Segurança Operacional

Análise de Tráfego

ITnerante

Gustavo Vilar

- Mini – CV
  - PPF / DPF – Papiloscopista Policial Federal
  - Pós-Graduado em Docência do Ensino Superior – UFRJ
  - Graduado em Ciência da Computação e Processamento de Dados – ASPER/PB
  - Aprovações: PRF 2002, PF 2004, MPU 2010, ABIN 2010, PCF-PF 2013

ITnerante

Gustavo Vilar

- Contatos:

http://www.itnerante.com.br/profile/GustavoPintoVilar
http://www.provasdeti.com.br/index.php/por-professor/gustavo-vilar.html

gustavopintovilar@gmail.com
p3r1t0f3d3r4l@yahoo.com.br

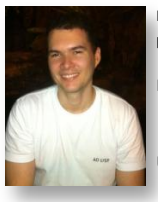
ITnerante

Escopo

- Abordar os assuntos mais recorrentes e com fortes tendências para concursos atuais
- Familiarizar o concursando com os tipos de questões mais frequentes.
- Abordar as metodologias de resolução de questões das principais bancas

ITnerante

Agradecimentos Especiais



Paulo Marcelo
paulo1410@gmail.com

Mini - CV
- Atualmente Analista de Redes Sênior da Infraero
- Pós-Graduado em Administração de Empresas - UNIFOR
- Graduado em Tecnologia da Informação - IFCE
- Concursos que assumiu: Dataprev 2009, Infraero 2011

ITnerante

Agradecimentos Especiais

Rafael Eduardo Barão
rafbarao@yahoo.com.br

Mini - CV
- Administrador de Redes do Poder Legislativo da cidade de Guarulhos –SP.
- Formado em 2010 no curso de Ciência da Computação pela Universidade de Sorocaba – UNISO.
- Principais aprovações:
  - PF 2013 – Perito Criminal Federal / SERPRO 2013 – Analista de Suporte / ANP 2012 – Analista Administrativo / CNJ 2012 – Analista Judiciário / CMG 2012 – Administrador de Redes / PCDF 2012 – Perito Criminal (Excedente) / DATAPREV 2008 – Analista de TI (Banco de Dados).

ITnerante

Bibliografia / Ferramentas

RFCs - IETF

FUNDAÇÃO CESGRANRIO

CespeUnB

Fundação Carlos Chagas

ITnerante

Análise de Tráfego – Carga Horária

- **22 vídeo aulas (07h30m00s / 00h20m30s)**
  - Conceitos e fundamentos
  - A transmissão da Informação
  - O quadro Ethernet
  - Composição dos cabeçalhos dos protocolos
    - IPv4
    - IPv6
    - ICMP
    - IGMP
    - UDP
    - SCTP
    - TCP
  - Ferramentas de captura e análise de tráfego
  - Desmontagem de 10 questões de concursos públicos

ITnerante

Segurança Operacional

Análise de Tráfego

ITnerante

Motivações para compreender a análise de tráfego

- Encontrar problemas numa rede de computadores
- Identificar hardware defeituoso
- Detectar intrusões e assinaturas de atividades maliciosas
- Análise de comportamento e desempenho da rede
- Análise de aplicações que geram ou recebem tráfego
- Análise da infraestrutura
- Análise do roteamento
- Preparação para concursos públicos

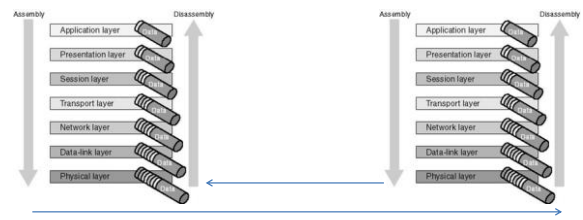
ITnerante

Análise de Tráfego - Conceitos

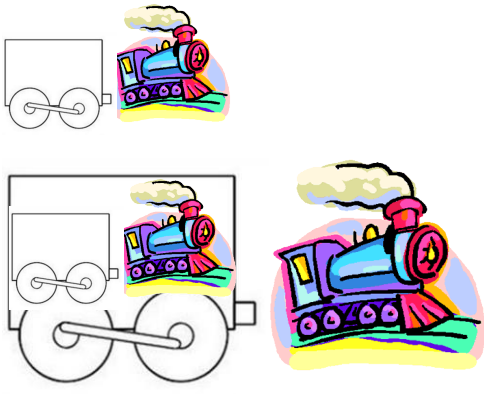
- Sniffing
  - Uso da interface de rede de uma máquina para receber o tráfego de rede que passa por ela, incluindo o tráfego não direcionado àquela máquina
- Modo promíscuo
  - Interface "escuta" o meio e captura todos pacotes
  - Rede compartilhada = captura de todo o tráfego
  - Rede segmentada = captura do tráfego local

ITnerante

Raio X da transmissão

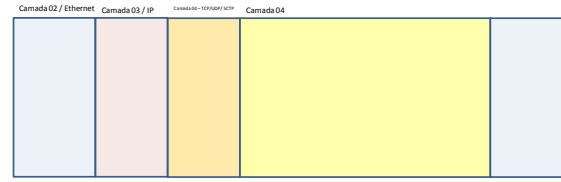


ITnerante



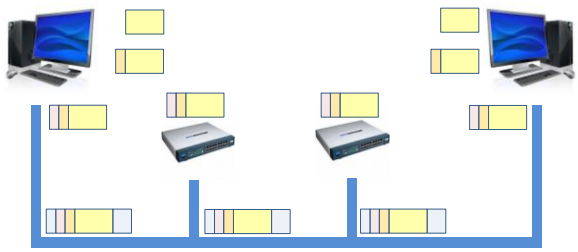
ITnerante

A composição dos cabeçalhos – Arq TCP/IP



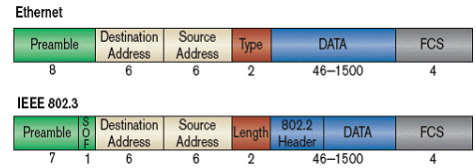
ITnerante

A dinâmica da transmissão



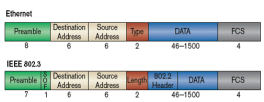
ITnerante

O quadro para transporte dos dados



ITnerante

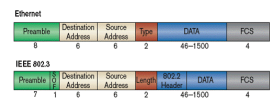
Ethernet x 802.3



- O padrão ethernet surgiu na década de 1970 nos laboratórios da Xerox - DIX
- 1980 o IEEE passou a administrar o padrão - IEEE 802.3
- O Ethernet é um conjunto de protocolos que lida com as camadas 1 e 2 do modelo de referência OSI
  - O Ethernet se preocupa com o aspecto físico da transmissão

ITnerante

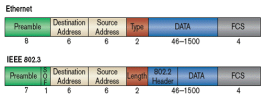
Ethernet x 802.3



- Frames Ethernet são "envelopes" para os pacotes TCP/IP
- Informações das camadas superiores serão transportados dentro do campo de dados (ou conteúdo)
- O que é cabeçalho + dados para a camada de cima é apenas dados para a camada de baixo
- 1500 bytes de payload = vários cabeçalhos + dados aplicação

ITnerante

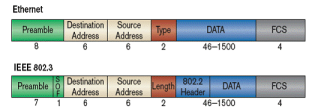
Ethernet x 802.3



- Preamble: sequência de bytes para sincronizar comunicação
- SOF – Start of frame: delimitador
- Destination Address: contém o endereço MAC do destinatário;
- Source Address: contém o endereço MAC do remetente;
- Type/Length: indica o tamanho em Bytes do campo de dados;
- Data: contem os dados que deverão ser passados a próxima camada, deve ter tamanho mínimo de 46 bytes e máximo de 1500 bytes;
- FCS – Frame Check Sequence: contém o Cyclic Redundancy Check (CRC).
  - Ele realiza a detecção de erros, não a correção

ITnerante

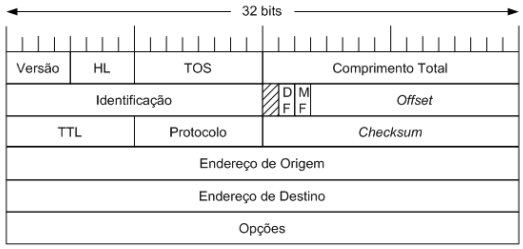
Ethernet x 802.3



- Redução do preâmbulo para 7 bytes
  - Usou o último byte como delimitador de INÍCIO DE QUADRO
- Transformação do campo tipo em comprimento
  - > 1500 = tipo
    - Interpretação do Ethernet
  - <= 1500 = tamanho
    - Interpretação do IEEE

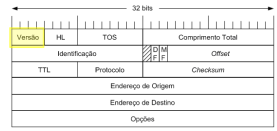
ITnerante

O cabeçalho IPv4



ITnerante

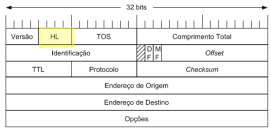
O cabeçalho IPv4– Campo Versão



- Version
  - Lido antes do processamento do datagrama
- IPv4
  - 0100
  - 4 decimal
- IPv6
  - 0110
  - 6 decimal

ITnerante

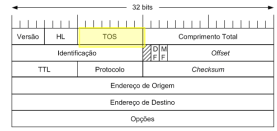
O cabeçalho IPv4– Campo HL



- Hlen
  - valor x 4 bytes. Na prática de 20 a 24 bytes.
  - palavras de 32 bits
  - Na teoria - 60 bytes
    - 2^4, pois o campo tem 4 bits
    - Cabeçalho = 20 (60) bytes + parte opcional de tamanho variável (40 bytes).

