Securing Internet Communication: TLS, cont'd Network security

CS 161: Computer Security

Prof. Raluca Ada Popa

Feb 27, 2018

Some slides credit David Wagner

Announcements

- Midterm grades released
- Regrades: Read the follow-ups on the threads to make sure your regrade request is somewhat warranted.
- No public repos for projects
- DSP letters
- Midterm 2 will be the Wed after spring break (Apr 4).

- Project 2 is live, it has 3 parts. **Prizes given!**
- Do not cheat. We have good tools and caught students already.

- Intro to Networking session Tuesday 8-10 at the Woz.

Certificates

- Browser compares domain *name* in cert w/ URL
  - Note: this provides an end-to-end property (as opposed to say a cert associated with an IP address)
- Browser accesses *separate* cert belonging to **issuer**
  - These are hardwired into the browser – and **trusted!**
  - There could be a *chain* of these …
- Browser applies issuer's public key to verify signature **S**, obtaining hash of what issuer signed
  - Compares with its own SHA-256 hash of Amazon's cert
- Assuming hashes match, now have high confidence it's indeed Amazon …
  - ***assuming signatory is trustworthy***

= assuming didn't lose private key; assuming didn't sign thoughtlessly

Certificates

- Want to use root CA as little as possible, access to the root key should be very infrequent
- Certificate chain:
  - Verisign can give a certificate to Google for google.com
  - Google can issue a certificate for finance.google.com

End-to-End ⇒ Powerful Protections

- Attacker runs a sniffer to capture our WiFi session?
  - (maybe by breaking crummy WEP security)
  - But: encrypted communication is unreadable
    - No problem!
- DNS cache poisoning gives client wrong IP address
  - Client goes to wrong server
  - But: detects impersonation
    - No problem!
- Attacker hijacks our connection, injects new traffic
  - But: data receiver rejects it due to failed integrity check
    - No problem!

Powerful Protections, cont.

- Attacker manipulates routing to run us by an eavesdropper or take us to the wrong server?
  - But: they can't read; we detect impersonation
    - No problem!
- Attacker slips in as a Man In The Middle?
  - But: they can't read, they can't inject
  - They can't even replay previous encrypted traffic
  - **No problem!**

Validating Amazon's Identity, cont.

- Browser retrieves cert belonging to the **issuer**
  - These are hardwired into the browser – and **trusted!**
- What if browser can't find a cert for the issuer?

This Connection is Untrusted

You have asked Firefox to connect securely to **www.mikestoolbox.org**, but we can't confirm that your connection is secure.

Normally, when you try to connect securely, sites will present trusted identification to prove that you are going to the right place. However, this site's identity can't be verified.

What Should I Do?

If you usually connect to this site without problems, this error could mean that someone is trying to impersonate the site, and you shouldn't continue.

Get me out of here!

▼ Technical Details

www.mikestoolbox.org uses an invalid security certificate.

The certificate is not trusted because the issuer certificate is not trusted.

(Error code: sec_error_untrusted_issuer)

▶ I Understand the Risks

Verify Certificate

Safari can't verify the identity of the website "www.mikestoolbox.org".

The certificate for this website was signed by an unknown certifying authority. You might be connecting to a website that is pretending to be "www.mikestoolbox.org", which could put your confidential information at risk. Would you like to connect to the website anyway?

? Show Certificate | Cancel | Continue

Validating Amazon's Identity, cont.

- Browser retrieves cert belonging to the **issuer**
  - These are hardwired into the browser – and **trusted!**
- What if browser can't find a cert for the issuer?
- If it can't find the cert, then warns the user that site has not been verified
  - Can still proceed, just without authentication
- Q: Which end-to-end security properties do we lose if we incorrectly trust that the site is whom we think?
- A: **All of them!**
  - Goodbye confidentiality, integrity, authentication
  - Man in the middle attacker can read everything, modify, impersonate

SSL / TLS Limitations

- Properly used, SSL / TLS provides powerful end-to-end protections
- So why not use it for *everything*??
- Issues:
  - Cost of public-key crypto (fairly minor)
    - Takes non-trivial CPU processing (but today a minor issue)
    - Note: *symmetric* key crypto on modern hardware is non-issue
  - Hassle of buying/maintaining certs (fairly minor)

SSL / TLS Limitations

- Properly used, SSL / TLS provides powerful end-to-end protections
- So why not use it for *everything*??
- Issues:
  - Cost of public-key crypto (fairly minor)
    - Takes non-trivial CPU processing (but today a minor issue)
    - Note: *symmetric* key crypto on modern hardware is non-issue
  - Hassle of buying/maintaining certs (fairly minor)
  - Integrating with other sites that don't use HTTPS
  - **Latency**: extra round trips ⇒ 1st page slower to load

SSL / TLS Limitations, cont.

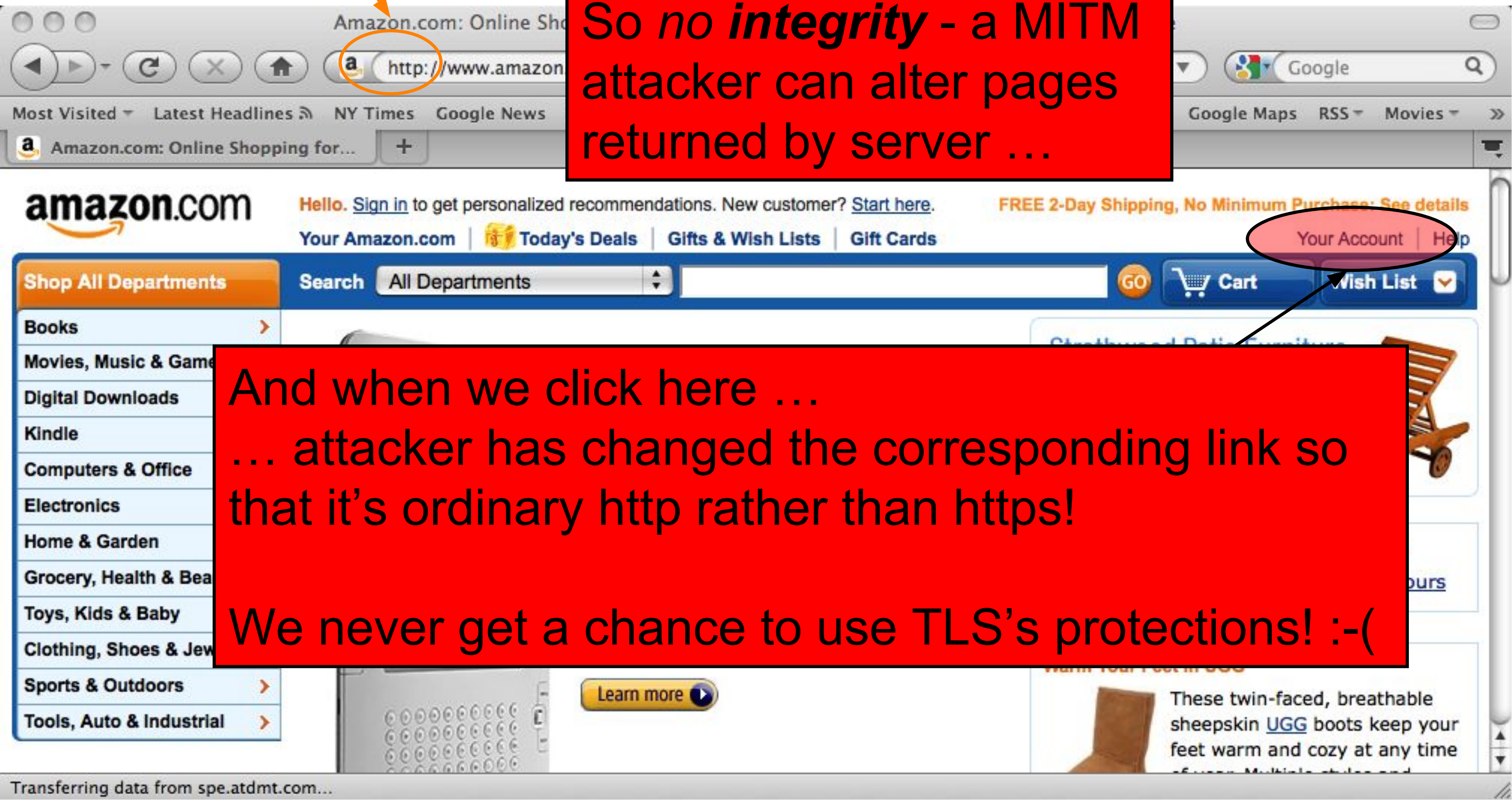
- Problems that SSL / TLS does **not** take care of ?
- TCP-level denial of service
  - SYN flooding
  - RST injection
    - (but does protect against data injection!)
- SQL injection / XSS / server-side coding/logic flaws
- Vulnerabilities introduced by server inconsistencies

SSL / TLS Limitations, cont.

- Problems that SSL / TLS does **not** take care of ?

- SQL injection / XSS / server-side coding/logic flaws
- Vulnerabilities introduced by server inconsistencies

Regular web surfing: http: URL



So *no* ***integrity*** - a MITM attacker can alter pages returned by server …

And when we click here …
… attacker has changed the corresponding link so that it’s ordinary http rather than https!

