S. No.	Date	Title	Page No.	Teacher's Sign / Remarks
1	16/7	Basic Networking		
2	23/7	Study of network cable		
3	30/7	Cisco packet Traces		
4	6/8	Setup and configure LAN		
5	9/8	wireshark		
6	10/8	Hamming code		
7	6/9	Sliding window		
8(a)	10/9	simulating virtual LAN		
8(b)	13/9	simulating wireless LAN		
9	20/9	classful subnetting		
10	29/9	Routers		
11(a)	11/10	Static route config		
11(b)	8/10	simulating RIP		
12a	15/10	Echo client		
13)	18/10	ping program		
14)	22/10	packet sniffers		
15)	25/10	weblogs		

Completed

Ex No: 1
Date 16/7/24

Aim :

To

used i

(i) ang -
(172. 1

[ultrai
gateway

[ultrai]

(172:
on ie

ii) hostna
l

iii) netsta
Ac

Prof. R

tcp

tcp

Ex No : 1 Basic Lenux Commando

Date 16/7/24

Aim :

To Study the various network commondo used in Lenux & windows.

(i) arp - a

(172.16.9.23) at 7C:5T:58:35:00:31 [ether] on emp 250

gateway (172.16.8.1) at 7C:5a:1c:5t:bc:45 [ether] on emp 50

(172:16:10:60) at d3:bb:C1:C5:ca:5t [ether) on emp 250

ii) hostname

localhost . local domain

iii) netstart

Active Unteernet Connection (w/o seewees)

Proof	Rev-Q	send Q	local add	force add	State
tcp	0	0	125:0.0.160		Desktop
tcp	0	0	125.0.0.1 67218		Desktop

(iv) ipconfig /all

windows IP configuration

HOSTNAME .... DESKTOP - 7TFKDUG

Primary DNS Suffix

Node type ... Mixed

v) nbstat -a

Display protocol states & current TCP/IP connection using NBT

NBSTAT [[-a RemoteName] [-A IP address] [-c] [-A] [-v] [-R] [-RR] [-s] [-S] [interval]

vi) nslookup www.rajalakshmi.org

default server unknown

address : 172.16.8.1

Non-authorised answer.

(vii) Pathping

usage : Pathping [-g host-list] [-h maximum hops] [-i address] [-n] [-p Period] [-q num-queries] [-w timeout] [-4] [-6] target name.

viii) ip

usage ip [option] OBJECT {COMMAND | help}

ip [- force] - batch filename

Where OBJECT = { link | address | addrlable | vaf | route | neigest ntable | tunnel | tuntap | macaddresses | mroute | monitory | xfrm | netns | 2tp | masec | top tuti token | net con | l | a | vrf }

ix) tcp dump -D

1) enp 2s0 Lup - running]

2) any (Pseudo : device that capture on all interface [UP - running]

3) lo (UP, running, loop back]

4) wtp 3s0

Ping :

usage : Ping {-aAb BdDf hLnOqrv kVvZ4]

[-c count][-i interval][- Jinterface]

[m mask][-M Pmtudis + option][-l preload]

[-p Pattern][-a tos][s packetsz][-s Sandsz]

Question & answer.

1) Ping to find reachability
2) mlu < web address >
3) ip address to show IP configuration
4) neststat
5) i pondig

Result.

Thus the various networking commands used in Linux & windows are studied

EX 2
DATE : 23/7

Study of network Cables

Aim :

Study of different types of network cables.

a) Understanding, different types of network cables used in network are

1) Unshielded Twisted pair (UTP) cable
2) Shailded twisted pair (STP) Cable
3) Coaxial cable
4) Fiber optical cable.

	Category	Max data termes minion	Adul disadvantage Advantage	Application	Image
UTP	Category 3	10 bps	Cheapir in → cost	10 box = 10 Mbps Fast ellumet	
	Category 5	Upto 100 mbps	→ easy to install as they have smaller overall diametere	gigabit ethernet fast ethernet	
	Category 5e	1 Gp ps	Disadvantages → moulprone to electromagentic Unterface & noise	Gugabit ethernet	
	10 Gbps		Advantages → Shelded → faster then Gb → less succefull to noise	Ginga bit ethernet IOG element rindely used	
STP	category 7	10 Gbps	Disvantage → expensive → greater and allation & effort	Gigabit ethernet 10G ethernet (100 m) maximum	

STP	category 7	10Gbps	Disvantage → expenssive → greater and allation & effort	Cregobit ethument 10G ettreenet (100 m)
Libus optics Cable	~~singlemode~~ multimode		Advantages → High speed → high bandwith → high security → long distance Disvantage expenssive ueequeen skilled Install	maximum distace to fbir optics wde is around 100 meters.
Coaxial cable	RG-6 RG-59 RG-11	10-10Mbps	→ High band with unmin to interaface → how loss bandiwrdth → Weesatile ① Isadvantages → limited distace o → cost → sign in buller.	speed to sigle n soom Television speed is high internet connection

b) make your own ethernet cross-over cable
straight cable.

Tools and parts needed.

→ ethernet cabling, CAT5R is ceterified for gigabit support, but CAT5 cabling works as well just overs short distance

→ A crimping tool: networking tool cheapest to push action the pins in plug & strip and cut the shulding of the cables.

→ Two RT45 Cables

→ Optional two plug shields.

Student abvervation:

1) What is the difference between cross cable and stuagint cable.

Cross cable: Comments different types of divies (pc to switch, router)

Straight Cable: Comments similar divies (pc to pc switch to routere)

2) Which type of cable connect two pc ?

Cross cable.

3) Which type at cable at used to connect a router / switch to your pc ?

Straight cable.

4) Write down your understanding ? Challenges faced to output received while making a pain aross / straight cable.

making a pair cross / straight cable unsolve carefully aligning wire pair in correct order with challenges including maintaining power wire alignment & avoid spinal interference. The output should establish a stable connection.

Result :

Theses the study to different networks cables are studied.

Ex: No 3

Date 30.7.21

CISCO packet Tracer.

Aim:

To study the packed tracer tool unstallition and uses of unterface overview Analayse the behavious of hellosve divices using CISCO packet Tracer simulator

1) From the network component box, click and drag & deep the below components

a) 4 Crenetic PC's & one Hub

b) 4 Crenetic PC's and one switch HOB using the cable the link LED should glow in green undidicating that link is up similarly connet remaining 5PC's to the HUB.

c) Simlarly connet at 4PC's to switch using copper, Stright through cable.

