

TIAGO RODRIGUES CHAVES

Orientador: Ricardo Augusto Rabelo de Oliveira

**ESTUDO DE CASO: AUTENTICAÇÃO IEEE 802.1X
BASEADA NO PROTOCOLO RADIUS E SERVIÇO DE
DIRETÓRIO LDAP APLICADO A REDE GIGAUFOPNET**

Ouro Preto
Dezembro de 2010

UNIVERSIDADE FEDERAL DE OURO PRETO
INSTITUTO DE CIÊNCIAS EXATAS
BACHARELADO EM CIÊNCIA DA COMPUTAÇÃO

**ESTUDO DE CASO: AUTENTICAÇÃO IEEE 802.1X
BASEADA NO PROTOCOLO RADIUS E SERVIÇO DE
DIRETÓRIO LDAP APLICADO A REDE GIGAUFOPNET**

Monografia apresentada ao Curso de Bacharelado em Ciência da Computação da Universidade Federal de Ouro Preto como requisito parcial para a obtenção do grau de Bacharel em Ciência da Computação.

TIAGO RODRIGUES CHAVES

Ouro Preto
Dezembro de 2010



UNIVERSIDADE FEDERAL DE OURO PRETO

FOLHA DE APROVAÇÃO

ESTUDO DE CASO: AUTENTICAÇÃO IEEE 802.1X BASEADA NO
PROTOCOLO RADIUS E SERVIÇO DE DIRETÓRIO LDAP
APLICADO A REDE GIGAUFOPNET

TIAGO RODRIGUES CHAVES

Monografia defendida e aprovada pela banca examinadora constituída por:

Dr. RICARDO AUGUSTO RABELO DE OLIVEIRA – Orientador
Universidade Federal de Ouro Preto

Bel. VITOR EMANUEL RODRIGUES DE ARAÚJO
Universidade Federal de Ouro Preto
Diretor - NTI/UFOP

Bel. ABELARD RAMOS FERNANDES
Universidade Presidente Antonio Carlos - UNIPAC
Gerente de Infraestrutura Computacional - NTI/UFOP

Ouro Preto, Dezembro de 2010

Resumo

Este trabalho, pretende com a utilização do protocolo IEEE 802.1X integrado com servidores RADIUS e LDAP, estudar e implantar autenticação, autorização e contabilidade do uso, incorporando a segurança necessária para o acesso a rede GigaUFOPnet.

Palavras-chave: Autenticação. 802.1x. Redes. Radius. LDAP.

Abstract

This work intends to use the IEEE 802.1X protocol integrated with RADIUS and LDAP servers, study and implement authentication, authorization and accounting of the use, incorporating the necessary security for network access GigaUFOPnet.

Palavras-chave: Authentication. 802.1x. Networks. Radius. LDAP

*O que não se conhece não se pode controlar.
O que não se controla não se pode mensurar.
O que não se mensura não se pode gerenciar.
O que não se gerencia não se pode aprimorar.
by Marcos Sêmola*

Agradecimentos

Agradeço a todos que, de alguma forma, contribuíram para a realização deste trabalho, em especial:

A Deus por ter guiado meu caminho até hoje.

Aos meus pais, Délcio e Elaine, por todo o apoio e força que me deram, não somente neste momento, mas em toda minha vida.

A minha irmã Tamires e meu irmão Tácio, pelo carinho.

A meus familiares e amigos que sempre estiveram ao meu lado, me incentivando.

Aos meus amigos do NTI, em especial a toda equipe de redes, por todos os ensinamentos e experiência profissional compartilhada durante esses anos.

Ao Prof. Ricardo Rabelo, meu orientador, pelos ensinamentos e conselhos.

Aos meus irmãos da República Partenon, pelos momentos inesquecíveis em Ouro Preto.

Sumário

1	INTRODUÇÃO	1
1.1	Contextualização	1
1.2	Objetivos	2
1.2.1	Objetivo Geral	2
1.2.2	Objetivos Específicos	2
1.3	Justificativa e Relevância	3
1.4	Estrutura do Trabalho	3
2	REVISÃO BIBLIOGRÁFICA	4
2.1	RADIUS	4
2.1.1	Histórico do RADIUS	4
2.1.2	Um Overview da Arquitetura AAA.	5
2.1.3	Descrição do Funcionamento	6
2.1.4	Soluções de Servidores RADIUS Existentes	7
2.2	IEEE 802.1X	8
2.2.1	Como funciona o padrão IEEE 802.1X	8
2.2.2	Extensible Authentication Protocol - EAP	10
2.3	LDAP	12
2.3.1	Histórico do LDAP	12
2.3.2	Protocolo LDAP	13
2.3.3	Serviço de Diretório <i>versus</i> Banco de Dados Relacional	14
2.3.4	Vantagens e Desvantagens do LDAP	15
2.4	SAMBA	16
2.4.1	Histórico do SAMBA	16
2.4.2	Controlador de Domínio (PDC)	16
3	METODOLOGIA	18
3.1	Desenvolvimento	19
3.1.1	Sistema Operacional	20
3.1.2	Integrando SAMBA ao OpenLdap	21

3.1.3	FreeRadius	22
3.1.4	Testando a configuração do Radius	24
3.1.5	Configuração dos Switchs	26
3.1.6	DHCP	27
3.1.7	Configuração dos Clientes 802.1X	29
3.1.8	Ferramentas de Administração	38
4	CARACTERIZAÇÃO DO PROBLEMA	42
4.1	Contextualização	42
4.2	Situação Atual da GigaUFOPnet	44
4.2.1	Unidades e Campi interligados pela GigaUFOPnet	44
4.2.2	Mapa de switches GigaUFOPnet	45
4.2.3	Usuários da GigaUFOPnet	46
4.2.4	Método Atual para Acessar a GigaUFOPnet	47
4.2.5	Cenários Comuns Sem a Implantação do Projeto	48
4.3	Situação Desejada	51
5	RESULTADOS	53
6	CONCLUSÕES E TRABALHOS FUTUROS	55
	APÊNDICE	57
A	Arquivos de Configuração do SAMBA	57
A.1	smb.conf	57
B	Arquivos de Configuração Integração SAMBA-LDAP	61
B.1	smbldap_bind.conf	61
B.2	smbldap.conf	62
C	Arquivos de Configuração do LDAP	64
C.1	ldap.conf	64
C.2	sldap.conf	65
D	Arquivos de Configuração do FreeRadius	67
D.1	radius.conf	67
D.2	dictionary	79
D.3	dictionary.3com	81
D.4	dictionary.tunnel	82
D.5	ldap.attrmap	84
D.6	users	86

D.7	clients.conf	87
D.8	proxy.conf	88
D.9	sql.conf	90
D.10	eap.conf	94
Referências Bibliográficas		96

Lista de Figuras

2.1	Estabelecimento de uma sessão radius	7
2.2	Infraestrutura para operar com 802.1X	9
2.3	Diálogo entre os componetes	11
2.4	Evolução do DAP para o LDAP	12
3.1	Instalação do CentOS 5.5	20
3.2	NTRadPing RADIUS Test Utility	25
3.3	Iniciando o Serviço Configuração Automática de Rede com Fio	29
3.4	Acesso ao Painel de Controle	30
3.5	Acesso aos Recursos de Rede	31
3.6	Acessando os Parâmetros TCP/IP	32
3.7	Configuração da Autenticação	33
3.8	Identificando a rede sem fio	34
3.9	Configurando a rede sem fio	35
3.10	Habilitado rede wireless com 802.1X	36
3.11	Fornecendo as Credenciais e Indicativo do Resultado	37
3.12	Logs do Processo	37
3.13	Comparação ente softwares suplicantes	38
3.14	Iniciando o Webmin	39
3.15	object class e atributos	40
3.16	Usuario Cadastrado	40
3.17	Acessando o daloRADIUS	41
4.1	Exemplo de Identificação e Autenticação.	43
4.2	Mapa de switches da GigaUFOPNet	45
4.3	Usuário cadastrado e conectado a rede através de um computador cadastrado	48
4.4	Usuário instala Hub/Switch para aumentar o número de pontos de acesso	49
4.5	Usuário isntala Acess Point	50
4.6	Situação desejada	51
5.1	Accounting do usuários após a conexão utilizando 802.1X	54

Lista de Tabelas

3.1	Pacotes necessários para instalação do SAMBA	21
3.2	Pacotes necessários para instalação do OpenLdap	21
3.3	Pacotes necessários para instalação do FreeRadius	22
3.4	Comando radiusd -X	24
3.5	Comando radtest	25
3.6	Log registrado pelo NTRadPing RADIUS Test Utility	26
3.7	Switch 3COM - Configuração de domínio	26
3.8	Switch 3COM - 802.1X como configuração global	26
3.9	dhcpd.conf	28
4.1	Unidades do Campus Ouro Preto	44
4.2	Unidades do Campus Mariana	44
4.3	Unidades do Campus João Monlevade	44
4.4	Número de Usuários da GigaUFOPnet em 2010	46

Lista de Siglas e Abreviaturas

AAA *Authentication, Authorization, and Accounting* – autenticação, autorização e contabilização.

CHAP: *Challenge Handshake Authentication Protocol* – protocolo de autenticação por negociação de desafio.

DHCP: *Dynamic Host Configuration Protocol* – protocolo de configuração dinâmica de terminais.

EAP: *Extensible Authentication Protocol* – protocolo de autenticação extensível.

EAPOL: *Extensible Authentication Protocol over LAN.*

GigaUFOPnet: Rede de computadores da Universidade Federal de Ouro Preto.

IEEE: *Institute of Electrical and Electronic Engineers.*

IETF: *Internet Engineering Task Force.*

IP *Internet Protocol* – protocolo de internet.

LAN: *Local Area Network* – rede de área local.

LCP: *Link Control Protocol* – protocolo para controle de link.

LDAP: *Lightweight Directory Access Protocol* – protocolo leve de acesso a diretórios.

MAC: *Media Access Control*

NAS: *Network Access Server* – servidor de acesso à rede.

NTI: Núcleo de Tecnologia da Informação - UFOP.

PAP: *PAP - Password Authentication Protocol* – protocolo de autenticação por senha

PPP: *Point-to-Point Protocol* – protocolo ponto a ponto.

RADIUS: *Remote Authentication Dial-in User Service.*

UFOP: Universidade Federal de Ouro Preto.

WAN: *Wireless Local Area Network* – rede de área local sem fio.

Capítulo 1

INTRODUÇÃO

1.1 Contextualização

Este trabalho tem seu foco no estudo do problema da autenticação dos usuários em uma rede de computadores, principal aspecto para a segurança segundo Barrosi e Foltran (2008). A dispensa de um procedimento de autenticação pode tornar a rede de computadores vulnerável ao uso mal intencionado de usuários não autorizados. A autenticação permite reconhecer, autorizar e contabilizar o acesso do usuário a infraestrutura de rede, aos sistemas computacionais e a Internet et al (2009).

Durante os últimos anos as redes de computadores tiveram um grande crescimento principalmente devido a popularização da Internet Balbonil (2006). Entretanto, a infraestrutura de redes de computadores necessita de atualizações para suportar novos protocolos para permitir seu gerenciamento e controle sobre os usuários da mesma.

Com a ampliação da rede de computadores da Universidade Federal de Ouro Preto, denominada GigaUFOPnet, interligando diversas unidades e campi da UFOP, a necessidade de autenticação dos usuários é o principal aspecto para a segurança, dessa forma, torna-se imprescindível o controle de acesso à rede e uma arquitetura de controle centralizada que se integre com o padrão AAA (Authentication, Authorization and Accounting) da IETF (Internet Engineering Task Force) definido em VOLLBRECHT (2000).

Este trabalho pretende com a utilização do padrão IEEE 802.1X integrado com servidores RADIUS e LDAP, estudar e implantar autenticação, autorização e contabilidade do uso, incorporando a segurança necessária para o acesso à rede GigaUFOPnet.

1.2 Objetivos

Nesta seção são apresentados os objetivos deste trabalho.

1.2.1 Objetivo Geral

O objetivo geral deste trabalho é buscar um avanço na estrutura atual da GigaUFOPnet, acrescentando o serviço de autenticação para garantir que apenas pessoas ou computadores autorizados possam acessar a infraestrutura de rede, os sistemas computacionais e a Internet.

1.2.2 Objetivos Específicos

Os objetivos específicos a serem atingidos são:

- (a) Garantir desempenho e confiabilidade do acesso a rede GigaUFOPnet;
- (b) Autenticar todos os usuários da GigaUFOPnet no momento da sua conexão;
- (c) Implantar políticas de acesso à rede de acordo com o papel de cada usuário;
- (d) Permitir ao usuário acesso a sua rede local e privilégios a partir de qualquer ponto da rede;
- (e) Contabilizar todos os acessos e uso dos usuários;
- (f) Identificar após a conexão, usuários mal intencionados utilizando a rede;
- (g) Disponibilizar acesso a rede para dispositivos móveis com segurança;
- (h) Utilizar VLANs dinâmicas;
- (i) Melhorar a utilização das faixas de IP.

1.3 Justificativa e Relevância

O controle de acesso visa proteger a GigaUFOPnet de ameaças de segurança especificando e controlando quem poderá acessar a rede, além de coletar as informações relacionadas à utilização, pelos usuários, dos recursos de um sistema. Esta informação pode ser utilizada para gerenciamento, planejamento, cobrança e responsabilização do usuário.

Com a implementação deste trabalho na estrutura atual da GigaUFOPnet, além da segurança incorporada, torna-se possível a criação de uma rede wireless, atendendo a demanda crescente para acomodar as necessidades de mobilidade dos usuários de acessarem serviços eletrônicos, conteúdos pedagógicos e a Internet a partir de locais além das salas de aula, laboratórios e escritórios.

A conclusão deste trabalho traz para todos os usuários da rede GigaUFOPnet a utilização de um serviço de comunicação confiável, melhorando o fluxo de acesso as informações e proporcionando segurança na comunicação.

1.4 Estrutura do Trabalho

O presente trabalho está dividido em seis capítulos, incluindo esta introdução, onde o problema da autenticação em redes de computadores é apresentado, além dos objetivos do trabalho, a relevância do problema e a motivação de se propor uma solução para resolvê-lo.

No capítulo 2, são apresentadas de acordo com os conceitos da literatura, as descrições das tecnologias que serão utilizadas para o desenvolvimento do trabalho, tais como, o padrão 802.1X, o protocolo RADIUS, o servidor de diretório LDAP e servidor de domínios SAMBA.

No capítulo 3, o problema da autenticação, objeto de estudo deste trabalho, é contextualizado. Apresentamos a situação atual da GigaUFOPnet e a situação que deseja-se alcançar após a conclusão do trabalho.

No capítulo 4 é descrito toda a metodologia adotada para o desenvolvimento do trabalho.

No capítulo 5 são apresentados os resultados obtidos e as experiências encontradas.

No capítulo 6 são apresentadas as conclusões e os trabalhos futuros.

Capítulo 2

REVISÃO BIBLIOGRÁFICA

Neste capítulo é apresentado uma breve descrição com conceitos básicos encontrados na literatura, que se relacionam com as tecnologias utilizadas para o desenvolvimento desse projeto, bem como o protocolo RADIUS, o padrão IEEE 802.1X, o servidor de diretório LDAP e o controlador de domínio SAMBA.

2.1 RADIUS

Imagine a Internet utilizada hoje sem nenhum tipo de controle, qualquer pessoa poderia, por exemplo, acessar o e-mail da outra, transferir dinheiro da conta bancária, fazer compras sem pagar, numa desordem total. Ao acessar uma máquina, um e-mail, uma conta bancária certamente você passará por algum tipo de autenticação. Uma das formas mais conhecidas de autenticação é através do login e senha, no qual você precisa dizer ao computador quem é você e só então ele verifica a sua autenticidade, checa quais são as suas permissões naquele sistema e entrega tudo aquilo que tem direito. Todo esse processo pode resume-se a um protocolo chamado RADIUS.

2.1.1 Histórico do RADIUS

O RADIUS é um protocolo utilizado para disponibilizar acesso a redes utilizando a arquitetura AAA. Inicialmente foi desenvolvido para uso em serviços de acesso discado. Atualmente é também implementado em pontos de acesso sem fio e outros tipos de dispositivos que permitem acesso autenticado a redes de computadores. O protocolo RADIUS é definido pela RFC 2865 et al (2000).

O desenvolvimento do RADIUS começou de fato em 1994, quando Steve Willens e Carl Rigney da Livingston Enterprise (hoje conhecida como Lucent) abriram o código fonte do servidor RADIUS para que outros desenvolvedores da Merit Networks, uma pioneira em cri-

ação de soluções para Internet na época, pudessem ajudar na construção do que hoje é um dos serviços mais utilizados na rede.

O RADIUS foi construído para atender uma necessidade de mercado da época. Naquele tempo, precisava-se de um produto que autenticasse, autorizasse e fizesse acompanhamento e monitoramento do uso de recursos de rede usado pelos usuários. Depois de uma primeira reunião entre os desenvolvedores dessas duas empresas, nasceu uma versão muito superficial do RADIUS. Hoje em dia, tanto a Lucent quanto a Merit Networks oferecem serviços de Internet ao público baseado no RADIUS sem nenhum tipo de cobrança.

2.1.2 Um Overview da Arquitetura AAA.

O RADIUS é construído em cima de um processo denominado AAA, que consiste em autenticação, autorização e accounting (acompanhamento/monitoramento do uso de recursos de rede pelo usuário). Apesar de o RADIUS ter sido construído antes da arquitetura AAA ser elaborada, ele foi o primeiro protocolo baseado na arquitetura AAA que mostrou de fato suas funções as indústrias da época. Este modelo monitora todos os passos do usuário, do começo ao fim da conexão Hassell (2002).

Para entendermos melhor como funciona esse processo, veja abaixo os passos que o RADIUS percorre:

1. Autenticação do usuário;
2. Autorização de serviços;
3. Monitoramento dos serviços fornecidos.

Veremos agora em detalhes cada passo acima:

1. Autenticação

A autenticação é o processo que verifica se uma combinação de login e senha são válidas para o sistema. O login poderá ser uma conta de usuário, uma conta de máquina, um certificado digital, etc. Por exemplo, quando você acessa a Internet para navegar na web, geralmente o método utilizado para se fazer autenticação é através de um login *ID* e uma senha, método de autenticação mais conhecido por todos.

2. Autorização

Autorização é o passo seguinte após a identificação do usuário. Agora o sistema irá verificar quais são os privilégios que ele terá naquele sistema. Por exemplo, em uma empresa de Tecnologia da Informação (TI) existem diferenças de perfis de usuário, alguns têm acesso a todo o sistema e outros apenas privilégios selecionados, a escolha do tipo de privilégio é feita pelo administrador da rede quando configurável ou pelo próprio

sistema operacional que define as políticas padrões de acesso. Aqui o servidor AAA irá fazer uma série de processos e análises para saber quem é quem, saber quais direitos e permissão de acesso o usuário tem.

3. Contabilização

O processo de monitoramento/gerenciamento de recursos de rede utilizado pelo usuário é chamado de *accounting*. Aqui o sistema acompanha cada passo dado por ele na hora da utilização dos serviços de rede. Este serviço é bastante utilizado por provedores de banda larga que cobram do usuário por exceder o limite de banda, a cada n bytes ultrapassados é cobrada uma taxa a mais na mensalidade do usuário. Esse serviço também pode ser utilizado para comprovar violações cometidas pelo usuário, uma vez que qualquer evento que ocorra durante a conexão entre o usuário e o servidor, é registrado num arquivo de log.

2.1.3 Descrição do Funcionamento

O protocolo RADIUS baseia-se no modelo cliente/servidor, tendo de um lado o *Network Access Server* - NAS como cliente e do outro o servidor RADIUS. O utilizador, o NAS e o servidor trocam mensagens entre si quando o utilizador pretende se autenticar para utilizar um determinado servidor de rede.

Uma mensagem RADIUS consiste num pacote contendo um cabeçalho RADIUS com o tipo de mensagem, podendo ainda ter atributos associados à mensagem, cada atributo RADIUS especifica uma parte de informação sobre a tentativa de ligação. Por exemplo, existem atributos RADIUS para o nome de utilizador, para a palavra passe do utilizador, para o tipo de serviço pedido pelo utilizador e para o endereço *Internet Protocol* - IP do servidor de acesso.

