Zjednodusene zaklady ARP,TCP/IP

Jiri Kubina

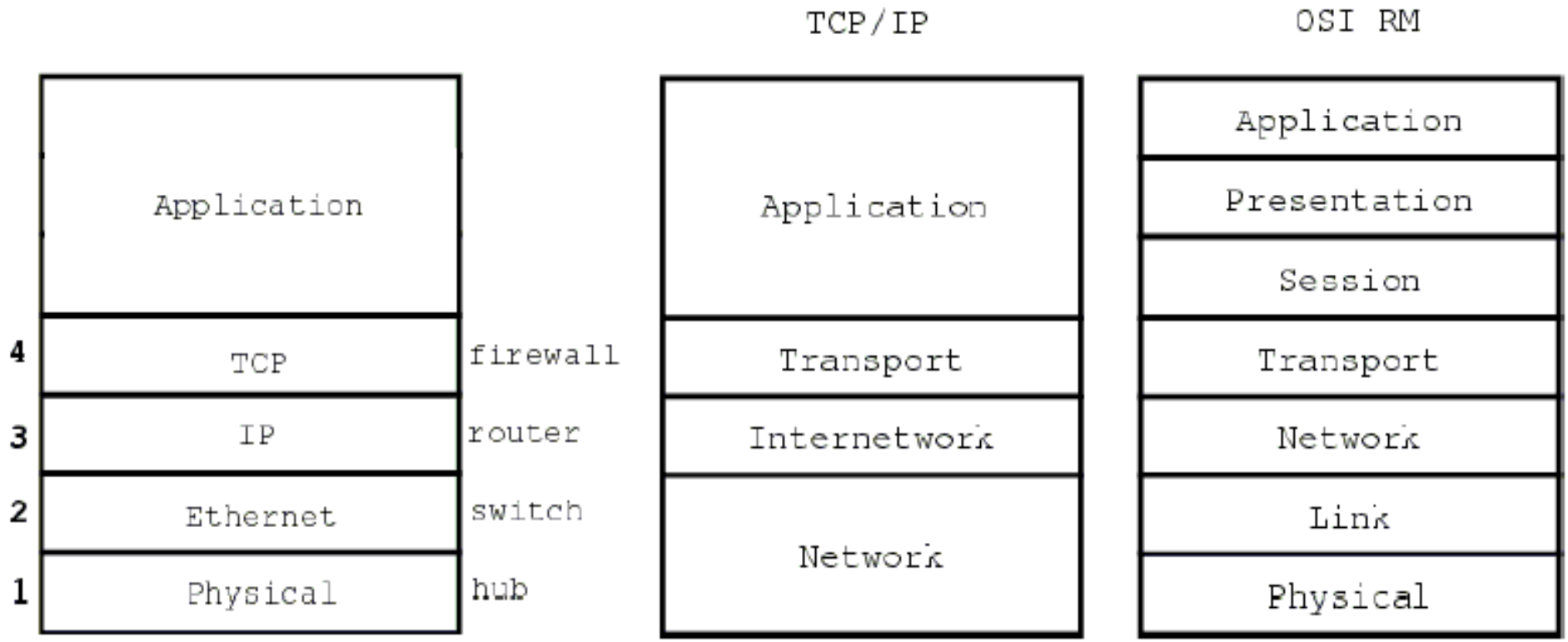
jiri.kubina@osu.cz

Ver. 1.0

leden 2006

Obsah

Upozorneni: Vetsina popisovanych principu je zjednodusena, nebo zkracena z vyukovych duvodu. Pro ziskani kompletnich informaci ctete RFC 791, 1042, 894, 826, 1027, 793


TCP/IP
OSI RM
Application
4
TCP
firewall
3
IP
router
2
Ethernet
switch
1
Physical
hub
Application
Transport
Internetwork
Network
Application
Presentation
Session
Transport
Network
Link
Physical

1.ARP - zjednoduseny popis prekladu IP na MAC

ARP – Address Resolution Protocol

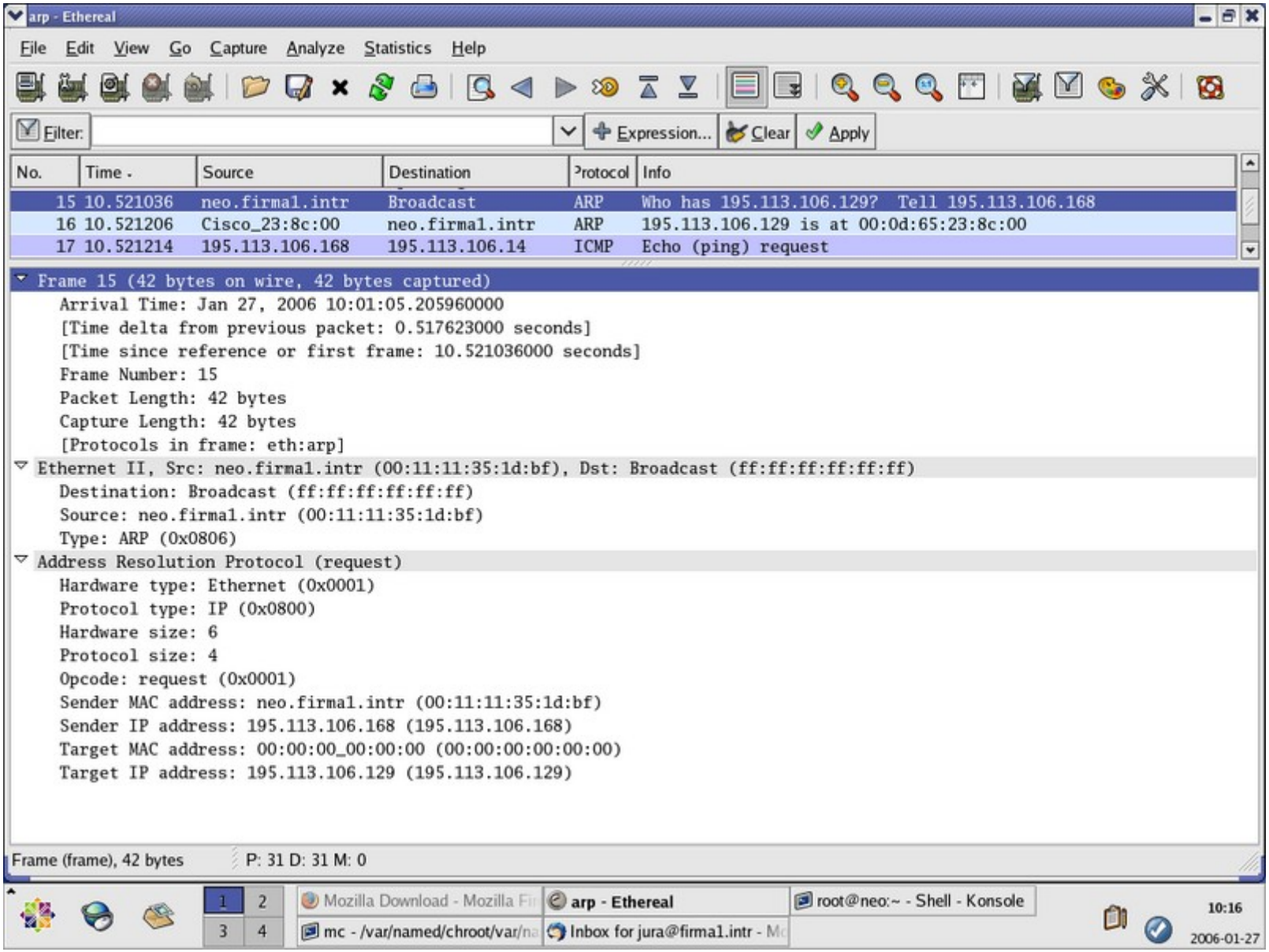
- Host A chce komunikovat pomoci IP s Host B na stejne siti. Pro komunikaci znam IP-odesilatele a IP-prijemce a tudiz jsem schopen sestavit IP-datagram /3 vrstva/. Problem je v tom ze IP-datagram musi byt zabalen v linkovem ramci /2 vrstva/ napr. ethernet. Aby jsem mohl vytvorit ethernetovy ramec potrebuji znat linkovou /MAC/ adresu odesilatele a prijemce. Odesilatel jsem ja a svou MAC adresu znam, zbyva tedy zjistit MAC prijemce.Toto resi protokol ARP.

- ARP resi problem zjisteni MAC adresy jineho PC na stejne siti ze znalosti jeho IP. Do site se vysle linkovy broadcast typ 1 /2 vrstva/ s cilovou adresou FF:FF:FF:FF:FF:FF. V tomto broadcastu uvede svoji MAC svoji IP a IP prijemce. Protoze je cilova MAC FF:FF:FF:FF:FF:FF dostane se ke vsem PC na tomto segmentu site. To PC ktere ma IP uvedenou v broadcastu se ozve linkovym datagramem typ 2 /opet 2 vrstva/ ale uz ne broadcastem ale na konkretni MAC adresu, kterou obdrzel v predchozim broadcastu.