ITnerante

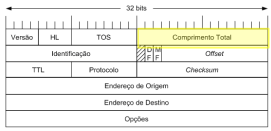
O cabeçalho IPv4– Campo TOS



- ToS
  - Diffserv
  - 6 bits iniciais - CODEPOINT
    - Indicam a QoS desejada
    - Differentiated Services Code Point (DSCP)
  - 2 bits FINAIS são usados para aviso explicito de congestionamento

ITnerante

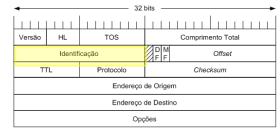
O cabeçalho IPv4– Campo Total Length



- Total Length
  - Pacote completo - 16 bits - 65535 octetos

ITnerante

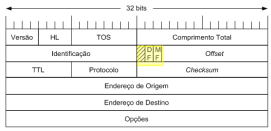
O cabeçalho IPv4– Campo ID



- Identificação
  - Id do datagrama
  - 16 bits
  - Repetido nos fragmentos

ITnerante

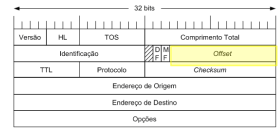
O cabeçalho IPv4– Campos de fragmentação



- Flags
  - Existe um bit não usado
  - DF
    - Roteador pode descartar
    - 1 está em uso
  - MF
    - Ultimo fragmento não possui esta marcação
    - Todos fragmentos possuem, exceto o último
    - 1 está em uso

ITnerante

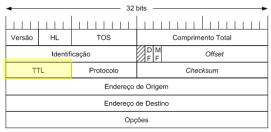
O cabeçalho IPv4– Campos de fragmentação



- Offset
  - Posição do fragmento no datagrama original
  - 13 bits
  - Múltiplo de 8 bytes - desconsiderar na análise de tráfego

ITnerante

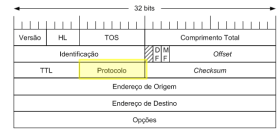
O cabeçalho IPv4– Campo TTL



- TTL - número máx de hops
  - Roteador que decrementa para 0 descarta o pacote e envia mensagem de erro à origem
  - Decrementa 1 unidade a cada segundo no roteador
  - 37 é um número aproximado de saltos para rodar a internet toda
  - CESPE considera que o TTL está associado ao tempo

ITnerante

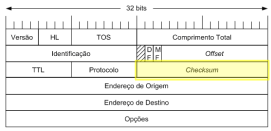
O cabeçalho IPv4– Campo Protocol



- Protocol: 8 bits
  - 6 - TCP
  - 17 - UDP
  - 132 - SCTP
  - 1 - ICMP
  - 2 - IGMP
  - 50 - ESP (IPSec)
  - 51 - AH (IPSec)

ITnerante

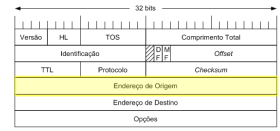
O cabeçalho IPv4– Campo Checksum



- Header checksum
  - Contém 0 para fins de cálculo de checksum, assim como os outros campos variáveis
  - válido somente para o cabeçalho
  - TTL, ToS, Checksum e flags são considerados preenchidos com 0 para efeitos de cálculo de checksum

ITnerante

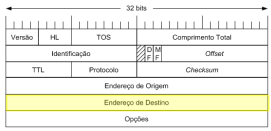
O cabeçalho IPv4– Campos de endereço



- Source IP address
  - Endereço IPV4 do remetente
  - 32 bits
  - Diferentemente do endereço físico, não se modifica durante o trajeto

ITnerante

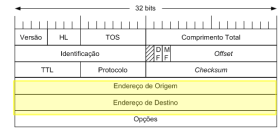
O cabeçalho IPv4– Campos de endereço



- Destination IP address
  - Endereço IPV4 do remetente
  - 32 bits
  - Diferentemente do endereço físico, não se modifica durante o trajeto

ITnerante

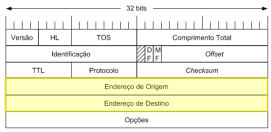
O cabeçalho IPv4– Ainda sobre endereços



- Blocos CIDR reservados para redes privadas
  - Todos 10.x.x.x
    - 8 bits rede
    - 24 bits hosts
  - 172.16.0.0 a 172.31.255.255
    - 12 bits rede
    - 20 bits hosts
  - 192.168.0.0 a 192.168.255.255
    - 16 bits rede
    - 16 bits host
- APIPA - Automatic Private Internet Protocol Addressing
  - 169.254.0.0/16 a 169.254.255.254/16
  - Microsoft
  - Não elencado na RFC 1918

ITnerante

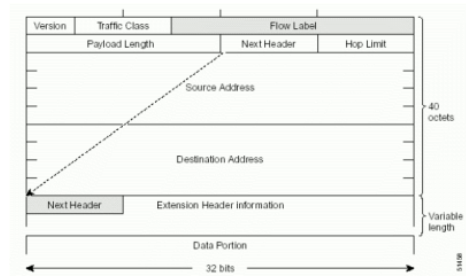
O cabeçalho IPv4– Ainda sobre endereços



- Endereços especiais
  - Tudo 0 - Endereço de origem inicial
  - Tudo 1 - broadcast Limitado
  - Rede + Tudo 0 - Endereço da rede
  - Rede + Tudo 1 - broadcast direcionado
  - 127.x.x.x - loopback

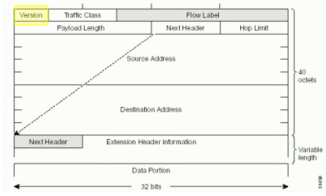
ITnerante

O cabeçalho IPv6



ITnerante

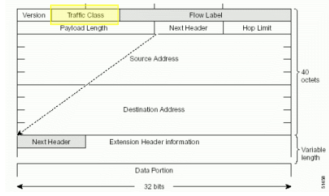
O cabeçalho IPv6



- Version (4 bits)
  - Versão do IP utilizada. No caso no IPv6, este campo vale 0110.
  - Segundo algumas pessoas, esse campo seria desnecessário e somente rouba ciclos de instruções. O campo Type do frame ethernet já serviria para identificar a carga o frame como IPv6

ITnerante

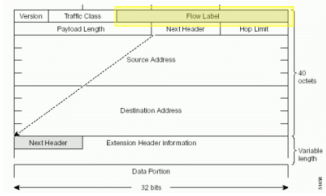
O cabeçalho IPv6



- Traffic Class Priority (8 bits) - Indica a prioridade com a qual o pacote deve ser tratado. Também referenciado como Serviços Diferenciados
  - 0 a 7 para aqueles que podem sofrer atraso
  - 8 a 15 para aqueles com tráfego em tempo real
  - QoS na camada de rede
  - 2 bits menos significativos não são usados para esse fim, mas para aviso explícito de congestionamento

ITnerante

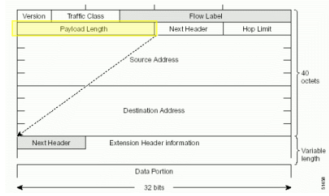
O cabeçalho IPv6



- Flow label (20 bits)
  - Identifica, juntamente com os campos Source Address e Destination Address, o fluxo ao qual o pacote pertence.
  - Valor 0 indica que não será usado
  - Valor <> 0 indica que os roteadores precisam dar tratamento ao fluxo. Trata-se de uma aproximação do conceito de circuitos virtuais