Os atributos RADIUS são utilizados para transmitir informações entre clientes RADIUS, proxies RADIUS e servidores RADIUS. Quando um utilizador da rede deseja utilizar um serviço ele envia os seus dados para o NAS.

O NAS é responsável por adquirir todos os dados do utilizador, que normalmente são o nome de utilizador e a respectiva palavra passe (no envio do NAS para o servidor a palavra passe é cifrada de modo a prevenir possíveis ataques) e enviá-los para o servidor RADIUS através de um pedido de acesso que se designa de *Access-Request*. Este é também responsável por processar respostas vindas do servidor RADIUS.

O servidor ao receber um pedido de acesso tenta a autenticação do utilizador, enviando em seguida a resposta para o NAS contendo um *Access-Reject* caso o acesso seja negado, *Access-Accept* caso o acesso seja aceito ou *Access-Challenge* caso seja pedida uma nova confirmação.

Após autenticação, são comparados e verificados alguns dados do pedido de modo que o servidor determine qual o grau de acesso pode ser dado a este utilizador que foi autenticado. O servidor RADIUS pode também ser configurado em proxy. Neste caso o servidor funcionará como cliente que redireciona os pedidos de acesso para outro servidor, ou seja, passa a ser responsável pela troca de mensagens entre o NAS e o servidor remoto.

A figura 2.1 ilustra a descrição do funcionamento do protocolo RADIUS.

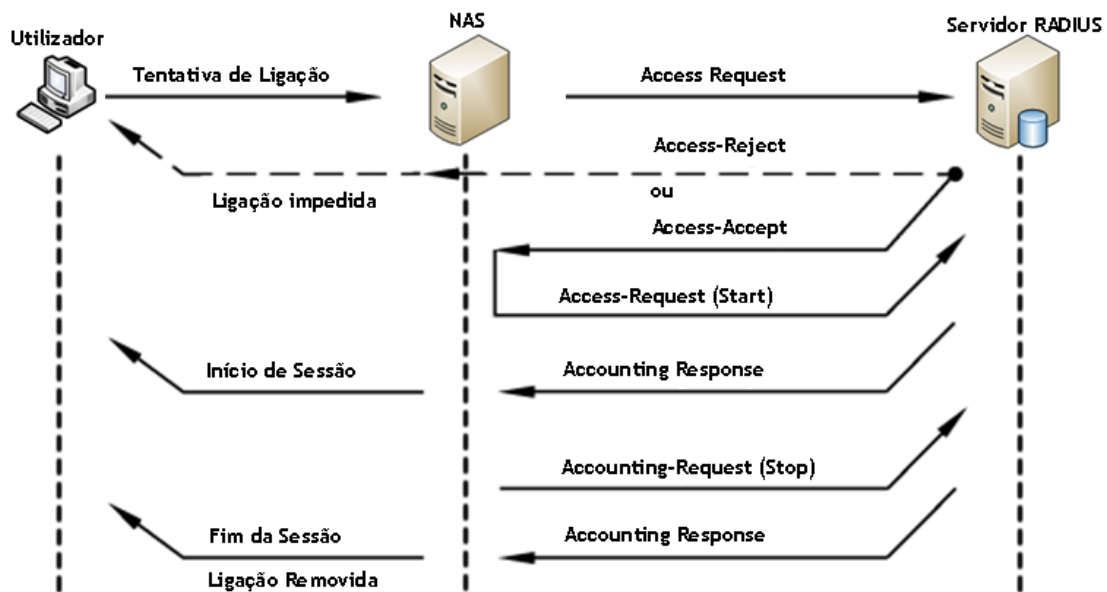


Figura 2.1: Estabelecimento de uma sessão radius

2.1.4 Soluções de Servidores RADIUS Existentes

Existem no mercado muitas soluções para servidores RADIUS. O FreeRadius é o servidor RADIUS mais utilizado para sistemas Linux. Este é responsável pela autenticação de pelo menos um terço dos usuários da Internet. Os restantes dos usuários encontram-se divididos entre os servidores restantes, destacando-se entre eles o Cisco Access Control Server (ACS) e o Microsoft Internet Authentication Service (IAS).

FreeRadius: O FreeRadius é uma implementação de RADIUS modular, de alta performance e rica em opções e funcionalidades. Esta inclui servidor, cliente, bibliotecas de desenvolvimento e muitas outras utilidades. Pode ser instalada em sistemas Linux e Machintosh FreeRadius (2010). Devido a estas características e tendo em conta o fato de ser uma aplicação open source esta será a implementação de RADIUS utilizada para o desenvolvimento do trabalho.

Cisco ACS: O Cisco Secure Access Control Server é uma política de controle de acessos que proporciona um ambiente centralizado de controle de autenticação, autorização e contabi-

lização de acesso de usuários de redes de computadores ACS (2010).

Microsoft IAS: O Internet Authentication Service é a implementação da Microsoft de um servidor RADIUS. Tal como é especificado pelo RADIUS o IAS oferece serviços de autenticação, autorização e contabilização centralizados para diferentes tipos de acessos de rede, incluindo wireless e Virtual Private Network VPN's. Funcionando como proxy, o IAS reencaminha as mensagens de autenticação e autorização para outros servidores RADIUS IAS (2010).

2.2 IEEE 802.1X

IEEE 802.1X é um padrão do Institute of Electrical and Electronic Engineers para controle de acesso à rede com base em portas. Faz parte do grupo IEEE 802.1, grupo de protocolos de redes. Ele fornece um mecanismo de autenticação para dispositivos que desejam conectar a uma LAN ou WLAN IEEE (2001).

Usar o 802.1X para controlar quem acessa uma rede é uma solução cada vez mais aplicada, o 802.1X pode ser configurado para exigir autenticação mútua entre o cliente e a rede, se não houver autenticação, as comunicações não são permitidas. O 802.1X trabalha com o protocolo Extensible Authentication Protocol - EAP Blunk e Vollbrecht (1998) para autenticar o cliente para a rede e a rede para o cliente, garantindo que ambos os lados se comuniquem com entidades reconhecidas.

2.2.1 Como funciona o padrão IEEE 802.1X

O 802.1X é projetado para trabalhar em qualquer tipo de rede: com ou sem fio. O 802.1X requer uma infraestrutura de suporte, clientes nominais que suportem o 802.1X, switches, pontos de acesso sem fio, um servidor RADIUS e algum tipo de banco de dados de contas, como o LDAP ou Active Directory.

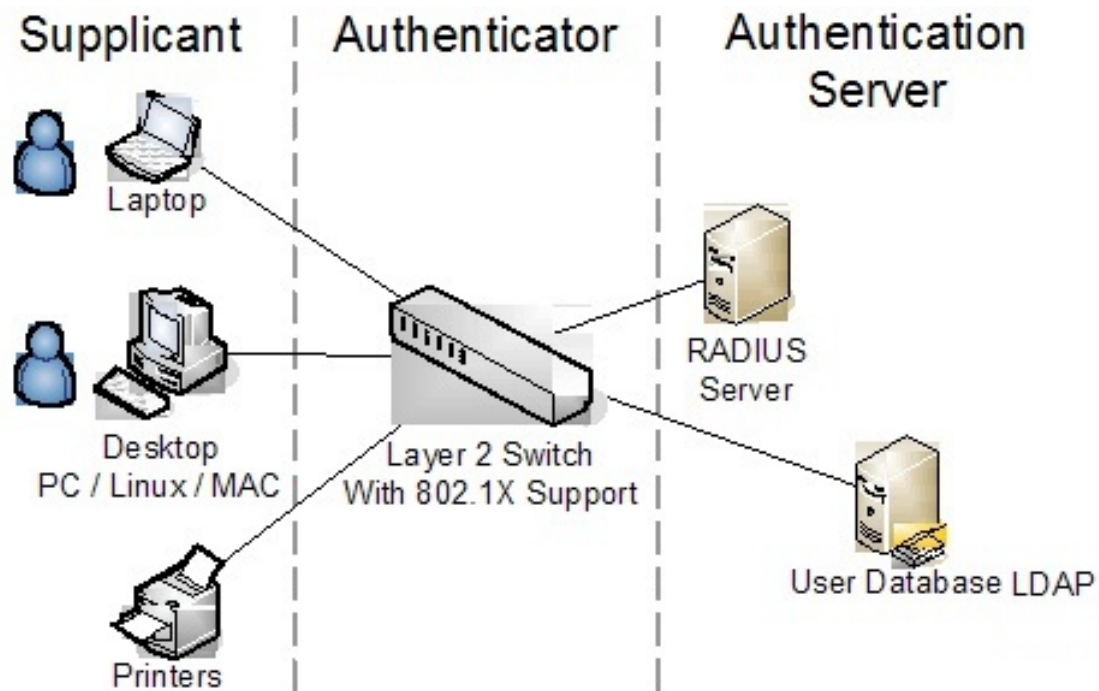


Figura 2.2: Infraestrutura para operar com 802.1X

Como pode ser visto na figura 2.2 um cliente, chamado *supplicant* suplicante faz uma conexão inicial para um *authenticator* autenticador, um switch de rede ou um ponto de acesso sem fio. O autenticador é configurado para exigir o 802.1X de todos os suplicantes e irá ignorar qualquer conexão de entrada que não se adequar. O autenticador solicita ao suplicante sua identidade, a qual ele passará adiante para o *authentication server* RADIUS.

Os elementos exigidos são:

- Software cliente, chamado de suplicante, em cada ponto de extremidade;
- Switches ou pontos de acesso habilitados para 802.1X, chamados de autenticadores, para "mediar" todas as comunicações ocorridas antes da atribuição do endereço IP;
- Servidor RADIUS, o servidor de autenticação, para gerenciar o processo AAA;

A inteligência dessa solução reside no suplicante e no servidor RADIUS, com o switch ou ponto de acesso simplesmente re-empacotando e distribuindo as informações.

2.2.2 Extensible Authentication Protocol - EAP

O RADIUS segue qualquer mecanismo necessário para autenticar o cliente que está entrando. Em geral, isto envolve a instalação de uma conversa EAP entre o suplicante e o servidor de autenticação - o autenticador é apenas um dispositivo de passagem aqui - e o estabelecimento de um método de autenticação dentro da conversa EAP. Note que o EAP em si não define qualquer tipo de segurança sozinho, os protocolos de identificação usados devem incorporar sua própria segurança.

EAP é o protocolo usado para autenticação, proposto para ampliar a funcionalidade de autenticação do protocolo ponto-a-ponto - PPP Simpson (1994). Antes limitado aos mecanismos providos pelo protocolo para controle de link, que eram o protocolo de autenticação - PAP e o protocolo de autenticação por negociação de desafio - CHAP Simpson (1996).

O PAP é um protocolo utilizado principalmente para autenticação em redes discadas, no qual o login e a senha trafegam em texto claro. O CHAP provê criptografia somente do usuário e senha, porém os dados também trafegam em texto claro.

Utilizando o EAP é possível ter independência de mecanismos de autenticação PPP, sendo assim uma alternativa interessante para interligação de redes visto a sua capacidade de adaptação a novos mecanismos.

Uma vantagem do uso do protocolo EAP é o aumento de vida útil dos equipamentos que possuem suporte ao padrão IEEE 802.1X, pois os mesmos passam a funcionar como intermediários entre o *host* cliente e o servidor de autenticação, não sendo necessário implementar mecanismos adicionais de segurança no próprio equipamento

Com a utilização de EAP para autenticação dos usuários a identidade real não é enviada antes do túnel de TLS codificado ser ativado. Depois que a identidade for enviada, o processo de autenticação começa. O protocolo usado entre o suplicante e o autenticador é o EAP, ou, mais corretamente, encapsulamento de EAP sobre a LAN EAPOL. O Autenticador re-encapsula as mensagens EAP para o formato de RADIUS e passa para o servidor de autenticação.

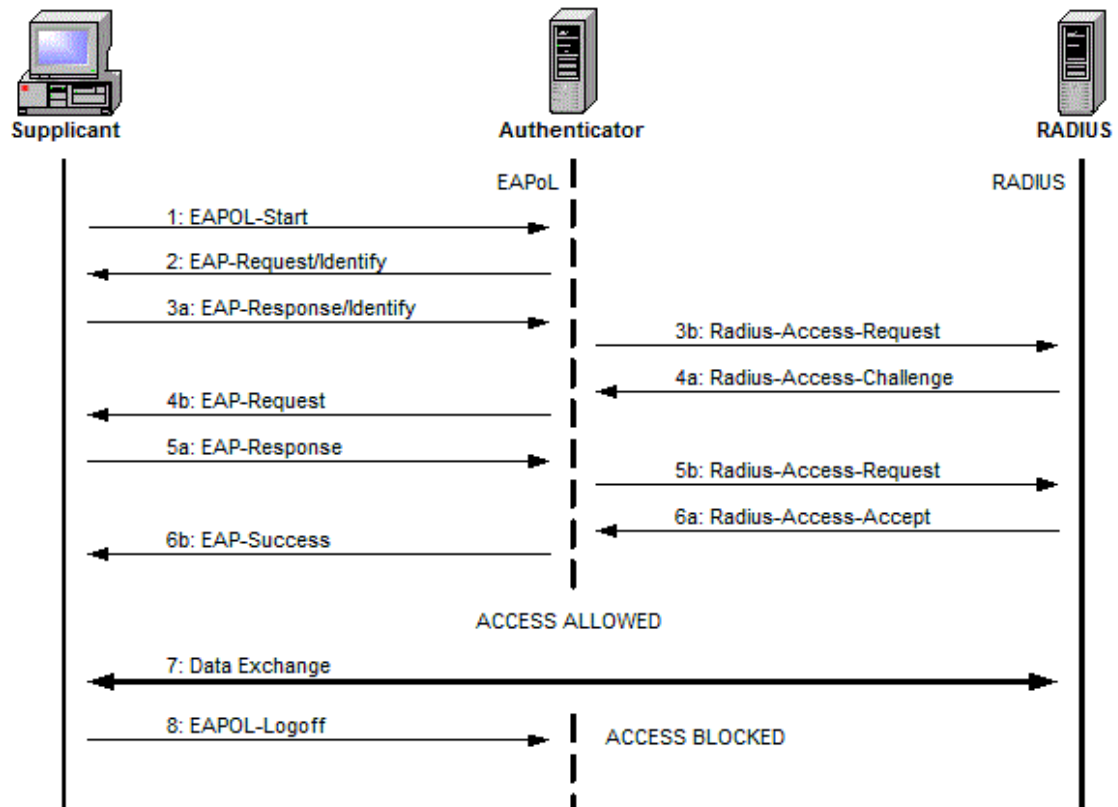


Figura 2.3: Diálogo entre os componentes

Durante a autenticação, o autenticador retransmite os pacotes entre o suplicante e o servidor de autenticação. Quando o processo de autenticação termina, o Servidor de autenticação envia uma mensagem de sucesso (ou fracasso, se a autenticação falhar). O autenticador abre então a "porta" para o suplicante, depois de uma autenticação próspera, é concedido acesso a outros recursos da rede ao suplicante. Figura 2.3

2.3 LDAP

2.3.1 Histórico do LDAP

Quando a *International Organization Standardization* (ISO) e o *Consultative Committee for International Telegraphy and Telephony* (CCITT) se juntaram no início da década de 80 para criar um serviço de mensagens (a série X.400), houve a necessidade de desenvolver um protocolo que organizasse entradas num serviço de nomes de forma hierárquica, capaz de suportar grandes quantidades de informação e com uma enorme capacidade de procura de informação. Esse serviço criado pelas duas instituições foi apresentado em 1988, denominando-se X.500, juntamente com um conjunto de recomendações e das normas ISO 9594. O X.500 especificava que a comunicação entre o cliente e o servidor de diretório usava o *Directory Access Protocol* (DAP) que era executado sobre a pilha de protocolos do modelo *Open Source Initiative* (OSI). O fato do X.500 ser muito complexo e de custo incompatível, levou os pesquisadores da Universidade de Michigan a criar um servidor *Lightweight Directory Access Protocol* (LDAP), que executava sobre o protocolo TCP/IP. A evolução do protocolo DAP para o LDAP pode ser observada na figura 2.4.

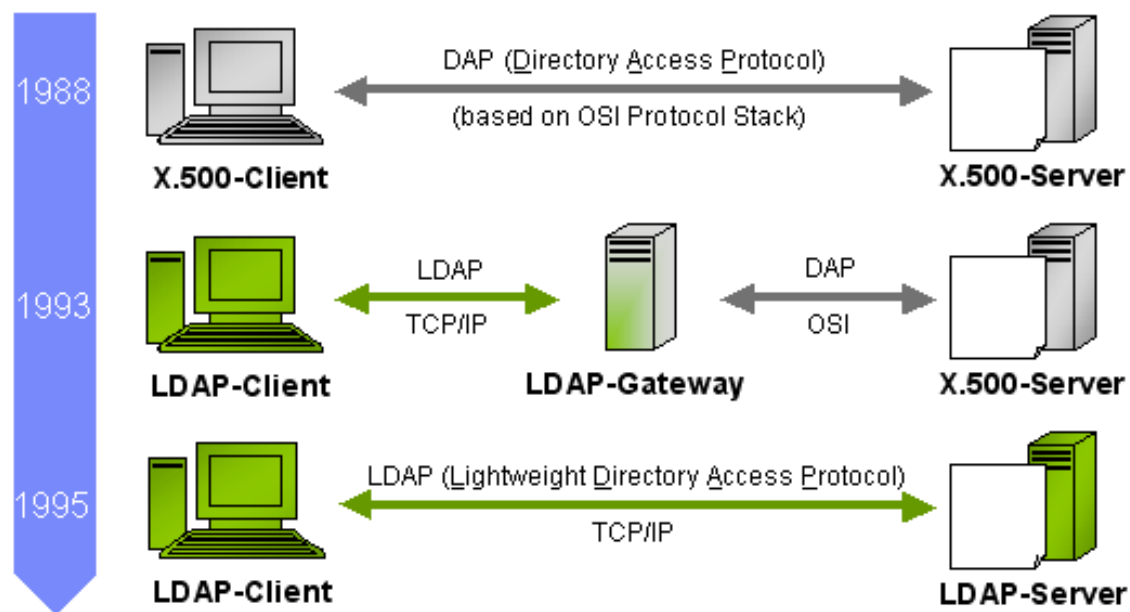


Figura 2.4: Evolução do DAP para o LDAP

Em 1993 o LDAP foi então apresentado como alternativa ao protocolo DAP para acesso a diretórios baseados no modelo X.500. Esse grupo de pesquisadores disponibilizou as fontes do *slapd* (*daemon* do OpenLDAP) na Internet e criou listas de discussão para divulgar e aperfeiçoar o novo serviço, sendo a sua evolução acompanhada por pessoas do mundo inteiro. Com a divulgação do *slapd* o LDAP deixou de ser uma mera alternativa ao DAP do X.500

e ganhou *status* de Serviço de Diretório completo, passando a competir diretamente com o X.500. Em Dezembro de 1997 o *Internet Engineering Task Force* (IETF) lançou a versão 3 do LDAP como proposta padrão para Serviços de Diretório Isquierdo (2001).

Atualmente várias empresas oferecem produtos com suporte ao LDAP, incluindo a Microsoft, Netscape e Novell. A *OpenLDAP Foundation* mantém e disponibiliza uma implementação *Open Source* do Serviço de Diretório LDAP, baseada na Universidade de Michigan, que inclui os seguintes módulos: *slapd*, *slurpd*, bibliotecas que implementam o protocolo LDAP, utilitários, ferramentas e exemplos de clientes LDAP. A evolução do OpenLDAP prossegue acompanhando a evolução dos padrões IETF.

2.3.2 Protocolo LDAP

Lightweight Directory Access Protocol, ou Protocolo Leve de Acesso a Diretórios, ou apenas LDAP, é um protocolo utilizado para atualizar e pesquisar diretórios, rodando sobre o protocolo de rede TCP/IP. Ele define não apenas a linguagem de consulta às informações, mas também o protocolo de rede que transporta as requisições e respostas do cliente para o servidor Campos (2010). Ao contrário de um banco de dados relacional, que necessita de uma biblioteca específica para implementar o seu protocolo de rede, qualquer cliente LDAP pode obter informações em qualquer servidor de diretório LDAP, já que o protocolo foi padronizado pelo IETF (*Internet Engineering Task Force*).

