UNIVERSIDADE TECNOLÓGICA FEDERAL DO PARANÁ
DEPARTAMENTO DE INFORMÁTICA
ESPECIALIZAÇÃO EM TELEINFORMÁTICA E REDES DE COMPUTADORES

ISSAM IBRAHIM

CONJUNTO DE PROTOCOLOS TCP/IP E SUAS FALHAS

CURITIBA
2011

Issam Ibrahim

CONJUNTO DE PROTOCOLOS TCP/IP E SUAS FALHAS

Trabalho apresentado ao curso de especialização em Teleinformática e Redes de Computadores da Universidade Tecnológica Federal do Paraná, como requisito parcial à obtenção do título de Especilista em Teleinformática e Redes de Computadores

Orientador:
Lincoln Herbert Teixeira

UNIVERSIDADE TECNOLÓGICA FEDERAL DO PARANÁ

Curitiba

Lista de Figuras

Resumo

Este trabalho tem como objetivo estudar as principais características do conjunto de protocolos TCP/IP e descrever alguns dos problemas clássicos. Foi feita uma análise das camadas que compõem este conjunto de protocolos e a descrição de problemas clássicos como SYN floods, IP spoofing, sequestro de conexão TCP, entre outros.

Palavras-chave: protocolos TCP/IP, falhas TCP/IP, SYN floods, IP spoofing, sequestro de conexão TCP, source routing, ataques ICMP.

Abstract

This final project is in order to discuss TCP/IP fails and describe the most frequent problems in this protocols. An analysis of the layers that make up this set of protocols and description of classical problems such as SYN flood, IP spoofing, TCP hijacking, among others.

Key words: TCP/IP protocols, TCP/IP fails, SYN floods, IP spoofing, TCP hijacking, source routing, ICMP attacks.

Sumário

1 Introdução

O TCP/IP é o protocolo de rede mais usado atualmente, devido a arquitetura da Internet [1].

Um protocolo é uma linguagem usada para permitir que dois ou mais computadores se comuniquem. Assim como acontece no mundo real, se eles não falarem a mesma língua eles não podem se comunicar.

O TCP/IP não é na verdade um protocolo único, mas sim um conjunto de protocolos - uma pilha de protocolos, como ele é mais chamado. Seu nome, por exemplo, já faz referência a dois protocolos diferentes, o TCP (Transmission Control Protocolo - Protocolo de Controle de Transmissão) e o IP (Internet Procolo - Protocolo de Internet). Existem muitos outros protocolos que compõem a pilha de protocolos TCP/IP, como o FTP, o HTTP, o SMTP e o UDP, além de muitos outros.

Este conjunto de protocolos, a base da Internet hoje, possui uma série de problemas básicos de segurança, como autenticação ou criptografia. Segurança passa a ser um desafio a cada dia que passa pois novas falhas vão sendo descobertas.

A segurança de redes é evidentemente um tema que transcende o TCP/IP, e envolve desde questões relacionados com os princípios físicos, eletromagnéticos ou ópticos de funcionamento da comunicação, até a chamada ”engenharia social”, que estuda (por exemplo) o comportamento das pessoas em relação à escolha ou ao uso de senhas. Naturalmente não serão abordadas todas essas questões, mas apenas fazer algumas observações práticas no tocante à forma com que os temas da segurança e do TCP/IP se entrelaçam. De fato, a análise ou o planejamento de uma política de segurança em redes IP dependem de um conhecimento sólido de TCP/IP [1].

2 *Como funcionam os Protocolos TCP/IP*

Neste capítulo, serão descritas algumas características do conjunto de protocolos TCP/IP. Mas antes de entrar em detalhes de como este conjunto de protocolos funciona, serão descritas algumas particularidades/conceitos importantes do protocolo TCP, de importância para o entendimento do trabalho.

2.1 Protocolo TCP

O TCP (acrônimo para o inglês de Transmission Control Protocol) é um dos protocolos sob os quais assenta o núcleo da Internet. A versatilidade e robustez desde protocolo tornou-o adequado a redes globais, já que este verifica se os dados são enviados de forma correta, na sequência apropriada e sem erros pela rede [2].

O TCP é um protocolo do nível da camada de transporte do modelo OSI e é sobre o qual assentam a maioria das aplicações cibernéticas, como o SSH, FTP, HTTP - portanto a World Wide Web.

2.1.1 Descrição do Funcionamento

O protocolo TCP especifica três fases durante uma conexão: estabelecimento da ligação, transferência e término de ligação. O estabelecimento da ligação é feito em três passos, enquanto que o término é feito em quatro. Durante a inicialização são inicializados alguns parâmetros, como o Sequence Number (número de sequência) para garantir a entrega ordenada e robustez durante a transferência.

2.1.2 Estabelecimento da Ligação

Tipicamente, numa ligação TCP existe aquele designado de servidor (que abre um socket e espera passivamente por ligações) num extremo, e o cliente no outro. O cliente inicia a ligação enviando um pacote TCP com a flag SYN activa e espera-se que o servidor aceite a ligação enviando um pacote SYN+ACK. Se, durante um determinado espaço de tempo, esse pacote não for recebido ocorre um timeout e o pacote SYN é reenviado. O estabelecimento da ligação é concluído por parte do cliente, confirmando a aceitação do servidor respondendo-lhe com um pacote ACK.

Durante estas trocas, são trocados números de sequência iniciais (ISN) entre os interlocutores que irão servir para identificar os dados ao longo do fluxo, bem como servir de contador de bytes transmitidos durante a fase de transferência de dados (sessão).

Este processo é chamado de Handshake ou aperto de mão. É o processo pelo qual duas máquinas afirmam uma a outra que a reconheceu e está pronta para iniciar a comunicação. O handshake é utilizado em protocolos de comunicação, tais como: FTP, TCP, HTTP, SMB,SMTP,POP3 etc. Na figura 2.1 é ilustrado o processo.

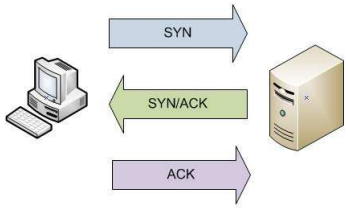

Figura 2.1: Three Way Handshake

No final desta fase, o servidor inscreve o cliente como uma ligação estabelecida numa tabela própria que contém um limite de conexões, o backlog. No caso do backlog ficar completamente preenchido a ligação é rejeitada ignorando (silenciosamente) todos os subsequentes pacotes SYN.