ITnerante

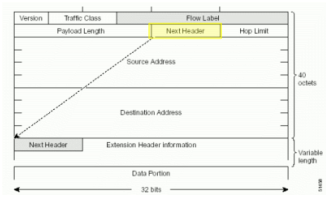
O cabeçalho IPv6



- Payload Length (16 bits)
  - Tamanho, em octetos, do restante do pacote, após o cabeçalho.
    - Somente os dados
    - Cabeçalhos de extensão possuem campo que identifica seu tamanho (Header Extension Length)
  - Como o cabeçalho deixou de ser contado, podemos ter uma carga de 65535 bytes em vez de apenas 65515 como no IPv4
  - No cabeçalho base IPv6 não existe o campo HEADER LENGTH, justamente por que ele tem tamanho fixo

ITnerante

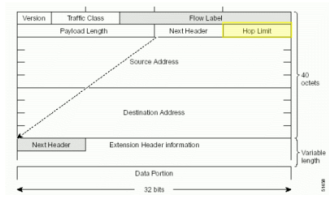
O cabeçalho IPv6



- Next Header (8 bits)
  - Indica o tipo do possível cabeçalho de extensão que segue o cabeçalho IPv6. Caso não esteja se utilizando cabeçalho de extensão, este campo indica a qual protocolo de transporte/rede o pacote deve ser repassado.
  - Essa informação pode ser o protocolo usado na camada de transporte, UDP – User Datagram Protocol ou TCP – Transmission Control Protocol, ou um cabeçalho de extensão
  - Último cabeçalho coloca em seu interior o número tradicional que indica o protocolo de entrega do pacote: 6 TCP, 17 UDP, 132 SCTP, etc...

ITnerante

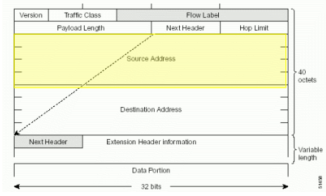
O cabeçalho IPv6



- Hop Limit (8 bits)
  - Número máximo de roteamentos que o pacote pode sofrer. O valor deste campo é decrementado a cada roteamento. Quando seu valor chega a zero o pacote é descartado. Similar ao campo Time to live do IPv4.

ITnerante

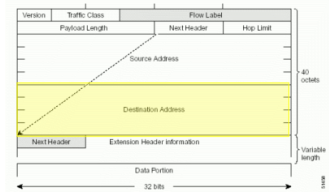
O cabeçalho IPv6



- Source Address (128 bits)
  - Endereço do remetente.
  - 8 quartetos em hexa
  - Zeros à esquerda podem ser suprimidos, zeros à direita JAMAIS
  - Grupos de bits zeros ou mais podem ser substituídos, somente uma única vez, por ::
    - 0123:0221:3456:3643:0988:0987:9088:9900
    - 123:221:3456:3643:988:987:9088:9900
  - Somente uma única vez

ITnerante

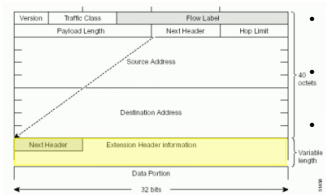
O cabeçalho IPv6



- Destination Address (128 bits)
  - Endereço do remetente.
  - 8 quartetos em hexa
  - Zeros à esquerda podem ser suprimidos, zeros à direita JAMAIS
  - Grupos de bits zeros ou mais podem ser substituídos, somente uma única vez, por ::
    - 0123:0221:3456:3643:0988:0987:9088:9900
    - 123:221:3456:3643:988:987:9088:9900
  - Somente uma única vez

ITnerante

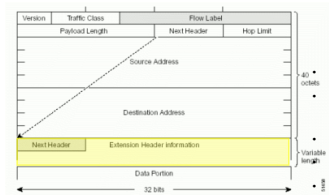
O cabeçalho IPv6



- Possibilidade de utilização de múltiplos cabeçalhos encadeados
- Estes cabeçalhos, caso sejam utilizados, devem aparecer entre o cabeçalho IPv6 e o bloco de dados
- Podem ser sempre adicionados novos cabeçalhos para satisfazer novas especificações
- Emissor deve mandar os cabeçalhos numa sequencia pré-definida, entretanto o receptor deve estar apto a tratá-los em qualquer ordem
- De acordo com a necessidade, novos tipos de cabeçalho de extensão podem ser criados (evolução do protocolo)

ITnerante

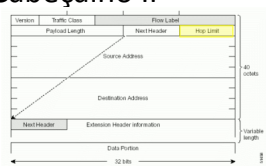
O cabeçalho IPv6



- Tipos (sequência) TODOS OPCIONAIS
  - Se forem usados, devem aparecer PREFERENCIALMENTE nessa ordem, logo após o cabeçalho fixo
  - Caso não haja nenhum cabeçalho de extensão, o cabeçalho base é diretamente seguido pela área de dados
  - Alguns possuem tamanho fixo e outros tamanhos variáveis
- Hop-by-Hop options - informações gerais para os roteadores;
- Routing - rota completa ou parcial a ser seguida;
  - usado pela origem para listar um ou mais nós intermediários que devem ser visitados até o pacote chegar ao destino
- Fragmentation - gerenciamento de fragmentos de datagrama;
- Authentication - verificação da identidade do remetente
- Encrypted security payload - informação sobre a criptografia
- Destination options - informação adicional sobre o destinatário

ITnerante

Identificadores importantes na análise de tráfego – Cabeçalho IP



- Free Net/Open BSD/MacOS X/ Impressoras HP com interface de rede: TTL 255
- Windows (98 ult. Versões /NT/2K/VISTA/7/8): TTL 128
- Linux: TTL 64
- Windows ME e anteriores: 32
- Roteadores CISCO: 255
- Roteadores Ciclades: 28
- OBS: TTL vai decrementando até chegar no ponto de captura. O S.O. é deduzido pela contagem de decrementos, pois o TTL só apresenta os valores padrão quando a captura é feita antes de ultrapassar o primeiro roteador

ITnerante

O processo de fragmentação – IPv4

- Fragmentação
  - É o processo no qual um pacote IP é fragmentado em unidades de menor tamanho para se acomodar a um menor MTU de rede.
  - No IPV4, a fragmentação ocorre dentro da infraestrutura da rede, ou seja, os roteadores a executam.
  - Apesar de existir o conceito de fragmentação transparente, na prática, essa remontagem é feita no destinatário.
  - A cada fragmento é acrescentado um novo cabeçalho IP, ou seja, ele aumenta de tamanho.
  - Cada fragmento possui não somente um identificador que o relaciona com o pacote original (packet id), mas um posicionador que diz em qual ponto do pacote original se encaixa o fragmento (offset)

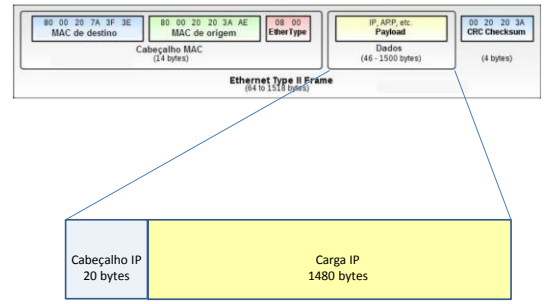
ITnerante

O processo de fragmentação – Ipv6

- Fragmentação
  - No IPv6 o responsável pela fragmentação é o host que envia o datagrama, e não os roteadores intermediários como no caso do IPv4
    - Fragmentação fim-a-fim
    - Redução de overhead nos roteadores
    - Rotas não podem ser alteradas tão facilmente
    - Inclusão de nova mensagem de erro ICMP: Descoberta do MTU na rede
    - No IPv6, os roteadores intermédios descartam os datagramas maiores que o MTU da rede
    - O MTU será o MTU mínimo suportado pelas diferentes redes entre a origem e o destino
    - Para isso o host envia pacotes ICMP de vários tamanhos; quando um pacote chega ao host destino, todos os dados a serem transmitidos são fragmentados no tamanho deste pacote que alcançou o destino
    - O processo de descoberta do MTU tem que ser dinâmico, porque o percurso pode ser alterado durante a transmissão dos datagramas
    - A informação de fragmentação é guardada num cabeçalho de extensão separado
    - Cada fragmento é iniciado por uma componente não fragmentável seguida de um cabeçalho do fragmento

