as “novas” funcionalidades não sejam utilizadas pela grande maioria dos usuários.
- **Diferenciação de produtos ou serviços:** Em função de sua liberdade de uso, modificação e distribuição, o *Software Livre* favorece a inovação. A

²⁵ [PORTER (1986)] introduz reconhecidas técnicas para análise da indústria e concorrentes; [PORTER (1992)] descreve como as empresas podem criar e sustentar vantagem competitiva.

²⁶ [FERRAZ (2002)] apresenta estudos de caso que demonstram o poder de pressão fornecedores e a ação benéfica da utilização de *Software Livre*.

metodologia bazar de produção permite que qualquer pessoa ou instituição retome um trabalho ou inicie uma nova vertente a partir de determinado ponto. Essa flexibilização é um diferencial do *Software Livre* quando comparado aos modelos proprietários. Além disso, o fato do código fonte estar disponível para todos, possibilita combinações de projetos com resultados rápidos e retorno imediato para a comunidade. Dessa forma, o conhecimento compartilhado favorece a diferenciação e a criação de novos produtos ou serviços.

- **Maior segurança e confiabilidade na execução de processos:** Conforme demonstrado na Seção 3.2.2, *Software Livre* oferece maior segurança porque a natureza da filosofia bazar (contando com a disponibilização do código fonte) permite realizar maior controle sobre o que está sendo desenvolvido, possibilitando que problemas e/ou falhas sejam detectados e resolvidos rapidamente, sem permanecerem ocultos por longos períodos.

Percebe-se que a competitividade de uma empresa pode ser largamente amparada pelas vantagens estratégicas oriundas da utilização de modelos abertos.

Assim como na iniciativa privada, as vantagens estratégicas assinaladas por Michael Porter, são diretamente aplicáveis ao setor público. O Estado, especialmente pelo seu cunho contemporâneo de bem-estar social²⁷, deve preocupar-se em reduzir poder de fornecedores, reduzir custos, promover diferenciação dos serviços prestados e oferecer segurança e confiabilidade em seus proceçimentos. Face a essa perspectiva, a Administração Pública deve permanecer atenta aos benefícios obtidos com a utilização de *Software Livre*, buscando, por sua vez, dotar-se de autonomia frente a imposições de fornecedores e promover bem-estar social.

A próxima seção irá abordar questões relativas ao *Software Livre* na Administração Pública.

3.5 *Software Livre* na Administração Pública

Na seção anterior, foram apresentados elementos que permitem concluir sobre a competitividade promovida pelo *Software Livre* em ambientes corporativos. Considerando que a Administração Pública não tem caráter competitivo, a análise deve ser feita sob a ótica da efetividade²⁸ da ação governamental. Assim, qualquer fator

²⁷ Estado de bem-estar, ou *Welfare State*: “Sistema econômico baseado na livre-empresa, mas com acentuada participação do Estado na promoção de benefícios sociais. Seu objetivo é proporcionar ao conjunto dos cidadãos padrões de vida mínimos, desenvolver a produção de bens e serviços sociais, controlar o ciclo econômico e ajustar o total da produção, considerando os custos e as rendas sociais” [SANDRONI (1989)].

²⁸ Neste trabalho, o conceito de efetividade está intimamente ligado à avaliação qualitativa dos serviços públicos. “Recupera-se, com o conceito de efetividade, a noção de que o governo deve, acima de tudo, prestar bom serviços. É a ótica da qualidade que começa a ser incorporada pelo

que possibilite atender uma demanda social com qualidade, ou organizar-se para tal, deve ser considerado como vantagem estratégica para o setor público. Nesse aspecto, a adoção do *Software* Livre, por parte do Estado, é amplamente amparada, principalmente, por três princípios constitucionais: impessoalidade, razoabilidade e eficiência²⁹.

- **Impessoalidade:** Uma vez que a Administração Pública deve dispensar igualdade de tratamento aos seus administrados, com vistas à impedir favorecimento de alguns indivíduos, é fundamental que fornecedores sejam enquadrados nesse princípio. Conforme destacado na Seção 3.4, a adoção de *Software* Livre reduz consideravelmente o poder de um fornecedor, impedindo que o Estado permaneça refém de tecnologia proprietária.

Outro fator relevante, salientado por [HEXSEL (2002)], refere-se que o desenvolvimento de *Software* Livre é orientado para benefícios de usuários, não de um fabricante. Se levarmos ao extremo da análise, é possível verificar que o investimento em treinamentos para usuários de *software* proprietário, têm como resultado, “empregados” treinados para o fabricante. Em outras palavras, após a “venda”, o fornecedor adquire como contrapartida, além dos recursos financeiros aplicados, mão-de-obra treinada, exclusivamente, para utilizar seus próprios produtos.

No caso do *Software* Livre, assim que um aplicativo é disponibilizado, ele torna-se um bem público, cuja utilização e evolução é determinada pela comunidade de usuários, com possibilidade de adequação e redistribuição de versão alterada.

- **Eficiência:** Uma vez que a Administração Pública deve prestar serviços da mais alta qualidade, buscando atingir os melhores resultados, a escolha sobre utilização de *software*, sob essa ótica, deve se direcionar pelas soluções *Livres!* Conforme destacado na Seção 3.4, a metodologia bazar de produção promove diferenciação de produtos e serviços, produzindo sistemas reconhecidamente robustos e seguros, por consequência, eficientes.

Além de maximizar resultados, outra vertente desse princípio refere-se a formas de minimizar custos, com utilização racional de métodos, técnicas e normas. [HEXSEL (2002)] salienta que o desembolso inicial na adoção de *Software* Livre é bem inferior ao modelo proprietário, podendo atingir valores irrisórios nessa comparação. Aliado a isso, *Software* Livre promove outro fator de redução de custos: não exige atualização constante do hardware³⁰.

modelo gerencial” [ABRUCIO (1997)].

²⁹ Os princípios basilares da Administração Pública foram tratados subseção 2.2.2

³⁰ [HEXSEL (2002)] definiu essa característica como não-obsoloscência dos componentes de hardware.

- **Razoabilidade:** Talvez esse seja o princípio que mais se relaciona às questões de utilização do *Software* Livre, uma vez que todas as tomadas de decisões devem contribuir efetivamente para satisfatório atendimento dos interesses públicos. Se a Lei não é capaz de atingir a todas as peculiaridades da administração, o gestor deve balizar-se pela lógica do interesse público sobre o privado (impessoalidade), pela eficiência no atingimento de suas metas, pela efetividade das ações de governo, corroborando na defesa dos interesses do Estado e dos seus cidadãos. Logo, todas as argumentações para adoção de *Software* Livre elencadas para *impessoalidade* e *eficiência* consolidam-se, também, sob o prisma da razoabilidade:

- redução do poder de fornecedor;
- não se fica refém de tecnologia proprietária;
- desenvolvimento orientado para benefícios de usuário;
- possibilidade de adequação de aplicativos e redistribuição de versão alterada;
- robustez e segurança;
- promover redução de custos e aproveitamento de hardware;

Aliados a esses tópicos, [HEXSEL (2002)] apresenta mais uma importante vantagem estratégica, relativa ao suporte abundante disponível gratuitamente pela Internet³¹:

Um dos preconceitos mais comuns sobre a aplicabilidade de *Software* Livre refere-se à falta de suporte. Esta reclamação carece de fundamento quando se considera a existência de uma comunidade internacional de usuários e programadores, acessível através de Internet, dispostos a colaborar na solução de eventuais problemas com os programas. Não é incomum de se obter auxílio do próprio projetista ou programador do aplicativo ou sistema para o qual se busca auxílio.

Em última instância, [HEXSEL (2002)] salienta que a dominação do mercado por padrões proprietários de um único fornecedor, de forma que uma parcela da economia nacional (ou regional) dependa desses sistemas, envolve riscos de alto impacto para o setor público, além de onerar o orçamento do ente da federação e o balanço financeiro, no caso de fornecedor estrangeiro. A adoção de *Software* Livre nesses casos diminuiria os riscos envolvidos e possibilitariam a criação de novos empregos em pequenas e médias empresas de desenvolvimento, suporte e manutenção, no âmbito federal, estadual e municipal.

³¹ O idioma mais utilizado na Internet (inglês) pode surgir como complicador para o acesso ao suporte.

3.6 Considerações finais

[DIMANTAS (2003)] argumenta que “uma sociedade não pode crescer se a sua capacidade de criação e inovação estiver limitada pelo desconhecimento ou pela imposição de um sistema fechado”. Por isso, defende a construção de uma “sociedade catalisada pela liberdade na utilização da tecnologia”, subsidiada pelo conhecimento livre e acessível para todos³². Nesse cenário, o *Software Livre* tem um importante papel a desempenhar: o de subsidiar o desenvolvimento humano.

O conceito do *Software Livre* está cada vez mais penetrando nas decisões da sociedade da informação. É uma tendência inexorável. O GNU/Linux está na boca do povo. O que isto significa? Trata-se de um avanço, não apenas do ponto de vista tecnológico, mas, sobretudo, no campo das idéias, do desenvolvimento humano. Essa nova forma de produzir e gerir o conhecimento promove um retorno da importância do ser humano no jogo da vida. O ser humano esteve, nos últimos séculos, submetido ao domínio do pensamento burocrático da Era Industrial, que priorizou a produção e o consumo de massa [DIMANTAS (2003)].

Software Livre torna-se uma boa alternativa para que o setor privado desenvolva vantagens estratégicas essenciais para competitividade das organizações. No setor público, essas vantagens – fortemente amparadas pelos princípios da impessoalidade, eficiência e, especialmente, razoabilidade – possibilitam melhoria na qualidade dos serviços prestados, além de promover desenvolvimento tecnológico e social. [PINHEIRO (2003)] defende a idéia que o Estado, como ente fomentador do desenvolvimento tecnológico e da democratização do acesso a novas tecnologias para a sociedade, não pode deixar de priorizar, em todos os setores, a utilização de programas abertos frente aos padrões proprietários.

Amparada nas argumentações apresentadas sobre a importância da adoção de *Software Livre* também no setor público, a SCAO elegeu o SAMBA para atender às necessidades de autenticação de usuários e controle de acesso aos arquivos de trabalho em meio magnético.

³² Esse conceito associa-se à espiral do conhecimento definida por [NONAKA; TAKEUCHI (1997)], onde a transformação de conhecimento tácito para explícito é regida pela socialização, internalização, externalização e combinação.

Capítulo 4

SAMBA

SAMBA é um conjunto de programas integrados desenvolvido para autenticação de usuários e compartilhamento de arquivos e impressoras em redes mistas. É distribuído sob licença GPL, caracterizando-se, conforme Subseção 3.3.1, *Software Livre*. Sua página oficial é <http://www.samba.org>, a qual possui *mirrors* distribuídos por todo mundo. A Figura 4.1 representa o *site* oficial, hospedado na Argentina¹.



Figura 4.1: Web Site do SAMBA

¹ Em 19/01/2004 não existia nenhum *mirror* brasileiro para o SAMBA.

O SAMBA destaca-se por ser uma ferramenta extremamente útil para quem deseja utilizar sistemas Windows e Unix em sua rede. “Rodando em uma máquina com Unix, permite que estações Windows acessem seus arquivos e impressoras, assim como possibilita o acesso a recursos compartilhados por sistemas Windows para máquinas Unix” [TS; ERKSTEIN; COLLIER-BROW (2003)]. A seguir, é apresentada uma definição formal:

O SAMBA é um servidor e conjunto de ferramentas que permite que máquinas Linux e Windows se comuniquem entre si, compartilhando serviços (arquivos, diretórios, impressão) através do protocolo SMB (*Server Message Block*)/CIFS² (*Common Internet File System*), equivalentes à implementação NetBEUI³ no Windows. O SAMBA é uma das soluções em ambiente Unix capaz de interligar redes heterogêneas [SILVA (2003)].

Com relação à heterogeneidade do SAMBA, [STATO FILHO (2002)] destaca a possibilidade de integração em outras plataformas, além das já mencionadas Unix e Windows, como IBM – OS/2, variantes do Unix, Macintosh, Amiga OS, Novel e Netware.

Uma das principais aplicações do SAMBA refere-se à sua utilização como PDC em uma rede. PDC é a sigla de *Primary Domain Controller*, um servidor que mantém uma base de dados de usuários, autenticando-os no seu domínio e liberando automaticamente os acessos previstos para cada um.

Para configurações mais apuradas a ferramenta apresenta ainda várias funcionalidades, a exemplo de:

- controle de acesso leitura/gravação por usuário autenticado ou por compartilhamento;
- criação de servidor de impressão;
- criação de servidor WINS⁴;
- controle de endereços válidos para acessar os recursos compartilhados;

² O CIFS baseia-se no SMB. O SMB foi uma extensão que a Microsoft adicionou ao DOS para redirecionar E/S de disco para um sistema conhecido como NetBIOS – *Network Basic Input/Output System* [NEMETH (2002)].

³ *NetBIOS Extended User Interface* – “criado para que as mensagens NetBIOS pudessem ser enviadas por uma rede Token Ring ou Ethernet. Assim como o NetBIOS, NetBEUI foi projetado para um ambiente de rede local e não incluía nenhuma informação para roteamento” [JÚNIOR (2000)].

⁴ *Windows Internet Name Service*. Segundo [JÚNIOR (2000)] é uma implementação para resolução de nomes em redes separadas por roteadores, utilizando NetBIOS.

- utilização de banco de dados de senha do sistema, `/etc/passwd`, bem como integração com PAM⁵ e LDAP⁶;

Existem outras funcionalidades e possibilidades que não serão tratadas neste trabalho. Uma lista mais detalhada é apresentada em [SILVA (2003)].

Conforme apresentado, o SAMBA é uma solução de grande utilidade por permitir integração entre sistemas operacionais diferentes. Aliado a isso, por se tratar de um *Software Livre* a ferramenta adquire estabilidade e confiabilidade, conseqüências imediatas da metodologia bazar⁷.

4.1 Histórico e características

De acordo com [NEMETH (2002)], o SAMBA foi originalmente criado por Andrew Tridgell⁸, que fez a engenharia reversa do protocolo SMB e publicou seu código em 1992. Seu surgimento é conseqüência da combinação de três fatores recentes na computação [TOSO (2003)]:

1. Necessidade de compartilhamento de arquivos.
2. Unix, e mais atualmente Linux vs Windows, ou estabilidade e robustez vs popularidade – necessidade de conjugar duas plataformas tão antagônicas⁹.
3. Redes de Computadores: rápida proliferação de redes locais (LAN – *Local Area Network*).

Segundo [NEMETH (2002)], os serviços básicos do SAMBA são:

- compartilhamento de arquivos;
- impressão em rede;
- autenticação e autorização;

⁵ *Pluggable Authentication Modules*, refere-se a um conjunto de bibliotecas que controlam as tarefas de autenticação de sistemas *nix e suas aplicações. Uma excelente referência sobre PAM é encontrada em [SICA; UCHÔA; SIMEONE (2003)].

⁶ *Lightweight Directory Access Protocol*, ou Protocolo Leve de Acesso à Diretórios, refere-se a um protocolo aberto para se fazer acesso a um servidor de informações, fornecendo meios para leitura e atualização de dados, como senha de usuários. Maiores detalhes podem ser obtidos em [DOMINGUES; SCHNEIDER; UCHÔA (2001)].

⁷ Conforme destacado na Seção 3.2.2.

⁸ Pela Australian National University – Camberra, Austrália [JÚNIOR (2000)].

⁹ “O Windows é uma plataforma popular [...] capaz de construir uma ponte sobre a lacuna entre o usuário e o cabo de rede que sai da parede. O Unix, por outro, lado é uma plataforma de infraestrutura sólida e escalonável” [NEMETH (2002)]. Não se discute aqui as facilidades de operação advindas das melhorias constantes dos sistemas GNU/Linux que o tornam um concorrente direto do Windows. O que se pretende neste tópico é chamar a atenção para o contexto em que o SAMBA emergiu.

- resolução de nomes;
- anúncio de serviço (servidor de arquivos e “pesquisa” por impressora).