3) click on desktop da and IP a

4) Observe distandation of sur

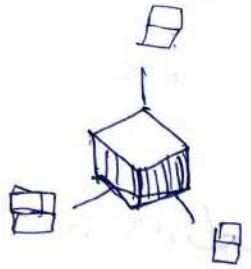
5) reapl connected

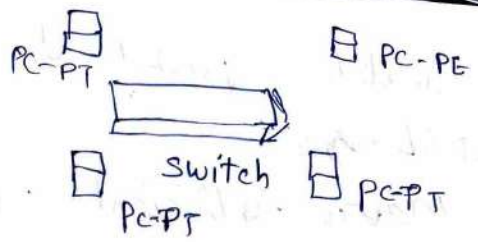
6) Observe PDO & about H observatio

9) Write in ctur them

hu

hen





3) Click on PC connected to hub, go to the deskstop dab click on IP configiuration & ethu and IP address & subnd mask.

4) Observe the flow of PDO from source PC-30 distanation PC by setting the ideation mode of summilation.

5) reapt step # 3 step # 5 for mode connected to switch

6) Observe how hub & switch are forwarding the PDO & write your obseervation & conclusion about the bhaiour at switch & hub student observation.

9) Write down the behaviour to switch HUB in turns of forwarding the packet near by them.

hub: bewad cast packetth to all ports.

hen sufficient - causes collision.

Switch : Packet to specific based on addevess

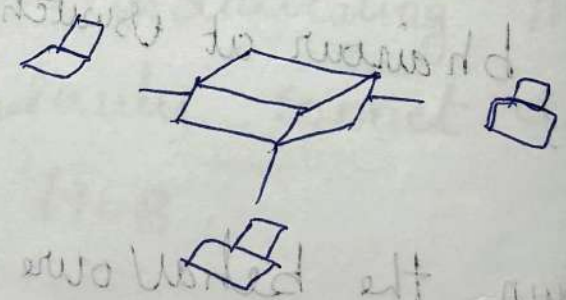
more efficient reduces collision

b) find out the metwove topology in your collage & draw a lable that topology in obtainbook.

staulogoy.

All cable to contual connetion point

If one cable beds or fails only topology in obseuritio book. Computer connected to cable is unoable to use anolive!



Result

The programe is executed sucessfully.

Ex NO: 4
Date : 6.8.24

Aim:
and
How to
step 1 :
requirin
step 2
one end
& other
3) Assion
1) log on
2) click
3) Right
select
select
assign
sub
4) Confi
5) Selec
→

EX NO: 4

Date · 6.8.24

Setup a configiue LAN

Aim: Setup a Configiue a LAN using switch and ethernet cable in your lab.

How to set up LAN?

Step 1 · Plan & design as apperopirate network requrriment & equpment location.

Step 2 Connect computer to network like wanted one end at ethnnd cable ins to your computer & other end to your network switch

3) Assion Ip address to your PC.

1) Log on to the clumit computer

2) Click network & internet connection.

3) Right click Lg. connection → go to cpwritles → select subnnd pecotocol → click on pewpertin → select use of pewpeties. Ip addurs option and assign Ip address. PC I - Ip address 10.1.1.1

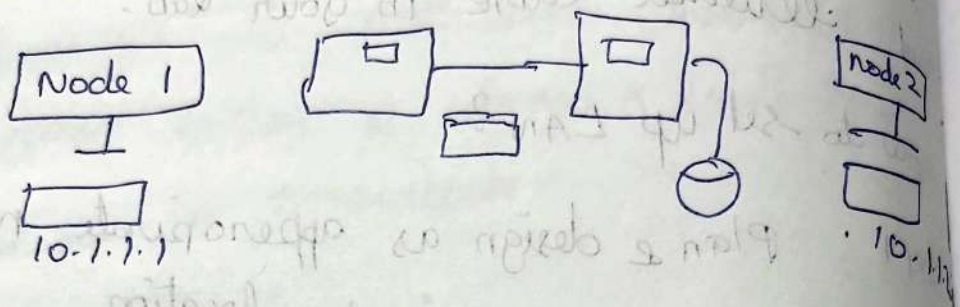
submet mark 25.5.0.0.0

4) Configies network suplem

5) select folder → go to peroperties → click → shauring → share with everyone.

6. try to accen the shared folder from other computers of reswork.

student observation



IP Configuration.

Node 1 IPV4 address 10.1.1.1

Node 2 IPV4 address 10.1.1.2

Outcome -

The file cent through click iresim.

[1/10.1.1.2]/usr/1 Rec] is sent succ

Result:

LAN setup in Lab is done sucessf

EXNO 5

Date 9.5.24

EXNO 5 Wireshark

Date 9.5.24

Aim:

exprement of Packet capture wireshark.

Packet Sniffer

→ sniff message being sent / received from Computer.

→ slows & display content to various network.

→ Passive programs.

→ never send packet itself

→ no packet addressed to it.

→ receive a copy to all packets.

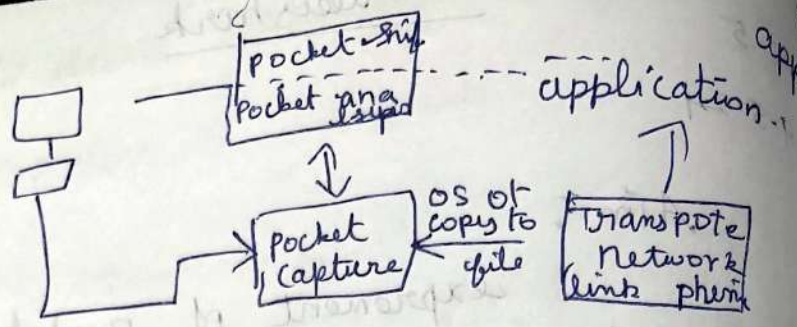
Packet sniffer structure diagonistic

tools TCP dump.

eg: tcdump – enx host 10.129.41.2 = to exe

3.02. wireshark:

– wireshark – exe 3.out.



wireshark:

→ network & analysis tool

→ formally known as ethernet.