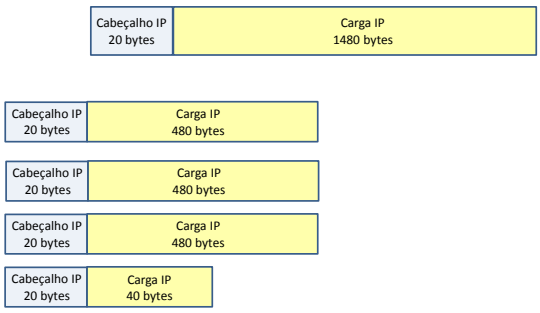
ITnerante

O processo de fragmentação



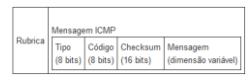
ITnerante

O processo de fragmentação



ITnerante

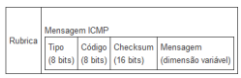
Protocolo ICMP - (RFC 792)



- Permite que roteadores enviem mensagens de erro ou controle para outros roteadores ou hosts (origem)
  - Utiliza o IP para transporte da mensagem
  - Não existe mensagem ICMP para erros de datagramas que transportam ICMP
  - Aparece quando há:
    - Impossibilidade de roteamento
    - Congestionamento na rede
    - Destino pode ser inalcançável por vários motivos:
      - Rede ou host inalcançáveis
      - Porta inalcançável
      - Rede ou host desconhecidos

ITnerante

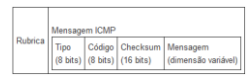
Protocolo ICMP - (RFC 792)



- Mensagens (principais)
  - Source Quench
  - Extinção de origem
    - retardamento das taxas de transmissão
    - tarefa deixada para camadas superiores
  - Time exceeded
    - TTL atingiu 0
    - O ICMP envia mensagem de erro correspondente a falha de remontagem de fragmentos
    - Nem todos fragmentos foram recebidos para remontagem
    - Só gera mensagem DE ERRO pela manipulação do fragmento inicial, evitando inundação
    - Sem o fragmento 0 não há mensagem de erro

ITnerante

Protocolo ICMP - (RFC 792)



- Mensagens (principais, continuação...)
  - Destination unreacheable
    - Não foi possível rotear ou entregar o datagrama
  - Redirect
    - Redirecionamento
    - host mudar tabela de roteamento
  - Echo request / reply
    - Ping
    - Envio de um pacote ping grande é útil para testar fragmentação e remontagem
  - Serviços que usam ICMP
    - Ping
    - Traceroute
      - Detecção de pontos intermediários

ITnerante

Protocolo IGMP

- Utilizado para Multicast
- Parte integrante do protocolo IP
- Mensagens são encapsuladas nos datagramas IP

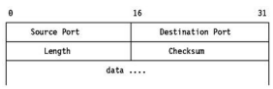
ITnerante

Protocolo IGMP

- Três tipos de mensagem
  - Host Membership Query – enviado pelo roteador para descobrir hosts e grupos
  - Host Membership Report – resposta do Host
  - Leave Group - host deixa o grupo
    - opcional
    - Ou não responder à mensagem HMQ
  - Roteador mantém listas com membros do multicast em suas tabelas

ITnerante

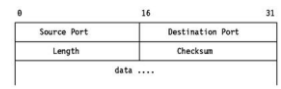
Cabeçalho UDP



- User Datagram Protocol
- Cabeçalho de 8 bytes
- Porta de origem usada para respostas, opcional
- Inclui os endereços de origem/destino do cabeçalho IP no cálculo do Checksum
- Checksum opcional
  - 0 se não for calculado
  - Se o valor é 0, armazena-se 16 bits = 1
- Não possui
  - Controle de erro
  - Controle de fluxo
  - Retransmissão
  - Confiabilidade
  - Orientação a conexão

ITnerante

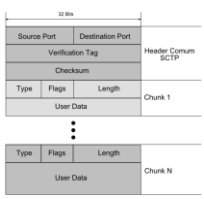
Cabeçalho UDP



- Especialmente útil em aplicações cliente/servidor
- Podem ser perdidos, duplicados ou fora de ordem
- Interações cliente/servidor e multimídia
- Usado principalmente em multimídia em tempo real
  - RTP (multiplexar diversos fluxos de dados de tempo real sobre um único fluxo de pacotes UDP, não tem controle de fluxo). RDP admite multidifusão e unidifusão.

ITnerante

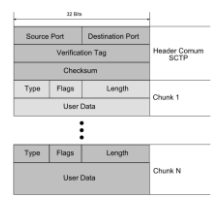
Cabeçalho SCTP



- Stream Control Transmission Protocol
- Cabeçalho de 12 bytes
- Foi desenvolvido inicialmente para transportar SS7 sobre IP
- Orientado a mensagens como o UDP e assegura transporte confiável, ordenado e controle de fluxo como o TCP
- Envia mensagens e informação de controle em diferentes Chunks
  - Cada um com seu próprio cabeçalho

ITnerante

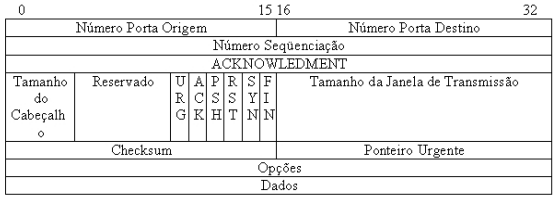
Cabeçalho SCTP



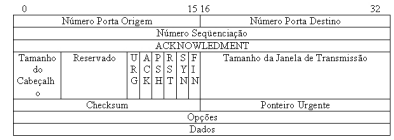
- Limitações do TCP que justificaram a criação de um protocolo intermediário
  - TCP é bastante usado para a transferência confiável de dados sobre redes IP
  - Não é adequado para comunicações de tempo real
  - Número crescente de aplicações têm implementado seus próprios mecanismos para a transferência confiável de dados sobre o UDP
  - Enquanto que o TCP acopla a transferência confiável com a ordenação estrita da entrega dos dados, o SCTP separa uma da outra (multi - streaming)

ITnerante

Cabeçalho TCP

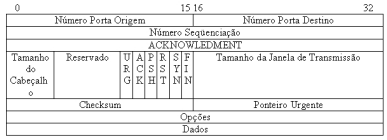


Cabeçalho TCP



- Source port
- Destination Port
- Sequence number
  - Primeiro byte que compõe o segmento
- Acknowledgement
  - Próximo byte
- Hlen
  - Unidades de 4 bytes
  - Limitado a 60 bytes: 2^4
- Reserved
  - 6 bits

Cabeçalho TCP



- Code bits ou flags
  - **Urg**: Prioridade sobre os outros pacotes não urgentes. Trabalha em conjunto com o ponteiro de urgente.
  - Sinais precisam ser enviados sem esperar que o programa leia octetos no fluxo
  - Programa receptor é informado de sua chegada
  - **Ack**: Indica de flag é de reconhecimento
  - **Psh**: Entrega imediata à camada de aplicação. Não espera completar o buffer para envio
  - **Rst**: fechamento abrupto
  - **Syn**: Estabelecimento / sincronização
  - **Fin**: Fechamento Elegante

Cabeçalho TCP

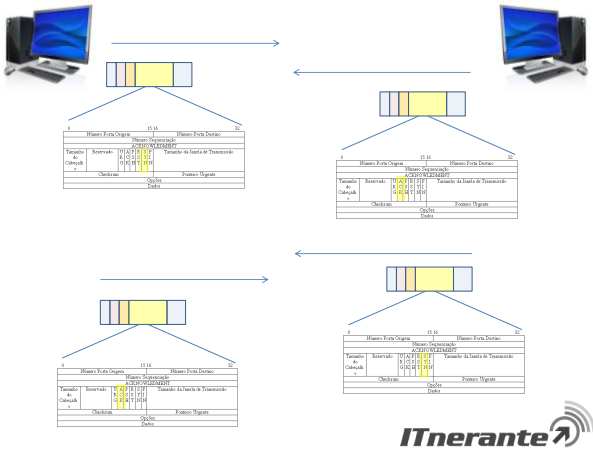
- Window: Controle de fluxo
- Checksum: Usa o pseudo cabeçalho que engloba alguns campos do cabeçalho IP
- Urgent Pointer: Casado com flag urgent. Indica a posição FINAL dos dados urgentes indicados no flag urg
- Options
  - MSS: Tamanho do payload TCP
  - Aumento de escala de janela para redes de alta velocidade
  - Marca de tempo
  - SACK
- Padding: Tornam o cabeçalho múltiplo de 32 bits

