TCP/IP protocol stack vizsgálata

Mérési jegyzőkönyv

v1.1

A mérést összeállította: Dr. Lencse Gábor, BME HIT, 2015.

A mérést végezték:	
A mérésvezető neve:	
A mérés helye:	
A mérési ideje:	
A jegyzőkönyv fájlneve: pl. KH1-szdu001-1.docx	

Tudnivalók:

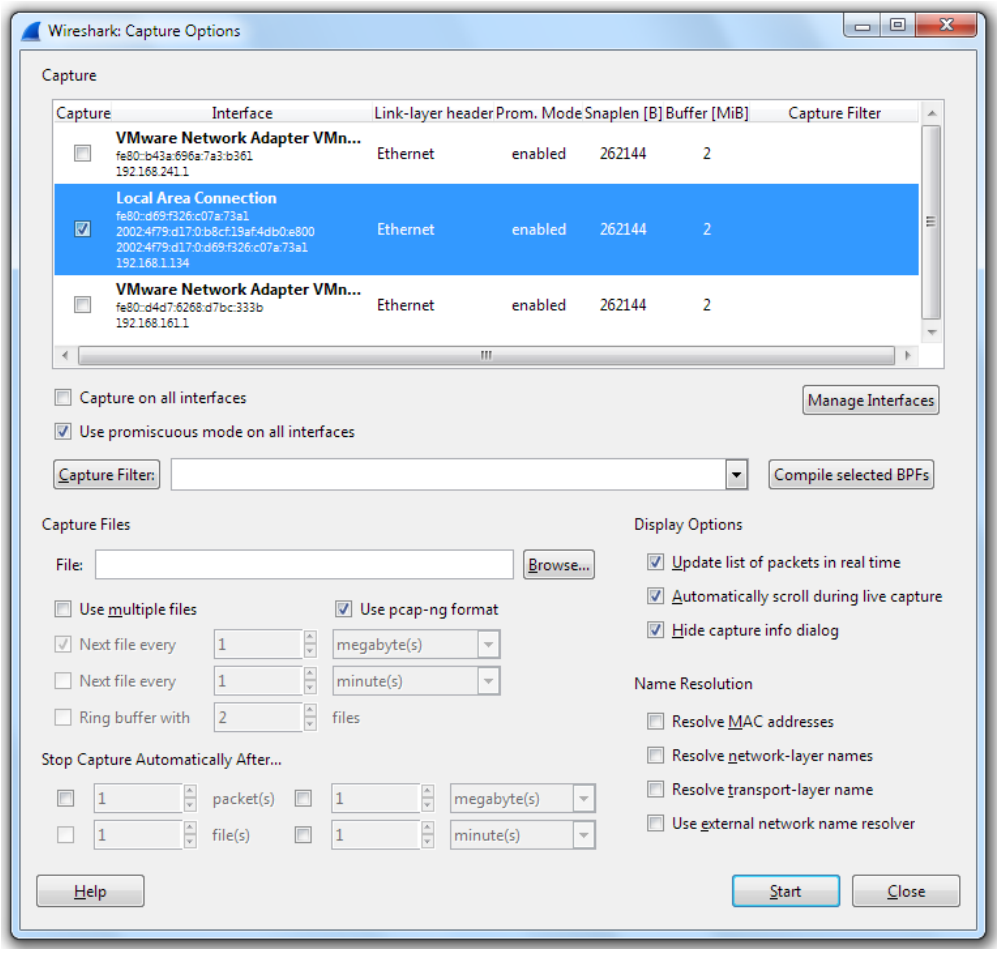
- A világossárga hátterű mezőket töltse ki értelemszerűen. Az „Egyéb megfigyelés” mezőt csak akkor töltse ki, ha van olyan érdekes megfigyelése, amit rögzítésre érdemesnek talál.
- Minden írott és elektronikus segédeszköz (pl. Internet is) használható.
- Törekedjen a mérőcsoportonként önálló munkára. Néhány feladatnál található „*Súgás*”, ezt csak akkor olvassa el, ha nélküle nem tudná a feladatot megoldani. Azonban ha valamit nem ért vagy elakad, akkor bátran kérjen segítséget a mérésvezetőjétől: ő azért van ott, hogy segítsen.
- Ne felejtse el, hogy a mérés célja az, hogy a mérést végző mindkét hallgató megértse a TCP/IP protocol stack működését, készség szintjén elsajátítsa a Wireshark kezelését, valamint gyakorlatot szerezzen a hálózati protokollok vizsgálatában. (Érdemes a számítógépet felváltva kezelni.)
- A jegyzőkönyvet úgy készítse el, hogy azt a mérés végén átnézi a mérésvezetője, és nem megfelelő felkészültség vagy szakmai színvonal esetén pótmérésre kötelezheti.
- Munka közben folyamatosan mentse a jegyzőkönyvet az `I:` meghajtóra, az utolsó mentésűt fogja a mérésvezető átnézni és értékelni. A mérés végén a jegyzőkönyvet mentse el saját magának is, hasznos lesz majd a vizsgára felkészülés során.
- Technikai tanács: mivel a jegyzőkönyvet nem kell kinyomtatni, nem kell takarékoskodni az oldalszámmal. Ha adott esetben egy logikai egység (például táblázat, részfeladat) közben oldaltörés lenne, az egység fölé tegyen be egy oldaltörést (Ctrl-Enter), hogy egyben lássa, ami összetartozik.
- *A 7-9. feladatok szorgalmi feladatnak számítanak. Teljesítésükkel plusz pontot szerezhet, ami a tárgyból a megajánlott jeles osztályzat egyik szükséges feltétele.*

1. feladat – A Wireshark alap szintű használatának megismerése

A feladat célja, hogy a hallgatók olyan szinten megismerjék a Wireshark protokoll analizátor program működését, hogy a további feladatoknál önállóan legyenek képesek használni azt. Ennek érdekében a Wireshark segítségével apró, konkrét feladatokat végeznek, melynek során különféle hálózati protokollokkal is találkoznak, de a fő cél most még nem azok tanulmányozása, tehát ne zavarja, ha nem érti mindenben a működésüket: később az is sorra fog kerülni!

Elvégzendő feladatok:

- Mivel a laborban IPv6 protokoll is működik, de most csak IPv4-gyel szeretnénk foglalkozni, ezért az érdemi munka megkezdése előtt az IPv6 protokoll használatát letiltjuk a hálózati interfészen. Ehhez nyissa meg a Start menü / Control Panel / All Control Panel Items / Network and Sharing Center ablakot, majd kattintson a Local Area Conntections feliratra, és a Properties / Networking fülön belül keresse meg: „Internet Protocol Version 6 (TCP/IPv6)”. Vegye ki előle a pipát és kattintson az OK gombra. (Ha más valaki már elvégezte ezt beállítást, akkor nincs teendője.)
- Indítsa el a Wireshark protokoll analizátor programot (Start menü / Wireshark).
- A kezdő képernyőn a Capture felirat oszlopában kattintson a „Capture Options” feliratra. A megnyíló dialógusablakban az interfészek listájából válassza ki a számítógépének fizikai hálókártyáját (Local Area Connection). (Legyen előtte pipa.) Majd ellenőrizze a többi beállítást, hogy az alábbiakban javasolt beállításoknak megfelelnek-e. (Amennyiben tisztában van az egyes beállítások jelentésével, bátran eltérhet a javaslattól.)