→ Capture packets in real time & display

→ Include formats, filters, color coding etc

→ troubleshoot.

→ examine security problem.

Download wireshark.

→ download & install from www.

Capturing packets.

→ Launching wireshark & double click on name of network interface.

colour rules.

→ Colours have been assegined for each packed view → following rules.

Filturing packets:

→ display ouchuly.

→ type into lelue box at top of windows & clicking apply.

TCP conversion.

→ right click on packet → follow → TCP → stream Inspect packet.

→ click a packet to view details of packet. dig down.

Flowgraph:

→ network untiface → statistics → Flow graph.

Students abrevation.

1) what is promisious mode?
A network interfaces Card that capture all traffic on network.

2) Does ARP packet has transport layer explain.

3) which transport layer protocol is used by one UDP

4) Pocket number used by HTTP Protocol.
→ 80

5) What is the bewadcast IP address?
Used to send a data to all device on a network. for IPV4 it is highest address in subnet.

Colouring rules:-

Result:

This the packet connectity tool wireshark is installed & executed.

Ex NO: 6
Date: 10/8

Error Detection & Connection using Hamming code.

Aim:

Write a programe to element deleliton & corese atton using harmming code make a test ~~was~~ even to unput data and verify error feature.

Create sender programe with below feature → Input to sender file should be a text of any length. programe is connect to disk to binary.

→ Apply hamming code concept on binary data and add redundant bits to it

→ Save this output in a file called Channel.

→ save this oupput in a file called

→ Create a program with below feature

→ receive program should read the unput forom channel file.

→ apply hamming code on binary data to check.

→ If ustuce an eveou display the position of euoer.

→ else elomove the eeduntant bits & convert the binary data to ASCII & display output.

student obersavation :

code :

Unpoid math.

```
def calculate - redundand - bits (k)
    for r is remage (1,12)
  if (2 ** r >= r + k + 1):
    return r
did positioning redundacy - bil (data, V, K)
        result append (1,1)
  for i, inrange (1, K+r+1);
        if (2 ** j == 1)
            j = j + 1
        result . append (data [p])
            P = P - 1
    return result [::-1] ;result .
def building - ham - code - code (data K, V)
        j = 0
```

```
j=0
c=0
for t in range (1, k+r+1):
    if (t==2**j).
        i = 2**j
        p= t
        c=0
while (p< k+r+1):
    for n in range (p, p+i)
    try:
        if (data[n]== '1' and
            p<k+0+1);
    c+=1
    except Index:
    p = p+i+1
    if (ct-2==0)
        data [i] = '0'
else
    data [i] = '1'
    j = j+1
if (Flag==0)
    Print (Bwult hamcode "+" jm
                [data [::-1])
def checking - ham - code (data, k,r)
    j=0
    c=0
```

```
for t in range (1, k+r+1)
    if (t==2 **j);
        i = 2**j
        p = f
        c = 0
while (p<k+r+1);
  for h is range (p, p+i)
  try ;
    if (data [h] == '1' and h(k+r+i).
        c+ = 1
  except index error
    break.
p = p+1+i
if (c%2 1 =0);
    position - rend bit (data, k, r)
  return Dos
  j = j+1
dest position - Ved bid (data, k, r);
  j=0
  Pos = 0
  t = building ham-code (data k, r)
    for i in range (1, k+r+1)
      if (i==2**j)
      Pos (10**j) * int data [i] + Pos
        j = j+1
```

```
Print ("encour position" + ssts
    entieum unit (str (pos), 2)
    c = input ("enter the catuing
to send")
    data = ", join (formate (ord (i),
    v = len(data)
    r = calculate - eudundant - bits(k)
Prient ("The string in bindary for
    join (data))
Prient ("The string after eudundant
is unpriented" + " join (data))
    Kag = 0
    data = building - ham-code (k, r)
    Kag = 1
Print ("transmission Line")
n = int (input ("Entun the position to
Change during transmision"))
if (math . sqrt (m) = math thoorimat
                sqrt (m)
Print ("you can't change the eudendant
```

```
else
    data [n]
Print ("Aft
    h = " " jo
    pos = unchu
    h [pos -
Print ("Th
    for (
output.
    enter th
The string
The string aft
    01110
Build harmcod
transmission
inter the po
After equ
error posi
```

```
else :
    data[n] = '0' if [data[n] = '1' ]else'1'
Print ( " After error " + " ".join (data [C :-1])
  h = " " join ( data[ : ])
  Pos = unching - ham - code (data , e,v)
  h [ Pos - 1] = "0" if [ h [ pos - 1 ] = "1" )
                                  else "1"
Print ( " The corrected error string : " +
        Jor ( h [1 :- 1]);
```

output.

enter the string you want to send red

The string in binary form : 011100100 1100101011 00100

The string after redemdent bit .

0111010011001010110101010101

Build harmcode : 0111001001100001 0101101 010000

transmission time ....

inter the position you want to change : 3

After equan : 0111001001100010101110 10100100

error position : 3

Corected string : 0111001 0/1100010101110
0100000

Result :

The code to buildings & checking the hamming code is successfully written and the output is verified.

EXNO: 7
Date : 6.9.24

Aim:

Write a programe to implement flow control at data link layer using sliding window control :simulate the flow of frams from one node to another

Create a sender program with following feauture.

1) Input window size from the user
2) Input a text messages from the user.
3) Consider 1 chaualate with following fuild [frame no , data]
4) Create frame with following fuild [frameno, data]
5) Send the frames.
6) wait for the auc from the exercise.
7) read a file called received - buffer
8) Check ack feld for the ACK number
9) If the Ack number is as expected send how set of frame accoudingly.

feature.

1) Readuce a file called sendeure - buffer

2) Check the frame no.

3) If the frame no are as expected write the opperopicate ack no in the receiveres bulfar file else write mack no is the exceeves buffer file.