Ainda sobre cabeçalho TCP

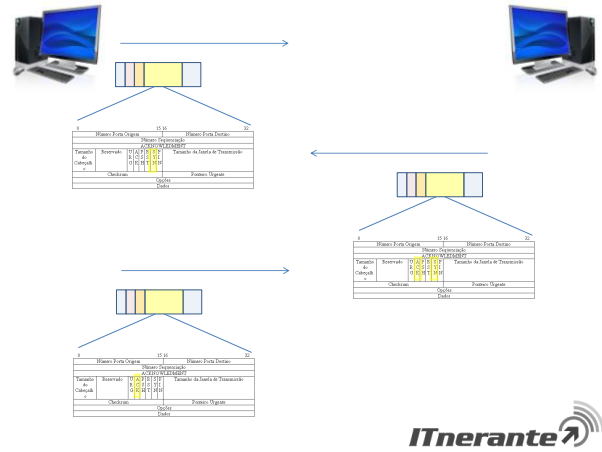
- Window: especifica o tamanho da parte de memória (buffer) disponível para os dados a receber
  - TCP window scale option é uma forma de aumentar o tamanho da janela de recepção além do limite dos 16 bits especificados no campo Window Size
- Alguns campos opcionais só aparecem com o Syn
- O campo de timestamp e sack aparecem normalmente noutros segmentos

Estabelecimento da conexão TCP

- Full-duplex e ponto-a-ponto
  - Não admite difusão e multidifusão
- Handshake de 3 vias: Explícito
  - Syn / Syn + Ack / Ack
    - Ack = Seq + 1 (próximo byte esperado)
    - Connection request (syn 1 ack 0)
    - Connection accepted (syn 1 ack 1)
- Piggybacking
  - É a inserção de mais informações do que a forma básica do protocolo, a fim de otimizar a movimentação de dados.



ITnerante

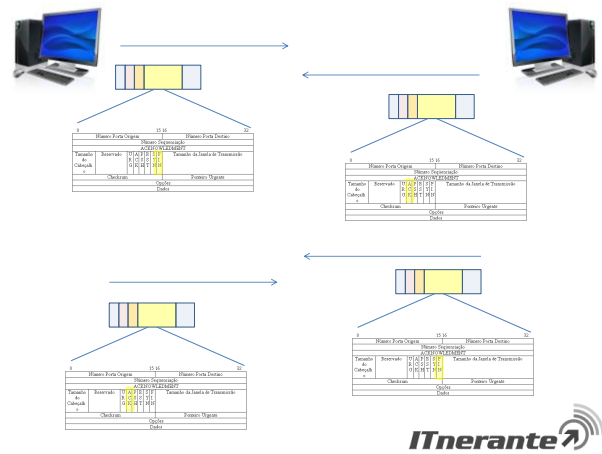


ITnerante

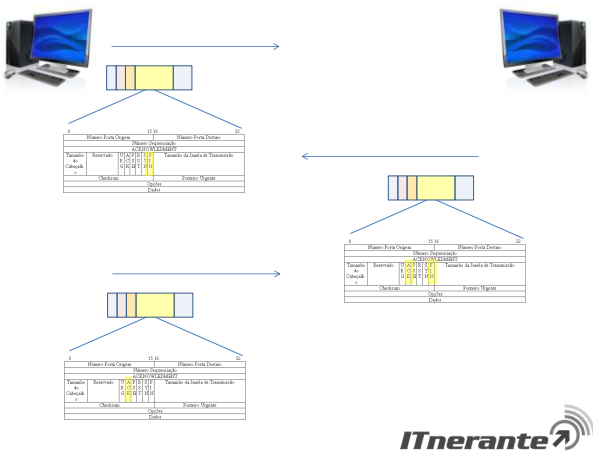
Encerramento da conexão TCP

- Encerramento formal da conexão TCP.
  - Buffers e variáveis liberados
- Simétrico
  - Cada direção é encerrada independentemente
  - FIN: Fecha em uma direção, dados permanecem fluindo na outra, até que o encerramento também se dê do outro lado
- Assimétrico
  - Abrupto
  - Transferências nas duas direções são terminadas e buffers liberados
  - Apesar do flag **RST**, Não existe reinicialização, mas sim o término da conexão.

ITnerante



ITnerante



ITnerante

O pseudo cabeçalho

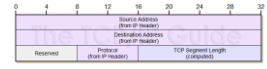


Figure 217: TCP "Pseudo Header" For Checksum Calculation

- Tamanho: 12 bytes
- End Origem: 4 bytes
- End Destino: 4 bytes
- Reservado: 1 byte
- Protocol: 1 byte
- Não é enviado, apenas usado para cálculo do checksum.

ITnerante

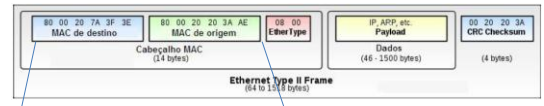
Camada de transporte - Portas

- 20, 21 – FTP
- 22 – SSH
- 23 – Telnet
- 25 – SMTP
- 49 – TACACS
- 53 – DNS
- 63 – WHOIS
- 67, 68 – DHCPv4
- 69 – TFTP
- 80 – HTTP
- 88 – KERBEROS
- 110 – POP3
- 123 – NTP
- 143 – IMAP
- 161, 162 – SNMP
- 179 – BGP
- 443 – HTTPS
- 992 – TELNETS
- 993 – IMAPS
- 995 – POP3S

- http://www.iana.org/assignments/service-names-port-numbers/service-names-port-numbers.txt

ITnerante

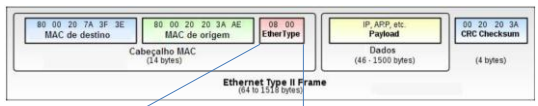
Identificadores importantes na análise de tráfego – Quadro 802.3



Os 3 primeiros bytes indicam o fabricante (Definido pelo IEEE). OS 3 últimos bytes indicam um número de série que é válido para aquele fabricante.

ITnerante

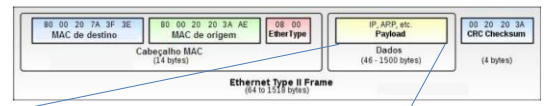
Identificadores importantes na análise de tráfego – Quadro 802.3



0x0800	Internet Protocol version 4 (IPv4)
0x0806	Address Resolution Protocol (ARP)
0x8035	Reverse Address Resolution Protocol
0x8100	VLAN-tagged frame (IEEE 802.1Q)
0x86DD	Internet Protocol Version 6 (IPv6)
0x888E	EAP over LAN (IEEE 802.1X)

ITnerante

Identificadores importantes na análise de tráfego – Quadro 802.3



- 16 Mbps Token Ring – 17914
- 4 Mbps Token Ring – 4464
- FDDI – 4352
- Ethernet UTP- 1500
- IEEE 802.3/802.2 – 1492
- 802.11: 2304 ANTES DA CRIPTOGRAFIA
  - WEP: + 8 bytes = 2312 bytes
  - WPA 2 (AES): + 16 bytes = 2320 bytes
  - WPA 1 OU 2 (tkip): + 20 bytes = 2324 bytes
- PPPoE (WAN Miniport) – 1480
- X.25 - 576

ITnerante

Principais Ferramentas de Análise de Tráfego - TCPDUMP

- Ferramenta utilizada para monitorar os pacotes trafegados numa rede de computadores
- Mostra os cabeçalhos dos pacotes que passam pela interface de rede
- Sua versão para windows chama-se WinDump
- Os filtros de captura são passados ao lado do executável seguido dos parâmetros
  - tcpdump -n -i eth0 port 80

ITnerante

Principais Ferramentas de Análise de Tráfego - TCPDUMP

- Parâmetros mais comuns
  - -i: Define a interface de rede
  - -n: Não converte IP em nome DNS
  - -x: Exibe o pacote em hexa
  - -X: Exibe o pacote em hexa + ASCII

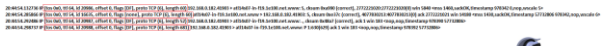
ITnerante

Principais Ferramentas de Análise de Tráfego - TCPDUMP

- Formato Padrão

- Com informações da camada de enlace (-e)

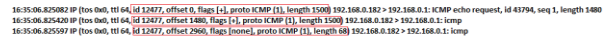
- Com informações do cabeçalho IP (-v)