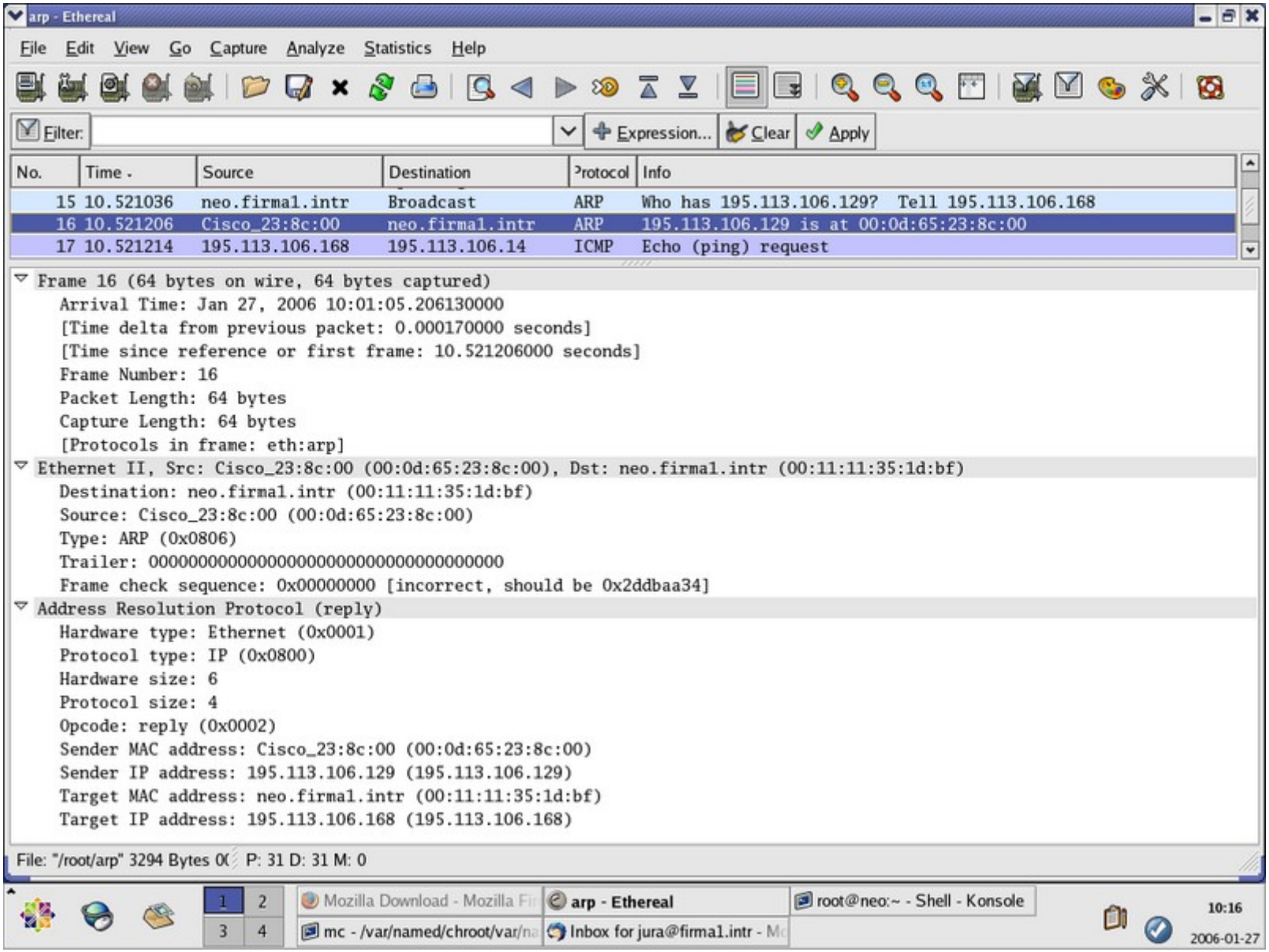
- Vypis programu tcpdump

09:54:17.063770 arp who-has 195.113.106.168 tell 195.113.106.167
09:54:17.063875 arp reply 195.113.106.168 is-at 00:11:11:35:1d:bf

ARP broadcast - request



ARP broadcast - reply



2.Strucny prehled IP protokolu

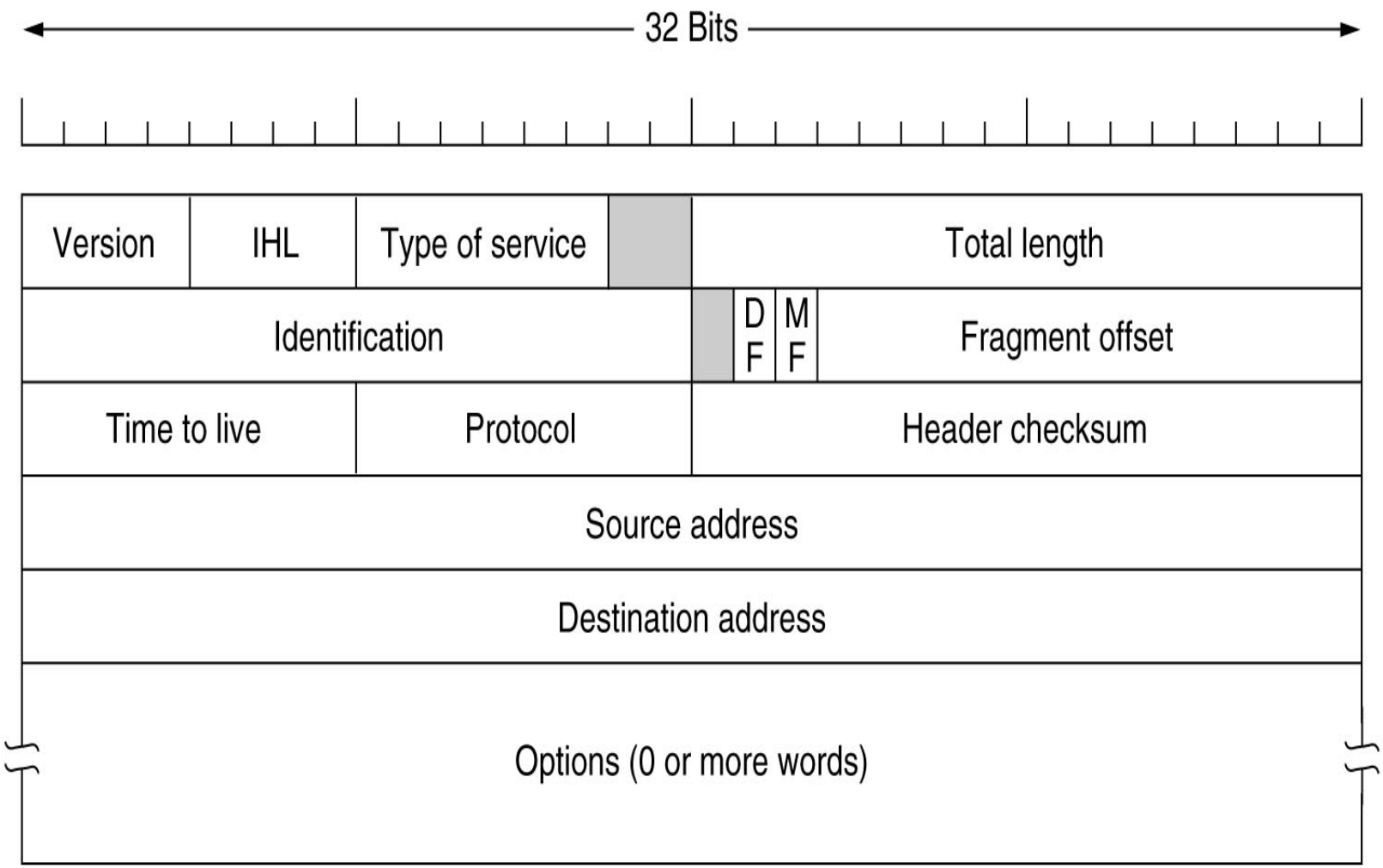
3. vrstva, sitova nespojovana sluzba posilani nezavisle smerovanych paketu

Nektere **linkove** protokoly jsou urceny pro dopravu dat v ramci lokalna site, jine linkove protokoly dopravuji data mezi sousednimi smerovaci rozsahlych siti.

IP-protokol na rozdil od linkovych protokolu dopravuje data mezi dvema libovolnymi pocitaci v Internetu, tzn. i pres nekolik LAN.

