Abbildung 158

# Part 5: Sniffing, Virtual Private Network (VPN)

## Theoretische Grundlagen

***Sniffing***

Der Begriff *Sniffing* stammt aus dem englischen und bedeutet übersetzt schnüffeln.

Mit Hilfe eines Tools der Netzwerkanalyse, welches als ein *Sniffer* bezeichnet wird, ist das „Sniffen“ möglich. Dies bedeutet unter anderem, dass der Datenverkehr eines Netzwerks auf Abnormitäten und potenziellen Gefahren überprüft werden kann. In Detail kann jedes übertragene Paket gelesen werden, Information der Quelle von Paketen und Zielort der Pakete festgestellt werden. Beispielweise kann man mit einem Sniffer Passwörter oder andere Daten auskundschaften.

***Virtual Private Network***

Virtual Private Network (*kurz: VPN*) ist eine Verschlüsselung, die es ermöglicht eine Verbindung über einen VPN-Server zu einem privaten Netzwerk in Echtzeit herzustellen. Dabei ist der Ort von dem man sich Verbinden möchte völlig irrelevant. Demzufolge ist beispielweise die Verbindung vom einem Rechner, welches sich in einem anderen Ort befindet als das Unternehmensnetzwerk, zum Unternehmensnetzwerk realisierbar. Da dies verschlüsselt funktioniert, ist es einem Dritten nicht möglich die übertragenen Informationen abzufangen.

## Exercise 1: Configure and set the networks shown below (figure1 and 2)

a) Please configure the networks depicted on figure 1 and 2.

Please scan the Network below( Figure 2) from **pnidX-mid-hh to pnidX-cnt-bln and from pnidX-cnt-bln to pnidX-mid-hh.**

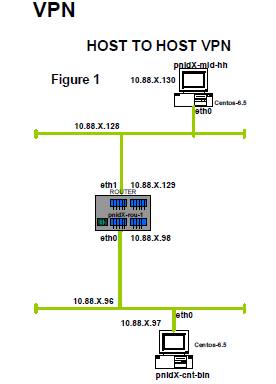
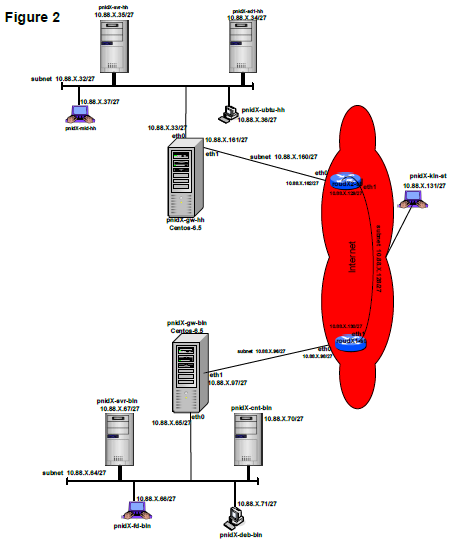


Abbildung 159

Abbildung 160

Wie in der Aufgabe gefordert, haben wir die Netze genauso konfiguriert wie in es im Figur 1 und Figur 2 vorgegeben ist.

Anschließend war es notwendig Zenmap auf beiden virtuellen Maschinen (pnid4-mid-hh und pnid4-cnlt-bln) zu installieren.

Zenmap ist ein Nmap Sicherheitsscanner – weitere Infos sind bereits auf S. 19 hinterlegt. Für die Installation von Zenmap haben wir folgendes Kommando genutzt:

**yum install nmap-frontend**

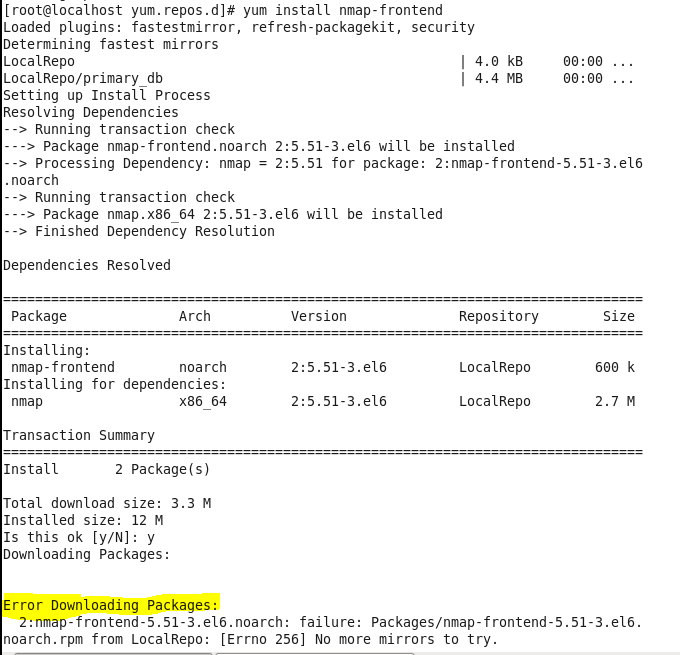


Abbildung 161

Doch wie auf der *Abbildung 161* zu sehen ist, konnte ein Packet nicht geladen werden. Dieses Packet haben wir dann wie auf *Abbildung 162* explizit installiert, sodass wir Zenmap vollständig installiert haben.

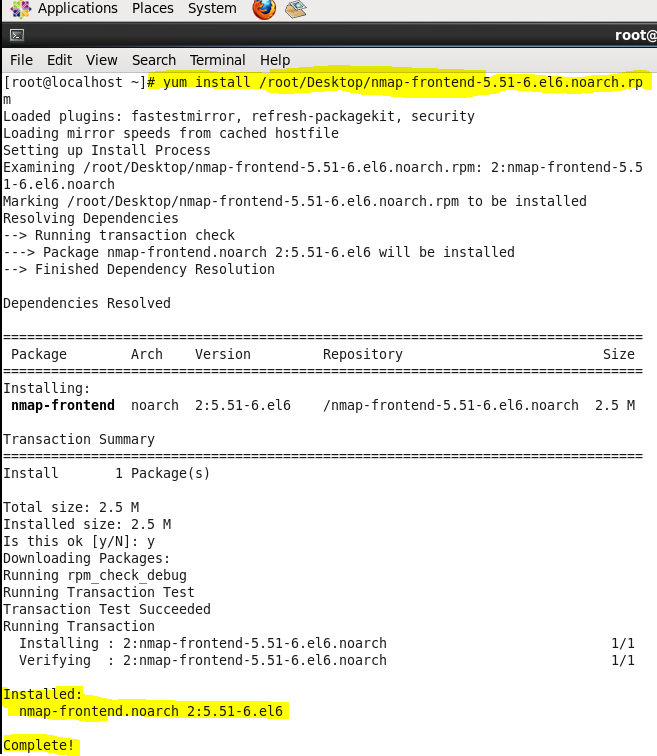


Abbildung 162

Folglich haben wir wie in der Aufgabe das Netz von **pnid4-mid-hh** nach **pnid4-cnt-bln** gescannt. Somit sah es für die Figur1 wie folgt aus:

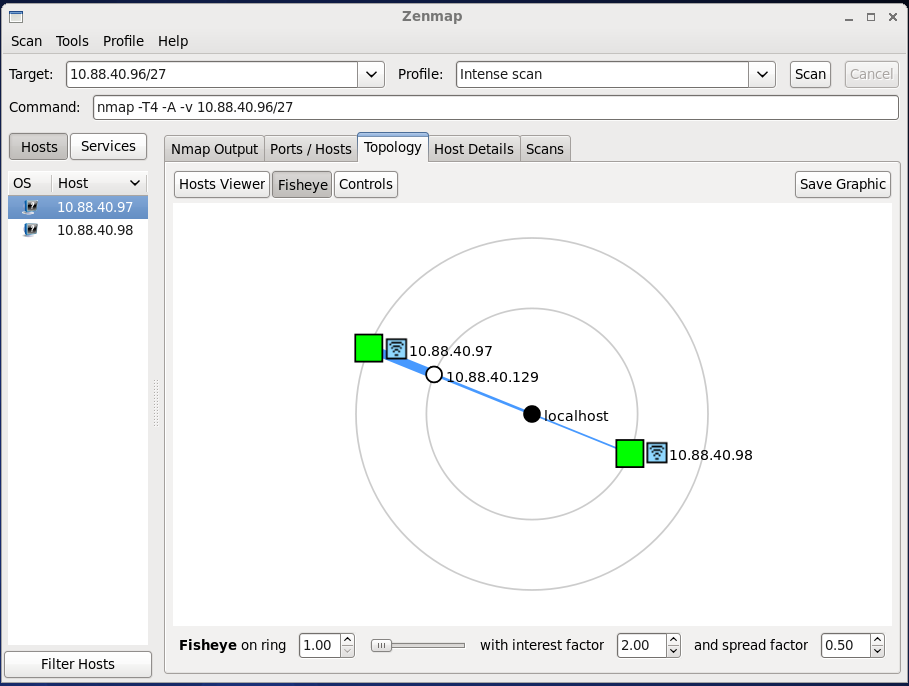


Abbildung 163

Im folgendem Screenshot ist das Netz **pnid4-mid-hh** nach **pnid4-cnt-bln** für Figur 2 gescannt:

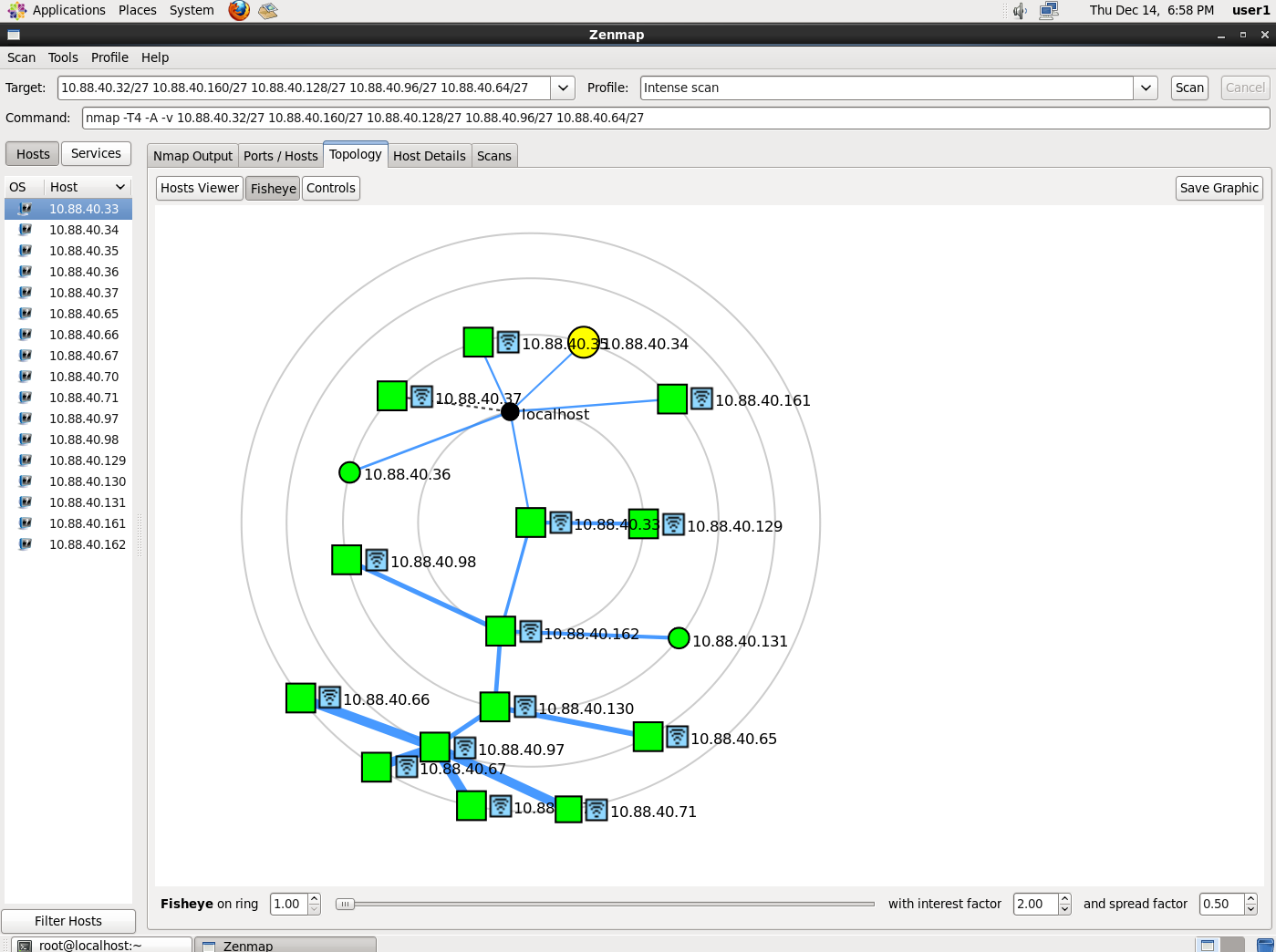


Abbildung 164

Dasselbe taten wir dann auch für das Netz von **pnid4-cnt-bln** nach **pnid4-mid-hh.**

Wie Figur 1vorgibt:

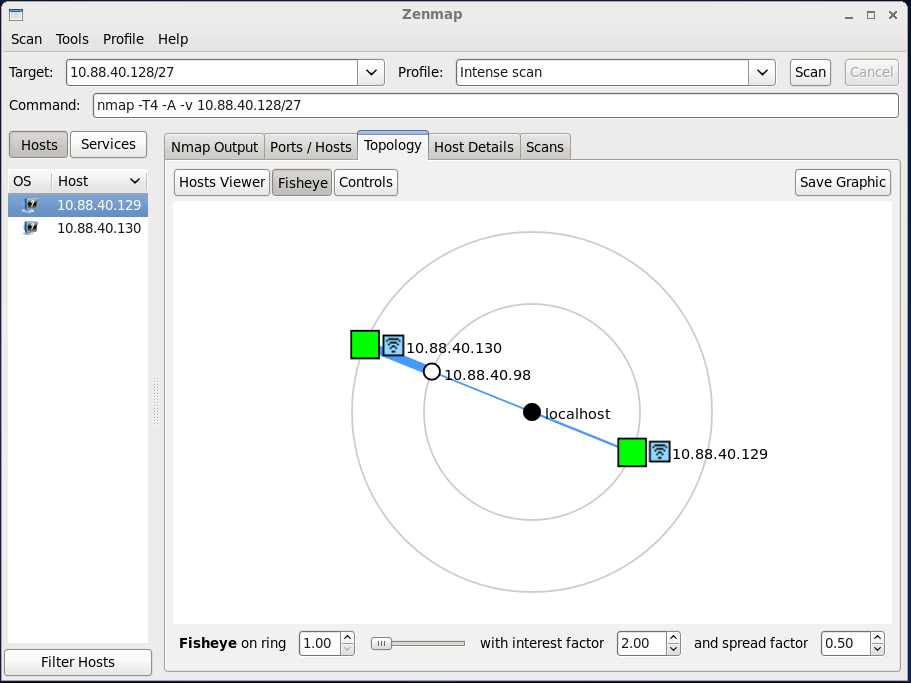


Abbildung 165

Wie Figur 2 vorgibt:

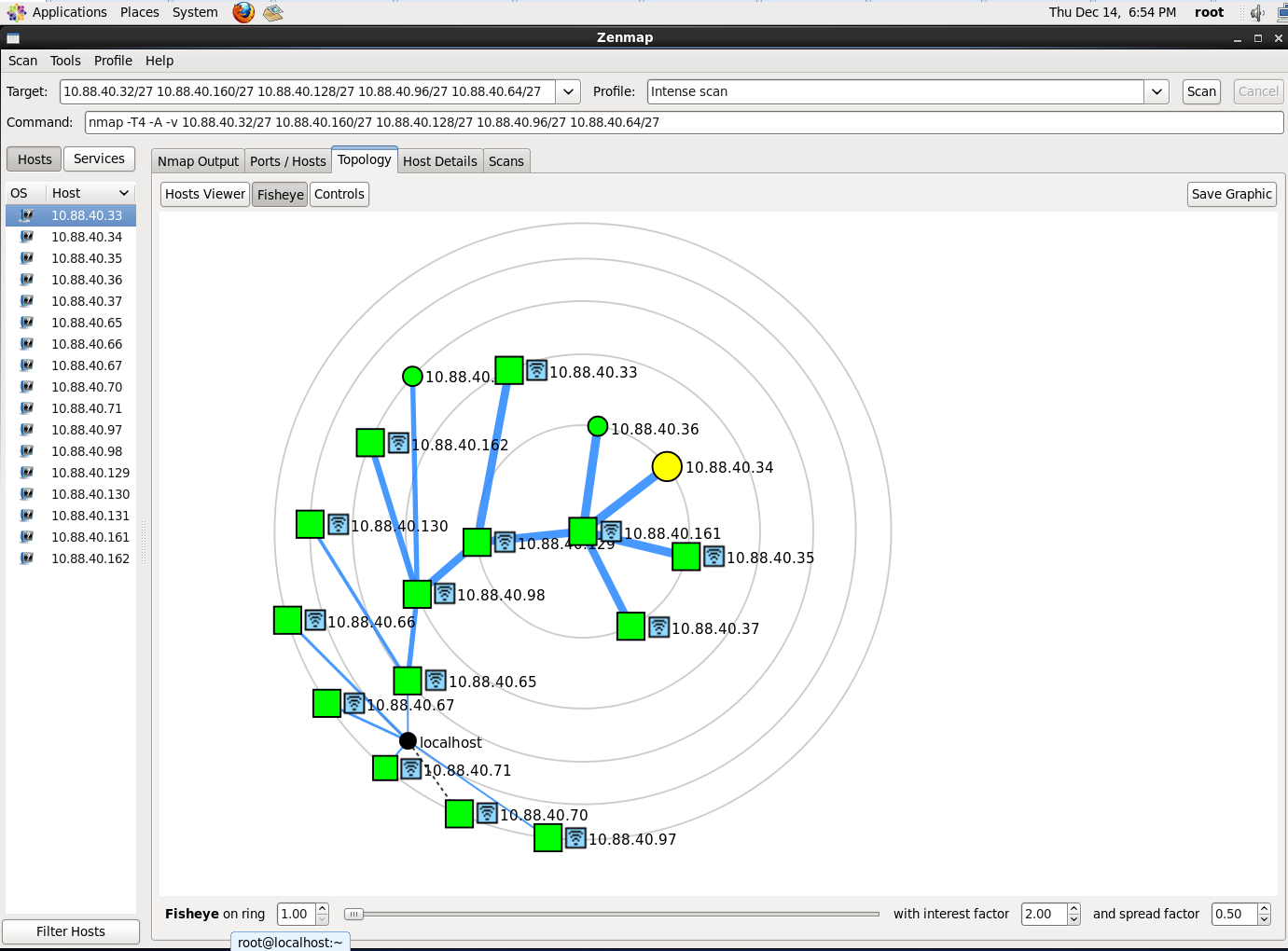


Abbildung 166

## Exercise 2: Getting started with network monitoring tools

**Installation of network monitoring tools**

Install Wireshark, Wireshark-gnome, vsftpd and telnetd from the Centos6.5 ISO on pnid4-mid-hh and pnid4-cnt-bln. Get familiar (e.g. by using the manuals) with the

following tools:

tcpdump Wireshark

The tools and manuals are available on the Centos6.5 ISO. Please refer to the internet for more information. You need tcpdump and Wireshark on at least one workstation and on your server.

For each exercise, analyze the traffic with different tools.

a) Start the virtual machines as depicted in figure 2 (pnidX-mid-hh server and pnidX- cnt-bln server) connected. Connect from the pnidX-mid-hh to the webserver running on the server pnidX-cnt-bln, if the httpd server on pnidX-cnt-bln is not available, you have to start the httpd server (search google to find out how to do this).

Follow several links on the server while watching packets flow (with Wireshark and tcpdump). Try other services as well (ftp, ssh, ...).

Note: Do not forget to install Wireshark and Telnetd Service on server pnidX-mid- hh and pnidX-cnt-bln.

Connect to the webserver running on 10.88.X.∆, (∆ is an integer >0<254) from a virtual machine. Set filters so you will see only the traffic from your virtual machine to the webserver. Set another filter so you can see only http traffic. Save the TCP-handshake with 10.88.X.∆ in a file, where X is your group number and ∆ is your subnet ip address.

In den theoretischen Grundlagen wurde bereits der Ausdruck *Sniffing* erklärt.

Dies wird in diesem Aufgabenteil nochmal aufgegriffen, da wir hier die *Sniffing*-Tools Wireshark und Tcpdump nutzen.

Wireshark ist eine freizugängliche Software, welches für die Netzwerküberwachung und grafische Aufbereitung von *Sniffern* dient. Dabei kann der Anwender sich die Inhalte und die Protokollheader der transportierten Datenpakete anzeigen lassen.

Auch das Tcpdump ist ein weltweit bekanntes Sniffing-Programm, welches Datenpakete mitschneiden kann. Tcpdump gibt es für die meisten Unix-Betriebssysteme.

Doch Tcpdump stellt im Gegensatz zu Wireshark keine grafische Oberfläche für die Analyse bereit. Stattdessen besteht die Möglichkeit, den Netzwerkverkehr über Kommandozeilen mitzuschneiden und diese dann in Wireshark grafisch zu analysieren.

Um die Analyse der Netzwerkpakete zwischen den Kommunikationsmedien ausarbeiten zu können, haben wir zunächst Wireshark installiert.

Zusätzlich haben wir Libsmi, eine C Biblothek welche den Zugang zu Informationen von MIB Modulen bietet, installiert.

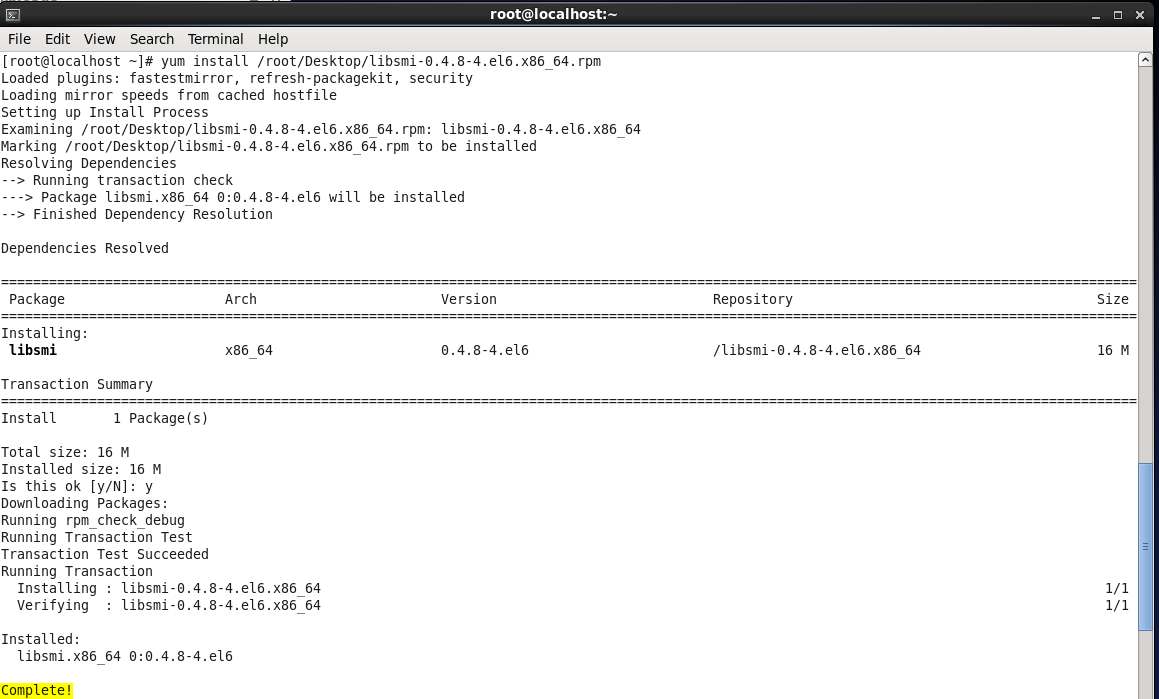
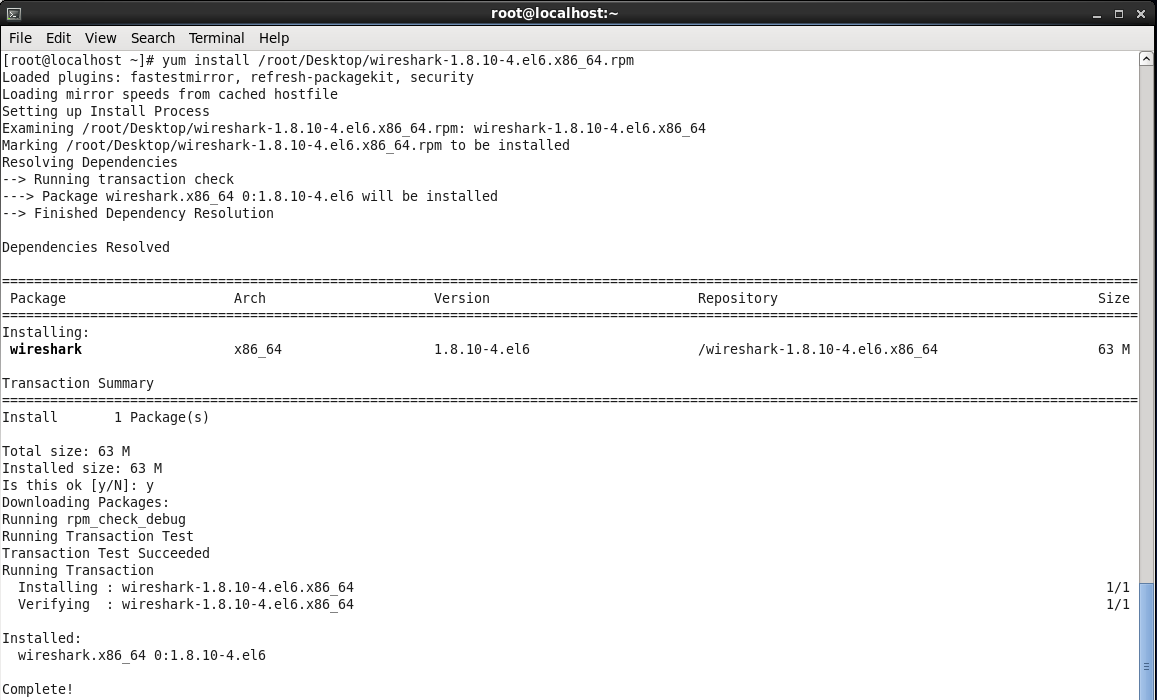
 