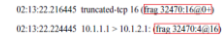
ITnerante

Principais Ferramentas de Análise de Tráfego - TCPDUMP

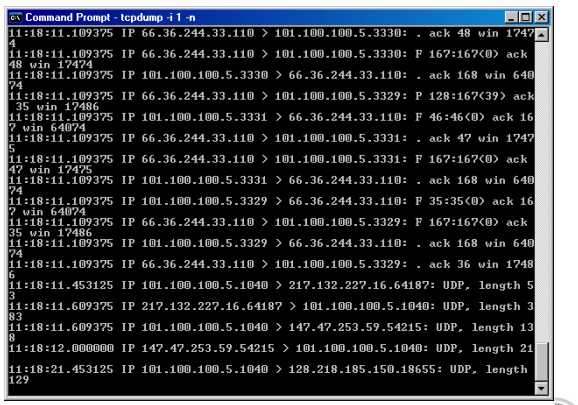
- Envolvendo fragmentação – Formato 01 (payload de 3000 bytes)



- Envolvendo fragmentação – Formato 02 (frag pcktID:size@offset+)



ITnerante



ITnerante

Principais Ferramentas de Análise de Tráfego - WIRESHARK

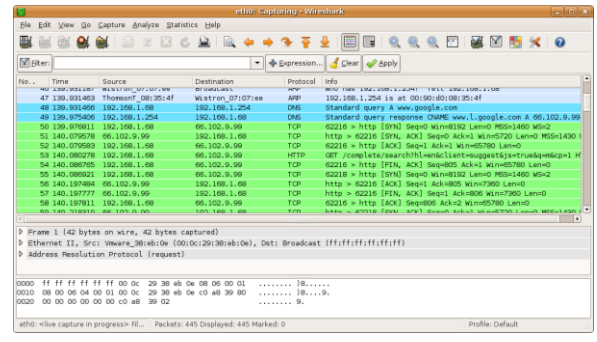
- É um analisador de tráfego para Windows e Linux
- É a continuação do Ethereal
- Suas funcionalidades são similares às do tcpdump, mas com interface GUI

ITnerante

Principais Ferramentas de Análise de Tráfego - WIRESHARK

- Tipos de filtros
  - Capture Filters
    - São filtros utilizados ANTES do início da captura
    - São os mesmos filtros utilizados no TCPDUMP
  - Display Filters
    - Realiza filtros no decorrer da captura
    - É somente um filtro para exibição

ITnerante

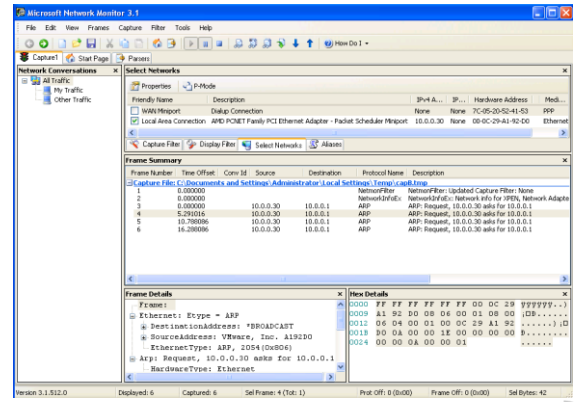


ITnerante

Principais Ferramentas de Análise de Tráfego – Microsoft NetMon

- NetMon captura pacotes utilizando o drive NDIS (Network Driver Interface Specification), sendo assim, captura tráfego da camada 2 (ex. beacons 802.11)
- Wireshark utiliza um drive separado, simulando um único protocolo de camada 2
- Outra diferença é que o NetMon separa e filtra o tráfego por processos, já o Wireshark não o faz

ITnerante



ITnerante

Questões de Aprendizagem

Análise de Tráfego

ITnerante

CPC – CESPE 2007 – Perito Criminal – Processamento de Dados

1. Quanto ao monitoramento de tráfego em uma rede, julgue os seguintes itens.

I - O tcpdump é um packet sniffer que possibilita a interceptação e apresentação de pacotes que trafegam por uma rede TCP/IP. Os dados nos pacotes interceptados podem ser armazenados em arquivos para posterior análise.

II - Um packet sniffer possibilita monitorar o tráfego em uma rede. Em uma rede Ethernet, para monitorar o tráfego destinado ao endereço de broadcast, a placa de interface com a rede precisa ser configurada no modo promíscuo.

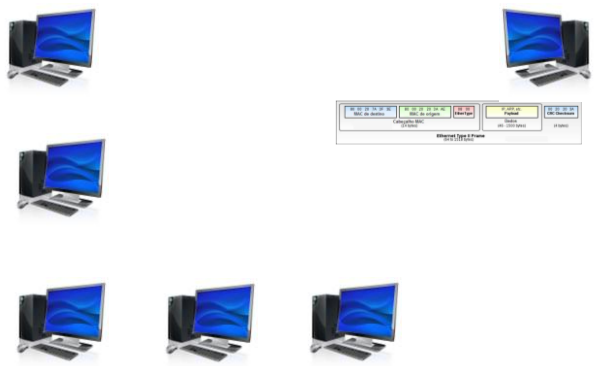
III - Em uma rede Ethernet, um packet sniffer pode ser usado para monitorar o tráfego destinado ao endereço de broadcast e a endereços de multicast, mas não tráfego unicast destinado à máquina com o packet sniffer.

IV - Há técnicas que podem ser usadas para se tentar identificar a presença de packet sniffers em redes Ethernet. Por exemplo, um pacote ARP pode ser enviado para um endereço que não seja o de broadcast. Se uma máquina responder a esse pacote, possivelmente tem uma placa de rede no modo promíscuo.

Estão certos apenas os itens

A. I e II.
B. I e IV.
C. II e III.
D. III e IV

ITnerante

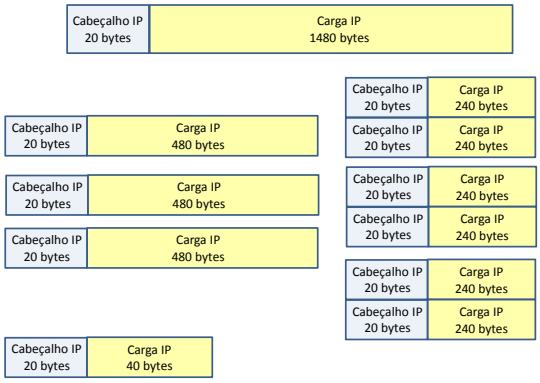


ANA – ESAF 2009 Analista Administrativo – Adm de Redes e Segurança

2. Ao fragmentar um fragmento, que não seja o último fragmento de um datagrama, o roteador IP deve

A. ativar o bit do flag 'mais fragmentos' em todos os sub-fragmentos que produzir, exceto no último deles.
B. ativar o bit do flag 'não fragmente'.
C. ficar inativo, pois é impossível ocorrer esta situação em redes IP.
D. ativar o bit do flag 'mais fragmentos' apenas do primeiro subfragmento que produzir.
E. ativar o bit do flag 'mais fragmentos' em todos os sub-fragmentos que produzir.

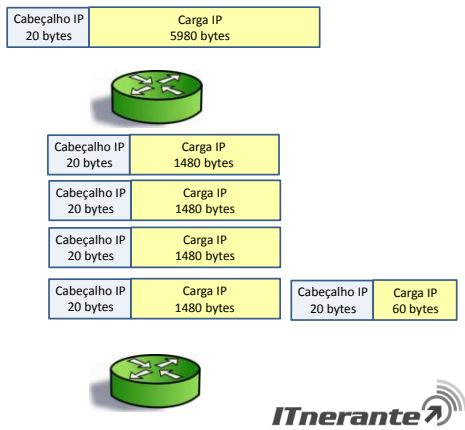
ITnerante



TRE MT – CESPE 2010 – Analista Judiciário – Análise de Sistemas

3. Em um enlace de comunicação de dados com MTU (maximum transmission unit) de 1.500 bytes, que conecta um roteador A a um roteador B, o roteador A recebe um datagrama de 6 kilobytes, a ser repassado ao roteador B. Esse enlace utiliza o protocolo IPv4, com cabeçalho padrão de 20 bytes, e permite a fragmentação. Com base nessas informações, é correto afirmar que

A. o último fragmento recebido pelo roteador B tem o campo de flag do cabeçalho IP ajustado para 1.
B. o primeiro fragmento tem o valor de deslocamento igual ao valor do cabeçalho IP.
C. o segundo fragmento tem deslocamento de 185, desde que o primeiro fragmento tenha sido enviado com o MTU máximo.
D. são necessários quatro fragmentos para transferir os 6 kilobytes do datagrama original.
E. o campo de flag do cabeçalho IP contém zero para todos os fragmentos, exceto o último.

