FACULDADE DE TECNOLOGIA SENAI DEDESENVOLVIMENTO GERENCIAL – FATESG

CURSO SUPERIOR DE TENCNOLOGIA EM REDE DE COMPUTADORES

Fabiano Santos Florentino

IMPLEMENTAÇÃO DE SEGURANÇA EM REDES SEM FIO UTILIZANDO FreeRADIUS COM OS PROTOCOLOS RADIUS E 802.1X

PROFESSOR-ORIENTADOR: Me. RAFAEL LEAL

GOIÂNIA

2011

Fabiano Santos Florentino

IMPLEMENTAÇÃO DE SEGURANÇA EM REDES SEM FIO UTILIZANDO FREERADIUS COM OS PROTOCOLOS RADIUS E 802.1X

Trabalho de conclusão de curso apresentado à Faculdade de Tecnologia SENAI de desenvolvimento Gerencial – FATESG, para obtenção do título de Graduado em Tecnologia em Rede de Computadores.
Orientador: Prof. Me. Rafael Leal

GOIÂNIA
2011

Fabiano Santos Florentino

IMPLEMENTAÇÃO DE SEGURANÇA EM REDES SEM FIO UTILIZANDO FREERADIUS COM OS PROTOCOLOS RADIUS E 802.1X

Trabalho de conclusão de curso apresentado à Faculdade de Tecnologia SENAI de desenvolvimento Gerencial – FATESG, para obtenção do título de Graduado em Tecnologia em Rede de Computadores.

Aprovado em _____de_____de 20____.

Banca Examinadora

_____

Prof. Orientador MSc. Rafael Leal

_____

Prof. MSc. Ricardo de Andrade Kratz

_____

Prof. MSc. Weysller Matuzinhos de Moura

LISTA DE SIGLAS E ABREVIATURAS

3DES – *Triple Data Encryption Standard*
AS – *Authentication Server*
AES – *Advanced Encryption Standard*
AP – *Access Point*

BSS - *Basic Service Set*
CHAP – *Challenge-handshake authentication protocol*
DES – *Data Encryption Standard*
DSSS – *Direct Sequence Spread Spectrum*
EAP – *Extensible Authentication Protocol*
EAPOL - *(EAP Over LAN) Logoff*
IEEE – *Institute of Eletrical and Electronics Engineers*
IETF - *Internet Engineering Task Force*
MAC - *Media Access Control address*
MD5 – *Message Digest Algorithm 5*
MS-CHAP – *Microsoft Challenge-handshake authentication protocol*
MS-CHAPv2 – *Microsoft Challenge-handshake authentication protocolVersio 2*
PEAP - *Protected Extensible Authentication Protocol*
PAP – *Protocol Authentication of Password*
PSK – *Pre-Shared Key*
RADIUS – *Remote Authentication Dial In User Service*
SHA-1 – *Secure Hash Algorithm 1*
TKIP – *Temporal Key Integrity Protocol*
TLS – *Transport Layer Security*
TTLS – *Tunneled Transport Layer Security*
WiFi – *Wireless Fidelity*
WEP – *Wired Equivalent Privacy*
WAP – *WiFi Protected Access*

LISTA DE ILUSTRAÇÕES

LISTA DE TABELAS

RESUMO

O presente trabalho tem o intuito de demonstrar o funcionamento dos protocolos de autenticação 802.1x e RADIUS bem como seus conceitos e estudo dos protocolos, para isso também será implementado um servidor com o serviço freeRadius, mostrando o funcionamento dos protocolos em uma rede local e wifi; Será também mostrado uma das formas de implementar o servidor utilizando os meios que o mesmo oferece, neste caso será demonstrado o servidor utilizando um banco de dados como meio de autenticação para usuários da rede.

Palavra-chave: Autenticação. Servidor. Protocolo.

SUMÁRIO

1. INTRODUÇÃO

Este trabalho tem por objetivo falar sobre segurança em redes sem fio e local por meio de um servidor de autenticação, servidor freeRADIUS. Este servidor, por sua vez, utiliza uma combinação de protocolos de autenticação, o 802.1x e o *Remote Authentication Dial In User Service* (RADIUS). Esses protocolos trabalham distintamente, um provendo a segurança entre os equipamentos que estão sendo utilizado na rede, o outro para a segurança dos usurários da rede local, tanto rede sem fio quando rede física. Essa combinação dos protocolos é feita através de um servidor, no caso, o freeRAIDIUS, que gerencia, verifica e contabiliza todo o acesso à rede, seja ela sem fio ou com fio. Esses equipamentos precisam possuir condições mínimas de configuração para atender os requisitos do servidor, por exemplo, ter suporte para os protocolos envolvidos.

802.1x é um padrão IEEE para controle de acesso à rede com base em portas; faz parte do grupo IEEE 802.1 de protocolos de redes de computadores. Provê um mecanismo de autenticação para dispositivos que desejam juntar-se a uma porta na *Local Area Network* (LAN) seja estabelecendo uma conexão ponto-a-ponto ou prevenindo acesso para esta porta se a autenticação falhar. É usado para a maioria dos *Access Points* sem fio 802.11 e é baseado no Protocolo de Autenticação Extensiva (EAP).

O RADIUS é um protocolo que visa à autenticação, autorização e gestão de utilizadores, para acesso à rede ou serviços de rede (conceito AAA). RADIUS é normalmente usado para gerir e tornar mais seguro o acesso à Internet ou às redes internas. O Protocolo RADIUS baseia-se num sistema cliente/servidor. O Servidor de RADIUS utiliza o conceito AAA para gerir o acesso à rede. Este conceito refere-se aos processos de autenticação e autorização e contabilização (*Accounting*) que são utilizados para estabelecer uma ligação à Internet ou utilizar aplicações de acesso à rede.

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 utilizadores na Internet. Os restantes utilizadores encontram-se divididos entre os restantes servidores, destacando-se entre eles o Cisco *Access Control Server* (ACS) e o Microsoft *Internet Authentication Service* (IAS).

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. Devido a estas características e tendo em conta o facto de ser uma aplicação *open source*, esta será a implementação de RADIUS utilizada para o desenvolvimento do trabalho.

1.1. OBJETIVOS GERAIS

Como objetivo geral este trabalho, demonstraremos uma implementação de servidor RADIUS, utilizando o freeRADIUS como solução. Esta solução provê um meio de autenticação com usuário e senha que serão previamente cadastrados em um banco de dados.

O banco utilizado nesta implementação será o MySQL, banco de dados grátis e muito robusto para esse tipo de solução. Será demonstrado também o uso dos protocolos de rede 802.1x e RADIUS, protocolos utilizados no servidor, para prover o meio seguro entre os equipamentos e os usuários tanto da rede sem fio quanto da rede local. Também criaremos um ambiente de rede seguro, o acesso só será permitido ao usuário por meio de um usuário e senha cadastrado no servidor freeRADIUS; autenticado o usuário em fim terá acesso aos recursos de rede disponíveis.

1.2. OBJETIVOS ESPECIFICOS

Demonstraremos os protocolos 802.1x e RADIUS junto com a solução de servidor freeRADIUS em um ambiente de rede, controlando todo o acesso feito através desse servidor. O acesso à rede será através de usuário e senha previamente cadastrados no servidor RADIUS, para controlar melhor os meios de acesso a rede, seja ela por cabo ou *wireless*.

1.3. MOTIVAÇÃO

O que nos motiva para desenvolver este trabalho é aprofundar os estudos na área de tecnologia em redes, procurando conhecer melhor as soluções de segurança existentes no mercado, proporcionando-nos um conhecimento melhor do assunto e compartilhando o mesmo.

1.4. METODOLOGIA

Realizaremos um estudo dos protocolos 802.1x, RADIUS e do processo de implementação do servidor de freeRADIUS, através da configuração dos switch para que se possa trabalhar com esses protocolos de forma que gerem uma segurança, visando a autenticação por meio de usuário e senha de acesso. Será feito o estudo analítico do conteúdo, procurando conhecer seus métodos de implementação.

1.5. ESTRUTURA DO TRABALHO

O trabalho será desenvolvido com base no estudo da implementação do serviço freeRADIUS, estudando as formas de se configurar o servidor RADIUS.

Com esse estudo e sua implementação em mãos, desenvolvemos o trabalho aqui presente explicando os conceitos do mesmo bem como algumas soluções existentes junto ao servidor com o protocolo de rede 802.1x e RADIUS, que serão fundamentais para toda a implementação e desenvolvimento do trabalho.

2. ***INSTITUTE OF ELECTRICAL AND ELECTRONICS ENGINEERS*** **(IEEE 802.11i)**

O padrão (IEEE) 802.11i será o substituto do protocolo *Wired Equivalent Privacy* (WEP). O novo padrão estava em discussão no *Task Group* i (Tgi) e tinha previsão de ser finalizado no final de 2003. Seu objetivo era resolver os diversos problemas encontrados no protocolo WEP, ligados a garantia da confidencialidade e integridade da comunicação, como apresentados anteriormente.