Abbildung 168

Abbildung 167

Anschließend haben wir in exakt dieser Reihenfolge folgendes installiert:

* Httpd – ein freinutzbarer Webserver im Internet von Apache Software Foundation

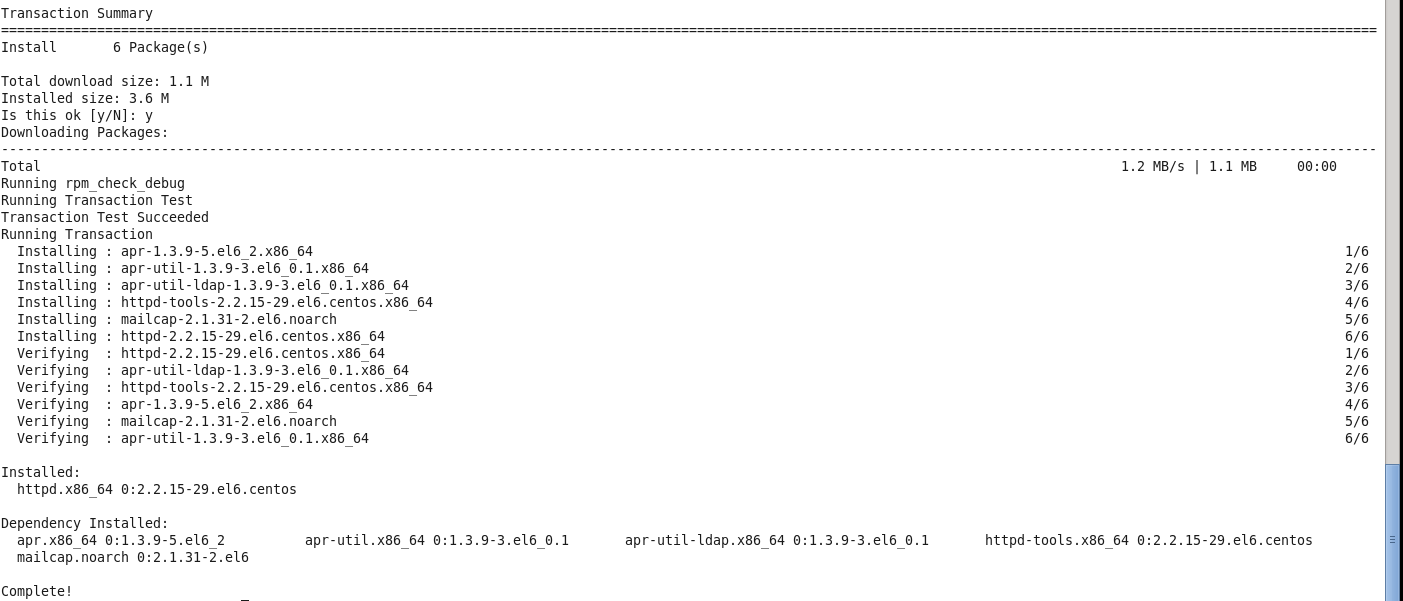
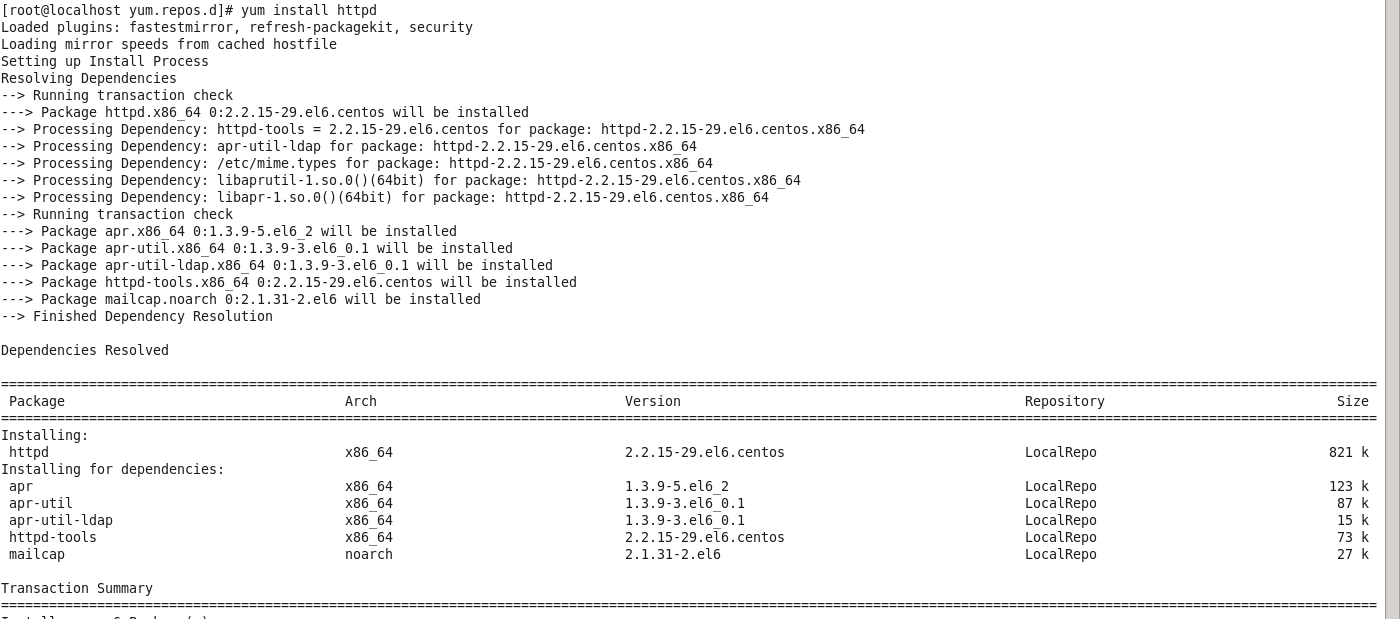


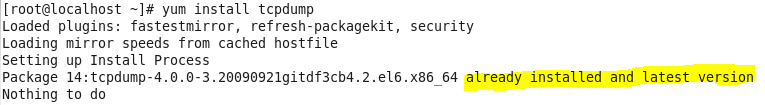
Abbildung 170

Abbildung 169



* Tcpdump – schneidet transportierte Datenpakete mit

Abbildung 171



* Telnet – ein Netzwerkprotokoll für die Verbindung zu einem anderen Rechner über den Telnet-Server

