The Internet

Part 2: Networks of Networks - Internet Workings

Notes for CSC 100 - The Beauty and Joy of Computing
The University of North Carolina at Greensboro

Reminders

Reminders for Nov. 10

Blown to Bits

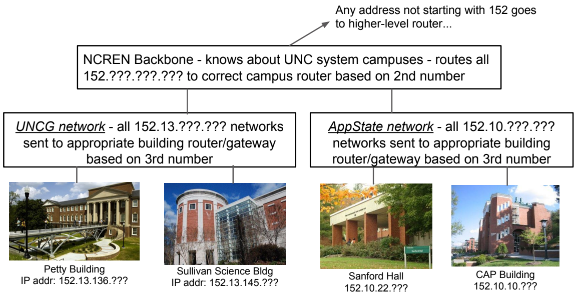
- Chapter 5: Contribute to online discussion by Wednesday
- Chapter 6: Start reading - reflection due Wed., 11/19

Project

- Proposals were almost all very good (check for feedback)
- Should be working on coding (assistance in Lab on Friday)
- Progress report due Friday (Nov. 14) at 5:00

Internet Protocol

Routing: Simplified



Any address not starting with 152 goes to higher-level router...

NCREN Backbone - knows about UNC system campuses - routes all 152.???.???.??? to correct campus router based on 2nd number

UNCG network - all 152.13.???.??? networks sent to appropriate building router/gateway based on 3rd number

AppState network - all 152.10.???.??? networks sent to appropriate building router/gateway based on 3rd number

Petty Building
IP addr: 152.13.136.???

Sullivan Science Bldg
IP addr: 152.13.145.???

Sanford Hall
152.10.22.???

CAP Building
152.10.10.???

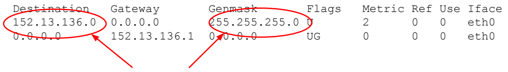
Routing
Introduction to Routing Tables

Routing controlled by “routing table” at each host/router/device - example:

```
Destination    Gateway        Genmask        Flags   Metric Ref Use Iface
152.13.136.0   0.0.0.0        255.255.255.0  U       2      0   0   eth0
0.0.0.0        152.13.136.1   0.0.0.0        UG      0      0   0   eth0
```

Routing
Introduction to Routing Tables

Routing controlled by “routing table” at each host/router/device - example:



```
Destination    Gateway        Genmask        Flags   Metric Ref Use Iface
152.13.136.0   0.0.0.0        255.255.255.0  U       2      0   0   eth0
0.0.0.0        152.13.136.1   0.0.0.0        UG      0      0   0   eth0
```

These are the most important things to understand

Your destination (who you want to send to) is matched against these specifications

1. Consider everything in binary
2. Everywhere there’s a “1” bit in the “Genmask” (also called a netmask) must match “Destination”
3. Where there are 0’s in the “Genmask” it doesn’t matter if it matches or not.

Everybody’s first question: Do I really have to convert everything to binary to understand this?!?

Answer: Usually not - the “Genmask” typically only uses a few values, mostly “0” and “255”

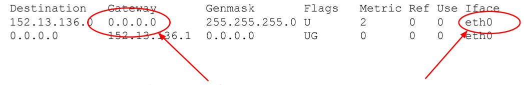
- “0” for an octet means you don’t care about this octet
- “255” for an octet means the whole octet must match

So… 152.13.136.0/255.255.255.0 matches any address that starts with 152.13.136

- Matches 152.13.136.12, 152.13.136.19, 152.13.136.252, ...

Routing
Introduction to Routing Tables

Example routing table (most of yours look like this):



```
Destination    Gateway        Genmask        Flags   Metric Ref Use Iface
152.13.136.0   0.0.0.0        255.255.255.0  U       2      0   0   eth0
0.0.0.0        152.13.136.1   0.0.0.0        UG      0      0   0   eth0
```

If Destination/Genmask matches, this tells us what to do

Example 1: Sending to 152.13.136.53 - matches first line in routing table

There is no “intermediate host” specified as a “gateway” and communication happens over interface “eth0”

So… Use the ARP protocol to find that host via “eth0”, and send using the resulting MAC address

Routing

Introduction to Routing Tables

Example routing table (most of yours look like this):

```
Destination    Gateway        Genmask        Flags   Metric Ref Use Iface
152.13.136.0   0.0.0.0        255.255.255.0  U       2      0   0   eth0
0.0.0.0        152.13.136.1   0.0.0.0        UG      0      0   0   eth0
```

Does not match!