Desde quando o grupo de trabalho do IEEE 802.11i iniciou os seus estudos, alguns fabricantes têm implementado algumas pré-normas no mercado para prevenir alguns tipos de vulnerabilidade, como:

- Ataque de dicionário ao *Extensible Authentication Protocol* (EAP) (**) O *frame* 802.11 é facilmente capturado, possibilitando que um intruso descubra uma senha usando o mecanismo de força bruta baseado em dicionário. É recomendado que seja utilizado métodos de autenticação como: (**) EAP; *Transport Layer Security* (TLS); *Spatial Reuse Protocol* (SRP); *Tata Teleservices Limited* (TTLS); e *Protected Extensible Authentication Protocol* (PEAP);
- Ataque a chave *default*: Como 802.1, não implementa um mecanismo de troca de chaves aleatório. Como isso, descobrir a chave é questão de tempo. É extremamente recomendado que se use algum mecanismo de troca dinâmica de chaves como SNMPv3 ou SSH;
- Ataque de (**)DOS (*Disk Operation System*) baseado no *frame* EAPOL (*EAP Over LAN*)-*logoff*: Como esse tipo de *frame* não é autenticado, alguém pode enviar um *frame* EAPOL *logoff* e desconectar um usuário. Pode-se filtrar esse tipo de solicitação no ponto de acesso (AP);
- Ataque de DOS baseado no *frame* EAPOL-*Start*: O atacante pode fazer um envio maciço de *frames* EAPOL start para sobrecarregar o ponto de acesso (AP) e tirá-lo de serviço. Isso pode ser evitado fazendo com que o AP não gaste muito recurso com o atendimento desse tipo de *frame*;
- Ataque de DOS baseado no espaço de identificação do EAP: O atacante e no ponto de acesso for a de serviço;
- Ataque de DOS baseado no envio antecipado do pacote de sucesso do EAP: O atacante pode enviar um pacote de sucesso do EAP antecipado para permitir que uma estação possa vista na rede antes que o ponto de acesso complete o processo de autenticação;
- Ataque de DOS baseado no pacote de falha do EAP: O atacante pode enviar um pacote de falha do EAP antecipado para não permitir que uma

estação seja vista na rede antes que o ponto de acesso complete o processo de autenticação;

- Ataque de DOS baseado na alteração do pacote EAP: O atacante pode modificar o conteúdo do pacote EAP. Para evitar esse tipo de ataque, se deve utilizar protocolos de criptografia como *Transport Layer Security* (TLS), *Protected Extensible Authentication Protocol* (PEAP) ou *Tunneled Transport Layer Security* (TTLS).

2.1. SEGURANÇA NO IEEE 802.11

Uma rede 802.11, conforme figura 1, a seguir, é formada por uma ou mais (**)BSS (*Basic Service Set*). Um BSS é formado por um conjunto de estações que utilizam o mesmo *Media Access Control address* (MAC) (**) e compartilham a mesma área física de transmissão, por exemplo, estações *wireless* em uma mesma sala utilizando o mesmo protocolo de transmissão formam um BSS. Um BSS conectado a outros BSSs forma um sistema de distribuição *Distribution System* (DS). A conexão do BSS ao sistema de distribuição é feita por um ponto de acesso *Access Point* (AP).

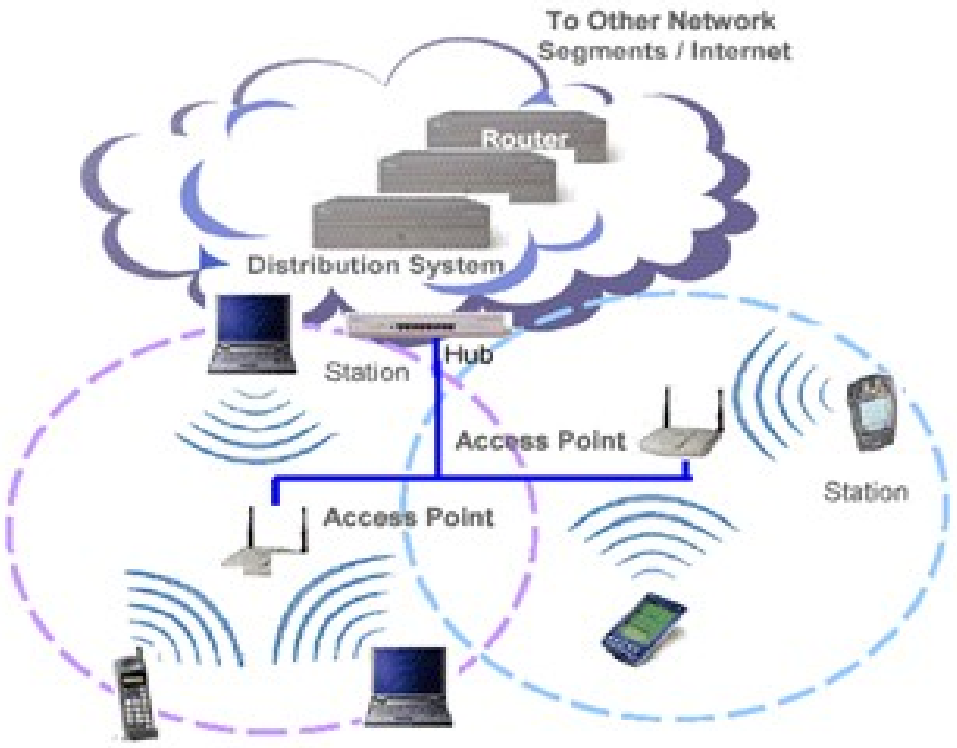

FIGURA 1 – Funcionamento de uma rede 802.11
Fonte: www.google.com.br (2011)

3. **IEEE 802.1X**

IEEE 802.1Xé um padrão IEEE para controle de acesso à rede com base em portas; faz parte do grupo IEEE 802.1 de protocolos de redes de computadores. Ele provê um mecanismo de autenticação para dispositivos que desejam juntar-se a uma porta na *Local Area Network* (LAN) , seja estabelecendo uma conexão ponto-a-ponto ou prevenindo acesso para esta porta se a autenticação falhar. É usado para a maioria dos *Access points* sem fio 802.11 e é baseado no Protocolo de Autenticação Extensiva (EAP). (Wikipédia 2011)

O padrão IEEE 802.1X foi concebido para oferecer autenticação, controle de acesso e distribuição de chaves criptográficas em redes locais com e sem fio. É importante destacar que o 802.1X não esteja ligado apenas ao padrão IEEE 802.11, mas a todos os padrões de redes locais e metropolitanas patrocinados pelo IEEE 802. Além disso, o padrão pode ser utilizado em conjunto com diversos protocolos de autenticação localizados nas camadas superiores. O grupo de trabalho 802.11i

definiu que o mecanismo de autenticação a ser utilizado no IEEE 802.11 deverá seguir o modelo do IEEE 802.1X. (Maia 2003)

3.1. VISÃO GERAL

A autenticação no IEEE 802.1X é realizada nos dois sentidos *mutual autentication* e utiliza o esquema de *challenge-response*. Existem três componentes a serem considerados no padrão: autenticador, *supplicant* e servidor de autenticação. Em uma rede sem fio, o autenticador é geralmente o AP e o *supplicant* uma estação que deseja conectar-se ao AP. O servidor de autenticação é o verdadeiro responsável por autenticar o *supplicant*, com base nas informações oferecidas por ele. O padrão não especifica qual servidor de autenticação deve ser utilizado, podendo ser, por exemplo, um servidor RADIUS. Na verdade, o servidor de autenticação não precisa ser necessariamente um elemento externo, podendo fazer parte do próprio AP. (Maia 2003)

Um nó *wireless* precisa autenticar-se antes de poder ter acesso aos recursos da LAN. 802.1X provêautenticação baseada em portas, que envolve comunicação entre o requisitante, o autenticador e o servidor deautenticação. O requisitante é comumente o *software* em um dispositivo cliente. Como um laptop, o autenticador é um Switch Ethernet ou AP sem fio, e a autenticação geralmente uma base de dados RADIUS. O autenticador atua como uma proteção secundária à rede. Não é permitido ao requisitante, por exemplo, dispositivo cliente, acesso através do autenticador no lado protegido da rede até que a identidade dorequisitante seja autorizada. Uma analogia a isso é prover um passaporte válido em um aeroporto antes de ser permitida a passagem pela segurança até o terminal. Com a autenticação baseada em portas 802.1X, o requisitante provê credenciais como nome de usuário / senha ou certificado digital, ao autenticador, e ele encaminha as credenciais até o servidor de autenticação para verificação. Se as credenciais são válidas (na base de dados do servidor de autenticação), o requisitante (dispositivo cliente) é permitido acessar os recursos localizados no lado protegido da rede. Sob detecção do novo cliente (requisitante), a porta na swtich (autenticador) é habilitada e mudada para oestado “não-autorizado”. Neste estado, apenas tráfego 802.1x é

permitido; outros tráfegos, como DHCP e HTTP, são bloqueados na camada de enlace. (Maia 2003)

O IEEE 802.1X implementa o protocolo EAP em conjunto com o EAP *over* LAN (EAPOL). O EAP permite o encapsulamento de diversos protocolos de autenticação oferecidos nas camadas superiores e, consequentemente, não definidos pelo 802.1X. Estes protocolos oferecem diferentes métodos de autenticação, como Kerberos, senhas, certificados digitais e chaves públicas. Desta forma, é possível que o usuário também seja autenticado e não apenas a estação."