We never get a chance to use TLS’s protections! :-(

“sslstrip” attack

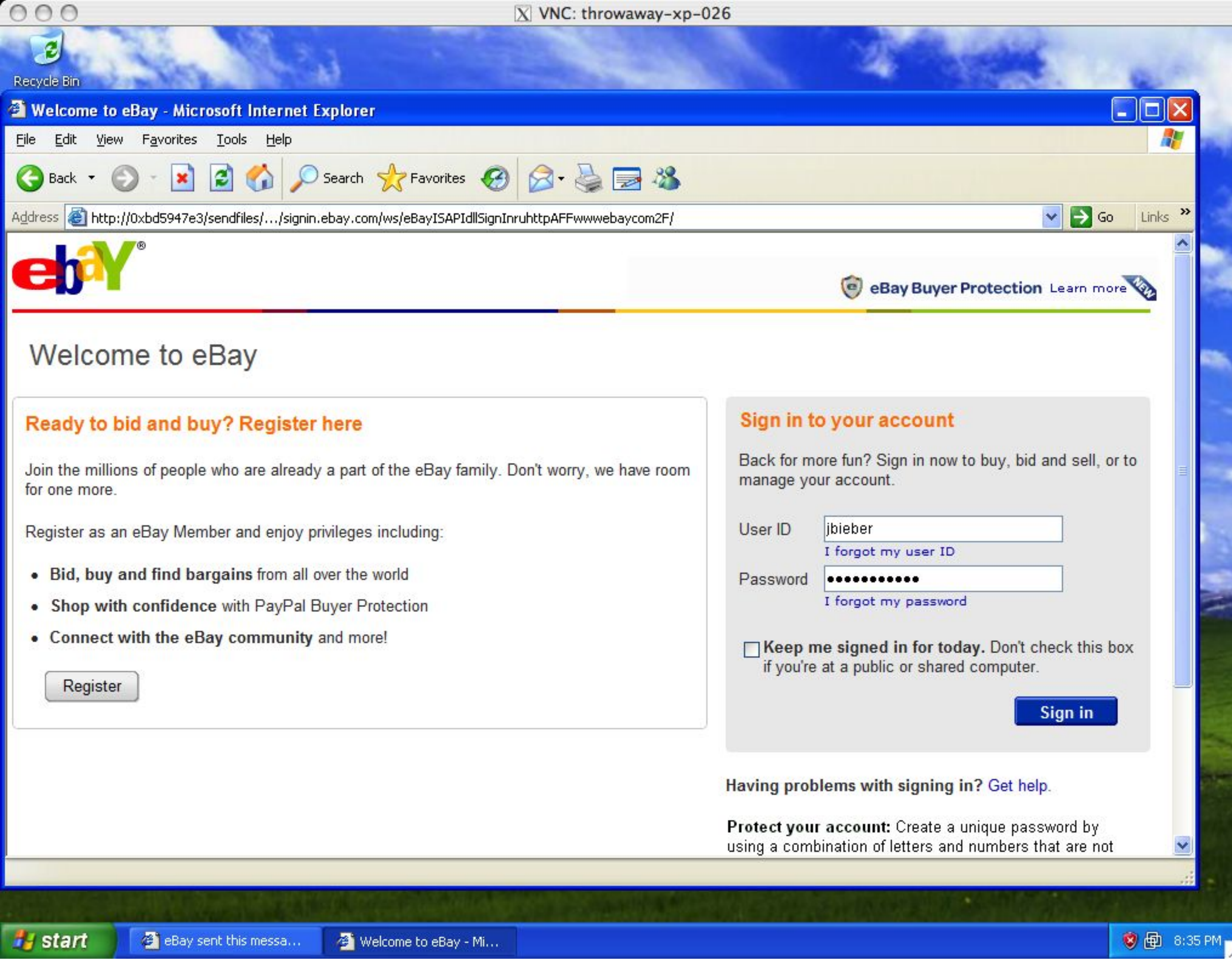
SSL / TLS Limitations, cont.

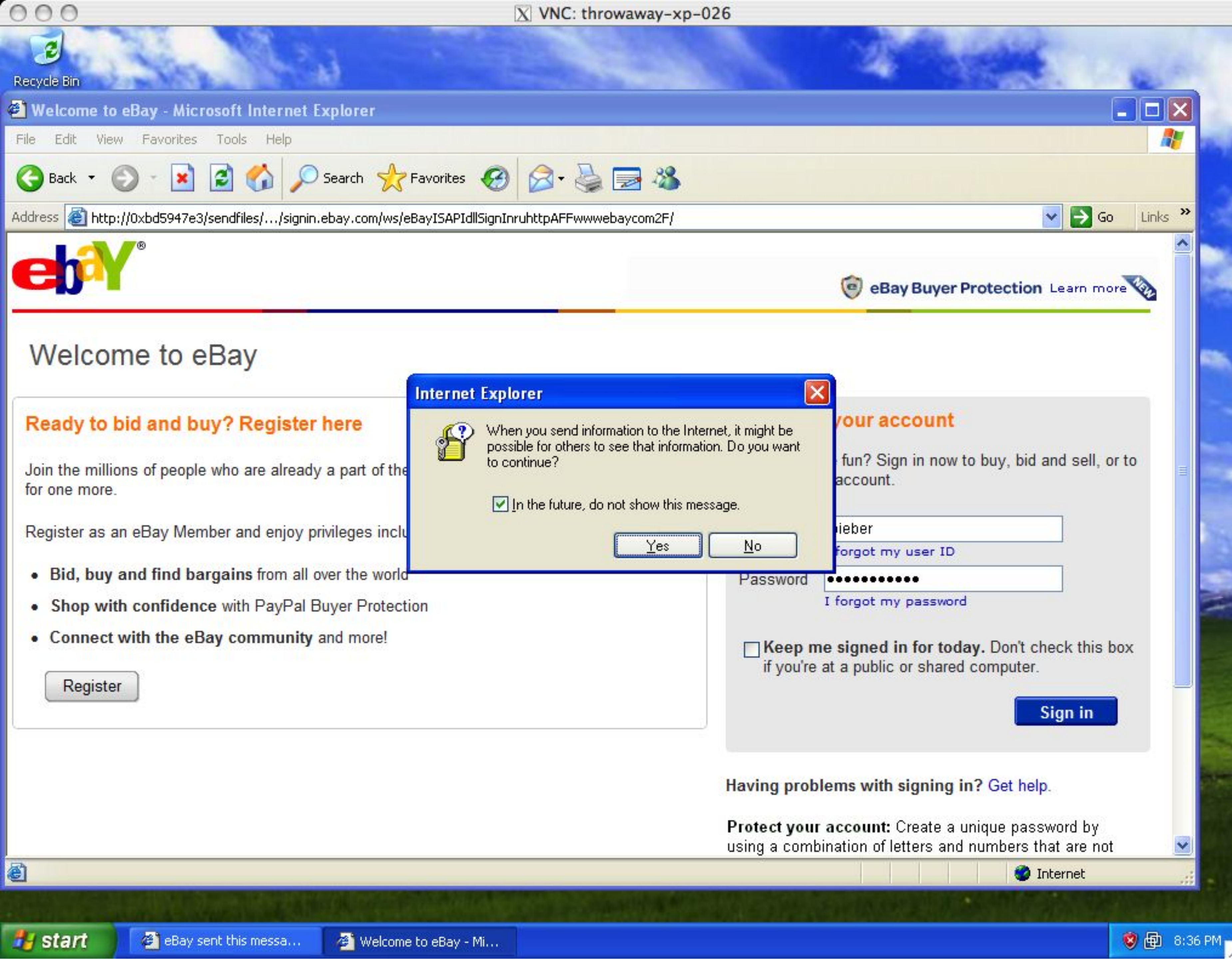
- Problems that SSL / TLS does **not** take care of ?

- SQL injection / XSS / server-side coding/logic flaws
- Vulnerabilities introduced by server inconsistencies
- Browser coding/logic flaws
- User flaws
  - Weak passwords
  - Phishing
- Issues of trust …