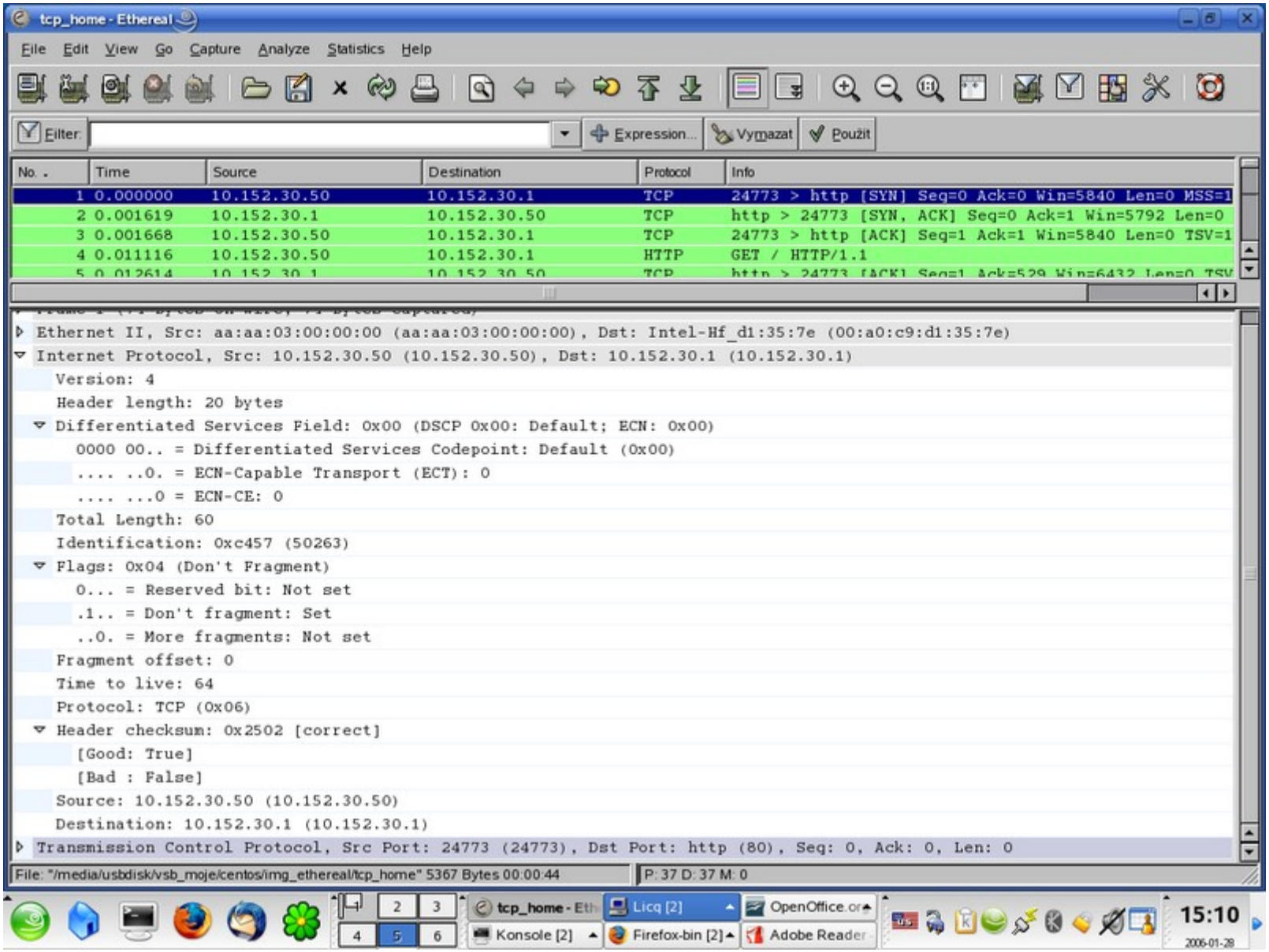
Pakety jsou od odesilatele k prijemci dopravovany-smerovany pres smerovace /router/. Po ceste muze byt cela rada smerovacu. Kazdy smerovac resi samostatne smerovani k nasledujicimu smerovaci.

IP - datagram

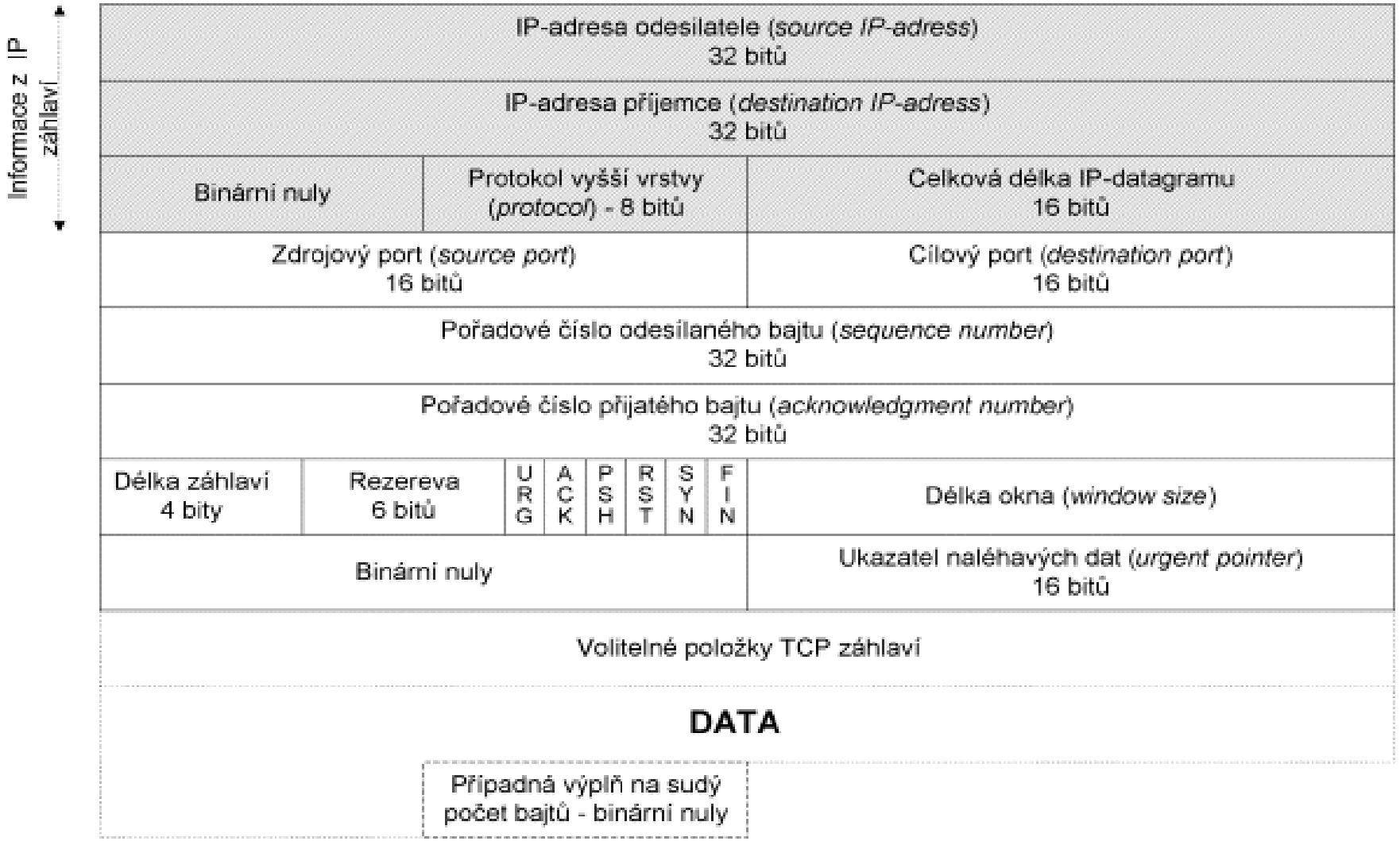

32 Bits
Version
IHL
Type of service
Total length
Identification
D F
M F
Fragment offset
Time to live
Protocol
Header checksum
Source address
Destination address
Options (0 or more words)

- **version** - verze protokolu bud Ipv4 nebo IPv6
- **header length** – informuje o delce hlavicky datagramu, protoze header obsahuje options s promennou delkou
- **type of service** - urcuje prioritu paketu pri posilani siti /dnes **Differentiated Services Field** /
- **total length** - delka IP paketu (max. 64kB)
- **identification** - ID paketu (docasne) jednoznacne v ramci zdroje, vsechny fragmenty paketu maji ID stejne
- **flags** - povoleni/zakaz fragmentace, indikace posledniho fragmentu fragmentovaneho paketu
- **Time to Live** - pocitadlo snizovane pri kazdem pruchodu smerovacem, proti zbloudilym paketum cyklujicim ve smerovaci smycce. Pri docitani do nuly se paket zahodi.
- **protocol** - cislo protokolu vyssi vrstvy neseneho v paketu
- **header checksum** – kontrolni soucet všech bytu hlavicky datagramu, nezahrnuje datovou cast paketu (!)
- **source & destination IP address** – IP adresa zdrojoveho a ciloveho pocitace (zarizeni)
- **options** - volitelne, promenna delka

