Uso de softwares Open Source na autenticação de acesso WEB em controlador de domínio NT

Aluno: Angelo Alberto Delben Filho, Orientador: Elio Lovisi Filho.

Departamento de Ciência da Computação
Faculdade de Ciência da Computação e Comunicação Social (FACICS)
Universidade Presidente Antônio Carlos
Campus Magnus - Campolide - MG
angelodelben@yahoo.com.br, eliolovisi@nextwave.com.br

Resumo. *O presente artigo apresenta uma proposta de uma aplicação prática de sistemas Open Source para autenticar navegação WEB em domínio NT através da configuração do Squid, um servidor proxy existente no mundo do software livre. Tem por finalidade facilitar a tarefa de administradores de redes, pois autenticação de serviços é muito importante por motivos de segurança e praticidade.*

1. Introdução

Em segurança de dados de computadores, os administradores de redes procuram os mais diversos meios para implementar confiabilidade, praticidade, tolerância a falhas, disponibilidade, flexibilidade e performance nos sistemas de suas empresas. Desta forma, os conceitos de criptografia, autenticação, backup, firewall e antivírus são as soluções mais importantes atualmente para administradores de redes.

Existe uma infinidade de serviços relacionados com o computador que necessitam manter a privacidade de cada usuário integrante do sistema assim como permitir o gerenciamento de usuários de acordo com o perfil de cada um.

Nesse contexto, este artigo visa apresentar os resultados da aplicação da autenticação como uma das formas mais importantes de se manter a identidade dos usuários e o gerenciamento dos mesmos no serviço de acesso à internet. No gerenciamento, o administrador de redes poderá controlar acessos, definir quotas para a transferência de dados e obter histórico de operações feitas pelos usuários, como exemplo, quantos sites foram acessados por dia. Visa também mostrar como configurar o *squid*, uma ferramenta *open source* (Código aberto ao público) que permite várias funções de gerenciamento e performance como cache (armazenar páginas mais usadas), autenticação, controle de banda (tráfego), controle de horário de acesso, seleção de páginas a serem negadas e histórico de acessos.

2. Controlador de domínio

Quando a quantidade de computadores envolvidos nas redes de computadores é muito pequena, o costume é de se usar *workgroups*, que nada mais são do que grupos de computadores que irão trabalhar na mesma rede. O conceito de rede pequena não está relacionado com a quantidade de computadores e sim com a utilização de recursos (compartilhamentos) existente nesta rede. Em se tratando de redes maiores, o uso de

workgroups já não é mais suficiente para que o administrador de redes possa gerenciar recursos da rede como compartilhamento de arquivos, impressoras, perfis móveis e níveis de permissões.

Assim sendo, algumas empresas proprietárias e outras de licença GPL (General Public License) colocam no mercado softwares denominados controladores de domínio. Como exemplos são citados o active directory do *Windows Server* ou NT e o SAMBA do linux.

A função de um controlador de domínio é autenticar usuários, serviços, armazenar o perfil móvel dos usuários e manter o nível de segurança. Neste caso será utilizado o PDC (Controlador de domínio primário) para autenticar o serviço de navegação à internet através de um servidor *Proxy* (Figura 1).