- Indítsa el a csomagelkapást a Start gomb megnyomásával.

- Miközben a Wireshark végzi a csomagelkapást, folytonosan meg is jeleníti az elkapott csomagokat.
  Segítség: Amennyiben nem jelenne meg a csomagok listája, akkor a View menü Packet list pontjának kiválasztásával tudja bekapcsolni a csomaglista megjelenítését.
  Szemrevételezéssel tekintse át a csomagok listájának oszlopaiban megjelenő információkat: sorszám, időpont, forráscím/célcím (IPv4 vagy IPv6 ha van, különben a link szintű protokoll „MAC" címei), protokoll (az adott adategységben szereplő legfelső), hossz, és végül egyéb információ: itt rövid összefoglalóként az adott adategységben szereplő legfelső szintű protokoll legfontosabb működését próbálja a Wireshark megjeleníteni.
- Miközben a Wireshark továbbra is végzi a csomagelkapást, indítson el egy Firefox böngészőt, és töltse le a http://www.bme.hu oldalt. Amikor az oldal letöltése befejeződött (nem kell az összes képet megvárni), állítsa le a csomagelkapást (a Wireshark menüsora alatti gombsor negyedik gombjával, ami egy piros négyzet).
- Szeretnénk kiszűrni a weboldal letöltésének forgalmából az egyéb háttérforgalmat. Ennek érdekében először derítse ki a saját számítógépének IP-címét. (A továbbiakban IP-cím alatt mindig IPv4 címet értünk, IPv6-tal majd a 3. mérésben foglalkozunk.) Nyisson meg egy Command Promptot (és a továbbiakban is hagyja megnyitva). Kérdezze le a Windows IP beállításait az **`ipconfig`** paranccsal. Állapítsa meg a fizikai interfészének IP-címét (Ethernet adapter Local Area Connection / IPv4 Address).

A számítógépem IP-címe:	**152.66.1.2**

- Kérdezze le a **`www.bme.hu`** webszerver IP-címét az **`nslookup www.bme.hu`** paranccsal.

A **`www.bme.hu`** gép IP-címe:	**152.66.115.203**

- Készítsen display filtert, amely csak azt a forgalmat jeleníti meg, amely a két gép között zajlik!

A display filter:	**ip.addr==152.66.1.2 && ip.addr==152.66.115.203**

- Gondolja végig még egyszer, hogy milyen forgalomnak kell megjelennie a fenti szűrés mellett (mindkét irány forgalma kell, és a két IP címnek a fentinek kell lennie), és ellenőrizze is a csomaglistában megjelenő forgalmat. Ha szükséges, kérjen segítséget a mérésvezetőjétől!
- Ha mindent hibátlanul végzett, akkor a csomaglistában legfelül megjelenő elem a helyi géptől a webszerver felé küldött TCP kapcsolat felépítésre irányuló kérés (SYN segment). (Amennyiben ez nem így van, akkor folytassa az előző ponttal.)
- Válassza ki a csomaglista első elemét, és ellenőrizze, hogy annak részletei (a benne található különböző protokollok fejrészeinek összefoglalói) megjelennek-e a csomaglista alatti ablakban.
  Segítség: Amennyiben nem jelennének meg csomag részletei, akkor a View menü Packet Details pontjának kiválasztásával tudja bekapcsolni a részletek megjelenítését.
  Tanulmányozza a csomag részleteit. Nyissa meg a legfelső sort (az elején található kis + jelre való kattintással), amelyben egy összefoglaló található azokról az adatokról, amit a Wireshark a csomagról nyilván tart.
- Nyissa meg a következő sort, amelyben az Ethernet fejrész mezői találhatók: célcím, forráscím és az Ethernet fölötti protokoll típusa (Type). Milyen számérték van a Type mezőben és az mit jelent?

A Type mező számértéke	**0x0800**
A következő protokollt jelenti:	**IPv4**

- Válassza ki a Type mezőt, és ellenőrizze az értékét a csomag bájtjait hexadecimális formában megjelenítő legalsó ablakrészben.
  Segítség: Amennyiben nem jelenne meg a csomag tartalma hexadecimálisan, akkor a View menü Packet Bytes pontjának kiválasztásával tudja bekapcsolni a megjelenítését.
  Milyen bájtsorrendet használnak ennek a kétbájtos adatnak a tárolására? (Jegyezze meg, hogy ez a hálózatos bájtsorrend a TCP/IP világban!)

A bájtsorrend megnevezése:	**network byte order / big endian**

- Nyissa meg az IP fejrészt, és azonosítsa az egyes mezőit. Keresse meg a csomagtartalom hexadecimális megjelenítésében azt a bájtot, amelyik a verziószám (Version) és a fejrész hossza (Header Length) értékét is tartalmazza. Magyarázza meg, hogy hogyan jön ki a Wireshark által megjelenített fejrész hossz (Header Length) érték!

A fejrész hossza:	**20**
A neki megfelelő bitmező által tárolt számérték:	**5**
A kettő közötti kapcsolat:	**5db 4-es oktettből áll össze**

- Az IP fejrész melyik mezőjéből derül ki, hogy az IP adatrészében milyen protokoll adategysége utazik?

A mező neve:	**Protocol**
A mező számértéke:	**6**
A számérték jelentése:	**TCP**

- Nyissa meg a TCP fejrészt, és tekintse át az egyes mezőit. Keresse meg a forrás- és cél portszámok értékét, valamit az opciók között a maximális szegmensméret értékét.

Source port:	**50172**
Destination port:	**80**
Options/ Maximum segment size:	**1460 bytes**

- Ha van olyan megfigyelése, amit érdemesnek talál rögzíteni, akkor azt tegye meg most. (Utána mással foglalkozunk, így nem fogja tovább látni ennek a csomagnak a tartalmát.)

Egyéb megfigyelés:	

- Most tesztelheti, hogy mennyire jól sikerült elsajátítania az eddigieket.
  Távolítsa el korábban beírt Display filtert és használja a következőt: ip.proto==17.
- Milyen szállítási szintű protokollt használnak a csomaglistában megjelenő csomagokban?

Szállítási protokoll neve:	**UDP**

- Puska: a DNS nem szállítási szintű protokoll (hanem alkalmazási szintű). Ha véletlenül ezt írta, akkor most javítsa ki!
  Súgás: Az IP protokoll melyik mezője adja meg a szállítási szintű protokollt?