O autenticador envia a identidade de autenticação 'EAP-request' ao requisitante, que, por sua vez, responde com o pacote 'EAP-response', que o autenticador encaminha ao servidor de autenticação." "Se o servidor de autenticação aceitar a requisição, o autenticador muda o estado da porta para o modo "autorizado" e o tráfego normal é autorizado. Quando o requisitante efetua um *logoff*, envia uma mensagem 'EAP-logoff' para o autenticador. (Maia 2003)

O autenticador, então, muda sua porta para o estado "não-autorizado", bloqueando novamente todo o tráfego não-EAP. Wikipédia (2011)

3.2. PROTOCOLOS DE AUTENTICAÇÃO

Os protocolos de autenticação mais comuns oferecidos pelas camadas superiores são o EAP-TLS, PEAP, EAP-TTSL e LEAP. É importante perceber, que o AP serve apenas como um meio para que as mensagens cheguem ao servidor de autenticação. Sendo assim, pode-se especificar qualquer mecanismo de autenticação sem a necessidade de alterar-se o AP. A Tabela 1, a seguir, apresenta algumas das características dos diversos protocolos de autenticação disponíveis. (Maia 2003)

Tabela 1 – Tabela de Protocolos de Autenticação.

Tipo EAP	Aberto / Proprietário	Autenticação Mutua	Credênciais de Autenticação		Chave Material	Usuário/Nome	RFC
			Suplicante	Autenticador			
MD5	Aberto	não	Usuário/Senha	Nada	não	sim	1321
TLS	Aberto	sim	Certificado	Certificado	sim	sim	2716
TTLS	Aberto	sim	Usuário/Senha	Certificado	sim	não	IETF Draft
PEAP	Aberto	sim	Usuário/Senha	Certificado	sim	não	IETF Draft
SIM	Aberto/GSM	sim	SIM		sim		IETF Draft
AKA	Aberto/UMTS	sim	USIM		sim		IETF Draft
SKE	Open/CDMA	sim			sim		IETF Draft
LEAP	Proprietário	sim	Usuário/Senha		sim	sim	NA

Fonte: Maia (2003)

O EAP-TLS foi proposto pela Microsoft e hoje é um padrão Internet. O TLS é baseado no *Secure Sockets Layer* (SSL) versão 3.0 e oferece autenticação mútua, utilizando certificados digitais e permite geração de chaves criptográficas. Os certificados digitais devem ser configurados individualmente em cada cliente da rede e no servidor de autenticação. O protocolo é suportado por *default* no MS-Windows XP e pode ser configurado nas demais versões do Windows. Já existem também versões do EAP-TLS para estações Linux e FreeBSD.

O *Protected Extensible Authentication Protocol* (PEAP) ainda está em processo de aceitação no *Internet EngineeringTask Force* (IETF). O *Protect* EAP oferece autenticação baseada em senha e exige que o servidor de autenticação possua um certificado digital, porém não exige certificados nos clientes. O protocolo foi adotado pela Microsoft no Windows XP e Windows Server 2003.

O *Tunnuled Transport Layer Security* (EAP-TTLS) também está em fase de aceitação pelo IETF. EAP-TTLS é uma extensão do EAP-TLS, pois utiliza a conexão segura TLS para trocar informações adicionais entre cliente e servidor. O EAP-TLS oferece autenticação mútua e unidirecional, na qual apenas o servidor é autenticado.

O *Lightweight Extensible Authentication Protocol* (LEAP) foi desenvolvido pela Cisco Systems e foi um dos primeiros protocolos de autenticação disponível para redes sem fio. O LEAP oferece diversas vantagens, como autenticação mútua, autenticação de usuário por senha e chaves dinâmicas. (Maia 2003)

3.3. **IMPLEMENTAÇÕES**

3.3.1. **Access Point sem fio**

Vendedores de *Access Points* (AP) Wi-Fi agora usam 802.11i que implementa 802.1X para AP *wireless* para corrigir as vulnerabilidades de segurança encontradas em WEP. O papel do autenticador é realizado tanto pelo AP em si via chave pré-compartilhada (referida também como WPA2-PSK) ou para empresas maiores, por identidade terceira, como um servidor RADIUS. Ele provê autenticação apenas para o cliente ou, mais apropriadamente, autenticação forte e mútua utilizando protocolos como EAP-TLS. Wikipédia (2011)

3.3.2. Softwares

Windows XP e Windows Vista suportam 802.1X para todas as conexões de rede por padrão. Windows 2000possui suporte no último *service Pack*. Windows Mobile 2003 e sistemas operacionais mais atuais também vêm com *client* nativo 802.1x. Windows XP possui maiores questões com mudança de endereço IP (VLANDinâmica) como resultado de uma validação de usuário 802.1X e a Microsoft não modificará estacaracterística que evitará estes problemas - um projeto para Linux conhecido como Open1X produz um cliente *Open Source*, Xsupplicant. O mais geralrequisitante WPA pode ser usado para conexões para redes com e sem fio 802.11. Ambos suportam um *range*bastante abrangente de tipos de EAP.MAC OS X oferece um suporte nativo desde 10.3. O iPhone e o iPod Touch suportam 802.1X, assim como olançamento do iPhone OS 2.0.

4. ***REMOTE AUTHENTICATION DIAL IN USER SERVICE* (RADIUS)**

O RADIUS é um protocolo amplamente utilizado para gerenciar o acesso dos mais diversos serviços de rede. Este protocolo define um padrão para troca de informações entre um servidor de acesso à rede NAS (*Network Access Server*) e um servidor AAA para realizar a autenticação, a autorização e as operações de gerenciamento de contas. Um servidor RADIUS AAA pode gerenciar, de forma eficiente, diferentes perfis de usuários para a autenticação dos mesmos, além de fornecer informações de configurações que especificam o tipo de serviço a ser entregue e as políticas de cada tipo de serviço, para garantir o uso apropriado de cada recurso disponível. Dentre os serviços que utilizam o RADIUS, podemos citar

as autenticações em redes sem-fio, conexões DSL e VPNs. Existem soluções pagas que implementam o RADIUS, mas também existem soluções de código aberto de qualidade como o FreeRadius, que conta com uma crescente base de usuários.

O RADIUS apresenta uma série de funcionalidades que o qualifica como um eficiente sistema de autenticação adaptável as mais diversas condições de rede. Serão descritas, a seguir, as principais vantagens. (CARVALHO, 2008)

4.1. MODELO CLIENTE/SERVIDOR

O RADIUS utiliza o modelo cliente/servidor. O NAS funciona como um cliente para o servidor RADIUS. O cliente é responsável por enviar as informações dos usuários que desejam acessar o serviço do NAS para o servidor RADIUS, que se encarregará de verificar a autenticidade do usuário e informar a sua validade para o NAS, que poderá retornar então a resposta adequada para o usuário. Desta forma, o NAS repassa a tarefa de autenticação para o servidor RADIUS, que retorna para o NAS informações fundamentais para controlar o uso de um determinado recurso por parte do usuário como, por exemplo, quais são os limites de acesso do usuário e qual é o tempo máximo de conexão antes de a mesma expirar. (CARVALHO, 2008)

4.2. SEGURANÇA

As transferências de dados realizadas entre o cliente e o servidor RADIUS são autenticadas através do uso de um segredo compartilhado (*shared secret*), que nunca é enviado pela rede. Este segredo é de prévio conhecimento tanto do cliente quanto do servidor, e é utilizado para garantir a autenticidade do usuário de um determinado serviço requisitado. As senhas de usuário são criptografadas para tentar garantir que nenhum usuário malicioso que esteja ouvindo a rede possa descobrir a senha do usuário. Além disso, outros métodos de autenticação podem

ser implementados, dependendo do grau de segurança requisitado pelo sistema. (CARVALHO, 2008)

4.3. **FLEXIBILIDADE E ADAPTABILIDADE**

Os dispositivos de rede como roteadores, servidores, e switches, muitas vezes não conseguem arcar com um grande número de usuários com informações de autenticação distintas. Através do RADIUS, estes dispositivos podem romper esta barreira e permitir a autenticação destes usuários através do uso de servidores RADIUS embarcados atuando como Proxies para servidores RADIUS de maior capacidade de processamento. (CARVALHO, 2008)