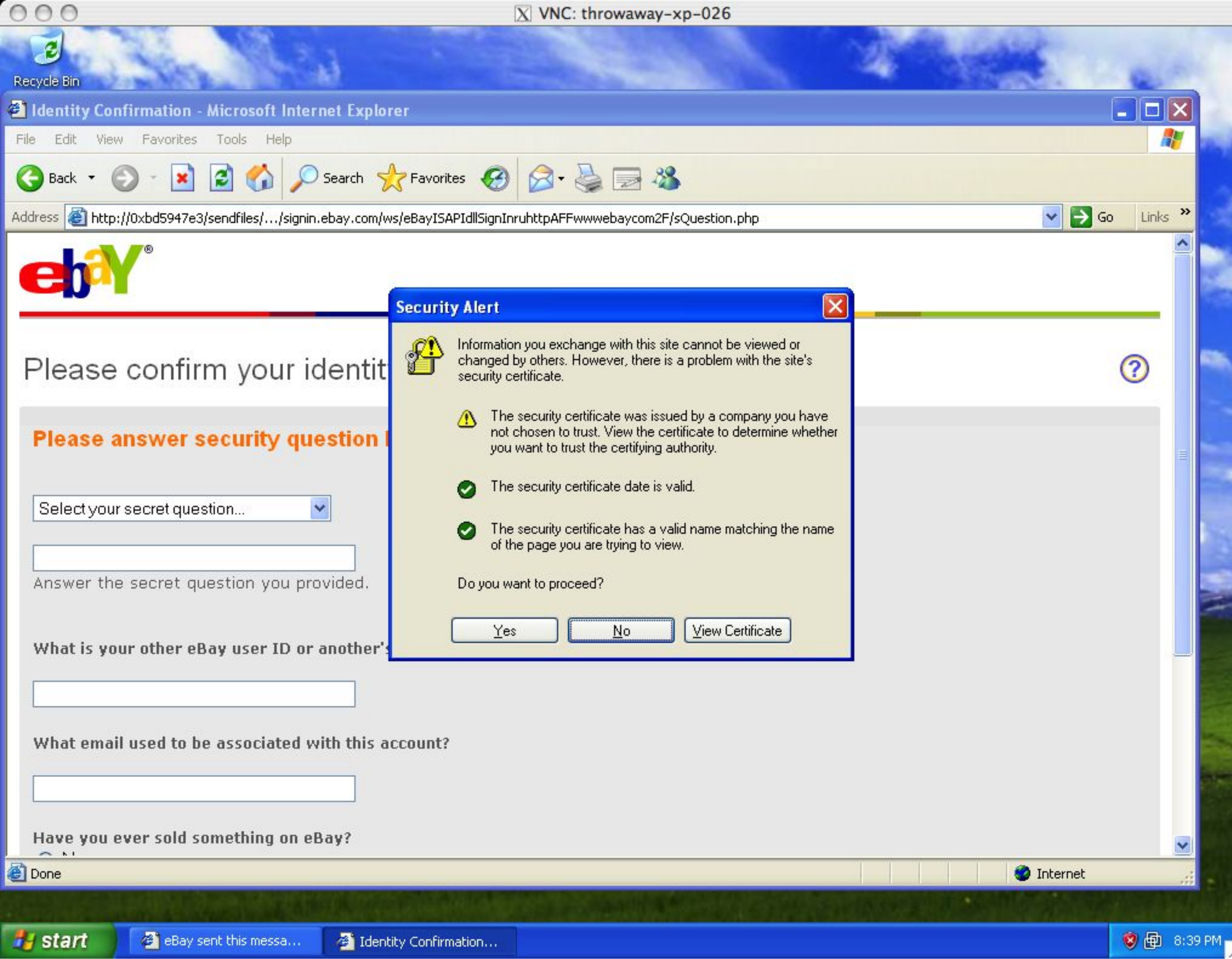
TLS/SSL Trust Issues

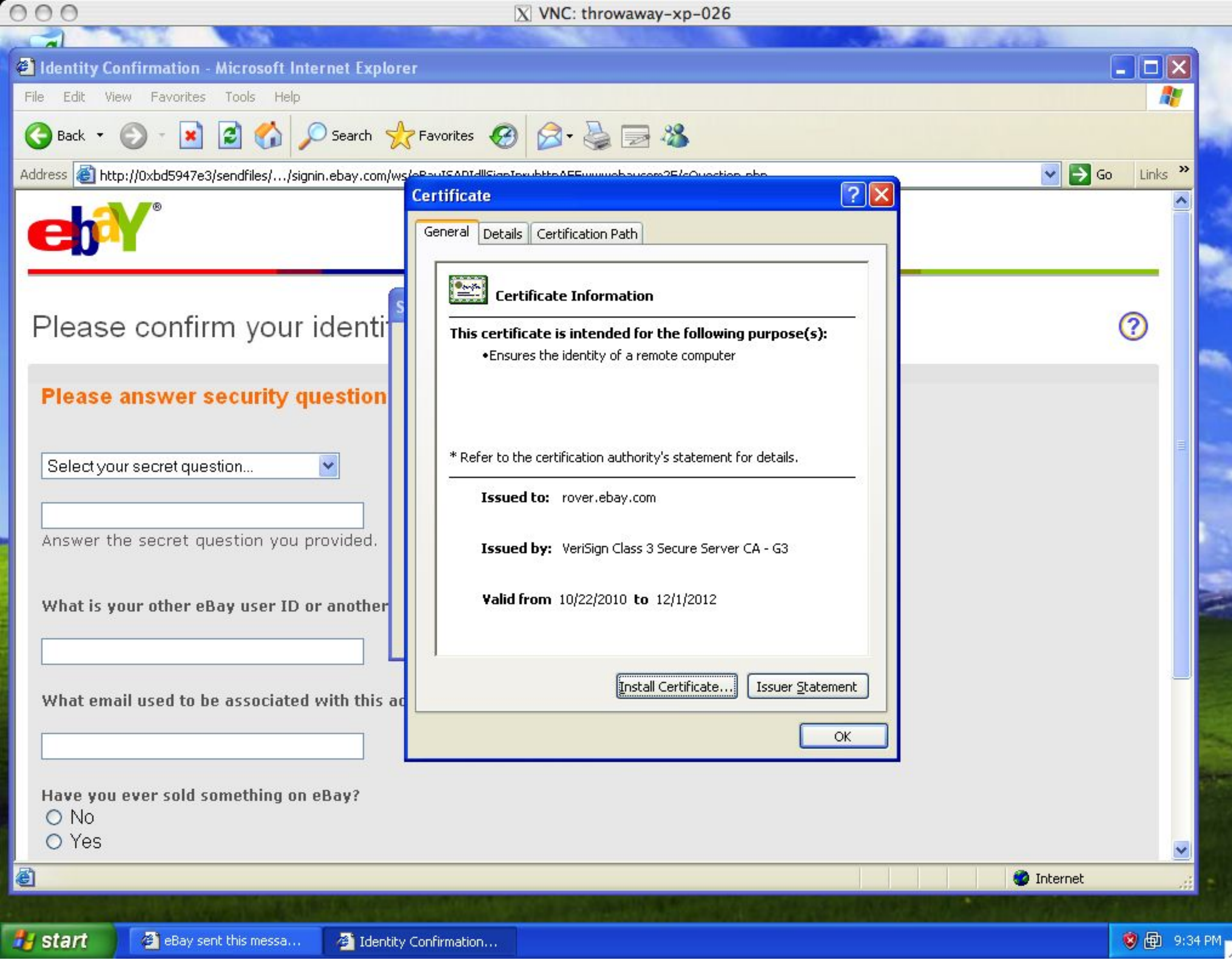
- User has to make correct trust decisions …

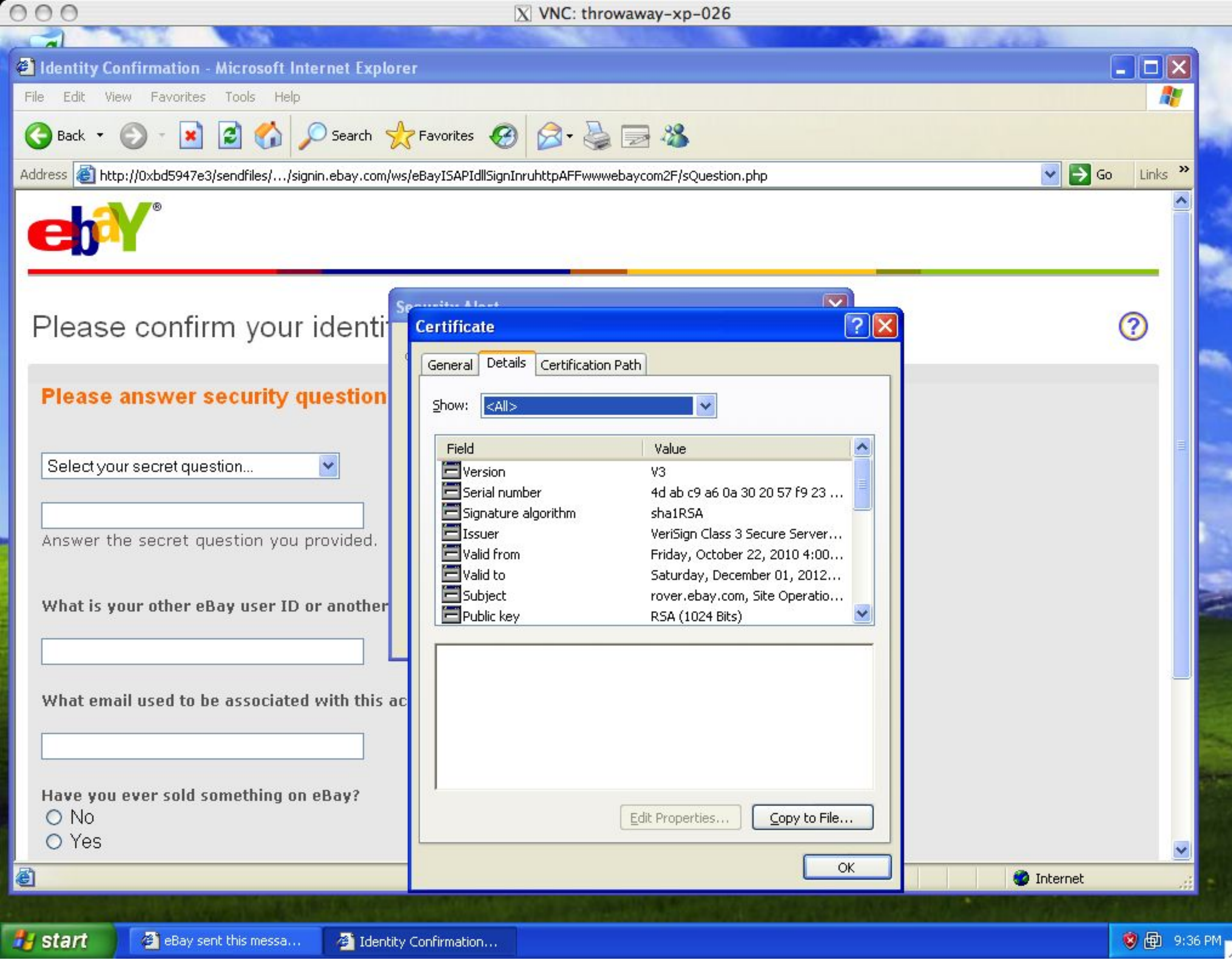

VNC: throwaway-xp-026
Recycle Bin
Welcome to eBay - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back
Search
Favorites
Address
http://0xbd5947e3/sendfiles/…/signin.ebay.com/ws/eBayISAPIdllSignInruhttpAFFwwwebaycom2F/
Go
Links
eBay Buyer Protection
Learn more
NEW
Welcome to eBay
Ready to bid and buy? Register here
Join the millions of people who are already a part of the eBay family. Don't worry, we have room for one more.
Register as an eBay Member and enjoy privileges including:
• Bid, buy and find bargains from all over the world
• Shop with confidence with PayPal Buyer Protection
• Connect with the eBay community and more!
Register
Sign in to your account
Back for more fun? Sign in now to buy, bid and sell, or to manage your account.
User ID
jbieber
I forgot my user ID
Password
I forgot my password
Keep me signed in for today. Don't check this box if you're at a public or shared computer.
Sign in
Having problems with signing in? Get help.
Protect your account: Create a unique password by using a combination of letters and numbers that are not
start
eBay sent this messa…
Welcome to eBay - Mi…
8:35 PM