Code:

```
impout time
impout threading
unpoint os
Clean slidding windowprotocal
def - Unit - (self, window protocol)
self window - size = windows - size
self frame - no = 0
self expected - ack - no = 0
self expected frame - no = 0
self sender - buffer = sender - buffer text "
self recever - buthes = "receiver - buffer text"
self sender - done = False
self done = False.
```

```
def sender (self)
while not self sender-done
frames = [ ]
Print (f"\n -- SENDING FRAMES (window size)
{ self windo-size}) ...")
Sou 1 in exchange (self window size)
if self frame-no < len ( self. message)
frame data = f"[frame no:
{Self-frame. DATA {self-manage [ selfFram]
Frame append -no+=1
if frames
with open (self-sender-buffer,"w") al
s. write ("\n" join ( frames)+"\m" )
if self frame no >= len (self menage):
self sender-done = True
Print ("All frame sent")
Print ("waiting you ack")
time sleep (2)
self. wait - for - ack ()
def wait - for - ack (self)
try:
: written (open (self - receive buffer,r")
ack - data = 1 read lines ()
```

```
does not excited waiting for frames...
continue
with open (self sender - buffer "r") as
    frame = s. readlines()
    if not frame
        continue
    Preint ("\n - - Recering frames...")
    ack - to-send = []
    for frame in frames
        frame-no - data = frame strip () split(",")
        frame - no = int (frame no split)(",")[1])
        data = data split (",")[1]
written open (self exercices buffer "w") as
Print ("All frames receives bstopping receiv
time sleep (2)
if - - name - - "main":
window - size - int (input) (inter window
                                    size)
sender - Throd = threading thread (target =
                    Pacolocol receiver)
sender - thread (start()
receiver - Thread start()
```

```
expert FileNot found error:
    print (f" receiver buffer dele {self
        receiver - buffer} not found")
return
for ack in ack - data
    ack - type , ack-no= ack strip().split()
    ack - no = int (ack - no)
if ack - type = "ACK"
    if ack-no = self expected - ack - not )
    print{{" ACK } ack - no}
self frame - expected - ack no- ack-no
else
print { f" unexpeted ACK received
expted { self expected - ack - no+1 }, got {ack-
    sending")
    while not self execeive - done
    time sleep (2)
    if not as path exush (self-sender-
                                back)
    Print (f" sender buffer' self - sender-
                                buffer}
```

sender - thread join ()

receiver - thread join ( )

Present ("\n Transmission completed)

output :

enter window size 4

enter text massage cute

Sending frame (window size 4)

sent : [Frame no:0 , Data C]

sent [Frame no 1 , Data V]

sent [Frame no 2 , Data T]

sent [Frame no 3, Data E]

all Frames sent

Waiting for ack

time pected Ack received expected

got 4 sending.

Ex.NO: 8a)
DATE: 7/10/24

Aim:
Simulate virtual LAN configuration using CISCO packet trace simulation

Reflection questions:

1) what is needed to allow hosts on VLAN 10 to communicate to hosts on VLAN 99?

To allow hosts on VLAN 10 to to communicate with hosts on VLAN 99, a router or Layer 3 switch is needed to perform inter-VLAN routing, enabling traffic to pass between the two VLANS

2) What are some primary benefits that an organization can receive through effective use of VLANS?

Primary benefits of effective VLAN use include improved network security through segmentation which limits broadcast traffic & enhanced network management.

Student Observation:

a) Draw & label the VLAN

b) Show the ip configuration for each device

ip Address : 192.168.10.2

Subnet Mask: 255.255.255.0

IP Address : 192.168.10.3

Subnet Mask: 255.255.255.0

IP Address : 192.168.99.2
Subnet Mask : 255.255.255.0

c) Write the commands used for VLAN configuration in switch

switch > enable
switch
switch (config)
switch(Config-vlan)
switch (config)
switch (config-vlan)
switch (config)
switch (config-if-range)
switch (config-if-range)
switch (config)
switch (config-if)
switch (config-if)

Result:
Thus the VLAN has been setup in the cisco traces.

Ex. No: 8 b)
Date: 7/10/24

Aim:
Configuration of wireless LAN using cisco packet tracer

a) What is SSID of a wireless router?
SSID is the name of a wireless network. It is used to uniquely identify a wireless LAN.

b) What is a security key in a wireless router?
A security key is a wireless router is a password (or) encryption key used to secure a wireless network.

Result:
Thus, the wireless LAN has been configured using cisco packet tracer & the output is verified.

Exp. 9
Ex: 2019

Classful Subnetting

Aim:
Implementation of subnetting in cisco packet tracer simulator.

Procedure:-
1) Create the network using switches, router and PCs
2) The IP address will be as follows
→ Router R1
* Gigabit Ethernet 0/0: 192.168.1.1
* Gigabit Ethernet 0/1: 192.168.2.1
Also enable the 'On' option for both gig ethernet on the page.

→ switch S1
* no IP
→ LAN-1
. PC 0
IP address: 192.168.1.11
Gate way: 192.168.1.1