- Válasszon ki egy olyan csomagot, amelyben DNS alkalmazási szintű protokoll utazik (a csomaglistában a Protocol mező „DNS”) és ezek közül is egy olyat, amelyben egy DNS kérés van (tehát nem válasz), ehhez használja az Info mezőt.
- Nyissa meg az UDP fejrészt és keresse meg a cél port számot

Destination port:	**53**

- Ha van olyan megfigyelése, amit érdemesnek talál rögzíteni, akkor azt tegye meg most.

Egyéb megfigyelés:	

- Amint eddig tapasztalta, a Display filterek segítségével a már elfogott forgalom csomagjai közül lehet kiválasztani azokat, amiket szeretnénk megjeleníteni. Ezzel szemben a Capture filterek azt adják meg, hogy mely csomagokat rögzítsen a Wireshark. Sajnos a kétféle szűrő szintaxisa teljesen más. Ízelítőül megnézünk egy olyan szűrést, amikor csak a http protokollt tartalmazó csomagokat rögzítjük. Ehhez használja a **`tcp port http`** Capture filtert és töltse le a sokkal egyszerűbb http://whale.hit.bme.hu weboldalt. (A Capture filtert a Wireshark nyomógomb sorának második gombjának (ha a kurzort fölé visszük, akkor a „Show the capture options…” szöveg jelenik meg) megnyomása után megjelenő dialógusdobozban a számítógép fizikai hálózati interfészének nevére történő dupla kattintás hatására megjelenő dialógusdobozban tudja megadni. A további lépéseket önállóan végezze.)
- Megjelentek a csomaglistában az elfogott csomagok?
  Segítség: Amennyiben nem jelentek meg a csomagok, akkor ellenőrizze, hogy eltávolította-e a csak UDP protokollt tartalmazó csomagok megjelenítéséhez használt Display filtert.
  A feladat elvégzését képernyőképpel dokumentálja itt lent:

- Ha van olyan megfigyelése, amit érdemesnek talál rögzíteni, akkor azt tegye meg most.

Egyéb megfigyelés:	

- Feltétlenül távolítsa el a beállított Capture filtert, mert különben gondot fog okozni a következő mérésnél!

2. feladat – Az ICMP protokoll használata

A feladat célja egyrészt az ICMP protokollal való kezdeti ismerkedés, másrészt az egyes rétegek egymásra épülésének, és az adott rétegbeli címek használatának a megértése.

Emlékeztető az ICMP protokoll elhelyezkedéséről:

Az ICMP protokoll minden IP implementáció kötelező része (tehát az ICMP nem szállítási szintű protokoll), de adategységei a TCP és UDP szállítási szintű protokollokhoz hasonlóan „az IP fölött" (tehát az IP datagram adat részében) utaznak.

Elvégzendő feladatok:

- Állapítsa meg a számítógépe fizikai Ethernet interfészéhez tartozó alapértelmezett átjáró IP-címét (**`ipconfig`** parancs, Ethernet adapter Local Area Connection / Default Gateway).

Default Gateway:	**152.66.1.254**

- Indítsa el a forgalom rögzítését a Wireshark programmal a számítógép fizikai interfészén, küldjön egy darab ICMP *echo request* üzenetet az alapértelmezett átjárónak (**`ping -n 1 <alapértelmezett átjáró IP-címe>`**[1]), majd egy másikat a whale.hit.bme.hu gépnek, (**`ping -n 1 whale.hit.bme.hu`**) végül állítsa le a forgalom rögzítését. Egy megfelelő display filter (icmp) segítségével jelenítse meg csak az ICMP üzeneteket.
- Töltse ki az alábbi táblázatot úgy, hogy az egyes mezőkbe ne neveket, hanem szükség szerint decimális vagy hexadecimális értékeket írjon:
  Súgás: ennek során meg kell nyitnia az egyes csomagok Ethernet, IP és ICMP fejrész mezőit, pusztán a csomaglista használata nem elégséges.
  Fontos technikai jótanács: Az egyes értékeket ne kézzel gépelje be, mert az nagyon hosszadalmas. Helyette Wiresharkban a kiválasztott mezőn az egér jobb gombja lenyomásának hatására megjelenő helyi menüben válassza ki a Copy funkciót, majd azt a lehetőséget, amit a vágólapra szeretne másolni (pl. Value).

		1. echo request	1. echo reply	2. echo request	2. echo reply
Ether-net	Source	**00:11:22:33:44:55**	**aa:bb:cc:dd:ee:ff**	**00:11:22:33:44:55**	**aa:bb:cc:dd:ee:ff**
	Destination	**aa:bb:cc:dd:ee:ff**	**00:11:22:33:44:55**	**aa:bb:cc:dd:ee:ff**	**00:11:22:33:44:55**
	Type	**IPv4 (0x0800)**	**IPv4 (0x0800)**	**IPv4 (0x0800)**	**IPv4 (0x0800)**
IP	Source	**152.66.1.2**	**152.66.1.254**	**152.66.1.2**	**152.66.248.88**
	Destination	**152.66.1.254**	**152.66.1.2**	**152.66.248.88**	**152.66.1.2**
	Protocol	**ICMP (1)**	**ICMP (1)**	**ICMP (1)**	**ICMP (1)**
ICMP	Type	**8**	**0**	**8**	**0**

A megfigyeltek alapján adjon választ a következő kérdésekre!

- Milyen protokoll fölött utaznak az ICMP üzenetek?

Az ICMP-t az … hordozza:	**IPv4**

- Miből derül ki, hogy ICMP üzenetről van szó (és nem TCP vagy UDP)?

1 Figyelem! A „<" és „>" jelek ún. metanyelvi zárójelek, azt a célt szolgálják, hogy valamilyen szöveges magyarázatot határoljanak, TILOS őket a parancsba beírni!

Az ICMP protokoll azonosító száma a IP fejrészben:	**1**

- Miből derül ki, hogy melyik ICMP üzenetről van szó?

Az ICMP üzenet fajtáját megadó mező neve az ICMP fejrészben:	**Type**

- Mi az oka annak, hogy a két megpingelt gép esetén (bár az IP címek eltérőek), a cél MAC címek azonosak?
  Segítség: a MAC és az IP címeket eltérő rétegben használjuk!

A két esetben a cél MAC címek azért egyeznek meg, mert:	**Az első esetben az alapértelmezett átjárót címeztük, második esetben pedig azt használtuk átjáróként.**

- Ha van olyan megfigyelése, amit érdemesnek talál rögzíteni, akkor azt tegye meg most.

Egyéb megfigyelés:	

3. feladat – Az ARP protokoll megismerése – első rész

A feladat célja az ARP protokoll működésének megismerése abban az esetben, amikor egy hálózati interfész MAC címének a kiderítésére használjuk. (További használatára még visszatérünk.)