Esses serviços são executados por dois *daemons*¹⁰: `smbd` e `nmbd`. O primeiro implementa serviços de compartilhamento de arquivos e impressão, além de autenticação e autorização; o `nmbd` fornece o serviço de resolução de nomes e anúncio de lista.

4.2 Instalação

O SAMBA pode ser instalado pela compilação de seus arquivos fontes, instalação geral para sistemas Unix/Linux, ou ainda por meio de pacotes pré-compilados. Em http://ar.samba.org/samba/ftp/Binary_Packages encontram-se pacotes de instalação para: Debian, Fedora, IRIX, LSB, Mandrake, OSF, RedHat, SuSE, HP, MVS, Novell, Solaris, VMS. A instalação geral, por meio dos arquivos fonte `.tar.gz`, para todos os sistemas **nix* (como o Slackware, por exemplo), encontra-se disponível em <http://ar.samba.org/samba/ftp>.

Caso o usuário faça a opção pela instalação dos pacotes `.tar.gz`, alguns passos deverão ser observados. Para exemplificação, será utilizada a forma geral¹¹ (padrão) de compilação e instalação sugerida por [TOSO (2003)], para Linux¹².

No diretório gerado na descompactação do arquivo `.tar.gz`, o administrador, utilizando a conta `root`¹³, deverá executar os passos discriminados na Figura 4.2.

```
# ./configure      → configuração do script para compilação;
# make            → compilação do aplicativo;
# make install    → instalação.
```

Figura 4.2: Compilação e instalação do SAMBA

Após a compilação e instalação dos pacotes, é interessante criar links para que os aplicativos sejam executados como se estivessem na *path*, além do arquivo `smb.conf` conforme Figura 4.3.

Finalmente, as linhas do arquivo `/etc/services`, que se referem a Figura 4.4, devem ser descomentadas, caso não estejam. Com esses procedimentos, o

¹⁰ *Daemon* “é um programa que roda em background, sempre pronto para receber solicitações de outros programas, executar o que foi pedido e retornar uma resposta. O termo vem da mitologia grega, onde os *Daemons* são espíritos guardiões” [MORIMOTO (2003)].

¹¹ Maiores informações para instalações personalizadas podem ser obtidas nos arquivos `README` e `INSTALL` no diretório criado após a descompactação.

¹² Esse procedimento pode ser adotado para praticamente Toda distribuição **nix*, com pequenas variações.

¹³ Conta de *superusuário* dos sistemas **nix*. “Dado seu extremo poder, a conta de `root` só deve ser usada em configurações e manutenções no sistema” [SICA; UCHÔA (2003)].

```
# ln -s /usr/local/samba/bin/* /usr/sbin/  
# touch /etc/smb.conf  
# ln -s /etc/smb.conf /usr/local/samba/lib/smb.conf
```

Figura 4.3: Ajustes na instalação do SAMBA

```
netbios-ns      137/tcp      #NETBIOS Name Service  
netbios-ns      137/udp      #NETBIOS Name Service  
netbios-dgm     138/tcp      #NETBIOS Datagram Service  
netbios-dgm     138/udp      #NETBIOS Datagram Service  
netbios-ssn     139/tcp      #NETBIOS Session Service  
netbios-ssn     139/udp      #NETBIOS Session Service  
microsoft-ds   445/tcp      #Direct-Hosted Service  
microsoft-ds   445/udp      #Direct-Hosted Service
```

Figura 4.4: Linhas a descomentar no arquivo `/etc/services`.

SAMBA está apto a operar. A instalação via pacotes pré-compilados normalmente faz todos ajustes citados. Por se tratar de uma ferramenta amplamente utilizada, a maioria das distribuições GNU/Linux disponibiliza versões do SAMBA em suas instalações.

4.3 Configuração

Embora o SAMBA execute funções complexas, sua configuração é relativamente simples e permanece concentrada em um único arquivo, `smb.conf`, tradicionalmente localizado na pasta `/etc` nas distribuições GNU/Linux. Em versões mais atuais, o arquivo `smb.conf` é encontrado no diretório `/etc/samba`.

A configuração do SAMBA pode ser feita de duas maneiras:

- edição direta do arquivo de configuração;
- configuração via *Web*, com a utilização da ferramenta SWAT¹⁴.

Essas maneiras, que serão detalhadas nas subseções posteriores, caracterizam-se por sua simplicidade. Com isso a ferramenta adquire o diferencial de ser fácil e extremamente configurável.

4.3.1 Edição do arquivo `smb.conf`

O arquivo de configuração `smb.conf` consiste em um arquivo texto (ASCII) composto por *seções* e *parâmetros*. Uma seção começa com sua identificação entre colchetes, por exemplo `[projetos]`, e abrange todo texto seguinte até que seja identificada uma nova seção. As seções contêm parâmetros escritos no formato

¹⁴ *Samba Web Administration Tool*.

“*diretiva = valor*”. O valor utilizado após o sinal de igual pode ser numérico, uma *string* ou um valor *booleano*, que pode ser fornecido como **yes** ou **no**, **true** ou **false**, ou mesmo **0** e **1**. A Tabela 4.1 apresenta as principais seções de um arquivo `smb.conf`.

Tabela 4.1: Seções do arquivo `smb.conf`

Seção	Descrição
[global]	Os parâmetros dessa seção serão aplicados ao servidor como um todo, ou serão padrões para seções que não definirem especificamente certos itens.
[homes]	Quando definida, especifica opções de acesso a diretórios particulares dos usuários. O compartilhamento é disponibilizado somente para seu dono após autenticação do usuário no sistema.
[printers]	Se estiver definida, essa opção permite que os clientes possam se conectar e utilizar qualquer impressora especificada e conectada à máquina local, por meio do arquivo <code>/etc/printcap</code> .
[netlogon]	Utilizada para ativar execução de <i>scripts</i> quando um usuário efetua <i>logon</i> no domínio especificado.
[profile]	Define um perfil de usuário quando o SAMBA é configurado como PDC, armazenando configurações de área de trabalho de clientes Windows.
[seção]	Qualquer outra referência de [seção] no arquivo <code>smb.conf</code> , que não seja uma das anteriormente destacadas, será tratada como um compartilhamento ou impressora.

O arquivo de configuração não faz distinção entre maiúsculas e minúsculas, não sendo, portanto, *case sensitive*. Os caracteres ‘#’ e ‘;’ são utilizados para inserção de comentários¹⁵, enquanto o símbolo ‘\’ indica continuação de linha, caso os parâmetros utilizados necessitem.

De acordo com [SICA; UCHÔA; SIMEONE (2003)], o arquivo `smb.conf` pode ser dividido em duas partes: a seção de configuração *global* do servidor, representada na Figura 4.5; e outra parte relativa aos compartilhamentos, conforme destaca a Figura 4.6.

Na seção *global* são definidos padrões essenciais para perfeito funcionamento do servidor, como domínio, identificação da máquina e níveis de autenticação.

Domínio: Identifica o domínio a qual a máquina pertencerá, por meio da diretiva *workgroup*. Caso o SAMBA venha a atuar como PDC, as diretivas

¹⁵ Todos os caracteres à direita de ‘#’ e ‘;’ em uma linha do arquivo `smb.conf` são ignorados.


```

# Configuração global
[global]
    workgroup = APC
    netbios name = SGPO
    server string = Samba %v em %h
; hosts allow = 192.168.1. 192.168.2. 127.
; interfaces = 192.168.12.2/24 192.168.13.2/255.255.255.0
; remote browse sync = 192.168.3.25 192.168.5.255
; remote announce = 192.168.1.255 192.168.2.44
    socket options = TCP_NODELAY SO_RCVBUF=8192 SO_SNDBUF=8192
    printcap name = /etc/printcap
    load printers = yes
    printing = cups
; guest account = pcguest
    log file = /var/log/samba/%m.log
    max log size = 0
    security = user
; password server = <NT-Server-Name>
    encrypt passwords = yes
    smb passwd file = /etc/samba/smbpasswd
    unix password sync = Yes
    passwd program = /usr/bin/passwd %u
    passwd chat = *New*password* %n\n *Retype*new*password*
%n\n *passwd:*all*authentication*tokens*updated*successfully*
; username map = /etc/samba/smbusers
; local master = no
; os level = 33
; domain master = auto
; preferred master = auto
; domain logons = yes
; logon script = %m.bat
; logon script = %U.bat
; logon drive = z:
; logon path = \\%H\Profile
; wins support = yes
; wins server = w.x.y.z
; wins proxy = yes
    dns proxy = no
; preserve case = no
; short preserve case = no
; default case = lower
; case sensitive = no

```

Figura 4.5: Arquivo `smb.conf` – seção global.

os level, *preferred master*, *local master*, *domain master* e *domain logons* deverão ser utilizadas. Essas diretivas servem para que o SAMBA invoque uma “eleição” para definição do mestre de domínio local e saia vencedor. Maiores detalhes podem ser obtidos em [JÚNIOR (2000)].

Identificação da máquina: Identifica o nome NetBIOS primário do servidor por meio da diretiva *netbios name*. Caso não seja ajustado, ele usará o *hostname* da máquina como valor padrão.

```

# Compartilhamentos
[homes]
    comment = Diretório Pessoal
    browseable = no
    writable = yes
[printers]
    comment = Impressoras
    path = /var/spool/samba
    printable = yes
[tmp]
    comment = Arquivos temporários
    path = /tmp
    read only = no
    public = yes
[public]
    comment = Acesso Público
    path = /home/samba
    public = yes
    writable = yes
    write list = @manut
[CENP]
    comment = Centro de Planejamento
    path = /home/cenp
    valid users = @cenp

```

Figura 4.6: Arquivo `smb.conf` – compartilhamentos.

Nível de autenticação: Define a política de acesso, autenticação e segurança dos compartilhamentos, mediante senhas, por meio da diretiva *security*. Possui quatro níveis de autenticação:

- *share* – os serviços terão uma senha para cada recurso compartilhado, disponível para todos usuários;
- *user* – os usuários possuirão contas no sistema operacional e respectivas senhas. Essas contas serão utilizadas para liberação de acesso aos compartilhamentos. É a opção *default* do SAMBA.
- *domain* – as contas de usuários serão validadas em outro servidor PDC que irá prover controle de acesso para as máquinas que fazem parte do seu domínio;
- *server* – o SAMBA tentará autenticar o usuário em outro servidor NT (ou SAMBA). No caso da autenticação falhar, será usado o nível de acesso *user* na base de usuários local.

A segunda parte do arquivo de configuração é formada pelos nomes de compartilhamentos e opções específicas para cada um. A Tabela 4.2 apresenta as principais opções de compartilhamento do SAMBA.

Tabela 4.2: Opções de compartilhamento do SAMBA.

Opção	Descrição
<code>comment</code>	Descrição do compartilhamento.
<code>browseable</code>	Especifica se o compartilhamento será visualizado na lista de itens compartilhados. Também é reconhecido como <i>browsable</i> .
<code>writeable</code>	Habilita escrita no dispositivo. Possui <i>writable</i> como sinônimo.
<code>valid users</code>	Lista de usuários com permissão para utilização do recurso.
<code>create mode</code>	Também pode ser escrito <i>create mask</i> e especifica a permissão padrão dos arquivos criados.
<code>directory mode</code>	Idêntico ao <i>create mode</i> , porém aplicado a diretórios. Também pode ser escrito <i>directory mask</i> .
<code>path</code>	Caminho em disco do serviço compartilhado.
<code>guest ok</code>	Também pode ser escrito como <i>public</i> , habilita acesso público (sem senha) ao compartilhamento.
<code>printable</code>	Habilita impressão no compartilhamento.
<code>read only</code>	Especifica o compartilhamento como apenas de leitura.
<code>write list</code>	Lista os usuários que têm acesso de escrita e leitura ao compartilhamento.
<code>read list</code>	Lista os usuários que têm acesso apenas de leitura ao compartilhamento.

4.3.2 Configuração via SWAT

É possível realizar alterações no arquivo `smb.conf` remotamente, via *browser*, por meio da ferramenta SWAT, cuja configuração foi tratada por [TOSO (2003)]:

Para a configuração do SWAT, o primeiro passo é entender sua forma de execução. O SWAT executa a partir do *inetd*, ou para algumas distribuições, a partir do *xinetd*. Dessa maneira, temos que configurar o *inetd* para acessar o SWAT e transmitir a ele o controle da sessão. O *inetd* possui um arquivo de configuração em `/etc/inetd.conf`, que deve ser editado para o acréscimo da seguinte linha:

```
swat stream tcp nowait.400 root /usr/sbin/swat swat
```

Depois dessa modificação, o arquivo `/etc/services` também necessita atualizações. Acrescente ao mesmo a linha:

```
swat          901/tcp
```

Após a configuração, a ferramenta pode ser acessada a partir de qualquer navegador, utilizando `http://nome_ou_ip_do_servidor:901`. Para realizar as configurações do SAMBA será necessário logar-se como *root*, conforme Figura 4.7.



Figura 4.7: Tela de *login* da interface SWAT.

A Figura 4.8 apresenta a interface básica do SWAT.

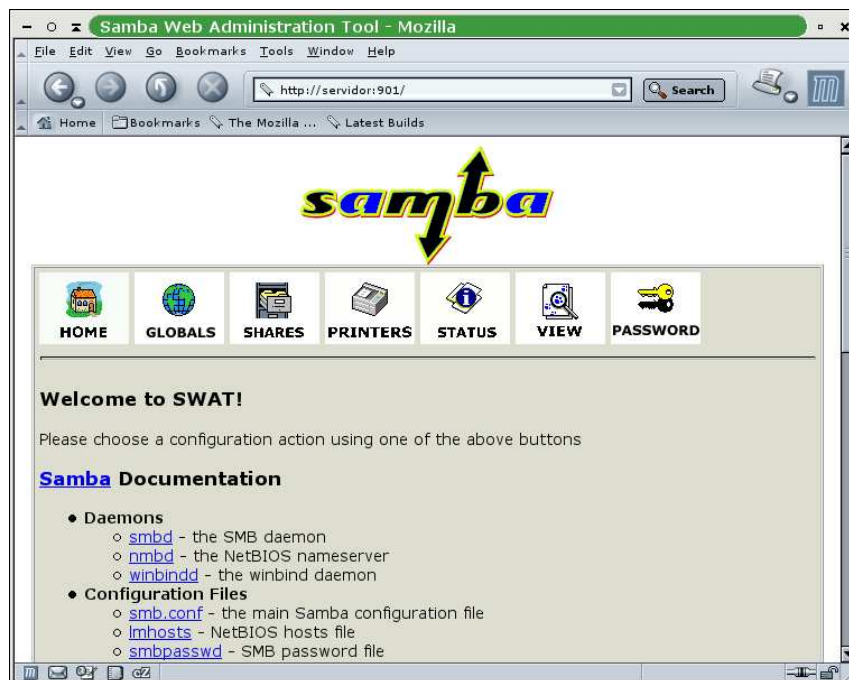


Figura 4.8: Administração do SAMBA via SWAT

4.4 Considerações finais

Existem vários outros itens e observações a serem destacadas sobre o SAMBA, suas seções e diretivas. O objetivo deste capítulo resume-se em apresentar os aspectos mais relevantes da ferramenta para subsidiar o estudo de caso tratado no Capítulo 5. Maiores informações sobre configuração do SAMBA podem ser obtidas nas referências bibliográficas constantes neste trabalho, como [SILVA (2003)], [LECHNYR (2004)], [JÚNIOR (2000)], entre outras.

Capítulo 5

Análise evolutiva

Este capítulo refere-se à evolução da rede de computadores da SCAO, no período compreendido entre janeiro de 1999 a dezembro de 2002.

5.1 Considerações iniciais

O autor desse trabalho ingressou na SCAO em novembro de 1998, exercendo a função de Auditor. Por possuir relativo conhecimento de Informática¹, entre as tarefas a desempenhar constava auxiliar na administração da rede de computadores da superintendência. Dessa forma, participou ativamente de todo o processo descrito neste capítulo.

5.1.1 O ambiente

Embora pertencesse à Secretaria de Estado da Fazenda (SEF), a SCAO mantinha endereço próprio. Sua rede de computadores estava ligada diretamente, via modem, à SEF, que provia conexões à internet, servidor de *e-mail* e acesso aos sistemas corporativos do Estado² por meio de emulação de terminais.