Example 2: Send to 152.13.15.23

Routing

Introduction to Routing Tables

Example routing table (most of yours look like this):

```
Destination    Gateway        Genmask        Flags   Metric Ref Use Iface
152.13.136.0   0.0.0.0        255.255.255.0  U       2      0   0   eth0
0.0.0.0        152.13.136.1   0.0.0.0        UG      0      0   0   eth0
```

Does match! (0.0.0.0/0.0.0.0 matches everything - called the “default route”)

Example 2: Send to 152.13.15.23

Routing

Introduction to Routing Tables

Example routing table (most of yours look like this):

```
Destination    Gateway        Genmask        Flags   Metric Ref Use Iface
152.13.136.0   0.0.0.0        255.255.255.0  U       2      0   0   eth0
0.0.0.0        152.13.136.1   0.0.0.0        UG      0      0   0   eth0
```

This is what you do...

Example 2: Send to 152.13.15.23

This time a gateway is defined - we send to 152.13.15.23 by sending through 152.13.136.1

152.13.136.1 is connected on intervace eth0

So use ARP on eth0 to find MAC address, but of the gateway (152.13.136.1) not the destination!

Packet goes to 152.13.136.1, addressed to 152.13.15.23 - now it’s not your problem any more

Trust the network!

Internet Protocol

Routing: Simplified - Let's try it!

Demo setup:

- Our IP address is 10.0.2.5
- We want to send a packet to 10.0.2.4

Routing table for our demo:

```
Destination   Gateway       Genmask        Flags   Metric  Ref  Use  Iface
10.0.2.0      0.0.0.0       255.255.255.0  U       1       0    0    eth0
0.0.0.0       10.0.2.1      0.0.0.0        UG      0       0    0    eth0
```

So: This is a local packet (how can you tell?)

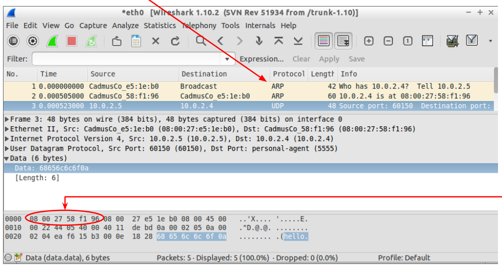
__Question__: What are the steps taken to send the packet?

Internet Protocol

Routing: Simplified - Let's try it!

Arp to find MAC address of destination

Sending the actual packet - note destination MAC filled in


*eth0 [Wireshark 1.10.2 (SVN Rev 51934 from /trunk-1.10)]
File Edit View Go Capture Analyze Statistics Telephony Tools Internals Help
Filter:
Expression... Clear Apply Save

```
No.  Time         Source             Destination        Protocol Length Info
   1 0.000000000  CadmusCo_e5:1e:b0  Broadcast          ARP       42 Who has 10.0.2.4?  Tell 10.0.2.5
   2 0.000505000  CadmusCo_58:f1:96  CadmusCo_e5:1e:b0  ARP       60 10.0.2.4 is at 08:00:27:58:f1:96
   3 0.000523000  10.0.2.5           10.0.2.4           UDP       48 Source port: 60150  Destination port:
```

```
▶Frame 3: 48 bytes on wire (384 bits), 48 bytes captured (384 bits) on interface 0
▶Ethernet II, Src: CadmusCo_e5:1e:b0 (08:00:27:e5:1e:b0), Dst: CadmusCo_58:f1:96 (08:00:27:58:f1:96)
▶Internet Protocol Version 4, Src: 10.0.2.5 (10.0.2.5), Dst: 10.0.2.4 (10.0.2.4)
▶User Datagram Protocol, Src Port: 60150 (60150), Dst Port: personal-agent (5555)
▼Data (6 bytes)
   Data: 68656c6c6f0a
   [Length: 6]
```

```
0000  08 00 27 58 f1 96 08 00  27 e5 1e b0 08 00 45 00   ..'X.... '.....E.
0010  00 22 44 05 40 00 40 11  de bd 0a 00 02 05 0a 00   ."D.@.@. ........
0020  02 04 ea f6 15 b3 00 0e  18 28 68 65 6c 6c 6f 0a   ........ .(hello.
```

Data (data.data), 6 bytes   Packets: 5 · Displayed: 5 (100.0%) · Dropped: 0 (0.0%)   Profile: Default

Routing

Sample Gateway Routing Table

More complex routing table for a gateway:

```
Destination   Gateway       Genmask          Flags Metric Ref  Use  Iface
152.13.136.0  0.0.0.0       255.255.255.0    U     2      0    0    eth0
152.13.10.0   0.0.0.0       255.255.255.0    U     2      0    0    eth1
152.13.145.0  152.13.10.13  255.255.255.0    UG    0      0    0    eth1
0.0.0.0       152.13.10.1   0.0.0.0          UG    0      0    0    eth1
```