Emlékeztető az ARP működéséről és megfontolás a teszteléshez:

Az ARP protokoll fő feladata egy adott IP-című hálózati interfész MAC-címének kiderítése. Fontos körülmény, hogy a kérdéses IP-cím gazdája a kérdezővel azonos hálózaton van, ezért az ARP protokollal kérdező fél *broadcast* segítségével képes azt elérni. A kérdéses IP-cím gazdája a választ már *unicast*tal küldi, hiszen a kérdező MAC címét ismeri. Az ARP protokoll cache-el, azaz bizonyos ideig tárolja az IP-cím MAC-cím párokat (ehhez fel tudja használni a broadcast címre küldött kérdéseket, amelyek tartalmazzák a kérdező fél IP-címét és MAC címét). Annak érdekében, hogy az ARP működését meg tudjuk figyelni, törölni fogjuk az ARP cache tábláját. Mivel ezt a törlést közvetlenül a mérés előtt kell elvégezni, készítünk egy batch fájlt, ami tartalmazza a törlés után a teszteléshez használt parancsot is. Mivel Windows 7 alatt az ARP cache tábla törléséhez rendszergazdai jogosultság szükséges, rendszergazdaként indítunk el egy Command promptot.

Elvégzendő feladatok:

- Indítson el rendszergazdaként egy Command promptot. (Például a Start menü keresőjében gépelje be, hogy cmd.exe, majd a megjelenő ikonra kattintson az egér jobb gombjával és a megjelenő helyi menüből válassza ki, hogy „Run as administrator”.)
- Azért, hogy ne „szemetelje” össze a Windows rendszert, váltson át a D: meghajtóra, készítsen magának egy könyvtárat és lépjen bele.
- Készítsen az aktuális könyvtárban a **`ping`** parancshoz egy batch fájlt **`myping1.bat`** néven a következő tartalommal (amely az ARP cache tábla törlése után egyetlen egy *ICMP echo request* üzenetet küld a paraméterként megadott számítógépnek, majd várja a választ):
  ```
  arp –d *
  ping -n 1 %1
  ```
- Indítsa el a Wiresharkkal a csomagelkapást a számítógép fizikai Ethernet interfészén, majd az elkészített batch fájl segítségével pingelje meg az alapértelmezett átjárót (ennek IP-címét az előző feladatban már kiderítette), végül állítsa le a csomagelkapást.
- Állapítsa meg a számítógép fizikai Ethernet interfészének a MAC-címét (**`ipconfig /all`** parancs, Ethernet adapter Local Area Connection / Physical Address).

MAC-cím:	**11:22:33:44:55:66**

- Készítsen olyan display filtert, ami kizárólag azokat a csomagokat jeleníti meg, amelynek a forrás vagy a cél MAC-címe fenti cím.

Display filter:	**eth.addr==11:22:33:44:55:66**

- Azonosítsa az ARP kérést és a választ, majd vizsgálja meg ennek a két üzenetnek a tartalmát. Töltse ki az alábbi táblázatot az *Ethernet* szintű információk alapján. A MAC címeknél a teljes 6 bájtos címet írja hexadecimálisan, ne a gyártó nevét. A keret fajtáját a *broadcast*, *multicast* és *unicast* lehetőségek közül válassza ki a cél MAC cím alapján. Az *Ethernet Type* mező értékét hexadecimálisan adja meg.

	forrás MAC címe	cél MAC címe	keret fajtája	Type értéke
ARP request	**11:22:33:44:55:66**	**ff:ff:ff:ff:ff:ff**	**broadcast**	**ARP (0x0806)**

ARP reply	**aa:bb:cc:dd:ee:ff**	**11:22:33:44:55:66**	**unicast**	**ARP (0x0806)**

- Ha helyesen dolgozott, akkor a Type oszlopban mindkét üzenetnél ugyanaz az érték szerepel. Akkor miből, hogyan derül ki, hogy melyik az ARP kérés és melyik a válasz?

Mely protokollban szerepel a kérdéses mező?	**ARP**
A mező neve:	**Opcode**
A mező értéke kérdésnél:	**1**
A mező értéke válasznál:	**2**

- Vizsgálja meg az ARP protokoll többi mezőjét is! Emlékeztetőül jegyezze fel az Opcode mező utáni 4 mező nevét (ezeket a következő feladatban használni fogjuk).

1. mező neve:	**Sender MAC address**
2. mező neve:	**Sender IP address**
3. mező neve:	**Target MAC address**
4. mező neve:	**Target IP address**

- A 4 mező közül az egyik értéke 0 a két ARP üzenet egyikében. Melyik mezőről és melyik ARP üzenetről van szó?

Mező neve:	**Target MAC address**
ARP üzenet:	**request**

- Ha van olyan megfigyelése, amit érdemesnek talál rögzíteni, akkor azt tegye meg most.

Egyéb megfigyelés:	

- Végezetül írassa ki az ARP cache tábla tartalmát (**`arp -a`**), és keresse meg benne az alapértelmezett átjáróhoz tartozó bejegyzést. Milyen típusú bejegyzés ez?

A bejegyzés típusa:	**dynamic**

4. feladat – Az ARP protokoll megismerése – második rész

A feladat célja az ARP protokoll működésének megismerése abban az esetben, amikor egy IPv4 cím egyediségének az ellenőrzésére (valamint a használatba vett IP-cím kihirdetésére) használjuk.

Emlékeztető az ARP Probe és az ARP Announcement működéséről:

Mielőtt egy eszköz egy IP-címet használatba venne, az *ARP Probe* üzenet segítségével tud meggyőződni arról, hogy az adott IP-címet nem használja-e már egy másik eszköz az adott hálózaton. Az ARP Probe üzenet egy speciális ARP Request, melyben a Sender Protocol Address mező 0.0.0.0. értéket tartalmaz azért, hogy az üzenetet vevő állomások ne tárolják el az IP-cím – MAC-cím párt, hiszen ha más valaki már használja a kérdéses IP-címet, akkor ez egy hibás összerendelés lenne (ARP cache szennyezés). Amennyiben kérdező fél nem kap választ (többszöri kérdés esetén sem), akkor *ARP Announcement* üzenet segítségével mindenkit értesít arról, hogy mostantól használni fogja az adott IP-címet. Az ARP Announcement egy speciális ARP Request, amelyben a Sender Protocol Address és a Target Protocol Address mezőben ugyanaz a kihirdetni kívánt IP-cím található. (Mivel ez is egy ARP Request, amit broadcast címre küldenek, ezért az összes állomás veszi és eltárolja a Sender Protocol Address és a Sender Hardware Address mezőkben található IP-cím – MAC-cím párt).

