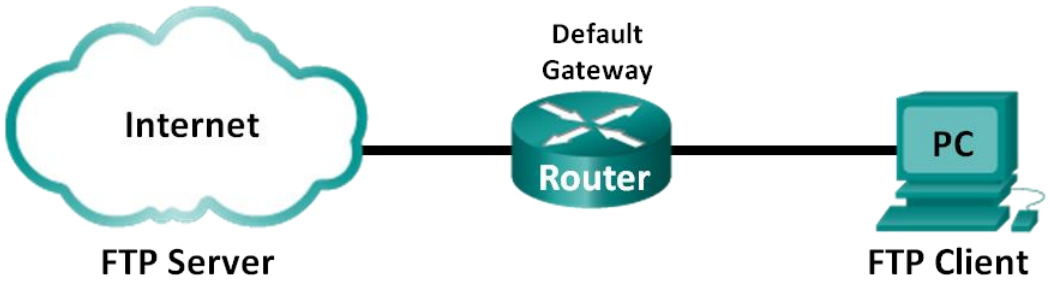
Laborator - Folosirea Wireshark-ului pentru a Examina Capturile FTP și TFTP

Topologie – Partea 1 (FTP)

Partea 1 va evidenția o captură TCP a unei sesiuni FTP.Această topologie este formată dintr-un calculator cu acces la Internet.



Topologie – Partea 2 (TFTP)

Partea 2 va evidenția o captură UDP a unei sesiuni TFTP.Calculatorul trebuie să aibă și o conexiune Ethernet și o conexiune de consolă la Switchul S1.

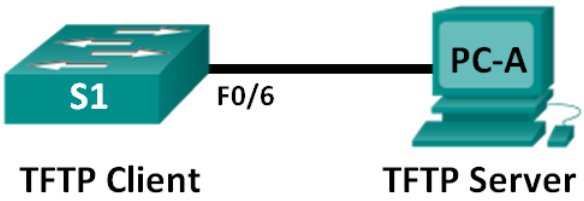


Tabela de Adresare (Partea 2)

Echipament	Interfață	Adresă IP	Subnet Mask	Default Gateway
S1	VLAN 1	192.168.1.1	255.255.255.0	N/A
PC-A	Placă de reţea	192.168.1.3	255.255.255.0	192.168.1.1

Obiective

Partea 1: Identificarea câmpurilor din antetul TCP și operare utilizând o captură Wireshark

Partea 2: Identificarea câmpurilor antetului UDP și operare utilizând o captură Wireshark

Context/Scenariu

Cele două protocoale din layer-ul de transport TCP/IP sunt TCP, definit în RFC 761 și UDP, definit în RFC 768.Ambele protocoale suportă comunicarea protocolului de layer superior.De exemplu, TCP este utilizat pentru a asigura suport layer-ului transport pentur protocoalele FTP și HTTP, împreună cu altele.UDP asigură suport layer-ului transport pentru DNS și TFTP împreună cu altele.

Notă: Înțelegerea componentelor din header-ul TCP și UDP și funcționarea acestora este o aptitudine importantă pentru inginerii de rețea.

În Partea 1 a acestui laborator, veți utiliza instrumentul open sourse Wireshark pentru a captura și analiza header-ul protocolului TCP pentru transferuri de fișiere FTP între calculatorul host și un server FTP anonim.Utilitarul liniei de comandă din Windows este utilizat pentru a conecta un server FTP anonim și pentru a descărca un fișier.În Partea 2 a acestui laborator veți utiliza Wireshark pentru a captura și analiza câmpurile header-ului UDP pentru transferuri de fișiere TFTP între calculatorul host și switchul S1.

Notă: Switchul folosit este un Cisco Catalyst 2960s cu Cisco IOS Release 15.0(2) (lanbasek9 image).Pot fi folosite și switchuri și versiuni IOS.În funcție de model și de versiunea Cisco IOS, comenzile disponibile și rezultatele produse pot fi diferite față de cele arătate la laboratoare.

Notă: Asigurați-vă că switchul a fost șters și că nu au configurații de pornire.Dacă nu sunteți sigur, contactați-vă instructorul.

Notă: Partea 1 presupune că există acces la Internet pe calculator și nu poate fi realizată folosind Netlab.Partea 2 este compatibilă cu Netlab.

Resurse Necesare – Partea 1 (FTP)

1 calculator (cu Windows 7, Vista sau XP cu acces la ecranul de comandă și la Internet și care să aibă instalat Wireshark)

Resurse Necesare – Partea 2 (TFTP)

- 1 Switch (Cisco 2960 cu Cisco IOS Release 15.0(2) imagine lanbasek9 sau comparabilă)
- 1 calculator (cu Windows 7, Vista sau XP cu Wireshark și un server TFTP, cum ar fi tftpd32)
- Cabluri de consolă pentru a configura echipamentele Cisco IOS prin intermediul porturilor de consolă
- Cablu Ethernet așa cum se arată în topologie

Partea 1: Identificați câmpurilor din antetul TCP și operare utilizând o captură Wireshark

În Partea 1, veți utiliza Wireshark pentru a captura o sesiune FTP și veți inspecta câmpurile header-ului TCP.

Pasul 1: Porniți o captură Wireshark.

a. Închideți traficul de rețea inutil, precum navigatorul web, pentru a limita traficul în timpul capturii Wireshark.

b. Porniți o captură Wireshark.

Pasul 2: Descărcați fișierul Readme.

c. Din ecranul de comandă tastați ftp ftp.cdc.gov.

d. Autentificați-vă la site-ul FTP pentru CDC (Centers for Disease Control and Prevention) cu utilizatorul **anonymus** și fără parolă.