IP – datagram


tcp_home - Ethereal
File Edit View Go Capture Analyze Statistics Help
Filter:
Expression... Vymazat Použít
No. . Time Source Destination Protocol Info
1 0.000000 10.152.30.50 10.152.30.1 TCP 24773 > http [SYN] Seq=0 Ack=0 Win=5840 Len=0 MSS=1
2 0.001619 10.152.30.1 10.152.30.50 TCP http > 24773 [SYN, ACK] Seq=0 Ack=1 Win=5792 Len=0
3 0.001668 10.152.30.50 10.152.30.1 TCP 24773 > http [ACK] Seq=1 Ack=1 Win=5840 Len=0 TSV=1
4 0.011116 10.152.30.50 10.152.30.1 HTTP GET / HTTP/1.1
Ethernet II, Src: aa:aa:03:00:00:00 (aa:aa:03:00:00:00), Dst: Intel-Hf_d1:35:7e (00:a0:c9:d1:35:7e)
Internet Protocol, Src: 10.152.30.50 (10.152.30.50), Dst: 10.152.30.1 (10.152.30.1)
Version: 4
Header length: 20 bytes
Differentiated Services Field: 0x00 (DSCP 0x00: Default; ECN: 0x00)
0000 00.. = Differentiated Services Codepoint: Default (0x00)
.... ..0. = ECN-Capable Transport (ECT): 0
....0 = ECN-CE: 0
Total Length: 60
Identification: 0xc457 (50263)
Flags: 0x04 (Don't Fragment)
0... = Reserved bit: Not set
.1.. = Don't fragment: Set
..0. = More fragments: Not set
Fragment offset: 0
Time to live: 64
Protocol: TCP (0x06)
Header checksum: 0x2502 [correct]
[Good: True]
[Bad : False]
Source: 10.152.30.50 (10.152.30.50)
Destination: 10.152.30.1 (10.152.30.1)
Transmission Control Protocol, Src Port: 24773 (24773), Dst Port: http (80), Seq: 0, Ack: 0, Len: 0
File: "/media/usbdisk/vsb_moje/centos/img_ethereal/tcp_home" 5367 Bytes 00:00:44
P: 37 D: 37 M: 0
tcp_home - Eth
Licq [2]
OpenOffice.or
Konsole [2]
Firefox-bin [2]
Adobe Reader
15:10

3.Hlavicka TCP



4.Navazani spojeni TCP

Navazani spojeni - Three way handshake

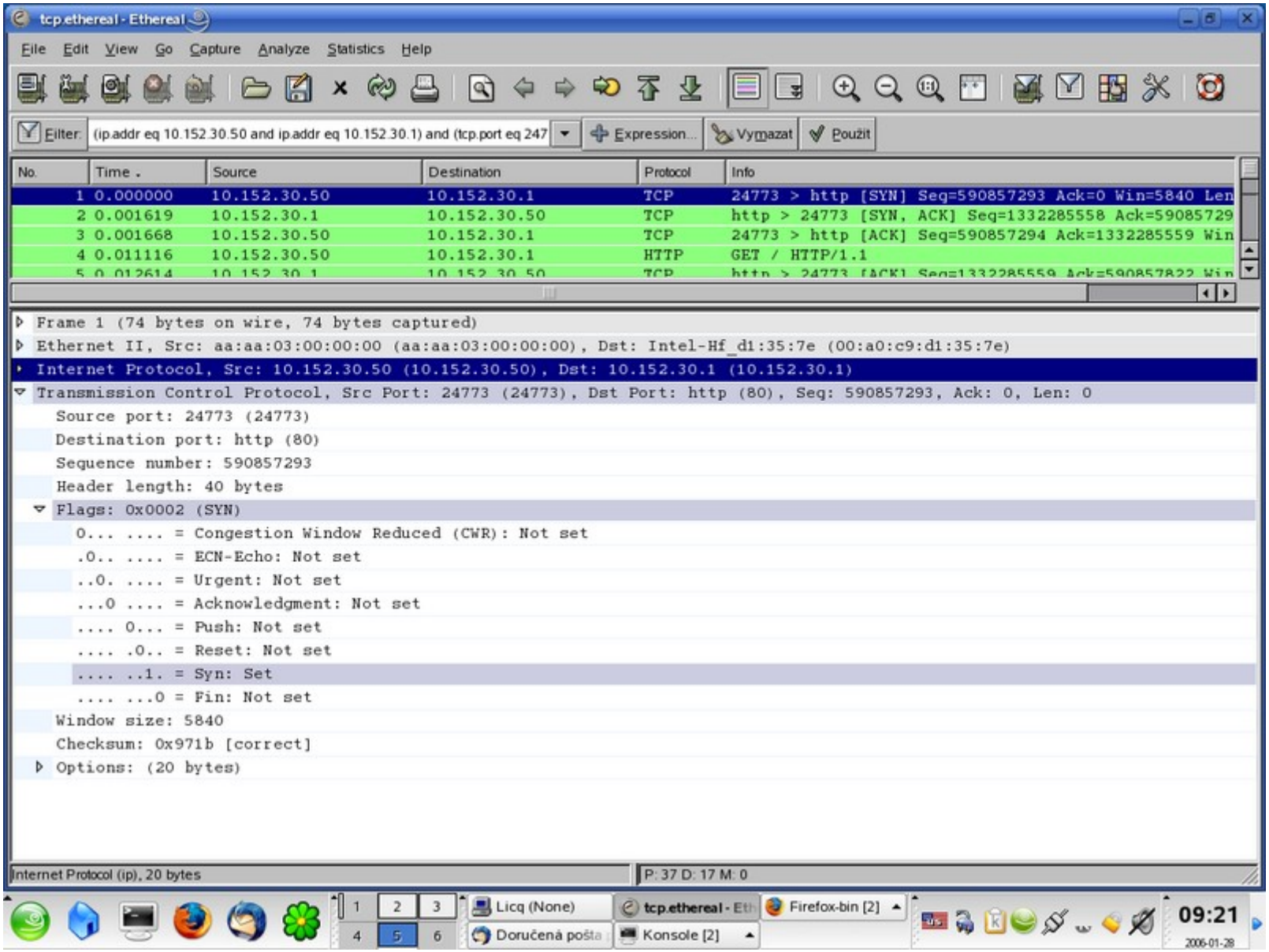
1.Odeslani synchronizacniho (SYN) paketu a Initial Sequence Number (ISN-A) /ACK number neni definovan/

Host A si preje navazat spojeni s Host B. Host A posle paket na Host B s nastavenym synchronizacnim bitem (SYN),ktery oznamuje vznik noveho spojeni a Initial Sequence Number (ISN), ktery umoznuje rozpoznavat /sledovat, urcovat/ pakety posilane mezi Host A a Host B.

Host A ------ SYN(ISN-A) ------> Host B

Poznamka: ISN jsou nahodna, aby se zabranilo pripadnemu ovlivneni zbloudilymi pakety ze zavreneho a brzy znovu otevreneho spojeni mezi stejnymi Hosty

Host A ------ SYN(ISN-A) ------> Host B

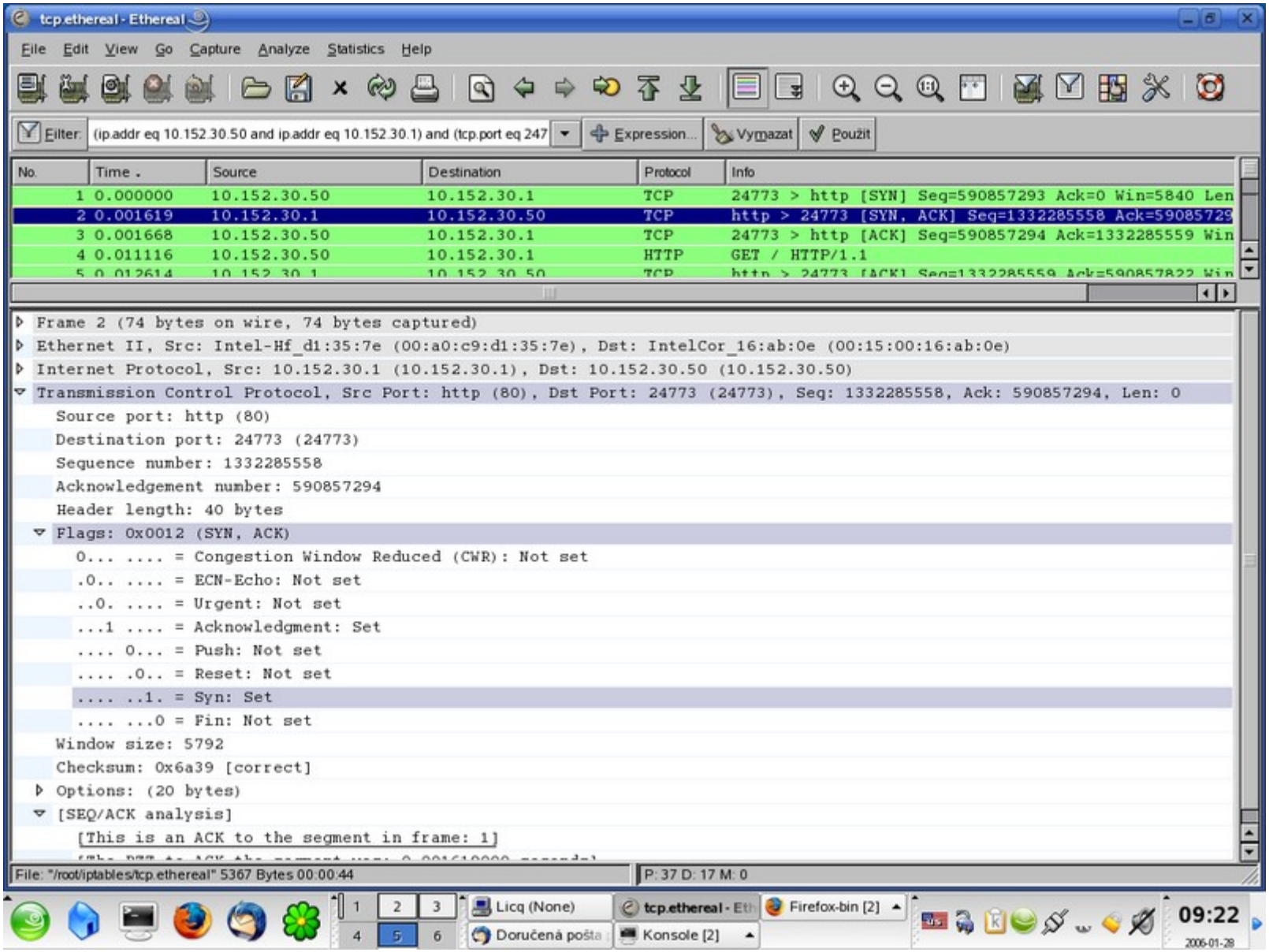


2.Odeslani potvrzovaciho (ACK) paketu

Host B na zadost /z Host A/ odpovida odeslanim paketu s nastavenym synchronizacnim bitem (SYN) a acknowledgment bitem (ACK) na Host A. Host B zasila svuj ISN-B a jako ACK number je odeslan Initial Sequence Number /z Host A/ plus 1 (ISN-A+1) ktere indikuje ze paket navazujici spojeni byl korektne prijat.