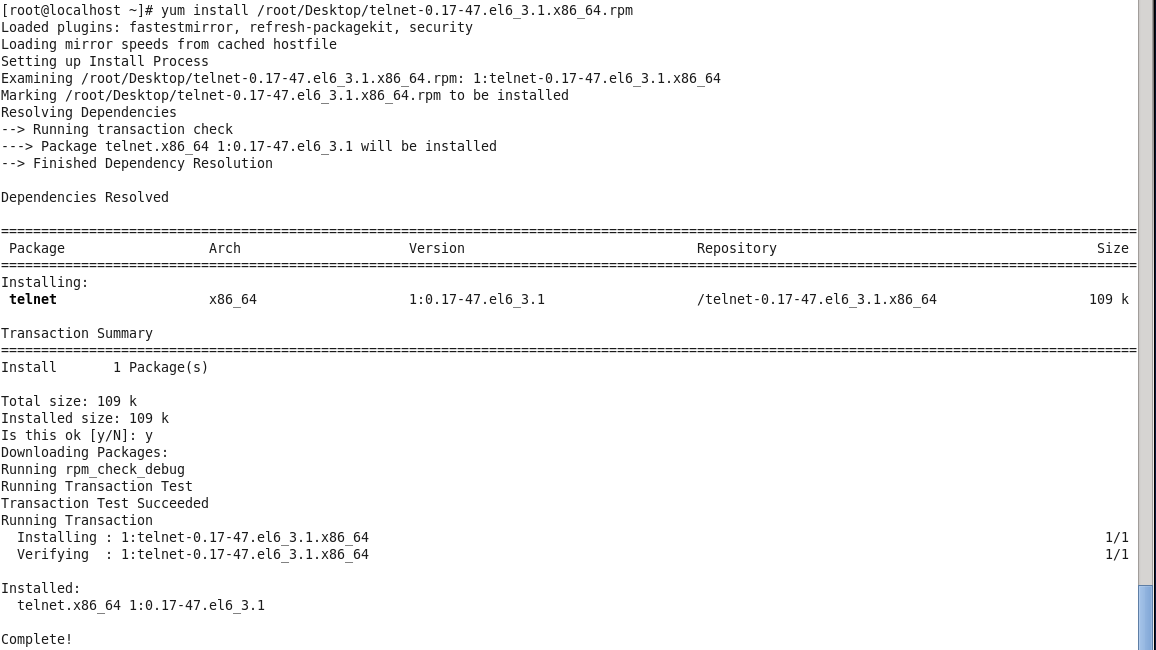


Abbildung 172

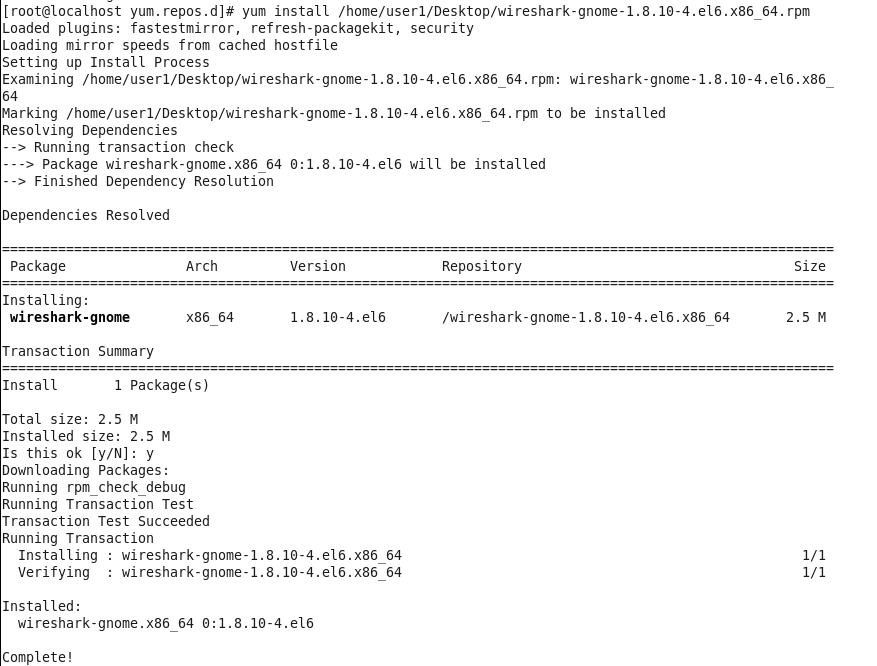
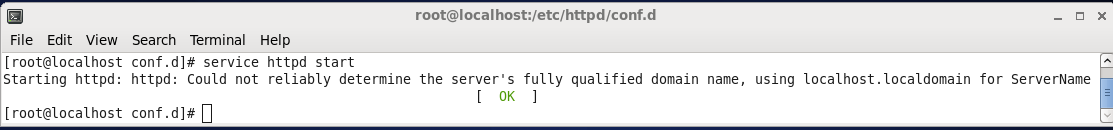


Abbildung 173

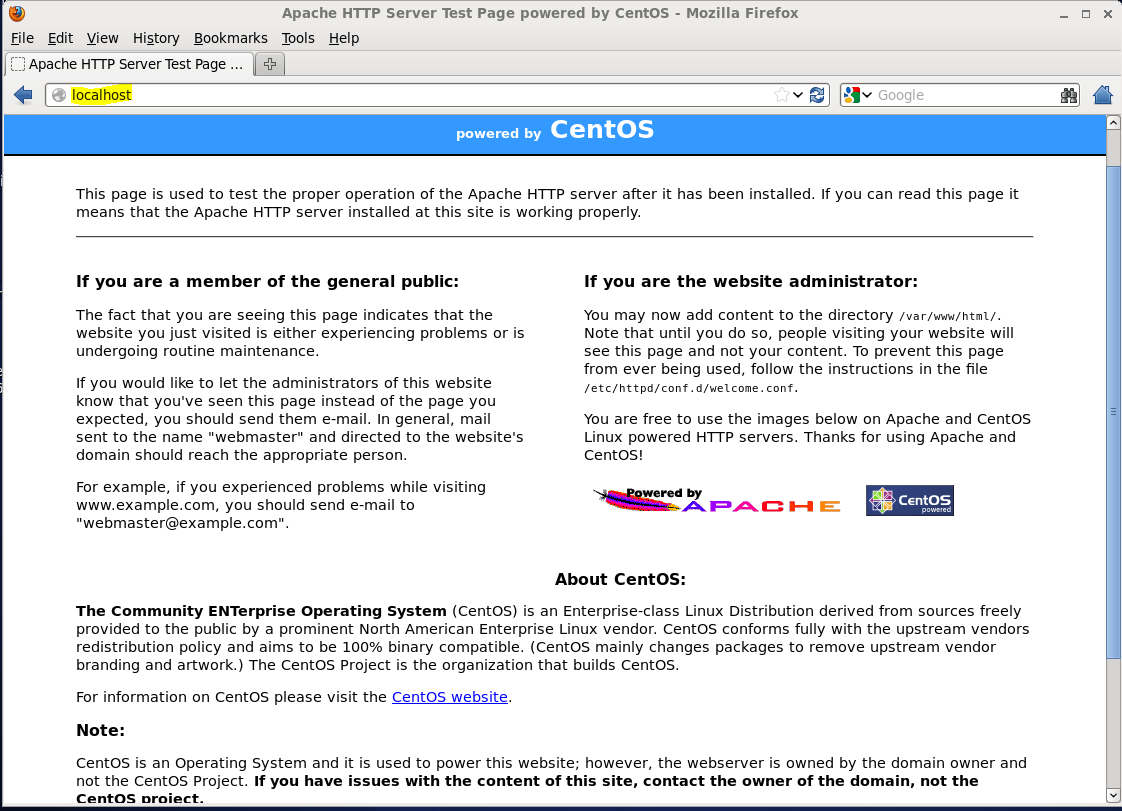
Nach den sämtlichen Installationen haben wir für den nächsten Schritt den Kommando *service httpd start* genutzt, somit sind die Dienste von Apache aktiviert worden.

Abbildung 174



Infolgedessen ist uns gelungen eine Verbindung zum Apache http Server herzustellen, wie man auf dem folgenden Screenshot sehen kann. Um dieses aufzurufen, haben wir in der Adresszeile die IP-Adresse unseres lokalen Host eingegeben.

Abbildung 175



Schließlich haben wir Wireshark gestartet und nach httpd gefiltert, sodass dann die Netzwerkpakete zum Server demonstriert wurden.

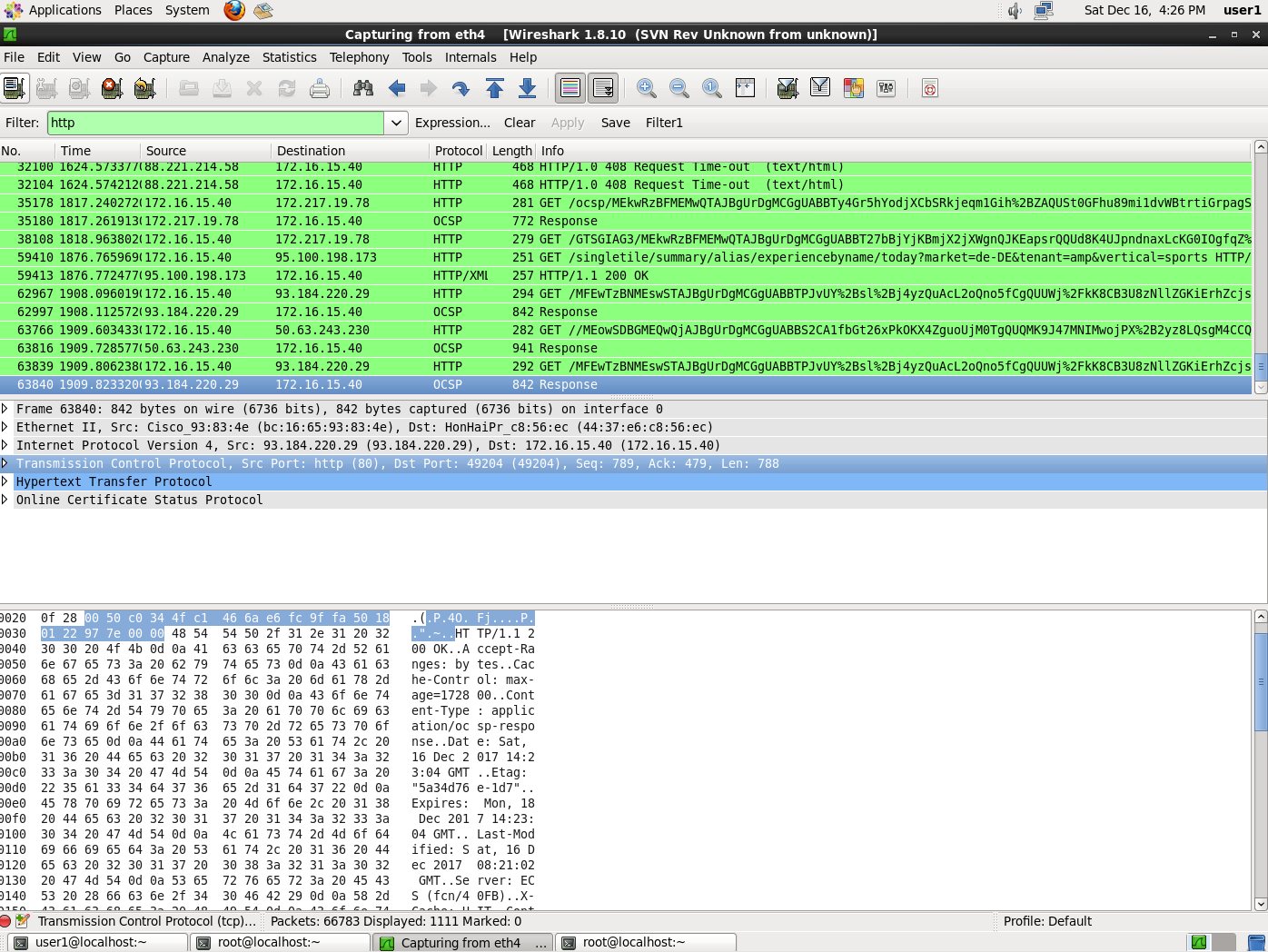


Abbildung 176

Wie in der Abbildung 177 zu erkennen, haben wir ebenfalls über Tcpdump den Netzverkehr im Terminal lassen.