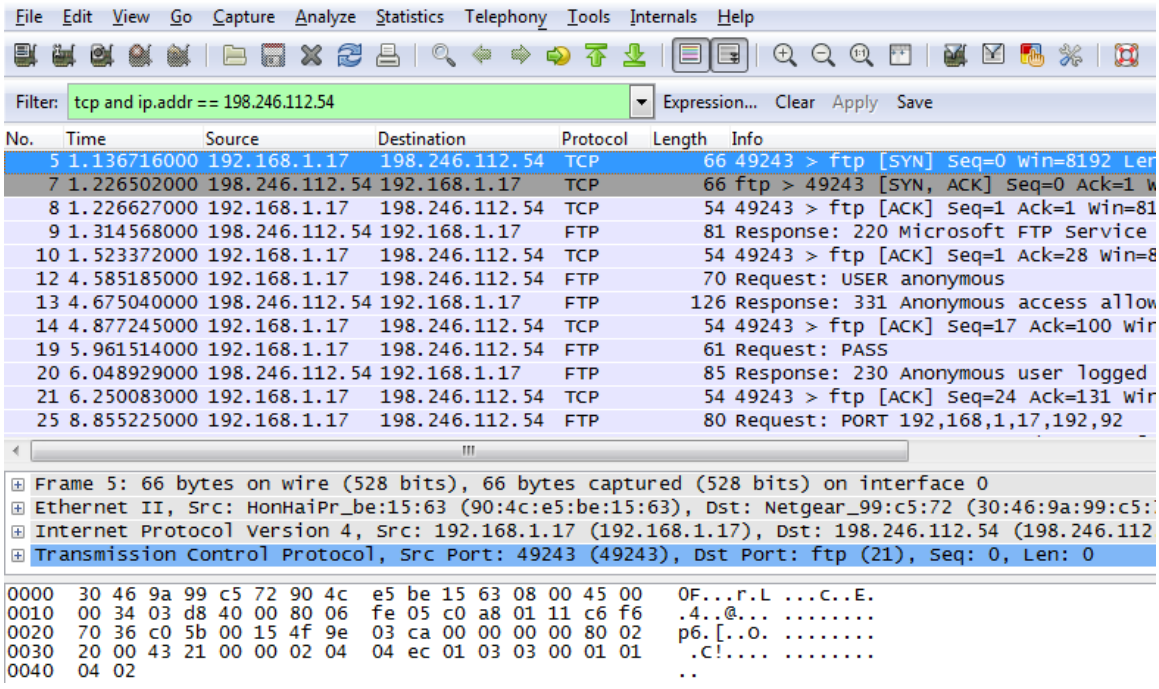
e. Localizați și descărcați fișierul **Readme**.

```
C:\Users\user1>ftp ftp.cdc.gov
Connected to ftp.cdc.gov.
220 Microsoft FTP Service
User (ftp.cdc.gov:(none)): anonymous
331 Anonymous access allowed, send identity (e-mail name) as password.
Password:
230 Anonymous user logged in.
ftp> ls
200 PORT command successful.
150 Opening ASCII mode data connection for file list.
aspnet_client
pub
Readme
Siteinfo
up.htm
w3c
web.config
welcome.msg
226 Transfer complete.
ftp: 76 bytes received in 0.00Seconds 19.00Kbytes/sec.
ftp> get Readme
200 PORT command successful.
150 Opening ASCII mode data connection for Readme(1428 bytes).
226 Transfer complete.
ftp: 1428 bytes received in 0.01Seconds 204.00Kbytes/sec.
ftp> quit
221
```

Pasul 3: Opriți captura Wireshark.

Pasul 4: Vizualizțai fereastra principală din Wireshark.

Wireshark a capturat mai multe pachete în timpul sesiunii FTP pe ftp.cdc.gov.Pentru a limita cantitatea de date pentru analiză, tastați **tcp și ip.addr == 198.246.112.54** în zona **Filter:** și dați clic pe **Apply**.Adresa IP 198.246.112.54 este adresa pentru ftp.cdc.gov.



Pasul 5: Analizați câmpurile TCP.

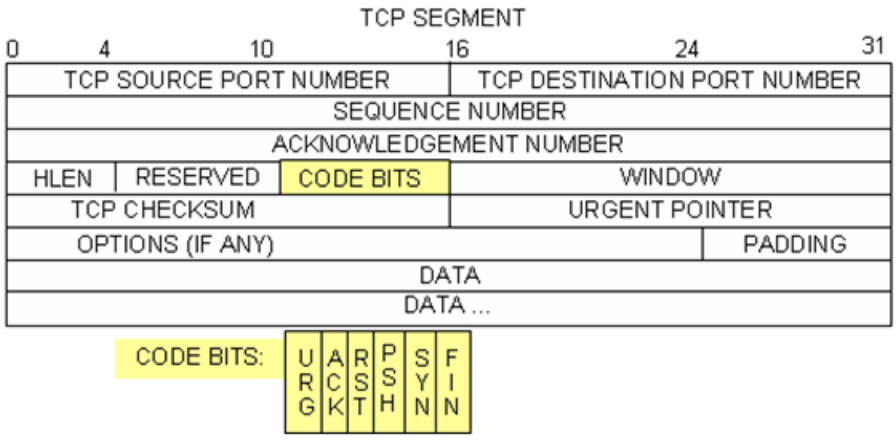
După ce a fost aplicat filtrul TCP, primele trei frame-uri din panoul cu lista de pachete arată protocolul TCP al layer-ului transport cum realizează o sesiune fiabilă.Secvența de [SYN], [SYN, ACK] și [ACK] arată three-way handshake-ul.

```
    5 1.136716000  192.168.1.17    198.246.112.54   TCP          66 49243 > ftp [SYN] Seq=0 Win=8192 Le
    7 1.226502000  198.246.112.54  192.168.1.17     TCP          66 ftp > 49243 [SYN, ACK] Seq=0 Ack=1
    8 1.226627000  192.168.1.17    198.246.112.54   TCP          54 49243 > ftp [ACK] Seq=1 Ack=1 Win=8
```

TCP este utilizat în timpul unei sesiuni pentru a controla livrarea datagramei, verifică sosirea acesteia și gestionează dimensiunea ferestrei.Pentru fiecare interschimbare de date între clientul și serverul FTP, este pornită o nouă sesiune TCP.La finalul transferului de date, sesiunea TCP este închisă.În final, când sesiunea FTP este finalizată, TCP efectuează un shutdown.