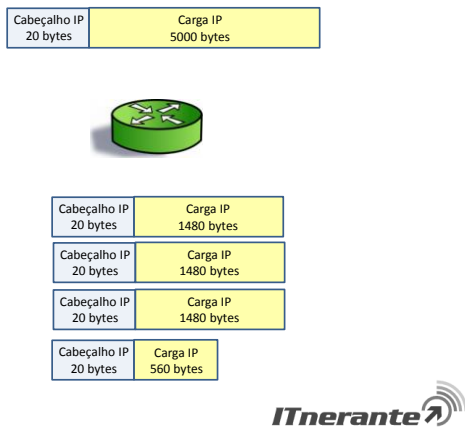


SEF-SC – FEPESE 2010 – Tecnologia da Informação

4. Suponha que um datagrama IP com 5.000 bytes de dados e cabeçalho de 20 bytes deve ser enviado através de um caminho de rede cuja unidade máxima de transmissão (MTU) é de 1500 bytes.

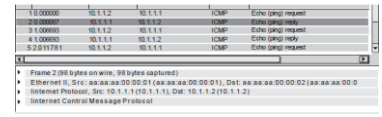
Assinale a alternativa correta a respeito dos fragmen-tos gerados pelo protocolo IP versão 4 a partir desse datagrama.

A. Os três primeiros fragmentos terão 1500 bytes de dados.
B. O primeiro fragmento terá o valor do campo identificação (identification) igual a 1, indi-cando que se trata do primeiro fragmento.
C. O valor do campo deslocamento do fragmento (fragment offset) do segundo fragmento será igual a 1480.
D. O valor do campo flag do quarto fragmento será igual a zero, para indicar que se trata do último fragmento do datagrama.
E. O valor do campo deslocamento do fragmento (fragment offset) de todos os fragmentos será igual a 20, para indicar que os dados do frag-mento iniciam após 20 bytes de cabeçalho.



DPE SP – FCC 2013 – Agente de Defensoria Pública – Administrador de Redes

5. Considere a seguinte tabela, obtida a partir de um aplicativo de captura de pacotes em uma rede de computadores.



Está correto afirmar que:

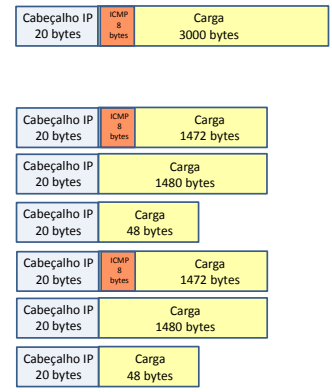
A. os endereços **Ethernet** de origem e destino dos computadores envolvidos nessa troca de dados são, respectivamente, aa:aa:aa:00:00:02 e aa:aa:aa:00:00:01
B. o protocolo utilizado no teste permite saber a rota para alcançar um **host**, mesmo na presença de **firewall**.
C. os endereços IP de origem e destino dos computadores envolvidos desde o inicio nas trocas de dados são, respectivamente, 10.1.1.1 e 10.1.1.2
D. o comando utilizado no teste tem a finalidade de testar a conectividade e o congestionamento da rede.
E. o protocolo utilizado no teste permite o controle de fluxo de pacotes na rede.

TRE RJ – CESPE 2012 – Técnico Judiciário – Operação de Computador

1 00.998952 IP 10.1.1.1 > 10.1.1.2: icmp 1480: echo request seq 4864 (frag 10550:1480@0+)
2 00.999881 IP 10.1.1.1 > 10.1.1.2: icmp (frag 10550:1480@1480+)
3 02.000787 IP 10.1.1.1 > 10.1.1.2: icmp (frag 10550:48@2960)
4 02.005395 IP 10.1.1.2 > 10.1.1.1: icmp 1480: echo reply seq 4864 (frag 3672:1480@0+)
5 02.007137 IP 10.1.1.2 > 10.1.1.1: icmp (frag 3672:1480@1480+)
6 02.008060 IP 10.1.1.2 > 10.1.1.1: icmp (frag 3672:48@2960)

6. Considerando a captura de tráfego mostrada acima, em que os endereços são fictícios, julgue os itens que se seguem

[77] A chegada dos fragmentos aos hosts de destino ocorreu fora da ordem de envio.
[78] Se os hosts tiverem máscaras de rede /24, eles estão em redes diferentes.
[79] O MTU dos enlaces em que se conectam os hosts é maior que 1.480.
[80] O datagrama IP que foi fragmentado carregava 3.008 bytes.
[81] Trata-se de tráfego consistente com a execução do comando ping no host 10.1.1.1.



MEC UNIPAMPA – CESPE 2009 – Analista Judiciário – Analista de TI Rede e Suporte

7. Considerando o trecho de captura de tráfego apresentado acima, julgue os próximos itens.

[81] A captura ilustra uma conexão em que o fluxo de dados é interativo.
[82] Os hosts da captura oferecem o mesmo valor inicial de janela deslizante.
[83] O encerramento da conexão não se deu de forma abrupta, mas totalmente dentro da normalidade.
[84] Apenas um dos hosts envolvidos na captura é capaz de tratar retransmissões seletivas.
[85] Apenas um dos hosts envolvidos na captura está conectado a uma rede cujo MTU é 1500.

```
0.280264 IP 10.1.1.1.1047 > 10.1.1.2.1100: S 0:0(0) win 65535 <mss 1460,nop,nop,sackOK>
0.280499 IP 10.1.1.2.1100 > 10.1.1.1.1047: S 0:0(0) ack 0 win 65535 <mss 1460,nop,nop,sackOK>
0.280560 IP 10.1.1.1.1047 > 10.1.1.2.1100: . ack 1 win 65535
0.282520 IP 10.1.1.2.1100 > 10.1.1.1.1047: P 1:19(18) ack 1 win 65535
0.413863 IP 10.1.1.1.1047 > 10.1.1.2.1100: . ack 19 win 65517
1.790006 IP 10.1.1.1.1047 > 10.1.1.2.1100: P 1:14(13) ack 19 win 65517
1.790368 IP 10.1.1.2.1100 > 10.1.1.1.1047: P 19:43(24) ack 14 win 65535
1.947466 IP 10.1.1.1.1047 > 10.1.1.2.1100: . ack 43 win 65493
3.596518 IP 10.1.1.1.1047 > 10.1.1.2.1100: P 14:27(13) ack 43 win 65493
3.690765 IP 10.1.1.2.1100 > 10.1.1.1.1047: . ack 27 win 65535
6.611284 IP 10.1.1.2.1100 > 10.1.1.1.1047: P 43:63(20) ack 27 win 65535
6.782028 IP 10.1.1.1.1047 > 10.1.1.2.1100: . ack 63 win 65473
8.195496 IP 10.1.1.1.1047 > 10.1.1.2.1100: P 27:33(6) ack 63 win 65473
8.195872 IP 10.1.1.2.1100 > 10.1.1.1.1047: P 63:91(28) ack 33 win 65535
8.195964 IP 10.1.1.1.1047 > 10.1.1.2.1100: F 33:33(0) ack 91 win 65445
8.196006 IP 10.1.1.2.1100 > 10.1.1.1.1047: F 91:91(0) ack 33 win 65535
8.196026 IP 10.1.1.1.1047 > 10.1.1.2.1100: . ack 92 win 65445
```

MPOG – CESPE 2009 – Analista Judiciário – Esp em Pol Púb e Gest Governamental

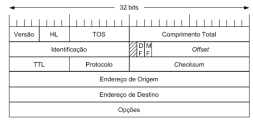
8. Considerando o trecho de captura de tráfego de rede apresentado acima, julgue os próximos itens.