4.4. **PROTOCOLO EXTENSÍVEL**

Ao utilizar um campo de atributos de tamanho variável em seus pacotes, o protocolo RADIUS permite que novos atributos sejam adicionados sem atrapalhar implementações prévias do protocolo. Através do campo atributos, também é possível estabelecer novos parâmetros e novos mecanismos de autenticação, sem necessariamente ter que alterar o formato do pacote. (CARVALHO, 2008)

4.5. COMPATIBILIDADE

Os servidores RADIUS podem verificar as credenciais de seus usuários em bancos de dados de fontes externas, como bancos de dados SQL, Kerberos e LDAP. Desta forma, a implementação de um servidor RADIUS pode ser realizada de forma a reaproveitar um banco de usuários já existente. Outro ponto interessante é que o RADIUS é amplamente utilizado, e praticamente todos os fabricantes de *hardware* produzem produtos compatíveis com o serviço. (CARVALHO, 2008)

4.6. *AUTHENTICATION, AUTHORIZATION, ACCOUNTING* (AAA)

O servidor RADIUS é um servidor AAA. Para poder ser considerado como tal, ele precisa ser capaz de autenticar usuários, lidar efetivamente com as requisições de autorização e prover a coleta de informações dos usuários (auditoria). (CARVALHO, 2008)

4.7. **AUTENTICAÇÃO (AUTHENTICATION)**

O servidor RADIUS é um servidor AAA. Para poder ser considerado como tal, ele precisa ser capaz de autenticar usuários, lidar efetivamente com as requisições de autorização e prover a coleta de informações dos usuários (auditoria). (CARVALHO, 2008)

4.8. AUTORIZAÇÃO (AUTHORIZATION)

A autorização se refere à associação de certos tipos de privilégios para uma entidade, baseados na própria autenticação da entidade e de quais serviços estão sendo requisitados. Dentre as políticas de autorização, podemos utilizar restrições em determinados horários, restrições de acordo com o grupo ao qual pertence o usuário e proteção contra múltiplas conexões simultâneas efetuadas pelo mesmo usuário como, por exemplo, de aplicações que utilizam estas políticas de autorização, podemos citar as políticas de Qualidade de Serviço, que podem fornecer mais banda de acordo com o serviço requisitado, o controle de certos tipos de pacotes, como ocorre no *traffic shapping*, dentre outros. (CARVALHO, 2008)

4.9. AUDITORIA (ACCOUNTING)

Accounting se refere ao monitoramento do comportamento dos usuários e de que forma estes consomem os recursos da rede. Estas informações podem ser muito úteis para melhor gerenciar os recursos de rede, para a cobrança de serviços e para o planejamento de quais setores da rede precisam ser melhorados. (CARVALHO, 2008)

4.10. **TOPOLOGIA E FUNCIONAMENTO**

O protocolo RADIUS segue a arquitetura servidor/cliente. O usuário que deseja utilizar um determinado serviço de rede envia as suas informações para o NAS solicitado (o NAS atua como um cliente para o servidor RADIUS), que pode solicitar a autenticação deste usuário a um servidor RADIUS AAA, na forma de uma mensagem de requisição de acesso (*Access- Request message*). De acordo com a resposta fornecida pelo servidor AAA, o cliente (NAS) pode então fornecer os serviços requisitados pelo usuário de acordo com as políticas e informações estabelecidas pelo servidor RADIUS AAA. Após receber uma requisição do cliente, o servidor RADIUS tenta promover a autenticação do usuário, e retorna as informações de configuração e as políticas a serem aplicadas para o mesmo.

Devido à grande flexibilidade do protocolo e devido às diferentes tecnologias agregadas ao RADIUS, conforme a figura 2, a seguir, o servidor pode ser configurado para autenticar os usuários localmente ou como um cliente Proxy que redireciona os pedidos de acesso para outro servidor AAA remoto. Quando utilizamos o servidor RADIUS desta forma, o servidor AAA que atua como Proxy passa a ser o responsável pela intermediação das mensagens trocadas entre o cliente e o servidor remoto. Um servidor RADIUS pode ser configurado para efetuar determinadas requisições localmente e atuar como Proxy para outros servidores remotos. Um exemplo muito prático e útil desta flexibilidade do RADIUS é a utilização do mesmo para a autenticação em serviços que executam em sistemas embarcados. Como os sistemas embarcados geralmente possuem limitações de gasto de energia e de espaço de armazenamento e memória, a utilização de um servidor AAA *embarcado* atuando somente como Proxy pode garantir a autenticação

segura de usuários de um serviço, como o acesso de uma rede sem-fio em um ponto de acesso.

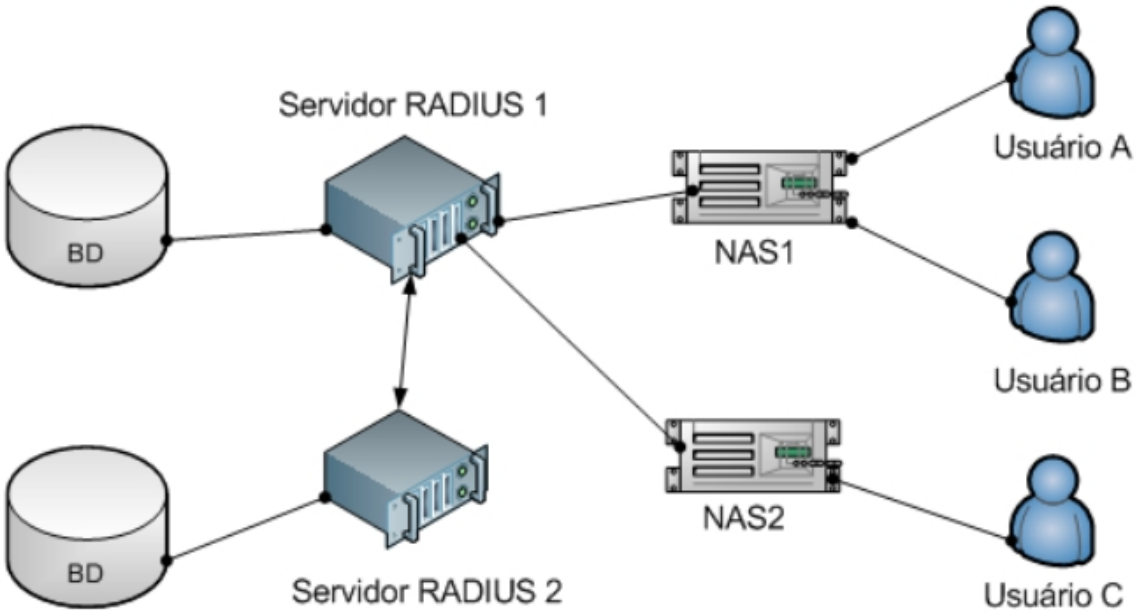

FIGURA 2 – Topologia e Funcionamento
Fonte: http://www.gta.ufrj.br/grad/08_1/radius/TopologiaeFuncionamento.html

Apresentamos um exemplo de uma configuração que utiliza o RADIUS na figura 2. A suponha que os serviços do NAS1 são autenticados pelo servidor RADIUS 1, enquanto os serviços do NAS 2 são autenticados pelo servidor RADIUS 2. Podemos observar um exemplo típico de um sistema utilizando RADIUS. O Usuário A, por exemplo, deseja utilizar um serviço fornecido pelo NAS1. Após o usuário A enviar um pedido para o NAS1, o NAS1 que atua como um cliente RADIUS, envia para o Servidor RADIUS 1 a requisição de acesso, indicando que o usuário A deseja utilizar o serviço fornecido pelo NAS 1. Neste exemplo, o servidor RADIUS 1 conseguiu autenticar o Usuário A com sucesso. O servidor RADIUS então informa ao NAS1 da autenticidade do usuário A, que pode agora utilizar o serviço por ele requisitado. Ainda na figura 2, o usuário C deseja utilizar um serviço fornecido pelo NAS 2. Para este caso, o servidor RADIUS 1 funciona apenas como

um *Proxy* para o servidor RADIUS 2, de forma que a intermediação do processo de autenticação entre o servidor RADIUS 2 e o NAS 2 será feita pelo servidor RADIUS 1. Neste exemplo, podemos verificar o grande grau de flexibilidade do Servidor RADIUS, que pode autenticar determinados serviços localmente e ao mesmo tempo intermediar a autenticação de outros serviços de autenticação remotos.