În Wireshark, informațiile detaliate despre TCP sunt disponibile în secțiunea din mijloc.Accesați prima datagramă TCP din calculatorul host și extindeți înregistrarea TCP.Datagrama TCP extinsă pare similară cu panoul cu detalii ale pachetului arătat mai jos.

```
⊞ Frame 5: 66 bytes on wire (528 bits), 66 bytes captured (528 bits) on interface 0
⊞ Ethernet II, Src: HonHaiPr_be:15:63 (90:4c:e5:be:15:63), Dst: Netgear_99:c5:72 (30:46:9a:99:c5:72)
⊞ Internet Protocol Version 4, Src: 192.168.1.17 (192.168.1.17), Dst: 198.246.112.54 (198.246.112.54)
⊟ Transmission Control Protocol, Src Port: 49243 (49243), Dst Port: ftp (21), Seq: 0, Len: 0
    Source port: 49243 (49243)
    Destination port: ftp (21)
    [Stream index: 0]
    Sequence number: 0    (relative sequence number)
    Header length: 32 bytes
  ⊟ Flags: 0x002 (SYN)
      000. .... .... = Reserved: Not set
      ...0 .... .... = Nonce: Not set
      .... 0... .... = Congestion Window Reduced (CWR): Not set
      .... .0.. .... = ECN-Echo: Not set
      .... ..0. .... = Urgent: Not set
      .... ...0 .... = Acknowledgment: Not set
      .... .... 0... = Push: Not set
      .... .... .0.. = Reset: Not set
    ⊞ .... .... ..1. = Syn: Set
      .... .... ...0 = Fin: Not set
    Window size value: 8192
    [Calculated window size: 8192]
  ⊞ Checksum: 0x4321 [validation disabled]
  ⊞ Options: (12 bytes), Maximum segment size, No-Operation (NOP), Window scale, No-Operation (NOP), Nc
```



Imaginea de mai sus este o diagramă a datagramei TCP.Este furnizată o explicație pentru fiecare câmp ca referință:

- **Numărul portului sursă TCP** aparține hostului cu sesiunie TCP care a deschis o conexiune.Valoarea este de obicei o valoare aleatoare mai mare de 1,023.
- **Numărul portului de destinație TCP** este utilizat pentru a identifica protocolului layer-ului superior sau aplicația de pe site-ul remote.Valorile din intervalul 0–1,023 reprezintă porturi cunoscute (“well-known”) și sunt asociate cu servicii și aplicații populare (așa cum se descrie în RFC 1700, precum Telnet, FTP, HTTP etc.)Combinația dintre adresa IP sursă, portul sursă, adresa IP de destinație și portul de destinație identifică în mod unic sesiunea pentru expeditor și destinatar.

Notă: În captura Wireshark de mai jos, portul de destinație este 21, care este FTP.Serverele FTP ascultă la portul 21 conexiuni de client FTP.

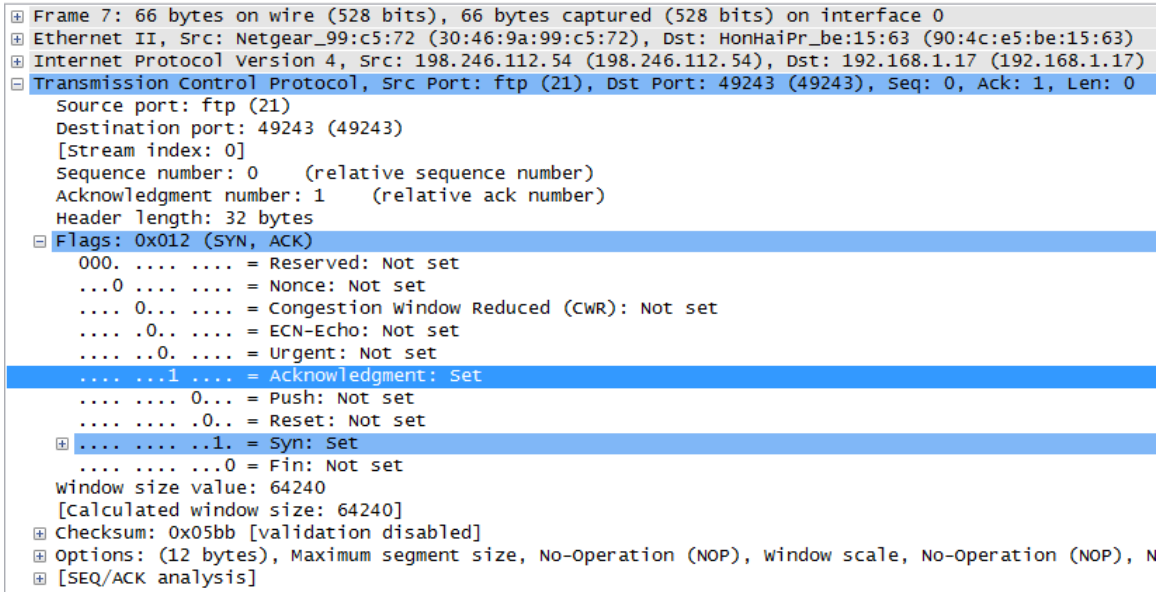
- **Numărul de secvență** specifică numărul ultimului octet într-un segment.
- **Numărul Acknowledgment** specifică următorul octet așteptat de destinatar.
- **Biții de cod** au o semnificație specială în managementul sesiunii și în tratamentul segmentelor.Valori interesante sunt:
  - ACK — Acknowledgement (confirmarea) primii unui segment.
  - SYN — Sincronizare, setat doar când o nouă sesiune TCP este negociată în timpul three-way handshake din TCP.
  - FIN — Finalizare, se solicită închiderea sesiunii TCP.
- **Dimensiunea Windows** este valoarea pentru alternarea ferestrelor; determină câți octeți pot fi trimiși înainte de a aștepta o confirmare.
- **Pointer-ul Urgent** este utilizat doar cu un flag Urgent (URG) atunci când expeditorul trebuie să trimită urgent date la destinatar.
- La **Options** este doar o opțiune, și este definită ca dimensiunea maximă a segmentului TCP (valoarea opțională).