. PC 1
IP: 192.168.1.12
gate way: 192.168.1.1

. PC 2
IP: 192.168.1.13
gateway: 192.168.1.1

→ Switch S_2

No IP

→ LAN-2

PC5

IP address : 192.168.2.11

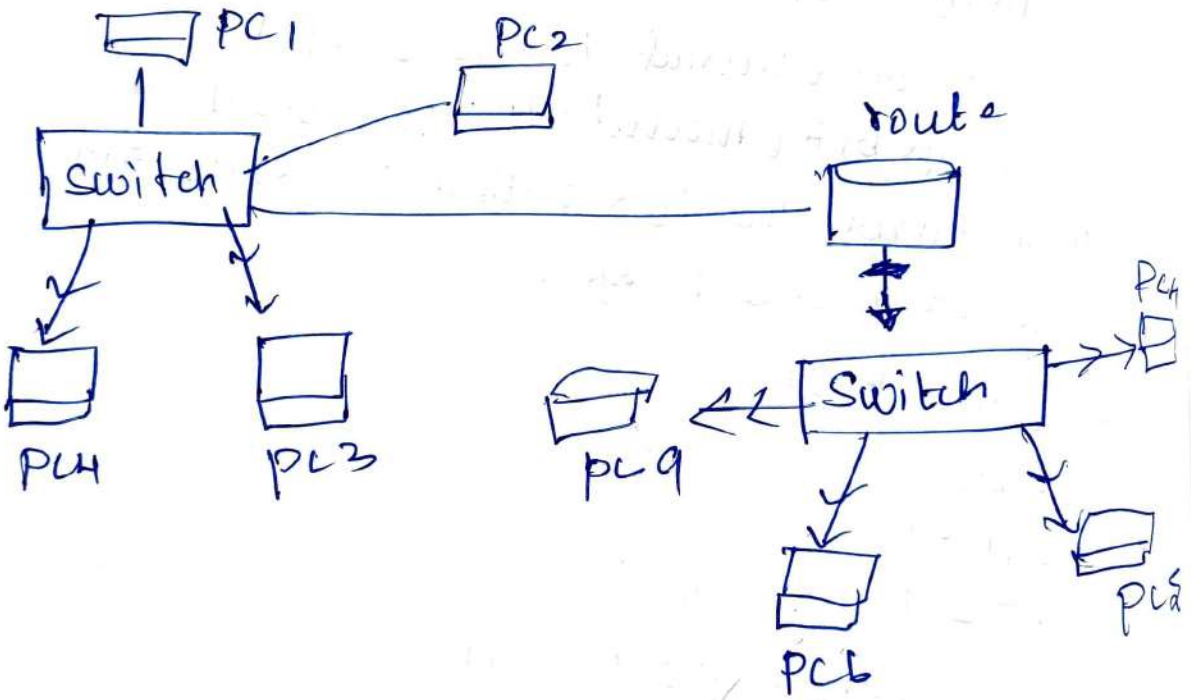
Gateway : 192.168.2.1

PC6

IP : 192.168.2.12

gateway : 192.168.2.1

Diagramatic representation.



Output:

Now lets assume the sender is PC1 & receiver PC7

While simulating & observing, we get the simulation panel

Simulation panel
event last.

Vis	time	Last device
	0.000	- - -
	0.003	PC1
	0.005	switch1
	0.008	router1
	0.010	switch2
	0.013	PC9
	0.015	switch2
	0.018	Router1
	0.020	switch1
	0.023	- - -

Reset simulation ▭ constant delay

play controls

|◁◁ ▷ ▷||

Fire Last status Source ds Type color
successful PC1 PC9 ICMP 0.000

Student observation

a) Write down the understanting of subnetting. is the process of sending a large IP network and manageable section called subnets. each subnet act as independent network & allow devices connected to it to communicate within the subnet & control traffic b/w subnet.

b) Advantages of implementing subnetting a network.

efficient IP management - based on required

reduce network congestion - Limit broadcast traffic to individual subnet.

Result:

The implementation of subnetting in CISO has been done successfully

RIP.

EX. NO: 11b

DATE: 18-10-24

Aim:

To simulate RIP using cisco packet tracer.

Procedure:

1. create network as using 2PC & 1 router as shown.

2. Assign IP address for PC3 & router ports.

PC0

IP - 10.1.1.1

Gateway: 10.1.1.2

PC1

IP - 200.1.1.1

Gateway - 222.2.2.1.2

PC2

IP - 222.2.2.2

Gateway: 222.2.2.1.2

Router 3

gig 0/0 - 20.1.1.1

0/1 - 192.163.1.1

0/2 - 10.1.1.1

Router 2

gig 0/0 - 20.1.1.2

0/1 - 172.1.1.1

0/2 - 200.1.1.2

Router 1

gig 0/0 - 217.1.1.2

0/1 - 222.2.2.12

3) click on router 3

→ click config → RIP

→ enter network 10.0.0.0 → Add

→ " " 20.0.0.0 → Add

→ " " 192.168.1.0 → Add

Thus step is added

4) Do same for router 2, 1, 24.

Router 2 → config → RIP

→ 20.0.0.0 → add

→ 172.1.0.0 → add

→ 200.1.1.0 → add

router 1 → config → RIP

→ 172.1.0.0 — add

→ 192.168.1.0 — add.

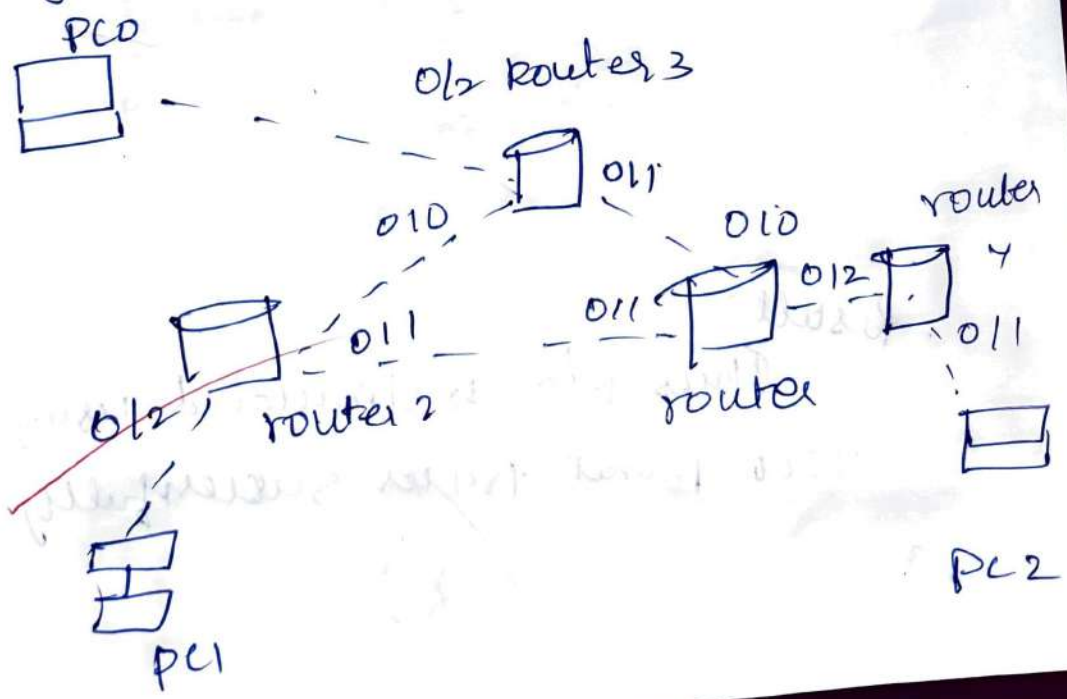
→ 217.1.1.0 — add

Router 4 → Config → RIP
→ 217.1.1.0 - add
→ 222.2.2.0 - add.

c) Now to display the running table click on router (say router 1)
→ then on CLI & type the command
exit
exit
show ip route.