A Secretaria mantinha rígido controle sobre as comunicações entre as redes internas e a internet, além dos acessos aos sistemas corporativos sob sua responsabilidade³, mas não disponibilizava soluções de autenticação e controle de acesso aos arquivos em cada rede local.

¹ Conceitos básicos de Unix – adquiridos na operação de Xenix e AIX – e boa desenvoltura com Sistemas Operacionais Microsoft – DOS e Windows 9x.

² Os sistemas disponibilizados eram: SIAFI – Sistema Integrado de Administração Financeira; SISAP – Sistema de Administração de Pessoal; SICAF – Sistemas de Informação, Controle e Arrecadação Fiscal; SIPRO – Sistema de Protocolo.

³ Os sistemas integrados de arrecadação (SICAF) e de administração financeira (SIAFI) são administrados pela SEF.

5.1.2 Riscos imediatos

A SCAO, em 31 de dezembro de 1998, não possuía um servidor de arquivos em operação. Um servidor *Sun* havia sido entregue dias antes, mas não possuía Sistema Operacional – cuja instalação só seria feita no ano seguinte, no mês de fevereiro. A rede de computadores era composta por máquinas utilizando sistema operacional Windows 95. A cultura de utilização de arquivos em rede ainda não se desenvolvera. Dessa forma, os usuários utilizavam as estações como computadores pessoais, salvando seus papéis de trabalho e relatórios de auditoria localmente. Em cada estação com Windows 95 havia um diretório compartilhado, sem restrições, denominado *compartilha*, que era eventualmente utilizado quando se desejava disponibilizar uma determinada informação para todos os usuários. Esse quadro gerava diversos incômodos:

Disputa de usuários por estações: Pelo fato de trabalharem localmente, a informação, na maioria das vezes, ficava armazenada em um determinado computador, gerando disputas para utilização das máquinas por parte dos usuários.

Duplicidade de arquivos: Uma prática eventualmente adotada por alguns usuários era a cópia dos arquivos de uma máquina para outra. Esse fato, além de gerar duplicidade de informações e desperdício, poderia confundir o auditor, que necessitava lembrar em qual máquina havia feito as últimas alterações de seu trabalho.

Inviabilidade para *backup* centralizado: Não era possível manter um *backup* geral de todos os trabalhos; essa responsabilidade ficava à cargo de cada auditor. Como a maioria dos trabalhos de auditoria gera um volume considerável de informações, a cópia em disquete tornava-se uma solução demorada e inviável. Nesse tempo a SCAO não contava com gravadores de CD em suas máquinas.

Falta de controle de acesso: Não havia como controlar o acesso aos arquivos de trabalho. Mesmo que fosse colocada senhas nos computadores⁴, haveria necessidade de divulgação ampla, o que inviabilizava o controle de acesso.

Risco de perdas de informações: Havia a possibilidade dos dados serem apagados, propositalmente ou não, sem possibilidade de identificação do responsável.

Risco de alterações de informações: Havia possibilidade dos dados serem modificados por outros usuários. Caso uma modificação fosse percebida o autor não poderia ser identificado e responsabilizado.

⁴ Senhas no *setup* de cada máquina.

Alta demanda de impressão: Para resguardar e respaldar as conclusões dos relatórios de auditoria emitidos, todo papel de trabalho era impresso e arquivado.

5.1.3 O ponto de partida

Na tentativa de minimizar os riscos do ambiente, foi estabelecida uma política de utilização dos recursos computacionais disponíveis, elaborada pelo autor deste trabalho, que contou com a contribuição de auditores interessados na melhoria da performance do ambiente e o apoio incondicional das diretorias da SCAO⁵.

Essa política de utilização de recursos foi adotada em janeiro de 1999, e será o marco de referência para o início do estudo de caso da evolução do sistema de compartilhamento e autenticação da superintendência.

5.2 A era da incerteza

Considerando que não haveria compra de *software* para auxiliar na administração da rede interna, e que todos os sistemas operacionais deveriam permanecer instalados, foram idealizadas algumas ações na tentativa de minimizar os riscos detectados. Nessa etapa, não foi cogitada a adoção de *Software Livre*.

5.2.1 Proposta e implementação

Como o sistema operacional disponível para utilização era o Windows 95, foi estabelecida a centralização dos arquivos utilizando os recursos de compartilhamento nativos do sistema operacional.

Foram selecionados dois computadores para servirem de repositório de arquivos, cada qual com um compartilhamento para toda a rede protegido por senha. Um dos computadores, denominado *Equipe04* serviria de repositório de arquivos pessoais, por meio de uma pasta compartilhada com o mesmo nome da máquina; e o outro, denominado *Equipe06*, armazenaria os papéis de trabalho e relatórios finais, também em um compartilhamento com mesmo nome da estação. A Figura 5.1 representa a situação proposta. Para utilização das pastas compartilhadas o usuário conectava-se ao(s) computador(es) utilizando o recurso *Ambiente de Rede* do Windows.

Os objetivos dessa implementação eram: eliminar a disputa de estações por parte dos usuários; eliminar duplicidades de arquivos; desenvolver soluções alternativas de *backup*, uma vez que os dados estariam concentrados; criar cultura de

⁵ Conforme comentado na Seção 2.2.1, a SCAO não contava com profissionais formados na área de Tecnologia da Informação.



Figura 5.1: Equipes 4 e 6 utilizadas como servidores de arquivos.

trabalho armazenado em rede. Havia expectativa para instalação do sistema operacional na máquina *Sun* e a possibilidade de utilização desse equipamento como servidor de arquivos.

5.2.2 Resultados

Com a nova política de utilização, os usuários começaram a desenvolver cultura de utilização de redes de computadores, acessando com naturalidade o *Ambiente de Rede*. Entretanto, como não foram utilizadas restrições para o armazenamento de senhas⁶, em pouco tempo todas as estações de trabalho já haviam armazenado as senhas de acesso aos dois “servidores”.

O objetivo de eliminar duplicidades de arquivos foi atingido, com relação a cópias de arquivos em computadores. Os usuários adotaram a prática de armazenamento de arquivos apenas nos micros identificados para essa finalidade. Houve maior flexibilização da utilização dos computadores, transferindo a “disputa” para a totalidade de computadores disponíveis.

Um fato constatado, refere-se à utilização do compartilhamento *Equipe04*. Originalmente, a proposta para esse equipamento era armazenar arquivos pessoais dos auditores; para tanto, cada auditor possuía uma pasta com seu nome (conforme Figura 5.1). Entretanto, a maioria dos usuários passaram a armazenar todos os arquivos nesse computador, indiferente se eram arquivos pessoais (legislação de apoio, material de cursos, manuais de orientação, pesquisas, etc...) ou relativos a trabalho de auditoria. Dessa forma, o micro *Equipe06* era pouco utilizado.

Como consequência da centralização dos arquivos, houve possibilidade de implantação de uma política de *backup*: os arquivos dos “servidores” eram compactados em outros computadores, em horário determinado, oferecendo uma alternativa mínima de recuperação de dados.

Em fevereiro daquele ano, o sistema operacional Solaris foi instalado no servidor *Sun*⁷. O Solaris disponibiliza naturalmente serviço de compartilhamento de

⁶ O armazenamento de senhas (cache de senhas) do Windows pode ser desativado pela edição direta do Registro (*regedit*), ou nas opções da ferramenta *Poledit*.

⁷ Neste trabalho os termos *servidor Sun*, *servidor Solaris* e *servidor de arquivos* são equivalentes.

arquivos via NFS, mas a SEF não havia adquirido clientes NFS para Windows. Por essa razão, a máquina *Sun* passou a servir apenas como nova opção de *backup*, feito via conexão FTP e posterior cópia dos arquivos para fita magnética⁸.

5.2.3 Comentários

A política de utilização de recursos computacionais implantada serviu para despertar cultura de utilização de rede por parte dos usuários e proporcionou maior flexibilidade de utilização dos computadores, diminuindo “disputas” por parte dos usuários.

O objetivo de eliminar duplicidades de arquivos também foi atingido. Apesar da maioria dos usuários manterem seus arquivos no “servidor” *Equipe04* houve concentração em um único local, repercutindo na eliminação de cópias indiscriminadas de arquivos por vários computadores da rede.

Outro objetivo alcançado foi a possibilidade de criação de uma solução de *backup*, decorrente da centralização dos arquivos e da utilização do servidor Solaris.

Entretanto, alguns riscos identificados ainda persistiam:

- falta de controle de acesso;
- risco de perdas de informações – deleção intencional ou involuntária;
- risco de alterações de informações.

Aliados aos riscos, algumas situações deveriam ser consideradas:

- uma pasta “particular” era amplamente acessada pelos outros usuários;
- o processo de *backup* era lento e não automatizado: primeiro fazia-se a cópia compactada dos “servidores” Windows (*Equipe04* e *Equipe06*) para outras estações. Em seguida esses arquivos compactados eram transferidos via FTP para o servidor Solaris para, finalmente, ser realizada a cópia em fita⁹;
- não havia possibilidade de identificar o autor de um arquivo, nem quando e por quem determinado compartilhamento fora acessado – reinava ainda a “incerteza”;
- para resguardar os trabalhos de auditoria ainda era necessária alta demanda de impressão;
- a utilização do servidor Solaris permanecia muito aquém de suas potencialidades.

⁸ A unidade de fita magnética pertencia ao servidor *Sun* e não poderia ser utilizada nas estações, em função da incompatibilidade de *hardware* e do contrato assinado pela SEF.

⁹ A experiência mostrou que esse procedimento era menos problemático do que a cópia direta dos arquivos para o servidor Solaris.

Diante desse quadro, foi considerada a utilização de um *Software* Livre para dotar de maior funcionalidade o servidor Solaris e possibilitar algum controle de acesso. A ferramenta escolhida foi o SAMBA, em função da sua capacidade de promover compartilhamento de arquivos localizados no Solaris para as estações Windows. A instalação e a utilização do SAMBA marcam o início da “era do compartilhamento” na SCAO.

5.3 A era do compartilhamento

Em dezembro de 1999 foi instalada a versão 1.9.9 do SAMBA, a mais recente e estável para o Solaris naquela ocasião.

Havia expectativa que o *software* realizasse efetivo controle no acesso aos arquivos da superintendência. Entretanto, por desconhecimento técnico, só fora possível realizar compartilhamento, de forma transparente, utilizando a diretiva *security = share*. O SAMBA não era um *software* conhecido por nenhum dos funcionários da SCAO, incluindo o autor deste trabalho¹⁰. Embora as leituras apontassem para a possibilidade de autenticação de usuários com a diretiva *security = user*, os testes realizados foram infrutíferos. Posteriormente, com a correta compreensão e configuração das estações Windows, foi implementada a autenticação por usuário, marcando o início da “era do controle”.

5.3.1 Proposta e implementação

Com o armazenamento de arquivos no servidor Solaris e a utilização da diretiva *security = share* no SAMBA, foram concebidas novas estratégias para utilização dos recursos computacionais. Todos os arquivos disponibilizados nos micros *Equipe04* e *Equipe06* foram transferidos para o servidor, conforme Figura 5.2, e organizados da seguinte forma:

- foram criados compartilhamentos, com senha, para cada trabalho executado pelas equipes de auditoria – à medida que novos trabalhos fossem definidos, novas pastas seriam compartilhadas para os membros da equipe, cuja senha seria divulgada internamente;
- as diretorias teriam acesso a todos os trabalhos por outro compartilhamento com senha – pasta *Trabalhos de Auditoria*;
- foi criado um diretório específico para armazenar os arquivos pessoais dos auditores, sem utilização de senhas¹¹.

¹⁰ Nessa época, o autor mantinha interesse em utilizar o GNU/Linux pela sua ligação com o Unix, mas ainda não havia despertado interesse no movimento *Software* Livre.

¹¹ Optou-se por não utilizar senha nesse diretório para evitar confusões na utilização de compartilhamentos distintos. Além disso, a utilização de uma senha comum a todos usuários descaracterizava sua finalidade.

Para reforçar a nova cultura de utilização de compartilhamentos, foram desabilitadas as opções de arquivamento de senhas das estações Windows.

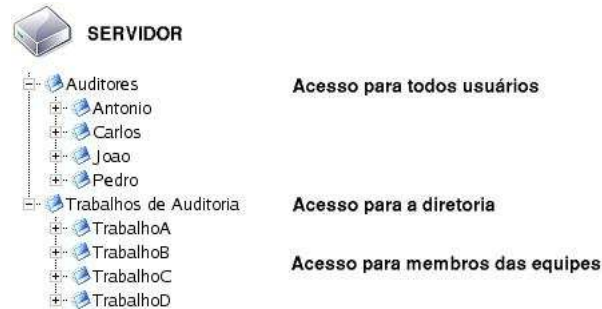


Figura 5.2: SAMBA utilizando diretiva *security = share*

Os objetivos dessa implementação eram:

- dar maior funcionalidade ao servidor de arquivos, uma vez que a referida máquina desempenhava apenas a função de receber os arquivos compactados via FTP, e gravá-los em fita;
- automatizar o processo de *backup*, em função da centralização dos arquivos no servidor Solaris;
- possibilitar maior controle de acesso aos trabalhos de auditoria;
- oferecer melhor disponibilidade dos arquivos, em função da capacidade de armazenamento e processamento da máquina aliada à estabilidade e robustez do sistema operacional Solaris.

5.3.2 Resultados

A centralização dos arquivos no servidor Solaris possibilitou um ganho considerável de produtividade no processo de *backup*, pois eliminou a necessidade de compactações e cópias entre estações e servidor.

Outro benefício gerado foi o salto qualitativo nos trabalhos de auditoria. Como a capacidade de armazenamento e processamento do servidor era muito superior às estações utilizadas inicialmente¹², tornou-se possível desenvolver pesquisas e análises em bases de dados copiadas das unidades auditadas. Essa nova possibilidade reduziu o tempo de conclusão dos trabalhos e promoveu significativamente a informática como ferramenta de auditoria.

¹²As estações utilizadas como servidores destacadas na “era da incerteza” possuíam processadores Pentium II 266MHz, contra o processamento de 450MHz do servidor.

Quanto aos compartilhamentos para as equipes, a prática demonstrou que nem todos usuários passaram a utilizá-los. A maioria dos auditores continuou armazenando seus arquivos de trabalho nas pastas pessoais, a exemplo do que ocorrera na “era da incerteza” com o micro *Equipe04*. Dessa forma, apesar da proposta de utilização de compartilhamentos com senha possibilitar melhoria no controle de acesso aos trabalhos de auditoria, esse objetivo não fora atingido, pela baixa adesão por parte dos usuários.

5.3.3 Comentários

A implantação do SAMBA serviu para promover a utilização do servidor Solaris, que até então permanecia subutilizado. A centralização dos arquivos reduziu a complexidade das rotinas de *backup* anteriormente estabelecidas. Aliado a esses fatores, a SCAO obteve melhoria na desenvoltura de seus trabalhos em função das novas possibilidades oferecidas pelo servidor de arquivos (estabilidade, poder de armazenamento e processamento) conseqüências da utilização de um sistema operacional robusto e estável, o Solaris, em conjunto com uma poderosa ferramenta de compartilhamento *Livre!* – o SAMBA.

Com relação ao controle de acesso aos trabalhos, a pouca adesão por parte dos usuários em utilizar o compartilhamento específico impediu a conclusão desse objetivo. Portanto, dos quatro objetivos estabelecidos três foram atingidos: funcionalidade do servidor de arquivos; automatização de *backup* e melhoria na disponibilização de arquivos.

Apesar da maioria dos objetivos terem sido alcançados, ainda restavam riscos que deveriam ser reduzidos: não havia efetivo controle de acesso; informações poderiam ser apagadas e/ou alteradas, intencional ou involuntariamente, sem possibilidade de identificação do usuários. Além de tratamento dos riscos identificados, duas questões relevantes mereciam maior atenção:

- pastas pessoais ainda eram amplamente acessadas e utilizadas para trabalhos em equipe;
- não era possível identificar autores de arquivos, nem qual usuário acessava determinado compartilhamento;

Portanto, apesar das vantagens destacadas e dos objetivos alcançados, restava ainda a necessidade de efetivo controle de acesso.