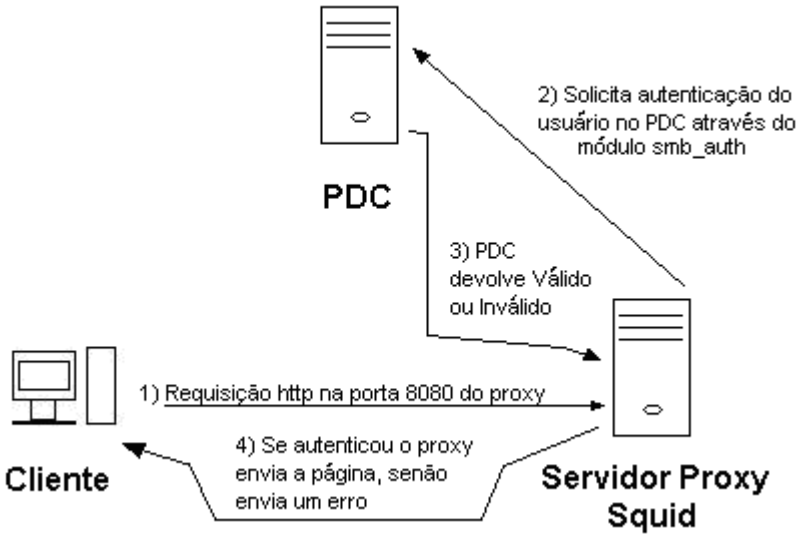


Figura 1. Modelo de Navegação com autenticação

3. Servidor Proxy

Com o advento da internet, o mundo inteiro está conectado na enorme teia mundial, a WWW (*World Wide Web*). Para um ponto acessar outro, comumente utiliza-se um *browser* (navegador) na qual a solicitação é feita através do protocolo[1] de comunicação TCP/IP (*Transfer Control Protocol / Internet Protocol*) onde os serviços, principalmente *http*(*HyperText Transfer Protocol*) e *ftp*(*File Transfer Protocol*), podem ser disponibilizados de um lado a outro do mundo.

Para haver comunicação de um lado a outro, é necessários que as duas pontas conheçam a mesma linguagem ou protocolo. O TCP/IP através do encapsulamento da camada de rede (IP) e transporte (TCP) fornece abstração para a camada de aplicação, onde estão inseridos os protocolos de serviços como, por exemplo, *http*(porta 80), *ftp* (portas 20 e 21), *dns* (porta 53), e *telnet* (porta 23) [IANA 2006].

[1] *Protocolo* é descrição formal da estrutura de mensagem e das regras que dois computadores devem obedecer ao trocar mensagens.

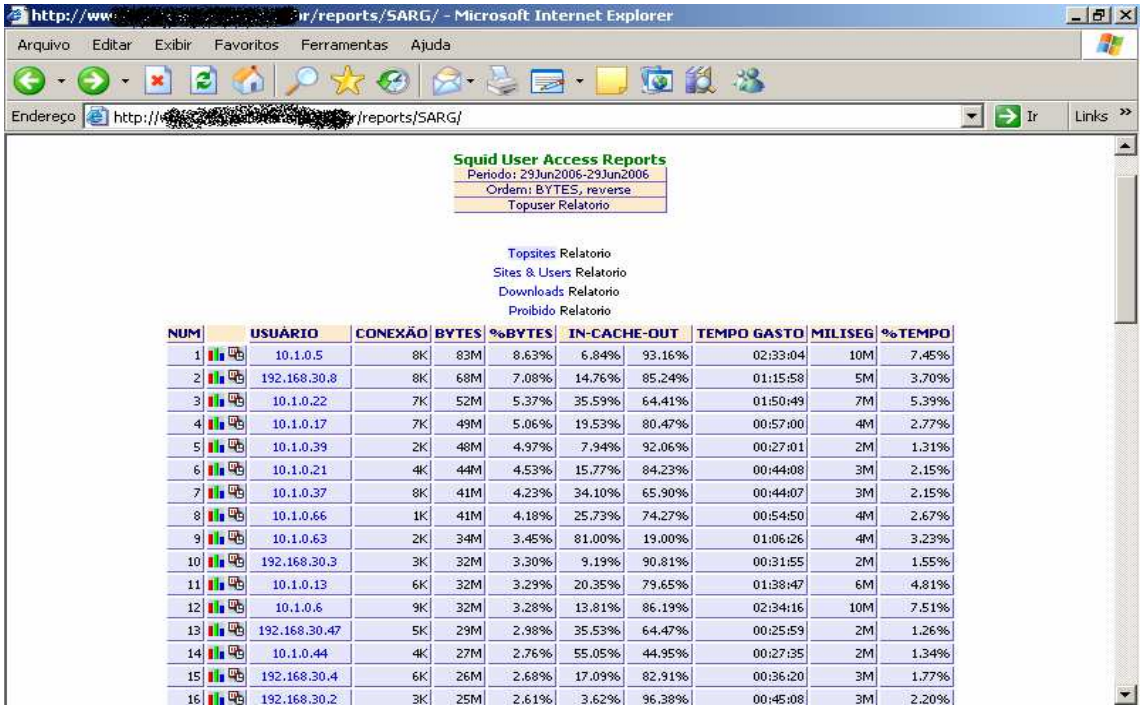
O serviço *http* foi criado para ser utilizado na transmissão de páginas e o *ftp* na transmissão de arquivos através do protocolo de comunicação. Basicamente, estes dois serviços compreendem mais de 90% da utilização da internet hoje.