Folosind captura Wireshark din configurarea primei sesiuni TCP (bit SYN setat la 1), completați cu informații despre header-ul TCP.

De la calculator la serverul CDC (doar bitul SYN este setat la 1):

Adresa IP Sursă	
Adresa IP de Destinație	
Numărul portului sursă	
Numărul portului de destinație	
Numărul de Secvență:	
Numărul de confirmare	
Lungimea Header-ului	
Dimensiunea pentru Windows:	

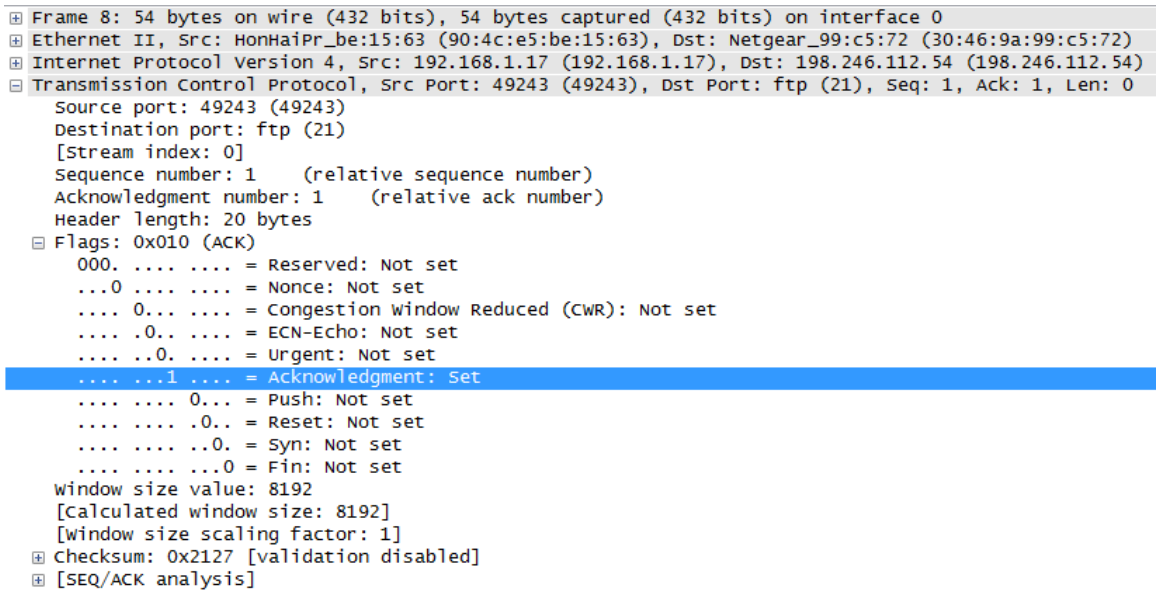
În a doua captură filtrată Wireshark, serverul FTP CDC confirmă interogarea de la calculator.Observați valorile biților SYN și ACK.



Completați următoarele informații cu privire la mesajul SYN-ACK.

Adresa IP Sursă	
Adresa IP de Destinație	
Numărul portului sursă	
Numărul portului de destinație	
Numărul de Secvență:	
Numărul de confirmare	
Lungimea Header-ului	
Dimensiunea pentru Windows:	

În etapa finală a negocierii pentru stabilirea comunicațiilor, calculatorul trimite o confirmare către server.Observați că doar bitul ACK este setat la 1, iar numărul de Secvență a fost incrementat cu 1.

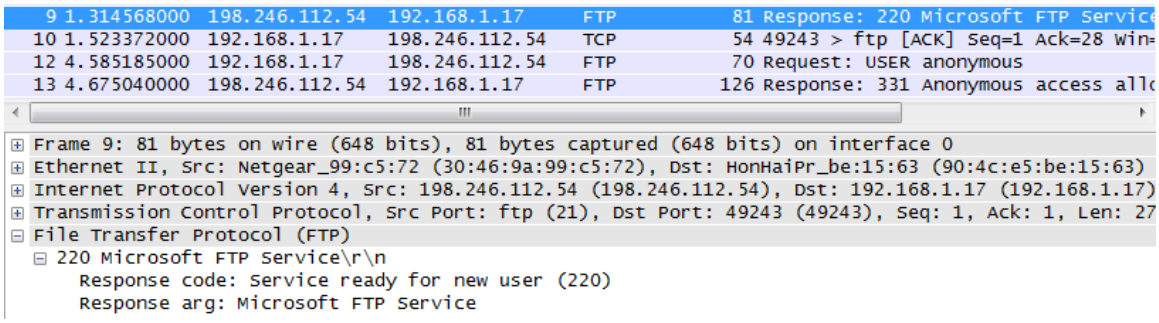


Completați informația cu privire la mesajul ACK.