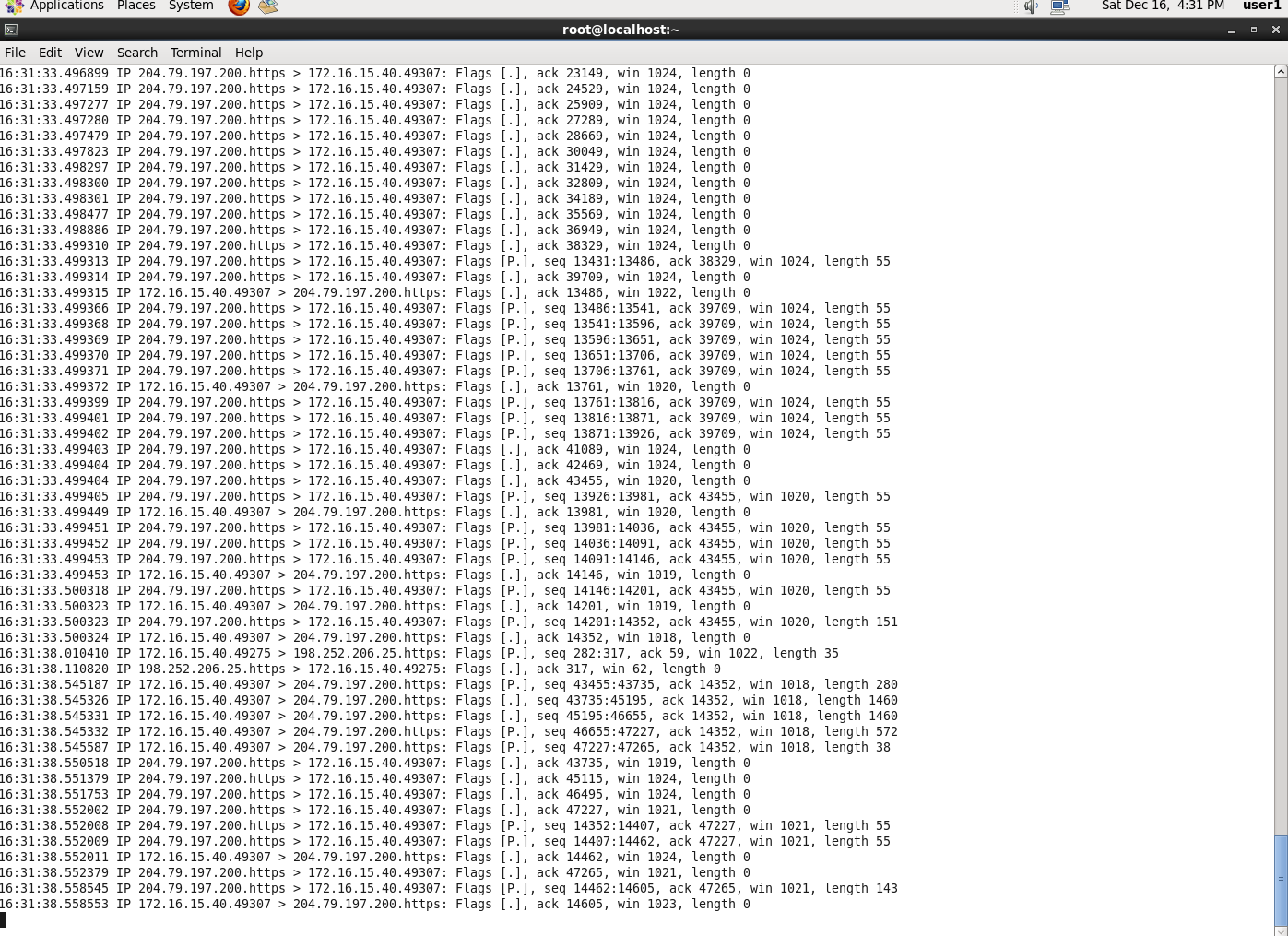


Abbildung 177

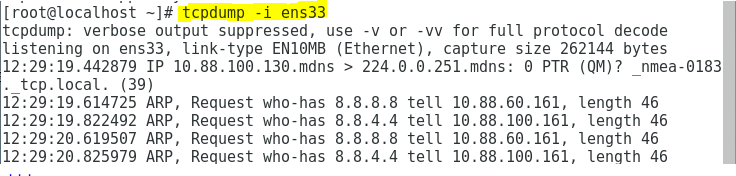
## Exercise 3: TCPDUMP

In this part of the exercise, you are going to explore tcpdump in more detail. In particular, you will answer more questions involving the use of tcpdump.

* Q1:  Please type and explain the syntax of a tcpdump command that captures packets containing IP datagram with source or destination IP address equal to 10.88.X. ∆ where ∆ is the IP address of pnidX-svr-hh(Centos7).

Der Kommando *tcpdump -i ens33* gibt an, dass die Netzwerkschnittstelle ens33 analysiert werden soll. Dieses wird solange durchgeführt bis der Service nicht explizit gestoppt wird.

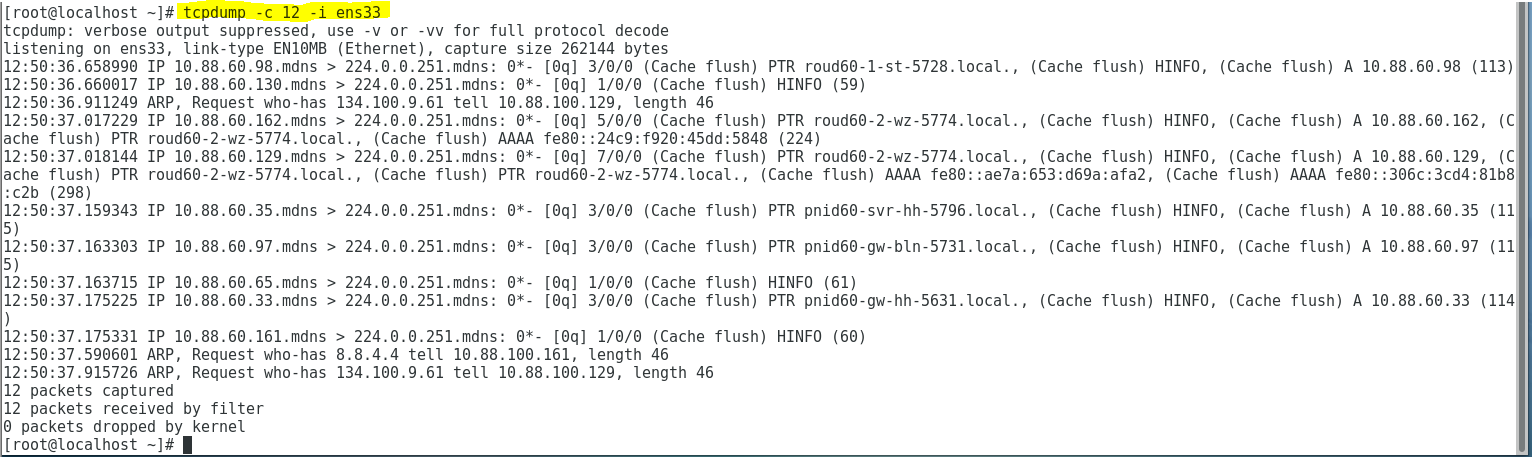
Abbildung 178



* Q2:  Please type and explain the syntax of a tcpdump command that captures 12 packets from the eth1 interface of your roudX1-st.

Der Kommando *tcpdump -c 12 -i ens33* analysiert ebenfalls die Netzwerkschnittstelle ens33, jedoch nicht wie beim Kommando von Q1 unendlich lang, sondern die Untersuchung wird nach dem 12 Packet automatisch abgebrochen. Dieses ist wegen dem -c praktikabel.

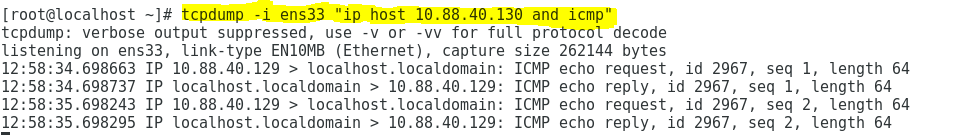
Abbildung 179



* Q3:  Please type and explain the syntax of a tcpdump command that captures packets containing ICMP messages with source or destination IP address equal to e.g. 10.88.X.∆.