Quando um destes serviços é acessado através de uma visita a um *website* (página da internet) ou através do *download* (recebimento) de um arquivo, uma requisição parte da máquina cliente até a máquina servidora. Geralmente, a distância entre os pontos é muito grande e/ou a qualidade das linhas de transmissão muito ruins fazendo com que a comunicação fique bem lenta. Levando em conta que as maiorias das informações resgatadas dos *websites* não mudam, ocorre desperdício de tempo toda vez que o *browser* tem que solicitar novamente uma imagem ou arquivo.

Para resolver estes problemas, utiliza-se do conceito de *cache* que é a capacidade de armazenar todas as informações mais recentes ou as mais acessadas. O Objetivo do servidor *proxy* é manter este *cache* além de outras funções como controle de banda e manter histórico de acessos por usuário.

Desta forma, quando uma requisição de um cliente passa pelo servidor *proxy*, o mesmo verifica na sua base de dados de arquivos em *cache* para verificar se existe uma cópia atual do objeto solicitado. Se existir a cópia, o servidor entrega imediatamente aquele objeto sem a necessidade de acessar o site, senão, ele terá que solicitar e aguardar o tempo que será gasto pelo tráfego da informação nas linhas de transmissão que talvez possam ser precárias.

Um software open source interessante é o SARG que é capaz de ler o arquivo de histórico do Squid e gerar informações importantes por período como quantidade de *downloads*, sites mais acessados, sites proibidos, porcentagem de acesso no *cache* (IN-CACHE), porcentagem de acesso fora do *cache* (OUT-CACHE) e tempo que o Proxy gastou com determinado computador (Figura 2). Através deste relatório é possível perceber que os usuários não são autenticados, pois ao invés do nome do usuário, aparece o endereço IP do computador. [SARG 2006]



NUM		USUÁRIO	CONEXÃO	BYTES	%BYTES	IN-CACHE-OUT		TEMPO GASTO	MILISEG	%TEMPO
1		10.1.0.5	8K	83M	8.63%	6.84%	93.16%	02:33:04	10M	7.45%
2		192.168.30.8	8K	68M	7.08%	14.76%	85.24%	01:15:58	5M	3.70%
3		10.1.0.22	7K	52M	5.37%	35.59%	64.41%	01:50:49	7M	5.39%
4		10.1.0.17	7K	49M	5.06%	19.53%	80.47%	00:57:00	4M	2.77%
5		10.1.0.39	2K	48M	4.97%	7.94%	92.06%	00:27:01	2M	1.31%
6		10.1.0.21	4K	44M	4.53%	15.77%	84.23%	00:44:08	3M	2.15%
7		10.1.0.37	8K	41M	4.23%	34.10%	65.90%	00:44:07	3M	2.15%
8		10.1.0.66	1K	41M	4.18%	25.73%	74.27%	00:54:50	4M	2.67%
9		10.1.0.63	2K	34M	3.45%	81.00%	19.00%	01:06:26	4M	3.23%
10		192.168.30.3	3K	32M	3.30%	9.19%	90.81%	00:31:55	2M	1.55%
11		10.1.0.13	6K	32M	3.29%	20.35%	79.65%	01:38:47	6M	4.81%
12		10.1.0.6	9K	32M	3.28%	13.81%	86.19%	02:34:16	10M	7.51%
13		192.168.30.47	5K	29M	2.98%	35.53%	64.47%	00:25:59	2M	1.26%
14		10.1.0.44	4K	27M	2.76%	55.05%	44.95%	00:27:35	2M	1.34%
15		192.168.30.4	6K	26M	2.68%	17.09%	82.91%	00:36:20	3M	1.77%
16		192.168.30.2	3K	25M	2.61%	3.62%	96.38%	00:45:08	3M	2.20%