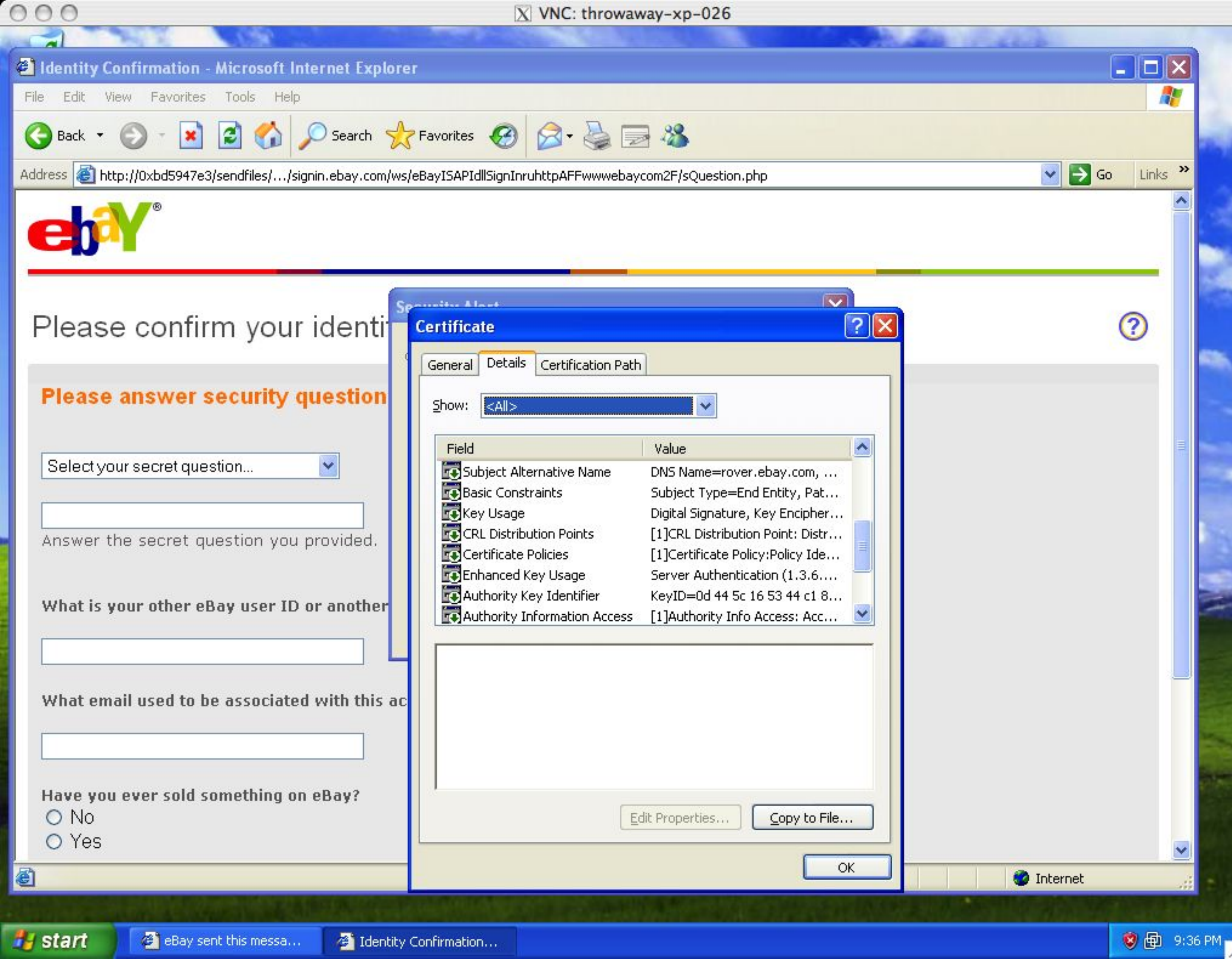


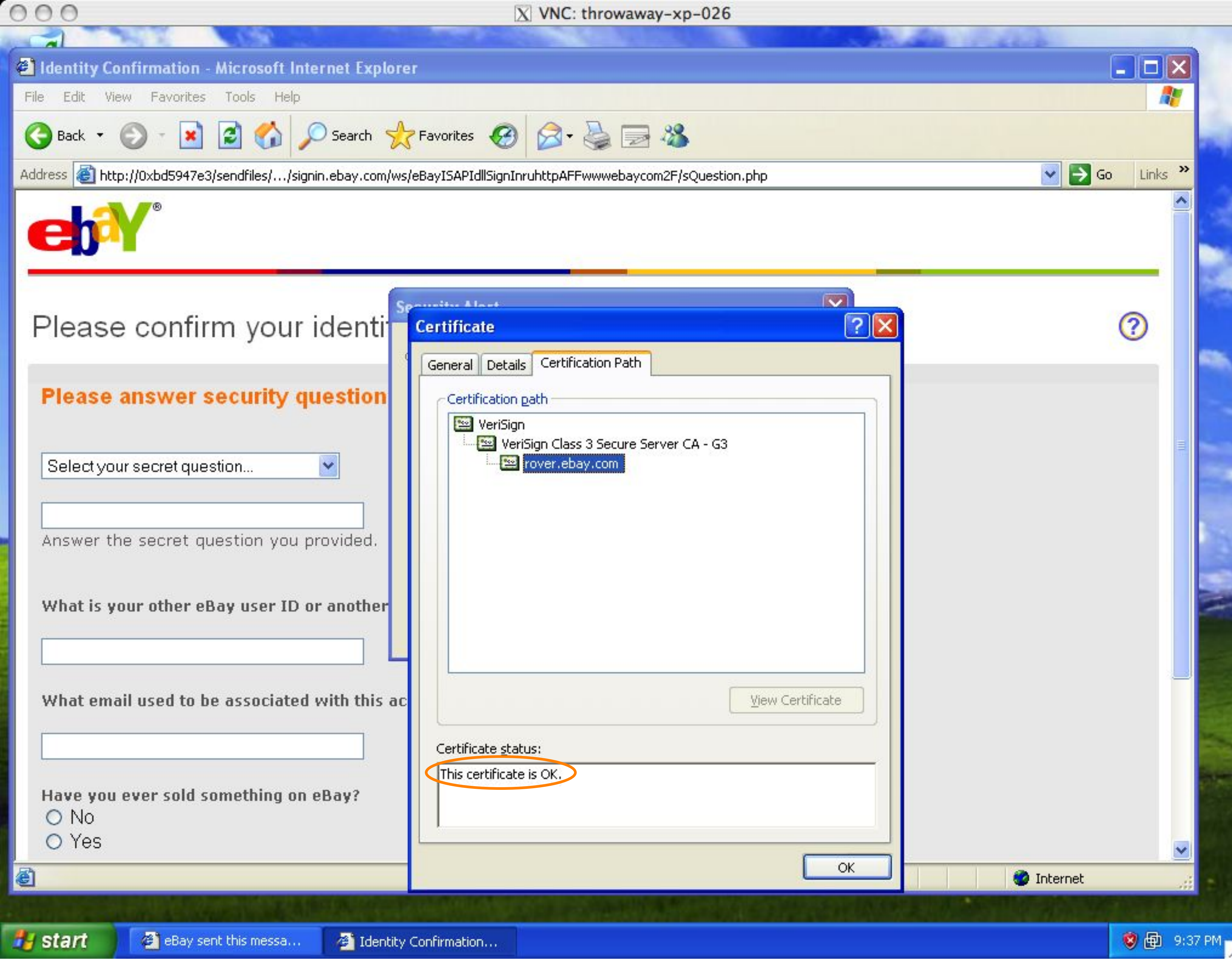
VNC: throwaway-xp-026
Recycle Bin
Welcome to eBay - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back Search Favorites
Address http://0xbd5947e3/sendfiles/…/signin.ebay.com/ws/eBayISAPIdllSignInruhttpAFFwwwebaycom2F/
Go Links
eBay Buyer Protection Learn more NEW
Welcome to eBay
Ready to bid and buy? Register here
Join the millions of people who are already a part of the
for one more.
Register as an eBay Member and enjoy privileges inclu
Bid, buy and find bargains from all over the world
Shop with confidence with PayPal Buyer Protection
Connect with the eBay community and more!
Register
Internet Explorer
When you send information to the Internet, it might be possible for others to see that information. Do you want to continue?
In the future, do not show this message.
Yes
No
your account
fun? Sign in now to buy, bid and sell, or to
account.
ieber
forgot my user ID
Password
I forgot my password
Keep me signed in for today. Don't check this box if you're at a public or shared computer.
Sign in
Having problems with signing in? Get help.
Protect your account: Create a unique password by using a combination of letters and numbers that are not
Internet
start
eBay sent this messa…
Welcome to eBay - Mi…
8:36 PM


VNC: throwaway-xp-026
Recycle Bin
Identity Confirmation - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back Search Favorites
Address http://0xbd5947e3/sendfiles/…/signin.ebay.com/ws/eBayISAPIdllSignInruhttpAFFwwwebaycom2F/sQuestion.php Go Links
Please confirm your identit
Please answer security question
Select your secret question...
Answer the secret question you provided.
What is your other eBay user ID or another's
What email used to be associated with this account?
Have you ever sold something on eBay?
Security Alert
Information you exchange with this site cannot be viewed or changed by others. However, there is a problem with the site's security certificate.
The security certificate was issued by a company you have not chosen to trust. View the certificate to determine whether you want to trust the certifying authority.
The security certificate date is valid.
The security certificate has a valid name matching the name of the page you are trying to view.
Do you want to proceed?
Yes
No
View Certificate
Done
Internet
start
eBay sent this messa…
Identity Confirmation…
8:39 PM