Output:
R.10.0.0.0/8 via 192.168.1.1 gig 0/0
R.20.0.0.0/8 via 192.168.1.1 gig 0/0
0.172.1.0.0/16 via variable connected 2 subnet 2 mask.

Diagramatic representation:



Ex No: 10
Date 29/10

Aim:
To de
using wireless

Procedure:
1. Arran
2. Assign
se

PC0
—
Lapt

PC1

La

P

Now

Result:
Thus RIP is simulated using cisco packet tracer successfully

Ex No: 10b Routar

Date 29/10

Aim:

To design and configuer an internetwork using wireless router DHCP Server & Internet cloud.

Procedure:

1. Arrange the pc & Laptop in the page
2. Assign in the IP address.

Server - 10.0.0.1

↳ services → DHCP → default gateway - 10.0.0.1
→ Service-on

PC0
- Desktop - DHCP ⊙

Laptop 0
- Desktop - DHCP ⊙

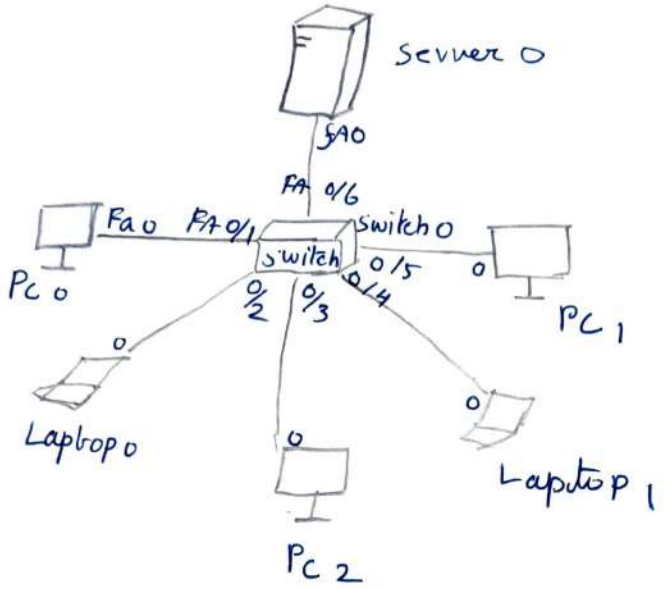
PC1
- Desktop - DHCP ⊙

Laptop 1
- Desktop - DHCP ⊙

PC 2
- Desktop - DHCP ⊙

Now ping from PCs to Servies.

Diagram



Output :

Pkg Laptop to PC_2

Event List

Vis	Time (sec)	Last device
	0.000	--
	0.001	Laptop 0
	0.002	Switch 0
	0.003	Pc 2
	0.004	switch 0
Visible	1.994	--

Result

Thus the Configuration of DHCP has been successfuly executed.

Exp 11a
Date

Static Routing Configuration

AIM :

To Simulate static routing configuration using asto packet tracer.

Procedure:

1) Arrange PC & Router in the manner the image below.

2) Assign IP address for routers and PCs

Routers 0 [2811]

→ IP FA 0/0 - 192.168.1.2
FA 0/1 - 192.168.2.3

Router 1 2811

→ IP FA0/0 - 192.168.1.4
FA0/1 - 192.168.3.3

PC0

IP - 192.168.2.7
gateway 192.168.2.3

PC1

IP - 192.168.2.3
gate way - 192.168.2.3

PC2

IP - 192.168.3.5
Gateway - 192.168.3.3

Pc 3

IP - 192.168.3.7

gateway-192.168.3.3

3. Now click on router 0

then config → static

Network - 192.168.3.0

Subnet - 255.255.255.0

nexthop 192.168.1.4

4. Now click on Router 1

then Config → static

Network - 192.168.2.0

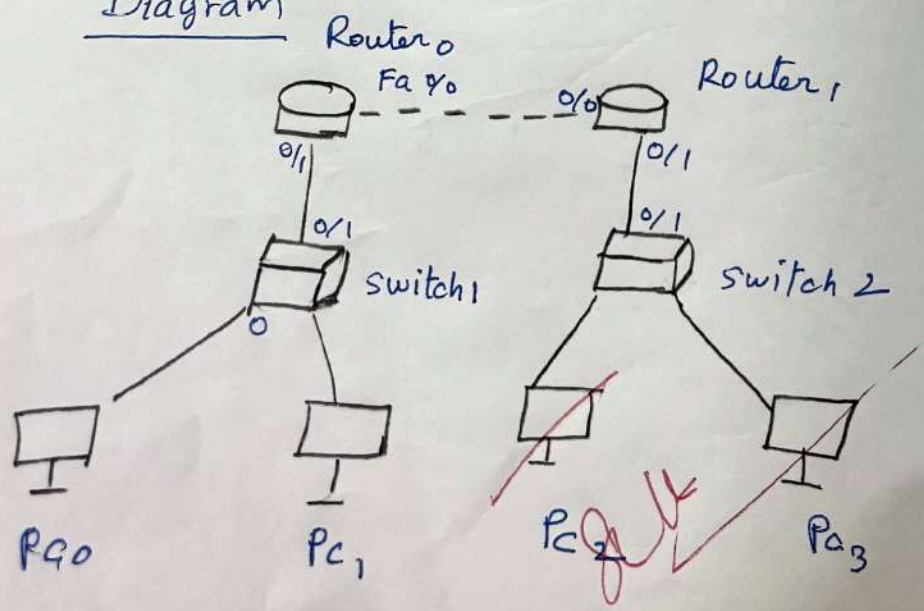
Subnet - 255.255.255.0

Nexthop - 192.168.1.2

~~Output~~.

Pinding ~~from~~ PC0 to PC2

Diagram



output

Ping from Pc

VIS	Time(sec)
	0.000
	0.001
	0.002
	0.003
	0.004
	0.005
	0.006
	0.007
	0.008
	0.009
	0.010

Fire	Laststatus
⊙	successfull

Result

Th

has bee

output

Ping from Pc0 to Pc2

VIS	Time(sec)	Last devices
	0.000	—
	0.001	Pc0
	0.002	Switch 0
	0.003	Router 0
	0.004	switch 1
	0.005	Router 1
	0.006	Pc 2
	0.007	switch 1
	0.008	Router 1
	0.009	Router 0
	0.010	switch 0

Fire	Last status	source	destination	type	colour	time (sec)	Period
⊙	Successfull	Pc0	PC2	ICMP		0.000	N

Result

Thus status routing configuration has been sussfully run & executes.