O LDAP é baseado no modelo cliente-servidor. Um ou mais servidores LDAP contêm os dados criando a árvore de Diretório LDAP. Um cliente LDAP conecta-se a um servidor, geralmente pela porta padrão 389, e faz uma requisição. O servidor responde ou exibe um ponteiro para onde o cliente pode conseguir a informação (tipicamente, outro servidor LDAP).

As operações básicas entre o cliente e o servidor são:

- Bind - autentica o cliente e especifica a versão do protocolo LDAP;
- Search - procura por entradas nos diretórios, recuperando-as ou não;
- Compare - compara o valor de uma entrada com o valor de um determinado atributo;
- ADD - adiciona uma nova entrada ao diretório;
- Delete - apaga uma entrada existente no diretório;
- Modify - modifica uma entrada existente no diretório;
- Modify DN - move ou renomeia uma entrada existente no diretório;
- Start TLS - protege a conexão através da TLS (*Transport Layer Security*);
- Abandon - aborta uma requisição prévia;

- Extend Operation - operação genérica utilizada para definir outras operações;
- Unbind - termina a conexão (importante salientar que esta operação não é a inversa da operação Bind).

O protocolo LDAP está na versão 3, e suporta uma variedade de métodos de autenticação através da operação “Bind”; onde as duas formas básicas são:

- Autenticação Simples: Nome distinto da entrada que está sendo conectada ao diretório e a senha dessa entrada;
- Autenticação Geral: Permite indicar as credenciais e o método de autenticação.

O cliente também pode solicitar um serviço LDAP com a identidade anônima, sem autenticar-se. Nesse caso o diretório também define quais informações podem ser acessadas por esse cliente.

2.3.3 Serviço de Diretório *versus* Banco de Dados Relacional

Uma das maiores diferenças entre o serviço de diretório e as bases de dados relacionais é a estrutura onde as informações são armazenadas. No LDAP elas são armazenadas em um diretório, ou seja, em uma estrutura em árvore, otimizada para operações de pesquisa e leitura e com baixo desempenho para operações de escrita e atualização. Já nos bancos de dados relacionais as informações são armazenadas em tabelas, ideais para grandes volumes de dados e execução de transações complexas.

Um banco de dados relacional pode exercer a mesma função de um diretório, mas é muito mais complexo e caro. Isto se deve ao fato de que no diretório as informações são muito mais lidas e procuradas do que escritas e atualizadas, e além disso, não existe um mecanismo padrão para a troca de informações entre diferentes Bancos de Dados.

Outra diferença é que a medida de performance dos bancos de dados relacionais é realizada em transações por segundo. Já no serviço de diretório, a medida de performance é realizada em consultas por segundo.

No LDAP não é possível relacionar diretamente dois ou mais atributos, já que se trata de uma base de dados estruturada hierarquicamente; ao contrário de uma base de dados relacional. A relação de atributos no LDAP ocorre explicitamente, ou seja, cada entrada deve possuir os atributos que são relacionados.

Portanto, quando usar um diretório?

- Quando as informações forem relativamente pequenas;
- Quando o modelo de informação armazenada for baseado em atributos. Caso a informação não possa ser expressa dessa forma, o uso de um diretório não é recomendável;

- Quando as informações forem mais lidas do que escritas; caso contrário, talvez seja melhor utilizar um banco de dados relacional.

2.3.4 Vantagens e Desvantagens do LDAP

As principais vantagens do LDAP são:

- É um padrão aberto;
- É otimizado para realizar pesquisas e leitura;
- Centraliza toda a informação proporcionando enormes benefícios tais como: um único ponto de administração, menos informação duplicada, maior transparência das informações;
- Possui um mecanismo de replicação da base incluído;
- Suporta mecanismos de segurança para autenticação (SASL) e para a troca de dados (TLS);
- Muitas aplicações e serviços possuem suporte ao LDAP.

As principais desvantagens do LDAP são:

- Em alguns casos não substitui as bases de dados relacionais;
- Pouco eficiente para operações de escrita e atualização;
- Integração com outros serviços e aplicações torna a implantação complexa.

2.4 SAMBA

O SAMBA não é apenas um aplicativo, mas sim um conjunto de aplicativos que se comunicam através do protocolo SMB, e é dividido em dois módulos principais: o servidor SAMBA, propriamente dito, e o *smbclient*, o cliente que permite acessar compartilhamento em outras máquinas. Usando o protocolo SMB, o SAMBA permite integrar servidores rodando Linux com uma rede Windows (ou até mesmo uma rede mista Windows/Linux, se utilizarmos as máquinas Linux apenas como clientes) da Costa (2010).

2.4.1 Histórico do SAMBA

O Samba é uma criação de Andrew Tridgell. De acordo com informações dadas no site oficial do software, Tridgell precisava montar um espaço em disco em seu PC para um servidor Unix. Esse PC rodava o sistema operacional DOS e, inicialmente, foi utilizado o sistema de arquivos NFS (Network File System) para o acesso. Porém, um aplicativo precisava de suporte ao protocolo NetBIOS (não suportado pelo NFS). A solução encontrada por Tridgell não foi tão simples: ele escreveu um sniffer (pequeno programa para captura de tráfego de dados em rede) que permitisse analisar o tráfego de dados gerado pelo protocolo NetBIOS, fez engenharia reversa no protocolo SMB (Server Message Block) e o implementou no Unix. Isso fez com que o servidor Unix aparecesse como um servidor de arquivos Windows em seu PC com DOS.

Esse código foi disponibilizado publicamente por Tridgell em 1992. Porém, tempos depois, o projeto foi posto de lado até que um determinado dia Tridgell decidiu conectar o PC de sua esposa ao seu computador com Linux. Porém, não encontrou nenhum meio melhor que seu código para fazer isso e assim o utilizou.

Tridgell descobriu que as documentações dos protocolos SMB e NetBIOS estavam atualizadas e assim voltou a dedicar-se ao projeto. Porém, uma empresa entrou em contato com ele reivindicando os direitos sobre o nome usado no software até então. Diante disso, Andrew Tridgell teve a idéia de procurar em um dicionário uma palavra que tivesse as letras s, m e b (de SMB) e acabou encontrando o termo "samba". A partir daí o projeto Samba cresceu e hoje Andrew Tridgell conta com uma excelente equipe de programadores e com milhares de usuários de sua solução espalhados pelo mundo.

2.4.2 Controlador de Domínio (PDC)

Em uma pequena rede, manter as senhas dos usuários sincronizadas entre as estações Windows e o servidor SAMBA não chega a ser um grande problema. No entanto, em redes de grande porte, pode se tornar um procedimento trabalhoso, consumindo tempo considerável em ajustes nas configurações.

Para solucionar o problema, existe a opção de usar o servidor SAMBA como um controlador primário de domínio (PDC), onde ele passa a funcionar como um servidor de autenticação para os clientes Windows e (opcionalmente) armazena os perfis de cada usuário, permitindo que eles tenham acesso a seus arquivos e configurações a partir de qualquer máquina onde façam logon.

Ao cadastrar um novo usuário no servidor SAMBA, ele automaticamente pode fazer logon em qualquer uma das estações configuradas. Ao remover ou bloquear uma conta de acesso, o usuário é automaticamente bloqueado em todas as estações. Isso elimina o problema de sincronismo entre as senhas no servidor e nas estações e centraliza a administração de usuários e permissões de acesso no servidor, simplificando bastante o trabalho de administração.

Capítulo 3

METODOLOGIA

Neste capítulo é apresentado a metodologia proposta para resolver o problema de autenticação para a GigaUFOPnet. O desenvolvimento do trabalho foi dividido em 4 etapas. Na primeira etapa foram realizados treinamentos e pesquisas para aprendizado das tecnologias que foram utilizadas para implementar o projeto. Respectivamente foram realizados os cursos: "Treinamento Linux - LDAP" e o curso "Linux - FreeRadius". Na segunda etapa foram realizados experimentos em laboratório para definir a melhor arquitetura a ser utilizada pra implantação da autenticação na rede GigaUFOPnet. Já na terceira etapa, etapa atual, toda a infraestrutura testada em laboratório será aplicada na rede de um prédio da UFOP. Na etapa final, a implantação em todos os campi da UFOP será realizada por partes para garantir o pleno funcionamento da rede GigaUFOPnet e nenhum prejuízo para os seus usuários. De maneira sistemática e coordenada todas as modificações necessárias serão aplicadas em todos os prédios da UFOP.

3.1 Desenvolvimento

O padrão IEEE 802.1X requer:

- uma infra-estrutura de suporte;
- clientes nominais que suportem o 802.1X;
- switches que podem participar no 802.1X;
- um servidor RADIUS;
- algum tipo de banco de dados de (como o Active Directory - LDAP).

Um computador com a seguinte configuração foi utilizado como servidor:

- CPU Intel(R) Xeon(TM)CPU 3.00 GHz
- 3 GB Memória RAM
- Disco Rígido de 320 GB;
- LAN 10/100;
- DVD.

Por não se tratar do foco deste trabalho, não será explicado detalhadamente como instalar e configurar todos os arquivos de configuração, no entanto, os arquivos mais importantes para processo de instalação serão apresentados.

3.1.1 Sistema Operacional



Figura 3.1: Instalação do CentOS 5.5

Desenvolvido a partir dos códigos-fonte do sistema operacional Red Hat Enterprise Linux (RHEL), o CentOS - acrônimo de Community ENTERprise Operating System - é uma distribuição Linux gratuita, voltada para o ambiente corporativo, e que agrega as vantagens técnicas da distribuição na qual se baseia: segurança, estabilidade e compatibilidade com diversos hardwares e pacotes criados especificamente para o RHEL. Cardozo (2007)

O sistema operacional deve ser instalado com as suas configurações padrões, especificando apenas os pacotes que estejam desabilitados que deverão ser utilizados posteriormente, por padrão o CentOS vem marcado com a opção "Desktop - Gnome", a qual desmarcaremos e utilizaremos a opção "Desktop - KDE".

Após a instalação do CentOS utilize o comando abaixo para atualizar todos os pacotes do sistema.

```
#yum update -y
```

3.1.2 Integrando SAMBA ao OpenLdap

Os pacotes SAMBA e OpenLdap devem ser instalados seguindo os comandos das Tabelas 4.1 e 4.2

```
#yum install -y samba-common-3.0.33-3.7.el5_3.1
#yum install -y samba-3.0.33-3.7.el5_3.1
#yum install -y system-config-samba-1.2.41-3.el5
#yum install -y samba-client-3.0.33-3.7.el5_3.1
#yum install -y smbldap-tools-0.9.5-1.el5.rf
#yum install -y pam_smb-1.1.7-7.2.1
```

Tabela 3.1: Pacotes necessários para instalação do SAMBA

```
#yum install -y openldap-servers
#yum install -y openldap-devel
#yum install -y smbldap
#yum install -y openldap-client
#yum install -y php-ldap
#wget http://dag.wieers.com/rpm/packages/
rpmforge-release/rpmforge-release-0.3.6-1.el5.rf.i386.rpm
#rpm -Uvh rpmforge-release-0.3.6-1.el5.rf.i386.rpm

#yum install -y openldap-clients-2.3.43-3.el5
#yum install -y php-ldap-5.1.6-23.2.el5_3
#yum install -y openldap-2.3.43-3.el5
#yum install -y smbldap-tools-0.9.5-1.el5.rf
#yum install -y nss_ldap-253-17.el5
#yum install -y openldap-devel-2.3.43-3.el5
#yum install -y openldap-servers-2.3.43-3.el5
```

Tabela 3.2: Pacotes necessários para instalação do OpenLdap

Toda a configuração relacionada com nomes, grupo de trabalho, tipo de servidor, log, compartilhamento de arquivos e impressão do SAMBA está localizada no arquivo **/etc/samba/smb.conf**.

Este arquivo é dividido em seções e parâmetros. As sessões utilizam nomes reservados para configurações específicas. São elas:

[global] Define configurações que afetam o comportamento de todo o servidor SAMBA, com efeitos em todas as compartilhamentos.

[homes] Especifica opções de acesso aos diretórios HOME dos usuários.

[printers] Define opções gerais para controle das impressoras do sistema. Este compartilhamento mapeia os nomes de todas as impressoras encontradas no **/etc/printcap**.

[profile] Define um perfil quando o servidor SAMBA é usado como controlador de domínio.

Outros nomes de sessões podem ser utilizados para definir compartilhamentos de impressoras ou arquivos.

O OpenLDAP possui dois arquivos de configuração: `slapd.conf` e `ldap.conf`. O `slapd.conf` é o mais importante, onde são realizadas todas as configurações do servidor LDAP. O arquivo `ldap.conf` indica apenas informações de identificação do servidor.

No arquivo `slapd.conf` estão definidos os esquemas LDAP que serão utilizados, estes devem ser descompactados no diretório de schemas do OpenLDAP.

3.1.3 FreeRadius

Os pacotes necessários para instalação do FreeRadius são apresentados na Tabela 4.3.

```
yum install -y freeradius
yum install -y freeradius-mysql
wget ftp://fr2.rpmfind.net/linux/dag/redhat/el5/
en/i386/dag/RPMS/radiusclient-0.3.2-0.2.el5.rf.i386.rpm
rpm -i radiusclient-0.3.2-0.2.el5.rf.i386.rpm
```

Tabela 3.3: Pacotes necessários para instalação do FreeRadius

Todos os arquivos de configuração do FreeRadius estão localizados no diretório `/etc/raddb/`, os principais são:

- **radiusd.conf:** Principal arquivo de configuração, nele configuram-se todos os parâmetros do servidor e também se habilita os módulos de AAA desejados. Responsável pelo daemon do radius e inclusões dos demais arquivos de configuração. O arquivo consiste do par valor-atributo, seções e comentários. O par atributo-valor estão no formato `nome=valor`. Uma seção começa com nome da seção, seguido na mesma linha pelo caracter `"{"`. A seção principal contém outras seções, ou outros pares valor-atributo. Uma seção termina com caractere `"}"`. Linhas começadas com caractere `#` são comentários, e ignoradas. Linhas em branco também são ignoradas. O arquivo é baseado em linha. Isto é, cada nova linha representa um comentário, uma seção ou um par atributo-valor. Não é possível especificar mais de um item na mesma linha.
- **dictionary:** Define todos os atributos RADIUS possíveis, usados em outros arquivos de configuração. O dicionário do servidor radius está localizado no diretório de configuração do serviço `/etc/raddb/dictionary`. Ele faz referências a outros arquivos de dicionários localizados em `/usr/local/share/freeradius`. Cada arquivo de dicionário contém uma lista de valores e atributos do Radius, que o servidor utiliza para traçar mapas entre descrições e outros dados.

`/etc/raddb/dictionary` - Este é o arquivo mestre do dicionário, que faz referência a dicionários pré-definidos e arquivos incluídos no servidor. Os novos atributos como as VLAN'S devem ser colocados neste arquivo.

`/etc/raddb/dictionary.3com` - possibilita comunicação entre o switch 3com e o Freeradius.

`/etc/raddb/dictionary.tunnel` - responsável pelo encapsulamento dos atributos de comunicação entre o switch e o FreeRadius.

- **ldap.attrmap:** Mapa de equivalência do dicionário de atributos Radius para o dicionário de atributos LDAP para ser usado pelos módulos de autenticação e autorização LDAP (rlm_ldap). A relação de equivalência entre os atributos radius e os atributos ldap estão neste arquivo. Se houver necessidade de adicionar ou alterar algum mapeamento de equivalência, este arquivo deve ser utilizado.

A relação de equivalência entre os atributos radius e os atributos ldap estão neste arquivo. Se houver necessidade de adicionar ou alterar algum mapeamento de equivalência, este arquivo deve ser utilizado.

- **users:** Base de usuários, neste arquivo pode-se cadastrar as credenciais dos usuários. Este arquivo contém informações de segurança e de autenticação para cada usuário. Requisições de contabilização não são processadas neste arquivo. Para isso é utilizado o arquivo `acct_users`. É necessário acrescentar as configurações sobre o switch para realizar a contabilização. O primeiro campo deste arquivo é o nome do usuário que pode ter até 253 caracteres. É mostrado na mesma linha com os requisitos de autenticação para este usuário. Pode ser incluído senha, nome servidor, porta do servidor, tipo de protocolo.
- **clients.conf:** Configuração dos dispositivos que farão as consultas ao Radius (NAS), tipo Access Point, switches etc. As configurações de clientes são definidas no arquivo `clients.conf`. Os clientes aqui não estão se referindo a usuários e sim a dispositivos que se encarregam de procurar o radius para validar o usuário. São equipamentos como cisco e 3Com ou serviços em qualquer servidor que se baseie numa autenticação radius. A principal alteração neste arquivo consiste em definir o `secret = Chave secreta compartilhada para comunicação entre o servidor radius e o cliente/NAS`.
- **proxy.conf:** É responsável por deixar as credencias recebidas no formato em que o FreeRadius precisa para realizar as suas operações, por exemplo, no sistema operacional windows com domínio, as credencias de nome de usuário são apresentadas no formato domínio/usuário, sendo necessário retirar o domínio da credencial do usuário para que este seja encontrado na base de dados.
- **sql.conf:** Arquivo responsável pelo gerenciamento dos logs do FreeRadius e direcioná-lo para o banco de dados apropriado.

- **eap.conf:** O arquivo eap.conf define os mecanismos de EAP suportados e como o servidor deve se comportar para cada requisição. Trata tanto da autenticação entre o suplicante e o servidor como da autenticação entre o NAS e servidores. Obs.: Os switches 3Com não suportam PEAP, apenas EAP.

3.1.4 Testando a configuração do Radius

Após a instalação e configuração do servidor, são apresentados três mecanismos para testar se as configurações estão funcionando corretamente no servidor FreeRadius.

1. Comando radiusd

No terminal do Linux, execute o comando:

```
#radiusd -X
```

Se estiver funcionando retornará uma mensagem como na Tabela 4.4:

```
...
read_config_files: reading dictionary
read_config_files: reading naslist
Using deprecated naslist file. Support for this will go away soon.
read_config_files: reading clients
read_config_files: reading realms
There appears to be another RADIUS server running on the authentication port 1812
```

Tabela 3.4: Comando radiusd -X

2. Comando radtest

O comando radtest acompanha o pacote do freeradius e é instalado com os demais aplicativos. Serve para testar autenticação e retorno de parâmetros.

Sintaxe:

```
radtest usuário senha servidor:porta_autenticacao portaNAS segredoradius
```

```
radtest user password radius-server radius-port nas-port-number secret
```

- **Usuário, senha** - login do usuário e senha;
- **servidor** - endereço ip do servidor radius;
- **porta_autenticacao** - porta no servidor onde o serviço radius atende solicitações de autenticação;
- **portaNAS** - porta do NAS, pode ser uma porta eletrônica (número do modem) ou virtual, apenas para controle. Para testes, coloque qualquer valor numérico.

- **segredoradius** - quando você faz uma solicitação de autenticação, seu endereço ip tem que estar cadastrado no arquivo clients.conf do servidor num par de ip/segredo. O ip será descoberto pelo servidor através do parâmetro NAS-IP-Address, o segredo você tem que informar aqui.