4.11. **ESTABELECENDO UMA SESSÃO**

O estabelecimento de uma sessão RADIUS ocorre com uma série de trocas de mensagens que objetivam fornecer um determinado serviço de rede para um usuário. Quando um NAS (cliente) é configurado para utilizar o RADIUS, quaisquer usuários que desejem acessar um serviço deste NAS precisam apresentar suas credenciais de autenticação para o NAS, por exemplo, através de uma tela de início de sessão, na qual o usuário pode inserir o seu nome e a sua senha. Após receber estas informações do usuário, o NAS pode autenticá-lo usando o RADIUS. Para realizar esta autenticação, o NAS, que atua como cliente do servidor RADIUS, envia um pacote do tipo Requisição de Acesso, que contém atributos como o nome do usuário, a sua senha, o seu número de identificação, dentre outros, conforme a figura 3, a seguir. As senhas de usuários são ocultadas através da utilização de um método baseado no algoritmo “RSA MessageDigest5” (MD5).

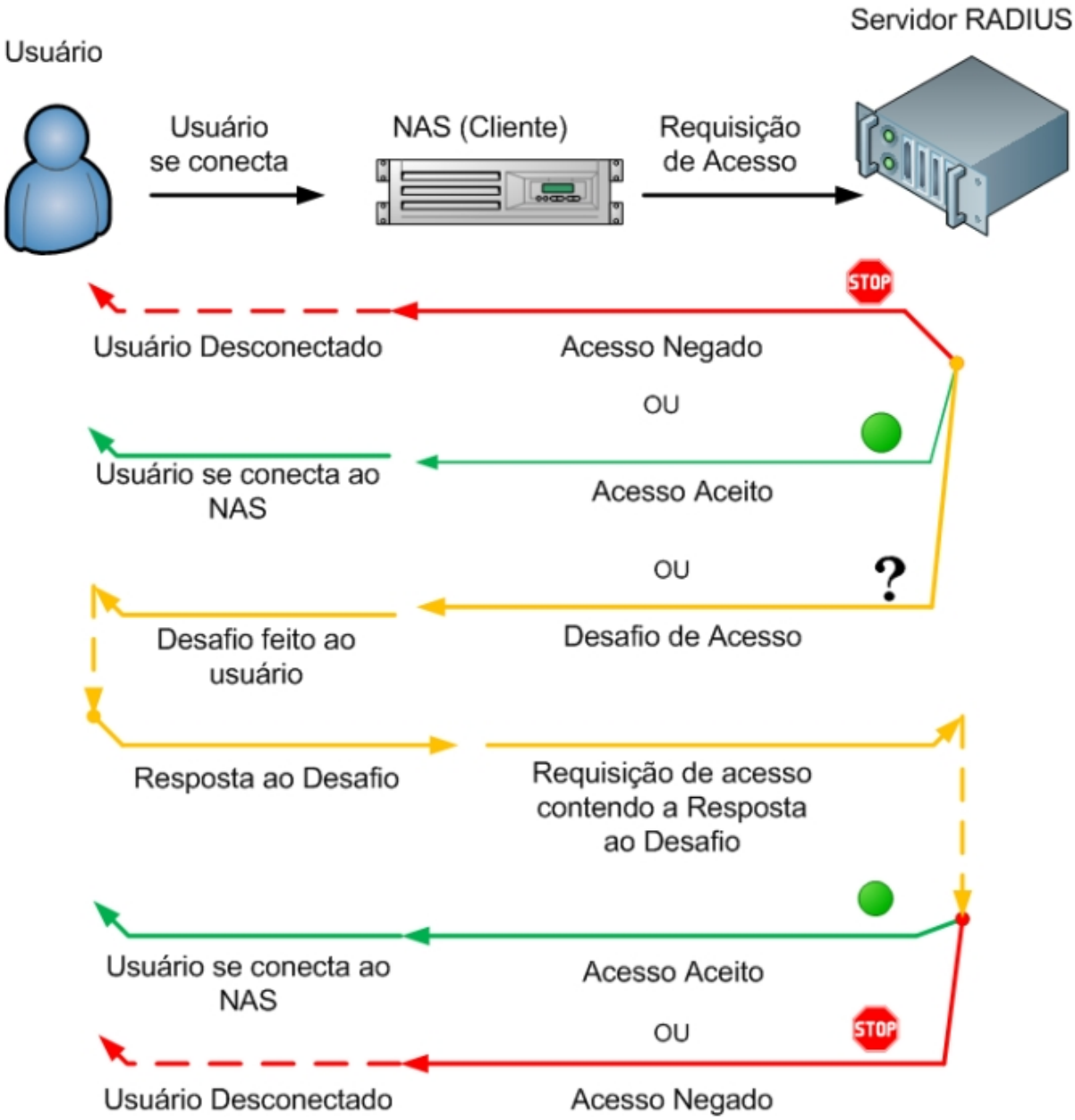

Figura 3 – Estabelecendo uma conexão
Fonte: www.google.com.br (2010)

A mensagem de Requisição de Acesso é enviada pela rede. Se não houver resposta em um limite de tempo pré-estabelecido, a Requisição de Acesso é re-enviada. O cliente ainda pode enviar a Requisição para outros servidores alternativos, no caso de falha na conexão com o servidor primário. Os algoritmos que permitem a troca de servidores são baseados no número de tentativas de acesso ou em um esquema baseado em round-robin.

Após o recebimento da mensagem de Requisição, o servidor RADIUS tenta validar o cliente. O servidor RADIUS descarta silenciosamente os clientes para os quais não possui um segredo compartilhado. Após a autenticação do cliente, o

RADIUS consulta um banco de dados de usuários para verificar se o usuário que está solicitando acesso ao NAS possui as permissões necessárias para ter o acesso garantido.

No caso em que o servidor RADIUS está atuando como um Proxy, ele repassa a Requisição de Acesso para o outro servidor RADIUS.

Se alguma das condições não for satisfeita pelo usuário, o RADIUS emite um pacote de acesso negado, que indica que a requisição feita pelo usuário é invalida. O NAS recebe esta mensagem e desconecta o usuário, podendo ou não enviar alguma mensagem de aviso. Se o usuário satisfizer a todas as condições, o servidor RADIUS pode, de forma a garantir uma segurança adicional, enviar um pacote do tipo Desafio de Acesso. Este desafio pode vir na forma de uma pergunta que somente o usuário sabe ou vir na forma de um código que somente os usuários com determinados dispositivos como *smartcards* ou *softwares* específicos conseguirão responder corretamente, garantindo a identidade do mesmo. A resposta ao Desafio de Acesso é outra mensagem do tipo Requisição de Acesso, no qual o campo com a senha do usuário é preenchida com a resposta criptografada do desafio e uma marcação para indicar que a requisição é uma resposta ao Desafio de Acesso.

Por fim, se todas as condições forem satisfeitas, e se o usuário responder corretamente ao desfio de acesso, o servidor RADIUS envia um pacote do tipo Acesso Aceito, que contém informações a respeito do serviço a ser oferecido. Dentre estas informações, podemos citar o tipo de serviço, o IP atribuído ao usuário, parâmetros de configuração, tempo de acesso máximo, o protocolo que será utilizado, etc. (CARVALHO, 2008)

4.12. Métodos De Autenticação Suportados

O RADIUS permite a utilização de diferentes métodos de autenticação, de forma que cada serviço utilize o método mais adequado para suprir suas necessidades.

4.12.1. Password Authentication Protocol (PAP)

O Protocolo de Autenticação por Senha é um protocolo de autenticação simples. Este protocolo transmite as senhas na forma de texto plano (sem criptografia) em ASCII. Justamente, devido a não-criptografia das senhas, este protocolo é considerado inseguro. Ele ainda é utilizado como uma última solução para efetuar a autenticação de usuários em cenários onde o servidor não suporta protocolos mais seguros. No processo de autenticação com *Password Authentication Protocol* (PAP), o cliente envia o nome do usuário e a sua senha e o servidor envia um ACK (acesso permitido) ou envia um NAK (acesso não permitido). Para o caso específico do RADIUS, caso o cliente não suporte outras formas de autenticação, os pacotes do tipo “Desafio de Acesso” são considerados como pacotes de “Acesso Negado”. (CARVALHO, 2008)

4.12.2. *Challenge-Handshake Authentication Protocol* (CHAP)

O *Challenge-Handshake Authentication Protocol* (CHAP - Protocolo de Autenticação por Desafios de Identidade) é um método relativamente seguro de autenticação, em comparação a protocolos mais simples como o PAP. O CHAP verifica periodicamente a identidade do usuário através de um reconhecimento em três etapas (*three-wayhandshake*). Esta autenticação ocorre no estabelecimento da conexão, e também pode ocorrer a qualquer momento após o seu estabelecimento. A verificação da autenticidade dos usuários é feita através do segredo compartilhado (*shared secret*) entre o usuário e o servidor RADIUS.