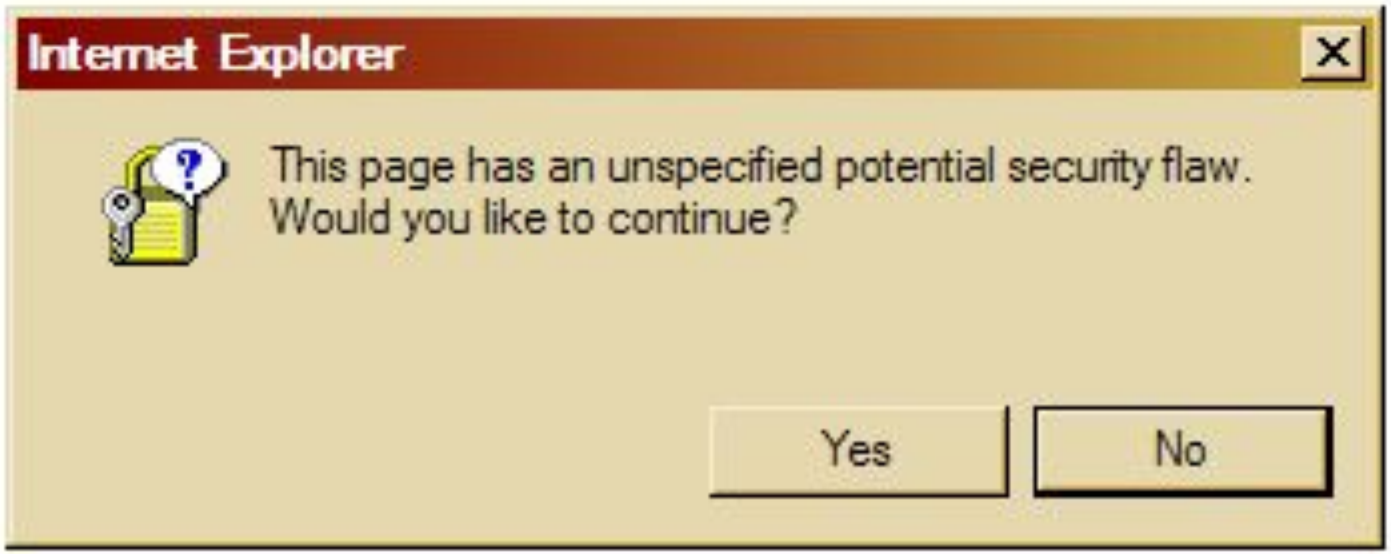

VNC: throwaway-xp-026
Identity Confirmation - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back Search Favorites
Address http://0xbd5947e3/sendfiles/.../signin.ebay.com/ws
Go Links
Please confirm your identi
Please answer security question
Select your secret question...
Answer the secret question you provided.
What is your other eBay user ID or another
What email used to be associated with this ac
Have you ever sold something on eBay?
No
Yes
Certificate
General
Details
Certification Path
Certificate Information
This certificate is intended for the following purpose(s):
•Ensures the identity of a remote computer
* Refer to the certification authority's statement for details.
Issued to: rover.ebay.com
Issued by: VeriSign Class 3 Secure Server CA - G3
Valid from 10/22/2010 to 12/1/2012
Install Certificate...
Issuer Statement
OK
Internet
start
eBay sent this messa...
Identity Confirmation...
9:34 PM


VNC: throwaway-xp-026
Identity Confirmation - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back Search Favorites
Address http://0xbd5947e3/sendfiles/.../signin.ebay.com/ws/eBayISAPIdllSignInruhttpAFFwwwebaycom2F/sQuestion.php
Go Links
Please confirm your identi
Please answer security question
Select your secret question...
Answer the secret question you provided.
What is your other eBay user ID or another
What email used to be associated with this ac
Have you ever sold something on eBay?
No
Yes
Certificate
General Details Certification Path
Show: <All>
Field Value
Version V3
Serial number 4d ab c9 a6 0a 30 20 57 f9 23 ...
Signature algorithm sha1RSA
Issuer VeriSign Class 3 Secure Server...
Valid from Friday, October 22, 2010 4:00...
Valid to Saturday, December 01, 2012...
Subject rover.ebay.com, Site Operatio...
Public key RSA (1024 Bits)
Edit Properties...
Copy to File...
OK
Internet
start
eBay sent this messa...
Identity Confirmation...
9:36 PM


VNC: throwaway-xp-026
Identity Confirmation - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back
Search
Favorites
Address
http://0xbd5947e3/sendfiles/.../signin.ebay.com/ws/eBayISAPIdllSignInruhttpAFFwwwebaycom2F/sQuestion.php
Go
Links
ebay
Please confirm your identi
Please answer security question
Select your secret question...
Answer the secret question you provided.
What is your other eBay user ID or another
What email used to be associated with this ac
Have you ever sold something on eBay?
No
Yes
Security Alert
Certificate
General
Details
Certification Path
Show:
<All>
Field
Value
Subject Alternative Name
DNS Name=rover.ebay.com, ...
Basic Constraints
Subject Type=End Entity, Pat...
Key Usage
Digital Signature, Key Encipher...
CRL Distribution Points
[1]CRL Distribution Point: Distr...
Certificate Policies
[1]Certificate Policy:Policy Ide...
Enhanced Key Usage
Server Authentication (1.3.6....
Authority Key Identifier
KeyID=0d 44 5c 16 53 44 c1 8...
Authority Information Access
[1]Authority Info Access: Acc...
Edit Properties...
Copy to File...
OK
Internet
start
eBay sent this messa...
Identity Confirmation...
9:36 PM


VNC: throwaway-xp-026
Identity Confirmation - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back Search Favorites
Address http://0xbd5947e3/sendfiles/…/signin.ebay.com/ws/eBayISAPIdllSignInruhttpAFFwwwebaycom2F/sQuestion.php
Go Links
Please confirm your identi
Please answer security question
Select your secret question...
Answer the secret question you provided.
What is your other eBay user ID or another
What email used to be associated with this ac
Have you ever sold something on eBay?
No
Yes
Certificate
General Details Certification Path
Certification path
VeriSign
VeriSign Class 3 Secure Server CA - G3
rover.ebay.com
View Certificate
Certificate status:
This certificate is OK.
OK
Internet
start
eBay sent this messa...
Identity Confirmation...
9:37 PM

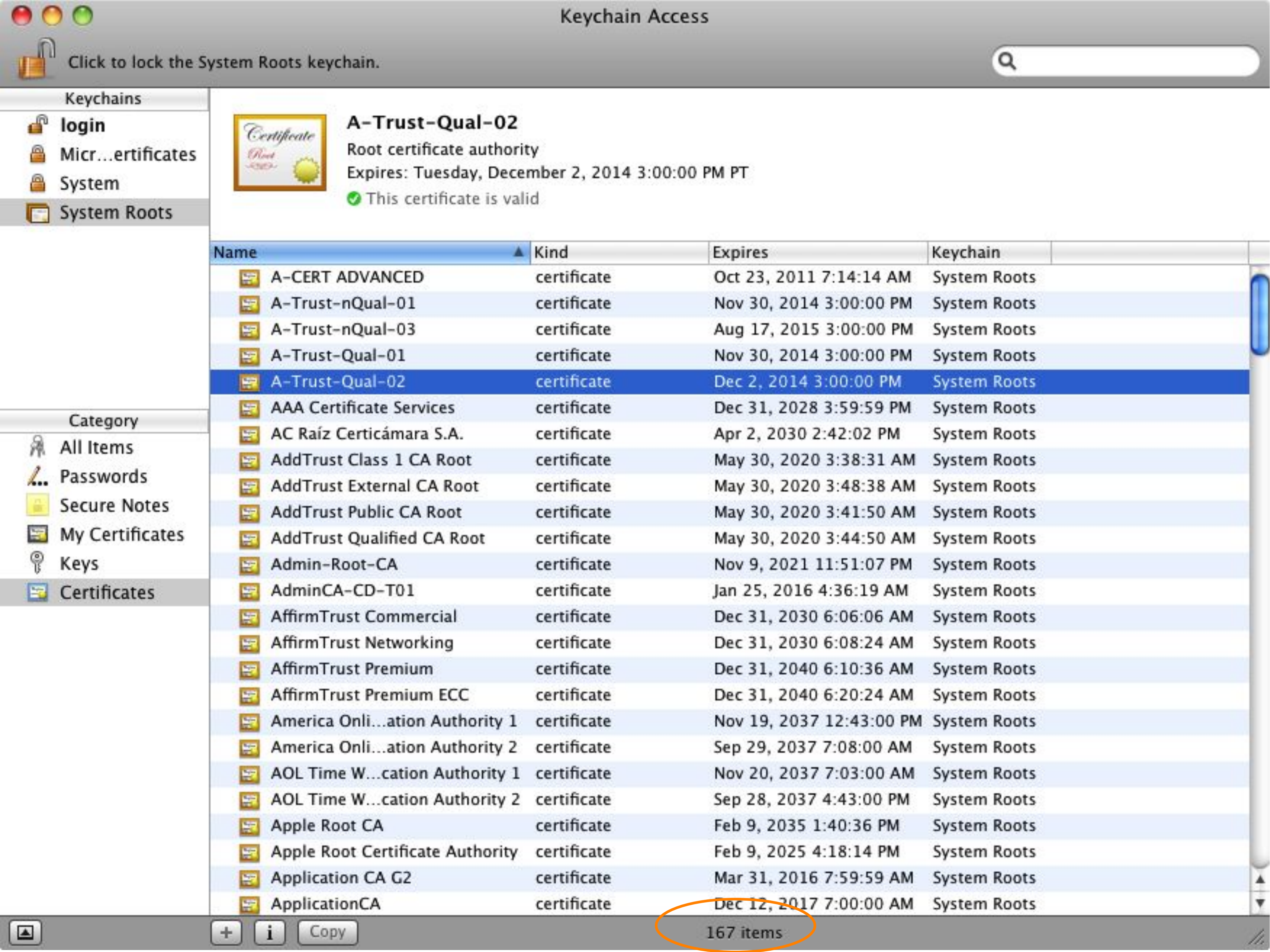
The equivalent as seen by most Internet users:



(note: an actual Windows error message!)

TLS/SSL Trust Issues, cont.

- "*Commercial certificate authorities protect you from anyone from whom they are unwilling to take money.*"
  - Matt Blaze, circa 2001
- So how many CAs do we have to worry about, anyway?