Exemplo, retorno comando radtest:

```
# radtest teste teste 127.0.0.1 0 testing123
```

```

Sending Access-Request of id 206 to 127.0.0.1 port 1812
User-Name = "teste"
User-Password = "teste"
NAS-IP-Address = 255.255.255.255
NAS-Port = 0
rad_recv: Access-Accept packet from host 127.0.0.1:1812, id=206, length=20

```

Tabela 3.5: Comando radtest

3. NTRadPing RADIUS Test Utility

Ferramenta Windows, para testar todos os tipos de pacotes radius: Autenticação, Autorização, Contabilidade e Status. Pode ser obtido em: <http://www.mastersoft-group.com/download/>

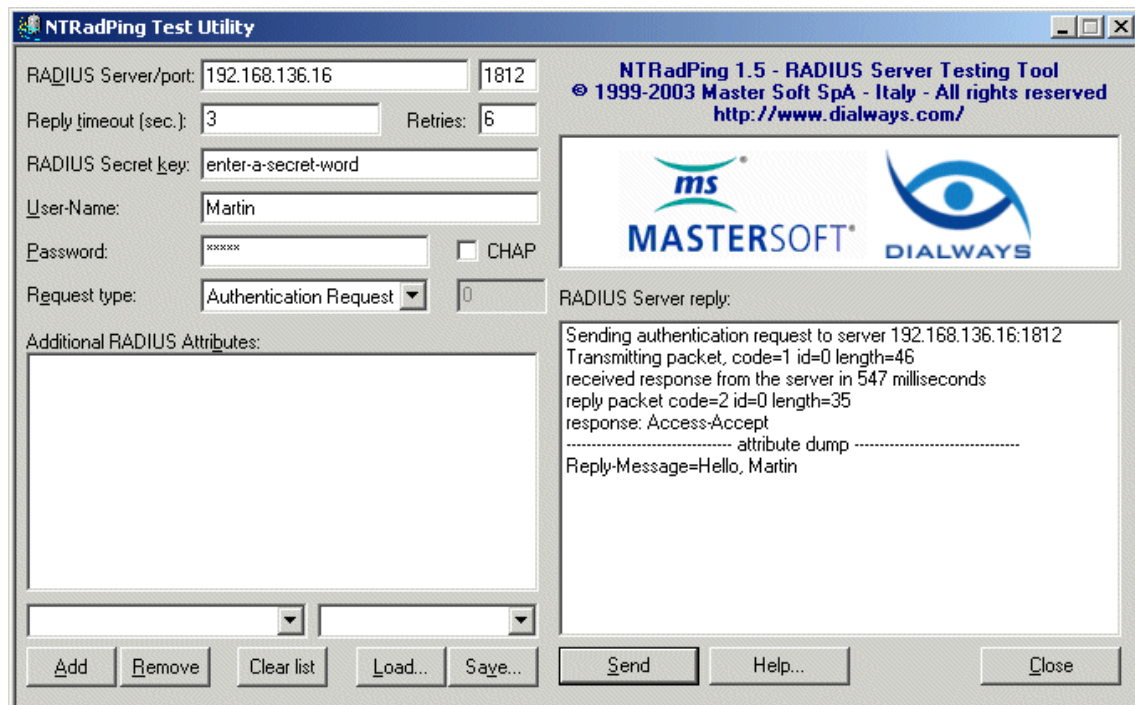


Figura 3.2: NTRadPing RADIUS Test Utility

Log do teste registrado em `/var/log/radius/radius.log`

```
Mon Oct 31 14:30:19 2010 : Debug: rad_check_password: Found Auth-Type Local
Mon Oct 31 14:30:19 2010 : Debug: auth: type Local
Mon Oct 31 14:30:19 2010 : Debug: auth: user supplied User-Password matches local
User-Password
Sending Access-Accept of id 1 to 192.168.3.83 port 4784
Mon Oct 31 14:30:19 2010 : Debug: Finished request 1
```

Tabela 3.6: Log registrado pelo NTRadPing RADIUS Test Utility

3.1.5 Configuração dos Switchs

A GigaUFOPnet é composta por switchs 3COM, modelos 5500 e 4500 para a utilização do padrão 802.1X é necessário:

1. Como um requerimento importante, o administrador sempre deverá utilizar a última versão do firmware disponível do 3ComOS;
2. Configurar o domínio;

```
#
domain default enable ufop.br
```

Tabela 3.7: Switch 3COM - Configuração de domínio

3. Definir o padrão 802.1X como configuração global;

```
#
dot1x
dot1x dhcp-launch
dot1x authentication-method eap
undo dot1x handshake enable
```

Tabela 3.8: Switch 3COM - 802.1X como configuração global

4. Configurar as portas do switch para trabalharem no padrão 802.1X.

3.1.6 DHCP

O DHCP, Dynamic Host Configuration Protocol, é um protocolo de serviço TCP/IP que oferece configuração dinâmica de terminais, com concessão de endereços IP de host e outros parâmetros de configuração para clientes de rede.

Neste projeto será necessário a instalação do DHCP com suporte ao padrão IEEE 802.1Q ou VLAN Tagging que permite compartilhar de uma ligação física de rede Ethernet por várias redes lógicas independentes.

A instalação do DHCP é bastante simples. Basta instalar o servidor DHCP através do yum.

```
#yum install dhcp
```

Depois que o script terminar de executar a instalação, o arquivo de configuração que precisa ser editado pode ser encontrado em **/etc/dhcpd.conf** . Tabela 4.9

Para cada interface Ethernet serão configuradas todas as subnets que serão utilizadas. No diretório **/etc/sysconfig/network-scripts/** deverão ser criadas as redes lógicas independentes de acordo com os IP's definidos pelo dhcp.conf associando a sua VLAN.


```
#
# DHCP Server Configuration file.
# see /usr/share/doc/dhcp*/dhcpd.conf.sample
#
option domain-name "UFOP";
option domain-name-servers 192.168.212.7, 192.168.0.81;
option netbios-name-servers 192.168.0.189;

default-lease-time 50;
max-lease-time 60;
ddns-update-style none;
authoritative;
log-facility daemon;
default-lease-time 99999999;
max-lease-time 99999999;

subnet 192.168.72.0 netmask 255.255.255.224 {
range 192.168.72.5 192.168.72.30;
option routers 192.168.72.1;
}

subnet 192.168.72.32 netmask 255.255.255.224 {
range 192.168.72.36 192.168.72.62;
option routers 192.168.72.34;
}

subnet 192.168.72.64 netmask 255.255.255.224 {
range 192.168.72.68 192.168.72.94;
option routers 192.168.72.65;
}

subnet 192.168.72.96 netmask 255.255.255.224 {
range 192.168.72.100 192.168.72.126;
option routers 192.168.72.97;
}
```

Tabela 3.9: dhcpd.conf

3.1.7 Configuração dos Clientes 802.1X

Configuração de Cliente - Rede com Fio (Windows XP)

Nesta etapa será configurado o padrão 802.1X na máquina CLIENTE-01, para concluir esse procedimento, você deve primeiro habilitar o serviço de Configuração Automática de Rede com Fio, que é desativado por padrão.

Clique no botão **Iniciar**. Na caixa de pesquisa, digite **services.msc** e pressione **Enter**. Se você for solicitado a informar uma senha de administrador ou sua confirmação, digite a senha ou forneça a confirmação.

Na caixa de diálogo Serviços, clique na guia **Padrão** na parte inferior do painel principal, clique com o botão direito do mouse em **Configuração Automática de Rede com Fio** e em **Iniciar**. Figura 4.3.

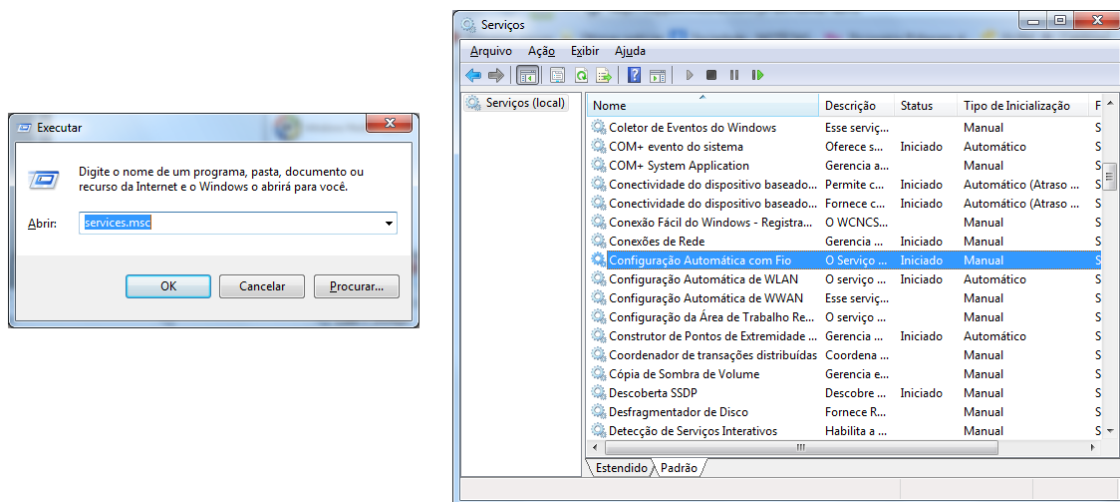


Figura 3.3: Iniciando o Serviço Configuração Automática de Rede com Fio

Após executar esta seleção, o administrador deverá acessar o menu principal de opções do Windows XP, bastando apenas clicar no botão **Iniciar** e selecionar o **Painel de controle**, o sistema apresentará as diversas aplicações associadas ao painel de controle, onde o administrador deverá executar, por meio de um duplo clique, a opção **Conexões de rede**, conforme apresentado na Figura 4.4.

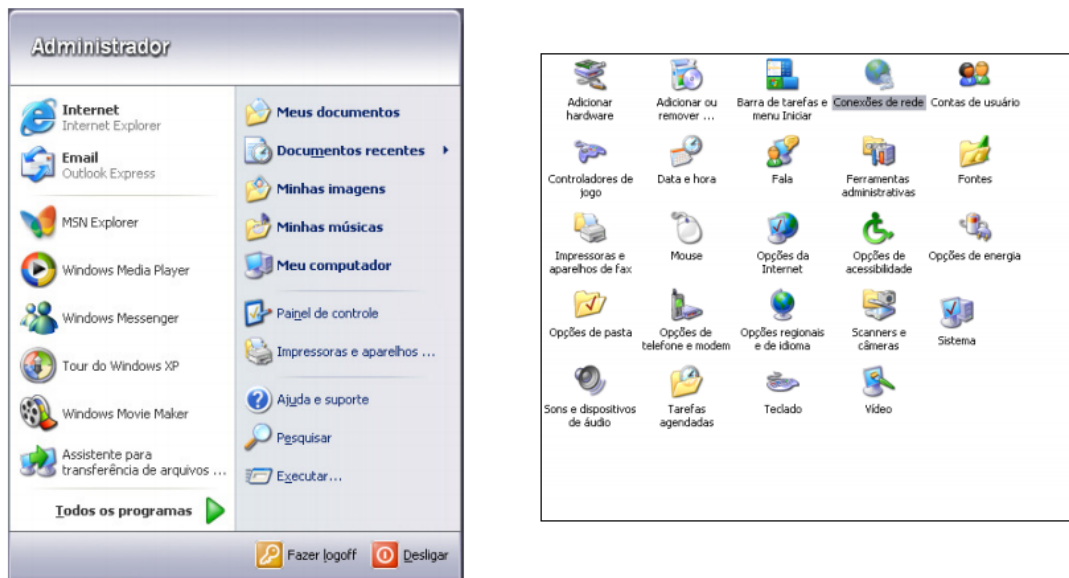


Figura 3.4: Acesso ao Painel de Controle

Seguidamente o Windows XP, apresentará a lista de conexões presentes na máquina CLIENTE-01, para acessar os recursos de rede, basta o administrador executar um duplo clique na conexão de rede. Como resposta, será apresentado uma janela onde o administrador deverá pressionar o botão **Propriedades**, conforme apresentado na Figura 4.5.

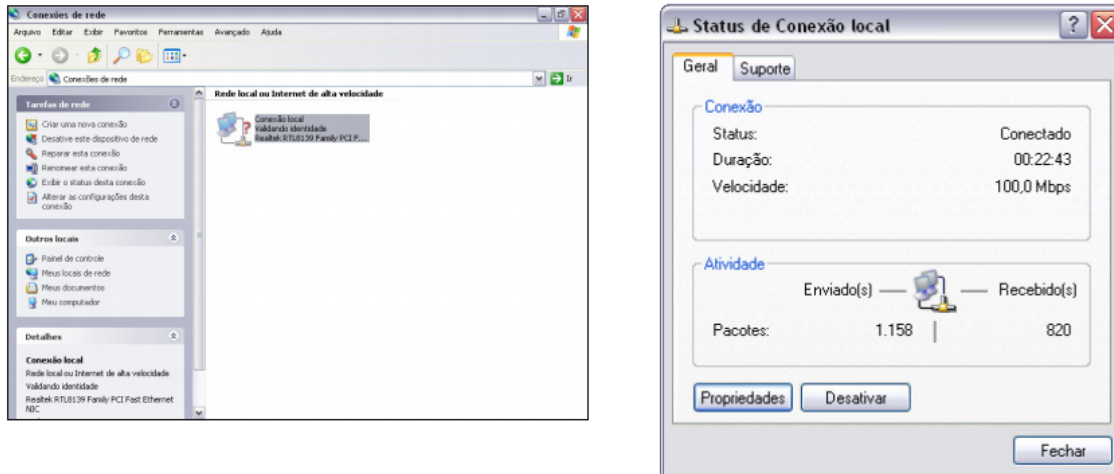


Figura 3.5: Acesso aos Recursos de Rede

Após solicitar o acesso as propriedades de rede, serão apresentados ao administrador os itens de configuração associados ao dispositivo de rede, onde o mesmo deverá selecionar a opção **Protocolo TCP/IP** e seguidamente pressionar o botão **Propriedades**, conforme apresentado na Figura 4.6.

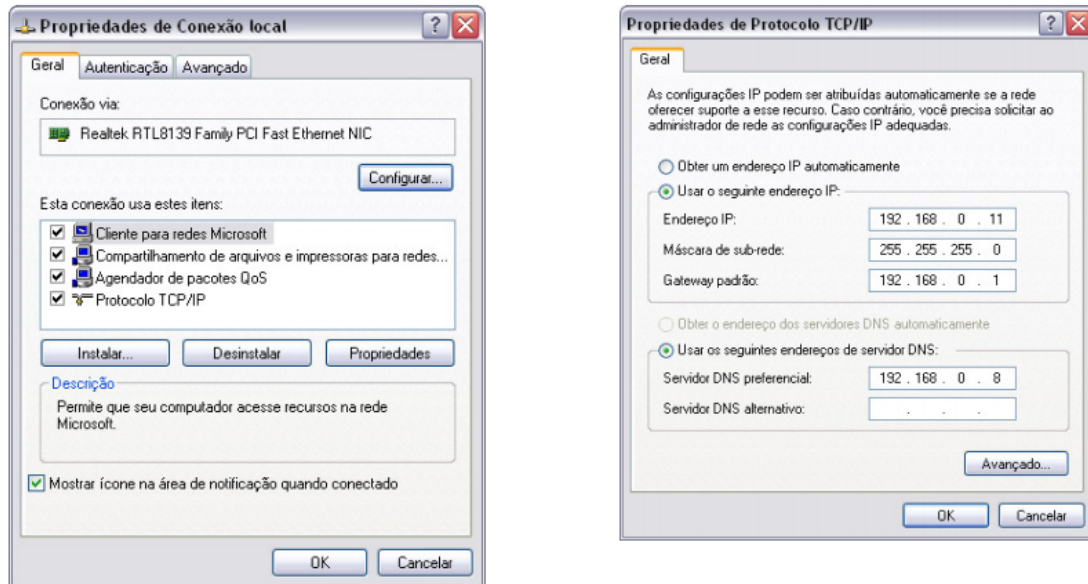


Figura 3.6: Acessando os Parâmetros TCP/IP

Para construção do ambiente de teste descrito neste material, é necessário que o administrador preencha ou habilite as opções de obtenção automática dos parâmetros TCP/IP com os valores associados a endereço IP, máscara de subrede, gateway padrão e servidor DNS. Após o preenchimento destes valores o administrador deverá pressionar o botão **Avançado**.

Como resultado da solicitação executada no passo anterior, o sistema apresentará uma janela conforme indicado na Figura 4.7, onde o administrador deverá selecionar a aba **Avançado** e garantir que a opção de Firewall não esteja selecionada.

Seguidamente o administrador deverá selecionar a aba Autenticação e selecionar a opção **Permitir autenticação IEEE 802.1X para rede** e na opção **Tipo de EAP:**, escolha um método de autenticação de rede que deseja usar. Figura 4.6.

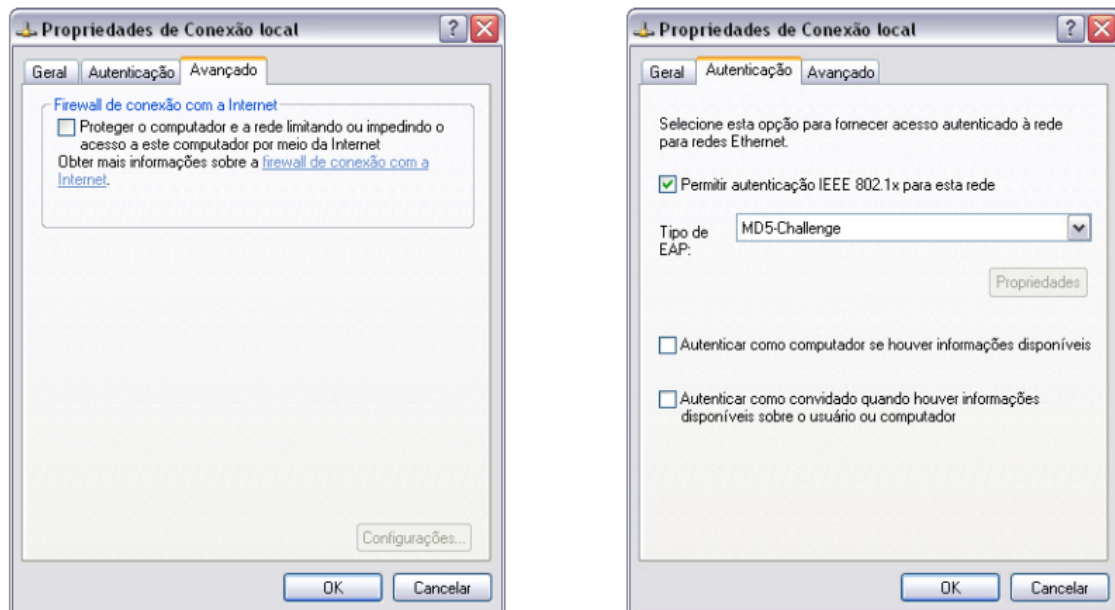


Figura 3.7: Configuração da Autenticação

OBS: Para o Windows 98 é necessário utilizar um cliente, pois o 98 não tem suporte nativo ao 802.1X.

Configuração de Cliente - Rede sem Fio (Windows XP)

Segue exemplo de configuração de cliente Windows XP para rede sem fio, também é válida para W2K, W2K3.

Identificando a rede. Figura 4.8.

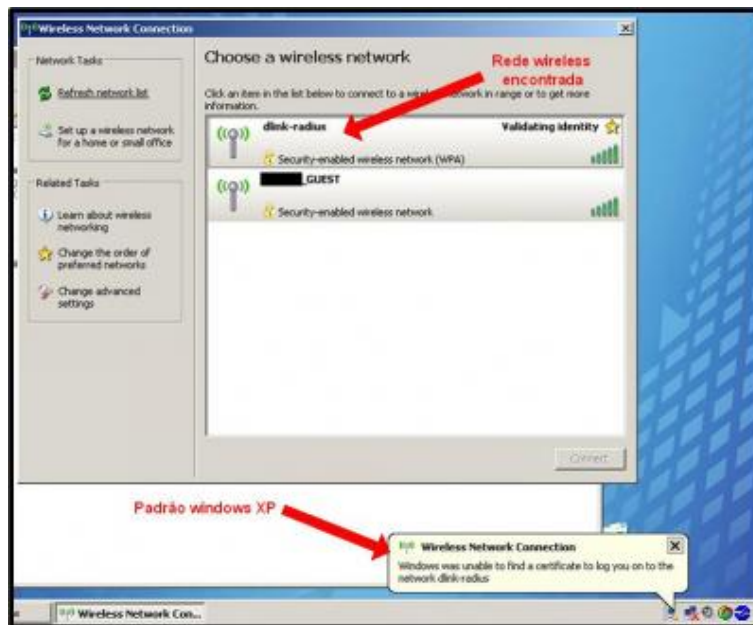


Figura 3.8: Identificando a rede sem fio

Dê dois cliques sobre esta rede, assim o perfil desta rede será adicionado, clique então em **change advanced Settings** e vá em **wireless network** conforme a figura 4.9, selecione a rede e clique em propriedades.