Elvégzendő feladatok:

- Állítsa át a számítógépe IP-címét manuálisan a következőre:
  172.16.<mérőhely száma>.<mérőhely száma>, a hálózati maszk pedig legyen: 255.255.255.0.
  Segítség: Start menü / Control Panel / All Control Panel Items / Network and Sharing Center, majd: Local Area Conntections / Properties / Networking fülön belül keresse meg:
  Internet Protocol Version 4 (TCP/IPv4) és ennek a tulajdonságai (Properties) között az „Obtain an IP address automatically" helyett válassza, hogy „Use the following address", és itt tudja beállítani az IP-címet és a maszkot. Mást nem kell beállítania; de számítson rá, hogy az Internet elérhetetlenné válik. Még mielőtt a beállítást elmentené, indítson el a Wiresharkban csomagelkapást a számítógépe fizikai interfészén! (Utána pedig kb. 10 másodperc múlva állítsa le.)
  (Ha véletlenül valamit elrontott, és ezért meg kell ismételnie a feladatot, akkor adjon hozzá 100-at a mérőhely számához. Harmadik próbálkozás ismét az eredeti értékkel mehet.)
- Egy megfelelő display filter (arp) segítségével jelenítse meg csak az ARP üzeneteket.
- Hány darab ARP Probe üzenetet talált? Az egymást követő ARP probe üzenetek küldése között mennyi idő telt el? Az utolsó ARP Probe után mennyi idő múlva küldte a számítógép az ARP Announcement üzenetet? Hogy nevezi a Wireshark az ARP Announcement üzenetet?

ARP Probe üzenetek száma:	**3**
Ismétlési idő köztük:	**1s**
ARP Announcement előtt eltelt idő:	**1s**
ARP Announcement üzenetet a Wireshark így nevezi:	**gratuitous ARP**

- Ha van olyan megfigyelése, amit érdemesnek talál rögzíteni, akkor azt tegye meg most.

Egyéb megfigyelés:	

- Bár az Internet most elérhetetlen, a hálózati beállításokat hagyja így, mert kiindulásként erre lesz szüksége a következő feladatban.

5. feladat – A DHCP protokoll megismerése

A feladat célja a DHCP protokoll működésének megismerése. Először az IP-cím beszerzésének folyamatát, majd az IP-cím megújítását, végül az idő előtti visszaadását vizsgáljuk meg.

Emlékeztető a DHCP protokoll elhelyezkedéséről és címhasználatáról:

A DHCP protokoll az alkalmazási rétegben működik. Üzenetei BOOTP üzenetekbe ágyazva, annak opcióiként jelennek meg. A BOOTP üzenetek IP fölött, UDP-be ágyazva haladnak. Amíg a kliensnek nincs érvényes IP-címe, addig 0.0.0.0-t használ. Broadcast esetén IP szinten természetesen a 255.255.255.255 címre küldi az üzenetet (Ethernet szinten pedig az FF:FF:FF:FF:FF:FF címre). UDP-ben a kliens portszáma: 68, a szerveré: 67.

Elvégzendő feladatok:

- Állítsa át a számítógépen az IP-cím beállítását manuálisról DHCP-re.
  Segítség: Az előbbi dialógusdobozban a „Use the following address" helyett válassza, hogy: „Obtain an IP address automatically". De még mielőtt a beállítást elmentené, indítson el a Wiresharkban csomagelkapást a számítógépe fizikai interfészén! (Utána pedig kb. 10 másodperc múlva állítsa le.)
- Egy megfelelő display filter (bootp) segítségével jelenítse meg csak a BOOTP üzeneteket.
- Törölje a display filtert, és jelenítse meg pontosan ugyanezeket az üzeneteket úgy, hogy most nem használhatja a fenti szűrőt, hanem saját szűrőt alkot. Törekedjen minél rövidebb megoldásra.
  Súgás: használja fel, hogy a BOOTP üzenetek a 67-es és 68-os UDP portok között utaznak, és ezeket a portokat más alkalmazások nem használják.

Saját display filter:	**udp.port==67 || udp.port==68**

- Tanulmányozza az IP-cím megszerzéséhez használt 4 üzenetet, és töltse ki az alábbi táblázatot.

	forrás MAC-cím	cél MAC-cím	forrás IP-cím	cél IP-cím
DHCP Discover	**11:22:33:44:55:66**	**ff:ff:ff:ff:ff:ff**	**0.0.0.0**	**255.255.255.255**
DHCP Offer	**00:ff:11:ee:22:dd**	**11:22:33:44:55:66**	**152.66.1.253**	**255.255.255.255**
DHCP Request	**11:22:33:44:55:66**	**ff:ff:ff:ff:ff:ff**	**0.0.0.0**	**255.255.255.255**
DHCP ACK	**00:ff:11:ee:22:dd**	**11:22:33:44:55:66**	**152.66.1.253**	**255.255.255.255**

- Gondolkozzon el rajta, hogy miért ezek az értékek kerültek a táblázatba! Fogalmazza meg megfigyeléseit az alábbi kérdések segítségével!
- A DHCP Discover küldésekor van-e a gépnek érvényes IP címe, illetve tudja-e, hogy kihez forduljon?

Megállapításaim:	**Nincs érvényes IP cím, mindenkinek küldi a kérést.**

- A DHCP Offer küldésekor az ajánlatot tevő szerver ismeri-e a kérést küldő gép MAC címét?

Megállapításaim:	**Ismeri a kérés küldő gép MAC címét mivel a kérés fejlécében szerepelt.**

- A DHCP Request küldésekor a kapott ajánlat alapján (és más ajánlat hiányában) a felajánlott címet megigénylő gép használhatja-e már az ajánlatban szereplő címet?

Megállapításaim:	**Még nem**

- Ha van olyan megfigyelése, amit érdemesnek talál rögzíteni, akkor azt tegye meg most.

Egyéb megfigyelés:	

- Tekintse most a DHCP ACK üzenetet. Mennyi időre kapta meg az IP-címet? Milyen további azonosítókat kapott még?

Bérleti idő:	**4 óra**
Hálózati maszk:	**255.255.255.128**
Router IP címe:	**152.66.1.254**
DNS szerver(ek):	**152.66.11.1** **152.66.12.1**

- Ha van olyan megfigyelése, amit érdemesnek talál rögzíteni, akkor azt tegye meg most.

Egyéb megfigyelés:	

- A számítógépek az IP-címüket annak lejárati ideje előtt automatikusan megújítják. A tanulmányozás érdekében ezt a folyamatot most kézzel fogjuk kiváltani. Indítson el csomagelkapást a számítógép fizikai interfészén, aztán adja ki a következő parancsot: **`ipconfig /renew`**, majd körülbelül 10 másodperc múlva állítsa le a csomagelkapást. (Most használhatja a bootp szűrőt is.)
- Hány DHCP üzenetet lát? Milyen IP-címet használt a kliens? Miért tehette ezt meg? Milyen címre küldte a kérését?