Exp 12a
Date 15/10

AIM:

- To implement echo create server, TCP/UDP sockets.

Algorithm:

Server-by:

→ Create a UDP Socket
→ Bind the socket to specific IP address (127.0.0.1) & Port (12345)
→ Contifrious listen for incoming message
→ when message received - decode it
→ display message along with sender address
→ Repeat infinetely

Client.py

→ Create UDP socket
→ Set a timeout for socket to avoid waiting
→ Send a predefined message hellow to Server IP address & port 12345
→ If no response received in finiest period Print timeout message.
→ Close socket after sending message.

CODE

Server-py

impost socket
def start - server
with socket

s. bind
print (F

While true
data, ad
pri

if -- name
Start

Clent - py.
dy Ping

with soc

S. § time
try
S.

CODE

Server-py

```
impost socket
def start - server (host = '127.0.0.1' post=12345);
  with socket . sockst (socket.AF.INET,
                        Socket . SOCK. DGRAM)
                                as s;
    s. bind ((host, post)).
    print (F"UDP server running on
                        {host};{port}")
    While true :
      data, addr = s. recvform (1024)
        print (F'received message from
                {addr}; decode ()}")
if -- name -- == "-- main --";
    start seiver ( )
```

Clent - Py.

```
dy Ping - Server (host = 127.0.0.1', port=1234
                                    5);
with Socket. Socket (Socket. AF. INET. Socket.
                                    DGRM)
                                    as s:
s. timeout +(s)
try ;
    s. Send to ( b' Hellow', (host, port))
```

```
Print ("message sent to server")
exept socket.timeout ;
    Print ("Request timeout")
if --name-- == "--main--"
    Ping-server()
```

Output

Server.py

Terminal

> Pylhon server.py

>>

udp server running on 127.0.0.1:12345

Clent.py

Terminal

> Paython Pruent.py

>>

message sent to server.

Server terminal;

Received message from ("127.0.0.1", 5000

Hello

Result

Thus the program of echo Clene server using UDP socket has been implemented & executed successfully.

Exp 13

Date 18/10

AIM

To im

Algorithm:

UDP Server

→ Create U

address &

→ wait for

→ Print mess

→ Send ba

UDP Client

→ Create UD

→ Send 'P

→ if a re

Re

→ if n

CoDe

Server.

import

dy start

Exp 13

Date 18/10

PING PROGRAM

AIM

To implement your own ping program

Algorithm:

UDP Server

→ Create UDP Soket & bind it to a subjects address & port

→ wait for message

→ Print message & clients address

→ Send back Pong to Client

UDP Client

→ create UDP socket & set a 2sec timeout

→ Send 'Ping' to server

→ if a response ("ping" recieved : Print Response & calculate RTT

→ if no response within 2sec Print request time out.

CoDe

Server.Py

```
import Socket
def start - sever (host = '127.0.0.1', Port
                    = 12345);
```

```
with socket.socket (socket.AF.
                    socket.sock_DGRAM)
Print (f "UDP server running on {host}; {
while true
    date, addr = s.recieved (1024)
    Print ("f" recewied message from
        {addr}; {date.decode ()}
    s.send to (b'pong', addr)
if __name__ == "__main__" ;
    Start_server ()
```

client.py

```
import time
import socket
def Ping_server (host = '127.0.0.1; Port=
    with socket.socket (socket.AF.INET,
                        Socket.sock_DGRAM)
try :
    S.Set time (2)
    Start = time.time ()
    S.Send to (b "ping", (host, port))
```

```
data , addr
data , addr =
end = time.
print (f "
    from {
except Socket
    print ("
if __name__
    Ping_
```

Output

Term

> Python

UDP serv
on 127.0

Received
(127.0

```
data , addr = s.recvfrom(1024)
data , addr = s.recv from(
end = time.time()
Print(f" Recved {data.decode()}
    from {addr} in {end - start:.2f}
                seconds")

except Socket.timeout ;
    Print("Request timed out")

if __name__ = "__main__";
    Ping - Server()
```

Output

Terminal	Terminal
> Python Server.py UDP sever running on 127.0.0.1 :12345 Recved message from (127.0.0.1, 50061 : Ping	> PythonClient.py Recved Pong from (127.0.0.1 ; 12345) in 0.00 seconds.

Result

Thus a Ping Program been executed successfully.

EX: NO 4 PACKET

DATE: 22/10/24

AIM:

Implement pa

sockets

CODE:

from scapy

from scaps

def packe

if ip i

in

p

s

d

Protocol:

if proto

p

elif pr

elif

else:

p

EX: NO: 4

PACKET SNIFFING

DATE: 22/10/24

AIM:

Implement packet shiffing using RAW sockets

CODE:

```
from scapy.all import sniff
from scapy.layers.inet import IP, TCP
                UDP, ICMP

def packet_callback(packet):
    if IP in packet:
        ip_layer = packet[IP]
        protocol = ip_layer.proto
        src_ip = ip_layer.src
        dst_ip = ip_layer.dst
protocol_name = ""
if protocol == 1:
    protocol_name = "ICMP"
elif protocol == 6
    protocol_name = "TCP"
elif protocol == 17
    protocol_name = "UDP"
else:
    protocol_name = "unknown protocol"
```

```
print("protocol : {protocol_name}")
print(f"source IP: {SRC_IP}")
print(f"Distination ID : {dst_ip}")
print("_"*50)
def main():
    sniff(iface="wlpi", prn=packet_call
        filter="ip", store=0)
if.name == "__main__"()
```

Output :

Protocol : TCP
Source IP : 20.247.184.142
Destination IP : 172.20.10.2
protocol : TCP
source IP :
Dest IP : 172.20.247.187
protocol : TCP
Source IP : 20.247.184.182
Dest IP : 172.20.10.2

Result :
Thus packet sniffing using RAW packets has implemented

EX. NO: 15
DATE :: 25.10.24.

Aim :
→ To analyze of web logs

PROCEDURE :
1. Run
2. Inp
3. Pres

INPUT :-

Input :
log fil

Cl

T

EXP. NO: 15

DATE: 25.10.24.