Keychain Access
Click to lock the System Roots keychain.
Keychains
login
Micr…ertificates
System
System Roots
Category
All Items
Passwords
Secure Notes
My Certificates
Keys
Certificates
A-Trust-Qual-02
Root certificate authority
Expires: Tuesday, December 2, 2014 3:00:00 PM PT
This certificate is valid
Name
Kind
Expires
Keychain
A-CERT ADVANCED certificate Oct 23, 2011 7:14:14 AM System Roots
A-Trust-nQual-01 certificate Nov 30, 2014 3:00:00 PM System Roots
A-Trust-nQual-03 certificate Aug 17, 2015 3:00:00 PM System Roots
A-Trust-Qual-01 certificate Nov 30, 2014 3:00:00 PM System Roots
A-Trust-Qual-02 certificate Dec 2, 2014 3:00:00 PM System Roots
AAA Certificate Services certificate Dec 31, 2028 3:59:59 PM System Roots
AC Raíz Certicámara S.A. certificate Apr 2, 2030 2:42:02 PM System Roots
AddTrust Class 1 CA Root certificate May 30, 2020 3:38:31 AM System Roots
AddTrust External CA Root certificate May 30, 2020 3:48:38 AM System Roots
AddTrust Public CA Root certificate May 30, 2020 3:41:50 AM System Roots
AddTrust Qualified CA Root certificate May 30, 2020 3:44:50 AM System Roots
Admin-Root-CA certificate Nov 9, 2021 11:51:07 PM System Roots
AdminCA-CD-T01 certificate Jan 25, 2016 4:36:19 AM System Roots
AffirmTrust Commercial certificate Dec 31, 2030 6:06:06 AM System Roots
AffirmTrust Networking certificate Dec 31, 2030 6:08:24 AM System Roots
AffirmTrust Premium certificate Dec 31, 2040 6:10:36 AM System Roots
AffirmTrust Premium ECC certificate Dec 31, 2040 6:20:24 AM System Roots
America Onli…ation Authority 1 certificate Nov 19, 2037 12:43:00 PM System Roots
America Onli…ation Authority 2 certificate Sep 29, 2037 7:08:00 AM System Roots
AOL Time W…cation Authority 1 certificate Nov 20, 2037 7:03:00 AM System Roots
AOL Time W…cation Authority 2 certificate Sep 28, 2037 4:43:00 PM System Roots
Apple Root CA certificate Feb 9, 2035 1:40:36 PM System Roots
Apple Root Certificate Authority certificate Feb 9, 2025 4:18:14 PM System Roots
Application CA G2 certificate Mar 31, 2016 7:59:59 AM System Roots
ApplicationCA certificate Dec 12, 2017 7:00:00 AM System Roots
Copy
167 items

TLS/SSL Trust Issues

- *"Commercial certificate authorities protect you from anyone from whom they are unwilling to take money."*
  - Matt Blaze, circa 2001
- So how many CAs do we have to worry about, anyway?
- Of course, it's not just their greed that matters …

News

Solo Iranian hacker takes credit for Comodo certificate attack

Security researchers split on whether 'ComodoHacker' is the real deal

By Gregg Keizer

March 27, 2011 08:39 PM ET

Comments (5) Recommended (37) Like 84

Computerworld - A solo Iranian hacker on Saturday claimed responsibility for stealing multiple SSL certificates belonging to some of the Web's biggest sites, including Google, Microsoft, Skype and Yahoo.

Early reaction from security experts was mixed, with some believing the hacker's claim, while others were dubious.

Last week, conjecture had focused on a state-sponsored attack, perhaps funded or conducted by the Iranian government, that hacked a certificate reseller affiliated with U.S.-based Comodo.

On March 23, Comodo acknowledged the attack, saying that eight days earlier, hackers had obtained nine bogus certificates for the log-on sites of Microsoft's Hotmail, Google's Gmail, the Internet phone and chat service Skype and Yahoo Mail. A certificate for Mozilla's Firefox add-on site was also acquired.

News

Solo Iranian hacker takes credit for Comodo certificate attack

Security researchers split on whether 'ComodoHacker' is the real deal

By Gregg Keizer

March 27, 2011 08:39 PM ET

Comments (5) Recommended (37) Like 84

Where did you learn about cryptography and hacking. Are there books in Persian? English books? Or are you self-taught, learning from the Internet?

> d) I'm self taught, books in Persian and English, but mostly papers in internet, short papers from experts like Bruce Schneier, RSA people (Ron, Adi and Leonard) and specially David Wagner. I learned programming in Qbasic when I was 9, I started learning cryptography when I was 13

funded or conducted by the Iranian government, that hacked a certificate reseller affiliated with U.S.-based Comodo.

On March 23, Comodo acknowledged the attack, saying that eight days earlier, hackers had obtained nine bogus certificates for the log-on sites of Microsoft's Hotmail, Google's Gmail, the Internet phone and chat service Skype and Yahoo Mail. A certificate for Mozilla's Firefox add-on site was also acquired.

CNET › News › InSecurity Complex › Fraudulent Google certificate points to Internet attack

Fraudulent Google certificate points to Internet attack

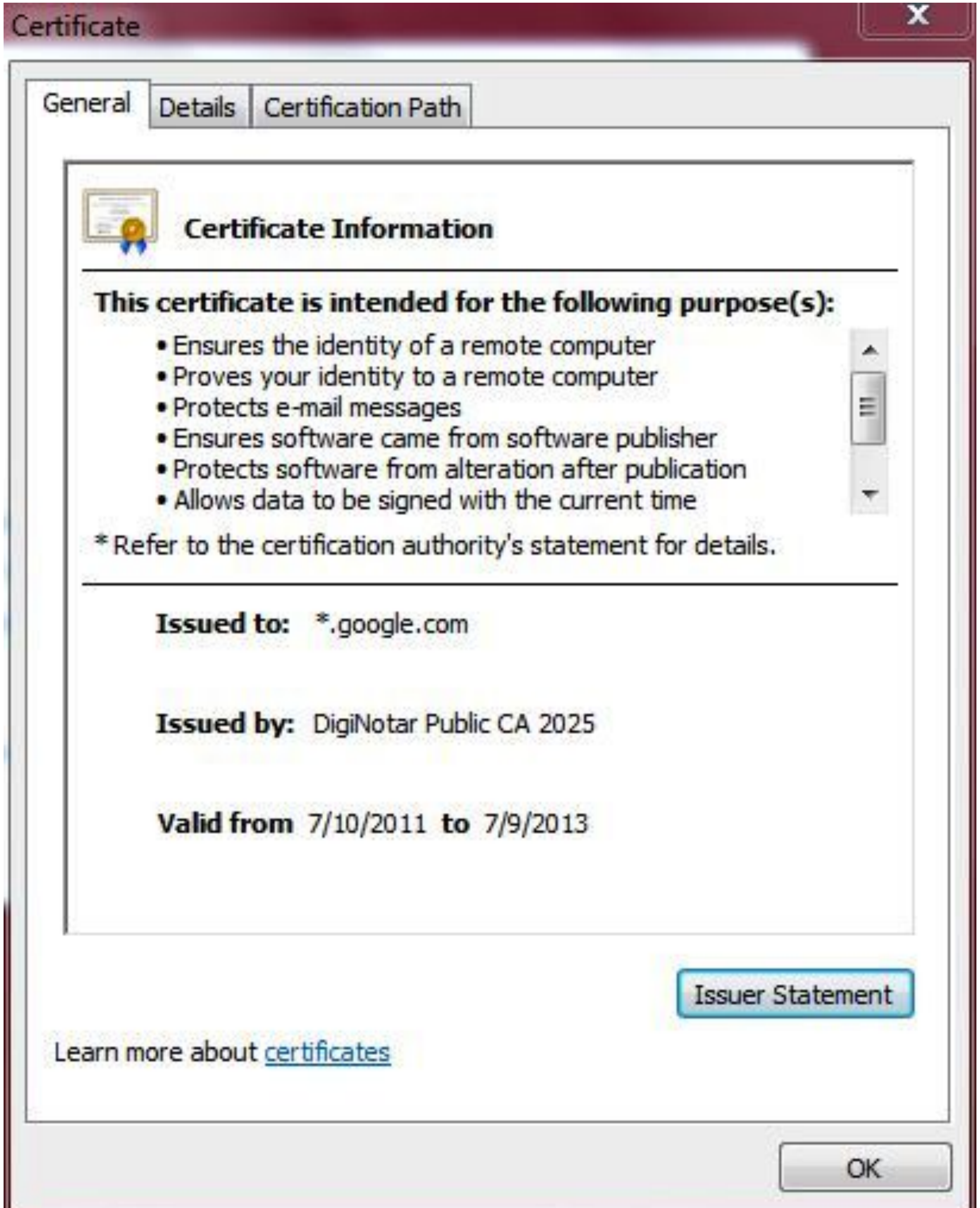
Is Iran behind a fraudulent Google.com digital certificate? The situation is similar to one that happened in March in which spoofed certificates were traced back to Iran.

by Elinor Mills | August 29, 2011 1:22 PM PDT

Follow

A Dutch company appears to have issued a digital certificate for Google.com to someone other than Google, who may be using it to try to re-direct traffic of users based in Iran.

Yesterday, someone reported on a Google support site that when attempting to log in to Gmail the browser issued a warning for the digital certificate used as proof that the site is legitimate, according to this thread on a Google support forum site.



This appears to be a fully valid cert using normal browser validation rules.

Only detected by Chrome due to its recent introduction of cert "pinning" – requiring that certs for certain domains **must** be signed by specific CAs rather than any generally trusted CA

October 31, 2012, 10:49AM

Final Report on DigiNotar Hack Shows Total Compromise of CA Servers

The attacker who penetrated the Dutch CA DigiNotar last year had complete control of all eight of the company's certificate-issuing servers during the operation and he may also have issued some rogue certificates that have not yet been identified. The final report from a

Evidence Suggests DigiNotar, Who Issued Fraudulent Google Certificate, Was Hacked *Years* Ago

from the *diginot* dept

The big news in the security world, obviously, is the fact that a **fraudulent Google certificate made its way out into the wild**, apparently targeting internet users in Iran. The Dutch company DigiNotar has put out a statement saying that **it discovered a breach** back on July 19th during a security audit, and that fraudulent certificates were generated for "several dozen" websites. The only one known to have gotten out into the wild is the Google one.