Figura 2. Relatório gerado no SARG

4. Autenticação

A autenticação é uma medida de segurança projetada para estabelecer a confiabilidade de uma transmissão, mensagem, origem, ou para significar a verificação de uma autorização individual para receber categorias específicas de informação. A forma mais comum de autenticação é através do nome de usuário e senha, mas se estes dados não forem transmitidos encriptados (em código) outras pessoas na rede podem obter estes dados facilmente comprometendo assim a segurança.

Nas versões 2.5 e superiores do Squid há suporte para três tipos distintos de autenticação: *Basic*, *Digest* and NTLM (NT LAN Manager).

A autenticação *basic* é baseada na transferência do nome de usuário e senha. Ela não é um protocolo seguro de autenticação, pois a senha dos usuários é enviada simplesmente em codificação BASE64 sem criptografia. [RFC 1999]

A autenticação digest assim como a autenticação *basic*, é baseada na transferência do nome de usuário e senha. Mas, neste caso, a senha dos usuários pode ser transmitida através de um formulário encriptado que é mais seguro que a simples codificação de BASE64.

O NTLM é mecanismo proprietário desenvolvido pela Microsoft e atualmente suporta apenas clientes de software Microsoft. Assim, com a autenticação digest, permite uma conexão encriptada com o servidor, mas ainda é menos segura que a autenticação *digest* e possui histórico de vulnerabilidades.

Os módulos de autenticação existentes para o Squid encontrados na literatura foram os seguintes:

LDAP> Autentica os usuários em banco de dados LDAP(Lightweight Directory Access Protocol). [OPENLDAP 2006]

MSNT> Autentica em um domínio Microsoft NT. [LIMA 2005]

PAM> O PAM (Módulo de Autenticação Plugável) permite usar uma variedade de sistemas de autenticação. Ele funciona como uma interface entre aplicação e a entidade autenticadora que pode ser um arquivo de senha, um banco de dados ou até mesmo um dispositivo de hardware.

SMB_AUTH> Autentica em servidor SMB (Windows NT ou Samba). Como este módulo serve pra autenticar em domínio NT, ele será colocado em prática neste artigo.

GETPWNAN> Autentica através dos nomes de usuários e senhas do sistema linux que por padrão estão nos arquivos *passwd* (usuários) e *shadow* (senha encriptada).

NCSA> Autentica em simples arquivos de senhas gerados pelo utilitário *htpasswd* que vem junto com o pacote do apache(servidor WEB). NCSA(National Center for Supercomputing Applications) é também o nome dado a um servidor WEB que deu origem ao famoso apache.

5. Recursos necessários

Para que seja possível implementar o assunto explanado é necessário um servidor Proxy através do software Squid e um controlador de domínio da plataforma Linux ou Windows. Na plataforma linux o serviço de controlador de domínio é exercida pelo samba [SAMBA 2006], e na plataforma Windows pelos sistemas operacionais seguintes: Windows NT, Windows 2000 Server e Windows 2003 Server [MICROSOFT 2006].