Host A <------ SYN(ISN-B)/ACK(ISN-A+1) ------ Host B

Host A <------ SYN(ISN-B)/ACK(ISN-A+1) ------ Host B

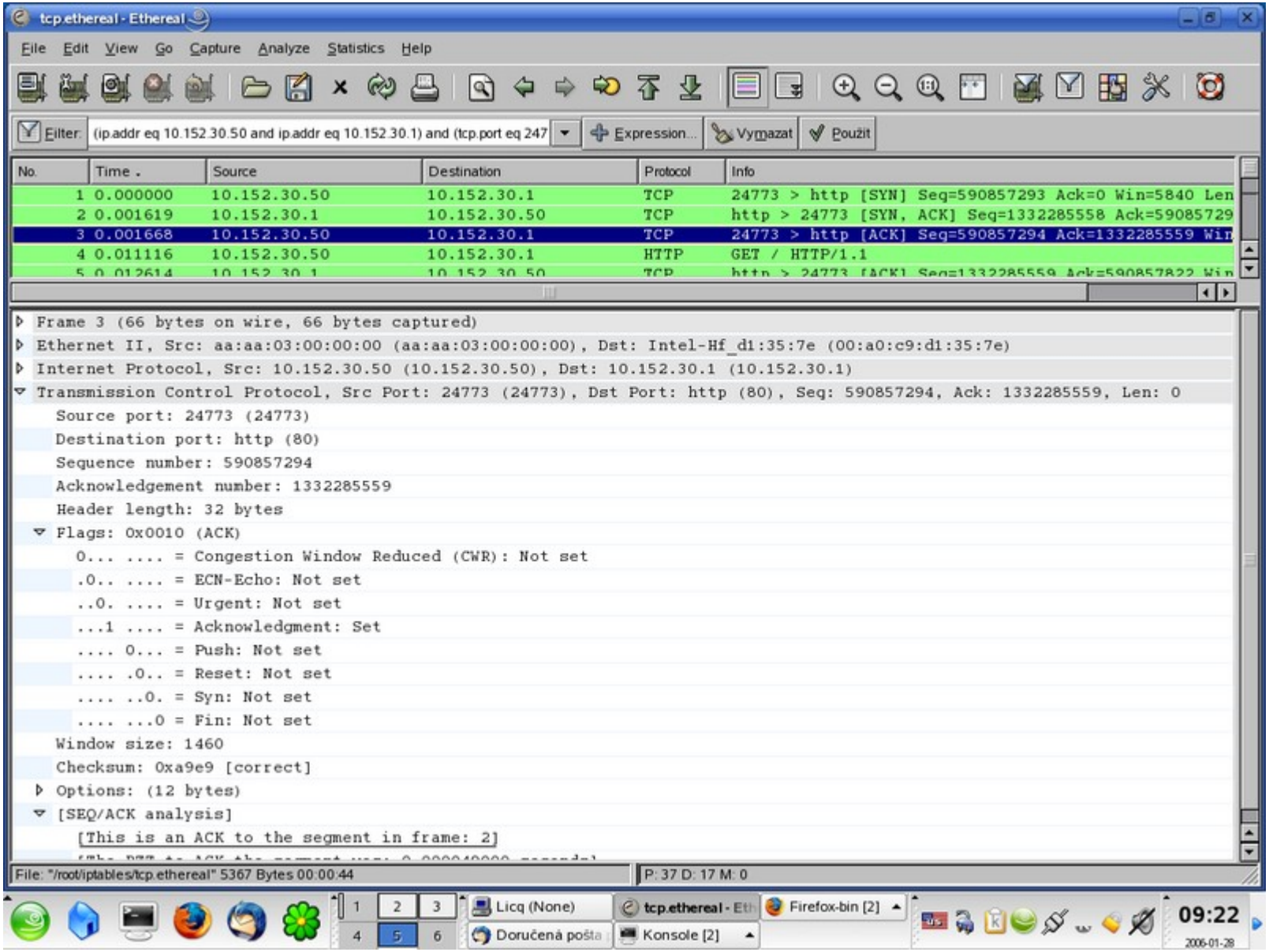


3.Dokonceni negotiation /vyjednavani, dohodnuti/ odeslanim finalniho acknowledgment (ACK) paketu na Host B.

Host A zasila zpet na Host B finalni ACK paket a sequnce number jako potvrzeni prijeti predchoziho paketu. Spojeni je kompletni, data mohou zacit tect mezi Host A a Host B.

Host A ------ ACK(ISN-B+1) ------> Host B

Host A ------ ACK(ISN-B+1) ------> Host B



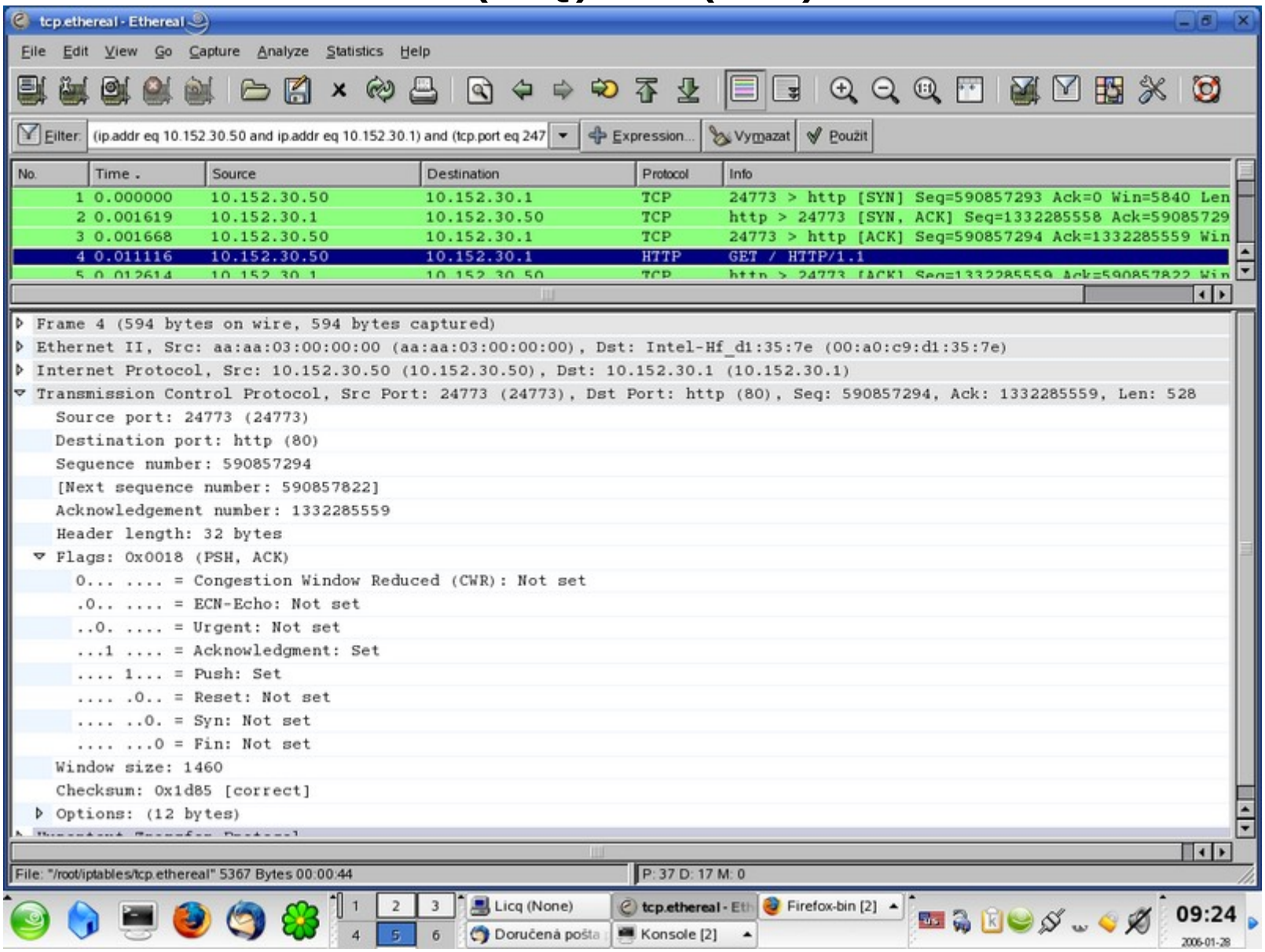
5.Datova vymena TCP

1.Prenos dat protokolu vyssi vrstvy pomoci paketu s priznakem PSH/ACK

Host A chce prenest data /konkretne protokolu HTTP - GET/ na Host B. Data zabali do TCP paketu a obvykle ho vybavi ho priznakem PSH/ACK. Priznak PSH neni povinny (vubec se na nej neda spolehnout). To ze paket nese nejaka data neni urceno prikazem PSH, ale tim , ze datova delka TCP paketu je vetsi nez nula ! Delka paketu je v nasem pripade 528B.