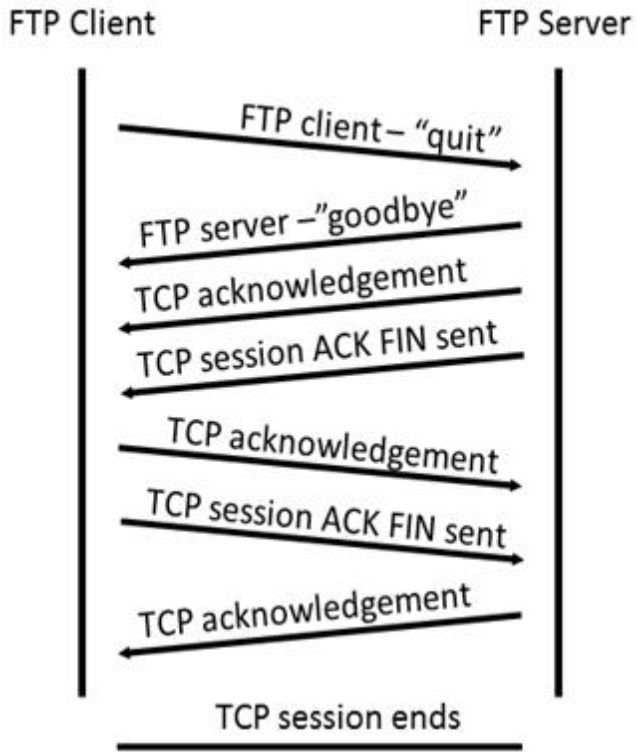
Adresa IP Sursă	
Adresa IP de Destinație	
Numărul portului sursă	
Numărul portului de destinație	
Numărul de Secvență:	
Numărul de confirmare	
Lungimea Header-ului	
Dimensiunea pentru Windows:	

Câte alte datagrame TCP conține un bit SYN?

După ce sesiunea este stabilită, traficul FTP poate apărea între serverul FTP și calculator.Clientul și serverul FTP comunică între ele; nu se știe că TCP are controlul și managementul sesiunii.Atunci când serverul FTP trimite un **Response: 220** la clientul FTP, sesiunea TCP de pe clientul FTP trimite o confirmare la sesiunea TCP de pe server.Această secvență este vizibilă în captura Wireshark de mai jos.



Atunci când sesiunea FTP a fost terminată, clientul FTP trimite o comandă pentru “quit”.Serverul FTP confirmă terminarea FTP cu un Răspuns: 221 Goodbye.La acest moment, sesiunea TCP de pe serverul FTP trimite o datagramă TCP la clientul FTP, anunțând terminarea sesiunii TCP.Sesiunea TCP a clientului FTP confirmă primirea datagramei de terminare, apoi trimite propria terminare a sesiunii TCP.Atunci când generatorul terminării TCP, serverul FTP, primește o terminare duplicată, o datagramă ACK este trimisă pentru a confirma terminarea iar sesiunea TCP este închisă.Secvența este vizibilă în diagrama și captura de mai jos.



Aplicând un filtru **ftp**, întreaga secvență a traficului FTP poate fi examinată cu Wireshark.Observați secvența evenimentelor în timpul sesiunii FTP.Utilizatorul anonim a fost utilizat pentru a obține fișierul Readme.După ce transferul fișierul este realizat, utilizatorul a finalizat sesiunea FTP.

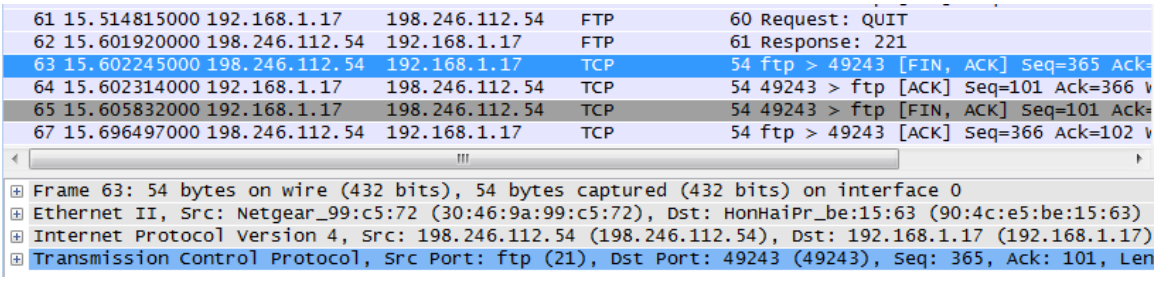
Filter: ftp — Expression... Clear Apply Save

No.	Time	Source	Destination	Protocol	Length	Info
9	1.314568000	198.246.112.54	192.168.1.17	FTP	81	Response: 220 Microsoft FTP Service
12	4.585185000	192.168.1.17	198.246.112.54	FTP	70	Request: USER anonymous
13	4.675040000	198.246.112.54	192.168.1.17	FTP	126	Response: 331 Anonymous access allowe
19	5.961514000	192.168.1.17	198.246.112.54	FTP	61	Request: PASS
20	6.048929000	198.246.112.54	192.168.1.17	FTP	85	Response: 230 Anonymous user logged i
25	8.855225000	192.168.1.17	198.246.112.54	FTP	80	Request: PORT 192,168,1,17,192,92
26	8.945530000	198.246.112.54	192.168.1.17	FTP	84	Response: 200 PORT command successful
27	8.955549000	192.168.1.17	198.246.112.54	FTP	60	Request: NLST
29	9.053034000	198.246.112.54	192.168.1.17	FTP	109	Response: 150 Opening ASCII mode data
39	9.347432000	198.246.112.54	192.168.1.17	FTP	78	Response: 226 Transfer complete.
42	12.621720000	192.168.1.17	198.246.112.54	FTP	80	Request: PORT 192,168,1,17,192,93
43	12.709658000	198.246.112.54	192.168.1.17	FTP	84	Response: 200 PORT command successful
44	12.722592000	192.168.1.17	198.246.112.54	FTP	67	Request: RETR Readme
45	12.811097000	198.246.112.54	192.168.1.17	FTP	118	Response: 150 Opening ASCII mode data
58	13.107294000	198.246.112.54	192.168.1.17	FTP	78	Response: 226 Transfer complete.
61	15.514815000	192.168.1.17	198.246.112.54	FTP	60	Request: QUIT
62	15.601920000	198.246.112.54	192.168.1.17	FTP	61	Response: 221

Aplicați din nou filtrul TCP în Wireshark pentru a examina terminarea sesiunii TCP.Patru pachete sunt transmise pentru terminarea sesiunii TCP.Deoarece conexiunea TCP este full-duplex, fiecare direcție trebuie terminată separat.Examinați adresele de destinație și sursă.