Durch das eingeben von *tcpdump -i ens33 “ip host 10.88.40.130 and icmp”* werden die Pakete, die ICMP Nachrichten enthalten, beim Netzwerkverkehr mit IP-Host 10.88.40.130 analysiert.

Abbildung 180



* Q4:  Please type and explain the syntax of a tcpdump command that captures packets containing IP datagram between pnidX-mid-hh and pnidX-cnt-bln with IP addresses 10.88.X.∆ and 10.88.X.P.

Hier wird die Netzwerkschnittstelle eth4, die Netzwerkpakete zwischen pnid4-mid-hh und pnid4-cnt-bln transportiert analysiert.

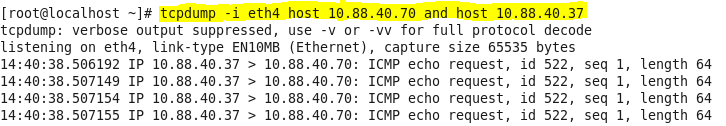
* 

Abbildung 181

* Q5:  Please type and explain a tcpdump filter expression that captures packets containing TCP segments with source or destination IP address equal to 10.88.X.∆, where ∆ is the ip address of pnidX-svr-bln.

Auch bei diesem Kommando ist es notwendig erst *tcpdump* zu tippen, sodass verstanden wird worum es sich handeln soll. Anschließend muss durch *-i Name der Netzwerkschnitte* wird verdeutlich um welche Netzwerkschnittstelle es sich handeln soll. Des Weiteren ist leicht zu erkennen, dass zwischen TCP Abschnitte und Ausgangs- bzw. die Zieladresse analysiert werden soll.

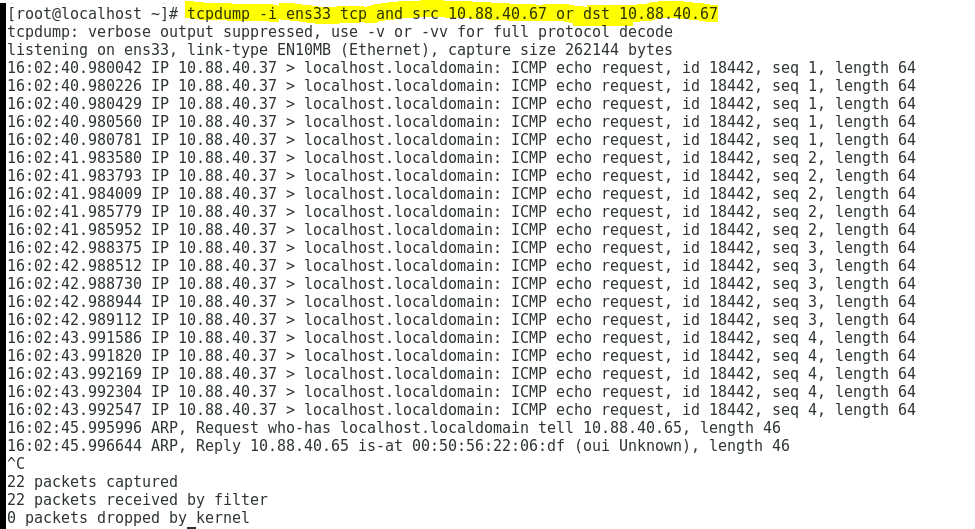


Abbildung 182

* Q6:  Please type and explain a tcpdump filter expression that, in addition to the constraints in Q5, only captures packets using port number 23. What is port number 23, which file in Linux/Windows contains port 23 and which path?

Mit Hilfe des Kommando *tcpdump -i ens33 tcp and scr 10.88.50.67 or dst 10.88.40.67 and tcp and port 23* erreichen wir im Grunde dasselbe wie bei Q5, nur mit der Ausnahme das hier nur der Port 23 berücksichtigt wird. Port 23 steht für Telnet und in Linux findet man diesen unter diesem Pfad: /etc/services/telnet 23/tcp

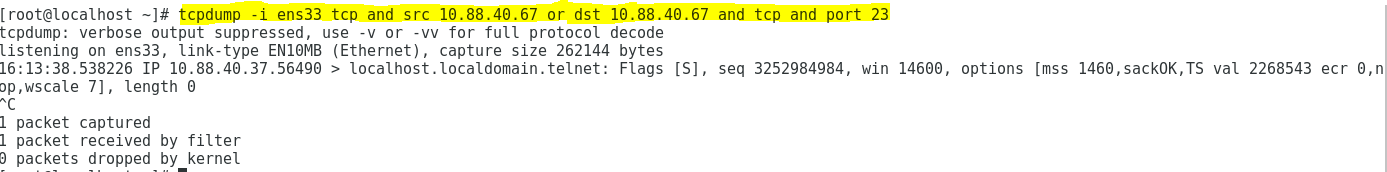


Abbildung 183

* Q7:  Please type and explain a tcpdump command that collects all ICMP packets with destination IP address 10.88.X.∆.

Der Kommando tcpdump -I ens33 -v dst net 10.88.40.67 and icmp fängt alle ICMP-Pakete die als Ziel die IP-Adresse 10.88.40.67 haben.

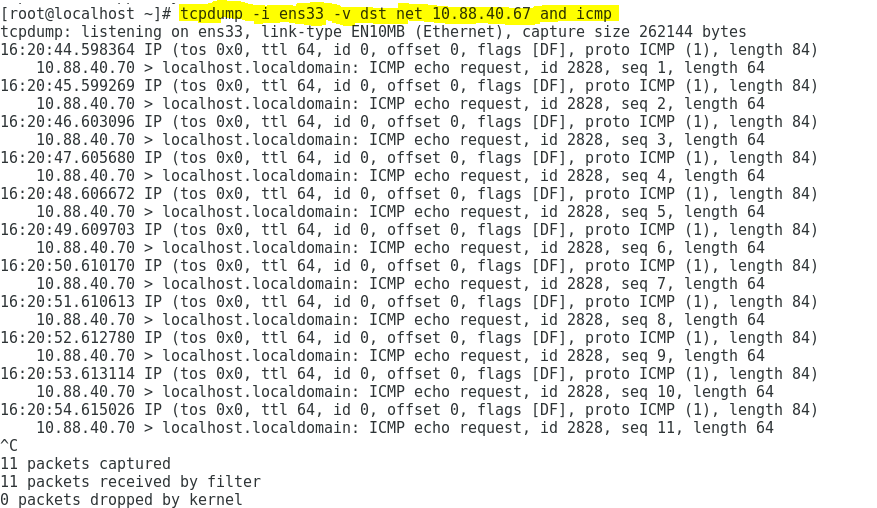


Abbildung 184

* Q8:  Assume we are about to capture all telnet packets destined to IP address 10.88.X.H on port 23 (telnet port). Type and examine the following command and explain if it is correct or not: tcpdump host 10.88.X.∆ and dst port 23 Accordingly, if not correct, please provide the syntax.

Dieser Kommando ist richtig, wenn man alle Telnet-Pakete die 10.88.40.35 auf dem Port 23 als Ziel erreichen möchte.



Abbildung 185

* Q9:  Please type and explain the syntax of a tcpdump command that collects all IP packets between host 10.88.X.∆ and host 10.88.X.P on each interface.

*Tcpdump -I any host 10.88.40.34 and host 10.88.40.38 and icmp* bewirkt, dass jedes einzeln übertragenes Netzwerkpaket zwischen 10.88.40.34 und 10.88.40.38 auf beiden Interfaces aufgezeichnet wird.

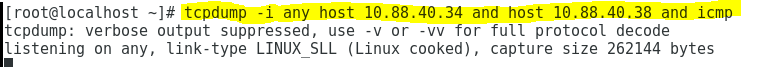


Abbildung 186

1. Start your virtual machine pnidX-mid-hh with IP-address 10.88.X.∆
2. Login as root.
3. Create a non-privileged account with the name e.g. “trumpX”.
4. Create a password for the above user. Make the password “clintonX”.
5. Try to ping another machine on your network from pnidX-mid-hh to  pnidX-cnt-bln.  ´

Um die Frage 10 (Q10) beantworten zu können haben wir zunächst die virtuelle Maschine pnid4-mid-hh gestartet, mit dem Kommando *su –* und dann das zugehörige Passwort eingegeben. Anschließend haben wir ein Konto mit dem Namen „trump4“ und Passwort als „clinton4“ eingerichtet. Mit einem ping von pnid4-mid-hh nach pnid4-cnt-bln haben wir bestätigt, dass beide virtuellen Maschinen erreichbar sind und wir mit den folgenden Schritten fortfahren können.

1. Try to telnet from pnidX-mid-hh to the virtual machine pnidX-cnt-bln, you  should be able to login as trumpX. If successful, just log out of pnidX-mid-hh.