O módulo autenticador *smb_auth* é obtido através da instalação de um pacote específico de cada distribuição linux ou através da compilação do código fonte fornecido pelo desenvolvedor [SMB_AUTH 1999].

5.1. Configuração do Squid

O pacote de instalação do Squid está na maioria das distribuições linux, portanto para instalá-lo basta estar conectado à internet ou ter em mãos todos os cd-roms e digitar:

apt-get install squid

Após o término da instalação, o arquivo de configuração *default* do squid já estará disponível em */etc/squid/squid.conf*. Este arquivo comanda toda a funcionabilidade do Squid, portanto, ele será o alvo das atenções. Além do Squid, será necessário instalar o módulo autenticador que irá receber uma solicitação do Squid,

fazer a interface entre o cliente e o controlador de domínio passando o resultado desta interação de volta para o Squid.

O módulo encontra-se no site do desenvolvedor[SMB_AUTH 1999]. Seguindo as instruções do site será possível compilar e instalar o *smb_auth* sem problemas. No Debian linux este módulo já vem com o pacote do Squid e no Conectiva Linux basta digitar:

apt-get install squid-auth

Como exemplo, vamos considerar que o PDC do domínio Windows tem o endereço IP 10.1.1.1, e o nome do domínio é DOMINIO. Desta forma, é necessário adicionar a seguinte linha abaixo no arquivo *squid.conf*:
authenticate_program /usr/local/squid/smb_auth -W DOMINIO -U 10.1.1.1

A documentação do *smb_auth* informa que é preciso incluir um arquivo com nome *proxyauth* na pasta netlogon do PDC para que possa ser acessada com permissão de leitura para qualquer usuário que possa acessar os serviços do proxy Squid. Este arquivo deve conter simplesmente a palavra *allow* e será acessado pela rede através do seguinte nome de compartilhamento *\\NomeNetbiosDoPDC\netlogon\proxyauth*.

A seguir uma noção da configuração básica do arquivo de configuração do Squid no que diz respeito à autenticação e a definição de acesso:

Definindo uma ACL[2] (*Access Control Lists*) chamada *autenticar*, do tipo *proxy_auth*
(ou seja, haverá autenticação utilizando a *tag authenticate_program*). O parâmetro
REQUIRED indica que qualquer usuário será aceito, desde que ele tenha uma senha no domínio.
acl autenticar proxy_auth REQUIRED

Definindo uma ACL chamada *usuarios*, também do tipo *proxy_auth*, mas desta vez
indicando os nomes de 2 usuários que terão tratamento diferenciado.
acl usuarios proxy_auth User1 User2

Estas ACLs não tem função na autenticação, mas definem uma ACL *horario*, como
sendo o período entre 07:00 e 22:00 e uma ACL chamada *todos*, que representa todos
os endereços IP de origem possíveis.
acl horario time 07:00-22:00
acl todos src 0/0

Definindo uma ACL chamada *negado*, do tipo *dstdomain* relacionando nomes de
domínio que se desejam bloquear
acl negado dstdomain porno.net

Definindo uma ACL chamada *excecao*, onde serão relacionados os usuários que
serão exceção à regra *negado*.
acl excecao proxy_auth chefe

Permitindo que os componentes da ACL *excecao* acessem os componentes da ACL
negado.
http_access allow excecao negado

---

[2] *ACL*(*Access Control Lists*) são listas de controle de acesso usadas pelo Squid para definir regras

Negando acesso a todos os sites relacionados na ACL *negado* que não foram liberados
até o momento com a regras das linhas anteriores(*exceção)*. Assim, se um usuário não
caiu na regra anterior, ele simplesmente não poderá acessar o site *www.porno.net*.
http_access deny negado

Permitindo acesso aos componentes da ACL *usuarios*. *User1* e *User2* vão poder
utilizar o Proxy independente do horário, desde que informem o *login* e a senha
corretamente.
http_access allow usuarios