În acest exemplu, serverul FTP nu mai are date de trimis în stream; trimite un segment cu flag-ul FIN setat în frame 63.Calculatorul trimite un ACK pentru a confirma primirea FIN-ului și pentru a termina sesiunea de la server la client în frame 64.

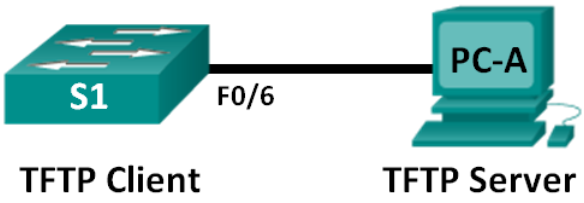
În frame 65, calculatorul trimite un FIN la serverul FTP pentru a termina sesiunea TCP.Calculatorul trimite un ACK pentru a confirma primirea FIN-ului și pentru a termina sesiunea de la server la client în frame 64.Acum sesiunea TCP s-a terminat între serverul FTP și calculator.



Partea 3: Identificați câmpurile header-ului și funcționarea UDP utilizând o captură Wireshark

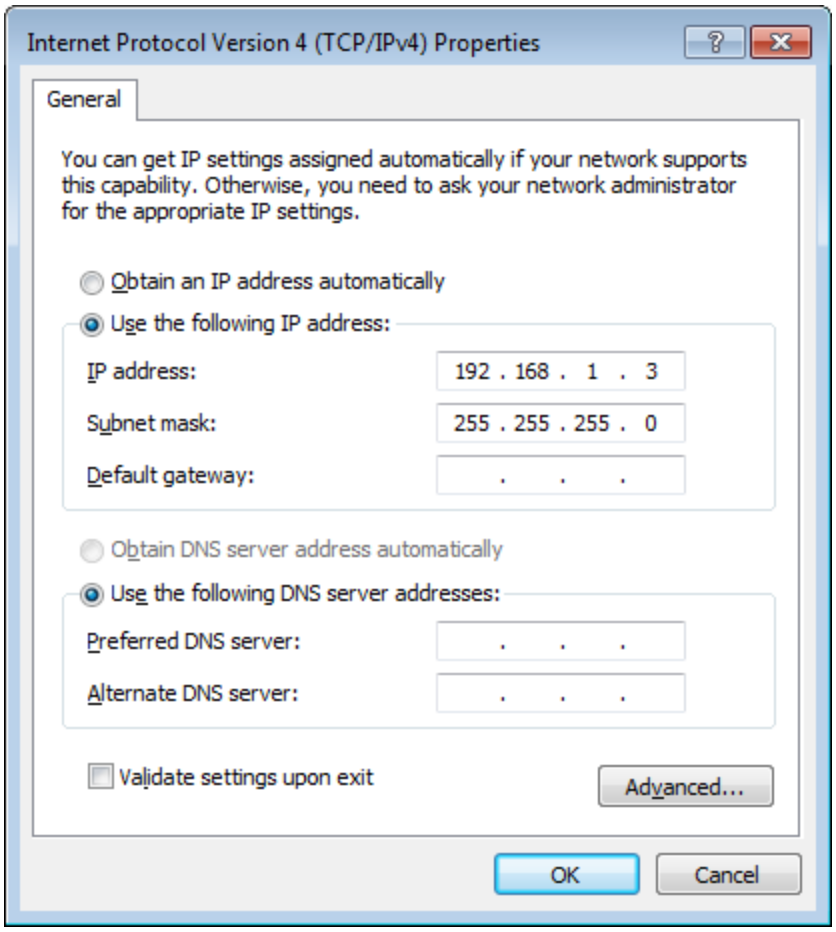
În Partea 2 veți utiliza Wireshark pentru a captura o sesiune TFTP și veți inspecta câmpurile header-ului UDP.

Pasul 1: Configurați topologia fizică și pregătiți-o pentru captura TFTP.



a. Stabiliți o conexiune Ethernet și de consolă între PC-A și switchul S1.

b. Dacă nu este efectuat deja, configurați manual adresa IP pe calculator la 192.168.1.3.Nu este necesar să setați gateway-ul implicit.



c. Configurați switchul.Alocați o adresă IP de 192.168.1.1 la VLAN 1.Verificați conectivitatea cu calculatorul dând ping la 192.168.1.3.Depanați dacă este necesar.

```
Switch> enable
Switch# conf t
Introduceți comenzile de configurare, câte una pe linie.La final puneți
CNTL/Z.
Switch(config)# host S1
S1(config)# interface vlan 1
S1(config-if)# ip address 192.168.1.1 255.255.255.0
S1(config-if)# no shut
*Mar  1 00:37:50.166: %LINK-3-UPDOWN: Interface Vlan1, changed state to up
*Mar  1 00:37:50.175: %LINEPROTO-5-UPDOWN: Line protocol on Interface Vlan1,
changed state to up
S1(config-if)# end
S1# ping 192.168.1.3
Tastați escape sequence pentru a întrerupe.
Sending 5, 100-byte ICMP Echos to 192.168.1.3, timeout is 2 seconds:
!!!!!
Rata de succes este de 100% (5/5), timpul pentru dus-întors min/mediu/max =
1/203/1007 ms.
```