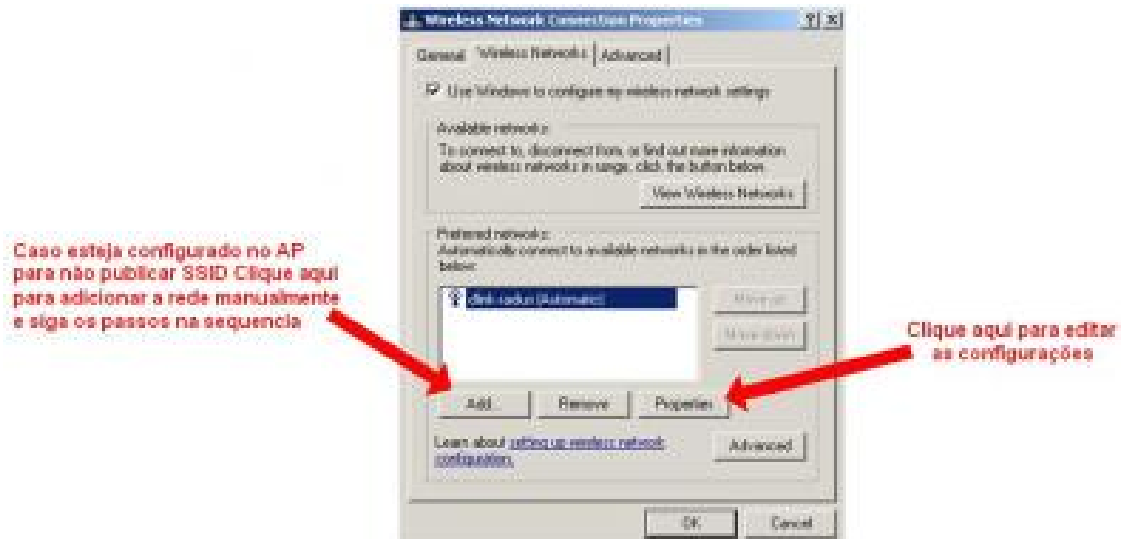


Figura 3.9: Configurando a rede sem fio

Por padrão o Windows XP ao identificar uma rede wireless com 802.1X habilitado, seleciona como credenciais de autenticação certificado digital (EAP-TLS), que é baseado em certificado digital (uma mensagem de que não pode encontrar o certificado para validar-se na rede deve ser indicado no balão), porém como queremos utilizar usuário e senha, temos que configurar para que seja solicitado este modelo de autenticação (EAP-PEAP-MSCHAPv2). Figura 4.10.

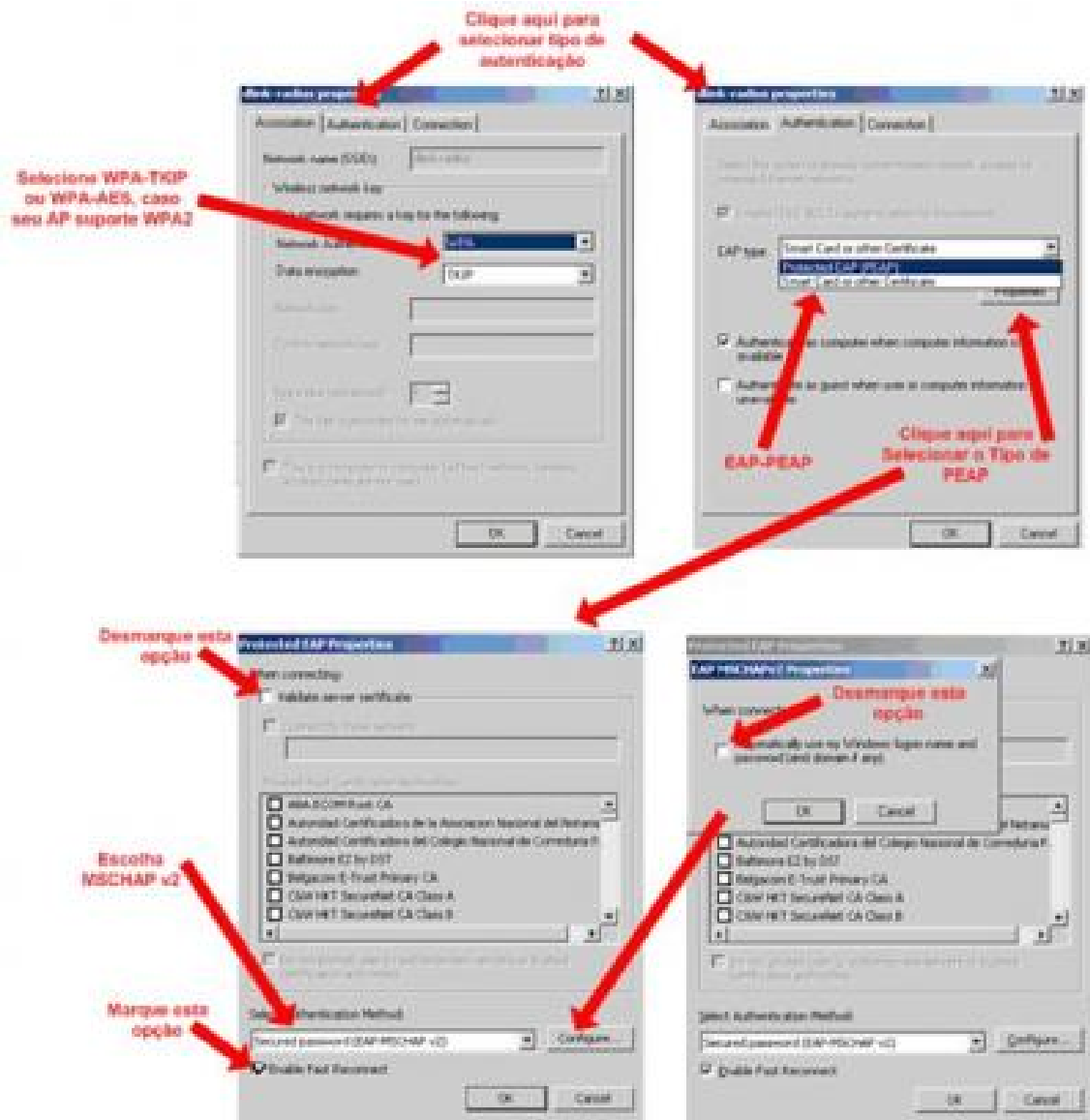


Figura 3.10: Habilitado rede wireless com 802.1X

Conforme indicado na Figura 4.10, você deve selecionar para a associação com o AP, WPA e TKIP como mecanismo de criptografia (AES caso use WPA2), depois na aba autenticação selecione **EAP-PEAP**, clique então em **propriedades** e desmarque a opção de **validar certificado** (como estamos usando TTLS temos a opção de autenticar apenas com usuário e senha), selecione **EAP-MSCHAPv2** e marque também a opção **Fast Reconnect** para evitar pedidos de reconexão com o XP.

OBS: Caso este balão não seja exibido, confira se o serviço **configurações zero sem fio** está ativo.

Configuração de Cliente - Software Livre (Linux)

Como opção ao cliente 802.1X desenvolvido pela Microsoft a comunidade de software livre produziu uma solução bastante interessante chamada oficialmente Xsupplicant. Projetado para rodar nos clientes Linux como um utilitário de linha de comando. As novas versões funcionam no Windows incluindo uma interface gráfica de usuário robusta. Este pacote reúne diversas facilidades administrativas e de acompanhamento da autenticação por meio de logs mais detalhados, associadamente o administrador pode controlar por meio deste cliente diversos parâmetros da autenticação. Figuras 4.11 e 4.12.

Os requisitos básicos da configuração para Windows XP valem para os demais sistemas operacionais que possuem suplicante 802.1X, ou seja, para outro SO basta configurar os mesmos parâmetros.

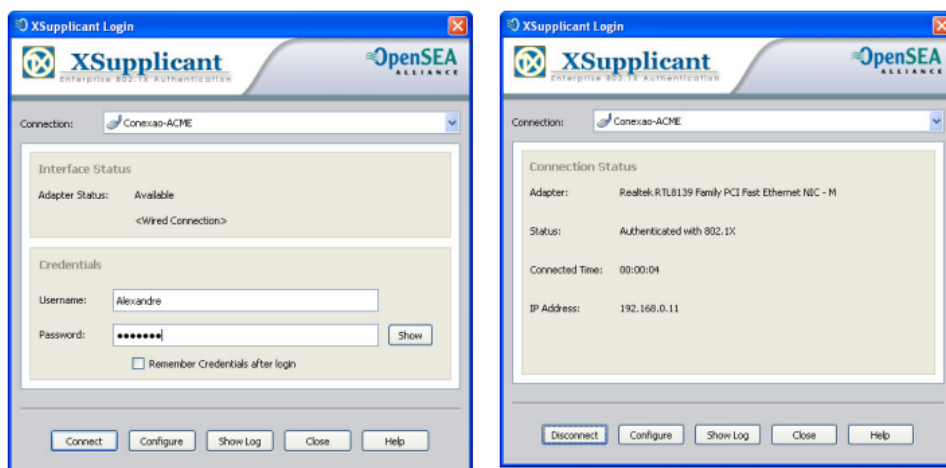


Figura 3.11: Fornecendo as Credenciais e Indicativo do Resultado

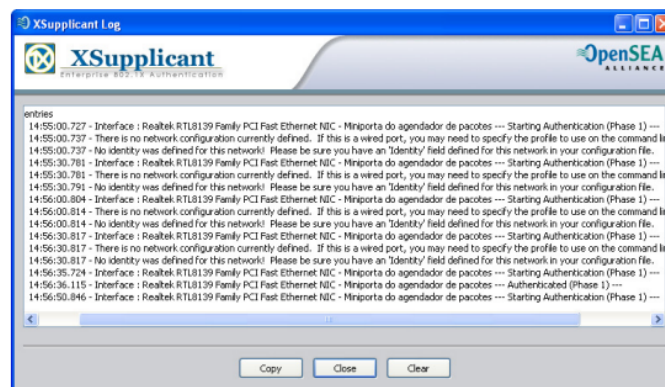


Figura 3.12: Logs do Processo

Além do Xsupplicant, existem no mercado diversos suplicantes que podem ser utilizados para o processo de autenticação baseado no padrão IEEE 802.1X. Dentre os mais conhecidos a Figura 4.13 mostra a comparação entre as funcionalidades destes suplicantes.

Cientes	98/ME	XP/2K	OS X	Linux	Pckt PC	TLS	PEAP	TTLS	Licença
Win Nativo	✗	✓	✗	✗	✗	✓	CHAP v2	✗	Nativo
OSX Nativo	✗	✗	✓	✗	✗	✓	✓	✓	Nativo
SecureW2	✗	✓	✗	✗	✓	✗	✗	✓	Free
Odyssey	✓	✓	✗	✗	✓	✓	✓	✓	\$\$
AEGIS	✓	✓	✓	✓	✓	✓	✓	✓	\$\$
wpa_supp	✓	✓	✗	✓	✗	✓	✓	✓	Free

Figura 3.13: Comparação ente softwares suplicantes

3.1.8 Ferramentas de Administração

Webmin

Webmin é uma ferramenta gráfica de gerenciamento de servidores Linux. O Webmin tem uma interface muito amigável e pode ser acessada de qualquer lugar através de um navegador. Exemplo: `https://(ip do servidor):(porta de utilização)`

Com o Webmin podemos configurar serviços, rede, hardware e sistema em modo gráfico.

Download

```
# wget http://sourceforge.net/projects/webadmin/files/webmin/1.530/webmin-1.530-1.noarch.rpm
```

Instalação

Abra o terminal como root e digite `rpm -ivh + o nome do pacote`

```
# rpm -ivh webmin-1.530-1.noarch.rpm
```

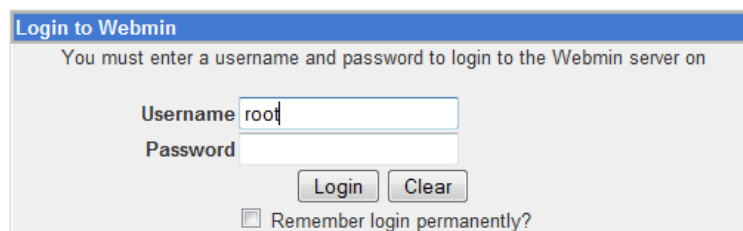
Se correu tudo bem aparecerá essa mensagem na tela:

Webmin install complete. You can now login to `https://nomedohost:10000/` as root with your root password.

Iniciando o Webmin

Abra seu navegador predileto e digite:

`https://nomedohost:10000/`



Login to Webmin

You must enter a username and password to login to the Webmin server on

Username

Password

Remember login permanently?

Figura 3.14: Iniciando o Webmin

Configurando o Webmin

Adicionando object class e atributos.

The screenshot shows the Webmin LDAP configuration interface. It includes sections for:

- Base for users:** From NSS config file, `ou=Usuarios,dc=cursou`
- Base for groups:** From NSS config file, `ou=Grupos,dc=cursou`
- Other objectClasses to add to new users:**
- Other objectClasses to add to new groups:**
- Show fields for given name and surname?:** Yes, No
- Object class to add for given name?:**
- Order for first name and surname:** Surname, Firstname, Firstname Surname
- Full path to sLappasswd program:**
- LDAP attributes:**
 - LDAP properties for all new users (In fieldname: value format):**
 - LDAP properties for modified users (In fieldname: value format):**
 - Extra LDAP user properties to allow editing of (In fieldname description format):**

Figura 3.15: object class e atributos

Usuario Cadastrado

Índice do Módulo

LDAP Attributes

For uid=tiago,ou=Usuarios,dc=networklogin,dc=ufop,dc=br

Attribute name	Values
radiusTunnelType	VLAN
radiusTunnelMediumType	TMT802
sambaSID	S-1-5-21-3562043904-3900313900-1968379896-2024
sambaPrimaryGroupSID	1009
cn	Tiago Rodrigues Chaves
uidNumber	512
sambaAcctFlags	[U]
gecos	Tiago Rodrigues Chaves
shadowLastChange	14309
sambaPwdLastSet	1288183253
userPassword	{md5}\$1\$88183253\$USV13kexTHGFxPbdw..b/
sambaLMPassword	CCF9155E3E7DB453AAD3B435B51404EE
uid	tiago
sambaPwdCanChange	1288183253
homeDirectory	/home/tiago
objectClass	posixAccount, shadowAccount, person, radiusprofile, sambaSamAccount
gidNumber	4
sambaNTPassword	3DBDE697D71690A769204BEB12283678
radiusTunnelPrivateGroupId	203
sn	Tiago Rodrigues Chaves
loginShell	/bin/sh

⏪ Voltar à user details | Voltar à lista de usuários

Figura 3.16: Usuario Cadastrado

daloRADIUS

daloRADIUS é uma plataforma web escrito em PHP e Javascript destinado ao gerenciamento de um servidor FreeRadius. Utiliza uma camada de abstração de dados para realizar o Accounting, possui gerenciamento de usuários, relatórios gráficos, contabilidade e se integra com o Google Maps para a geo-localização (SIG).

Instalando e configurando o daloRADIUS

Com FreeRADIUS + MySQL instalado, instale o daloRADIUS. Antes da instalação, verifique se o Apache, PHP e MySQL foram instalados e estão funcionando.

Obtenha a versão mais recente do daloRADIUS <http://sourceforge.net/projects/daloradius/files/>

Extraia o tarball para `/var/www/daloradius`.

Importe o esquema daloRADIUS DB a partir de `/var/www/contrib/daloradius/db/mysql-daloradius.sql`

```
# Mysql-u root-raio p <mysql-daloradius.sql
```

Edite o arquivo `/var/www/library/daloradius/daloradius.conf.php` com as credenciais de acesso para a base de dados.

Acesse o daloRADIUS através de um navegador web em `http:// <nomedohost> /daloradius`.

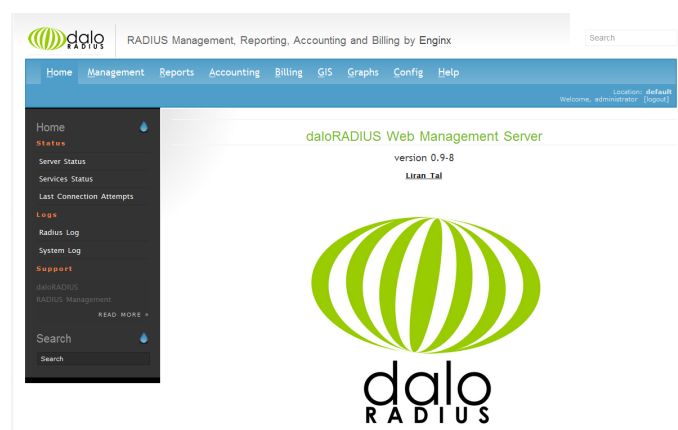


Figura 3.17: Acessando o daloRADIUS

Capítulo 4

CARACTERIZAÇÃO DO PROBLEMA

Este capítulo caracteriza o problema estudado neste trabalho. Na Seção 3.1 o problema é contextualizado. A Seção 3.2 apresenta detalhadamente a situação atual da GigaUFOPnet. É apresentada na Seção 3.3 a situação desejada após a conclusão do trabalho.

4.1 Contextualização

Controles de acesso, físicos ou lógicos, têm como objetivo proteger equipamentos, aplicativos e arquivos de dados contra perda, modificação ou divulgação não autorizada. Os sistemas computacionais, bem diferentes de outros tipos de recursos, não podem ser facilmente controlados apenas com dispositivos físicos, como cadeados, alarmes ou guardas de segurança.

A proteção aos recursos computacionais em uma rede de computadores é feita normalmente por meio de um identificador de usuário e por uma senha durante o processo de entrada no sistema. Fucapi (2010)

A necessidade de autenticação de usuários em uma rede de computadores é o principal aspecto para a segurança da informação e está associada à possibilidade de acesso restrito a uma determinada área ou serviço da rede, ou seja, se não for possível identificar uma pessoa que esteja tentando acessar uma rede de computadores, nenhum outro tópico de segurança fará sentido.

O crescimento de serviços em redes aumenta a necessidade de segurança para implantar um sistema de autenticação e identificação confiável.

Para assegurar que apenas pessoas ou computadores autorizados possam acessar ou modificar os dados armazenados em uma rede, se deve ter obrigatoriamente a identificação, como método para estabelecer a identidade do usuário no sistema e a autenticação, como meio para verificar a veracidade da identidade do usuário. Estas duas condições estão intimamente

relacionadas e são freqüentemente relacionadas em conjunto, contudo descrevem duas funções separadas e distintas:

Identificação - Processo que ocorre durante o login inicial quando uma pessoa provê algum tipo de identificação de segurança, como um nome de usuário único, que identifica o mesmo para o sistema "Este é quem eu sou". É uma afirmação de identidade;

Autenticação - Processo de verificação que exige do usuário uma prova de sua identidade, também única, e que comprove a veracidade da mesma para o sistema, tipicamente uma senha, para afirmar que a identidade está sendo assumida pelo seu legítimo dono. A autenticação assegura para o sistema, "Esta é uma informação privada que prova que eu sou quem digo ser". É a prova da identidade.

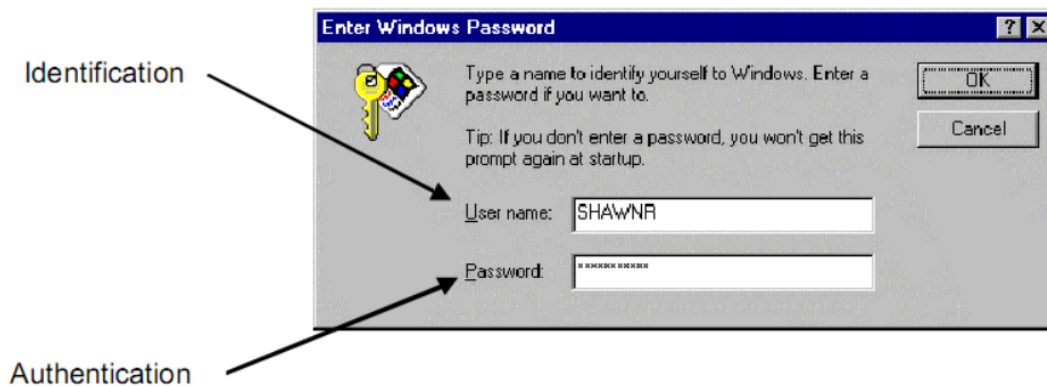


Figura 4.1: Exemplo de Identificação e Autenticação.