Host A ----- PSH(SEQ)/ACK(ACK) ------> Host B

Host A ----- PSH(SEQ)/ACK(ACK) ------> Host B

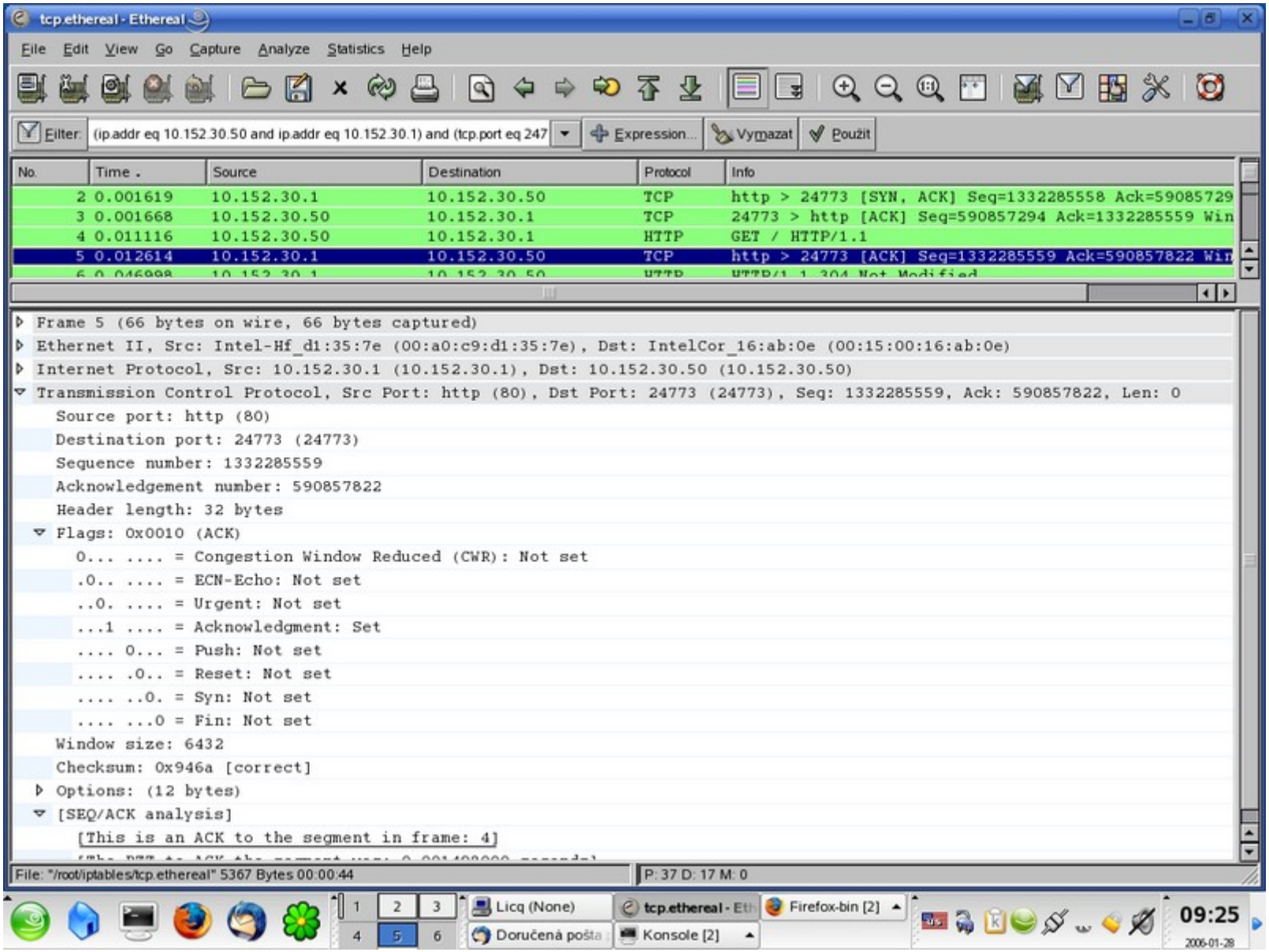


2.Potvrzeni prijeti paketu s daty ACK

Host B potvrzuje prijeti paketu s daty paketem s priznakem ACK. Delka pakeu je 0B.

Host A <------ ACK(SEQ+528) ------ Host B

Host A <------ ACK(SEQ+528) ------ Host B



6.Ukonceni spojeni TCP

Muze inicionat libovolna strana /jak klient tak server/.

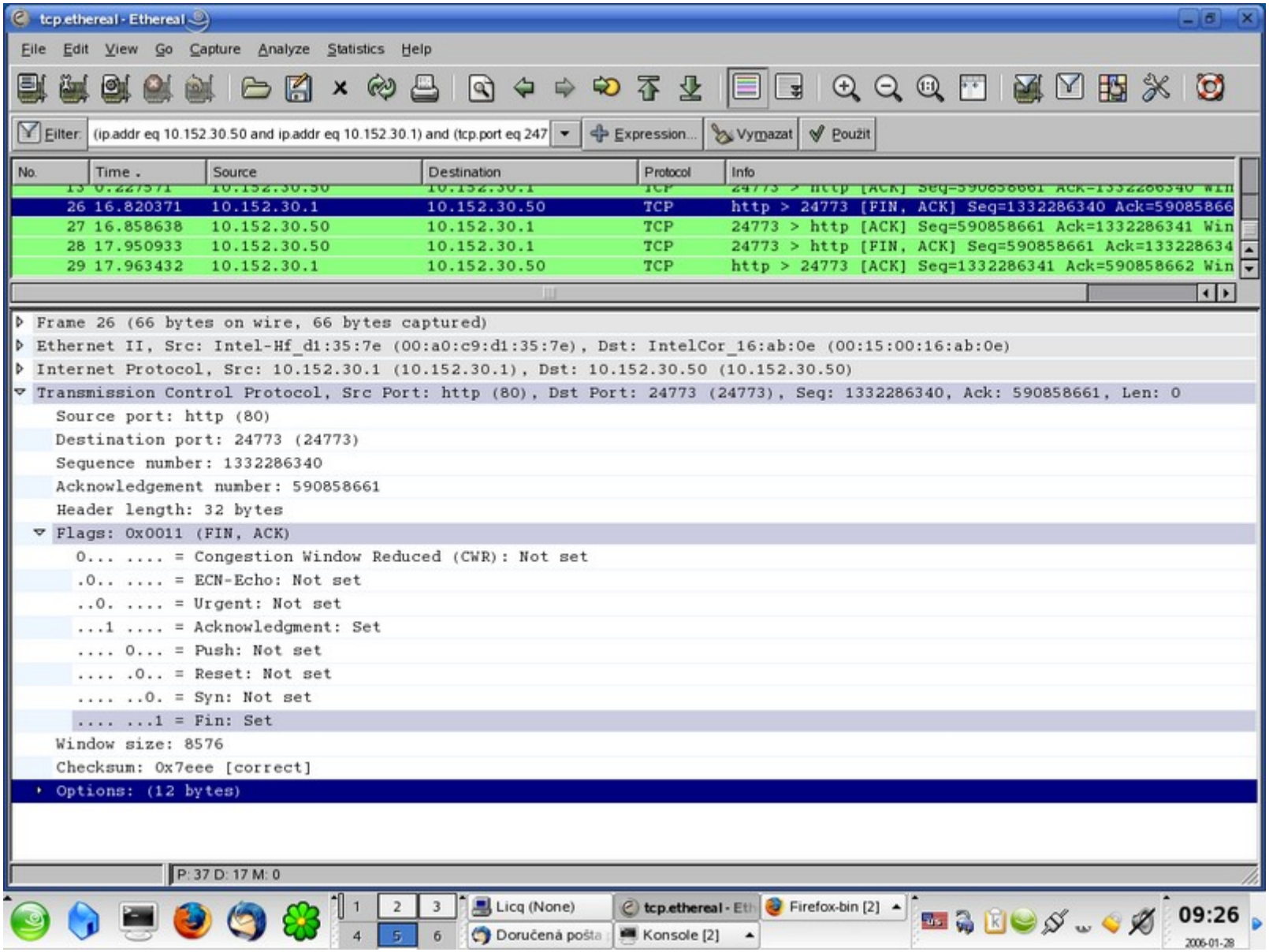
-**half closed** - FIN pouze z jedne strany
-**closed** - FIN z obou stran

1.Odeslani FIN/ACK paketu

Host B jiz odeslal vsechna data na Host A a chce proto ukoncit spojeni. Posila proto na Host A paket s priznakem FIN/ACK. Host B timto zahajil aktivni ukonceni spojeni a Host A jiz nyni muze provest pouze pasivni ukonceni spojeni. Od teto chvile jiz nemuze Host B odesilat data /pakety s priznakem PSH. Druha strana vsak muze v odesilani dat pokracovat, dokud sama neprovede ukonceni spojeni /tomuto stavu se rika half closed/.Paket se posuzuje jako by mel delku 1B.