5.4 A era do controle

Conforme destacado na seção anterior, o SAMBA não era uma ferramenta completamente dominada pelos funcionários da SCAO. Após ter seu interesse despertado

para as questões relativas ao movimento *Software Livre*, e adotar sistema operacional GNU/Linux em seu micro doméstico, o autor deste trabalho iniciou, em janeiro de 2002, uma pesquisa para utilizar o SAMBA como servidor de autenticação para a rede interna da SCAO. Para tanto, realizou testes a partir de um equipamento que utilizava a versão 7.0 do sistema operacional Conectiva Linux¹³, rodando a versão 2.2.2 do SAMBA. Nesses experimentos verificou que as configurações das estações de trabalho Windows não foram feitas corretamente, à época da instalação do SAMBA no servidor Solaris.

Era chegada a hora de aproveitar a ferramenta em sua plenitude. Assim, em fevereiro de 2002, a versão 2.2.4 do SAMBA foi instalada no servidor, utilizando a diretiva *security = user*. Esse acontecimento marcou o início de uma nova política de utilização de recursos computacionais na SCAO: a era do controle.

5.4.1 Proposta e implementação

Com a utilização da diretiva *security = user*, no SAMBA, foram estabelecidas novas diretrizes para utilização da rede de computadores, na intenção de criar mecanismos de controle de acesso individuais:

1. O SAMBA foi configurado para ser o controlador de domínio (PDC) da rede interna, por meio da utilização das diretivas detalhadas na Figura 5.3.

```
[global]
  workgroup = SCAO
  netbios name = SERVIDOR

  domain logons = Yes
  os level = 65
  preferred master = Yes
  domain master = Yes
  local master = Yes
```

Figura 5.3: Diretivas para utilização do SAMBA como PDC.

2. As estações Windows foram configuradas para efetuarem *logon* no domínio SAMBA¹⁴ – Figura 5.4.
3. Foi criado um *login*, no Solaris, para cada funcionário da superintendência, utilizado pelo SAMBA para autenticação dos usuários no domínio especificado¹⁵.

¹³ Página oficial: <http://www.conectiva.com.br>.

¹⁴ No servidor, o caminho para execução do *script de logon* foi definido utilizando a diretiva *logon script*, no caso: *logon script = %U.bat*.

¹⁵ Essa sincronização entre SAMBA e o sistema operacional é obtida por meio da utilização da diretiva *smb passwd file*, no caso do Solaris e dos sistemas *nix: *smb passwd file = /etc/passwd*.

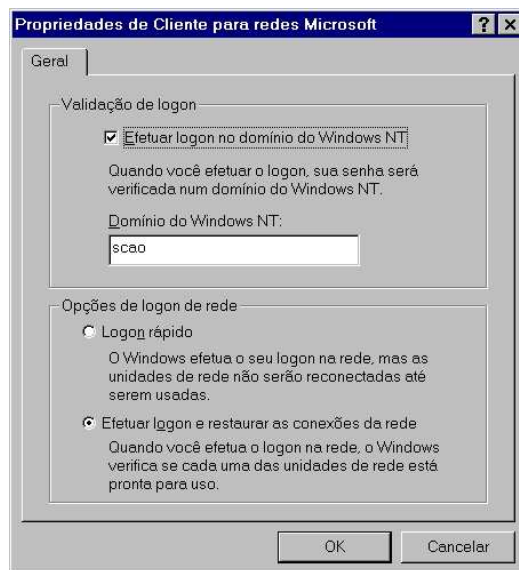


Figura 5.4: Definindo domínio nas estações Windows.

4. Foi criado um compartilhamento pessoal para cada usuário, inacessível para os demais, utilizando a seção *homes* do SAMBA.
5. Para cada trabalho de auditoria foi criado um compartilhamento específico, contendo a lista dos usuários que iriam acessá-lo. A Figura 5.5 apresenta um exemplo de um compartilhamento para uma equipe de 5 auditores, com acesso *somente leitura* para um usuário.

```
[Analit]
comment = Auditoria Analitica
path = /dados/scao/trabalhos/analit
username = albertav, monicagb, osmart, danielcr, charleas
read list = osmart
only user = Yes
root preexec = /adm/comandos/smbalog/smbcontrol %U %S %m %u %I INI
root postexec = /adm/comandos/smbalog/smbcontrol %U %S %m %u %I FIM
```

Figura 5.5: Exemplo de um compartilhamento para trabalho de auditoria.

6. Os compartilhamentos disponíveis para cada usuário seriam disponibilizados de forma automática por meio de scripts de inicialização, conforme exemplificado na Figura 5.6. A utilização desses *scripts* permitiu que os compartilhamentos fossem mapeados pelas estações Windows e “montados” em um *drive* específico, por exemplo G, conforme Figura 5.7.


```

net time \\servidor /set /yes -y

net use G: \\servidor\homes -y
net use H: \\servidor\coorden -y
net use I: \\servidor\consulta -y
net use J: \\servidor\util -y
net use P: \\servidor\analit -y
net use X: \\servidor\temp -y

```

Figura 5.6: Script de inicialização de um usuário.

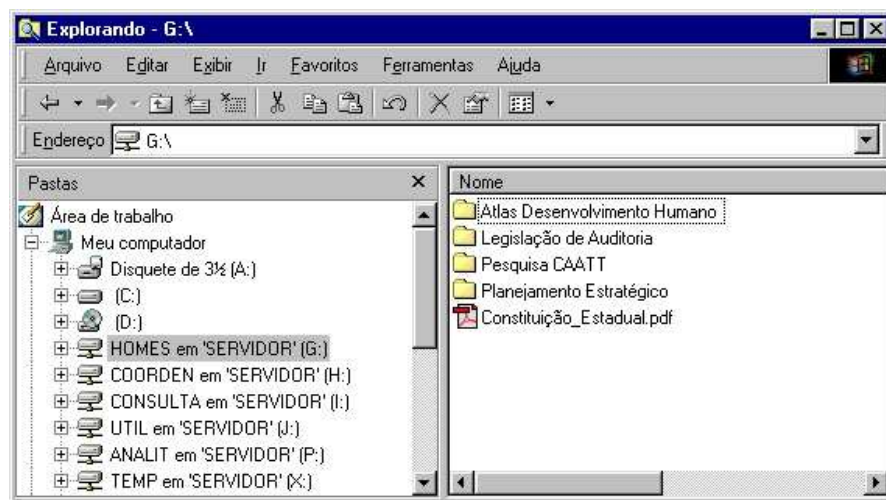


Figura 5.7: Compartilhamentos mapeados em clientes Windows.

7. Foram criadas rotinas para monitoramento de acesso aos compartilhamentos. Essas rotinas consistiam em *shell scripts* executadas a partir do servidor, possibilitando registro de:

- *login* do usuário solicitante;
- nome *NetBIOS* da máquina de onde partiu a solicitação;
- endereço IP da máquina de onde partiu a solicitação;
- data e hora do acesso ao compartilhamento.

O Apêndice C apresenta as rotinas utilizadas para estabelecimento de registro de acessos efetuados.

As implementações tiveram como objetivos realizar controle de acesso individual aos compartilhamentos, incluindo registro de acessos e promover a utilização das pastas destinadas aos trabalhos de auditoria. Dessa forma, procurou-se minimizar os riscos apresentados na Seção 5.1.2.

5.4.2 Resultados

A utilização da diretiva *security = user* serviu para promover controle de acesso por usuário, eliminando possíveis confusões de utilização de várias senhas para acessos distintos. Bastava que o usuário efetuasse *logon* (Figura 5.8) para que todos os compartilhamentos, a ele autorizados, fossem automaticamente disponibilizados.

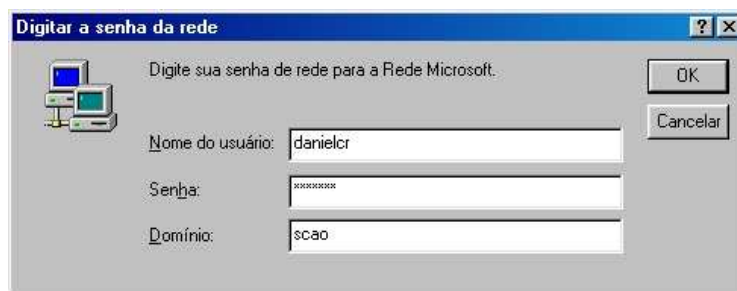


Figura 5.8: Tela de login das estações Windows.

As pastas pessoais, ganharam finalmente características de acesso exclusivo. Os arquivos de trabalho, por sua vez, passaram a ser armazenados nas pastas correspondentes, e acessados apenas por membros das equipes, eliminando riscos de deleção e alteração de informações por outros usuários.

A utilização das rotinas de monitoramento (*shell scripts*) promoveram registros preciosos para identificar data e hora de acesso aos compartilhamentos, por usuário e máquina. Como consequência, a responsabilização pelos papéis de trabalho em meio magnético foi amplamente promovida e o número de impressões reduzido.

5.4.3 Comentários

A nova dinâmica de operação da rede de computadores promoveu mudanças significativas na utilização do servidor de arquivos e na forma de trabalho das equipes de auditoria. Foi implantando um sistema eficiente de controle de acesso aos compartilhamentos definidos para cada usuário; arquivos de trabalho passaram a ser acessados apenas por membros das equipes de auditoria; pastas pessoais passaram a ter acesso exclusivo e; rotinas de monitoramento possibilitaram responsabilizar usuários por alterações efetuadas nos papéis de trabalho.

Dessa maneira, a utilização do SAMBA permitiu estabelecer mecanismos de controle de acesso aos arquivos armazenados em meio magnético, garantindo confiabilidade, integridade, confidencialidade e responsabilização, alcançando os objetivos estabelecidos.

5.5 Considerações finais

Este capítulo teve como objetivo realizar uma análise evolutiva da utilização da rede de computadores da SCAO, no período compreendido entre janeiro de 1999 e dezembro de 2002, com ênfase na adoção da ferramenta SAMBA como solução para o problema de compartilhamento de arquivos e autenticação de usuários.

5.5.1 Análise dos resultados

A evolução partiu de uma situação de insegurança, com duplicidades e armazenamentos indiscriminados de arquivos, risco de perdas e alterações de informações sem possibilidade de identificação de autores, e alcançou níveis efetivos de controle de acesso, com autenticação de usuários, compartilhamentos definidos por equipes e possibilidade de rastreamento – trilhas de auditoria.

Apesar de não possuir corpo técnico especializado na área de Tecnologia da Informação, a superintendência foi capaz de promover melhorias para incrementar segurança na sua rede interna. Isso tornou-se possível em função da utilização de *Software Livre*, no caso o SAMBA, que mostrou-se bastante estável, de fácil configuração e com abundante suporte disponível na Internet¹⁶, permitindo estabelecer mecanismos de controle de acesso aos dados armazenados sem desembolso financeiro. Aliado a isso, os fatores destacados na Subseção 2.5 não se caracterizaram em impedimentos: os sistemas operacionais foram mantidos, uma vez que o SAMBA realizou integração de forma transparente entre o Solaris e as estações Windows, e os princípios basilares da Administração Pública foram respeitados. Destaca-se que a argumentação apresentada na Seção 3.5, foi comprovada nesta experiência:

- não houve necessidade de contratação de fornecedor para implementar a solução – princípio da Impessoalidade aplicado;
- os resultados obtidos foram os melhores possíveis, sem desembolso financeiro para aquisição de licença do *software* ou atualizações de hardware – princípio da Eficiência aplicado;
- com relação ao princípio da Razoabilidade, além dos itens anteriormente destacados, temos:
 - a tecnologia utilizada permaneceu sob domínio da SCAO;
 - o desenvolvimento do SAMBA é orientado para benefícios de seus usuários, não de um fabricante;
 - a ferramenta apresentou robustez e segurança na sua utilização.

¹⁶ As dificuldades encontradas para utilização da diretiva *security = user*, mencionadas na Seção 5.3, foram decorrentes de configuração incorreta das estações Windows.

Portanto, a adoção do SAMBA permitiu solucionar o problema motivador deste trabalho, demonstrando eficiência e facilidade de operação.

5.5.2 Novas perspectivas

Com a promulgação da Lei Delegada 92, de 29 de janeiro de 2003, a SCAO passou integrar a Auditoria Geral do Estado de Minas Gerais, desvinculando-se da Secretaria de Estado da Fazenda [MINAS GERAIS (2003)]. A nova estrutura da AUGE, destacada na Subseção 2.1.1, conjugou culturas organizacionais, metodologias e tecnologias diferentes. Além disso, o órgão não contou, em 2003, com recursos orçamentários suficientes para custeio de suas atividades. Por essa razão as estruturas originais das várias superintendências agregadas foram mantidas. Por conseqüência, foram utilizados dois servidores de arquivos para duas redes distintas: um servidor Windows NT atendendo a "rede" do Gabinete, Assessoria Técnica e Superintendência de Planejamento, Gestão e Finanças; e o SAMBA atendendo a "rede" das Superintendências Centrais de Auditoria de Gestão (SCAG) e Operacional (SCAO). A Superintendência Central de Correição Administrativa não contou com um servidor de arquivos e autenticação em 2003.

Considerando a natureza pública da AUGE, a importância das informações armazenadas, os benefícios decorrentes da utilização de padrões abertos e a experiência positiva resultante da instalação do SAMBA na SCAO, a adoção de *Software Livre* torna-se uma excelente alternativa para realizar a integração dos computadores das diversas superintendências e manter controle de acesso aos arquivos. Nesse sentido, a utilização de sistema operacional GNU/Linux nos servidores e estações de trabalho é fortemente recomendada, em função de suas características de segurança, controle e estabilidade nativas¹⁷. O Apêndice D apresenta relação das principais distribuições GNU/Linux disponíveis no mercado.

Juntamente com GNU/Linux, existem soluções *Livres!* que podem substituir com igual ou superior performance os modelos proprietários. Dessa forma, a AUGE poderá adotar novas soluções para aplicativos de escritório (editores de texto, planilhas eletrônicas, programas de apresentação), a exemplo da suíte OpenOffice¹⁸; gerenciadores de banco de dados, como PostgreSQL¹⁹; navegadores internet, como Mozilla²⁰ e Galeon²¹; clientes de *e-mail*, como o Balsa²² ou o próprio Mozilla; entre as inúmeras interfaces gráficas disponíveis, sem perda de qualidade²³.

¹⁷ Por ser multiusuário, o sistema disponibiliza automaticamente mecanismos de proteção individuais ou por grupos, além de oferecer ampla possibilidade de composição de trilhas de auditoria.

¹⁸ <http://www.openoffice.org>.

¹⁹ <http://postgresql.org>.

²⁰ <http://www.mozilla.org>.

²¹ <http://galeon.sourceforge.net>.

²² <http://balsa.gnome.org>.

²³ [BACIC (2003)] apresenta uma tabela de similaridade entre programas Windows e seus equiva-

[ARAÚJO (2003)] realizou um estudo comparativo para determinar a melhor escolha, no quesito *confiabilidade e segurança*, entre três “conjuntos” de sistemas operacionais e pacotes de escritório, conjugados da seguinte forma: sistema operacional proprietário e pacote de escritório proprietário; sistema operacional proprietário e pacote de escritório aberto e; sistema operacional aberto e pacote de escritório aberto. Para tanto utilizou três soluções combinadas: [1] sistema operacional Microsoft Windows e pacote de escritório Microsoft Office; [2] sistema operacional Windows e pacote de escritório Open Office; [3] sistema operacional GNU/Linux e pacote de escritório Open Office. Utilizando modelo de comparação AHP²⁴, o estudo apresentou o seguinte resultado:

[...] a literatura pesquisada tem evidenciado que sistemas abertos são, em geral, menos vulneráveis que sistemas proprietários, especialmente no caso dos sistemas operacionais Linux e Windows, conforme aqui exposto. No que tange os pacotes de escritório não foi encontrado evidência que distinga as opções aqui testadas, de forma que o sistema operacional se mostra decisivo para se determinar a segurança e a confiabilidade de uma estação de trabalho. Assim, considera-se a Opção 3 [Linux + OpenOffice], por contar com sistema operacional Linux, mais segura que as Opções 1 [Windows + Microsoft Office] e 2 [Windows + OpenOffice] [ARAÚJO (2003)].