Embora outros métodos de autenticação estejam disponíveis, as senhas ainda são a forma mais comum de autenticação usada nas redes de computadores. Pinheiro (2009)

4.2 Situação Atual da GigaUFOPnet

GigaUFOPnet é a rede de computadores da UFOP, administrada pelo NTI - Núcleo de Tecnologia da Informação, seu projeto foi idealizado em 2005 e executado nos anos de 2006 e 2007 dentro da UFOP. Com a implantação do projeto, o fluxo da rede de dados foi ampliado, links que operavam com velocidades de 10/100MBps, passaram a operar com até 1000MBps, possibilitando o uso de novas tecnologias como a vídeo-conferência, que aproxima o contato entre as diversas unidades e campi da UFOP.

A GigaUFOPnet, interliga diversas unidades e campi da UFOP, nas cidade de Ouro Preto, Mariana e João Monlevade.

4.2.1 Unidades e Campi interligados pela GigaUFOPnet

Campus Ouro Preto
Unidades do Morro do Cruzeiro
IFAC - Instituto de Filosofia, Artes e Cultura
EFAR - Escola de Farmácia (Centro)
EM - Escola de Minas (Centro)/REMOP
Centro de Convenções/Reitoria/PROEX/NAJOP/ASSUFOP

Tabela 4.1: Unidades do Campus Ouro Preto

Campus Mariana
ICHS - Instituto de Ciências Humanas e Sociais
ICSA - Instituto de Ciências Sociais e Aplicadas

Tabela 4.2: Unidades do Campus Mariana

Campus João Monlevade
ICEA - Instituto de Ciências Exatas e Aplicadas

Tabela 4.3: Unidades do Campus João Monlevade

4.2.3 Usuários da GigaUFOPnet

Os usuários da GigaUFOPnet estão divididos em três categorias, corpo docente, corpo técnico-administrativo e corpo discente. O corpo docente conta com 752 professores, o corpo técnico-administrativo é composto por 751 funcionários, quanto ao corpo discente, são 13.571 alunos na graduação, sendo 5.195 na modalidade a distância. Na pós-graduação, são 929 alunos. Sendo assim, atualmente a GigaUFOPnet possui 16.003 usuários que utilizam seus recursos computacionais.

Categoria	Quantidade
Professores	752
Técnicos administrativos	751
Alunos graduação (<i>presencial</i>)	8376
Alunos graduação (<i>distância</i>)	5195
Alunos pós-graduação	929
Total	16003

Tabela 4.4: Número de Usuários da GigaUFOPnet em 2010

4.2.4 Método Atual para Acessar a GigaUFOPnet

Cada dispositivo de rede utilizado tanto para redes ethernet como para redes sem fio, deve ter um número único de identificação definido pelo fabricante e controlado pelo IEEE, permitindo assim, teoricamente, identificar de forma inequívoca um equipamento em relação a qualquer outro fabricado mundialmente, seja ele de fabricantes diferentes. RUFINO (2007)

Como cada placa de rede possui um número de endereço físico (Endereço MAC), a maneira utilizada atualmente para controlar o acesso dos usuários a GigaUFOPnet, é associando o endereço MAC do computador do usuário a um número IP válido. O endereço MAC é cadastrado no servidor DHCP associado a determinado número IP, de forma que o computador do usuário receba sempre este número IP, restringindo assim o acesso somente quem estiver com o equipamento cadastrado. Essa técnica visa somente autorizar o equipamento e não o usuário. Os dados dos usuários ficam cadastrados em outra base de dados que associa o número IP atribuído, ao seu endereço MAC e aos seus dados pessoais.

Porém esta solução adotada apresenta uma série de problemas que podem ser listados abaixo:

- Necessidade de pré-configuração de cada computador para acesso a rede;
- Problemas de acesso e segurança;
- Falsificação/Clonagem de endereços Mac para acesso indevido;
- Possibilidade de acesso indevido de algum ponto pré-configurado disponível;
- Problemas na identificação do usuário conectado à rede;
- Distribuição estática de VLANs;
- Inviabilidade de construção de uma rede sem fio controlada e segura;
- Dificuldade de permitir mobilidade aos usuários para acessarem à rede;
- Alta demanda de números de IP's válidos.

4.2.5 Cenários Comuns Sem a Implantação do Projeto

Os cenários apresentados abaixo são comuns no cotidiano da GigaUFOPnet e a solução dos casos que causam falhas ao funcionamento da rede, tornam-se confusos e dispendiosos para a equipe de administração de redes de computadores devido a falta de um processo de autenticação que estabeleça proteção e controle para a rede interna.

- **Cenário 1: Situação Normal.**

Rede normal com um PC de usuário conectado a um switch Ethernet. O PC do usuário obtém o endereço IP cadastrado para o seu equipamento do servidor DHCP quando ele entra na rede. Tudo está bem.

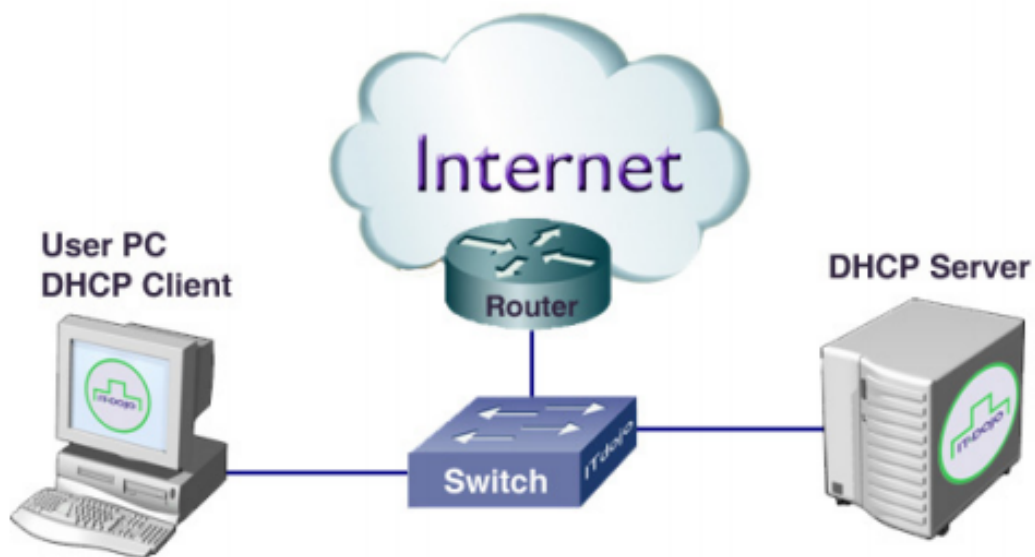


Figura 4.3: Usuário cadastrado e conectado a rede através de um computador cadastrado

- **Cenário 2: Situação propensa a causar falhas de segurança.**

O usuário traz seu próprio switch ou hub e conecta à rede clonando o MAC do seu computador cadastrado. De acordo com as configurações realizadas, possivelmente através da utilização do computador cadastrado como um servidor NAT, o usuário conecta seu laptop pessoal e o computador. Tanto o computador e o laptop pessoal passam a obter endereços de IP inválidos, dificultando a identificação do mesmo na rede. Dessa forma o laptop pessoal do usuário está na rede, assim como qualquer outro computador que for conectado ao hub/switch.

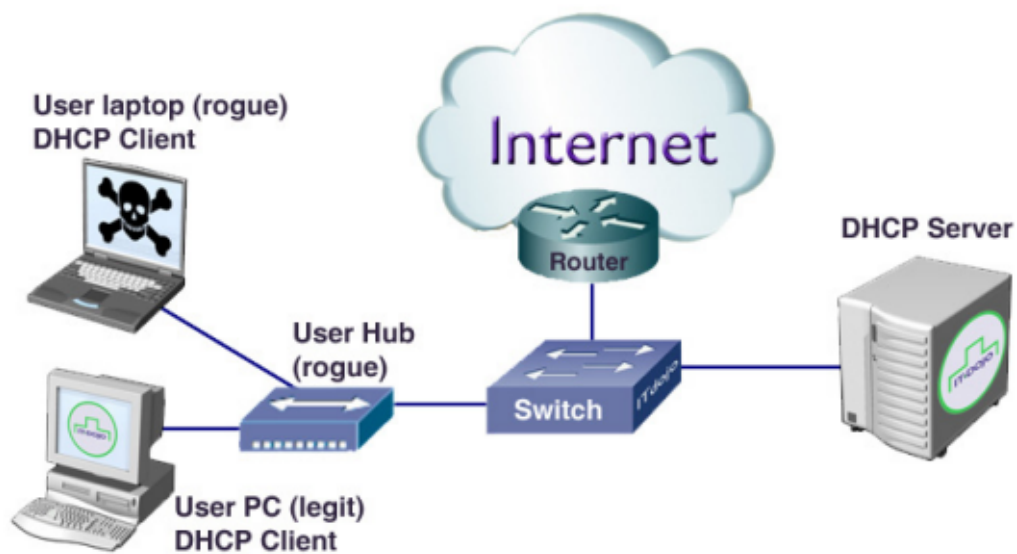


Figura 4.4: Usuário instala Hub/Switch para aumentar o número de pontos de acesso

Aqui está uma lista de algumas das coisas ruins que podem acontecer:

- O usuário pode cometer incidentes de segurança sem ser identificado;
- O laptop do usuário pode estar infectado por um vírus que pode infectar a rede;
- O usuário pode instalar softwares de compartilhamento de rede promovendo pirataria;
- O usuário poderá fazer uso de protocolos e ou programas que estão em violação direta da política de segurança corporativa.

- **Cenário 3 : Situação muito propensa a causar falhas de segurança.**

Talvez o pior cenário possível é quando um usuário faz por conta própria a instalação de um ponto de acesso sem fio (AP) para que ele possa ter conectividade sem fio com o seu laptop pessoal durante o trabalho. Este é um problema cada vez mais comuns entre os usuários da GigaUFOPnet, que costumam colocar estes AP's sem qualquer forma de proteção. Configurando apenas um ponto de acesso aberto e sem criptografia necessária. Este cenário torna não só o laptop do usuário um cliente DHCP mas também faz cada usuário dentro da faixa (radiofrequência) do AP um potencial cliente.

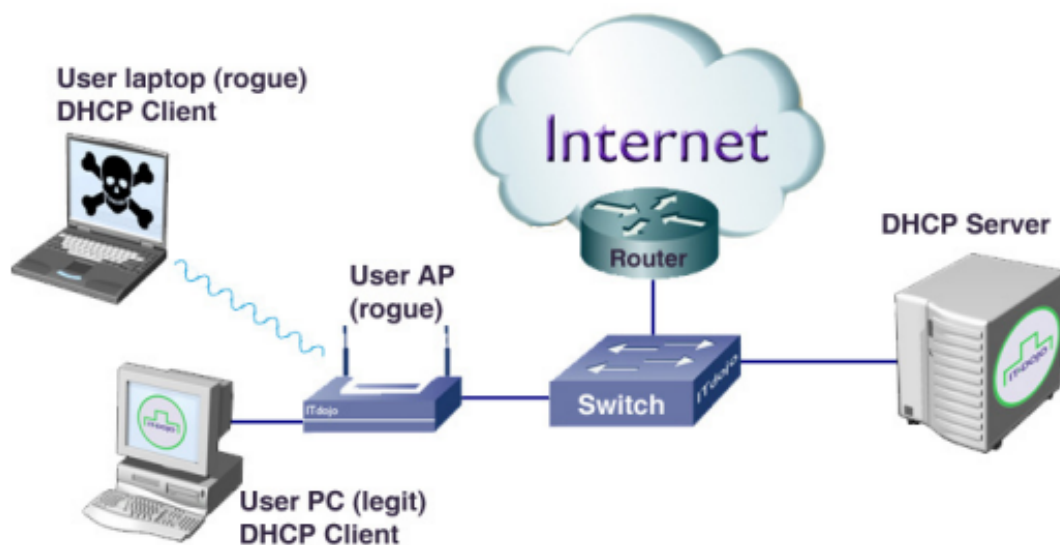


Figura 4.5: Usuário instala Access Point

Aqui está uma lista de todas as coisas ruins que podem acontecer:

- Tudo do Cenário 2, mas agora se aplicam a qualquer um na área de cobertura do AP que tenha placa de rede sem fio em seu laptop, PDA, etc.

- Contabilizar todos os acessos e uso dos usuários;
- Identificar usuários mal intencionados utilizando a rede;
- Criar Login e Senha únicos para acesso a rede e outros serviços da UFOP;
- Disponibilizar acesso a rede para dispositivos móveis com segurança.

Capítulo 5

RESULTADOS

Os resultados apresentados durante a realização deste estudo garantem que com a implantação deste projeto em toda a GigaUFOPnet, o processo de autenticação baseado no padrão IEEE 802.1X, reforçará a segurança da rede e acrescentará uma solução de controle de acesso abrangente, garantindo desempenho e confiabilidade no acesso.

Os testes realizados garantem a autenticação segura de todos os usuários da GigaUFOPnet. Quando um usuário fizer logon no computador, a autenticação na rede será realizada usando as mesmas credenciais do usuário para acessar o computador local.

Uma grande vantagem desta configuração é possibilidade de permitir aos usuários acessarem diversos computadores em seu setor/departamento, de acordo com as políticas definidas para o seu perfil, utilizando sempre as mesmas credenciais. Dessa forma, será possível para os usuários terem acesso a sua rede local e privilégios a partir de qualquer ponto da rede.

Apesar dos diversos suplicantes existentes, dado o contexto da GigaUFOPnet os testes comprovaram que a solução utilizando um controlador de domínio, integrado ao suplicante nativo do sistema operacional para o processo de autenticação no momento do logon, mostrou-se a mais efetivo para o acesso à rede através dos computadores da própria instituição.

Os suplicantes em sua maioria apesar de realizarem o processo de autenticação corretamente, dado o cenário em que a utilização dos computadores da instituição em sua maioria são compartilhados por diversos usuários, torna-se necessário um mecanismo intuitivo para que o usuário acesse a rede e após o uso encerre a sua conexão.

Dado a característica do acesso à rede integrado ao logon do computador e as demais funcionalidades que podem ser exploradas com a utilização de um controlador de domínio, como compartilhamento de arquivos. A escolha desta arquitetura tornou-se ideal para a GigaUFOPnet.

Está arquitetura também permite o acesso à rede de computadores pessoais sem a necessidade de adicionar estes computadores ao domínio, os mesmos poderão utilizar o próprio suplicante nativo do sistema operacional.

Todos os acessos serão contabilizados, através da ferramenta daloRadius é possível exibir

os logs de acesso de todos os usuários, estes dados podem ser filtrados por nome de usuário, IP, data e horário de acesso e etc. Assim usuários mal intencionados utilizando a GigaUFOPnet poderão ser identificados para responderem por incidentes de segurança cometidos.

Toda a arquitetura projetada para a rede com fio pode ser estendida para a rede sem fio da GigaUFOPnet, permitindo acesso à rede para dispositivos móveis com segurança. Através da adição de um Switch Controller e Access Points que suportem 802.1X, a nova arquitetura permitirá a implantação de uma rede wireless gerenciável, escalável e com recursos de segurança equivalentes aos recursos presentes na rede cabeada.

The screenshot shows a web interface for RADIUS Accounting. The top navigation bar includes Home, Management, Reports, Accounting (selected), Billing, GIS, Graphs, Config, and Help. Below the navigation bar, there are tabs for General, Custom, Hotspot, and Maintenance. The main content area is titled "All Users Accounting" and features a "CSV Export" button. A table displays accounting records with columns for ID, HotSpot, Username, IP Address, Start Time, Stop Time, Total Time, Upload (Bytes), Download (Bytes), Termination, and NAS IP Address. The table contains 11 rows of data, with the last row (ID 1706) showing a termination reason of "NAS-Error".

ID	HotSpot	Username	IP Address	Start Time	Stop Time	Total Time	Upload (Bytes)	Download (Bytes)	Termination	NAS IP Address
1696		host/REDES01.UFOP	200.131.72.61	2010-12-02 11:19:29	2010-12-02 11:19:34	5 seconds	7.37 Kb	3.93 Kb	255	200.131.208.236
1697		UFOP=5C=5C=5C=5Cvlan202	200.131.72.93	2010-12-02 11:19:35	2010-12-02 11:19:39	4 seconds	5.2 Kb	3.99 Kb	255	200.131.208.236
1698		host/REDES01.UFOP	200.131.72.61	2010-12-02 11:19:40	2010-12-02 11:20:23	43 seconds	27.28 Kb	13.36 Kb	255	200.131.208.236
1699		host/REDES01.UFOP		2010-12-02 11:25:26	2010-12-02 11:25:39	14 seconds	2.05 Kb	4.27 Kb	255	200.131.208.236
1700		UFOP=5C=5C=5C=5Ccluclano	200.131.72.61	2010-12-02 11:25:40	2010-12-02 11:25:45	5 seconds	4.57 Kb	3.5 Kb	255	200.131.208.236
1701		host/REDES01.UFOP	200.131.72.61	2010-12-02 11:25:45	2010-12-02 11:41:01	15 minutes, 17 seconds	67.85 Kb	105.95 Kb	NAS-Error	200.131.208.236
1702		host/REDES01.UFOP	200.131.72.61	2010-12-02 13:22:15	2010-12-02 13:22:11	19 seconds	9.63 Kb	8.41 Kb	255	200.131.208.236
1703		UFOP=5C=5C=5C=5Cvlan202	200.131.72.93	2010-12-02 13:22:11	2010-12-02 13:23:03	53 seconds	36.67 Kb	29.41 Kb	255	200.131.208.236
1704		host/REDES01.UFOP	200.131.72.61	2010-12-02 13:23:04	2010-12-02 13:23:18	14 seconds	15.22 Kb	9.07 Kb	255	200.131.208.236
1705		UFOP=5C=5C=5C=5Csergio	200.131.72.126	2010-12-02 13:23:18	2010-12-02 13:23:27	10 seconds	10.22 Kb	4.78 Kb	255	200.131.208.236
1706		host/REDES01.UFOP	200.131.72.61	2010-12-02 13:23:28	2010-12-02 13:25:51	2 minutes, 23 seconds	64.8 Kb	41.64 Kb	255	200.131.208.236

Figura 5.1: Accounting do usuários após a conexão utilizando 802.1X

Capítulo 6

CONCLUSÕES E TRABALHOS FUTUROS

A implantação de uma solução de controle de acesso acrescenta vários benefícios importantes em uma rede de computadores.

Podemos afirmar que a implementação de um processo de autenticação na rede GigaUFOPnet irá assegurar que o NTI esteja gerenciando adequadamente a segurança da rede de computadores, protelar por muito tempo a implantação do 802.1X contribuirá para tornar a segurança da rede cada vez mais ineficaz.

Por se tratar de um projeto inovador no contexto das IFES - Instituições Federais de Ensino Superior, mesmo que os elementos utilizados sejam bem conhecidos individualmente, existe pouca pesquisa no que se diz respeito à integração destes elementos.

A integração de todos os serviços e a adição do controlador de domínios SAMBA, permitindo o acesso a rede no momento do logon, pode ser considerada a maior dificuldade encontrada, devido a falta de documentação. Além disso, muito planejamento é necessário para modificar a arquitetura da GigaUFOPnet e acrescentar o serviço de autenticação sem prejudicar o uso contínuo da rede por seus usuários. Portanto, não se trata de um projeto trivial a ser realizado em curto prazo.

Este trabalho tem como principal contribuição o estudo da implementação do processo de autenticação baseado no padrão IEEE 802.1X utilizando o protocolo Radius e o serviço de diretório LDAP na GigaUFOPnet. Algumas vantagens da implementação deste projeto além de toda a segurança acrescentada, está no fato de utilizar apenas softwares livres, o que diminui expressivamente o gasto com licenças proprietárias.

Esta solução ainda está em fase de testes, mas já demonstra melhorias que serão incorporadas na administração dos usuários que utilizam os serviços computacionais providos pela GigaUFOPnet.