2.1.3 Transferência de Dados - sessão

Durante a fase de transferência o TCP está equipado com vários mecanismos que asseguram a confiabilidade e robustez: números de sequência que garantem a entrega ordenada, código detector de erros (checksum) para detecção de falhas em segmentos específicos, confirmação de recepção e temporizadores que permitem o ajuste e contorno de eventuais atrasos e perdas de segmentos [2].

Observando-se o cabeçalho TCP, existem permanentemente um par de números de sequência, doravante referidos como número de sequência e número de confirmação (ACKnowledgement). O emissor determina o seu próprio número de sequência e o receptor confirma o segmento usando como número ACK o número de sequência do emissor. Para manter a confiabilidade, o receptor confirma os segmentos indicando que recebeu um determinado número de bytes contíguos. Uma das melhorias introduzidas no TCP foi a possibilidade do receptor confirmar blocos fora da ordem esperada.

A remontagem ordenada dos segmentos é feita usando os números de sequência, de 32 bit, que reiniciam a zero quando ultrapassam o valor máximo. Assim, a escolha do ISN torna-se vital para a robustez deste protocolo.

As confirmações de recepção (ACK) servem também ao emissor para determinar as condições da rede. Dotados de temporizadores, tanto os emissores como receptores podem alterar o fluxo dos dados, contornar eventuais problemas de congestão e, em alguns casos, prevenir o congestionamento da rede.

2.1.4 Término da Ligação

A fase de encerramento da sessão TCP é um processo de quatro fases, em que cada interlocutor responsabiliza-se pelo encerramento do seu lado da ligação. Quando um deles pretende finalizar a sessão, envia um pacote com a flag FIN ativa, ao qual deverá receber uma resposta ACK. Por sua vez, o outro interlocutor irá proceder da mesma forma, enviando um FIN ao qual deverá ser respondido um ACK.

Pode ocorrer, no entanto, que um dos lados não encerre a sessão. Chama-se a este tipo de evento de conexão semi-aberta. O lado que não encerrou a sessão poderá continuar a enviar informação pela conexão, mas o outro lado não. Na figura 2.2 é ilustrado o término de uma ligação.

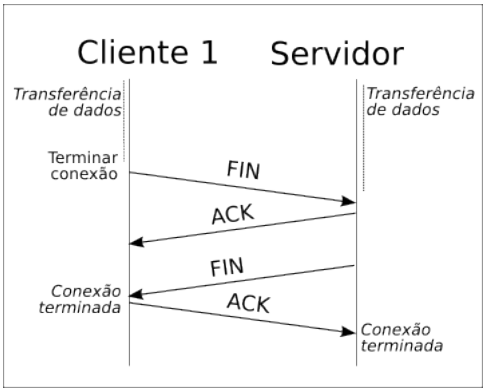

Figura 2.2: Término da Conexão

2.2 O conjunto de Protocolos TCP/IP

A arquitetura do TCP/IP pode ser vista na figura 2.3:

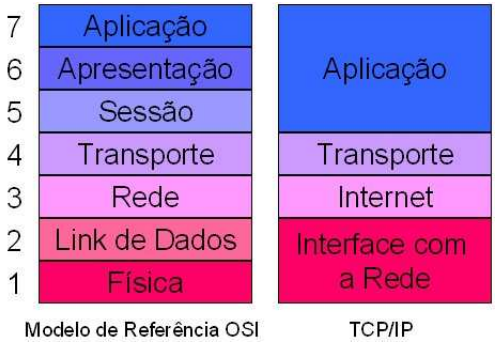

Figura 2.3: Modelo de Camadas TCP/IP

O TCP/IP tem quatro camadas. Os programas se comunicam com a camada de Aplicação. Na camada de Aplicação encontraremos os protocolos de aplicação tais como o SMTP (para e-mail), o FTP (para a transferência de arquivos) e o HTTP (para navegação web). Cada tipo de programa se comunica com um protocolo de aplicação diferente, dependendo da finalidade do programa [2].

Após processar a requisição do programa, o protocolo na camada de Aplicação se comunicará com um outro protocolo na camada de Transporte, normalmente o TCP. Esta camada é responsável por pegar os dados enviados pela camada superior, dividi-los em pacotes e enviá-los para a camada imediatamente inferior, a camada Internet. Além disso, durante a recepção dos dados, esta camada é responsável por colocar os pacotes recebidos da rede em ordem (já que eles podem chegar fora de ordem) e também verificam se o conteúdo dos pacotes está intacto.

Na camada Internet o IP (Internet Protocol, Protocolo Internet), que pega os pacotes recebidos da camada de Transporte e adiciona informações de endereçamento virtual, isto é, adiciona o endereço do computador que está enviando os dados e o endereço do computador que receberá os dados. Esses endereços virtuais são chamados endereços IP. Em seguida os pacotes são enviados para a camada imediatamente inferior, a camada Interface com a Rede. Nesta camada os pacotes são chamados datagramas.

A camada Interface com a Rede receberá os pacotes enviados pela camada Internet e os enviará para a rede (ou receberá os dados da rede, caso o computador esteja recebendo dados). O que está dentro desta camada dependerá do tipo de rede do tipo de computador que estiver sendo usando. Atualmente praticamente todos os computadores utilizam um tipo de rede chamado Ethernet (que está disponível em diferentes velocidades; as redes sem fio também são redes Ethernet) e, portanto, você deve encontrar na camada Interface com a Rede as camadas do Ethernet, que são Controle do Link Lógico (LLC), Controle de Acesso ao Meio (MAC) e Física, listadas de cima para baixo. Os pacotes transmitidos pela rede são chamados quadros.

Nas seções a seguir, serão abordadas, com mais detalhes as camadas e os protocolos TCP/IP.

2.2.1 Camada de Aplicação

Esta camada faz a comunicação entre os programas e os protocolos de transporte. Existem vários protocolos que operam na camada de aplicação. Os mais conhecidos são o HTTP (HyperText Transfer Protocol, Protocolo de Transferência Hipertexto), o SMTP (Simple Mail Transfer Protocol, Protocolo Simples de Transferência de Correspondência), o FTP (File Transfer Protocol, Protoloco de Transferência de Arquivos), o SNMP (Simple Network Management

Protocol, Protocolo Simples de Gerenciamento de Redes), o DNS (Domain Name System, Sistema de Nome de Domínio) e o Telnet.