1. DHCP üzenet típusa:	**Request**
2. DHCP üzenet típusa:	**ACK**
Kliens IP-címe:	**152.66.1.2**
Miért használhatja?	**Mert már korábban megkaptuk, és érvényes még a bérlet.**
Kliens üzenetében a cél IP-cím:	**152.66.1.253**

- Végül megnézzük, hogyan adhatjuk vissza az IP címünket a bérleti idő lejárta előtt. Ehhez indítson el csomagelkapást a számítógép fizikai interfészén, aztán adja ki a következő parancsot: **`ipconfig /release`**, majd körülbelül 10 másodperc múlva állítsa le a csomagelkapást. (Most is használhatja a bootp szűrőt.)
- Először is ellenőrizze a parancs kimenetét. Van-e most érvényes IP-címe a számítógépének?

Megfigyelésem:	**Itt nincs IP címe**

- Kérdezze le újra a számítógépe IP-címét az **`ipconfig`** paranccsal. Mit tapasztal? Miféle IP-cím ez?

Megfigyelésem:	**Lett IP címe**
Az IP-cím értéke:	**169.254.123.45**
Az IP-cím fajtája:	**link-local cím**

- Búcsúzóul nézze meg a DHCP Release üzenet tartalmát.
- Ha van olyan megfigyelése, amit érdemesnek talál rögzíteni, akkor azt tegye meg most.

Egyéb megfigyelés:	

- Állítsa helyre a számítógépén a hálózat működőképességét egy **`ipconfig /renew`** paranccsal.

FONTOS: A fenti feladatok elvégzése után már rendelkezik elegendő ismerettel és gyakorlattal ahhoz, hogy a számítógépe hálózati interfészeinek a beállítását önállóan elvégezze, karakteres parancsokat hajtson végre, illetve a Wireshark protokollanalizátort önállóan használja, ezért számítson arra, hogy a továbbiakban nem fogunk minden lépést (pl. használjon szűrőt) „szájbarágós” módon megadni.

6. feladat – TCP kapcsolat felépítés és bontása

A feladat célja a TCP kapcsolatok felépítésének és bontásának megismerése.

Emlékeztető a TCP működéséről:

A TCP mindkét irányban megbízható átvitelt nyújtó protokoll. A megbízható átvitel érdekében az átvitt oktetteket (az oktett a bájt megnevezése a TCP/IP terminológiában) sorszámozza és nyugtázza. Egy TCP kapcsolat felépítésekor szükséges, hogy a felek mindkét irányú átvitelt illetően egyeztessék a kezdő sorszámot. Ezenkívül a *forgalomszabályozás*hoz (flow-control) használt *ablakméret*et (Window size) is egyeztetik mindkét irányra, ami tulajdonképpen egy hitelkeret: az adott fél ennyi oktettet küldhet a már nyugtázott oktetteken túl. Ezeket az egyeztetéseket az ún. *három utas kézfogás*sal (3-way handshake) oldják meg. A kapcsolat lebontásának is rendezetten kell megtörténnie, ez a *négy utas kézfogás* (4-way handshake).

Elvégzendő feladatok:

- Indítsa el a forgalom rögzítését a Wireshark programmal, Web böngészővel nyissa meg a http://whale.hit.bme.hu oldalt, és utána várjon még egy percet, mielőtt leállítja a forgalom rögzítését. (Közben már tanulmányozhatja a csomaglistában a forgalmat. Feltételezzük, hogy szűrőket önállóan használ, amikor szüksége van rá.)
- Azonosítsa a három utas kézfogás lépéseit. Milyen vezérlőbitek aktívak az egyes lépések esetén?

1. lépésben (K-->S) aktív:	**SYN**
2. lépésben (S-->K) aktív:	**SYN és ACK**
3. lépésben (K-->S) aktív:	**ACK**

- Vizsgálja meg a 3 szegmens TCP fejrészét. Naplózza a sorszám (valóságos) kezdőértékét mindkét irányban. (A Wireshark ehhez képest relatív sorszámot jelenít meg, de most a ténylegesen elküldött értékre vagyunk kíváncsiak.)

Sorszám a K-->S irányban:	**2896665622**
Sorszám a S-->K irányban:	**161460765**

- Naplózza a nyugta valóságosan elküldött értékét is mindkét irányban.

A K-->S irányra vonatkozó, de éppen ezért a S-->K irányban küldött nyugta értéke:	**2896665623**
A S-->K irányra vonatkozó, de éppen azért K-->S irányban küldött nyugta értéke:	**161460766**

- A fentiek alapján adja meg a nyugta mező értékének értelmezését szövegesen.

A nyugta mező *n* értéke az jelenti, hogy:	**A nyugtázott csomag sorszámához egyet hozzáadunk.**

- Ha van olyan megfigyelése, amit érdemesnek talál rögzíteni, akkor azt tegye meg most.

Egyéb megfigyelés:	

- Azonosítsa a négy utas kézfogás lépéseit. Milyen vezérlőbitek aktívak az egyes lépések esetén?

1. lépésben aktív:	**FIN és ACK**
2. lépésben aktív:	**ACK**
3. lépésben aktív:	**FIN és ACK**
4. lépésben aktív:	**ACK**

- Ha van olyan megfigyelése, amit érdemesnek talál rögzíteni, akkor azt tegye meg most.

Egyéb megfigyelés:	

7. feladat – Adott cél felé használt útválasztók vizsgálata

A feladat célja a ***traceroute*** *parancs működésének megismerése.*

Emlékeztető a **traceroute** működéséről:

A **traceroute** feladata annak kiderítése, hogy egy adott cél felé milyen útválasztókon (router) keresztül jut el egy datagram, valamint ezek válaszidejének a meghatározása. A **traceroute** ennek érdekében UDP adatcsomagot szokott küldeni a célként megadott gép felé 1-től 1-esével növekvő TTL értékkel: minden értékkel 3 adatcsomagot (olyan célport számra, ahol nem figyel alkalmazás). A kapott hibaüzenet fajtája alapján tudja eldönteni, hogy az egy közbülső útválasztótól vagy a céltól jött.

Eltérések a Windowsban:

1. A program neve **tracert.exe**. (Ez a DOS-ból származó legfeljebb 8 karakteres fájlnévhossz maradványa.)
2. A program nem UDP csomagot, hanem ICMP echo request üzenetet használ.

Elvégzendő feladatok:

- Indítsa el a forgalom rögzítését a Wireshark programmal, adja ki a **tracert whale.hit.bme.hu** parancsot, figyelje meg a program kimenetét, végül állítsa le a forgalom rögzítését. Jegyzőkönyvezze a kiadott parancsokat és azok kimenetét.