TLS/SSL Trust Issues

- *"Commercial certificate authorities protect you from anyone from whom they are unwilling to take money."*
  - Matt Blaze, circa 2001
- So how many CAs do we have to worry about, anyway?
- Of course, it's not just their greed that matters …
- … and it's not just their diligence & security that matters …
  - *"A decade ago, I observed that commercial certificate authorities protect you from anyone from whom they are unwilling to take money. That turns out to be wrong; they don't even do that much."* - Matt Blaze, circa 2010

Conclusion

- Use SSL/TLS to secure communications end-to-end
- Relies on trustworthiness of certificates

Network Security

- intro to networking
- network attacks

CS 161: Computer Security

Prof. Raluca Ada Popa

February 27, 2018

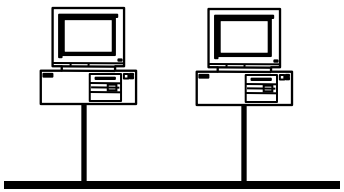
Some slides credit David Wagner.

Networking overview

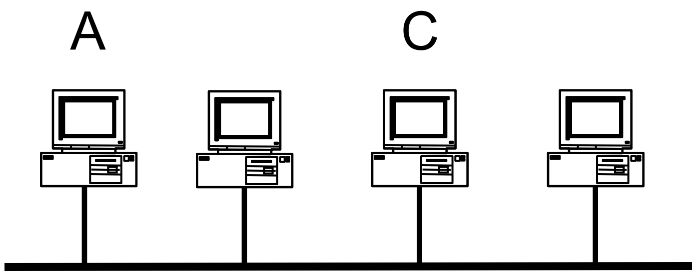
Pay attention to this material (part of 168) because you will need this to understand it for the class

There will be a review session too

Local-Area Networks



point-to-point



shared

How does computer A send a message to computer C?

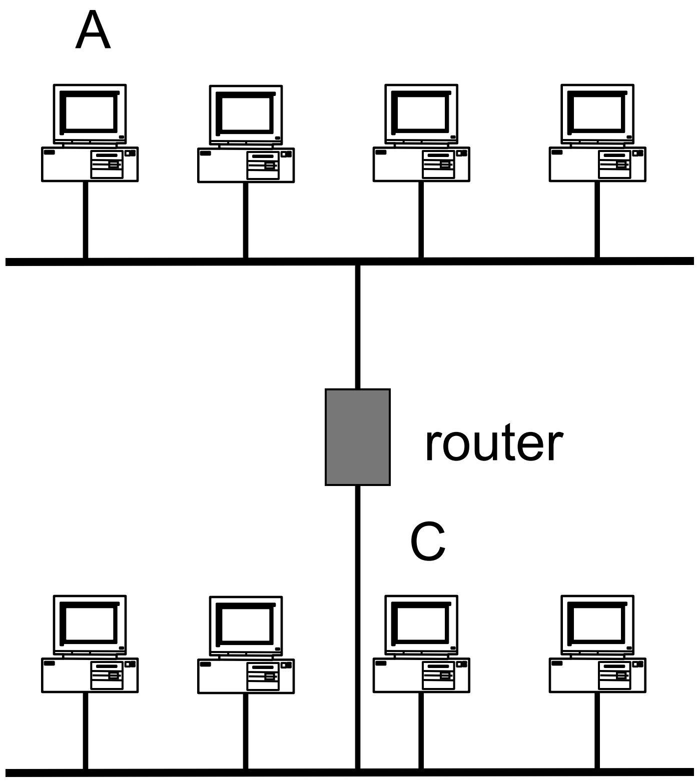
Local-Area Networks (LAN): Packets

```
Source: A
Destination: C
Message: Hello world!
```

A	C	Hello world!

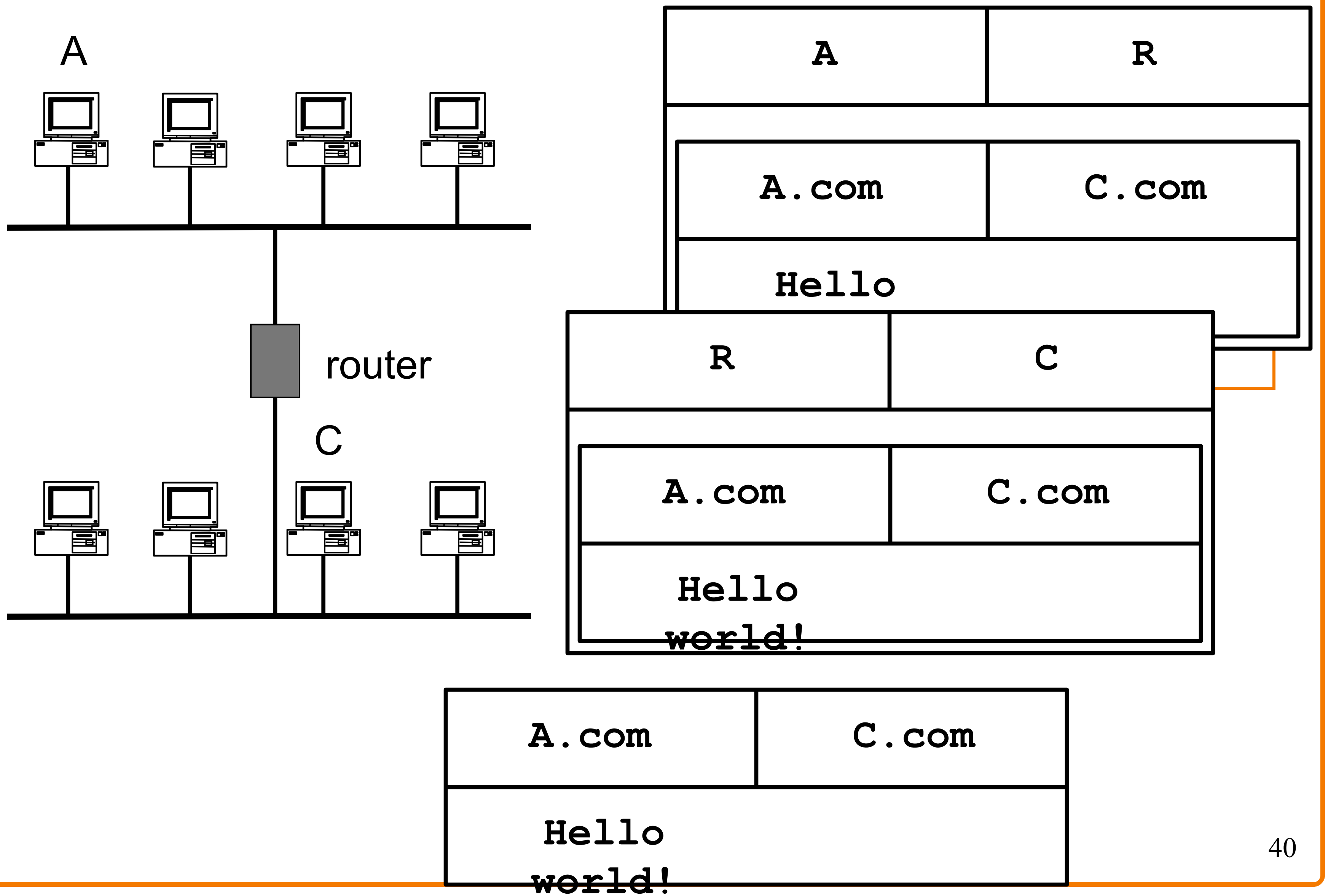
A	C
Hello world!	

Wide-Area Networks



How do we connect two LANs?

Wide-Area Networks



Key Concept #1: *Protocols*

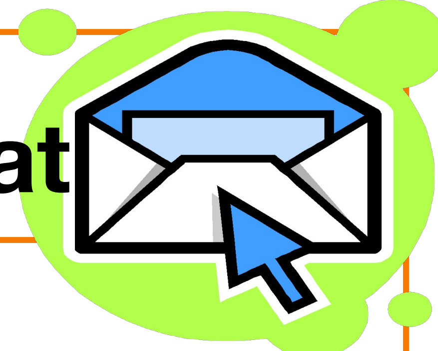
- A protocol is an agreement on how to communicate
- Includes syntax and semantics
  - How a communication is specified & structured
    - Format, order messages are sent and received
  - What a communication means
    - Actions taken when transmitting, receiving, or timer expires
- Example: making a comment in lecture?
  1. Raise your hand.
  2. Wait to be called on.
  3. Or: wait for speaker to **pause** and vocalize
  4. If unrecognized (after timeout): say "excuse me"

Key Concept #2: *Dumb Network*

- Original Internet design: interior nodes ("routers") have no knowledge* of ongoing connections going through them
- **Not** how you picture the telephone system works
  - Which internally tracks all of the active voice calls
- Instead: the postal system!
  - Each Internet message ("packet") self-contained

* Today's Internet is full of hacks that violate this

Self-Contained IP Packet Format



IP = Internet *Protocol*

<table>
<tr><td>4-bit Version</td><td>4-bit Header Length</td><td>8-bit Type of Service (TOS)</td><td colspan="2">16-bit Total Length (Bytes)</td></tr>
<tr><td colspan="3">16-bit Identification</td><td>3-bit Flags</td><td>13-bit Fragment Offset</td></tr>
<tr><td colspan="2">8-bit Time to Live (TTL)</td><td>8-bit Protocol</td><td colspan="2">16-bit Header Checksum</td></tr>
<tr><td colspan="5">32-bit Source IP Address</td></tr>
<tr><td colspan="5">32-bit Destination IP Address</td></tr>
<tr><td colspan="5">Payload (remainder of message)
.
.
.
.
.</td></tr>
</table>