Notes:

- Two different interfaces: "eth0" and "eth1"
- Each one defines an IP prefix for direct connections (152.13.136 and 152.13.10)
- "Helper entry" for 152.13.145 prefix specifies specific gateway for that network
  - *This isn't strictly necessary but can speed things up*
- Default route - anything not matching three given prefixes in 152.13 goes to 152.13.10.1

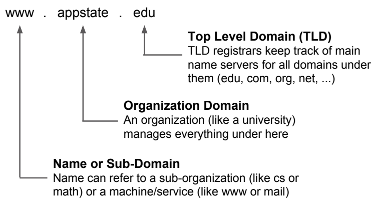
__Question__: What happens when 152.13.10.7 sends a packet to 152.13.145.83?

But I know a name, not an address!

Naming on the Internet

I want to connect to a name (e.g., www.appstate.edu) rather than a number

Names are also hierarchical



www . appstate . edu

Top Level Domain (TLD)
TLD registrars keep track of main name servers for all domains under them (edu, com, org, net, ...)

Organization Domain
An organization (like a university) manages everything under here

Name or Sub-Domain
Name can refer to a sub-organization (like cs or math) or a machine/service (like www or mail)

But I know a name, not an address!

Mapping from a name to an address: DNS

DNS = "Domain Name System"

DNS servers map from names to IP addresses (and vice-versa, sometimes!)

Super-simplified view:

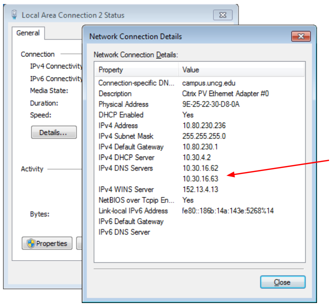
- I know the IP address (not the name!!!) of a DNS server I can use
- I ask it for the IP address of www.appstate.edu
- It returns IP address (152.10.1.83)

View from Unix utility "host":

```
user@host ~ $ host www.appstate.edu
www.appstate.edu has address 152.10.1.83
user@host ~ $
```

But I know a name, not an address!

Locating DNS servers in Windows 7



Network Connection Details:

Property	Value
Connection-specific DN...	campus.uncg.edu
Description	Citrix PV Ethernet Adapter #0
Physical Address	9E-25-22-30-D8-0A
DHCP Enabled	Yes
IPv4 Address	10.80.230.236
IPv4 Subnet Mask	255.255.255.0
IPv4 Default Gateway	10.80.230.1
IPv4 DHCP Server	10.30.4.2
IPv4 DNS Servers	10.30.16.62 10.30.16.63
IPv4 WINS Server	152.13.4.13
NetBIOS over Tcpip En...	Yes
Link-local IPv6 Address	fe80::186b:14a:143e:5268%14
IPv6 Default Gateway	
IPv6 DNS Server	

Note these DNS servers are for IPv4

Sending to a Name

Question: What happens when sending to csdept.appstate.edu?

Summary of addressing/naming

Full set of actions for previous question

Problem: Host at 152.13.136.16 wants to contact csdept.appstate.edu

- Need to find IP address of www.appstate.edu, so need to locate IP address of DNS server (from my settings) - in our example that's 152.13.10.15
- I need to contact 152.13.10.15 - checking my routing table, that doesn't match my network (line 1) but does match default route - gateway is 152.13.136.1
- I need to communicate with 152.13.136.1 (my gateway), but I need a MAC address, not an IP address! So I send an ARP packet "Who has 152.13.136.1?"
- Gateway responds: "I have 152.13.136.1 at 00:1d:92:97:a2:55"
- I receive this message and store this IP <-> MAC address mapping for later
- I send "To: 152.13.10.15 - DNS query: where is csdept.appstate.edu?" to MAC address 00:1d:92:97:a2:55 (and other networks send subsequent ARP/routing to get this to 152.13.10.15)
- I eventually receive a response "csdept.appstate.edu has address 152.10.10.45"
- I look at IP address: 152.10.10.45 is not local, so must go through gateway again
- ARP efficiency: we just saw that gateway was at 00:1d:92:97:a2:55, so I can just re-use that (no ARP needed for now - but will eventually "expire")
- I send "To: 152.10.10.45 - Packet for csdept.appstate.edu" to 00:1d:92:97:a2:55

Doesn't describe what happens in other networks, and...
Believe it or not, even this description is slightly simplified!

We can send packets, now what?