Assegurando que qualquer usuário(exceto os da ACL *usuarios*, já cobertos pela regra
anterior) que tente acessar seja autenticado e que esteja preso ao horário estipulado.
Se o usuário tentar acessar fora deste horário ele será barrado.
http_access allow autenticar horario

Negando acesso a todos. Assim, se um usuário não caiu nas regras anteriores, ele
simplesmente não poderá usar o proxy.
http_access deny todos

Após efetuar as alterações no arquivo de configurações, o processo do Squid deve reler o arquivo com a seguinte linha de comando:
squid –k reconfigure

5.2. Configuração dos clientes

Para adicionar um computador cliente em um domínio existente deve-se acessar o "Meu Computador > Propriedades > Identificação de rede > Propriedades" e colocar o nome do computador e o nome do domínio, que é definido no PDC (Figura 3).

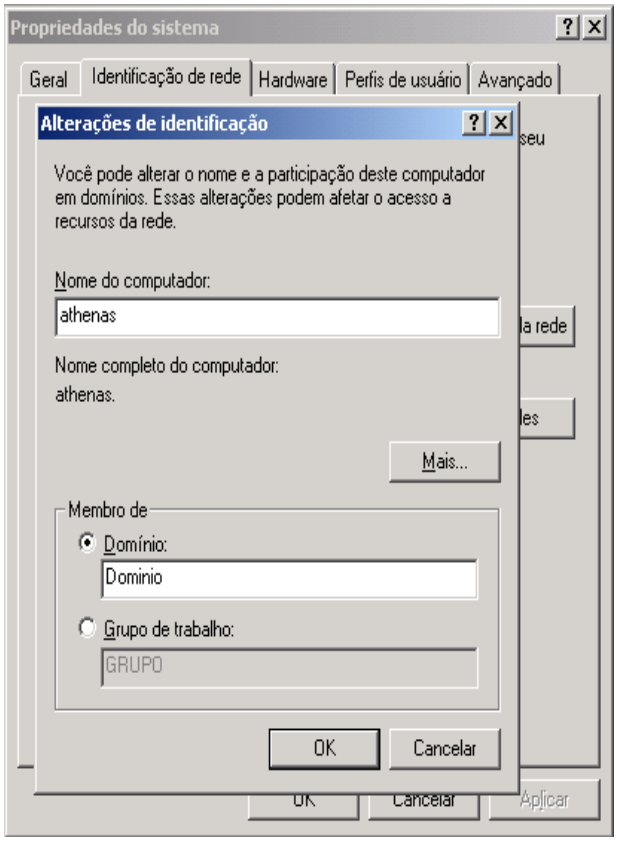


Figura 3. Propriedades do sistema

Na tela de identificação que é aberta a seguir, será solicitado o login e a senha do administrador do domínio para a inserção do computador *athenas* no domínio DOMINIO. É normal que a conexão inicial demore uns 30 segundos. Se tudo der certo, irá aparecer a mensagem de sucesso no ingresso ao domínio (Figura 4).

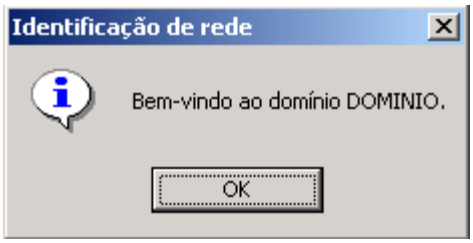


Figura 4. Mensagem de ingresso ao domínio

Para configurar a navegação de um cliente baseado em um servidor proxy com autenticação, deve-se observar as configurações do Proxy no Internet Explorer (Figura 5). No campo endereço, utiliza-se o endereço IP do servidor proxy e a porta TCP onde este serviço de proxy é oferecido. Como o servidor proxy em questão autentica em domínio NT, quando este computador for acessar WEB, primeiramente ele passa pelo servidor proxy, o qual irá iniciar o processo de autenticação enviando um formulário solicitando *login* e senha no primeiro acesso de cada instância do *browser* aberto pelo usuário.