Após a etapa de estabelecimento da conexão, o servidor RADIUS envia um desafio para o usuário. O usuário então emite uma resposta que contém o hash do segredo compartilhado. O servidor de Autenticação então verifica o valor do hash enviado e o compara com o hash gerado por ele mesmo. Caso o valor esteja correto, o servidor envia um reconhecimento positivo (ACK). Caso contrário, o servidor finaliza a conexão. Em intervalos de tempo aleatórios, o servidor realiza novamente um desafio para o usuário. (CARVALHO, 2008)

4.12.3. Extensible Authentication Protocol (EAP)

O *Extensible Authentication Protocol* (EAP) - Protocolo de Autenticação Extensível) é um arcabouço de autenticação usado principalmente em conexões PPP e em redes sem-fio. O EAP provê funções de negociação a respeito de quais métodos de autenticação serão utilizados e, além disso, oferece suporte aos múltiplos métodos de autenticação, chamados de métodos EAP. Dentre estes métodos, podemos citar o EAP-MD5, EAP-TLS (muito utilizado em redes sem-fio) e EAP-TTLS. O servidor que deseja autenticar o cliente, requisita informações adicionais a respeito do cliente, e algum método EAP é então solicitado para prover a autenticação. (CARVALHO, 2008)

5. IMPLEMENTANDO O SERVIDOR RADIUS

Aqui será demonstrada a implementação do servidor RADIUS, bem como suas funções básicas de funcionalidade e arquivos de configuração.

Primeiramente, será realizada a instalação de alguns programas para que o servidor RADIUS funcione adequadamente. Essa instalação terá como base de

autenticação um banco de dados do próprio freeRadius implementado com a solução de banco de dados MySQL. Para a manipulação do banco de dados é necessária uma ferramenta de gerenciamento, essa ferramenta será o phpMyadmin. Para a instalação do phpMyadmin é necessário ter instalado, primeiramente, uma solução de servidor *web*, aqui será instalado o APACHE com suporte a PHP.

No terminal do servidor faremos os seguintes comandos:

#apt-getinstall apache2

Após a instalação do apache instalaremos o suporte para PHP:

#apt-get install php5 php5-mysql php5-radius

Com o comando anterior, será instalado o suporte para PHP tanto para o servidor de aplicação quanto para as implementações posteriores do phpmyadmin e freeradius.

Para usar o modo de autenticação de usuário e senha pelo banco de dados, precisamos também implementar a solução de banco de dados no nosso servidor, aqui 2929er utilizando o MySQL. Para a instalação do MySQL utilizaremos os seguintes comandos no terminal:

#apt-get installmysql-server mysql-client

O comando anterior fará a instalação da aplicação de banco de dados MySQL.

Para podermos manipular com maior facilidade as informações contidas no banco de dados do servidor RADIUS, se pode instalar uma solução como o phpmyadmin. Ele possui uma interface *web* que possibilita a manipulação das informações no banco de dados do servidor RADIUS. Para a instalação do phpmyadmin, digitamos no terminal o seguinte comando:

#apt-get install (1)phpmyadmin

Com as instalações anterior, o servidor está pronto para receber as implementações do servidor freeRadius. Depois de feita as implementações anteriores, nosso servidor está pronto para receber a solução freeRadius. Para as

implementações do freeRadius, será necessária a instalação dos seguintes programas, segundo o comando no terminal:

#apt-get install freeradius freeradius-mysql

O servidor de freeRadius está basicamente instalado e pronto para receber as configurações necessárias para a implementação de segurança de autenticação de usuário e senha. Para um teste básico no servidor, podemos fazer o seguinte comando:

#radtest root “senha root” 127.0.0.1 1812 testing123

O comando anterior é uma ferramenta do próprio freeRadius para testar as conexões dos usuários ao servidor, no qual radtest é o programa que realiza a verificação do usuário “root” com a própria senha do usuário, o endereço do servidor RADIUS, que no caso é local “127.0.0.1”, e a senha do servidor RADIUS “testing123” senha essa padrão do freeRadius.

(1) Essa ferramenta só pode ser utilizada se feita as implantações descritas acima, a instalação do servidor *WEB* (APACHE) e o suporte a linguagem de programação PHP.

6. CONFIGURANDO O SERVIDOR

Para que o servidor RADIUS atue com o banco de dados, é preciso realizar algumas configurações prévias para que o servidor RADIUS trabalhe de forma que busque as informações dos usuários no banco de dados.

Primeiro, navegamos até o diretório no qual se encontra instalado o servidor, “/etc/freeradius”. Neste diretório, como demonstrado anteriormente, estão todos os arquivos de configuração do servidor. Começaremos pelo arquivo radiusd.conf, com o comando a seguir:

#vim radiusd.conf

Navegamos pelo arquivo até encontrar a linha “# $ INCLUDED 31sql.conf”. A linha estará comentada. Descometamos a linha retirando o símbolo “#” do começo.

Essa é a linha na qual se autoriza a configuração do arquivo sql.conf, no mesmo diretório, editamos o arquivo 31sql.conf. Navegamos no arquivo até encontrar a linha *Login* e *Password*, informamos o *login* e senha do banco de dados MYSQL, este *login* e senha devem possuir autorização para fazer consulta e editar o banco de dados do freeRadius.

No mesmo diretório, navegamos até a pasta “sites-avalible”, na qual encontramos o arquivo *default*. Será necessária descometa todas as linhas que possuem “sql” para que o servidor trabalhe com as consultas ao banco de dados, com o comando a seguir:

#vim default

Voltando ao diretório de instalação do freeRadius, precisamos configurar o *Access Point,* isso é feito no arquivo clientes.conf, como a seguir:

#vim clientes.conf

Como descrito no comando anterior, precisamos colocar as informações correspondentes ao AP. Procuramos no arquivo pelas configurações que por padrão acompanham o arquivo do servidor, e junto às linhas colocamosda seguinte forma as configurações:

```
Client 192.168.0.2/24 {
        secret          = freeradius
        shortname               = dlink-150
        nastype                 = other
}
```

client 192.168.0.2/24 {} : endereçopadrão do access point.
Secret: senha de comunicação entre o servidor e o *Access Point*.
Shortname: nome demonstrativo do Access Point.
Nastype: tipo do Access Point.

6.1. AUTENTICAÇÃO

Aqui demonstraremos uma das formas de implementação de segurança do freeRadius baseada em usuário e senha. O servidor possui um banco de dados no qual estará cadastrado previamente todos os utilizadores da rede onde o servidor RADIUS atuará. Para que essa segurança seja implementada, será necessária a criação do banco de dados do freeRadius. Por padrão o serviço free Radius dispõe de um *schema* padrão para a criação do banco de dados. Navegamos até o seguinte diretório para encontrar o *schema* do banco de dados, "/etc/freeradius/SQL/ mysql", e dentro desse diretório encontramos os arquivos schema.sql necessário para a criação das tabelas do banco de dados do freeRadius. Para criar o banco, digitamos no terminal.

mysql –u root –p
Enterpassword:

Feito o comando anterior, será pedido à senha do usuário do banco de dados root. Colocamos a senha, conforme os seguintes comandos:

#mysql>createdatabase radius;

Criamos o banco de dados para o servidor freeRadius, com os seguintes comandos:
#mysql> grant all on radius.* to root@localhost identified by 'radius';

O comando anterior habilita o acesso ao banco para as consultas dos usuários cadastrados.

A partir desse ponto, o banco de dados do freeRadius esta criado e pronto para receber as tabelas que se necessita para o cadastro dos usuários e das configurações do servidor, onde o freeRadius atuará. Essas tabelas estão pré-configuradas em um arquivo que vem na instalação do freeRadius. Esse arquivo se chama schema.sql, arquivo no qual esta toda a criação das tabelas necessárias para o freeRadius, esse arquivo se encontra no diretório /etc/freeradius/32er/mysql.