A camada de aplicação comunica-se com a camada de transporte através de uma porta. As portas são numeradas e as aplicações padrão usam sempre uma mesma porta. Por exemplo, o protocolo SMTP utiliza sempre a porta 25, o protocolo HTTP utiliza sempre a porta 80 e o FTP as portas 20 (para transmissão de dados) e 21 (para transmissão de informações de controle).

O uso de um número de porta permite ao protocolo de transporte (tipicamente o TCP) saber qual é o tipo de conteúdo do pacote de dados (por exemplo, saber que o dado que ele está transportando é um e-mail) e, no receptor, saber para qual protocolo de aplicação ele deverá entregar o pacote de dados, já que, como estamos vendo, existem inúmeros. Assim, ao receber um pacote destinado à porta 25, o protocolo TCP irá entregá-lo ao protocolo que estiver conectado a esta porta, tipicamente o SMTP, que por sua vez entregará o dado à aplicação que o solicitou (no caso, um programa de e-mail).

Na figura 2.4 é ilustrada como a camada de aplicação funciona:

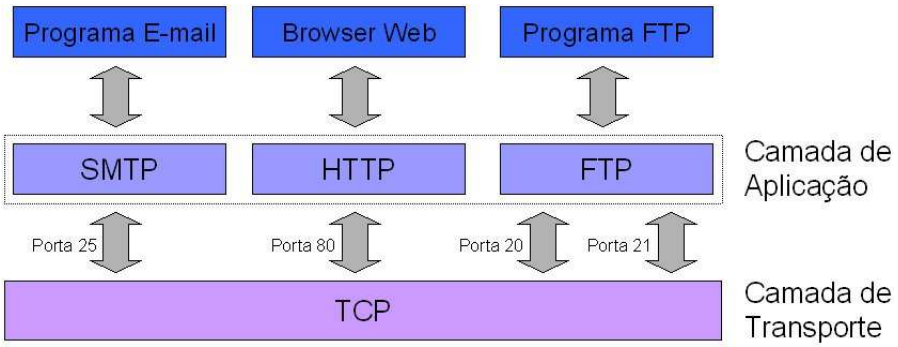

Figura 2.4: Como a camada de aplicação funciona

2.2.2 Camada de Transporte

Na transmissão de dados, a camada de transporte é responsável por pegar os dados passados pela camada de aplicação e transformá-los em pacotes. O TCP (Transmission Control Protocol, Protocolo de Controle da Transmissão) é o protocolo mais usado na camada de Transporte. Na recepção de dados, o protocolo TCP pega os pacotes passados pela camada Internet e trata

de colocá-los em ordem, já que os pacotes podem chegar ao destino fora de ordem, confere se os dados dentro dos pacotes estão íntegros e envia um sinal de confirmação chamado ”acknowledge”(”ack”) ao transmissor, avisando que o pacote foi recebido corretamente e que os dados estão íntegros. Se nenhum sinal de confirmação (”ack”) for recebido (ou porque o dado não chegou ao destino ou porque o TCP descobriu que dado estava corrompido), o transmissor enviará novamente o pacote perdido.

Enquanto que o TCP reordena os pacotes e usa mecanismos de confirmação de recebimento - o que é desejável na transmissão de dados - existe um outro protocolo que opera nesta camada que não tem esses recursos. Esse protocolo é o UDP (User Datagram Protocol - Procolo de Datagrama do Usuário).

Por essa razão o TCP é considerado um protocolo confiável, enquanto que o UDP é considerado um protocolo não confiável. O UDP é tipicament usado quando nenhum dado importante está sendo transmitido, como requisições DNS. Como o UDP não reordena os pacotes e num usa mecanismo de confirmação, ele é mais rápido que o TCP.

Durante a transmissão de dados, tanto o UDP quanto o TCP receberão os dados passados da camada de Aplicação e adicionarão a esses dados um cabeçalho. Na recepção de dados, o cabeçalho será removido antes de os dados serem enviados para a porta apropriada. Neste cabeçalho estão várias informações de controle, em particular o número da porta de origem, o número da porta de destino, um número de seqüência (para a confirmação de recebimento e mecanismos de reordenamento usado pelo TCP) e uma soma de verificação (chamada checksum ou CRC, que é um cálculo usado para verificar se o dado foi recebido intacto no destino). O cabeçalho UDP tem 8 bytes, enquanto que o cabeçalho TCP tem entre 20 e 24 bytes (dependendo se o campo opções estiver sendo ou não usado).

Na figura 2.5 é ilustrado o pacote de dados gerado na camada de transporte. Este pacote de dados será enviado para a camada Internet (se estamos transmitindo dados) ou será recebido da camada Internet (se estamos recebendo dados).

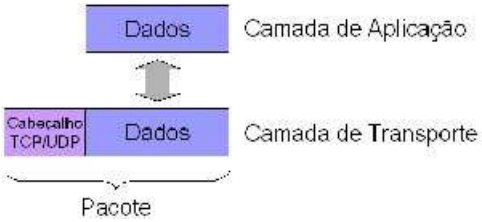

Figura 2.5: Pacote de dado na camada de transporte

2.2.3 Camada Internet

Nos protocolos TCP/IP, cada computador é identificado com um endereço virtual único, chamado endereço IP. A camada Internet é responsável por adicionar um cabeçalho ao pacote de dados recebidos da camada de Transporte onde, entre outros dados de controle, será adicionado também o endereço IP de origem e o endereço IP de destino, isto é, o endereço IP do computador que está enviando os dados e o endereço IP do computador que deverá recebê-los.

A placa de rede de cada computador tem um endereço físico. Este endereço está gravado na memória ROM da placa de rede e é chamado endereço MAC. Dessa forma, em uma rede local se o computador A quiser enviar dados para o computador B, ele precisará saber o endereço MAC do computador B. Enquanto que em uma pequena rede local os computadores podem facilmente descobrir o endereço MAC de todos os PCs, esta não é uma tarefa tão simples em uma rede global como a Internet.

Se nenhum esquema de endereçamento virtual for usado, vé necessário saber o endereço MAC do computador de destino, o que não é apenas uma tarefa complicada, mas também não ajuda no roteamento dos pacotes, já que este endereço não usa uma estrutura em árvore (em outras palavras, enquanto o endereçamento virtual usado na mesma rede terá endereços seqüenciais, com o endereçamento MAC o computador com o endereço MAC seguinte ao seu pode estar em qualquer lugar do mundo).