O referido estudo corrobora a tese da adoção de *Software Livre* para promover segurança das informações armazenadas, sendo portanto, perfeitamente aplicável à AUGÉ.

Em casos extremos, como aplicativos específicos desenvolvidos para plataforma Windows, pode-se utilizar a solução Wine²⁵, que permite execução alternativa em ambientes Linux, Unix ou Solaris.

Além das opções destacadas, a adoção de GNU/Linux possibilitará aproveitamento de maquinário de baixa capacidade de armazenamento e processamento, por meio de redirecionamento da saída gráfica de máquinas mais robustas. Mais detalhes sobre essa funcionalidade podem ser obtidos em [X.ORG (2004)].

Quanto à segurança das informações, a adoção de GNU/Linux possibilitará a utilização de ferramentas de verificação de integridade de arquivos, contribuindo para promover fidedignidade dos papéis de trabalho armazenados no servidor. Maiores detalhes sobre ferramentas de verificação de integridade de arquivos podem ser obtidas em [DOMINGUES (2003)].

lentes em GNU/Linux.

²⁴ O método AHP *Analytic Hierarchy Process* foi desenvolvido por Thoma L. Saaty é um modelo sistemático para comparação de opções a partir de uma lista de critérios. Maiores informações podem ser obtidas em [EHRlich (1996)].

²⁵ <http://www.winehq.com>.

Pelos argumentos apresentados, a adoção de *Software* Livre pela AUGE servirá para realizar integração entre as Superintendências e promover segurança e confiabilidade nos trabalhos, vindo a marcar o início de nova etapa evolutiva em toda sua estrutura: a "era da qualidade".

Capítulo 6

Conclusão

A Informação, cada vez mais, se consolida como um elemento estratégico, altamente valioso, que precisa ser cuidadosamente administrado no âmbito das corporações. Nesse contexto, existe preocupação constante em se desenvolver mecanismos de autenticação de usuários de computadores e controle de acesso aos dados armazenados.

Para tanto, existem soluções tecnológicas disponíveis, caracterizadas *Software Livre*, que permitem liberdade de utilização, aprimoramentos e redistribuição. Entre essas soluções, a ferramenta SAMBA permite estabelecer autenticação de usuários e controle de acesso aos arquivos, além de integrar diferentes sistemas operacionais.

O estudo de caso da utilização do SAMBA como solução para autenticação de usuários e controle de acesso aos arquivos da Auditoria Geral do Estado de Minas Gerais – AUGE, possibilitaria uma avaliação da aplicabilidade da ferramenta, vindo a servir de referência para adoção de plataformas livres na Administração Pública.

Dessa forma, foi apresentado o modelo organizacional da AUGE, resultado de incorporações e desmembramentos de outras estruturas, e as peculiaridades decorrentes de sua formação. O ambiente de análise foi definido na Superintendência Central de Auditoria Operacional, pelo seu histórico de utilização da ferramenta SAMBA.

O ambiente de análise apresentou algumas considerações de ordem operacional e legal. Os aspectos operacionais caracterizaram-se em limitações de ordem técnica, administrativa e financeira. Os aspectos legais referiam-se aos princípios basilares da Administração Pública: Legalidade, Impessoalidade, Moralidade, Publicidade, Eficiência e Razoabilidade; que deveriam ser observados e diretamente aplicados.

O presente trabalho realizou, ainda, uma discussão sobre a relevância das informações armazenadas nos computadores da SCAO terminando por definir o problema motivador deste trabalho: o estabelecimento de mecanismos de controle

de acesso aos arquivos da rede interna da SCAO de forma a garantir confidencialidade, confiabilidade e integridade das informações, levando em conta todo contexto apresentado.

Para subsidiar as argumentações acerca da adoção de *Software* Livre, foi realizada uma apresentação do termo, sua conceituação e a filosofia na qual está inserido. Foi traçado um comparativo com o modelo proprietário, ressaltando que o ponto chave da filosofia do *Software* Livre permanece centrado na publicação do código fonte.

Essa discussão compreendeu também uma breve descrição histórica do movimento *Software* Livre com ênfase no processo de elaboração baseado na ação colaborativa, definida como “modelo bazar” em contraposição ao modelo tradicional, definido como “modelo catedral”. Foram apresentados argumentos que permitiram concluir que a metodologia bazar, empregada na criação de *Software* Livre, tem como resultado aplicativos de alta qualidade, em curto espaço de tempo e com rápida detecção e correção de falhas, devido à divulgação do código fonte para uma comunidade de programadores e colaboradores cada vez mais pró-ativa.

Por consequência, a adoção de *Software* Livre possibilita estabelecimento de vantagens estratégicas no ambiente corporativo: redução do poder de fornecedores, redução de custos, diferenciação de produtos ou serviços, com maior segurança e confiabilidade na execução de processos. Na Administração Pública, além das vantagens assinaladas, os princípios constitucionais da Impessoalidade, Razoabilidade e Eficiência constituem-se sólidos argumentos para que o Estado priorize adoção de ferramentas livres, visando melhoria na qualidade dos serviços prestados e promoção de desenvolvimento tecnológico e social.

Para servir de referência ao estudo de caso, foi apresentada a ferramenta SAMBA, suas características e informações referentes a instalação e configuração básica.

Diante dos elementos apresentados, realizou-se uma análise evolutiva da política de utilização dos recursos computacionais da SCAO. Partiu-se, em 1999, de um contexto com alto risco de perdas e alterações de informações. O estudo de caso comprovou que a adoção do SAMBA proporcionou estabelecimento de mecanismos que garantiram confiabilidade, integridade, confidencialidade das informações bem como realização de auditorias. Por essa razão, o problema motivador dessa monografia foi adequadamente solucionado.

Diante do exposto, o referido estudo de caso corrobora as conclusões apresentadas sobre as vantagens estratégicas obtidas por meio da utilização de *Software* Livre nas corporações, especialmente na Administração Pública. Em decorrência dessa constatação, a AUGE poderá promover a integração de todos os computadores das diversas superintendências por meio da utilização de *Software* Livre, proporcionando segurança e confiabilidade no acesso às informações.

Em trabalhos futuros, pretende-se desenvolver estudos específicos sobre utilização de *Software* Livre nas estações de trabalho, especialmente sistemas opera-

cionais GNU/Linux, sob a ótica da produtividade e desenvoltura por parte dos usuários. Além disso, pretende-se retomar discussão sobre gerenciamento estratégico da informação, e realizar estudo sobre gestão do conhecimento, no ambiente colaborativo do movimento *Software Livre*, a exemplo da comunidade Linux.

Referências Bibliográficas

- [ABRUCIO (1997)] ABRUCIO, Fernando Luiz. *O impacto do modelo gerencial na Administração Pública*. Brasília, Cadernos ENAP no. 10, 1997.
- [ARAÚJO (2003)] ARAÚJO, Wagner Frederico Gomes de. *O uso de Softwares de Código Aberto e Softwares Livres na Administração Pública como uma opção gerencial: elaborando um modelo de decisão baseado em múltiplos critérios*. Belo Horizonte, Fundação João Pinheiro, 2003. Monografia de conclusão do Curso Superior em Administração Pública.
- [BACIC (2003)] BACIC, Nicolas Michel. *O software livre como alternativa ao aprisionamento tecnológico imposto pelo software proprietário*. Disponível em <http://www.rau-tu.unicamp.br/nou-rau/softwarelivre>. Último acesso: 17 de fevereiro de 2004.
- [BRASIL (1988)] BRASIL. *Constituição da República Federativa do Brasil, 1988*. Disponível em <http://legis.senado.gov.br/con1988>. Último acesso: 6 de janeiro de 2004.
- [CARVALHO FILHO (2003)] CARVALHO FILHO, José dos Santos. *Manual de Direito Administrativo*. Rio de Janeiro, Lumens Juris, 10a. ed. 2003.
- [CFC (1997)] CONSELHO FEDERAL DE CONTABILIDADE. *NBC T 11 - Normas de Auditoria Independente das Demonstrações Contábeis*. Disponível em <http://www.cfc.org.br/uparq/legislacao/nbct11.PDF>. Último acesso: 29 de janeiro de 2004.
- [CISNEIROS (2003)] CISNEIROS, Hugo. *Introdução ao Linux*. Disponível em http://tln.conectiva.com.br/intro_linux. Atualizado em 04/02/2003. Último acesso em 27 de janeiro de 2004.
- [DAVENPORT (2003)] DAVENPORT, Thomas H. *Ecologia da Informação*. São Paulo, Futura, 6a. ed. 2003.
- [DI PIETRO (2001)] DI PIETRO, Maria Sylvia Z. *Direito Administrativo*. São Paulo, Atlas, 13a. ed. 2001.

- [DIBONA; OKCMAN; STONE (1999)] DIBONA, Chris; OCKMAN, Sam; STONE, Mark. *Open Sources. Voices from the Open Source Revolution*. O'Reilly & Associates, 1999.
- [DIMANTAS (2003)] DIMANTAS, Hernani. Parangolé Brasil, in SILVEIRA, Sérgio Amadeu da; CASSINO, João (Org.) *Software livre e inclusão digital* São Paulo, Conrad Livros, 2003. pp 329-39.
- [DOMINGUES; SCHNEIDER; UCHÔA (2001)] DOMINGUES, Marcos Aurélio; SCHNEIDER, Bruno de Oliveira; UCHÔA, Joaquim Quinteiro. Autenticação de Sistemas Linux usando OpenLDAP. In: *SEMAC2001 – XII Semana da Computação – IV Workshop em Linux, Internet e Aplicações*. São José do Rio Preto: UNESP, 2001. Disponível em <http://www.comp.ufla.br/~joukim/extensao>. Último acesso: 6 de fevereiro de 2004.
- [DOMINGUES (2003)] DOMINGUES, Marcos Aurélio. *Comparação de Ferramentas de Verificação de Integridade de Arquivos*. Lavras, UFLA, 2003. Monografia de conclusão do curso de Especialização em Administração de Redes Linux.
- [EHRlich (1996)] EHRlich, Pierre Jacques. *Modelos quantitativos de apoio às decisões*. RAE - Revista de Administração de Empresas, v.36, n.1 e 2, 1996.
- [FERRAZ (2002)] FERRAZ, Nelson Corrêa T. *Vantagens Estratégicas do Software Livre para o Ambiente Corporativo*. São Paulo, PUC-SP, 2002. Monografia de conclusão do curso Master Business Information Systems.
- [FSF (2001a)] FREE SOFTWARE FOUNDATION. *Licenças de Software Livre*. Disponível em <http://www.gnu.org/licenses/licenses.pt.html>. Atualizado em 15/09/2001. Último acesso em 15 de janeiro de 2004.
- [FSF (2001b)] FREE SOFTWARE FOUNDATION. *Várias Licenças e Comentários sobre elas*. Disponível em <http://www.gnu.org/licenses/license-list.pt.html>. Atualizado em 31/07/2001. Último acesso em 15 de janeiro de 2004.
- [FSF (2003)] FREE SOFTWARE FOUNDATION. *O que é Software Livre*. Disponível em <http://www.gnu.org/philosophy/free-sw.pt.html>. Atualizado em 13/07/2003. Último acesso em 11 de janeiro de 2004.
- [FSF (2003a)] FREE SOFTWARE FOUNDATION. *GNU General Public License*. Disponível em <http://www.gnu.org/copyleft/gpl.html>. Atualizado em 26/05/2003. Último acesso em 15 de janeiro de 2004.

- [FSF/UNESCO (2004)] FREE SOFTWARE FOUNDATION & UNITED NATIONS EDUCATIONS, SCIENTIFIC AND CULTURAL ORGANIZATION. *FSF/UNESCO Free Software Directory*. Disponível em <http://www.gnu.org/directory>. Atualizado em 13/01/2004. Último acesso em 14 de janeiro de 2004.
- [GATES (1976)] GATES III, William Henry. *An Open Letter do Hobbysts*. Disponível em <http://www.blinkenlights.com/classiccmp/gateswhine.html>. Último acesso em 12 de janeiro de 2004.
- [HEXSEL (2002)] HEXSEL, Roberto André. *Propostas de Ações de Governo para Incentivar o Uso de Software Livre*. Curitiba, UFPR 2002. Relatório Técnico RT-DINF 004/2002. Disponível em <http://www.inf.ufpr.br/~roberto>. Último acesso em 12 de janeiro de 2004.
- [JÚNIOR (2000)] JÚNIOR, Décio. *Samba. Guia de Consulta Rápida*. São Paulo, Novatec, 2000.
- [LECHNYR (2004)] LECHNYR, David. *The Unofficial Samba HOWTO*. Disponível em <http://hr.uoregon.edu/davidrl/samba>. Último acesso: 4 de fevereiro de 2004.
- [McFARLAN (1998)] McFARLAN, F. Warren. A Tecnologia da Informação Muda a Sua Maneira de Competir, in MONTGOMERY, Cyntia A.; PORTER, Michael E. (Org.). *Estratégia: A Busca da Vantagem Competitiva*. Rio de Janeiro, Campus, 1998.
- [McGEE; PRUSAK (1995)] McGEE, James; PRUSAK, Laurence. *Gerenciamento Estratégico da Informação*. Rio de Janeiro, Campus, 1995.
- [MEIRELLES (1996)] MEIRELLES, Hely Lopes. *Direito Administrativo Brasileiro*. Rio de Janeiro, Malheiros Editores, 21a. ed. 1996.
- [MINAS GERAIS (1985)] MINAS GERAIS. *Lei Delegada no. 6, de 28 de julho de 1985*. Dispõe sobre a estrutura orgânica da Administração Estadual, estabelece normas para modernização institucional, e dá outras providências. Disponível em <http://www.almg.gov.br/njmg/dirinjmg.asp>. Último acesso: 30 de dezembro de 2003.
- [MINAS GERAIS (1989)] MINAS GERAIS. *Constituição do Estado de Minas Gerais*. Disponível em <http://www.almg.gov.br/coes/coes.asp>. Último acesso: 6 de janeiro de 2004.

- [MINAS GERAIS (1995)] MINAS GERAIS. *Decreto 36825, de 27 de abril de 1995*. Dispõe sobre a Auditoria Geral do Estado. Disponível em <http://www.almg.gov.br/njmg/dirinjmg.asp>. Último acesso: 30 de dezembro de 2003.
- [MINAS GERAIS (1996)] MINAS GERAIS. *Lei 12159, de 27 de maio de 1996*. Altera a estrutura orgânica da Secretaria de Estado de Recursos Humanos e Administração, extingue o Instituto Estadual de Desenvolvimento de Recursos Humanos – IEDRHU – e dá outras providências. Disponível em <http://www.almg.gov.br/njmg/dirinjmg.asp>. Último acesso: 30 de dezembro de 2003.
- [MINAS GERAIS (1998)] MINAS GERAIS. *Lei 12984, de 30 de julho de 1998*. Dispõe sobre o Sistema Estadual de Finanças e a estrutura orgânica da Secretaria de Estado da Fazenda e dá outras providências. Disponível em <http://www.almg.gov.br/njmg/dirinjmg.asp>. Último acesso: 30 de dezembro de 2003.
- [MINAS GERAIS (2003)] MINAS GERAIS. *Lei Delegada 92, de 29 de janeiro de 2003*. Dispõe sobre o Sistema Estadual de Auditoria Interna, reestrutura a Auditoria Geral do Estado e dá outras providências. Disponível em <http://www.almg.gov.br/njmg/dirinjmg.asp>. Último acesso: 30 de dezembro de 2003.
- [MORIMOTO (2003)] MORIMOTO, Carlos E. *Dicionário Termos Técnicos de Informática*. 3a. ed. – versão e-book. Disponível em <http://www.guiadohardware.net>. Último acesso: 8 de fevereiro de 2004.
- [MOTTA (1999)] MOTTA, Carlos Pinto Coelho (Coord.). *Curso Prático de Direito Administrativo*. Belo Horizonte, Del Rey, 1999.
- [NEMETH (2002)] NEMETH, Evi et al. *Manual de Administração do Sistema UNIX®*. Bookman, 3a. ed. 2002.
- [NONAKA; TAKEUCHI (1997)] NONAKA, Ikujiro; TAKEUCHI, Hirotaka. *Criação de Conhecimento na Empresa*. Rio de Janeiro. Campus, 3a. ed. 1997.
- [PERENS (1999)] PERENS, Bruce. The Open Source Definition, in DIBONA, Chris; OCKMAN, Sam; STONE, Mark. *Open Sources. Voices from the Open Source Revolution*. O’Reilly & Associates, 1999. pp. 79-86.
- [PINHEIRO (2003)] PINHEIRO, Walter. A luta pelo Software Livre no Brasil, in SILVEIRA, Sérgio Amadeu da; CASSINO, João (Org.) *Software livre e inclusão digital*. São Paulo, Conrad Livros, 2003. pp 275-86.