Para a configuração das tabelas, será usado os seguintes comandos:

#mysql –u root –p radius <schema.sql
Password:

Feito o comando anterior, será necessário inserir a senha do banco de dados com o usuário root do banco de dados. Depois de colocado a senha, e se não retornar nenhum erro, a operação foi realizada corretamente. Para verificarmos as tabelas no banco, podemos utilizar o phpmyadmin para visualizar as tabelas, conforme a figura 4, a seguir:

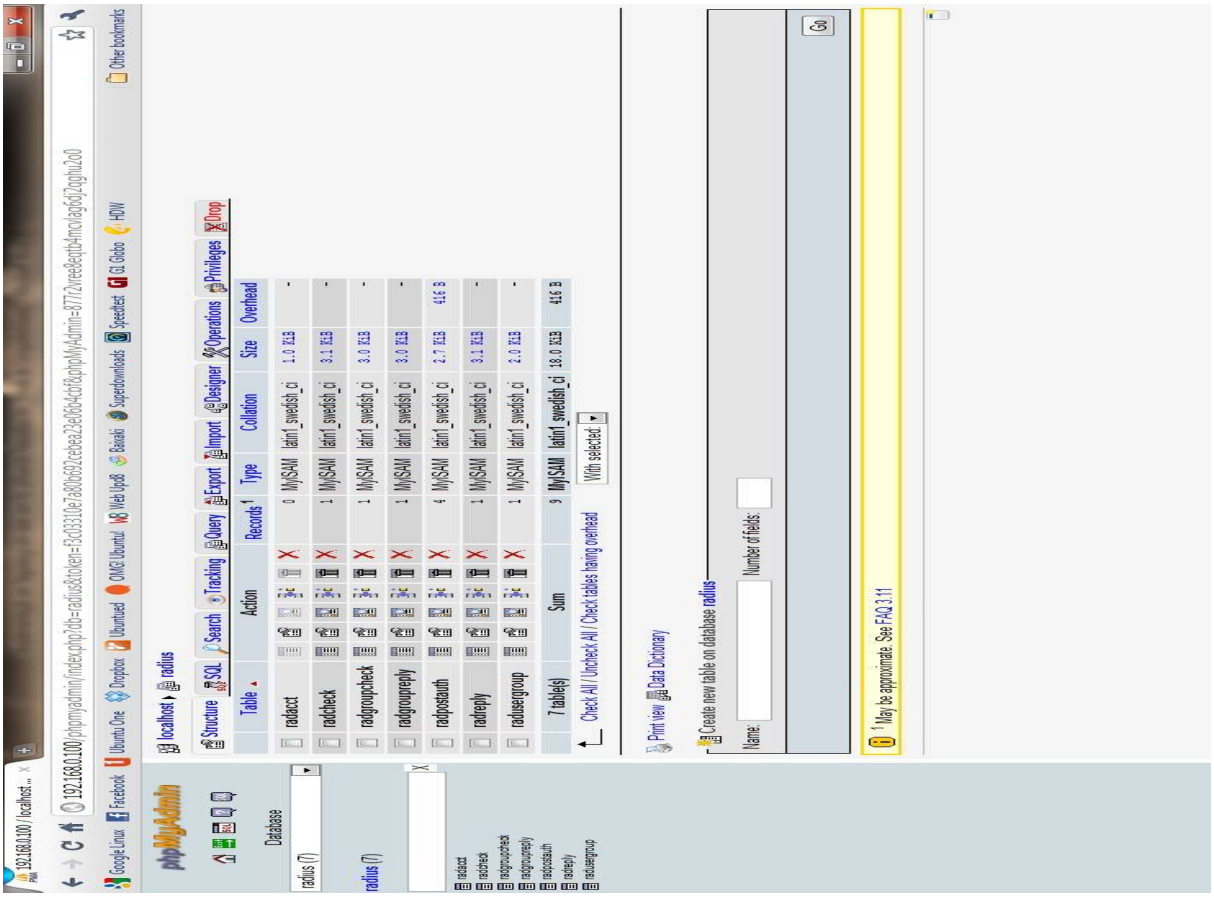

FIGURA 4 – Banco de Dados freeRadius

Fonte: Fonte: Autoria Própria

Agora será necessário configurar os grupos de usuários e os usuários que utilizarão a rede.

Navegamos até a tabela radcheck. Nela será cadastrado todos os usuários da rede local, somente mediante a este cadastro que o usuário terá acesso aos recursos disponíveis na rede, conforme a figura 5, a seguir:

FIGURA 5 – Banco de Dados freeRadiusRadcheck

Fonte: Autoria Própria

Nesta tabela, preenchemos os campos ‘username’ com o nome do usuário da rede, o campo ‘atribute’, no qual preenchemos com ‘User-Password’, o campo ‘op’ para determinar o operador que o freeRadius utilizará para esse usuário e o campo ‘value’ com a senha do usuário cadastrado.

O próximo passo 3434er cadastrar o grupo no qual este usuário atuará, e cadastrar o usuário no grupo, para cadastrar o usuário navegamos até a tabela radgroupcheck, conforma a figura 6, a seguir:

FIGURA 6 – Banco de Dados freeRADIUSRadgroupcheck
Fonte: Autoria Própria

Nesta tabela, preenchemos os campos ‘groupname’ com o nome do grupo que foi cadastrado os usuários, o campo ‘atribute’ com o mesmo atributo do usuário da tabela de usuários, o campo ‘op’ também com o atributo do usuário e o campo ‘value’ com uma senha para o grupo.

Seguindo com a configuração do grupo, precisa preencher a tabela radgroupreply, conforme a figura 7, a seguir:

FIGURA 7 – Banco de Dados freeRADIUSRadgroupreply
Fonte: Autoria Própria

Nesta tabela, será configurado o grupo cadastrado para que o servidor RADIUS possa responder às requisições referentes aos usuários cadastrados neste grupo. Preenchemos os campos ‘groupname’, ‘atribute’, ‘op’ e ‘value’ correspondentes às mesmas configurações do grupo na tabela anterior.

A próxima tabela é um *log* do sistema o qual indica as tentativas de conexão dos usuários da rede local, conforme a figura 8, a seguir:

FIGURA 8 – Banco de Dados freeRADIUSRadpostauth
Fonte: Autoria Própria

Esta tabela é preenchida automaticamente a cada tentativa de conexão na rede. Ela informa o usuário que tentou fazer o *login* na rede, a senha do mesmo a informação de acesso e a data que foi realizada a tentativa de conexão.

Na tabela seguinte será cadastrado o usuário do grupo. Esta tabela contém os usuários cadastrados para que o servidor RADIUS responda às requisições de tentativa de acesso do sistema, conforme a figura 9, a seguir:

FIGURA 9 – Banco de Dados freeRADIUSRadreply
Fonte: Autoria Própria

Nesta tabela preenchemos os campos ‘username’, ‘atribute’, ‘op’ e ‘value’ de acordo com as mesmas informações do usuário contidas na tabela radcheck.

Na próxima tabela, faremos a configuração do usuário com o grupo, conforme a figura 10, a seguir:

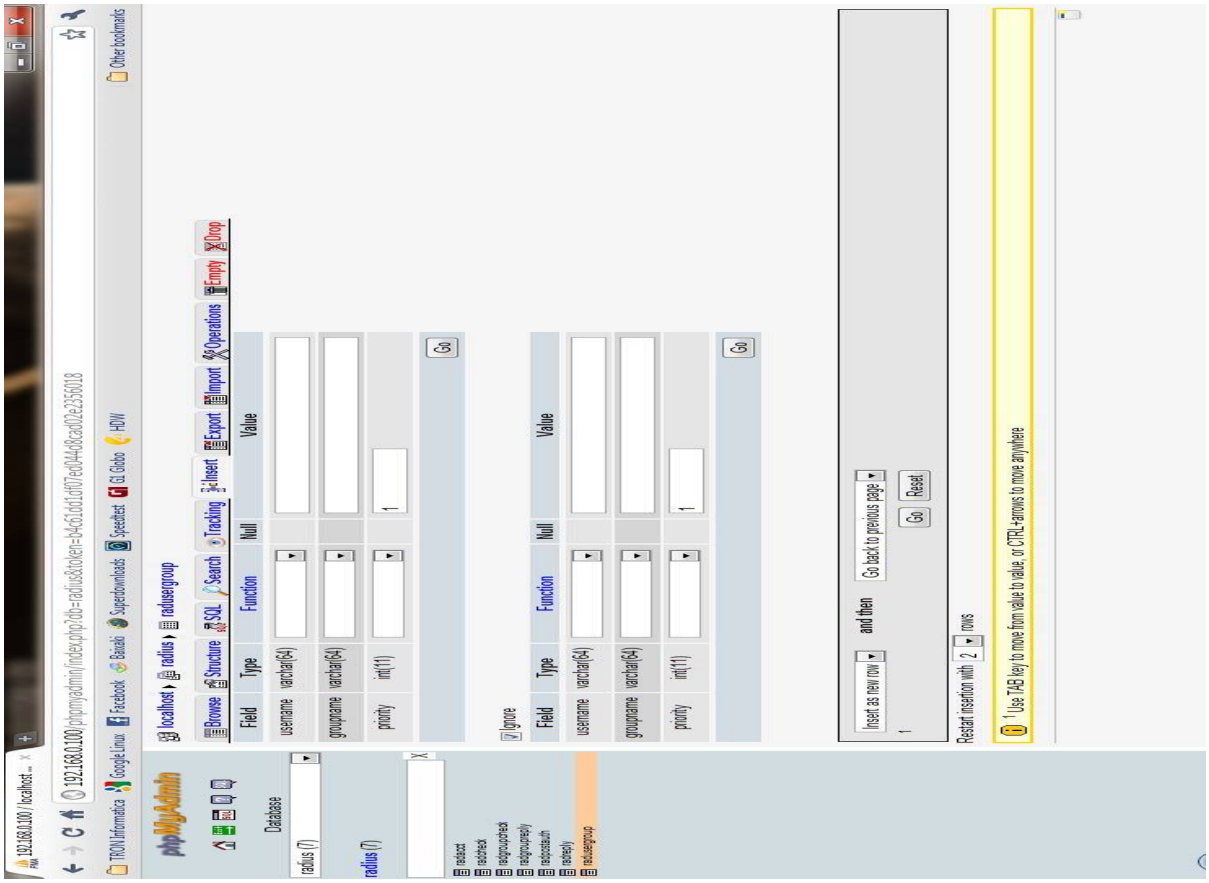

FIGURA 10 – Banco de Dados freeRADIUSRadusergroup
Fonte: Autoria Própria

Nesta tabela, preenchemos o ‘username’, ‘groupname’ e a ‘priority’, os campos são sugestivos, o usuário preenche com o nome do usuário e o grupo que ele pertencera e a prioridade que o grupo terá quando uma requisição de acesso for feita no servidor. Para testarmos as configurações feitas anteriormente, podemos usar o mesmo teste que fizemos para testar a instalação do servidor com a ferramenta radtest.