[48] A captura em apreço ilustra uma conexão TCP com todas as suas fases, com tráfego interativo.
[49] Assumindo que a captura apresentada adira ao modelo cliente-servidor, o cliente seria o host 10.1.1.1 e servidor, o host 10.1.1.2.
[50] Segundo a captura em questão, ocorrem retransmissões de pacotes.

```
1  0.055429 IP (tos 0x0, ttl 128, id 2442, offset 0, flags [DF], proto: TCP (6), length: 48) 10.1.1.1.2373 >
   10.1.1.2.7777: S, cksum 0x9764 (correct), 160520737:160520737(0) win 64240 <mss 1460,nop,nop,sackOK>
2  0.055990 IP (tos 0x0, ttl 128, id 2691, offset 0, flags [DF], proto: TCP (6), length: 48) 10.1.1.2.7777 >
   10.1.1.1.2373: S, cksum 0xb8a6 (correct), 3778458614:3778458614(0) ack 160520738 win 17520 <mss
   1460,nop,nop,sackOK>
3  0.056088 IP (tos 0x0, ttl 128, id 2443, offset 0, flags [DF], proto: TCP (6), length: 40) 10.1.1.1.2373 >
   10.1.1.2.7777: ., cksum 0x2eea (correct), ack 1 win 64240
4  0.095338 IP (tos 0x0, ttl 128, id 2450, offset 0, flags [DF], proto: TCP (6), length: 1064) 10.1.1.1.2373
   > 10.1.1.2.7777: P, cksum 0x24b7 (correct), 1:1025(1024) ack 1 win 64240
5  0.095444 IP (tos 0x0, ttl 128, id 2451, offset 0, flags [DF], proto: TCP (6), length: 1500) 10.1.1.1.2373
   > 10.1.1.2.7777: P, cksum 0xc78f (correct), 1025:2485(1460) ack 1 win 64240
6  0.098918 IP (tos 0x0, ttl 128, id 2698, offset 0, flags [DF], proto: TCP (6), length: 40) 10.1.1.2.7777 >
   10.1.1.1.2373: ., cksum 0xdbb6 (correct), ack 2485 win 17520
7  0.099035 IP (tos 0x0, ttl 128, id 2452, offset 0, flags [DF], proto: TCP (6), length: 1500) 10.1.1.1.2373
   > 10.1.1.2.7777: P, cksum 0x970f (correct), 2485:3945(1460) ack 1 win 64240
8  0.099073 IP (tos 0x0, ttl 128, id 2453, offset 0, flags [DF], proto: TCP (6), length: 1500) 10.1.1.1.2373
   > 10.1.1.2.7777: P, cksum 0x1825 (correct), 3945:5405(1460) ack 1 win 64240
9  0.099109 IP (tos 0x0, ttl 128, id 2454, offset 0, flags [DF], proto: TCP (6), length: 1500) 10.1.1.1.2373
   > 10.1.1.2.7777: P, cksum 0x738f (correct), 5405:6865(1460) ack 1 win 64240
10 0.103041 IP (tos 0x0, ttl 128, id 2705, offset 0, flags [DF], proto: TCP (6), length: 40) 10.1.1.2.7777 >
   10.1.1.1.2373: ., cksum 0xd04e (correct), ack 5405 win 17520
```

CTI Renato Archer – CESPE 2008 – Tecnologista Pleno – Segurança de Sistemas de Informação

```
# tcpdump -i eth0 -l -n -x port 25
tcpdump: listening on eth0
14:17:51.950111 192.168.0.9.1100>192.168.0.1.25:
P 1043394526:1043394554(28)...
    4500 0044 9481 0000 4006 64d8 c0a8 0009
    c0a8 0001 044c 0019 3e30 efde 679c eea4
    5018 37ff 03b9 0000 7263 7074 2074 6f3a
    203c 7565 6461
```



9. Com base no resultado do comando tcpdump mostrado acima, julgue os itens a seguir.

[68] O cabeçalho do pacote mostra que é utilizada a versão IPv4 com um header length de 5, ou seja, 5 palavras de 4 bytes cada (20 bytes ao todo).

[69] O campo ToS (type of service) corresponde a 06 e total length corresponde a 0044, ou seja, 4 ×16 + 4 = 68 bytes, dos quais os 20 primeiros são os que correspondem ao cabeçalho.

[70] O source address é c0a80009, ou seja, 192.168.0.9. O protocolo utilizado é o TCP e o endereço destino é 192.168.0.1.


```
I     0.771929 IP (tos 0x10, ttl 64, id 46018, offset 0, flags [DF], proto: TCP (6), length: 60)
      1.1.1.1.1111  >  2.2.2.2.2222:  S,  cksum  0x1db2  (correct),  0:0(0)  win  5840  <mss
      1460,sackOK,timestamp 2538826 0,nop,wscale 6>

II    0.994556 IP (tos 0x0, ttl 50, id 20037, offset 0, flags [DF], proto: TCP (6), length: 44)
      2.2.2.2.2222 > 1.1.1.1.1111: S, cksum 0x9e62 (correct), 0:0(0) ack 1 win 5840 <mss 1460>

III   0.994605 IP (tos 0x10, ttl 64, id 46019, offset 0, flags [DF], proto: TCP (6), length: 40)
      1.1.1.1.1111 > 2.2.2.2.2222: ., cksum 0xb61f (correct), 1:1(0) ack 1 win 5840

IV    3.909380 IP (tos 0x10, ttl 64, id 46020, offset 0, flags [DF], proto: TCP (6), length: 47)
      1.1.1.1.1111 > 2.2.2.2.2222: P, cksum 0xa89d (correct), 1:8(7) ack 1 win 5840

V     4.220509 IP (tos 0x0, ttl 50, id 20038, offset 0, flags [DF], proto: TCP (6), length: 40)
      2.2.2.2.2222 > 1.1.1.1.1111: ., cksum 0xb618 (correct), 1:1(0) ack 8 win 5840

VI    4.220591 IP (tos 0x0, ttl 50, id 20041, offset 0, flags [DF], proto: TCP (6), length: 40)
      2.2.2.2.2222 > 1.1.1.1.1111: F, cksum 0xae04 (correct), 2068:2068(0) ack 8 win 5840

VII   4.220607 IP (tos 0x10, ttl 64, id 46021, offset 0, flags [DF], proto: TCP (6), length: 40)
      1.1.1.1.1111 > 2.2.2.2.2222: ., cksum 0xb618 (correct), 8:8(0) ack 1 win 5840

VIII  4.223374 IP (tos 0x0, ttl 50, id 20040, offset 0, flags [DF], proto: TCP (6), length: 647)
      2.2.2.2.2222 > 1.1.1.1.1111: P, cksum 0xe4c5 (correct), 1461:2068(607) ack 8 win 5840

IX    4.223381 IP (tos 0x10, ttl 64, id 46022, offset 0, flags [DF], proto: TCP (6), length: 40)
      1.1.1.1.1111 > 2.2.2.2.2222: ., cksum 0xb618 (correct), 8:8(0) ack 1 win 5840

X     4.229617 IP (tos 0x0, ttl 50, id 20039, offset 0, flags [DF], proto: TCP (6), length: 1500)
      2.2.2.2.2222 > 1.1.1.1.1111: ., cksum 0xbf1b (correct), 1:1461(1460) ack 8 win 5840

XI    4.229632 IP (tos 0x10, ttl 64, id 46023, offset 0, flags [DF], proto: TCP (6), length: 40)
      1.1.1.1.1111 > 2.2.2.2.2222: ., cksum 0xa29c (correct), 8:8(0) ack 2069 win 8760

XII   4.231280 IP (tos 0x10, ttl 64, id 46024, offset 0, flags [DF], proto: TCP (6), length: 40)
      1.1.1.1.1111 > 2.2.2.2.2222: F, cksum 0xa29b (correct), 8:8(0) ack 2069 win 8760

XIII  4.452312 IP (tos 0x0, ttl 50, id 20042, offset 0, flags [DF], proto: TCP (6), length: 40)
      2.2.2.2.2222 > 1.1.1.1.1111: ., cksum 0xae03 (correct), 2069:2069(0) ack 9 win 5840
```

BASA – CESPE 2010 – TI – Redes e telecomunicações

10. Considerando a captura de tráfego apresentada acima, na forma de segmentos numerados de I a XIII, julgue os itens que se seguem.

[91] A captura apresenta apenas uma conexão TCP, estabelecida nos segmentos de I a III e encerrada nos segmentos VI e de XI a XIII.

[92] O segmento XII consiste em uma retransmissão do segmento XI.

[93] Não é consistente a afirmativa de que a captura foi realizada no host 1.1.1.1.

[94] Houve entrega fora de ordem nos segmentos de IV a X.

[95] É consistente a afirmativa de que houve perda de segmentos na captura.

Gabarito

1. B
2. E
3. C
4. D
5. E
6. E, E, C, C, C
7. C, C, C, E, E
8. E, C, E
9. C, E, C
10. C, E, E, E, E