- [PORTER (1986)] PORTER, Michael E. Porter. *Estratégia Competitiva*. Rio de Janeiro. Campus, 15a. ed. 1986.
- [PORTER (1992)] PORTER, Michael E. Porter. *Vantagem Competitiva*. Rio de Janeiro. Campus, 4a. ed. 1992.
- [RAYMOND (1998)] RAYMOND, Eric S. *The Cathedral and the Bazaar*. Disponível em <http://www.catb.org/~esr/writings/cathedral-bazaar>. Último acesso: 9 de janeiro de 2003.
- [RUSSEL; GANGEMI (1991)] RUSSEL, Debora; GANGEMI, G. T. *Computer Security Basics*. Sebastopol, O'Reilly & Associates, 1991.
- [SANDRONI (1989)] SANDRONI, Paulo (Org.). *Dicionário de Economia*. São Paulo, Best Seller, 1989.
- [SICA; UCHÔA (2003)] SICA, Fernando Cortez; UCHÔA, Joaquim Quinteiro. *Administração de Sistemas Linux*. Lavras, UFLA/FAEPE, 2003. Curso de Pós Graduação “Lato Sensu” (Especialização) à distância em Administração de Redes Linux.
- [SICA; UCHÔA; SIMEONE (2003)] SICA, Fernando Cortez; UCHÔA, Joaquim Quinteiro; SIMEONE Luiz Eduardo. *Administração de Redes Linux*. Lavras, UFLA/FAEPE, 2003. Curso de Pós Graduação “Lato Sensu” (Especialização) à distância em Administração de Redes Linux.
- [SILVA (2002)] SILVA, José Afonso da. *Curso de Direito Constitucional Positivo*. São Paulo, Malheiros, 2002.
- [SILVA (2003)] SILVA, Gleydson Maziolli. *Guia Foca Linux. Versão Avançado – 6.29*. Disponível em <http://focalinux.cipsga.org.br>. Último acesso: 30 de dezembro de 2003
- [SILVEIRA; CASSINO (2003)] SILVEIRA, Sérgio Amadeu da; CASSINO, João (Org.) *Software livre e inclusão digital*. São Paulo, Conrad Livros, 2003.
- [STAIR (1998)] STAIR, Ralph M. *Princípios de Sistemas de Informação. Uma abordagem gerencial*. Rio de Janeiro, LTC, 2a. ed. 1998.
- [STALLMAN (2003)] STALLMAN, Richard. *Linux e o Projeto GNU*. Disponível em <http://www.gnu.org/gnu/linux-and-gnu.pt.html>. Atualizado em 21/11/2003. Último acesso em 12 de janeiro de 2004.
- [STATO FILHO (2002)] STATO FILHO, André. *Domínio Linux. Do Básico a Servidores*. São Paulo, Visual Books, 2002.

- [TOSO (2003)] TOSO, Rodrigo Franco. *Mini Curso: Samba*. Universidade Federal de Lavras, Encontro técnico de Pós-Graduação em Administração de Redes Linux - ARL. 2003. Disponível em <http://www.comp.ufla.br/~rtoso/pages/publicacoes.html>. Último acesso: 19 de janeiro de 2004.
- [TS; ERKSTEIN; COLLIER-BROW (2003)] TS, Jay; ERKSTEIN, Robert; COLLIER-BROW, David. *Using Samba*. O'Reilly & Associates, 2ed. 2003.
- [UCHÔA (2003a)] UCHÔA, Kátia Cilene Amaral. *Introdução à Cibercultura*. Lavras, UFLA/FAEPE, 2003. Curso de Pós Graduação "Lato Sensu" (Especialização) à distância em Administração de Redes Linux.
- [UCHÔA (2003b)] UCHÔA, Joaquim Quinteiro. *Segurança em Redes e Criptografia*. Lavras, UFLA/FAEPE, 2003. Curso de Pós Graduação "Lato Sensu" (Especialização) à distância em Administração de Redes Linux.
- [VALOIS (2003)] VALOIS, Djalma. Copyleft, in SILVEIRA, Sérgio Amadeu da; CASSINO, João (Org.) *Software livre e inclusão digital* São Paulo, Conrad Livros, 2003. pp 287-317.
- [X.ORG (2004)] X.ORG. *About the X Window System*. Disponível em <http://www.x.org/X11.html>. Último acesso: 18 de janeiro de 2004.

Apêndice A

Lei Delegada 92, de 29/01/2003

Dispõe sobre o Sistema Estadual de Auditoria Interna, reestrutura a Auditoria Geral do Estado e dá outras providências.

O Governador do Estado, no uso da atribuição que lhe foi conferida pela Resolução nº 5.210, de 12 de dezembro de 2002, da Assembléia Legislativa do Estado de Minas Gerais, decreta a seguinte Lei:

CAPÍTULO I

Do Sistema Estadual de Auditoria Interna

Seção I

Da Finalidade

Art. 1º - O Sistema Estadual de Auditoria Interna tem por finalidade o exercício das atividades de auditoria nos órgãos e entidades da administração pública do Poder Executivo, em fundos especiais instituídos por lei estadual de cujos recursos participe o Estado, em entidade em que o Estado tenha participação acionária direta ou indireta, bem como o exercício da atividade de correição administrativa dos servidores dos órgãos e entidades da Administração Pública do Poder Executivo.

Parágrafo único - O artigo 4º da Lei Delegada nº 49, de 2 de janeiro de 2003, fica acrescido do seguinte inciso V:

“Art. 4º -
V - Sistema Estadual de Auditoria Interna.”

Art. 2º - O Sistema Estadual de Auditoria Interna compreende 3 (três) áreas de atividades fim:

I - Subsistema de Auditoria Operacional tem por finalidade organizar e orientar as atividades de auditoria operacional da ação governamental, nos órgãos e entidades da Administração Pública Estadual, a fim de adequar os sistemas de controle interno, a qualidade do desempenho das áreas em relação à finalidade, aos objetivos e às competências, metas e políticas públicas, em consonância com os princípios da legalidade, impessoalidade, moralidade, publicidade, razoabilidade, eficácia, efetividade e economicidade;

II - Auditoria de Gestão tem por finalidade complementar, com alternativas políticas e estratégicas de gestão, os resultados técnicos disponibilizados pela auditoria operacional;

III - Correição Administrativa tem por finalidade promover correições gerais ou parciais nos órgãos e entidades da Administração Pública do Poder Executivo.

Seção II

Dos Objetivos

Art. 3º - O Sistema Estadual de Auditoria Interna tem por objetivos:

I - na área de Auditoria Operacional:

a) planejar e subsidiar a formulação da política de auditoria operacional do Estado;

b) implementar um subsistema de auditoria operacional adequado às necessidades e peculiaridades do Estado;

c) exercer o controle das receitas e dos gastos públicos;

II - na área de Auditoria de Gestão:

a) subsidiar a formulação e o ajustamento das políticas sob a responsabilidade do Sistema Central de Coordenação Geral, Planejamento, Gestão e Finanças;

b) desenvolver a consciência sobre o significado social da função auditoria interna como instrumento de avaliação e controle da gestão pública;

c) sistematizar a função auditoria em consonância com a continuidade da ação governamental;

d) articular-se com os órgãos de controle externo, com o objetivo de implantar as disposições constitucionais de integração do sistema de controle interno;

III - na área de correição administrativa:

a) exercer a correição administrativa relativa ao servidor público.

Seção III

Da Composição do Sistema

Art. 4º - O Sistema Estadual de Auditoria Interna tem a seguinte composição:
I - órgão central: Auditoria Geral do Estado;
II - unidade setorial: unidade de auditoria de órgão da Administração Direta;
III - unidade seccional: unidade de auditoria de entidade da Administração Indireta.

CAPÍTULO II

Das Atividades Organizadas em Subsistemas

Seção I

Da Estrutura e do Funcionamento das Atividades Organizadas em Subsistema
Art. 5º - As atividades de auditoria operacional do Estado são estruturadas na forma de Subsistema e tem por finalidade organizar e orientar as atividades de auditoria operacional da ação governamental, nos órgãos e entidades da Administração Pública Estadual, a fim de adequar os sistemas de controle, a qualidade do desempenho das áreas em relação à finalidade, aos objetivos e às competências, metas e políticas públicas, em consonância com os princípios da legalidade, impessoalidade, moralidade, publicidade, razoabilidade, eficácia, efetividade e economicidade.

Seção II

Da Composição do Subsistema

Art. 6º - O subsistema a que se refere o artigo anterior apresenta a seguinte composição:

I - Unidade Administrativa Central:

a) Superintendência Central de Auditoria Operacional;

II - Unidades Administrativas Setoriais: as que desenvolvem as atividades indicadas no artigo anterior em Secretarias de Estado e Órgãos Autônomos;

III - Unidades Administrativas Seccionais: as que desenvolvem as atividades indicadas no artigo anterior em entidades da administração indireta;

§ 1º - Os agentes do Subsistema de Auditoria Operacional subordinam-se, administrativa e diretamente, ao respectivo titular da Secretaria de Estado ou ao dirigente de órgão autônomo ou de entidade nos quais estejam atuando.

§ 2º - Os agentes do Subsistema de Auditoria Operacional subordinam-se tecnicamente à Superintendência Central de Auditoria Operacional.

CAPÍTULO III

Da Auditoria Geral do Estado – AGE¹

Art. 7º - A Auditoria Geral do Estado de Minas Gerais, criada na Lei Delegada nº 6, de 20 de agosto de 1985, é órgão autônomo diretamente subordinado ao Governador do Estado e tem sua organização definida nesta Lei.

Parágrafo único - Para os efeitos desta Lei, a expressão “Auditoria Geral do Estado”, a palavra “Auditoria” e a sigla “AGE” se equivalem.

Seção I

Da Finalidade e da Competência

Art. 8º - A Auditoria Geral do Estado tem por finalidade o exercício de atividades de:

I - auditoria e correição administrativa nos órgãos e entidades da Administração Pública do Poder Executivo;

II - auditoria em fundos especiais instituídos por lei estadual de cujos recursos participe o Estado, bem como em entidade em que o Estado tenha participação acionária direta ou indireta;

III - auditoria de gestão.

Art. 9º - Compete à Auditoria Geral do Estado:

I - zelar para que a atividade da administração pública se desenvolva segundo os princípios da legalidade, impessoalidade, moralidade, publicidade, razoabilidade, eficácia, efetividade e economicidade;

II - participar da formulação do programa de governo e das decisões a ele relativas;

III - verificar o cumprimento de normas e diretrizes do programa de governo e de sua eficácia;

IV - zelar pelo patrimônio público;

V - verificar o cumprimento da missão institucional dos órgãos e entidades da Administração Pública do Poder Executivo;

¹ Originalmente a sigla da Auditoria Geral do Estado era AGE. Entretanto, a Emenda à Constituição Estadual no. 56, de 11 de julho de 2003, criou a Advocacia-Geral do Estado, cuja sigla foi definida também como AGE. Por essa razão, desde a promulgação da referida Emenda à Constituição, a Auditoria Geral do Estado passou a ser identificada pela sigla AUGE, que foi utilizada em todo este trabalho.

- VI - acompanhar a repercussão pública e política das ações do Governo;
- VII - estabelecer o planejamento estratégico do Subsistema Estadual de Auditoria Operacional;
- VIII - verificar a adoção das providências sugeridas ou recomendadas em relatórios, pareceres e informações expedidos pelo Subsistema Estadual de Auditoria Operacional e pelas auditorias externas, e estabelecer prazos para esclarecimento e saneamento das deficiências e irregularidades apontadas;
- IX - articular-se com órgãos e entidades da Administração Pública do Poder Executivo, com o Ministério Público e o Tribunal de Contas do Estado, com o objetivo de realizar ações eficazes de combate à malversação dos recursos públicos;
- X - requisitar aos órgãos e entidades da Administração Pública, bem como a outras organizações com que se relacione, documentos e informações de qualquer classificação de sigilo, necessários ao desempenho de suas atribuições;
- XI - assessorar os dirigentes de órgãos e entidades no desempenho de suas atribuições, contribuindo para a integração entre as funções de planejamento, administração geral, finanças e contabilidade públicas das ações governamentais;
- XII - articular-se com as áreas externas, inclusive dos demais Poderes do Estado, cuja atuação seja relacionada com o sistema de controle interno, no sentido de uniformizar os entendimentos sobre matéria de interesse comum;
- XIII - promover a normatização, sistematização e padronização das normas e procedimentos de auditoria, no âmbito do Subsistema de Auditoria Operacional, em articulação com os órgãos sistêmicos de Modernização Administrativa e Planejamento Institucional;
- XIV - promover a normatização, sistematização e padronização das normas e procedimentos de auditoria, no âmbito do Subsistema de Auditoria Operacional, em articulação com os órgãos sistêmicos de Modernização Administrativa e Planejamento Institucional;
- XV - coordenar, supervisionar e orientar, normativa e tecnicamente, as atividades desenvolvidas pelas unidades de auditoria;
- XVI - exercer a correição administrativa relativa ao servidor público;
- XVII - exercer outras atividades correlatas.

Seção II

Da Estrutura Orgânica

Art. 10 - A Auditoria Geral do Estado - AGE tem a seguinte estrutura orgânica:

I - Gabinete;

II - Assessoria Técnica;

III - Superintendência Central de Auditoria Operacional:

a) Diretoria Central de Auditoria;

b) Diretoria Central de Desenvolvimento e Pesquisa;

IV - Superintendência Central de Auditoria de Gestão;

- a) Diretoria Central de Auditoria de Contas;
 - b) Diretoria Central de Auditorias Especiais
 - V - Superintendência Central de Correição Administrativa:
 - VI - Superintendência de Planejamento, Gestão e Finanças:
 - a) Diretoria de Administração e Finanças;
 - b) Diretoria de Planejamento e Orçamento;
- § 1º - As competências e a descrição das unidades previstas neste artigo, serão estabelecidas em decreto.
- § 2º - Os cargos correspondentes às unidades mencionadas neste artigo são de livre nomeação e exoneração do Governador do Estado.

CAPÍTULO IV

Disposições Finais

Art. 11 - As unidades de Auditoria Setorial e Auditoria Seccional integrantes das estruturas orgânicas das Secretarias de Estado, Órgãos Autônomos, Fundações e Autarquias são subordinados tecnicamente ao Auditor Geral do Estado.

Art. 12 - Os acervos técnicos e patrimoniais da Superintendência Central de Auditoria Operacional e da Superintendência Central de Correição Administrativa, utilizados no desempenho das atividades de auditoria operacional e correição administrativa, ficam transferidos para a Auditoria Geral do Estado.

Art. 13 - A subordinação técnica dos agentes do Subsistema de Auditoria Operacional, de que trata o artigo 6º, efetivar-se-á mediante a:

I - observância das diretrizes estabelecidas pela Superintendência Central de Auditoria Operacional;

II - observância das normas e técnicas de auditoria, estabelecidas pelos órgãos normativos, para a função de auditoria interna;

III - elaboração e execução dos planos anuais de auditoria, com orientação e aprovação da Superintendência Central de Auditoria Operacional;

IV - utilização dos planos e roteiros de auditoria disponibilizados pela Superintendência Central de Auditoria Operacional, bem como das informações, padrões e parâmetros técnicos para subsídios dos trabalhos de auditoria;

V - observância dos padrões de desempenho e de elaboração dos relatórios de auditoria definidos pela Superintendência Central de Auditoria Operacional;

VI - monitoria da efetividade das ações de auditoria.