Hier haben wir von der virtuellen Maschine pnid4-mid-hh die virtuelle Maschine pnid4-cnt-cln „getelnet“. Dies war durchführbar mit dem Kommando telnet 10.88.40.37.



Abbildung

1. Now, as root on virtual machine pnidX-mid-hh, start tcpdump with the  command tcpdump –vvv > pnidX-mid-hh-dump.1

Dieser Kommando gibt sehr viele Informationen. Je mehr V´s desto mehr Information, wobei das Erste V für very steht, das Zweite V auch für very steht und das Dritte V für verbose.



Abbildung 44

* + 1. 8. On pnidX-mid-hh, open another shell and telnet from pnidX-mid-hh to pnidX- cnt-bln and login as user “trumpX”. Logout of pnidX-cnt-bln.

Auch in diesem Schritt haben wir uns (pnid4-mid-hh) Mithilfe von telnet unter trump4 angemeldet. Hinterher haben wir uns vom pnid4-cnt-bln abgemeldet.

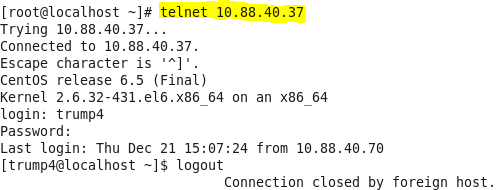
* + 1. 

Abbildung 45

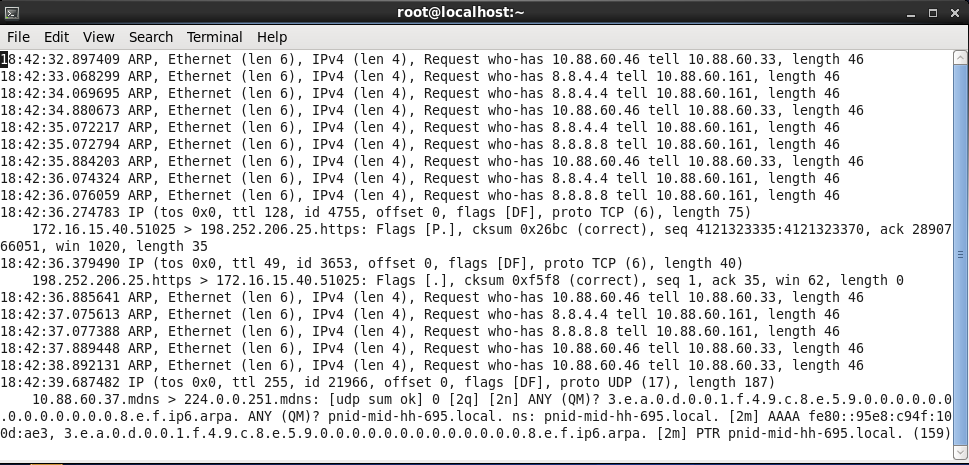
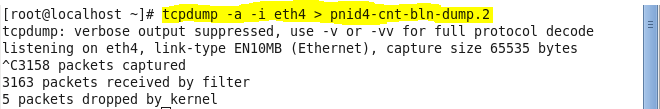
* + 1. 9. Stop tcpdump, examine the file pnidX-mid-hh-dump.1 with vi.
       - 1. Durch den Kommand vi pnid4-mid-hh-dump.1 haben wir den folgenden Inhalt anzeigen können:
    2. 

Abbildung 47

* + 1. 10. Now, as root on pnidX-mid-hh, again start tcpdump with the command tcpdump  –a –i ethX (X for your interface) > pnidX-mid-hh-dump.2
    2. Hier haben wir den Kommando *tcpdump -a -I eth4 > pnid4-cnt-bln-dump.2* ausgeführt.

Abbildung 48

* + 1. 

1. Again, open another shell on pnidX-mid-hh and ssh from pnidX-mid-hh to  pnidX-cnt-bln.

After logging in to pnidX-cnt-bln just logout again.

Hier haben wir den Kommand *ssh* an *10.88.40.37* ausgeführt.

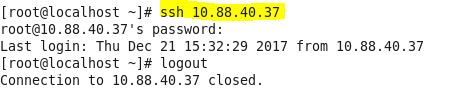


Abbildung 49

1. Examine the pnidX-mid-hh-dump.2

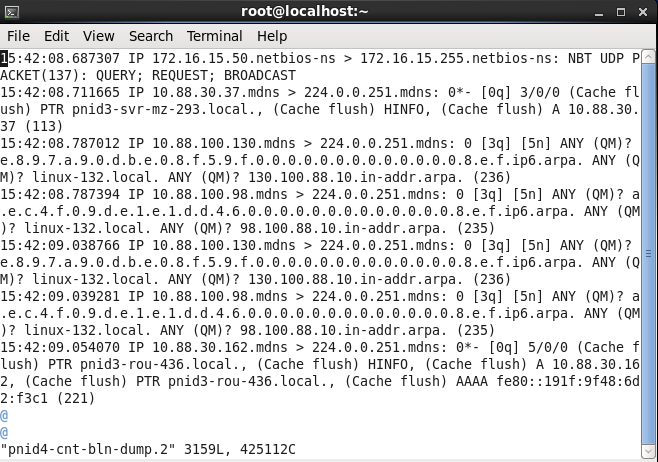
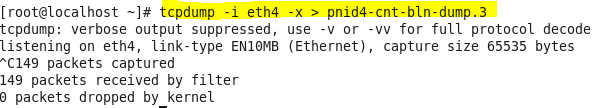
Wie in diesem Schritt gefordert, haben wir die durch vi pnidX-mid-hh-dump.2 den Inhalt angezeigt.

Abbildung 50

1. Finally, again as root on pnidX-mid-hh, again start the tcpdump with the  command tcpdump -x > pnidX-mid-hh-dump.3

Hier haben wir durch den Kommando tcpdump -I eth4 -x > pnid4-cnt-bln-dump.3 die Datenpakete der Netzwerkschnittstelle eth4 mitgeschnitten.

Abbildung 51



1. Again, open another shell on pnidX-mid-hh and from pnidX-mid-hh ftp to pnidX- cnt-bln. When you login to pnidX-cnt-bln, just logout.

In diesem Fall haben wir ftp 10.88.40.37 ausgeführt.

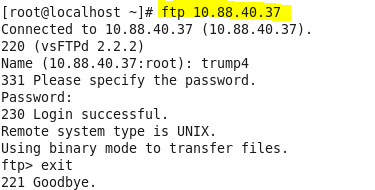
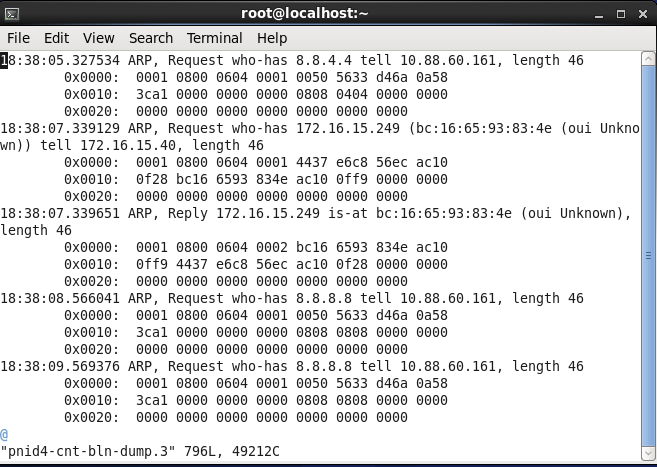


Abbildung 52

1. Examine the pnidX-mid-hh-dump.3 file.

Abbildung 53



Q10: Now answer the following questions:

1. What is the difference in the three tcpdump commands? In their outputs?

Der Kommand *tcpdump -i eth4 -vvv > pnid4-cnt-bln-dump.1* gibt sehr viele Infromationen aus. Bei je mehr V im Kommando werden desto mehr Details bereitgestellt.

Für *tcpdump -a -i eth4 > pnid4-cnt-bln.dump.2* hier werden alle Pakete von der Netzwerkschnittstelle angesammelt und als ASCII ausgegeben.

1. Is it possible to discover “trumplX” password with tcpdump?  How would you do this?

Theoretisch kann man mit tcpdump ein Passwort “sniffen”. Mit egrep kann man dies beispielweise durch folgenden Kommando Passwörter aufspüren:

*tcpdump port http or port ftp or port smtp or port imap or port pop3 -l -A | egrep -i 'pass=|pwd=|log=|login=|user=|username=|pw=|passw=|passwd=|password=|pass:|user:|username:|password:|login:|pass |user ' --color=auto --line-buffered -B20*

1. How can you tell if a network interface is running in “promiscuous mode”?

Anhand dem Kommando netstat -i kann man sehen ob ein Interface auf „promiscuous mode“ abläuft. Falls ein P unter der Bezeichnung „Flg“ bedeutet dies, dass für dieses Interface „promiscuous mode“ angeschaltet ist.

1. Can a normal, unprivileged user, run tcpdump?

Nur root kann tcpdump ausführen, da andere User keine Rechte dafür haben.