Após a primeira autenticação o usuário fica livre de ter que autenticar novamente enquanto o browser estiver aberto.

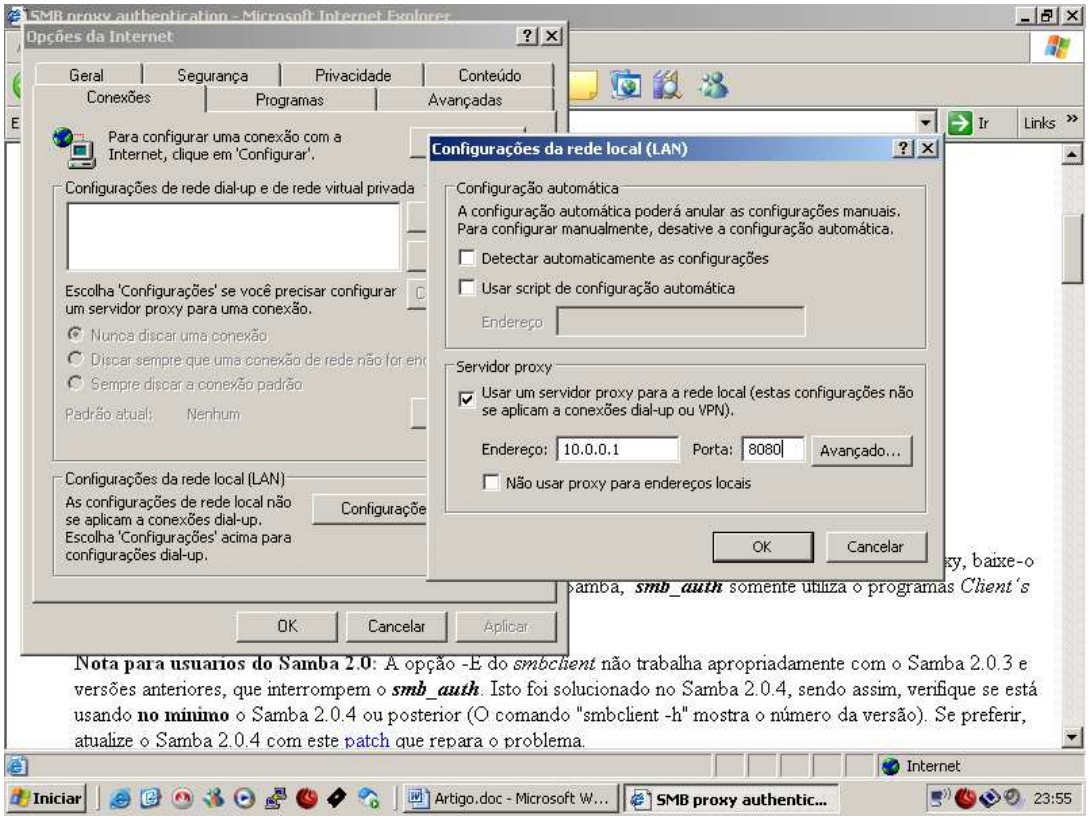


Figura 5. Configurações de Proxy

6. Considerações finais

Os resultados obtidos demonstram que é possível de forma acessível a todos utilizar o controle de acesso a usuários com poucas linhas de configuração. Todos os usuários autenticados permite a geração de relatórios específicos e um controle de sites inadequados por qualquer que seja o motivo.

Todos os módulos possuem um problema grave de segurança quando usam a autenticação na forma *basic*, pois tanto o nome do usuário como a senha trafega sem criptografia pela rede local. Os módulos que autenticam em domínio NT são os mais perigosos, pois a senha de acesso à internet é a mesma utilizada para acessar a todos os computadores da rede.