Roteamento é o caminho que os dados devem usar para chegar ao destino. Em todas as redes conectadas à Internet existe um dispositivo chamado roteador, que faz a ponte entre os computadores de uma rede local e a Internet. Todo roteador tem uma tabela contendo as redes conhecidas e também uma configuração chamada gateway padrão apontando para outro rotea-

dor na Internet. Quando um computador envia um pacote de dados para a Internet, o roteador conectado à rede primeiro verifica se ele conhece o computador de destino - em outras palavras, o roteador verifica se o computador de destino está localizado na mesma rede ou em uma rede que ele conhece a rota. Se ele não conhecer a rota para o computador de destino, ele enviará o pacote para seu gateway padrão, que é outro roteador. Este processo é repetido até que o pacote de dados chegue ao seu destino.

Há vários protocolos que operam na camada Internet: IP (Internet Protocol, Protocolo de Internet), ICMP (Internet Control Message Protocol, Protocolo de Controle de Mensagens Internet), ARP (Address Resolution Protocol, Protocolo de Resolução de Endereços) e RARP (Reverse Address Resolution Protocol, Protocolo de Resolução de Endereços Reversos). Os pacotes de dados são enviados usando o protocolo IP.

O IP pega os pacotes de dados recebidos da camada de Transporte (do protocolo TCP se você está transmitindo dados como e-mails ou arquivos) e os divide em datagramas. O datagrama é um pacote que não contém nenhum tipo de confirmação de recebimento (acknowledge), o que significa que o IP não implementa nenhum mecanismo de confirmação de recebimento, isto é, ele é um protocolo não confiável.

Durante a transferência de dados o protocolo TCP será usado acima da camada Internet (ou seja, acima do IP) e o TCP implementa mecanismo de confirmação de recebimento. Portanto apesar de o protocolo IP não verificar se o datagrama chegou ao destino, o protocolo TCP fará esta verificação. A conexão será então confiável, apesar do IP sozinho ser um protocolo não confiável.

Na figura 2.6 é ilustrado o datagrama gerado na camada Internet pelo protocolo IP. É interessante notar que o que a camada Internet vê como sendo ”dados”é o pacote completo que ela recebe da camada de Transporte, que inclui o cabeçalho TCP ou UDP. Este datagrama será enviado para a camada Interface com a Rede (se estiver transmitindo) ou pode ter sido recebido da camada Interface com a Rede (se estiver recebendo dados).

O cabeçalho adicionado pelo protocolo IP inclui o endereço IP de origem, o endereço IP de destino e várias outras informações de controle.

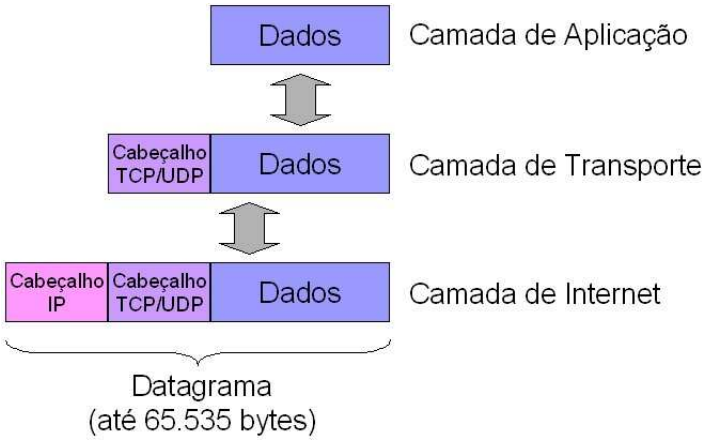

Figura 2.6: Datagrama na camada de Internet

O campo de dados do datagrama não tem um tamanho fixo. Como os datagramas serão transmitidos pela rede dentro de quadros produzidos pela camada Interface com a Rede, normalmente o sistema operacional configurará o tamanho do datagrama IP para ter o tamanho máximo da área de dados do quadro de dados usado em sua rede. O tamanho máximo do campo de dados dos quadros que são transmitidos pela rede é chamado MTU, Maximum Transfer Unit, ou Unidade de Transferência Máxima.

As redes Ethernet - que são o tipo de rede mais comum hoje em dia, incluindo sua encarnação sem fio - pode transportar até 1.500 bytes de dados, ou seja, seu MTU é de 1.500 bytes. Por isso o sistema operacional configura automaticamente o protocolo IP para criar datagramas IP com 1.500 bytes em vez de 65.535 (que não caberia no quadro).

O TCP/IP é um conjunto de protocolos que lida com as camadas 3 a 7 do modelo de referência OSI. O Ethernet é um conjunto de protocolos que lida com as camadas 1 e 2 do modelo de referência OSI - o que significa que o Ethernet se preocupa com o aspecto físico da transmissão de dados. Por isso eles se complementam, já que precisamos das sete camadas completas (ou suas equivalentes) para estabelecer uma conexão de rede.

Outra característica que o protocolo IP permite é a fragmentação. Até chegar a seu destino o datagrama IP provavelmente passará por várias outras redes no meio do caminho. Se todas as redes no caminho entre o computador transmissor e o receptor usarem o mesmo tipo de rede (por exemplo Ethernet), todos os roteadores trabalharão com a mesma estrutura do quadro (isto

é, o mesmo tamanho de MTU).

No entanto, se aquelas outras redes não forem redes Ethernet, elas podem usar um tamanho diferente de MTU. Se isto acontecer, o roteador que está recebendo os quadros com o MTU configurado com 1.500 bytes dividirá o datagrama IP em quantos quadros forem necessários para atravessar a rede com o tamanho de MTU menor. Ao chegar no roteador que tem sua saída conectada a uma rede Ethernet, este roteador remontará o datagrama original.

Na figura 2.7, é possível ver um exemplo de fragmentação. O quadro original usa um MTU de 1.500 bytes. Quando o datagrama chega a uma rede com o tamanho de MTU de 620 bytes, cada quadro tem de ser dividido em três quadros (dois com 600 bytes e um com 300 bytes). Em seguida o roteador na saída desta rede (roteador 2) remonta o datagrama original.