Como trabalhos futuros, o próximo passo desse projeto será aplicar toda a infraestrutura testada em laboratório na rede de algum prédio da UFOP.

Posteriormente, a implantação em todos os campi da UFOP será realizada por partes para garantir o pleno funcionamento da rede GigaUFOPnet e nenhum prejuízo para os seus usuários. De maneira sistemática e coordenada todas as modificações necessárias serão aplicadas em todos os prédios da UFOP.

Outro trabalho futuro que deverá ser realizado, será o desenvolvimento, em conjunto com a equipe de Solução da Informação do NTI, de um sistema administrativo, integrado a Minha UFOP, que permitirá toda a administração dos usuários da GigaUFOPnet, esta unificação da base de dados utilizada nos sistemas Minha UFOP, permitirá o uso das mesmas credenciais já utilizadas para acessar os sistemas desenvolvidos pelo NTI atualmente. Esta ferramenta também deverá contar com uma ferramenta de Accounting específica para as necessidades da GigaUFOPnet, substituindo a ferramenta daloRadius.

Ainda podemos citar, a integração de outros serviços como o VOIP RNP e WEBMAIL, criando assim uma base única para acesso a todos os serviços computacionais oferecidos pela UFOP.

Apêndice A

Arquivos de Configuração do SAMBA

A.1 smb.conf

```
#===== Global Settings =====

[global]

#
workgroup = UFOP
server string = Universidade Federal de Ouro Preto %v
netbios name = SMBUFOP

; interfaces = lo eth0 192.168.12.2/24 192.168.13.2/24
; hosts allow = 127. 192.168.12. 192.168.13.

# ----- Logging Options -----
#
log file = /var/log/samba/%m.log
max log size = 50

security = user
#socket options = TCP_NODELAY SO_SNDBUF=8192 SO_RCVBUF=8192
passdb backend = ldapsam:ldap://localhost
#Acima esta a configuracao de onde o servidor samba ira pegar a base de dados.

# ----- Domain Members Options -----
```

```
domain master = yes
domain logons = yes

logon path =
logon home =
logon script = logon.bat

; add machine script = smbldap-useradd -t 0 -w "%u"

# ----- Browser Control Options -----
#
local master = yes
os level = 200
preferred master = yes

#----- Name Resolution -----

wins support = yes
wins proxy = no
dns proxy = no

# ----- Printing Options -----
#
load printers = yes
cups options = raw
; printcap name = /etc/printcap
#obtain list of printers automatically on SystemV
; printcap name = lpstat
; printing = cups

# ----- Filesystem Options -----
#BASE AUTENTICACAO LDAP
ldap admin dn = cn=admin,dc=networklogin,dc=ufop,dc=br
ldap ssl = off
ldap delete dn = no
ldap user suffix = ou=Usuarios
ldap group suffix = ou=Grupos
```

```
ldap machine suffix = ou=Maquinas
ldap suffix = dc=networklogin,dc=ufop,dc=br
#permitir que usuarios membros do grupo "Domain Admins"
#insiram estacoes no dominio samba
enable privileges = Yes
#controle ACL pelo Windows Explorer
map acl inherit = Yes
inherit acls = Yes
inherit permissions = Yes
nt acl support = Yes
add machine script = /usr/sbin/smbldap-useradd -t 0 -w "%u"
#
Level2 oplocks = no
ntlm auth = Yes
lanman auth = Yes
client ntlmv2 auth = Yes
use client driver = Yes
client lan auth = Yes

#===== Share Definitions =====

[homes]
comment = Home Directories
browseable = no
writable = yes
valid users = %S
; valid users = MYDOMAIN%S

;[printers]
; comment = All Printers
; path = /var/spool/samba
; browseable = no
; guest ok = no
; writable = no
; printable = yes

# Un-comment the following and create the netlogon directory for Domain Logons
```



```
[netlogon]
comment = Network Logon Service
path = /etc/samba/netlogon
guest ok = yes
writable = yes
share modes = no

# Un-comment the following to provide a specific roving profile share
# the default is to use the user's home directory
; [Profiles]
; path = /var/lib/samba/profiles
; browseable = no
; guest ok = yes

# A publicly accessible directory, but read only, except for people in
# the "staff" group
; [publico]
; comment = Public Stuff
; path = /home/publico
; public = yes
; writable = yes
; printable = no
; write list = +staff
```

Apêndice B

Arquivos de Configuração Integração SAMBA-LDAP

B.1 smbldap_bind.conf

```
#####  
# Credential Configuration #  
#####  
# Notes: you can specify two differents configuration if you use a  
# master ldap for writing access and a slave ldap server for reading access  
# By default, we will use the same DN (so it will work for standard Samba  
# release)  
slaveDN="cn=admin,dc=networklogin,dc=ufop,dc=br"  
slavePw="*****"  
masterDN="cn=admin,dc=networklogin,dc=ufop,dc=br"  
masterPw="*****"
```

B.2 smbldap.conf

```
#####
# General Configuration
#####

SID="S-1-5-21-3562043904-3900313900-1968379896"
sambaDomain="UFOP"

#####
# LDAP Configuration
#####

slaveLDAP="localhost"
slavePort="389"
masterLDAP="localhost"
masterPort="389"
ldapTLS="0"
ldapSSL="0"
verify="require"
cafile="/etc/smbldap-tools/ca.pem"
clientcert="/etc/smbldap-tools/smbldap-tools.iallanis.info.pem"
clientkey="/etc/smbldap-tools/smbldap-tools.iallanis.info.key"
suffix="dc=networklogin,dc=ufop,dc=br"
usersdn="ou=Usuarios,${suffix}"
computersdn="ou=Maquinas,${suffix}"
groupsdn="ou=Grupos,${suffix}"
idmapdn="ou=Idmap,${suffix}"
sambaUnixIdPool="sambaDomainName=UFOP,${suffix}"
scope="sub"
hash_encrypt="SSHA"
crypt_salt_format="%s"

#####
# Unix Accounts Configuration
#####

userLoginShell="/bin/bash"
userHome="/home/%U"
```

```
userHomeDirectoryMode="700"
userGecos="System User"
defaultUserGid="513"
defaultComputerGid="515"
skeletonDir="/etc/skel"
defaultMaxPasswordAge="45"
```

```
#####
```

```
# SAMBA Configuration
```

```
#####
```

```
userSmbHome="//PDC-SRV/%U"
userProfile="//PDC-SRV/profiles/%U"
userHomeDrive="H:"
userScript="logon.bat"
mailDomain="iallanis.info"
```

```
#####
```

```
# SMBLDAP-TOOLS Configuration (default are ok for a RedHat)
```

```
#####
```

```
with_smbpasswd="0"
smbpasswd="/usr/bin/smbpasswd"
with_slappasswd="0"
slappasswd="/usr/sbin/slappasswd"
```

Apêndice C

Arquivos de Configuração do LDAP

C.1 ldap.conf

```
#
# LDAP Defaults
#

# See ldap.conf(5) for details
# This file should be world readable but not world writable.

BASE dc=networklogin, dc=ufop, dc=br
HOST 127.0.0.1
###BASE dc=ntiufop,dc=org,dc=br
#URI ldap://ldap.example.com ldap://ldap-master.example.com:666

#SIZELIMIT 12
#TIMELIMIT 15
#DEREF never
###TLS_CACERTDIR /etc/openldap/cacerts
###TLS_CACERT /etc/openldap/cacerts/nti.pem
###TLS_REQCERT never
####ldap_version 3
###$URI ldap://localhost/
```

C.2 sldap.conf

```
#
# See slapd.conf(5) for details on configuration options.
# This file should NOT be world readable.
#
include /etc/openldap/schema/core.schema
include /etc/openldap/schema/cosine.schema
include /etc/openldap/schema/inetorgperson.schema
include /etc/openldap/schema/nis.schema
include /etc/openldap/schema/freeradius.schema
include /etc/openldap/schema/samba.schema
# Allow LDAPv2 client connections.  This is NOT the default.
allow bind_v2

# Do not enable referrals until AFTER you have a working directory
# service AND an understanding of referrals.
#referral ldap://root.openldap.org

pidfile /var/run/openldap/slapd.pid
argsfile /var/run/openldap/slapd.args
loglevel          256

#####
# ldbm and/or bdb database definitions
#####

database bdb
suffix "dc=networklogin,dc=ufop,dc=br"
rootdn "cn=admin,dc=networklogin,dc=ufop,dc=br"
rootpw {SSHA}GgvbLOEftbTBCBjUX3v/00iYFG54KCVA

directory /var/lib/ldap

# Indices to maintain for this database
index objectClass          eq,pres
index ou,cn,mail,surname,givenname  eq,pres,sub
index uidNumber,gidNumber,loginShell eq,pres
```

```
index uid,memberUid          eq,pres,sub
index nisMapName,nisMapEntry eq,pres,sub
#Indice para replica dados
index entryCSN                eq
index entryUUID               eq
#Indice para SAMBA
index sambaSID eq
index sambaPrimaryGroupSID eq
index SambaDomainName eq

access to attrs=userPassword,sambaLMPassword,sambaNTPassword
    by self write
    by anonymous auth
    by * none
access to *
    by * read
```

Apêndice D

Arquivos de Configuração do FreeRadius

D.1 radius.conf

```
##
## radiusd.conf -- FreeRADIUS server configuration file.
##
prefix = /usr
exec_prefix = /usr
sysconfdir = /etc
localstatedir = /var
sbindir = /usr/sbin
logdir = ${localstatedir}/log/radius
raddbdir = ${sysconfdir}/raddb
radacctdir = ${logdir}/radacct

confdir = ${raddbdir}
run_dir = ${localstatedir}/run/radiusd

log_file = ${logdir}/radius.log

libdir = /usr/lib

pidfile = ${run_dir}/radiusd.pid

user = radiusd
```



```
group = radiusd

max_request_time = 30

delete_blocked_requests = no

cleanup_delay = 10

max_requests = 1024

bind_address = *

port = 0

hostname_lookups = no

allow_core_dumps = no

regular_expressions = yes
extended_expressions = yes

log_stripped_names = yes

log_auth = yes

log_auth_badpass = yes
log_auth_goodpass = yes

usercollide = no

lower_user = no
lower_pass = no

nospace_user = no
nospace_pass = no
  The program to execute to do concurrency checks.
checkrad = ${sbindir}/checkrad
```

```
security {

max_attributes = 200

reject_delay = 1

status_server = no
}

proxy_requests = yes
$INCLUDE ${confdir}/proxy.conf

$INCLUDE ${confdir}/clients.conf

snmp = no
$INCLUDE ${confdir}/snmp.conf

thread pool {

start_servers = 5

max_servers = 32

min_spare_servers = 3
max_spare_servers = 10

max_requests_per_server = 0
}

modules {

pap {
encryption_scheme = clear
}

chap {
authtype = CHAP
}
}
```

```
pam {

pam_auth = radiusd
}

unix {

cache = no

cache_reload = 600

shadow = /etc/shadow

radwtmp = ${logdir}/radwtmp
}

$INCLUDE ${confdir}/eap.conf

mschap {

with_ntdomain_hack = yes
}

ldap {
server = "localhost"
identity = "cn=admin,dc=networklogin,dc=ufop,dc=br"
password = *****
basedn = "ou=Usuarios,dc=networklogin,dc=ufop,dc=br"
filter = "(uid=%{Stripped-User-Name:-%{User-Name}})"

start_tls = no

access_attr = "uid"

dictionary_mapping = ${raddbdir}/ldap.attrmap

ldap_connections_number = 5
```

```
password_attribute = sambaNTPassword
groupname_attribute = cn
    groupmembership_filter = "(|(&(objectClass=SambaGroupMapping)(member=%{Ldap-UserDn}))
(&(objectClass=GroupOfUniqueNames)(uniquemember=%{Ldap-UserDn})))"
timeout = 4
timelimit = 3
net_timeout = 1
set_auth_type = yes
}

realm IPASS {
format = prefix
delimiter = "/"
ignore_default = yes
ignore_null = yes
}

realm suffix {
format = suffix
delimiter = "@"
ignore_default = no
ignore_null = no
}

realm realmpercent {
format = suffix
delimiter = "%"
ignore_default = no
ignore_null = no
}

realm ntdomain {
format = prefix
delimiter = "\\\"
ignore_default = yes
ignore_null = yes
```

```
}
```

```
checkval {  
  item-name = Calling-Station-Id  
  check-name = Calling-Station-Id  
  data-type = string  
}
```

```
preprocess {  
  huntgroups = ${confdir}/huntgroups  
  hints = ${confdir}/hints
```

```
  with_ascend_hack = no  
  ascend_channels_per_line = 23
```

```
  with_specialix_jetstream_hack = no
```

```
  with_cisco_vsa_hack = no  
}
```

```
files {  
  usersfile = ${confdir}/users  
  acctusersfile = ${confdir}/acct_users  
  preproxy_usersfile = ${confdir}/preproxy_users
```

```
  compat = no  
}
```

```
detail {  
  
  detailfile = ${radacctdir}/${Client-IP-Address}/detail-%Y%m%d  
  detailperm = 0600
```

```
}
```

```
  detail auth_log {  
    detailfile = ${radacctdir}/${Client-IP-Address}/auth-detail-%Y%m%d
```

```
detailperm = 0600
}

detail reply_log {
detailfile = ${radacctdir}/${Client-IP-Address}/reply-detail-%Y%m%d

detailperm = 0600
}

sql_log {
path = ${radacctdir}/sql-relay
acct_table = "radacct"
postauth_table = "radpostauth"

Start = "INSERT INTO ${acct_table} (AcctSessionId, UserName, \
NASIPAddress, FramedIPAddress, AcctStartTime, AcctStopTime, \
AcctSessionTime, AcctTerminateCause) VALUES \
('${Acct-Session-Id}', '${Stripped-User-Name:-${User-Name}}', '${NAS-IP-Address}', \
'${Framed-IP-Address}', '%S', '0', '0', '');"
Stop = "INSERT INTO ${acct_table} (AcctSessionId, UserName, \
NASIPAddress, FramedIPAddress, AcctStartTime, AcctStopTime, \
AcctSessionTime, AcctTerminateCause) VALUES \
('${Acct-Session-Id}', '${Stripped-User-Name:-${User-Name}}', '${NAS-IP-Address}', \
'${Framed-IP-Address}', '0', '%S', '${Acct-Session-Time}', \
'${Acct-Terminate-Cause}');"
Alive = "INSERT INTO ${acct_table} (AcctSessionId, UserName, \
NASIPAddress, FramedIPAddress, AcctStartTime, AcctStopTime, \
AcctSessionTime, AcctTerminateCause) VALUES \
('${Acct-Session-Id}', '${Stripped-User-Name:-${User-Name}}', '${NAS-IP-Address}', \
'${Framed-IP-Address}', '0', '0', '${Acct-Session-Time}', '');"

Post-Auth = "INSERT INTO ${postauth_table} \
(user, pass, reply, date) VALUES \
('${Stripped-User-Name:-${User-Name}}', '${Stripped-User-Password:-Chap-Password}', \
'${reply:Packet-Type}', '%S');"
}

acct_unique {
```

```
key = "Stripped-User-Name, Acct-Session-Id, NAS-IP-Address, Client-IP-Address, NAS-Port"
}
```

```
$INCLUDE ${confdir}/sql.conf
```

```
radutmp {
```

```
filename = ${logdir}/radutmp
```

```
username = %{Stripped-User-Name:-%{User-Name}}
```

```
case_sensitive = no
```

```
check_with_nas = yes
```

```
perm = 0600
```

```
callerid = "yes"
```

```
}
```

```
radutmp sradutmp {
```

```
filename = ${logdir}/sradutmp
```

```
perm = 0644
```

```
callerid = "no"
```

```
}
```

```
attr_filter {
```

```
attrsfile = ${confdir}/attrs
```

```
}
```

```
sqlcounter dailycounter {
```

```
counter-name = Daily-Session-Time
```

```
check-name = Max-Daily-Session
```

```
sqlmod-inst = sql
```

```
key = User-Name
```

```
reset = daily
```

```
query = "SELECT SUM(AcctSessionTime - \
          GREATEST((%b - UNIX_TIMESTAMP(AcctStartTime)), 0)) \
```

```
        FROM radacct WHERE UserName='%{%k}' AND \
        UNIX_TIMESTAMP(AcctStartTime) + AcctSessionTime > '%b'"

        AcctStartTime::ABSTIME::INT4 + AcctSessionTime > '%b'"
        UserName='%{%k}' AND AcctStartTime > FROM_UNIXTIME('%b')"
    }

sqlcounter monthlycounter {
counter-name = Monthly-Session-Time
check-name = Max-Monthly-Session
sqlmod-inst = sql
key = User-Name
reset = monthly

query = "SELECT SUM(AcctSessionTime - \
        GREATEST((%b - UNIX_TIMESTAMP(AcctStartTime)), 0)) \
        FROM radacct WHERE UserName='%{%k}' AND \
        UNIX_TIMESTAMP(AcctStartTime) + AcctSessionTime > '%b'"
        FROM_UNIXTIME('%b') AND FROM_UNIXTIME('%e')"
}

always fail {
rcode = fail
}

always reject {
rcode = reject
}

always ok {
rcode = ok
simulcount = 0
mpp = no
}

expr {
}

digest {
}
```



```
exec {
wait = yes
input_pairs = request
}

exec echo {

wait = yes

program = "/bin/echo %{User-Name}"

input_pairs = request

output_pairs = reply

}

ippool main_pool {

range-start = 192.168.1.1
range-stop = 192.168.3.254

netmask = 255.255.255.0

cache-size = 800

session-db = ${raddbdir}/db.ippool

ip-index = ${raddbdir}/db.ipindex

override = no

maximum-timeout = 0
}

}
```

```
instantiate {  
  
    exec  
    expr  
}  
authorize {  
    preprocess  
    auth_log  
    chap  
    mschap  
    suffix  
    ntdomain  
    IPASS  
    eap  
    files  
    ldap  
}  
  
authenticate {  
  
    Auth-Type PAP {  
        pap  
    }  
  
    Auth-Type CHAP {  
        chap  
    }  
  
    Auth-Type MS-CHAP {  
        mschap  
    }  
  
    unix  
  
    Auth-Type LDAP {  
        ldap  
    }
```

```
eap
}

preacct {
preprocess

acct_unique
suffix
files
}
accounting {
sql
radutmp
sql_log
detail
}
session {
radutmp
}
post-auth {
sql
sql_log
Post-Auth-Type REJECT {
sql
}
}
pre-proxy {

}
post-proxy {
eap
}
```

D.2 dictionary

```
#
# This is the master dictionary file, which references the
# pre-defined dictionary files included with the server.
#
# Any new/changed attributes MUST be placed in this file, as
# the pre-defined dictionaries SHOULD NOT be edited.
#
# $Id: dictionary.in,v 1.4 2004/04/14 15:26:20 aland Exp $
#

#
# The filename given here should be an absolute path.
#
$INCLUDE /usr/share/freeradius/dictionary
#$INCLUDE /etc/raddb/dictionary.3com

#
# Place additional attributes or $INCLUDEs here.  They will
# over-ride the definitions in the pre-defined dictionaries.
#
# See the 'man' page for 'dictionary' for information on
# the format of the dictionary files.

#
# If you want to add entries to the dictionary file,
# which are NOT going to be placed in a RADIUS packet,
# add them here.  The numbers you pick should be between
# 3000 and 4000.
#

#ATTRIBUTE My-Local-String 3000 string
#ATTRIBUTE My-Local-IPAddr 3001 ipaddr
#ATTRIBUTE My-Local-Integer 3002 integer

VALUE      Acct-Type      None      3
VALUE      Acct-Type      System    1
VALUE      Acct-Type      Detail    2
```

VALUE	Acct-Type	SQL	0	
ATTRIBUTE	Tunnel-Type		64	integer has_tag
ATTRIBUTE	Tunnel-Medium-Type		65	integer has_tag
ATTRIBUTE	Tunnel-Private-Group-Id	81	string	has_tag
VALUE	Tunnel-Type	VLAN	13	
VALUE	Tunnel-Medium-Type	TMT802	6	