WEB LOGS

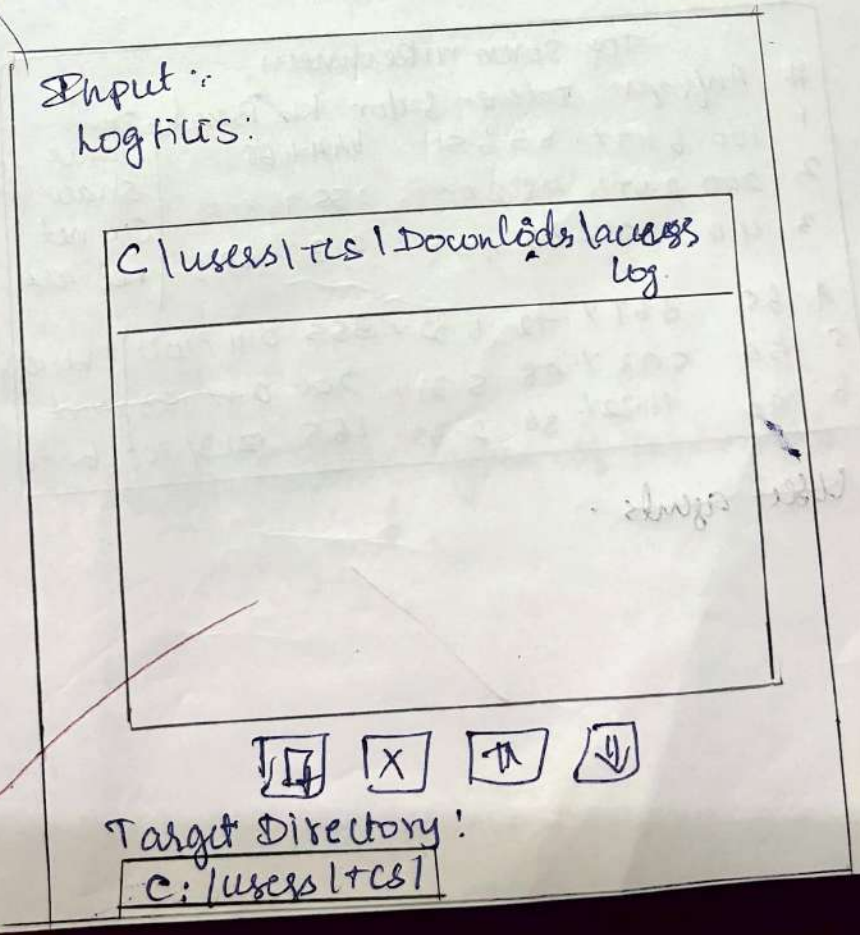
Aim:

→ To analyze the different types of web logs using webalize tools.

PROCEDURE:

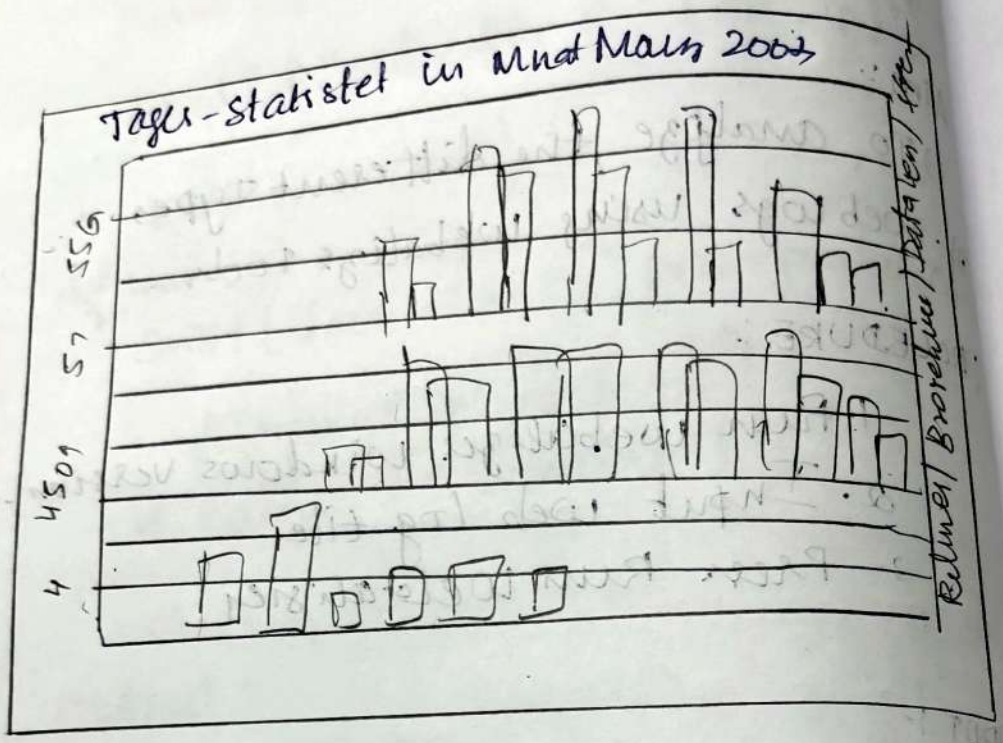
1. Run webalizer windows version.
2. Input web log file.
3. Press Run Webalizer

INPUT:-



OUTPUT

MONTHLY STATISTICS



HOSTS.

TOP 20 von 171 Rechnern

#	Anfragen	Daten	Sektor	Kb	Besuche	Dom	name
1	100	6.47%	83	6.51%	644.468		shawville
2	200	3.67%	48	200%	235.78%	32	e.c.net
3	44	2.85%	38	4.65%	0.32%	500	net.au
4	65	3.67%	72	6.32%	355	0.11%	100	thl.com
5	50	5.93%	98	5.34%	200	0.12%	150	aol
6	92	11.32%	36	2.32	155	21.3%	200	bc.ca

~~Use agents~~.

Top 1 verweise

Anfragen Besuht verweis

1 1546 100.00% 234 100.00%
(Discrete)

Anfragen Kb Besuch Ansveder

1 1546 100.00% 10649 100.00%
234

Unbekannte
Adress
(100%

use Agents

Result:
Thus different types of web logs was analyzed.