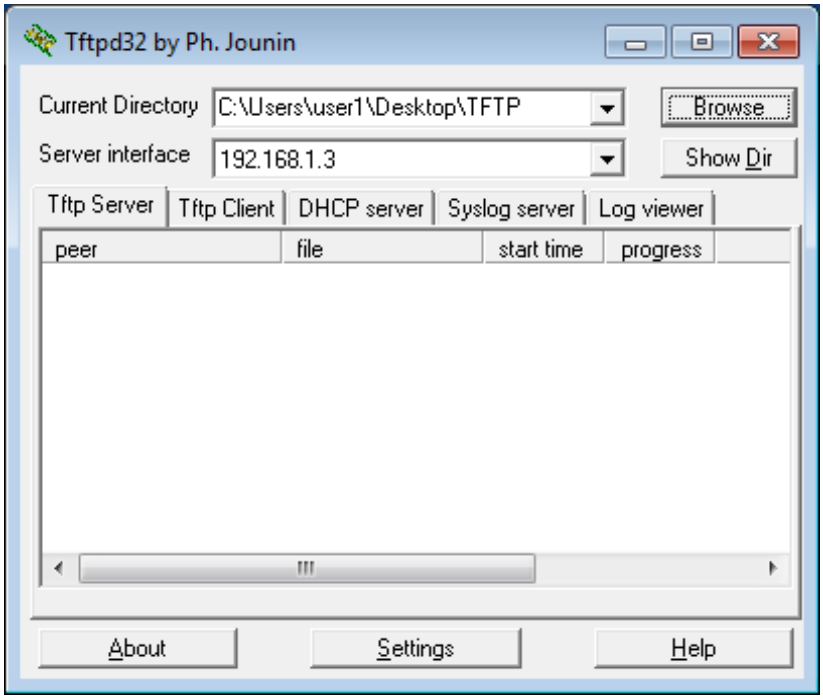
Pasul 2: Pregătiți serverul TFTP pe calculator.

a. Dacă nu există deja, creați un folder pe desktop denumit TFTP.Fișierul de pe switch va fi copiat în locația asta.

b. Porniți **tftpd32** pe calculator.

c. Dați clic pe **Browse** și modificați directorul curent în **C:\Users\user1\Desktop\TFTP** înlocuind user1 cu numele de utilizator al dumneavoastră.

Serverul TFTP ar trebui să arate astfel:



Observați că în Current Directory este afișată interfața utilizatorului și a serverului (PC-A) ca adresă IP de 192.168.1.3.

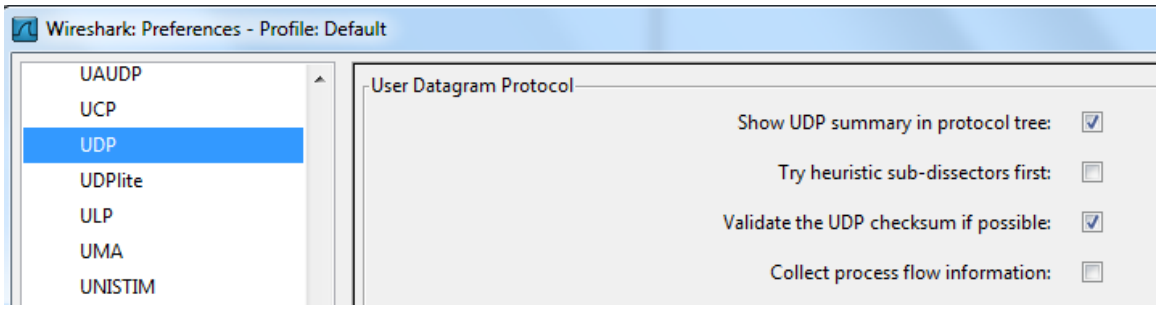
d. Testați abilitatea de a copia un fișier folosind TFTP de pe switch pe calculator.Depanați dacă este necesar.

```
S1# copy start tftp
Address or name of remote host []? 192.168.1.3
Destination filename [s1-confg]?
!!
1638 bytes copied in 0.026 secs (63000 bytes/sec)
```

Dacă vedeți că fișierul s-a copiat (ca în output-ul de mai sus), atunci sunteți pregătit să treceți la următorul pas.Dacă nu, atunci depanați.Dacă obțineți eroarea %Error opening tftp (Permission denied), verificați mai întâi dacă firewall-ul blochează TFTP, iar apoi copiați într-o locație unde numele de utilizare are permisiuni adecvate, cum ar fi pe desktop.

Pasul 3: Capturați o sesiune TFTP în Wireshark.

a. Deschideți Wireshark.Din meniul Edit, alegeți **Preferences** și dați clic pe **+** pentru a extinde **Protocols**.Dați scroll în jos și selectați **UDP**.Dați clic pe căsuța **Validate the UDP checksum if possible** și apoi pe **Apply.**Apoi clic pe **OK**.

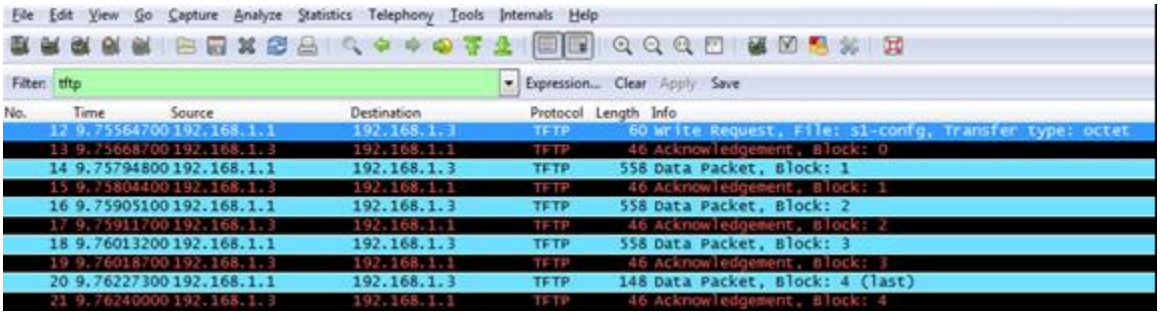


Nota instructorului: Aceasta este o modificare a versiunilor anterioare pentru acest laborator deoarece tehnologia s-a modificat.Căutați "checksum offloading in Wireshark".

b. Porniți o captură Wireshark.

c. Rulați comanda **copy start tftp** pe switch.

d. Opriți captura Wireshark.



e. Setați filtrul la **tftp**.Outputul dumneavoastră ar trebui să fie similar cu cel de mai sus.Transferul TFTP este utilizat pentru a analiza operațiile UDP ale layer-ului transport.