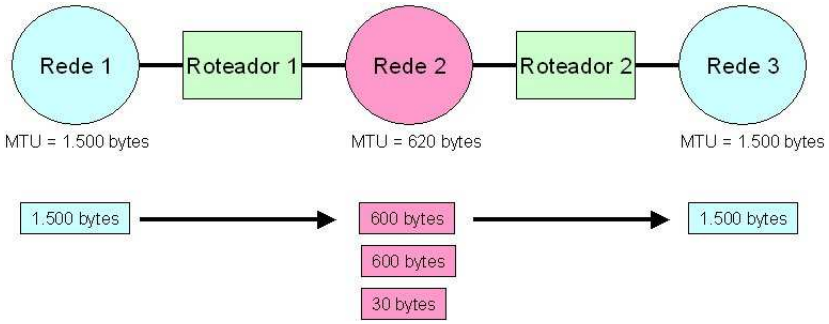

Figura 2.7: Fragmentação de Datagrama

2.2.4 Camada Interface com a rede

Os datagramas gerados na camada Internet serão passados para a camada Interface com a Rede, durante a transmissão de dados, ou a camada de Interface com a Rede pegará os dados da rede e os enviará para a camada de Internet, na recepção dos dados. Esta camada é definida pelo tipo de rede física a qual o computador está conectado. Quase sempre p computador estará conectado a uma rede Ethernet.

O TCP/IP é um conjunto de protocolos que lida com as camadas 3 a 7 do modelo de referência OSI, enquanto que o Ethernet é um conjunto de protocolos que lida com as camadas 1 e 2 do modelo de referência OSI - o que significa que o Ethernet lida com os aspectos físicos da

transmissão de dados. Por isso um complementa o outro, já que precisamos das sete camadas completas (ou suas equivalentes) para estabelecer uma conexão de rede.

O Ethernet tem três camadas: LLC (Controle do Link Lógico), MAC (Controle de Acesso ao Meio) e Física. O LLC e o MAC correspondem, juntas, a segunda camada do modelo de referência OSI. É ilustrado na figura 2.8 a arquitetura do Ethernet.

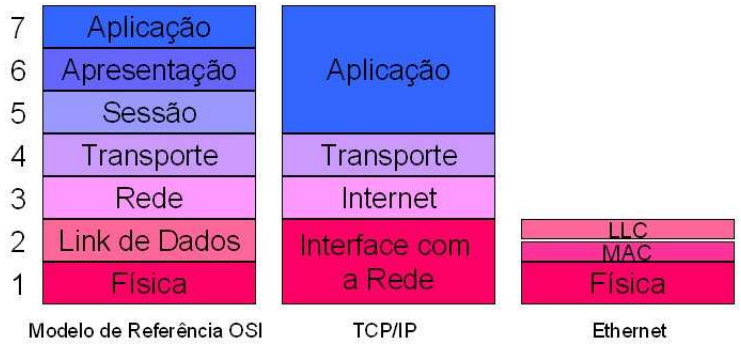

Figura 2.8: Arquitetura Ethernet

A camada LLC é a responsável por adicionar informações de que protocolo na camada Internet foi o responsável por gerar os dados. Dessa forma, durante a recepção de dados da rede esta camada no computador receptor tem que saber que protocolo da camada de Internet ele deve entregar os dados. Esta camada é definida pelo protocolo IEEE 802.2.

A camada de Controle de Acesso ao Meio (MAC) é a responsável por montar o quadro que será enviado para a rede. Esta camada é responsável por adicionar o endereço MAC de origem e de destino - lembrando que o endereço MAC é um endereço físico de uma placa de rede. Os quadros que são destinados a outras redes utilizarão o endereço MAC do roteador da rede como endereço de destino.

A camada Física é a responsável por converter o quadro gerado pela camada MAC em sinais elétricos (se for uma rede cabeada) ou eletromagnéticos (se for uma rede sem fio).

As camadas LLC e MAC adicionam suas informações de cabeçalho ao datagrama recebido da camada Internet. Uma estrutura completa de quadros gerados por essas duas camadas pode ser vista na figura 2.9.

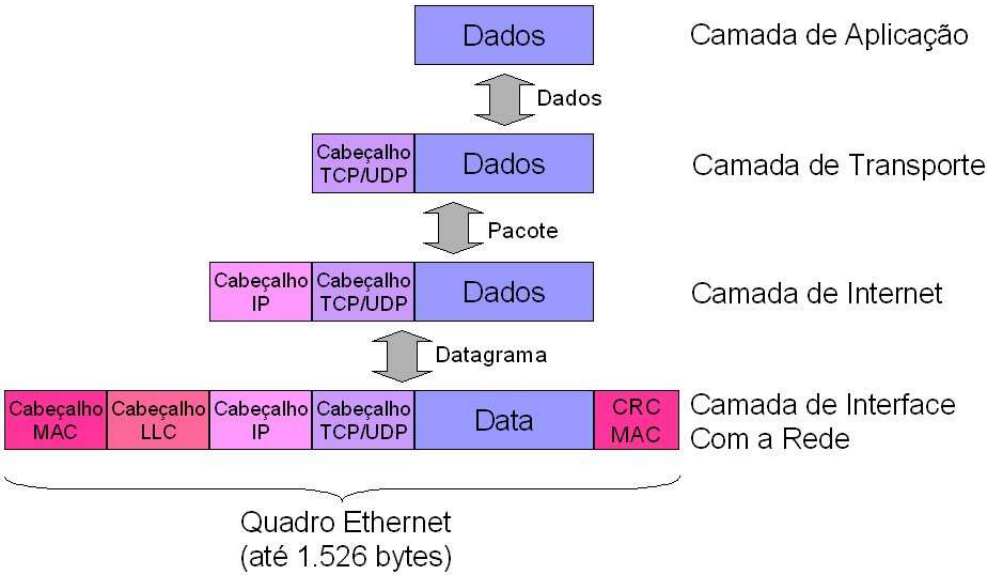

Figura 2.9: Quadro na camada de Interface com a Rede

3 *Falhas de Segurança nos Protocolos TCP/IP*

O conjunto de protocolos TCP/IP podem ser considerado como um dos conjuntos de protocolos mais importantes (se não o mais) dos dias de hoje. A maneira com que a rede internet cresceu nos últimos anos é gigantesca: passamos de pequenas redes locais para uma imensa rede global que conecta o mundo inteiro. Entretanto, este crescimento tem trazido problemas de segurança. Quando o conjunto de protocolos TCP/IP foi criado, a dimensão da internet não se comparava aos dias atuais e segurança não era um problema a se preocupar. Com o passar do tempo foram sendo descobertos problemas não necessariamente triviais de se corrigir. A seguir, serão estudados alguns dos problemas clássicos de segurança dos protocolos TCP/IP [3].

3.1 SYN Floods

SYN flood ou ataque SYN é uma forma de ataque de negação de serviço (também conhecido como Denial of Service - DoS) em sistemas computadorizados, na qual o atacante envia uma sequência de requisições SYN para um sistema-alvo visando uma sobrecarga direta na camada de transporte e indireta na camada de aplicação do modelo OSI [4].