A kiadott parancsok és kimenetük:	**tracert whale.hit.bme.hu** **Tracing route to whale.hit.bme.hu [152.66.248.88]** **over a maximum of 30 hops:** **1 1 ms 1 ms <1 ms rtr1.net.bme.hu [152.66.1.254]** **2 1 ms 1 ms 1 ms whale.hit.bme.hu [152.66.248.88]** **Trace complete.**

- Vizsgálja meg a Wireshark által rögzített forgalmat! Keresse meg a helyi géptől a **whale.hit.bme.hu** gép felé küldött első ICMP echo request üzenetet. Ennek alapján töltse ki az alábbi táblázatot.

Az IP fejrész mezői	Source	152.66.1.2
	Destination	152.66.248.88
	Protocol	ICMP (1)
Az ICMP protokoll mezői	Type	8 (Echo (ping) request)
	Code	0
	Identifier (Big Endian)	1 (0x0001)
	Sequence no. (Big Endian)	21 (0x0015)

- Keresse meg a fenti ICMP echo request üzenet által kiváltott ICMP hibaüzenetet! A hibaüzenet mely része alapján tudja biztosan, hogy a fenti datagram váltotta ki?
  Súgás: használja a fenti táblázat értékeit!

Az azonosításhoz használt rész megnevezése:	**A sequence number megegyezik.**

- Keresse meg az utolsó olyan ICMP echo request üzenetet, amelyre válaszként "Time-to-live exceeded" ICMP hibaüzenet érkezett! Mennyi az IP fejrész TTL mezőjének értéke?

IP TTL értéke:	**1**

- Mennyi az IP fejrész TTL mezőjének értéke a következő ICMP echo request üzenetnél?

IP TTL értéke:	**2**

- Erre milyen üzenetet küldött vissza a **`whale.hit.bme.hu`** gép?

ICMP Type:	**0 (Echo reply)**

- Mi az oka ennek az üzenetnek?

Az üzenet magyarázata:	**2 ugrás távolságra van, ezért megkapta az kérést.**

- Összefoglalásul fogalmazza meg, hogy hogyan használja fel a **`tracert`** parancs a működése során a kapott ICMP üzeneteket? (Mit jelentenek ezek számára, melyiknél mit kell tennie?)

A kapott ICMP üzenettől függően a teendő:	**TTL üzenet esetén újabb kérést küld 1-gyel nagyobb TTL-lel. Echo reply üzenet esetén megáll.**

- Mennyi az IP protokoll esetén a TTL maximális kezdőértéke? (Segítség: a TTL mező 8 bites.)
- A **`tracert`** parancs kimenete alapján a helyi géptől az **`whale.hit.bme.hu`** gépig hány útválasztón halad keresztül a datagram?

TTL legnagyobb kezdőértéke:	**255**
Érintett útválasztók száma:	**1**

- Adjon ki egy **`ping whale.hit.bme.hu`** parancsot, majd jegyzőkönyvezze a kiadott parancsot és kimenetét.

A kiadott parancs és kimenete:	**C:\Users\student>ping whale.hit.bme.hu** **Pinging whale.hit.bme.hu [152.66.248.88] with 32 bytes of data:** **Reply from 152.66.248.88: bytes=32 time=1ms TTL=254** **Reply from 152.66.248.88: bytes=32 time=1ms TTL=254** **Reply from 152.66.248.88: bytes=32 time=1ms TTL=254** **Reply from 152.66.248.88: bytes=32 time=1ms TTL=254** **Ping statistics for 152.66.248.88:** **Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),** **Approximate round trip times in milli-seconds:** **Minimum = 1ms, Maximum = 1ms, Average = 1ms**

- Mennyi a **`ping`** által kiírt TTL érték a visszaérkező csomagnál?

A **`ping`** által kiírt TTL:	**254**

- Milyen összefüggést talál a TTL lehetséges legnagyobb kezdőértéke, az érintett útválasztók száma és a **`ping`** által kiírt TTL érték között? Magyarázza is meg!

Összefüggés és magyarázat:	**Maximális TTL-lel indult a válasz, az útválasztó ezt 1-gyel csökkentette.**

- Ha van olyan megfigyelése, amit érdemesnek talál rögzíteni, akkor azt tegye meg most.

Egyéb megfigyelés:	

- A következőkben két távolabbi szervert fogunk tesztelni. Az első a **`www.tilb.sze.hu`** gép. Vizsgálja meg a **`tracert`** és a **`ping`** parancsok eredményét.

A kiadott parancsok és kimenetük:	**C:\Users\student>tracert www.tilb.sze.hu** **Tracing route to www.tilb.sze.hu [193.224.130.173]** **over a maximum of 30 hops:** **1 1 ms 1 ms 1 ms rtr1.net.bme.hu [152.66.1.254]** **2 1 ms 1 ms 1 ms rtr2.net.bme.hu [152.66.0.72]** **3 1 ms 1 ms * rtr3.net.bme.hu [152.66.0.78]** **4 2 ms 1 ms 1 ms tg0-1-0-1.rtr.bme.hbone.hu [152.66.0.126]** **5 5 ms 4 ms 4 ms tg0-0-0-6.rtr1.vh.hbone.hu [195.111.100.43]** **6 3 ms 3 ms 3 ms tg0-0-0-0.rtr.tatabanya.hbone.hu [195.111.111.253]** **7 4 ms 4 ms 4 ms eth1-1.rtr.gyor.hbone.hu [195.111.111.150]** **8 19 ms 4 ms 4 ms wsc6k.sze.hu [193.224.129.17]** **9 4 ms 4 ms 4 ms ns.tilb.sze.hu [193.224.128.28]** **10 4 ms 4 ms 4 ms paloalto.tilb.sze.hu [193.225.151.66]** **11 4 ms 4 ms 4 ms www.tilb.sze.hu [193.224.130.173]** **Trace complete.** **C:\Users\student>ping www.tilb.sze.hu** **Pinging www.tilb.sze.hu [193.224.130.173] with 32 bytes of data:** **Reply from 193.224.130.173: bytes=32 time=4ms TTL=54** **Reply from 193.224.130.173: bytes=32 time=4ms TTL=54** **Reply from 193.224.130.173: bytes=32 time=4ms TTL=54** **Reply from 193.224.130.173: bytes=32 time=4ms TTL=54** **Ping statistics for 193.224.130.173:** **Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),** **Approximate round trip times in milli-seconds:** **Minimum = 4ms, Maximum = 4ms, Average = 4ms**

- Feltéve, hogy a korábban megállapított összefüggés itt is érvényes, a **`www.tilb.sze.hu`** gép milyen IP TTL kezdőértékkel küldi az ICMP echo reply üzeneteket?

IP TTL kezdőértéke most:	**64**

- Most tesztelje a **`www.inf.unideb.hu`** gépet is. Mit tapasztalt? Mi lehet ennek az oka? Egészítse ki a **`tracert`** parancs működésének megfogalmazását is.