O Squid, a partir da versão 2.5 permite a autenticação na forma *digest*, o qual impede que a senha trafegue na rede sem criptografia.

Este trabalho poderá ser utilizado mais adiante como idéia para a autenticação de outros serviços como POP3 e SMTP, que são serviços responsáveis pelo envio e recebimento de e-mails.

7. Referências Bibliográficas

[VESPERMAN 2001] Vesperman, Jennifer. **Autenticação e o Squid**. Disponível em: http://geocities.yahoo.com.br/cesarakg/AuthenticationAndSquid.html. Acesso em 15 fev. 2006

[LIMA 2005] Lima, Willian da Rocha. **Instalando Squid 2.5 e Autenticando no Active Directory**. Disponível em: http://www.linuxit.com.br/section-printpage-670.html. Acesso em 15 fev. 2006

[SQUID 2006] Proxy Squid. **Squid Web Proxy Cache**. Disponível em: http://www.squid-cache.org. Acesso em 16 fev. 2006

[OPENLDAP 2006] OpenLDAP Foundation. **Introduction to OpenLDAP Directory Services**. Disponível em: http://www.openldap.org/doc/admin23. Acesso em 26 fev. 2006

[SMB_AUTH 1999] Módulo de autenticação em Proxy. **SMB Proxy Authentication**. Disponível em: http://www.hacom.nl/~richard/software/smb_auth_pt_br.html. Acesso em 12 fev. 2006

[CAMPOS 2003] Campos, Augusto César. **Autenticação no Squid**. Disponível em: http://br-linux.org/tutoriais/000402.html. Acesso em 25 fev. 2006

[LLC 2002] LLC, SecurityGlobal.net. **Squid Proxy Caching Server Msntauth Authentication Module Format String Hole Lets Remote User Execute Arbitrary Code on the Server**. Disponível em: http://www.securitytracker.com/alerts/2002/Jun/1004446.html. Acesso em 03 mar. 2006

[RFC 1999] RFC, Request for Coments. **HTTP Authentication: Basic and Digest Access Authentication**. Disponível em: http://www.ietf.org/rfc/rfc2617.txt?number=2617. Acesso em 20 mar. 2006

[JUNIOR 2003] Junior, Alceu Rodrigues de Freitas. **Squidnomicon**. Disponível em: http://www.geocities.com/glasswalk3r/linux/squidnomicon-online.html. Acesso em 24 mar. 2006

[BASTOS 2004] Bastos, Eri Ramos. **Configurando um Squid Ninja**. Disponível em: http://www.linuxman.pro.br/squid. Acesso em 25 mar. 2006

[MICROSOFT 2006] Microsoft, Corporation. **Installing Windows Server 2003 as a Domain Controller**. Disponível em: http://www.microsoft.com/technet/prodtechnol/windowsserver2003/technologies/directory/activedirectory/stepbystep/domcntrl.mspx. Acesso em 25 mar.2006

[SAMBA 2006] SMB file/printer Server for Unix. **Opening Windows to a Wider World**. Disponível em: http://www.samba.org – Acesso em 25 mar. 2006

[SARG 2006] Gerador de relatórios. **Squid Analysis Report Generator**. Disponível em: http://sarg.sourceforge.net. Acesso em 04 abr. 2006

[IANA 2006] The Internet Corporation for Assigned Names and Numbers. **Port Numbers**. Disponível em: http://www.iana.org/assignments/port-numbers. Acesso em 04 abr. 2006

[MARCELO 2003] Marcelo, Antônio. **Guia Rápido do Administrador de Redes Squid:** Configurando o Proxy para linux. Rio de Janeiro: Brasport, 2003. 80 p.

[SATOMI et al. 2006] Satomi, Emerson; Silva, Rafael Peregrino; Rigues, Rafael Pereira. **Linux Magazine**. São Paulo: Linux New Media do Brasil, 2006. 98 p.