Um cliente malicioso, que implemente intencionalmente um protocolo TCP errado e incompleto, pode não mandar esta última mensagem ACK (do three way handshake explicado acima). O servidor irá esperar por isso por um tempo, já que um simples congestionamento de rede pode ser a causa do ACK faltante.

Esta chamada conexão semi-aberta explora a boa-fé do protocolo TCP que espera por um certo tempo e algumas tentativas de restabelecimento de um sinal ACK válido para retomar a

comunicação. A resposta maliciosa ao comando SYN gerada pelo cliente pode ocupar recursos no servidor (memória e processamento) ou causar prejuízos para empresas usando softwares licenciados por conexão (aumento de conexões ”ativas”). Pode ser possível ocupar todos os recursos da máquina, com pacotes SYN. Uma vez que todos os recursos estejam ocupados, nenhuma nova conexão (legítima ou não) pode ser feita, resultando em negação de serviço. Alguns podem funcionar mal ou até mesmo travar se ficarem sem recursos desta maneira. É ilustrado na figura 3.1 um exemplo de ataque SYN flood.

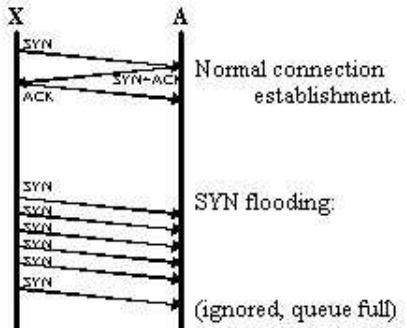

Figura 3.1: Exemplo de SYN Flood

Algumas contra-medidas para este ataque são os SYN cookies. Apenas máquinas Sun e Linux usam SYN cookies.

3.2 IP Spoofing

IP spoofing é uma técnica que consiste em mascarar (spoof) pacotes IP utilizando endereços de remetentes falsificados [4].

Devido as características do protocolo IP, o reecaminhamento de pacotes é feito com base numa premissa muito simples: o pacote deverá ir para o destinatário (endereço-destino) e não há verificação do remetente - não há validação do endereço IP com relação ao router anterior (que encaminhou o pacote). Assim, torna-se trivial falsificar o endereço de origem através de uma manipulação simples do cabeçalho IP. Assim, vários computadores podem enviar pacotes fazendo-se passar por um determinado endereço de origem, o que representa uma séria ameaça para os sistemas baseados em autenticação pelo endereço IP.

Esta técnica, utilizada com outras de mais alto nível, aproveita-se, sobretudo, da noção de confiabilidade que existe dentro das organizações: supostamente não se deveria temer uma máquina de dentro da empresa, se ela é da empresa. Por outro lado, um utilizador torna-se também confiável quando se sabe de antemão que estabeleceu uma ligação com determinado serviço. Esse utilizador torna-se interessante, do ponto de vista do atacante, se ele possuir (e estiver usando) direitos privilegiados no momento do ataque.

Existem métodos para evitar estes ataques, como a aplicação de filtros de pacotes nos gateways: faz sentido bloquear pacotes provindos da rede externa com endereços da rede local.

3.3 Sequestro de Conexão TCP

> Este tipo de ataque explora um ”estado dessincronizado”da comunicação TCP. Quando o número de sequência de um pacote recebido não é o mesmo que o número de sequência esperado, a conexão é dita dessincronizada. Dependendo do valor real do número de sequência recebido, o protocolo TCP pode ou não descartar o pacote. O protocolo TCP usa um janelas deslizantes pra permitir uma comunicação eficiente. Assim, se o pacote recebido não for o esperado, mas está dentro da janela atual, o pacote será salvo na premissa de que ele será recebido mais tarde. Se o pacote recebido está fora da janela atual, ele será descartado (CHRIS CHAMBERS, JUSTIN DOLSKE, and JAYARAMAN IYER).

Assim, quando dois hosts estão dessincronizados, eles passam a ignorar pacotes um do outro. Um atacante pode injetar pacotes forjados com os números de seqüência correta (e potencialmente modificar ou adicionar comandos para a comunicação). Obviamente, isto requer que o atacante esteja localizado no caminho de comunicação entre os dois hosts para que ele possa escutar, a fim de replicar pacotes sendo enviados. A chave para este ataque é a criação do estado dessincronizado. Existem dois modos de atuação: um é durante o handshake (aperto de mão) de três etapas do TCP e o outro é no meio de uma conexão já estabelecida. O sequestro de conexão se aproveita de um ”estado desincronizado”na comunicação TCP [4].

Pacotes ignorados acabam por gerar ACKs, ao invés de serem completamente ignorados. Quando um receptor recebe pacotes com número de sequência incorretos, ele responde um ACK com o número de sequência esperado, porém este receptor irá ignorar o pacote recebido pois ele não é o esperado. Assim, um grande número de ACKs pode ser gerado neste ataque, o que

pode ser usado para a sua detecção.

O sequestro de conexões TCP permite que atacantes vejam e alterem informações privadas. Na figura 3.2 é possível ver um exemplo onde o mediador seria o elemento X.

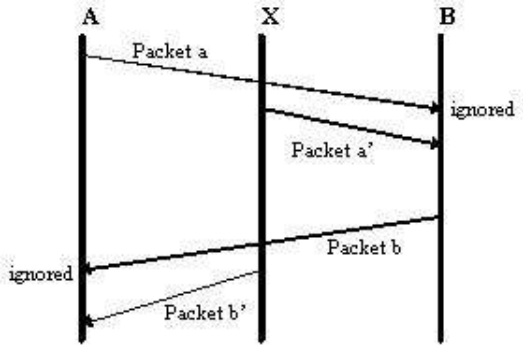

Figura 3.2: Sequestro de conexão TCP

3.4 Source Routing

Source Routing pode ser considerada uma variação de IP spoofing. Neste problema de segurança, o roteamento de origem permite que o host de origem especifique o caminho (rota) que o receptor deve usar para responder.

> Um atacante pode tirar proveito disso especificando um rota que direciona as respostas para um caminho que pode ser, por exemplo, para o próprio computador do computador ou para uma outra sub-rede. Embora simples, este ataque pode não ser tão bem-sucedido, pois roteadores são normalmente configurados para descartar pacotes com roteamento de origem habilitado (CHRIS CHAMBERS, JUSTIN DOLSKE, and JAYARAMAN IYER).

Na figura 3.3 é possível ver um exemplo de como o destino do pacote é alterado.