Parágrafo único - A execução dos trabalhos de auditoria, pelos agentes do Subsistema de Auditoria Operacional, será coordenada pela Superintendência Central de Auditoria Operacional.

Art. 14 - O servidor investido em cargo ou função de auditor e afins ou no desempenho de atribuições inerentes à atividade de auditoria atuará, única e exclusivamente, no gerenciamento, na execução e no apoio técnico dessa atribuição.

Parágrafo único - A indicação de servidor para o exercício da atividade de auditoria será feita em observância aos critérios e qualificação profissional definidos pela Auditoria Geral do Estado - AGE.

Art. 15 - Esta lei entra em vigor na data de sua publicação.

Art. 16 - Revogam-se as disposições em contrário, especialmente os artigos 1º ao 9º da Lei nº 13.466, de 12 de janeiro de 2000.

Palácio da Liberdade, em Belo Horizonte, aos 29 de janeiro de 2003.

AÉCIO NEVES - Governador do Estado.

Apêndice B

Tradução da Licença GPL para português

GPL – Licença Pública Geral GNU

Versão 2, junho de 1991

This is an unofficial translation of the GNU General Public License into Brazilian Portuguese. It was not published by the Free Software Foundation, and does not legally state the distribution terms for software that uses the GNU GPL – only the original English text of the GNU GPL does that. However, we hope that this translation will help Brazilian Portuguese speakers understand the GNU GPL better.

Esta é uma tradução¹ não-oficial da Licença Pública Geral GNU (“GPL GNU”) para o português do Brasil. Ela não foi publicada pela Free Software Foundation, e legalmente não afirma os termos de distribuição de software que utiliza a GPL GNU – apenas o texto original da GPL GNU, em inglês, faz isso. Contudo, esperamos que esta tradução ajude aos que utilizam o português do Brasil a entender melhor a GPL GNU.

Copyright ©1989, 1991 Free Software Foundation, Inc. 675 Mass Ave, Cambridge, MA 02139, USA

A qualquer pessoa é permitido copiar e distribuir cópias desse documento de licença, desde que sem qualquer alteração.

¹ Disponível em http://www.magnux.org/doc/GPL-pt_BR.txt. Último acesso: 15 de janeiro de 2004.

B.1 Introdução

As licenças de muitos software são desenvolvidas para restringir sua liberdade de compartilhá-lo e mudá-lo. Contrária a isso, a Licença Pública Geral GNU pretende garantir sua liberdade de compartilhar e alterar software livres – garantindo que o software será livre e gratuito para os seus usuários. Esta Licença Pública Geral aplica-se à maioria dos software da Free Software Foundation e a qualquer outro programa cujo autor decida aplicá-la. (Alguns outros software da FSF são cobertos pela Licença Pública Geral de Bibliotecas, no entanto.) Você pode aplicá-la também aos seus programas.

Quando nos referimos a software livre, estamos nos referindo a liberdade e não a preço. Nossa Licença Pública Geral foi desenvolvida para garantir que você tenha a liberdade de distribuir cópias de software livre (e cobrar por isso, se quiser); que você receba o código-fonte ou tenha acesso a ele, se quiser; que você possa mudar o software ou utilizar partes dele em novos programas livres e gratuitos; e que você saiba que pode fazer tudo isso.

Para proteger seus direitos, precisamos fazer restrições que impeçam a qualquer um negar estes direitos ou solicitar que você deles abdique. Estas restrições traduzem-se em certas responsabilidades para você, se você for distribuir cópias do software ou modificá-lo.

Por exemplo, se você distribuir cópias de um programa, gratuitamente ou por alguma quantia, você tem que fornecer aos recebedores todos os direitos que você possui. Você tem que garantir que eles também recebam ou possam obter o código-fonte. E você tem que mostrar-lhes estes termos para que eles possam conhecer seus direitos.

Nós protegemos seus direitos em dois passos: (1) com copyright do software e (2) com a oferta desta licença, que lhe dá permissão legal para copiar, distribuir e/ou modificar o software.

Além disso, tanto para a proteção do autor quanto a nossa, gostaríamos de certificar-nos que todos entendam que não há qualquer garantia nestes software livres. Se o software é modificado por alguém mais e passado adiante, queremos que seus recebedores saibam que o que eles obtiveram não é original, de forma que qualquer problema introduzido por terceiros não interfira na reputação do autor original.

Finalmente, qualquer programa é ameaçado constantemente por patentes de software. Queremos evitar o perigo de que distribuidores de software livre obtenham patentes individuais, o que tem o efeito de tornar o programa proprietário. Para prevenir isso, deixamos claro que qualquer patente tem que ser licenciada para uso livre e gratuito por qualquer pessoa, ou então que nem necessite ser licenciada.

Os termos e condições precisas para cópia, distribuição e modificação se encontram abaixo:

B.2 Licença Pública Geral GNU

Termos e condições para cópia, distribuição e modificação

0. Esta licença se aplica a qualquer programa ou outro trabalho que contenha um aviso colocado pelo detentor dos direitos autorais informando que aquele pode ser distribuído sob as condições desta Licença Pública Geral. O “Programa” abaixo refere-se a qualquer programa ou trabalho, e “trabalho baseado no Programa” significa tanto o Programa em si como quaisquer trabalhos derivados, de acordo com a lei de direitos autorais: isto quer dizer um trabalho que contenha o Programa ou parte dele, tanto originalmente ou com modificações, e/ou tradução para outros idiomas. (Doravante o processo de tradução está incluído sem limites no termo “modificação”.) Cada licenciado é mencionado como “você”.

Atividades outras que a cópia, a distribuição e modificação não estão cobertas por esta Licença; elas estão fora de seu escopo. O ato de executar o Programa não é restringido e o resultado do Programa é coberto apenas se seu conteúdo contenha trabalhos baseados no Programa (independentemente de terem sido gerados pela execução do Programa). Se isso é verdadeiro depende do que o programa faz.

1. Você pode copiar e distribuir cópias fiéis do código-fonte do Programa da mesma forma que você o recebeu, usando qualquer meio, desde que você conspícua e apropriadamente publique em cada cópia um aviso de direitos autorais e uma declaração de inexistência de garantias; mantenha intactas todos os avisos que se referem a esta Licença e à ausência total de garantias; e forneça a outros recebedores do Programa uma cópia desta Licença, junto com o Programa.

Você pode cobrar pelo ato físico de transferir uma cópia e pode, opcionalmente, oferecer garantia em troca de pagamento.

2. Você pode modificar sua cópia ou cópias do Programa, ou qualquer parte dele, assim gerando um trabalho baseado no Programa, e copiar e distribuir essas modificações ou trabalhos sob os termos da seção 1 acima, desde que você também se enquadre em todas estas condições:

- a) Você tem que fazer com que os arquivos modificados levem avisos proeminentes afirmando que você alterou os arquivos, incluindo a data de qualquer alteração.
- b) Você tem que fazer com que quaisquer trabalhos que você distribua ou publique, e que integralmente ou em partes contenham ou sejam

derivados do Programa ou de suas partes, sejam licenciados, integralmente e sem custo algum para quaisquer terceiros, sob os termos desta Licença.

- c) Se qualquer programa modificado normalmente lê comandos interativamente quando executados, você tem que fazer com que, quando iniciado tal uso interativo da forma mais simples, seja impresso ou mostrado um anúncio de que não há qualquer garantia (ou então que você fornece a garantia) e que os usuários podem redistribuir o programa sob estas condições, ainda informando os usuários como consultar uma cópia desta Licença. (Exceção: se o Programa em si é interativo mas normalmente não imprime estes tipos de anúncios, seu trabalho baseado no Programa não precisa imprimir um anúncio.)

Estas exigências aplicam-se ao trabalho modificado como um todo. Se seções identificáveis de tal trabalho não são derivadas do Programa, e podem ser razoavelmente consideradas trabalhos independentes e separados por si só, então esta Licença, e seus termos, não se aplicam a estas seções quando você distribui-las como trabalhos em separado. Mas quando você distribuir as mesmas seções como parte de um todo que é trabalho baseado no Programa, a distribuição como um todo tem que se enquadrar nos termos desta Licença, cujas permissões para outros licenciados se estendem ao todo, portanto também para cada e toda parte independente de quem a escreveu.

Desta forma, esta seção não tem a intenção de reclamar direitos ou contestar seus direitos sobre o trabalho escrito completamente por você; ao invés disso, a intenção é a de exercitar o direito de controlar a distribuição de trabalhos, derivados ou coletivos, baseados no Programa.

Adicionalmente, a mera adição ao Programa de outro trabalho não baseado no Programa (ou de trabalho baseado no Programa) em um volume de armazenamento ou meio de distribuição não faz o outro trabalho parte do escopo desta Licença.

3. Você pode copiar e distribuir o Programa (ou trabalho baseado nele, conforme descrito na Seção 2) em código-objeto ou em forma executável sob os termos das Seções 1 e 2 acima, desde que você faça um dos seguintes:
 - a) O acompanhe com o código-fonte completo e em forma acessível por máquinas, que tem que ser distribuído sob os termos das Seções 1 e 2 acima e em meio normalmente utilizado para o intercâmbio de software; ou,
 - b) O acompanhe com uma oferta escrita, válida por pelo menos três anos, de fornecer a qualquer um, com um custo não superior ao custo de

distribuição física do material, uma cópia do código-fonte completo e em forma acessível por máquinas, que tem que ser distribuído sob os termos das Seções 1 e 2 acima e em meio normalmente utilizado para o intercâmbio de software; ou,

- c) O acompanhe com a informação que você recebeu em relação à oferta de distribuição do código-fonte correspondente. (Esta alternativa é permitida somente em distribuição não comerciais, e apenas se você recebeu o programa em forma de código-objeto ou executável, com oferta de acordo com a Subseção b acima.)

O código-fonte de um trabalho corresponde à forma de trabalho preferida para se fazer modificações. Para um trabalho em forma executável, o código-fonte completo significa todo o código-fonte de todos os módulos que ele contém, mais quaisquer arquivos de definição de “interface”, mais os “scripts” utilizados para se controlar a compilação e a instalação do executável. Contudo, como exceção especial, o código-fonte distribuído não precisa incluir qualquer componente normalmente distribuído (tanto em forma original quanto binária) com os maiores componentes (o compilador, o “kernel” etc.) do sistema operacional sob o qual o executável funciona, a menos que o componente em si acompanhe o executável.

Se a distribuição do executável ou código-objeto é feita através da oferta de acesso a cópias de algum lugar, então ofertar o acesso equivalente a cópia, do mesmo lugar, do código-fonte equivale à distribuição do código-fonte, mesmo que terceiros não sejam compelidos a copiar o código-fonte com o código-objeto.

- 4. Você não pode copiar, modificar, sub-licenciar ou distribuir o Programa, exceto de acordo com as condições expressas nesta Licença. Qualquer outra tentativa de cópia, modificação, sub-licenciamento ou distribuição do Programa não é válida, e cancelará automaticamente os direitos que lhe foram fornecidos por esta Licença. No entanto, terceiros que de você receberam cópias ou direitos, fornecidos sob os termos desta Licença, não terão suas licenças terminadas, desde que permaneçam em total concordância com ela.
- 5. Você não é obrigado a aceitar esta Licença já que não a assinou. No entanto, nada mais o dará permissão para modificar ou distribuir o Programa ou trabalhos derivados deste. Estas ações são proibidas por lei, caso você não aceite esta Licença. Desta forma, ao modificar ou distribuir o Programa (ou qualquer trabalho derivado do Programa), você estará indicando sua total aceitação desta Licença para fazê-los, e todos os seus termos e condições para copiar, distribuir ou modificar o Programa, ou trabalhos baseados nele.

6. Cada vez que você redistribuir o Programa (ou qualquer trabalho baseado nele), os recebedores adquirirão automaticamente do licenciador original uma licença para copiar, distribuir ou modificar o Programa, sujeitos a estes termos e condições. Você não poderá impor aos recebedores qualquer outra restrição ao exercício dos direitos então adquiridos. Você não é responsável em garantir a concordância de terceiros a esta Licença.
7. Se, em consequência de decisões judiciais ou alegações de infringimento de patentes ou quaisquer outras razões (não limitadas a assuntos relacionados a patentes), condições forem impostas a você (por ordem judicial, acordos ou outras formas) e que contradigam as condições desta Licença, elas não o livram das condições desta Licença. Se você não puder distribuir de forma a satisfazer simultaneamente suas obrigações para com esta Licença e para com as outras obrigações pertinentes, então como consequência você não poderá distribuir o Programa. Por exemplo, se uma licença de patente não permitirá a redistribuição, livre de "royalties", do Programa, por todos aqueles que receberem cópias direta ou indiretamente de você, então a única forma de você satisfazer a ela e a esta Licença seria a de desistir completamente de distribuir o Programa.

Se qualquer parte desta seção for considerada inválida ou não aplicável em qualquer circunstância particular, o restante da seção se aplica, e a seção como um todo se aplica em outras circunstâncias.

O propósito desta seção não é o de induzi-lo a infringir quaisquer patentes ou reivindicação de direitos de propriedade outros, ou a contestar a validade de quaisquer dessas reivindicações; esta seção tem como único propósito proteger a integridade dos sistemas de distribuição de software livres, o que é implementado pela prática de licenças públicas. Várias pessoas têm contribuído generosamente e em grande escala para os software distribuídos usando este sistema, na certeza de que sua aplicação é feita de forma consistente; fica a critério do autor/doador decidir se ele ou ela está disposto a distribuir software utilizando outro sistema, e um licenciado não pode impor qualquer escolha.

Esta seção destina-se a tornar bastante claro o que se acredita ser consequência do restante desta Licença.

8. Se a distribuição e/ou uso do Programa são restringidos em certos países por patentes ou direitos autorais, o detentor dos direitos autorais original, e que colocou o Programa sob esta Licença, pode incluir uma limitação geográfica de distribuição, excluindo aqueles países de forma a tornar a distribuição permitida apenas naqueles ou entre aqueles países então não excluídos. Nestes casos, esta Licença incorpora a limitação como se a mesma constasse escrita nesta Licença.

9. A Free Software Foundation pode publicar versões revisadas e/ou novas da Licença Pública Geral de tempos em tempos. Estas novas versões serão similares em espírito à versão atual, mas podem diferir em detalhes que resolvem novos problemas ou situações.

A cada versão é dada um número distinto. Se o Programa especifica um número de versão específico desta Licença que se aplica a ele e a "qualquer nova versão", você tem a opção de aceitar os termos e condições daquela versão ou de qualquer outra versão publicada pela Free Software Foundation. Se o programa não especifica um número de versão desta Licença, você pode escolher qualquer versão já publicada pela Free Software Foundation.

10. Se você pretende incorporar partes do Programa em outros programas livres cujas condições de distribuição são diferentes, escreva ao autor e solicite permissão. Para o software que a Free Software Foundation detém direitos autorais, escreva à Free Software Foundation; às vezes nós permitimos exceções a este caso. Nossa decisão será guiada pelos dois objetivos de preservar a condição de liberdade de todas as derivações do nosso software livre, e de promover o compartilhamento e reutilização de software em aspectos gerais.