D.3 dictionary.3com

```
#
# dictionary.3com
#
# 3Com specific attributes
#
#

VENDOR      3Com                43

#
# 3Com Attributes
#

ATTRIBUTE   3Com-User-Access-Level  1                Integer  3Com

#
#      3Com-User-Access-Level Values
#

VALUE       3Com-User-Access-Level  Monitor           1
VALUE       3Com-User-Access-Level  Manager           2
VALUE       3Com-User-Access-Level  Administrator     3
```

D.4 dictionary.tunnel

```
#
# dictionary.tunnel
#
# Experimental tunneling attributes.
#
#
# Version: $Id: dictionary.tunnel,v 1.7 2003/03/24 23:21:13 aland Exp $
#

#
# Tunneling Attributes
#
ATTRIBUTE Tunnel-Type 64 integer has_tag
ATTRIBUTE Tunnel-Medium-Type 65 integer has_tag
ATTRIBUTE Tunnel-Client-Endpoint 66 string has_tag
ATTRIBUTE Tunnel-Server-Endpoint 67 string has_tag
ATTRIBUTE Tunnel-Connection-Id 68 string has_tag
ATTRIBUTE Tunnel-Password 69 string has_tag,encrypt=2
ATTRIBUTE Tunnel-Private-Group-Id 81 string has_tag
ATTRIBUTE Tunnel-Assignment-Id 82 string has_tag
ATTRIBUTE Tunnel-Preference 83 integer has_tag
ATTRIBUTE      Acct-Tunnel-Packets-Lost 86 integer
ATTRIBUTE Tunnel-Client-Auth-Id 90 string has_tag
ATTRIBUTE Tunnel-Server-Auth-Id 91 string has_tag

VALUE Framed-Protocol PPTP 9

# Tunnel Type

VALUE Tunnel-Type PPTP 1
VALUE Tunnel-Type L2F 2
VALUE Tunnel-Type L2TP 3
VALUE Tunnel-Type ATMP 4
VALUE Tunnel-Type VTP 5
VALUE Tunnel-Type AH 6
VALUE Tunnel-Type IP 7
```

```
VALUE Tunnel-Type MIN-IP 8
VALUE Tunnel-Type ESP 9
VALUE Tunnel-Type GRE 10
VALUE Tunnel-Type DVS 11
VALUE Tunnel-Type IP-in-IP 12
VALUE Tunnel-Type VLAN 13
#
```

```
# Tunnel Medium Type
```

```
VALUE Tunnel-Medium-Type IP 1
VALUE Tunnel-Medium-Type X25 2
VALUE Tunnel-Medium-Type ATM 3
VALUE Tunnel-Medium-Type Frame-Relay 4
VALUE Tunnel-Medium-Type TMT802 6
```


D.5 ldap.attrmap

```
#
# Mapping of RADIUS dictionary attributes to LDAP directory attributes
# to be used by LDAP authentication and authorization module (rlm_ldap)
#
# Format:
#   ItemType RADIUS-Attribute-Name ldapAttributeName
#
# Where:
#   ItemType           = checkItem or replyItem
#   RADIUS-Attribute-Name = attribute name in RADIUS dictionary
#   ldapAttributeName   = attribute name in LDAP schema
#
# If $GENERIC$ is specified as RADIUS-Attribute-Name, the line specifies
# a LDAP attribute which can be used to store any RADIUS
# attribute/value-pair in LDAP directory.
#
# You should edit this file to suit it to your needs.
#

checkItem $GENERIC$ radiusCheckItem
replyItem $GENERIC$ radiusReplyItem

checkItem Auth-Type radiusAuthType
checkItem Simultaneous-Use radiusSimultaneousUse
checkItem Called-Station-Id radiusCalledStationId
checkItem Calling-Station-Id radiusCallingStationId
checkItem LM-Password sambaLMPassword
checkItem NT-Password sambaNTPassword
checkItem SMB-Account-CTRL-TEXT sambaAcctFlags
checkItem Expiration radiusExpiration
checkItem NAS-IP-Address radiusNASIpAddress

replyItem Service-Type radiusServiceType
replyItem Framed-Protocol radiusFramedProtocol
replyItem Framed-IP-Address radiusFramedIPAddress
replyItem Framed-IP-Netmask radiusFramedIPNetmask
```

```
replyItem Framed-Route radiusFramedRoute
replyItem Framed-Routing radiusFramedRouting
replyItem Filter-Id radiusFilterId
replyItem Framed-MTU radiusFramedMTU
replyItem Framed-Compression radiusFramedCompression
replyItem Login-IP-Host radiusLoginIPHost
replyItem Login-Service radiusLoginService
replyItem Login-TCP-Port radiusLoginTCPPort
replyItem Callback-Number radiusCallbackNumber
replyItem Callback-Id radiusCallbackId
replyItem Framed-IPX-Network radiusFramedIPXNetwork
replyItem Class radiusClass
replyItem Session-Timeout radiusSessionTimeout
replyItem Idle-Timeout radiusIdleTimeout
replyItem Termination-Action radiusTerminationAction
replyItem Login-LAT-Service radiusLoginLATService
replyItem Login-LAT-Node radiusLoginLATNode
replyItem Login-LAT-Group radiusLoginLATGroup
replyItem Framed-AppleTalk-Link radiusFramedAppleTalkLink
replyItem Framed-AppleTalk-Network radiusFramedAppleTalkNetwork
replyItem Framed-AppleTalk-Zone radiusFramedAppleTalkZone
replyItem Port-Limit radiusPortLimit
replyItem Login-LAT-Port radiusLoginLATPort
replyItem Reply-Message radiusReplyMessage

replyItem Tunnel-Type radiusTunnelType
replyItem Tunnel-Medium-Type radiusTunnelMediumType
replyItem Tunnel-Private-Group-Id radiusTunnelPrivateGroupId
```

D.6 users

```
DEFAULT      Auth-Type = ldap
Fall-Through = 1
```

```
DEFAULT Auth-Type = MS-CHAP
```

```
DEFAULT Service-Type == Framed-User
Framed-IP-Address = 255.255.255.254,
Framed-MTU = 576,
Service-Type = Framed-User,
Fall-Through = Yes
```

```
DEFAULT Framed-Protocol == PPP
Framed-Protocol = PPP,
Framed-Compression = Van-Jacobson-TCP-IP
```

```
DEFAULT Hint == "CSLIP"
Framed-Protocol = SLIP,
Framed-Compression = Van-Jacobson-TCP-IP
```

```
DEFAULT Hint == "SLIP"
Framed-Protocol = SLIP
tch, the user is denied access.
```

```
user-name Auth-Type = System, 3Com-User-Access-Level = Administrator
```

```
user-name Auth-Type := Local, User-Password == "password"
```

D.7 clients.conf

```
#
# clients.conf - client configuration directives
#
#####

#####

client 127.0.0.1 {

secret = *****

shortname = localhost

nastype      = other # localhost isn't usually a NAS...
}

client 192.168.0.0/24 {
secret = *****
shortname = UFOP
}

client 192.168.72.0/24 {
secret = *****
shortname = LAB8021x
}

client 192.168.1.0/24 {
secret = *****
shortname = Wirelles
}
client 192.168.0.191/24 {
    secret          = *****
    shortname       = MAP-UFOP
}
}
```

D.8 proxy.conf

```
#
# proxy.conf - proxy radius and realm configuration directives
#
# This file is included by default.  To disable it, you will need
# to modify the PROXY CONFIGURATION section of "radiusd.conf".
#
#####
#
# Proxy server configuration

proxy server {

synchronous = no

retry_delay = 5

retry_count = 3

dead_time = 120

default_fallback = yes

post_proxy_authorize = no

}

#####
#
# Configuration for the proxy realms.
#

realm LOCAL {
type = radius
authhost = LOCAL
accthost = LOCAL
nostrip
}
```

```
realm UFOP {
    type          = radius
    authhost      = LOCAL
    accthost      = LOCAL
}

realm UFOP2 {
    type          = radius
    authhost      = 192.168.0.198:1600
    accthost      = 192.168.0.198:1600
secret          = *****
}

realm host {
    type          = radius
    authhost      = LOCAL
    accthost      = LOCAL
}
```

D.9 sql.conf

```
#
# Configuration for the SQL module, when using MySQL.
#

sql {

driver = "rlm_sql_mysql"
server = "localhost"
login = "radius"
password = "*****"
radius_db = "radius"
acct_table1 = "radacct"
acct_table2 = "radacct"
postauth_table = "radpostauth"
authcheck_table = "radcheck"
authreply_table = "radreply"
groupcheck_table = "radgroupcheck"
groupreply_table = "radgroupreply"
usergroup_table = "usergroup"
nas_table = "nas"
deletestalesessions = yes
sqltrace = yes
sqltracefile = ${logdir}/sqltrace.sql
num_sql_socks = 5
connect_failure_retry_delay = 60

#####
# Query config: Username
#####

sql_user_name = "%{Stripped-User-Name:-%{User-Name:-DEFAULT}}"

#####

authorize_check_query = "SELECT id, UserName, Attribute, Value, op \
FROM ${authcheck_table} \
WHERE Username = '%{SQL-User-Name}' \
```

```
ORDER BY id"

authorize_reply_query = "SELECT id, UserName, Attribute, Value, op \
    FROM ${authreply_table} \
    WHERE Username = '${SQL-User-Name}' \
    ORDER BY id"

authorize_group_check_query = "SELECT ${groupcheck_table}.id,${groupcheck_table}.
GroupName,${groupcheck_table}
.Attribute,${groupcheck_table}.Value,${groupcheck_table}.op FROM ${groupcheck_table}
,${usergroup_table} WHERE ${usergroup_table}.Username = '${SQL-User-Name}'
AND ${usergroup_table}.GroupName = ${groupcheck_table}.GroupName ORDER BY
${groupcheck_table}.id"

authorize_group_reply_query = "SELECT ${groupreply_table}.id,
${groupreply_table}.
GroupName,${groupreply_table}.Attribute,${groupreply_table}.Value,
${groupreply_table}.op FROM ${groupreply_table},${usergroup_table}
WHERE ${usergroup_table}.Username = '${SQL-User-Name}' AND $
{usergroup_table}.GroupName = ${groupreply_table}.GroupName
ORDER BY ${groupreply_table}.id"

#####
accounting_onoff_query = "UPDATE ${acct_table1} SET AcctStopTime='%S',
AcctSessionTime=unix_timestamp('%S') - unix_timestamp(AcctStartTime),
AcctTerminateCause='${Acct-Terminate-Cause}', AcctStopDelay =
'${Acct-Delay-Time}' WHERE AcctSessionTime=0 AND AcctStopTime=0 AND NASIPAddress=
'${NAS-IP-Address}' AND AcctStartTime <= '%S'"

accounting_update_query = "UPDATE ${acct_table1} \
    SET FramedIPAddress = '${Framed-IP-Address}', \
    AcctSessionTime = '${Acct-Session-Time}', \
    AcctInputOctets = '${Acct-Input-Octets}', \
    AcctOutputOctets = '${Acct-Output-Octets}' \
    WHERE AcctSessionId = '${Acct-Session-Id}' \
    AND UserName = '${SQL-User-Name}' \
    AND NASIPAddress= '${NAS-IP-Address}'"

accounting_update_query_alt = "INSERT into ${acct_table1}
```



```
(AcctSessionId, AcctUniqueId, UserName, Realm, NASIPAddress,
NASPortId, NASPortType, AcctStartTime, AcctSessionTime, AcctAuthentic,
ConnectInfo_start, AcctInputOctets, AcctOutputOctets, CalledStationId,
CallingStationId, ServiceType, FramedProtocol, FramedIPAddress,
AcctStartDelay) values('%{Acct-Session-Id}', '%{Acct-Unique-Session-Id}
', '%{SQL-User-Name}', '%{Realm}', '%{NAS-IP-Address}', '
'%{NAS-Port}', '%{NAS-Port-Type}', DATE_SUB('%S',INTERVAL (%{Acct-Session-Time:-0}
+ %{Acct-Delay-Time:-0}) SECOND), '%{Acct-Session-Time}',
'%{Acct-Authentic}', '', '%{Acct-Input-Octets}', '%{Acct-Output-Octets}',
'%{Called-Station-Id}', '%{Calling-Station-Id}', '%{Service-Type}',
'%{Framed-Protocol}', '%{Framed-IP-Address}', '0')
```

```
accounting_start_query = "INSERT into ${acct_table1}
(AcctSessionId, AcctUniqueId, UserName, Realm, NASIPAddress,
NASPortId, NASPortType, AcctStartTime, AcctStopTime, AcctSessionTime,
AcctAuthentic, ConnectInfo_start, ConnectInfo_stop, AcctInputOctets,
AcctOutputOctets, CalledStationId, CallingStationId, AcctTerminateCause,
ServiceType, FramedProtocol, FramedIPAddress, AcctStartDelay, AcctStopDelay)
values('%{Acct-Session-Id}', '%{Acct-Unique-Session-Id}', '%{SQL-User-Name}
', '%{Realm}', '%{NAS-IP-Address}', '%{NAS-Port}', '%{NAS-Port-Type}', '%S',
'0', '0', '%{Acct-Authentic}', '%{Connect-Info}', '', '0', '0',
'%{Called-Station-Id}', '%{Calling-Station-Id}', '', '%{Service-Type}',
'%{Framed-Protocol}', '%{Framed-IP-Address}', '%{Acct-Delay-Time}', '0')
```

```
accounting_start_query_alt = "UPDATE ${acct_table1} SET
AcctStartTime = '%S', AcctStartDelay = '%{Acct-Delay-Time}', ConnectInfo_start =
'%{Connect-Info}' WHERE AcctSessionId = '%{Acct-Session-Id}' AND UserName =
'%{SQL-User-Name}' AND NASIPAddress = '%{NAS-IP-Address}'"
```

```
accounting_stop_query = "UPDATE ${acct_table2} SET AcctStopTime =
'%S', AcctSessionTime = '%{Acct-Session-Time}', AcctInputOctets =
'%{Acct-Input-Octets}', AcctOutputOctets = '%{Acct-Output-Octets}',
AcctTerminateCause = '%{Acct-Terminate-Cause}', AcctStopDelay =
'%{Acct-Delay-Time}', ConnectInfo_stop = '%{Connect-Info}'
WHERE AcctSessionId
= '%{Acct-Session-Id}' AND UserName = '%{SQL-User-Name}'
AND NASIPAddress = '%{NAS-IP-Address}'"
```

```

accounting_stop_query_alt = "INSERT into ${acct_table2}
(AcctSessionId, AcctUniqueId, UserName, Realm, NASIPAddress, NASPortId,
NASPortType, AcctStartTime, AcctStopTime, AcctSessionTime, AcctAuthentic,
ConnectInfo_start, ConnectInfo_stop, AcctInputOctets, AcctOutputOctets,
CalledStationId, CallingStationId, AcctTerminateCause, ServiceType,
FramedProtocol, FramedIPAddress, AcctStartDelay, AcctStopDelay)
  values('${Acct-Session-Id}', '${Acct-Unique-Session-Id}',
'${SQL-User-Name}', '${Realm}', '${NAS-IP-Address}', '${NAS-Port}',
'${NAS-Port-Type}', DATE_SUB('%S', INTERVAL (${Acct-Session-Time:-0}
+ ${Acct-Delay-Time:-0}) SECOND), '%S', '${Acct-Session-Time}',
'${Acct-Authentic}', '', '${Connect-Info}', '${Acct-Input-Octets}',
'${Acct-Output-Octets}', '${Called-Station-Id}', '${Calling-Station-Id}',
'${Acct-Terminate-Cause}', '${Service-Type}', '${Framed-Protocol}',
'${Framed-IP-Address}', '0', '${Acct-Delay-Time}')"

group_membership_query = "SELECT GroupName
FROM ${usergroup_table} WHERE UserName='${SQL-User-Name}'"
#####
# Authentication Logging Queries
#####
# postauth_query - Insert some info after authentication
#####

postauth_query = "INSERT into ${postauth_table} (id, user, pass, reply, date)
values ('', '${User-Name}', '${User-Password:-Chap-Password}',
'${reply:Packet-Type}', NOW())"
}

```

D.10 eap.conf

```
#
# Whatever you do, do NOT set 'Auth-Type := EAP'. The server
# is smart enough to figure this out on its own. The most
# common side effect of setting 'Auth-Type := EAP' is that the
# users then cannot use ANY other authentication method.
#
# $Id: eap.conf,v 1.4.4.3 2006/04/28 18:25:03 aland Exp $
#
eap {

    timer_expire      = 60

    ignore_unknown_eap_types = no

    cisco_accounting_username_bug = no

    md5 {
    }

    leap {
    }

    gtc {

        auth_type = PAP
    }

    tls {
        private_key_password = whatever
        private_key_file = ${raddbdir}/certs/cert-srv.pem

        certificate_file = ${raddbdir}/certs/cert-srv.pem

        # Trusted Root CA list
        CA_file = ${raddbdir}/certs/demoCA/cacert.pem
    }
}
```

```
dh_file = ${raddbdir}/certs/dh
random_file = ${raddbdir}/certs/random

fragment_size = 1024

include_length = yes

check_crl = no

}

ttls {

default_eap_type = md5

copy_request_to_tunnel = no

use_tunneled_reply = no
}

peap {

default_eap_type = mschapv2

}

mschapv2 {
}
}
```

Referências Bibliográficas

- ACS (2010). Cisco secure access control server. <http://www.cisco.com/en/US/products/sw/secursw/ps2086/index.html> .Acessado em: 08/11/2010.
- Balbonil, M. (2006). Nic.br anuncia resultados da pesquisa sobre o uso da internet no brasil. Technical report, Núcleo de Informação e Coordenação (NIC.br).
- Barrosi, L. G. e Foltran, C. (2008). Autenticação iee 802.1x em redes de computadores utilizando tls e eap. *4º Encontro de Engenharia e Tecnologia dos Campos Gerais*.
- Blunk, L. e Vollbrecht, J. (1998). Rfc 2284 - ppp extensible authentication protocol (eap). Technical report.
- Campos, A. (2010). Revista do linux - integração de rede com diretórios ldap. <http://augustocampos.net/revista-do-linux/025/rede.html> - Acesso em: 16 Nov 2010.
- Cardozo, H. A. M. (2007). Análise do centos 5. <http://ha-mc.org/?q=node/1> - Acessado em 20 Nov. 2010.
- da Costa, P. H. A. (2010). *Samba: Windows e Linux em rede*. Linux New Media do Brasil Editora Ltda.
- et al, C. R. (2000). Rfc 2865 - remote authentication dial in user service (radius). Technical report.
- et al, T. C. (2009). Projeto network login. Technical report, NTI - Núcleo de Tecnologia da Informação da UFOP.
- FreeRadius (2010). Freeradius. <http://www.freeradius.org/> .Acessado em: 08/11/2010.
- Fucapi (2010). Segurança da informação - cartilha de orientação - integriadde, confidencialidade e disponibilidade. https://portal.fucapi.br/download/cartilha_si.pdf - Acessado em 19 Nov. 2010.
- Hassell, J. (2002). *Radius*. O'Reilly, 1ª edição edição.

- IAS (2010). Microsoft internet authentication service. <http://technet.microsoft.com/en-us/network/bb643123.aspx> .Acessado em: 08/11/2010.
- IEEE (2001). 802.1x - port based network access control. <http://www.ieee802.org/1/pages/802.1x.html>, Acessado em: 13/11/2010.
- Isquierdo, G. S. (2001). Integração do serviço de diretório ldap com o serviço de nomes corba. *Dissertação Apresentada ao Instituto de Matemática e Estatística da Universidade de São Paulo*.
- Pinheiro, J. M. (2009). Auditoria e análise de segurança da informação - aula 06 - segurança e confiabilidade. http://www.projetoderedes.com.br/aulas/aulas_ugb_auditoria_e_analise.php - Acessado em 19 Nov. 2010.
- RUFINO, N. M. D. O. (2007). *Segurança Em Redes Sem Fio - 2ª Edição*. NOVATEC.
- Simpson, W. (1994). Rfc 1661 - the point-to-point protocol (ppp). Technical report.
- Simpson, W. (1996). Rfc 1994 - ppp challenge handshake authentication protocol (chap). Technical report.
- VOLLBRECHT, J. (2000). Aaa authorization framework. RFC 2904, Internet Engineering Task Force.