3.5 Ataques ICMP

Atacantes podem forjar pacotes ICMP, alterando a tabela de roteamento do host ou fazer um ataque pacotes de do tipo DoS, forjando ”destino inalcançável”ou ”time to live excedido”. A solução para isso é fazer com que os pacotes ICMP possam ser filtrados pela filtragem de pacotes.

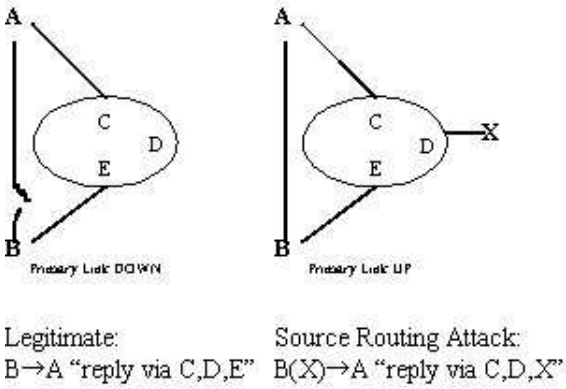

Figura 3.3: Exemplo de Source Routing

4 *Mecanismos de Defesa e Proteção*

No que concerne diretamente ao TCP/IP (em todas as suas camadas), pode-se dizer que a segurança gira sempre em torno de dois eixos básicos, que são o isolamento físico e o suporte a criptografia nos protocolos.

> Assim, quando pensamos em utilizar um switch que impeça o broadcast de pacotes ethernet para todas as interfaces da LAN a fim de precaver-nos contra sniffers, estamos isolando. Quando implantamos um firewall para bloquear as portas ou os IPs que não oferecem serviços à Internet, estamos isolando. Quando dividimos a rede privada em duas metades, uma exclusiva para uso interno e outra mista, com máquinas que oferecem serviços para a rede interna e para a Internet, estamos isolando. Quando bloqueamos a entrada de emails com attachments para evitar a entrada de viruses que exploram debilidades de segurança de alguns clientes de correio eletrônico, estamos isolando. Quando dividimos os serviços por várias máquinas a fim de não somarmos as debilidades de segurança de todos eles num único ponto, estamos isolando. Por outro lado, quando utilizamos um servidor web seguro no lugar de um não seguro num site de comércio eletrônico, estamos criptografando. Quando substituimos o telnet pelo ssh como protocolo para abrir sessões remotas, ou o ftp pelo ssh, estamos criptografando, assim como quando implantamos uma VPN através de um túnel TCP com criptografação nas duas pontas. A criptografação prescinde do isolamento físico, e opta por tornar inútil a captura da informação (DOUGLAS E. COMER).

São especializações particularmente importantes para a área de segurança a filtragem de pacotes e todas as formas de autenticação. É também pertinente às questões de segurança os protocolos para sincronização de relógios, como o NTP, para o qual existe muitos servidores na Internet. Sem eles, torna-se complicado rastrear eventos ao longo de várias máquinas, pois os logs que elas geram apresentarão timestamps dessincronizados.

4.1 Filtragem de Pacotes

A filtragem de pacotes consiste em aplicar ao roteamento de pacotes regras de descarte que impeçam a entrada ou a saída de pacotes dirigidos ou provenientes de determinados endereços ou portas. O roteamento IP baseia-se na aplicação de regras aos dados de cabeçalho de cada pacote IP. Bem, a filtragem consiste em adicionar regras que ao invés de servirem para escolher a interface de envio de pacote, prestam-se a determinar se um pacote será efetivamente roteado ou meramente descartado. Note que no momento em que o pacote vai ser roteado, ele é um buffer na memória da máquina. Descartar esse pacote significa meramente liberar esse buffer para ser reutilizado.

A filtragem de pacotes pode ser realizada por um equipamento especializado (um ”firewall”), pelo roteador que opera como gateway da rede corporativa com a Internet, ou por alguma máquina intermediária. A filtragem de pacotes frequentemente é feita pela mesma máquina que implementa a tradução de endereços para possibilitar às máquinas da rede interna o acesso aos serviços da Internet.

4.2 Autenticação

De modo geral, autenticar consiste em provar a identidade. Assim, quando nos identificamos perante um servidor dizendo que somos o usuário fulano, o servidor espera que provemos isso mostrando que conhecemos um segredo que apenas o usuário fulano conhece (uma senha). Essa é a forma mais comum de autenticação, e está presente em vários protocolos TCP, como por exemplo o FTP, o POP e o TELNET.

Em muitos casos, a autenticação faz uso de uma base de dados centralizada que possui a tabela de todos os usuários e as suas senhas, ou então ”assinaturas”das suas senhas. Neste caso, faz-se necessário existir um protocolo de autenticação que defina a forma da comunicação entre a máquina que está autenticando o usuário e a máquina que contém a base de dados de autenticação. Esses protocolos estão implementados na forma de serviços baseados em TCP/IP: o NIS, que centraliza as informações de login em redes Unix, o RADIUS, utilizado para

centralizar informações de autenticação PPP em provedores de acesso e também os serviços de autenticação próprios do compartilhamento de recursos do Windows, que podem operar sobre TCP/IP.

Uma autenticação bem-sucedida provoca a concessão de privilégios para aquele que autenticou-se. Esse privilégio pode consistir na capacidade de ler uma caixa postal (no caso do protocolo POP), ou de rotear pacotes através do provedor de acesso (no caso do PPP), ou de abrir uma sessão de comandos num computador remoto (no caso do TELNET ou do SSH), ou de fazer upload ou download de arquivos (no caso do FTP).

Assim, é fácil entender a relação direta da autenticação nas suas diversas formas com a segurança de redes. A obtenção de privilégios pode ser um primeiro passo para uma ação criminosa, e por isso a autenticação deve estar cercada por muitos cuidados na definição dos protocolos.

4.3 Firewalls

Um outro elemento importante para a segurança são os firewalls. Eles previnem a rede local contra os perigos da rede externa. Um firewall frequentemente é instalado no ponto que a rede local é conectada a Internet. Todo o tráfego que entra ou sai para Internet passa pelo firewall.

Graças a isso ,o firewall controla todo o fluxo entre a rede local e a Internet e assim pode conferir se o tráfego é aceitável. Quanto a um tráfego ser aceitável ou não isto depende da segurança configurada para esta rede.