Packets are small: Typically under 1500 bytes

I want a picture - several hundred thousand bytes
… *Now what?*

Transport layer:

- UDP: Packet-by-packet communication
- TCP: Packets organized in reliable streams

Application layer examples:

- HTTP: Uses TCP streams to transmit from a web server
- DNS: Domain name service over UDP
- SMTP: Email transmission (server to server) over TCP
- IMAP: Email (server to client) over TCP
- SSH: Encrypted login over TCP
- ...

Where can things go wrong?

Situation: I want to privately access www.bankofamerica.com

What can go wrong?

Where can things go wrong?

Situation: I want to privately access www.bankofamerica.com

What can go wrong?

Issue 1: “privately” - packets are like postcards, with visible contents!
Problem: Any intermediate hop can see everything

Issue 2: Did the DNS lookup give me the correct IP address?
Problem: Can someone plant false records in my DNS server?
Problem: Can someone point me to a rogue DNS server?
Problem: Can someone send back a fake response (UDP is unreliable!)

Issue 3: Did I really connect to the stated IP address?
Problem: Could someone change routing tables to fake me out?
Problem: Could malware change IP address on the fly?

What could a well-funded party with access to Internet backbone do?

Where can things go wrong?

Situation: I want to privately access www.bankofamerica.com

What

Crypto to the rescue!

Issue 1: “privately” - packets are like postcards, with visible contents!
Problem: A *erything*

Encrypt contents for privacy

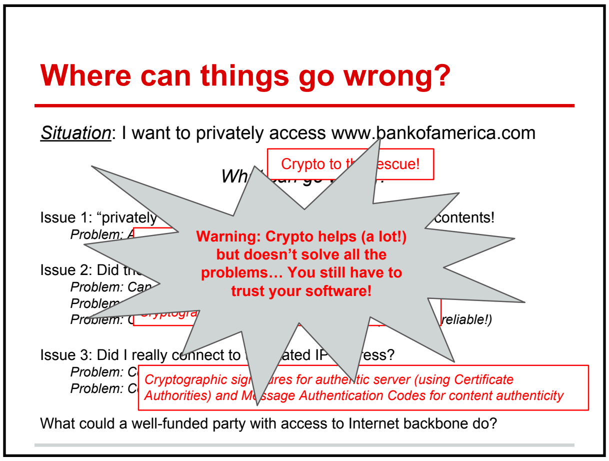
Issue 2: Did the DNS lookup give me the correct IP address?
Problem: Can someone plant false records in my DNS server?
Problem: C
Problem: C *reliable!)*

Cryptographic signature for integrity (DNSSEC)

Issue 3: Did I really connect to the stated IP address?
Problem: C
Problem: C

Cryptographic signatures for authentic server (using Certificate Authorities) and Message Authentication Codes for content authenticity

What could a well-funded party with access to Internet backbone do?



Where can things go wrong?

Situation: I want to privately access www.bankofamerica.com

Crypto to th escue!

Wh

Issue 1: “privately contents!

Problem: A

Warning: Crypto helps (a lot!) but doesn’t solve all the problems… You still have to trust your software!

Issue 2: Did th

Problem: Can

Problem

Problem: C *Cryptogra* *reliable!)*

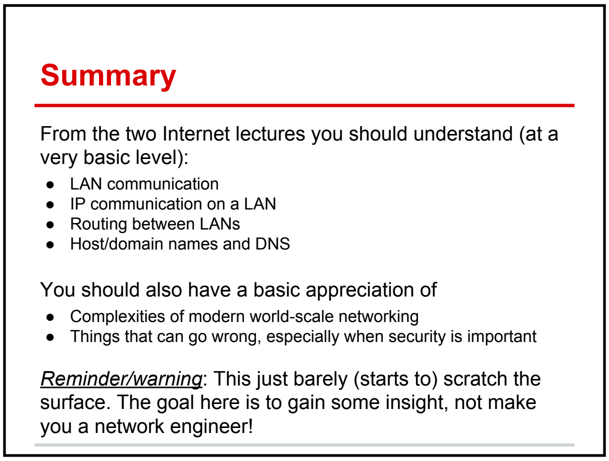
Issue 3: Did I really connect to ated IP ess?

Problem: C

Problem: C

Cryptographic sig ures for authe tic server (using Certificate Authorities) and M ssage Authentication Codes for content authenticity

What could a well-funded party with access to Internet backbone do?



Summary

From the two Internet lectures you should understand (at a very basic level):

- LAN communication
- IP communication on a LAN
- Routing between LANs
- Host/domain names and DNS

You should also have a basic appreciation of

- Complexities of modern world-scale networking
- Things that can go wrong, especially when security is important

Reminder/warning: This just barely (starts to) scratch the surface. The goal here is to gain some insight, not make you a network engineer!