Host A <------ FIN(SEQ-B)/ACK(ACK) ------ Host B

Host A <------ FIN(SEQ-B)/ACK(ACK) ------ Host B



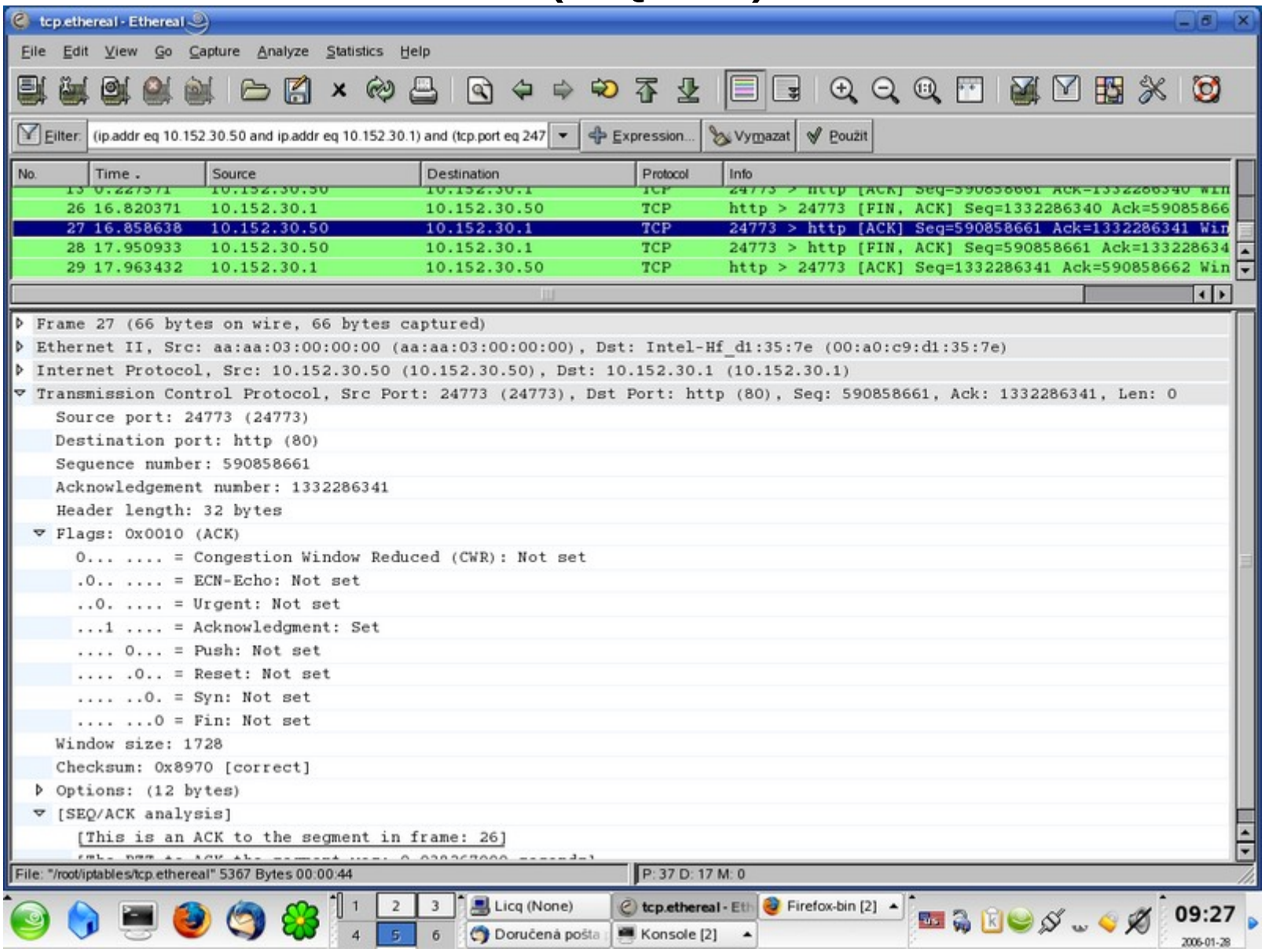
2.Potvrzeni prijeti FIN paketu

Host A potvrzuje uzavreni spojeni druhou stranou /provadi pasivni ukonceni spojeni/. Je mozne v tomto paketu odeslat i priznak FIN a tim provest uzavreni celeho spojeni. Paket se posuzuje jako by mel delku 0B.

Host A ------ ACK(SEQ-B+1) ------> Host B

Nyni muze Host A stale posilat data na Host B a ten mu je bude potvrzovat pakety s priznakem ACK. Pokud jiz Host A nema co odesilat a preje si celkove ukoncit spojeni nasleduje dalsi krok.

Host A ------ ACK(SEQ-B+1) ------> Host B

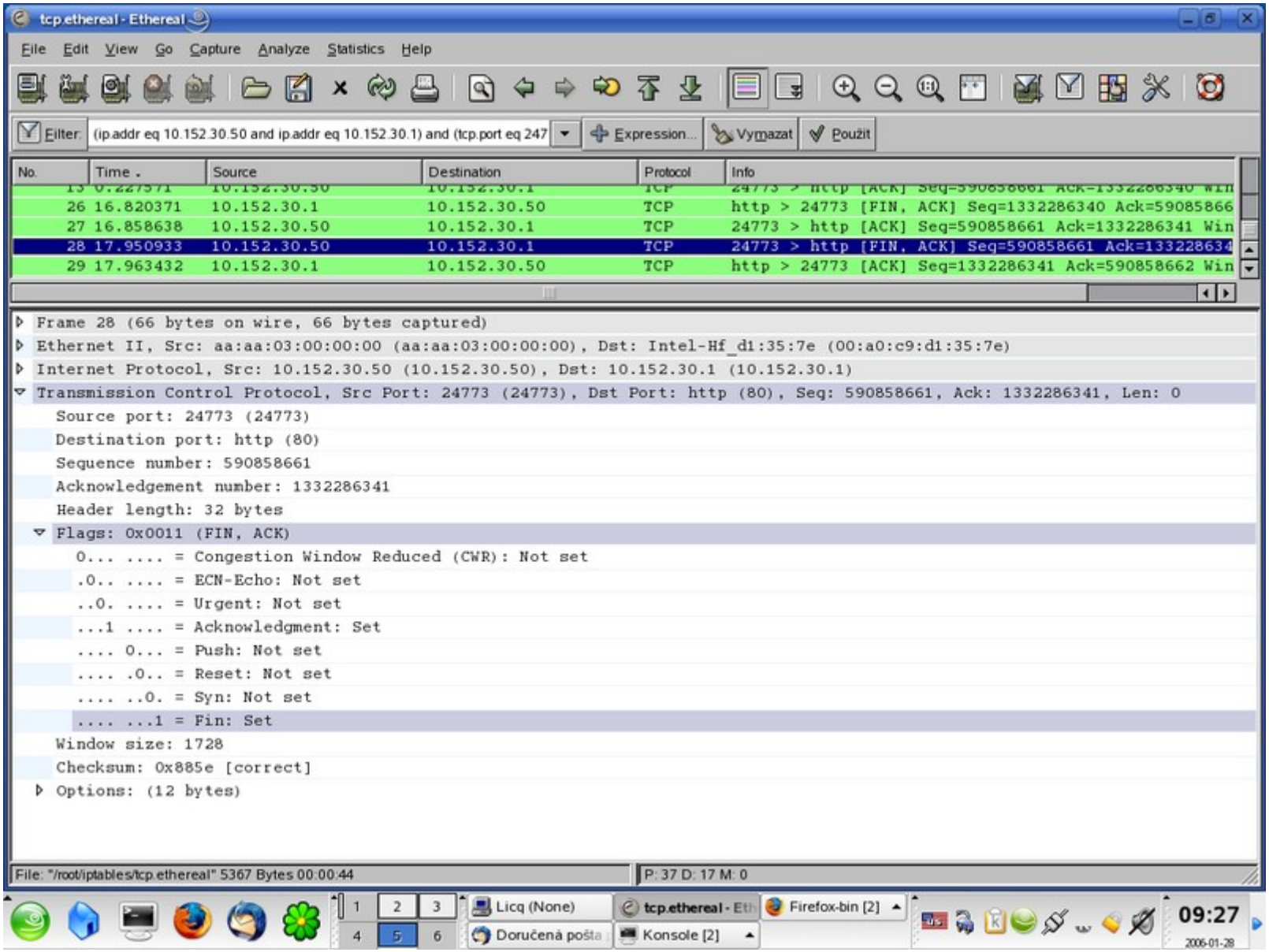


3.Ukonceni spojeni i z druhe strany

Host A jiz nema co odesilat a preje si ukoncit spojeni. Odesle na Host B paket s priznakem FIN/ACK a tim ukonci spojeni i z druhe strany. Paket se posuzuje jako by mel delku 1B.