7. CONFIGURANDO O *ACCESS POINT*

Realizadas as configurações no servidor, é necessário configurar o *Access Point* para que o servidor responda às requisições que partiram da rede WiFi, conforme a figura 12, a seguir:

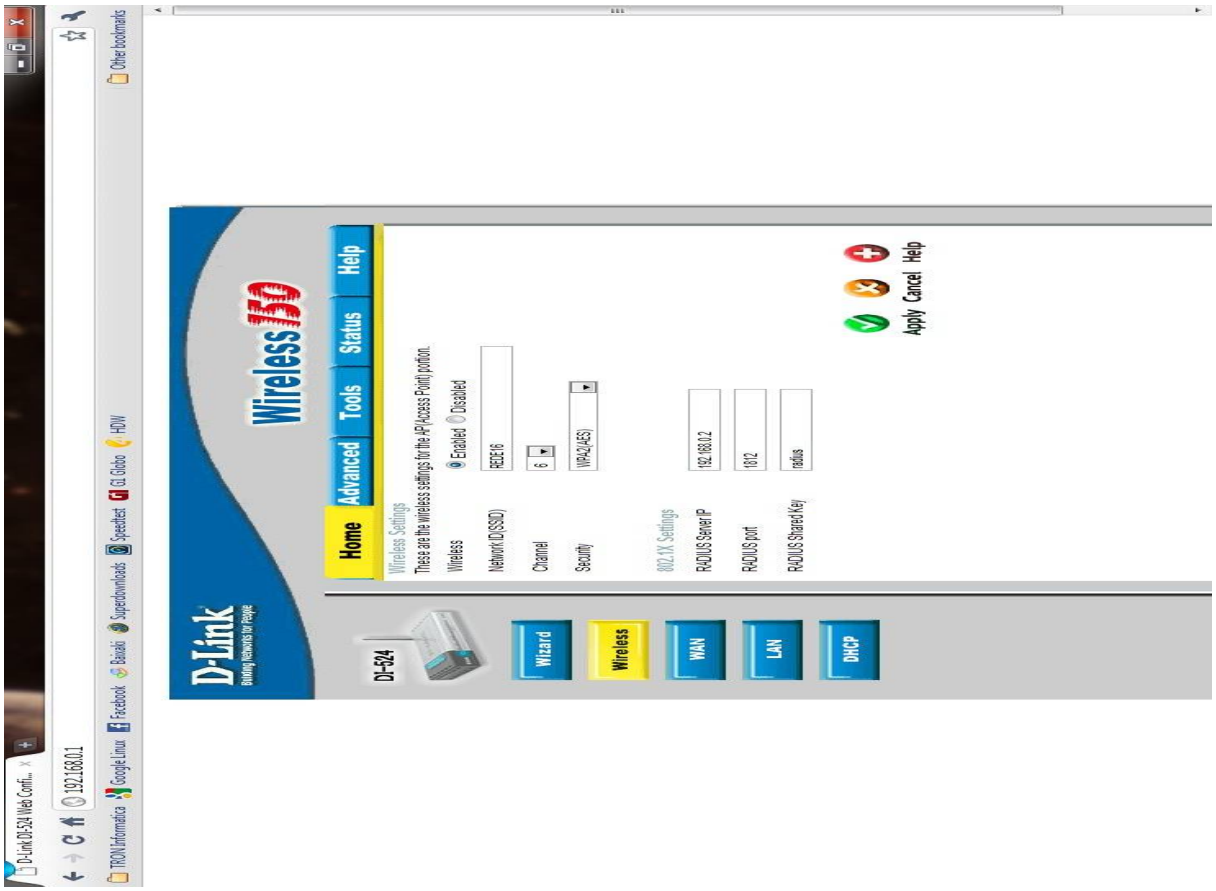

FIGURA 11 – Configurando Acess Point
Fonte: Autoria Própria

Analisando a tela da figura 12, podemos preceber a configuração em um *Access Point* D-Link150. Nas configurações *wireless*, selecionamos a segurança como WPA, WPA2(AES) e ou 802.1x, dependendo da configuração de nosso servidor RADIUS, neste caso, selecionamos WPA2(AES). No RADIUS Server IP. Devemos colocar o IP do nosso servidor RADIUS, em RADIUS port colocamos a porta correspondente do servidor, por padrão a porta do servidor RADIUS é 1812 e, por fim, em RADIUS Shared Key, colocamos a senha que foi configurada previamente no arquivo clients.conf.

8. CONFIGURANDO O CLIENTE (USUÁRIO)

A conexão do usuário é muito simples; quando selecionada a rede, conforme a figura 13, a seguir, o usuário terá de inserir o nome de usuário e a senha que estão previamente cadastrada no banco de dados do servidor RADIUS.

FIGURA 12 – Autenticação do Cliente (Usuário)
Fonte: Autoria Própria

CONSIDERAÇÕES FINAIS

A segurança de uma rede sem fio dependente do nível da segurança implantada pelo administrador, independente da tecnologia, sistemas operacionais e outros fatores; sempre haverá um risco eminente e, principalmente, teremos vulnerabilidade.

Lembramos que o trabalho apresenta uma solução simples, porém bem eficaz utilizando um servidor de autenticação com banco de dados e protocolos de criptografias existentes na própria solução.

Em conclusão ao trabalho realizado, podemos notar que qualquer tipo de tecnologia empregada para reduzir os riscos de ataque cresce conforme a necessidade das redes sem fio ou local. As tecnologias estão evoluindo com o passar do tempo, tornando mais difícil a ação de ataques. Se a tecnologia em rede sem fio for implementada em um ambiente, que seja adotado um método de autenticação e que as partes que estão envolvidas, como ponto de acesso, o servidor de banco de dados e o servidor de autenticação, estejam plenamente configurados para, em fim, aumentar o nível de segurança e diminuir os riscos de ataques.

REFERÊNCIAS BIBLIOGRÁFICAS E SITES

AESWinner.Nacional Institute of Standards and Tecnology, NIST.http://www.nist.gov/public_affairs/releases/g00-176.htm.

AMODEI JR, Aurélio. Esquema de Modulacão do IEEE 802.11. Rio de Janeiro 2003 http://www.gta.urrj.br/seminarios/semi2003_1/aurelio/

Hill, J., “An Analysis of the RADIUS Authentication Protocol” (2001), InfoGard Laboratories.http://www.untruth.org/%7Ejosh/security/radius/radius- auth.html

Hugo EijiTibana Carvalho, RADIUS, http://www.gta.ufrj.br/grad/08_1/radius/index.html, Acesso Maio 2011.

Institute of Eletrical and Electronics Engineers, http://www.ieee.org/, Acessado em Agosto 2010.

Interlink Networks ,“Wireless LAN Access Control and Authentication”,http://www.interlinknetworks.com/whitepapers/WLAN_Access_Control.pdf

Interlink Networks, “ History of the AAA RADIUS Server and RADIUS Protocol”,http://www.interlinknetworks.com/app_notes/History%20of%20RADIUS.pdf

Rigney, C., RFC 2866 “RADIUS Accounting”(2000), http://www.ietf.org/rfc/rfc2866.txt

Rigney, C., Wilens, S., Rubens, A., Simpson, W., RFC 2865, "Remote Authentication

Dial In User Service, (RADIUS)"(2000) http://www.ietf.org/rfc/rfc2865.txt“The

FreeRADIUS Project”**.** http://www.freeradius.org/

Rigney, C., Willats, W., Calhoun, C., RFC 2869, "RADIUS Extensions"(2000), http://www.ietf.org/rfc/rfc2869.txt

Under-Linux, http://www.under-linux.com, Acessado em Agosto de 2010.

Ventura, H., “Diameter: next generation´s AAA protocol”, http://www.diva- portal.org/diva/getDocument?urn_nbn_se_liu_diva-1195- 1__fulltext.pdf

Viva o Linux, http://www.vivaolinux.com.br, Acessado em Agosto em 2010.

WiFi, http://www.wifi.org/, Acessado em Agosto 2010.