A kiadott parancsok és kimenetük:	**C:\Users\student>tracert www.inf.unideb.hu** **Tracing route to w6.inf.unideb.hu [193.6.135.136]** **over a maximum of 30 hops:** **1 1 ms 1 ms 1 ms rtr1.net.bme.hu [152.66.1.254]** **2 1 ms 1 ms 1 ms rtr2.net.bme.hu [152.66.0.72]** **3 * 1 ms * rtr3.net.bme.hu [152.66.0.78]** **4 2 ms 1 ms 1 ms tg0-1-0-1.rtr.bme.hbone.hu [152.66.0.126]** **5 5 ms 7 ms 14 ms tg0-0-0-6.rtr1.vh.hbone.hu [195.111.100.43]** **6 6 ms 5 ms 5 ms be1.rtr.debrecen.hbone.hu [195.111.111.218]** **7 * * * Request timed out.** **8 * * * Request timed out.** **9 * * * Request timed out.** **10 * * * Request timed out.** **11 * * * Request timed out.** **12 * * * Request timed out.** **13 * * * Request timed out.** **14 * * * Request timed out.** **15 * * * Request timed out.** **16 * * * Request timed out.** **17 * * * Request timed out.** **18 * * * Request timed out.**
A jelenség és magyarázata:	**Egy tűzfal eldobta az ICMP kéréseinket.**
A **`tracert`** parancs válasz hiányában:	**Továbbhalad**
És befejezi a működését, ha:	**Eléri a maximális hopszámot.**

- Ha van olyan megfigyelése, amit érdemesnek talál rögzíteni, akkor azt tegye meg most.

Egyéb megfigyelés:	

8. feladat – TCP window scaling

A feladat fő célja a TCP window scaling vizsgálata. Egyben látunk egy példát TCP opciók használatára is.

Emlékeztető a TCP window scaling működéséről:

A TCP fejrészben a forgalomszabályozáshoz használt vételi ablak (receive window) méretét megadó *Window* mező 16 bites. Megfelelően nagy átviteli késleltetés és adatsebesség esetén a 16 biten kifejezhető legnagyobb egész szám (65535) túl kicsi vételi ablak méretet tesz lehetővé. Az *ablak skálázás* (window scaling) használatával az ablakméret megnövelhető. Ha mindkét fél támogatja a megoldást, akkor a TCP fejrészben szereplő ablakméret (Window) mező értékét az opcióban megadott mértékben kell balra tolni a ténylegesen használt ablakméret kiszámításához. (Például n=8 esetén a 8 balra tolás 2^8=256-tal való szorzásnak felel meg.) A window scaling opciót a felek csak a kapcsolat felépítésekor jelzik egymásnak (amikor a SYN flag aktív). Ennek értéke irányonként eltérő is lehet. A Wireshark kényelmi funkciója, hogy megjegyzi ezt az értéket, és számunkra a csomaglistában (a relatív nyugata értékhez hasonlóan) a 2^n*Window képlet szerint kiszámított ablakméretet jeleníti meg.

A vizsgálat során egy távoli, kellően nagy átviteli sebességgel rendelkező szervert fogunk használni.

Elvégzendő feladatok:

- Először nyissa meg egy böngészőben a http://dev.tilb.sze.hu/TCP/ oldalt, aztán indítson el egy Wireshark csomagelkapást, majd a böngészőben kattintson rá a 100MB nevű (és méretű) fájlra. Válassza a fájl elmentését, azután várja meg, amíg a fájl teljesen letöltődik, majd állítsa le a csomagelkapást. (Amennyiben a letöltés 1 percnél tovább tartana, akkor kb. 1 perc után állítsa le a letöltést és a csomagelkapást is.)
- Keresse meg a kapcsolat felépítéséhez használt három utas kézfogás lépéseit, és irányonként határozza meg a *window scaling factor* értékét.

	mező értéke (n) (Shift count)	szorzó értéke (2^n) (Multiplier)
Kliens üzeni a szervernek, tehát a letöltésre vonatkozik	**8**	**256**
Szerver üzeni a kliensnek, tehát a feltöltésre vonatkozik	**6**	**64**

- Keressen egy olyan TCP szegmenst, amire illeszkedik a tcp.window_size>70000 szűrő. Vizsgálja meg a TCP szegmens fejrészében a Window mező tényleges értékét, és a számítás helyességét. (Nyissa meg a TCP fejrészt, álljon rá annak Window mezőjére (ezt a Wireshark úgy nevezi, hogy: window size value), és nézze meg a mező hexadecimális értékét.)

A TCP fejrészben a Window mező értéke hexadecimálisan:	**0189**
A Wireshark által kiírt decimális érték (window size value)	**393**
A Wireshark által kiszámított ablakméret (Calculated window size)	**100608**
Helyes-e a számítás?	**Helyes**

- Ha van olyan megfigyelése, amit érdemesnek talál rögzíteni, akkor azt tegye meg most.

Egyéb megfigyelés:	

9. feladat – TCP kapcsolatok számának meghatározása

A feladat fő célja az önálló problémamegoldás gyakorlása. Továbbá érdekes megfigyelés, hogy egy web böngésző egy összetett weboldal letöltésekor párhuzamosan több TCP kapcsolatot használ, és ezek száma különféle böngészők esetén eltérő lehet.

Megoldandó feladat:

- Wireshark segítségével vizsgálja meg, hogy a Firefox, a Google Chrome, és az Internet Explorer böngészők hány TCP kapcsolatot használnak a http://whale.hit.bme.hu/tesztalbum letöltéséhez. A feladatot lehetőleg önállóan oldja meg. (Amennyiben 3 perc elteltével sincs ötlete, hogy hogyan oldja meg a feladatot, akkor megnézheti a következő oldalon található súgást). A megoldáshoz használt megjelenítési szűrőt és az eredményeket az alábbiakban rögzítse.

Display Filter:	**tcp && tcp.flags.syn==1**

	Firefox	Google Chrome	Internet Explorer
TCP kapcsolatok száma	**6**	**6**	**6**

- Ha van olyan megfigyelése, amit érdemesnek talál rögzíteni, akkor azt tegye meg most.

Egyéb megfigyelés:	

SÚGÁS: Képzelje el, hogy valahány kígyót darabora vágtak és darabjaikat (az összeset) kiterítették egy ponyvára. Azt kell megmondania, hogy hány kígyó darabjai vannak ott. Mit fog megszámolni? Most gondolkozzon el rajta, és ne olvassa tovább!

Ha ez még nem volt elég, akkor további súgás: elegendő csak a kígyófejeket megszámolni. (Természetesen a kígyófarkak megszámolása is jó megoldás.) A TCP kapcsolatok esetén akkor mit kell megszámolnia? Ügyeljen arra is, hogy csak azokat a TCP kapcsolatokat számolja meg, amelyek az adott oldal letöltéséhez tartoznak, és persze mindegyiket csak egyszeresen vegye figyelembe!