Host A ------ FIN(SEQ-A)/ACK(SEQ-B+1) ------> Host B

Host A ------ FIN(SEQ-A)/ACK(SEQ-B+1) ------> Host B

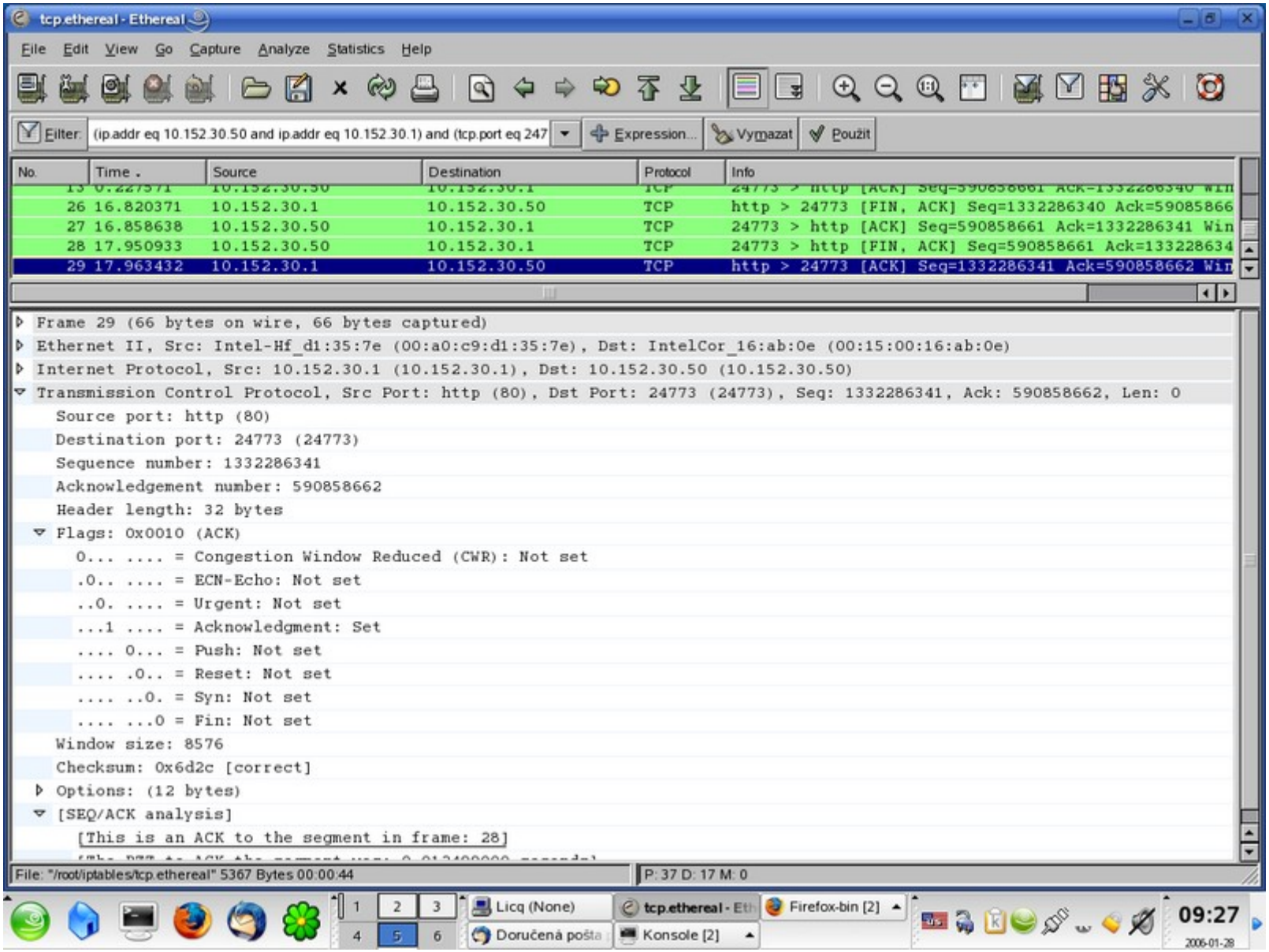


4.Potvrzeni celkoveho ukonceni spojeni ACK

Host B provede potvrzeni celkoveho ukonceni spojeni paketem s priznakem ACK. Po tomto paketu je spojeni ukonceno a data nemohou tect ani jednim smerem.

Host A <------ ACK(SEQ-A+1) ------ Host B

Host A <------ ACK(SEQ-A+1) ------ Host B

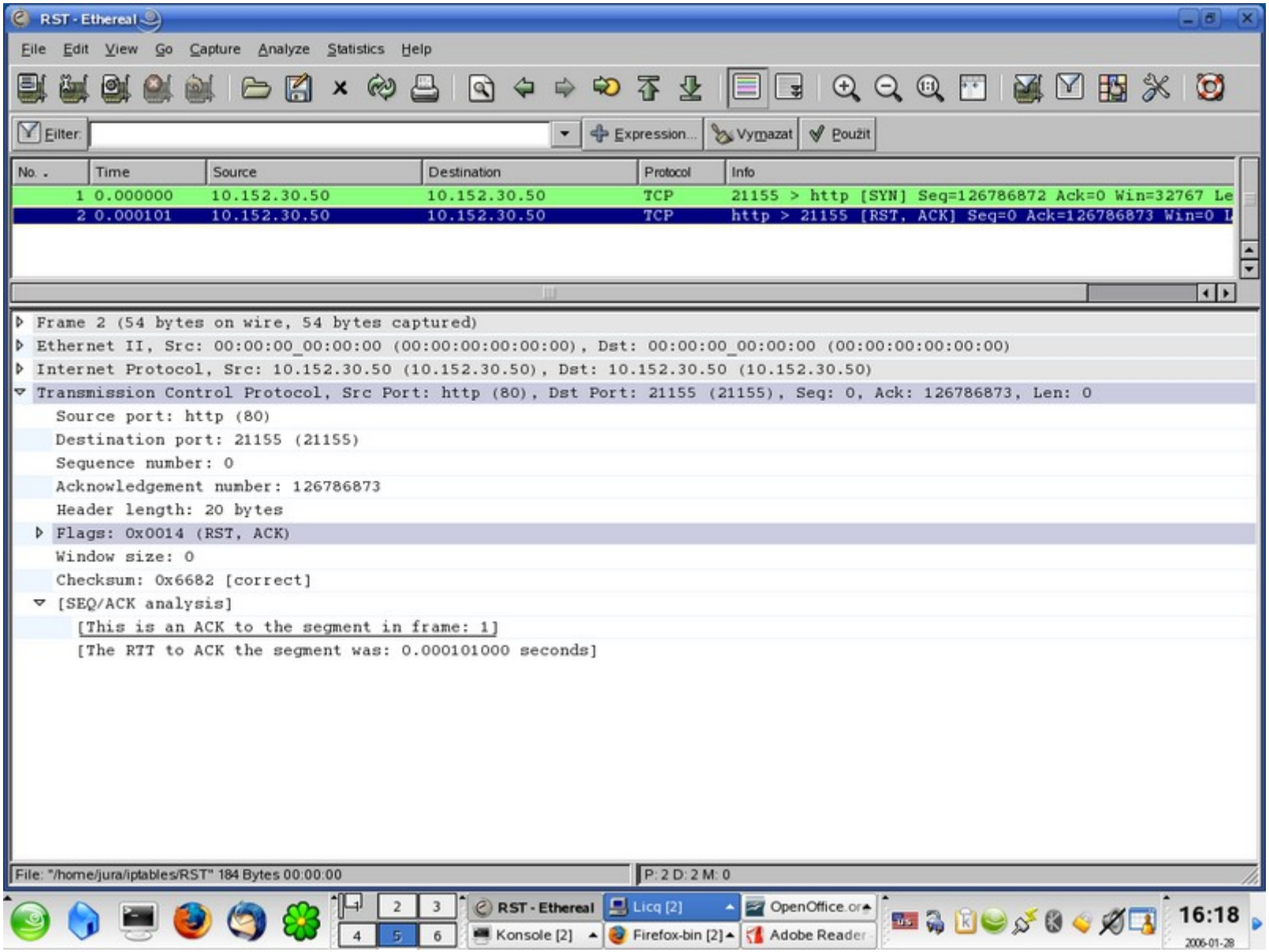


6.Reset spojeni TCP

Spojeni se odmita nastavenim priznaku RST v TCP paketu. Toto se deje obvykle ve dvou pripadech:

- Klient pozaduje spojeni na TCP portu na kterem server neposloucha
- Aplikace odmita dale pokracovat v jiz navazanem spojeni

Reset spojeni TCP



7.Pouzite zdroje a nastroje

Zdroje

- Velky pruvodce protokoly TCP/IP a systemem DNS – L.Dostalek A.Kabelova
- V prezentaci jsou pouzity obrazky z vyse uvedene knihy.
- V prezentaci byly pouzity obrazky z knihy A.S.Tanenbaum: Computer Networks. Fourth edition. Pearson Education (Prentice Hall) 2003. ISBN 0-13-038488-7

Nastroje

- ethereal
- tcpdump
- OpenOffice

Dekuji za pozornost