În Wireshark, informația detaliată a UDP-ului este disponibilă în panoul cu detalii ale pachetului.Accesați prima datagramă UDP de pe host și mutați cursorul la panoul cu detalii.Poate fi necesar să ajustați panoul cu detalii și să extindeți înregistrarea UDP prin clic pe caseta de extindere a procolului.Datagrama UDP extinsă ar trebui să semene cu diagrama de mai jos.

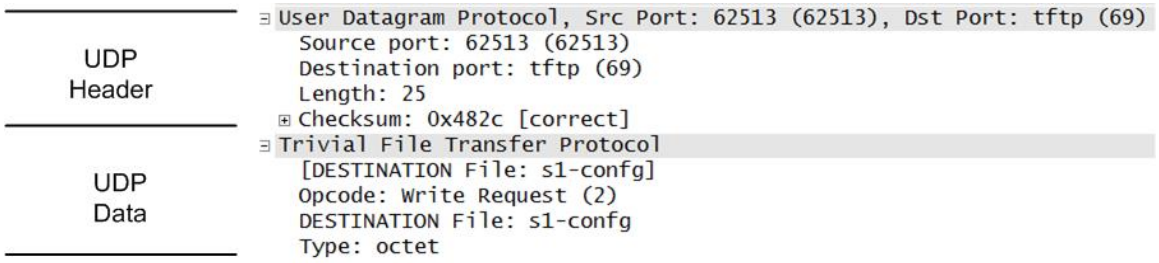
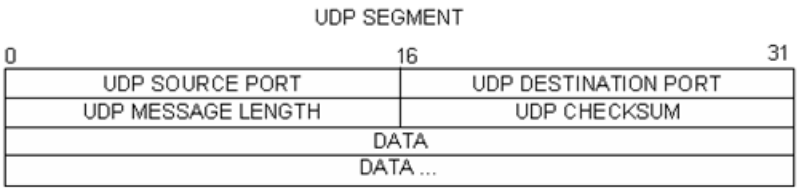


Figura de mai jos este o diagramă a datagramei UDP.Informația din header este dispersată, în comparație cu datagrama TCP.Similar cu TCP, fiecare datagramă UDP este identificată de portul sursă UDP și portul de destinație UDP.



Folosind captura Wireshark de pe prima datagramă UDP, completați informațiile despre header-ul UDP.Valoarea checksum este o valoarea hexazecimală (în baza 16), indicată de codul)x de la început:

Adresa IP Sursă	
Adresa IP de Destinație	
Numărul portului sursă	
Numărul portului de destinație	
Lungimea Mesajului UDP:	
Checksum UDP:	

Cum verifică UDP integritatea datagramei?

_____

_____

Examinați primul frame returnat de la serverul tftpd.Completați informația despre header-ul UDP.

Adresa IP Sursă	
Adresa IP de Destinație	
Numărul portului sursă	
Numărul portului de destinație	
Lungimea Mesajului UDP:	
Checksum UDP:	

```
⊟ User Datagram Protocol, Src Port: 58565 (58565), Dst Port: 62513 (62513)
    Source port: 58565 (58565)
    Destination port: 62513 (62513)
    Length: 12
  ⊞ Checksum: 0x8372 [incorrect, should be 0xa385 (maybe caused by "UDP checksum offload"?)]
⊟ Trivial File Transfer Protocol
    [DESTINATION File: s1-confg]
    Opcode: Acknowledgement (4)
    Block: 0
```

Observați că mesajul UDP de răspuns are un port sursă UDP diferit, dar acest port sursă este utilizat pentru restul de transfer TFTP.Deoarece nu există conexiune fiabilă, doar portul sursă original folosit pentru a începe sesiunea TFTP este utilizat pentru a păstra transferul TFTP.

Observați și că UDP Checksum este incorect.Acest lucru este probabil cauzat de descărcarea checksum-ului UDP.Puteți învăța mai multe despre asta căutând “UDP checksum offload”.

Reflecție

Acest laborator a furnizat o oportunitate pentru a analiza operațiile protocolului TCP și UDP din sesiunile FTP și TFTP capturate.În ce măsură TCP gestionează comunicația diferit față de UDP?

_____

_____

Provocare

Deoarece nici FTP, nici TFTP nu sunt protocoale sigure, toate datele transferate sunt trimise în text clar.Se include orice ID de utilizator, parole sau conținut de fișier cu text în clar.Analizând sesiunea FTP de layer superior vom identifica rapid ID-ul utilizatorului, parola și parolele fișierului de configurare.Examinarea datelor în TFTP de layer superior este mai complicată, dar câmpul de date poate fi examinat iar ID-ul utilizatorului de configurare și informația parolei pot fi extrase.

Cleanup

Dacă nu vi se menționează altceva de către instructor.

1) Ștergeți fișierele care au fost copiate în calculatorul dumneavoastră.
2) Ștergeți configurarea pe switchul **S1**.
3) Ștergeți adresa IP manuală din calculator și restaurați conectivitatea la Internet.