Logicamente um firewall é um separador, um analizador. A implementação física de um firewall varia muito. O mais comum é um firewall ser constituído por um conjunto de componentes de hardware - um roteador, um computador ou a uma combinação de roteadores, computadores e redes com softwares apropriados. Existe diversas maneiras de configurar estes equipamentos, esta configuração depende da segurança que quer ser dada para a rede.

Por ser um concentrador de todo o tráfego da rede local para a Internet é nele que estão colocadas todas as medidas de segurança. É mais eficiente e econômico colocar todas as medidas

de segurança e tecnologias em apenas um local da rede do que tê-las espalhadas pela rede.

Muitos dos serviços que a Internet oferece são inerentemente inseguros. O firewall obriga a segurança no site, permitindo somente serviços aprovados passar através dele, os serviços que tinham suas regras setadas.

4.4 Ferramentas para Verificação de Segurança no Unix

Vamos apresentar algumas ferramentas usadas para a verificação de segurança no Unix.

4.4.1 COPS

O COPS é uma ferramenta de segurança para ambientes Unix, ele faz uma série de verificações como determinar se arquivos importantes estão em modo inadequado, verificar as senhas fáceis, etc [5].

4.4.2 TRIPWIRE

O objetivo do TRIPWIRE é detectar que já houve algum tipo de intrusão no sistema. Ele é um analizador de integridade de arquivos e diretórios, um utilitário que compara um conjunto de arquivos com informações previamente armazenadas numa base de dados. Qualquer diferença é detectada e armazenada num log, incluindo arquivos que foram deletados ou criados. Geralmente, ele é executado do cron de tempos em tempos, e permite que se conclua, com um bom grau de confiança, que os arquivos binários principais não foram alterados [6].

4.4.3 SATAN

Satan (Security Analysis Tool for Auditing Network) é o mais complexo sistema de auditoria para sistemas Unix disponível. Ele coleta a maior quantidade possível de informações sobre um host, examina os serviços de rede como o NFS, o ftp, rexd entre outros [7].

> As informações que são relatadas pelo SATAN incluem tanto os tipos de serviços disponibilizados pelo host, quanto furos potenciais nestes serviços. Estes furos geralmente são causados por erros de configuração ou bugs conhecidos

> dos diversos daemons. O SATAN pode também ser utilizado para verificar a topologia de uma rede, serviços oferecidos, tipos de hardware e software, etc (JUERGEN HAAS).

4.4.4 TIGER

Muito parecido com o SATAN, o TIGER, é um conjunto de Bourne Shell (bash) scripts, programas em C e arquivos de dados que visam fazer uma auditoria de segurança num sistema Unix [8].

Sua maior diferença em relação ao SATAN está no fato de que a ênfase do TIGER esta em erros de permissões de arquivos, e não na área de serviços de rede, que é o caso do SATAN.

> Basicamente, quando é executado, TIGER procura por um arquivo de configuração (geralmente .tigerrc) que irá limitar ou ampliar a quantidade de testes executados dependendo do desejo do usuário. Ele então irá executar uma série de scripts e retornará as falhas de segurança encontradas (JAVIER FERNANDEZ-SANGUINO).

5 *Conclusão*

Há uma série de estudos e pesquisas sendo realizados a respeito das vulnerabilidades dos protocolos TCP/IP. O conjunto de protocolos TCP/IP possui uma série de falhas e não existe uma solução trivial para isso, já que para cada falha/vulnerabilidade existe uma solução, isso quando a solução também não apresentar uma vulnerabilidade.

Neste caso, um conjunto composto de conhecimentos passa a ser necessário para se obter segurança dentro dos protocolos TCP/IP. Conhecer ferramentas de segurança (além de um profissional que saiba como administrá-las nos mínimos detalhes), conhecimento a respeito de possíveis novas falhas descobertas, criptografia, além de um firewall devidamente configurado. Este ítens bem trabalhados e de maneira combinada visam aumentar a segurança, porém não garantem 100% de defesa, pois como já descrito anteriormente, uma solução pode abrir brechas para novas vulnerabilidades.

Isso significa que esta é uma área que precisa de muito estudo ainda, pois não existe uma solução única e trivial, carecendo de investimentos e atenção.

Referências Bibliográficas

[1] COMER, D. E. *Internetworking with TCP/IP*. [S.l.]: Hardcover. Introdução aos conceitos de TCP/IP.

[2] HUNT, C. *TCP/IP*. [S.l.]: TCP/IP Network Administration, 1998. O Reilly.

[3] MORIMOTO, C. E. *Guia de Redes*. [S.l.]: Sul Editores. Conceitos de Redes.

[4] CHAMBERS, C. *TCP/IP Security*. [S.l.]: http://www.linuxsecurity.com/resource_files/documentation/t security.html. Acessado em Junho de 2011.

[5] LAWRENCE, A. *Security: COPS*. [S.l.]: http://aplawrence.com/Security/cops.html. Acessado em Junho de 2011.

[6] SOFTWARE, S. O. do. *Tripwire*. [S.l.]: http://www.tripwire.com/. Acessado em Junho de 2011.

[7] HAAS, J. *SATAN (Security Administrator Tool for Analyzing Networks)*. [S.l.]: http://linux.about.com/cs/linux101/g/SATAN__Security.htm. Acessado em Junho de 2011.

[8] FERNANDEZ-SANGUINO, J. *Tiger: The Unix security audit and intrusion detection tool*. [S.l.]: http://www.nongnu.org/tiger/. Acessado em Junho de 2011.

Ministério da Educação
Universidade Tecnológica Federal do Paraná
Diretoria de Pesquisa e Pós-Graduação do *Campus* Curitiba
Departamento Acadêmico de Eletrônica
Especialização em Teleinformática e Redes de Computadores

TERMO DE APROVAÇÃO

Título da Monografia (Conjunto de Protocolos TCP/IP e suas falhas)

por

Issam Ibrahim

Esta dissertação foi apresentada às 16 horas do dia 15 de Junho de 2011 como requisito parcial para a obtenção do título de ESPECIALISTA EM TELEINFORMÁTICA E REDES DE COMPUTADORES, Universidade Tecnológica Federal do Paraná. O candidato foi argüido pela Banca Examinadora composta pelos professores abaixo assinados. Após deliberação, a Banca Examinadora considerou o trabalho aprovado.

NOTA: NOVE INTEIROS E CINCO DECIMOS
(9,5)

Prof. Lincoln Herbert
(UTFPR)

Prof. Walter Godoy
(UTFPR)

Visto da Coordenação

Prof. Dr. Walter Godoy Júnior
Coordenador do Curso