AUSÊNCIA DE GARANTIAS

11. UMA VEZ QUE O PROGRAMA É LICENCIADO SEM ÔNUS, NÃO HÁ QUALQUER GARANTIA PARA O PROGRAMA, NA EXTENSÃO PERMITIDA PELAS LEIS APLICÁVEIS. EXCETO QUANDO EXPRESADO DE FORMA ESCRITA, OS DETENTORES DOS DIREITOS AUTORAIS E/OU TERCEIROS DISPONIBILIZAM O PROGRAMA “NO ESTADO”, SEM QUALQUER TIPO DE GARANTIAS, EXPRESSAS OU IMPLÍCITAS, INCLUINDO, MAS NÃO LIMITADO A, AS GARANTIAS IMPLÍCITAS DE COMERCIALIZAÇÃO E AS DE ADEQUAÇÃO A QUALQUER PROPÓSITO. O RISCO TOTAL COM A QUALIDADE E DESEMPENHO DO PROGRAMA É SEU. SE O PROGRAMA SE MOSTRAR DEFEITUOSO, VOCÊ ASSUME OS CUSTOS DE TODAS AS MANUTENÇÕES, REPAROS E CORREÇÕES.
12. EM NENHUMA OCASIÃO, A MENOS QUE EXIGIDO PELAS LEIS APLICÁVEIS OU ACORDO ESCRITO, OS DETENTORES DOS DIREITOS AUTORAIS, OU QUALQUER OUTRA PARTE QUE POSSA MODIFICAR E/OU REDISTRIBUIR O PROGRAMA CONFORME PERMITIDO ACIMA, SERÃO RESPONSABILIZADOS POR VOCÊ POR DANOS, INCLUINDO QUALQUER DANO EM GERAL, ESPECIAL, ACIDENTAL OU CONSEQÜENTE, RESULTANTES DO USO OU INCA-

PACIDADE DE USO DO PROGRAMA (INCLUINDO, MAS NÃO LIMITADO A, A PERDA DE DADOS OU DADOS TORNADOS INCORRETOS, OU PERDAS SOFRIDAS POR VOCÊ OU POR OUTRAS PARTES, OU FALHAS DO PROGRAMA AO OPERAR COM QUALQUER OUTRO PROGRAMA), MESMO QUE TAL DETENTOR OU PARTE TENHAM SIDO AVISADOS DA POSSIBILIDADE DE TAIS DANOS.

FIM DOS TERMOS E CONDIÇÕES

Como Aplicar Estes Termos aos Seus Novos Programas

Se você desenvolver um novo programa, e quer que ele seja utilizado amplamente pelo público, a melhor forma de alcançar este objetivo é torná-lo software livre que qualquer um pode redistribuir e alterar, sob estes termos.

Para isso, anexe os seguintes avisos ao programa. É mais seguro anexá-los logo no início de cada arquivo-fonte para reforçarem mais efetivamente a inexistência de garantias; e cada arquivo deve possuir pelo menos a linha de "copyright" e uma indicação de onde o texto completo se encontra.

<uma linha que forneça o nome do programa e uma idéia do que ele faz.> Copyright (C) <ano> <nome do autor>

Este programa é software livre; você pode redistribuí-lo e/ou modificá-lo sob os termos da Licença Pública Geral GNU, conforme publicada pela Free Software Foundation; tanto a versão 2 da Licença como (a seu critério) qualquer versão mais nova.

Este programa é distribuído na expectativa de ser útil, mas SEM QUALQUER GARANTIA; sem mesmo a garantia implícita de COMERCIALIZAÇÃO ou de ADEQUAÇÃO A QUALQUER PROPÓSITO EM PARTICULAR. Consulte a Licença Pública Geral GNU para obter mais detalhes.

Você deve ter recebido uma cópia da Licença Pública Geral GNU junto com este programa; se não, escreva para a Free Software Foundation, Inc., 59 Temple Place, Suite 330, Boston, MA 02111-1307, USA.

Inclua também informações sobre como contactá-lo eletronicamente e por carta.

Se o programa é interativo, faça-o mostrar um aviso breve como este, ao iniciar um modo interativo:

Gnomovision versão 69, Copyright (C) ano nome do autor

O Gnomovision não possui QUALQUER GARANTIA; para obter mais detalhes digite 'show w'. Ele é software livre e você está convidado a redistribuí-lo sob certas condições; digite 'show c' para obter detalhes.

Os comandos hipotéticos 'show w' e 'show c' devem mostrar as partes apropriadas da Licença Pública Geral. Claro, os comandos que você usar podem ser ativados de outra forma que 'show w' e 'show c'; eles podem até ser cliques do mouse ou itens de um menu – o que melhor se adequar ao programa.

Você também deve obter do seu empregador (se você trabalha como programador) ou escola, se houver, uma “declaração de ausência de direitos autorais” sobre o programa, se necessário. Aqui está um exemplo; altere os nomes:

Yoyodyne, Inc., aqui declara a ausência de quaisquer direitos autorais sobre o programa 'Gnomovision' (que executa interpretações em compiladores) escrito por James Hacker.

<assinatura de Ty Coon>, 1o. de abril de 1989 Ty Coon, Vice-presidente

Esta Licença Pública Geral não permite incorporar seu programa em programas proprietários. Se seu programa é uma biblioteca de sub-rotinas, você deve considerar mais útil permitir ligar aplicações proprietárias com a biblioteca. Se isto é o que você deseja, use a Licença Pública Geral de Bibliotecas GNU, ao invés desta Licença.

Apêndice C

Rotinas de monitoramento do SAMBA

O SAMBA permite execução de rotinas antes e depois do acesso a um determinado compartilhamento. Isso é conseguido por meio das diretivas destacadas:

- *preexec* e *postexec* – definem execução de rotinas antes (pre) e após (post) o acesso a um determinado compartilhamento. Nesse caso as rotinas são executadas no sistema operacional com a identificação (UID) do usuário solicitante;
- *root preexec* e *root postexec* – os comandos são executados no sistema operacional com permissões de superusuário (*root*).

Para refinar o controle do servidor de arquivos da SCAO foram utilizadas rotinas em todos compartilhamentos para os trabalhos de auditoria, disparados pelas diretivas *root preexec* e *root postexec*, conforme Figura C.1.

```
root preexec = /adm/comandos/sambalog/smbcontrol %U %S %m %u %I INI
root postexec = /adm/comandos/sambalog/smbcontrol %U %S %m %u %I FIM
```

Figura C.1: *Scripts* executados para os trabalhos de auditoria.

Além disso, no compartilhamento *netlogon*, foi utilizada a diretiva *root preexec*, conforme Figura C.2, objetivando uma pesquisa mais rápida para identificação das máquinas que efetuaram *logon* no SAMBA.

```
root preexec = /age/aplic/adm/comandos/sambalog/smbmaqlog %m %I %S %U
```

Figura C.2: *Script* executado no compartilhamento *netlogon*.

As rotinas *smbcontrol* e *smbmaqllog* serão discutidas nas próximas seções, após a apresentação das variáveis utilizadas como parâmetros.

C.1 Variáveis utilizadas nas rotinas

O SAMBA possui um conjunto de variáveis que permitem determinar comportamentos específicos. A tabela C.1 apresenta as variáveis utilizadas como parâmetros¹ nas três rotinas de controle da SCAO.

Tabela C.1: Variáveis utilizadas nas rotinas *smbcontrol* e *smbmaqllog*.

Nome	Significado
%I	endereço IP do cliente
%m	nome NetBIOS da máquina cliente
%S	nome do compartilhamento corrente
%u	nome do usuário do compartilhamento corrente
%U	nome do usuário requisitado para o compartilhamento corrente

C.2 Rotina *smbcontrol*

A rotina *smbcontrol* está representada na Figura C.3. Essa rotina, recebe seis parâmetros indicados na diretiva *root preexec* ou *root postexec*: nome de usuário da seção; nome do compartilhamento; nome NetBIOS da máquina cliente; nome do usuário do compartilhamento corrente; endereço IP da máquina cliente e; um parâmetro textual “INI” ou “FIM” que representa se o usuário está prestes a iniciar (*root preexec*) ou encerrar (*root postexec*) seu acesso ao compartilhamento. De posse desses parâmetros a rotina armazena-os em variáveis internas, respectivamente: *usr_secao*, *maquina*, *recurso*, *usr_serv*, *endip* e *inicio_fim*. Além desses parâmetros, o *script* armazena a data e hora corrente na variável *auxdata*². Após “coletar” todos os dados, a rotina irá alimentar dois arquivos de *log* específicos, por meio da variável *frase*.

Os arquivos de *log* gerados são determinados pelas variáveis *arquivo_user* e *arquivo_share*. Servem para realizar controle por usuário e por compartilhamento

¹ Os parâmetros INI e FIM utilizadas no *script smbcontrol* não são variáveis previstas no SAMBA. Servem para identificar se o acesso ao compartilhamento foi iniciado ou finalizado, no arquivo de *log*.

² O SAMBA prevê a variável %T para mostrar a data e hora atual. Entretanto nos testes realizados com o Solaris, o parâmetro %T retornou apenas a data atual. Por essa razão, nas rotinas *smbcontrol* e *smbmaqllog*, foi adotada a estratégia de buscar diretamente do sistema a data e hora atual.

```
#!/bin/sh

usr_secao=$1
maquina=$3
recurso=$2
usr_serv=$4
endip=$5
inicio_fim=$6

auxdata=`date`
arquivo_user=/dados/log/samba/user/$usuario.log
arquivo_share=/dados/log/samba/share/$recurso.log

frase='`$inicio_fim: $auxdata $recurso $endip $maquina $usr_serv`'
echo $frase >> $arquivo_user

frase='`$inicio_fim: $auxdata $endip $maquina $usr_secao`'
echo $frase >> $arquivo_share
```

Figura C.3: Rotina *smbcontrol*.

acessado. Exemplos de *arquivo_user* e *arquivo_share* são destacados nas Figuras³ C.4 e C.5, respectivamente.

```
INI: Thu Nov 14 14:14:28 BRST 2002 LC33 xxx.xxx.xxx.77 cemig01 danielcr
INI: Thu Nov 14 14:14:28 BRST 2002 CPD xxx.xxx.xxx.77 cemig01 danielcr
INI: Thu Nov 14 14:14:28 BRST 2002 Analit xxx.xxx.xxx.77 cemig01 danielcr
INI: Thu Nov 14 14:14:29 BRST 2002 DCDP xxx.xxx.xxx.77 cemig01 danielcr
INI: Thu Nov 14 14:14:29 BRST 2002 GPess xxx.xxx.xxx.77 cemig01 danielcr
FIM: Thu Nov 14 17:14:32 BRST 2002 Analit xxx.xxx.xxx.77 cemig01 danielcr
FIM: Thu Nov 14 17:16:51 BRST 2002 CPD xxx.xxx.xxx.77 cemig01 danielcr
FIM: Thu Nov 14 17:16:51 BRST 2002 DCDP xxx.xxx.xxx.77 cemig01 danielcr
FIM: Thu Nov 14 17:16:51 BRST 2002 LC33 xxx.xxx.xxx.77 cemig01 danielcr
FIM: Thu Nov 14 17:16:51 BRST 2002 GPess xxx.xxx.xxx.77 cemig01 danielcr
INI: Thu Nov 14 18:17:40 BRST 2002 Analit xxx.xxx.xxx.21 alfenas danielcr
FIM: Thu Nov 14 18:25:17 BRST 2002 Analit xxx.xxx.xxx.21 alfenas danielcr
```

Figura C.4: Trecho do log gerado pelo script *smbcontrol* para usuários.

```
INI: Thu Nov 14 13:55:52 BRST 2002 xxx.xxx.xxx.21 alfenas imaculsb
INI: Thu Nov 14 13:57:57 BRST 2002 xxx.xxx.xxx.16 diamantina terezacr
INI: Thu Nov 14 14:14:29 BRST 2002 xxx.xxx.xxx.77 cemig01 danielcr
FIM: Thu Nov 14 14:16:51 BRST 2002 xxx.xxx.xxx.77 cemig01 danielcr
INI: Thu Nov 14 14:23:25 BRST 2002 xxx.xxx.xxx.19 passos charleas
FIM: Thu Nov 14 14:36:11 BRST 2002 xxx.xxx.xxx.19 passos charleas
FIM: Thu Nov 14 15:26:29 BRST 2002 xxx.xxx.xxx.16 diamantina terezacr
FIM: Thu Nov 14 16:04:26 BRST 2002 xxx.xxx.xxx.21 alfenas imaculsb
```

Figura C.5: Log gerado pelo script *smbcontrol* para compartilhamentos.

³ Os endereços IP foram mascarados por questões de segurança.

Com essa implementação foi possível estabelecer mecanismos de identificação de acesso (trilhas de auditoria) especialmente para usuários e compartilhamentos. A próxima seção irá demonstrar uma nova alternativa para registro de acesso por máquinas.

C.3 Rotina *smbmaqlog*

A rotina *smbmaqlog* está representada na Figura C.6. Seu funcionamento é aná-

```
#!/bin/bash

usuario=$4
maquina=$1
recurso=$3
endip=$2

auxdata=`date`
arquivo=/dados/log/samba/maquinas/$maquina.log

frase='`$auxdata: $recurso $usuario $endip`'
echo $frase >> $arquivo
```

Figura C.6: Rotina *smbmaqlog*.

logo à rotina *smbcontrol*: parâmetros são passados pelo SAMBA, nesse caso, apenas pela diretiva *root preexec*, que são organizados pela rotina e armazenados em arquivos de *log* cujos nomes são as identidades NetBIOS das estações.

O objetivo dessa implementação é permitir uma pesquisa mais imediata para as máquinas que solicitaram acesso ao SAMBA. Apesar desses dados estarem disponíveis nos dois arquivos de *log* obtidos da rotina *smbcontrol*, a verificação de acesso por máquinas “não conhecidas” pela SCAO seria mais complexa e tediosa se esses fossem utilizados. Com a utilização do *script smbmaqlog* torna-se mais fácil identificar que uma máquina desconhecida tenha conseguido acesso ao servidor. Um trecho de um arquivo de log gerado é demonstrado na Figura C.7.

```
Thu Nov 7 14:33:42 BRST 2002: netlogon sandrasr xxx.xxx.xxx.24
Thu Nov 7 15:35:16 BRST 2002: netlogon marcioro xxx.xxx.xxx.24
Fri Nov 8 10:05:17 BRST 2002: netlogon marcioro xxx.xxx.xxx.24
Fri Nov 8 13:15:46 BRST 2002: netlogon marcioro xxx.xxx.xxx.24
Fri Nov 8 18:22:40 BRST 2002: netlogon charleas xxx.xxx.xxx.24
```

Figura C.7: Trecho de *log* gerado pelo *script smbmaqlog*.

Apêndice D

Distribuições GNU/Linux disponíveis no mercado.

Distribuição	Versão	Comentário
Conectiva www.conectiva.com.br	9	Distribuição brasileira destinada a servidores e <i>desktop</i> .
Debian www.debian.org	3.0r2	Famosa distribuição Linux, indicada para servidores. Sua construção só utiliza pacotes completamente GPL.
Definity www.definitylinux.com.br	2.0	Distribuição brasileira voltada à instalação de <i>Desktop</i> e servidores.
Fedora www.fedora.us	1	Projeto desenvolvido pela Red Hat. Foi criado para ser o sucessor do mais conhecido Linux.
Gentoo www.gentoo.org	1,4	Com ferramentas básicas, destina-se ao uso pessoal.
Insigne www.insignesoftware.com	2.0	Distribuição Linux brasileira para ser comercializada no formato OEM.
Kurumin www.guiadohardware.info	2.12	Baseado na distribuição Knoppix, com a diferença de ter suporte à língua portuguesa, versão Brasil.
Lindows www.lindows.com	4.5	Destinado ao usuário iniciante, a empresa tentou criar um sistema que executasse programas Windows e Linux, mas falhou.
Lycoris www.lycoris.com	Update 3	Antes, esta distribuição era chamada Redmond Linux. A sua interface é idêntica à do Windows.

Distribuição	Versão	Comentário
Mandrake www.mandrake.com	9.2	Uma distribuição completa e com muitas facilidades para usuários iniciantes no Linux.
SmoothWall www.smoothwall.org	2.0	Minidistribuição Linux que pode ser instalada em computadores obsoletos. Possui recursos exclusivos para criação de servidores.
Slackware www.slackware.com	9.1	Famosa distribuição Linux indicada para servidores.
SUSE www.suse.com	9.0	Distribuição Linux desenvolvida na Alemanha, muito completa e destinada a <i>desktop</i> e servidor de rede.
Turbolinux www.turbolinux.com	10	Distribuição Linux desenvolvida na Ásia. É muito completa e destinada a <i>desktop</i> e servidor de rede.

Fonte: Revista PC-Master – fevereiro de 2004.