Header **is like a letter envelope: contains all info needed for delivery**

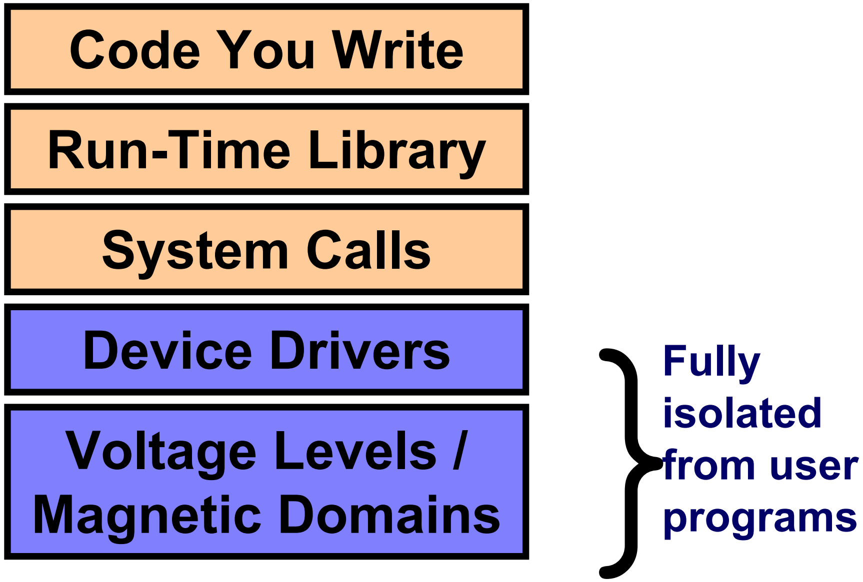
Key Concept #2: *Dumb Network*

- Original Internet design: interior nodes ("routers") have no knowledge* of ongoing connections going through them
- **Not**: how you picture the telephone system works
  - Which internally tracks all of the active voice calls
- Instead: the postal system!
  - Each Internet message ("packet") self-contained
  - Interior routers look at destination address to forward
  - If you want smarts, build it "*end-to-end*", not "hop-by-hop"
  - Buys simplicity & robustness at the cost of shifting complexity into end systems

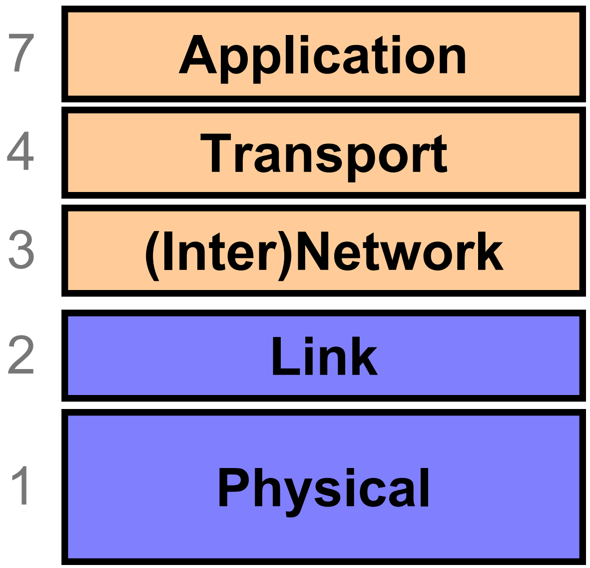
* Today's Internet is full of hacks that violate this

Key Concept #3: *Layering*

- Internet design is strongly partitioned into layers
  - Each layer relies on services provided by next layer below …
  - … and provides services to layer above it
- Analogy:
  - Consider structure of an application you've written and the "services" each layer relies on / provides



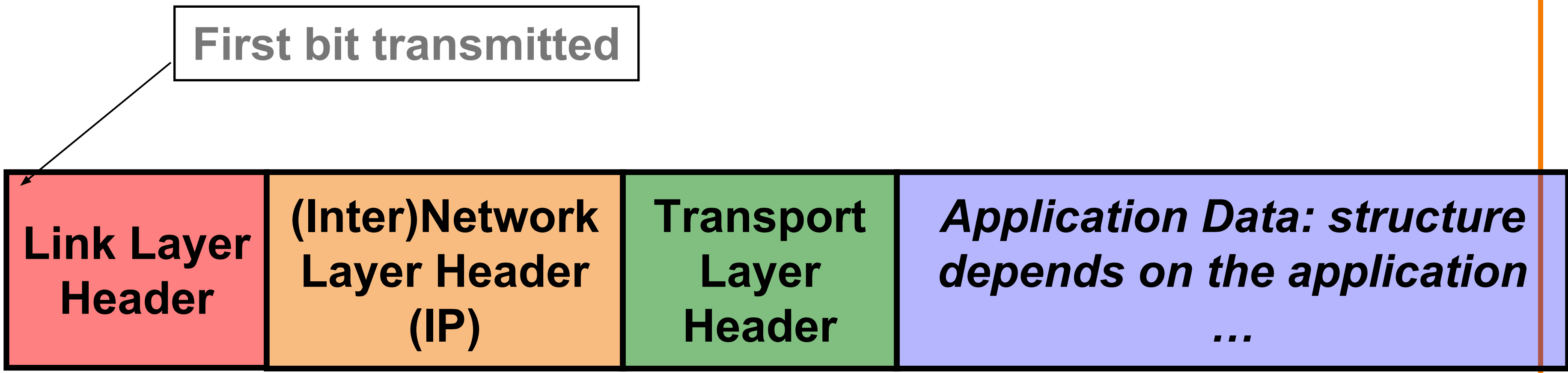
Internet Layering (“Protocol Stack”)



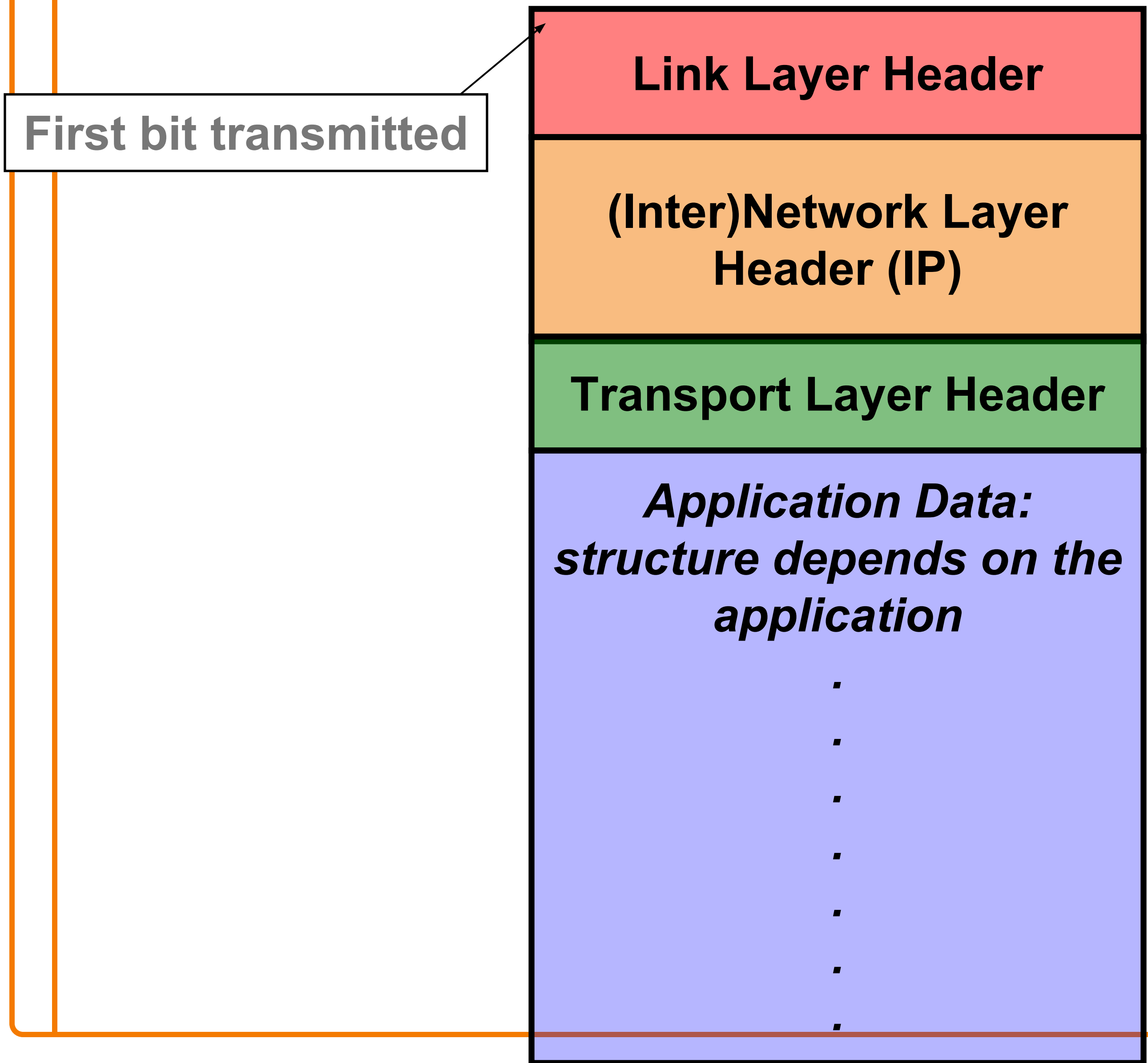
Note on a point of potential confusion: these diagrams are always drawn with lower layers **below** higher layers …

But diagrams showing the layouts of packets are often the *opposite*, with the lower layers at the **top** since their headers <u>precede</u> those for higher layers

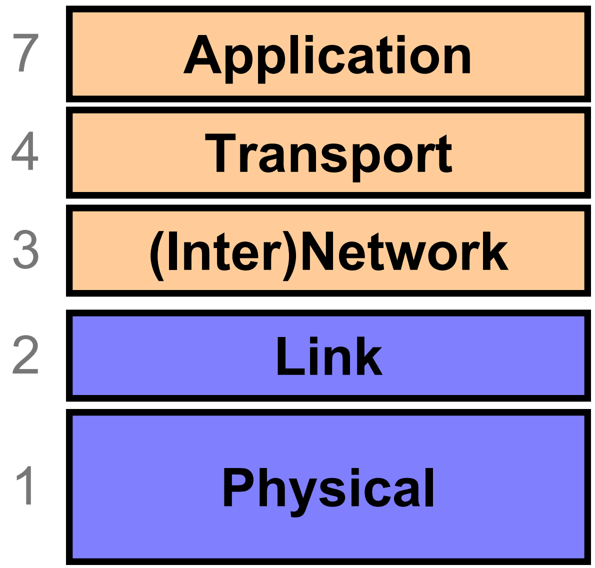
Horizontal View of a Single Packet



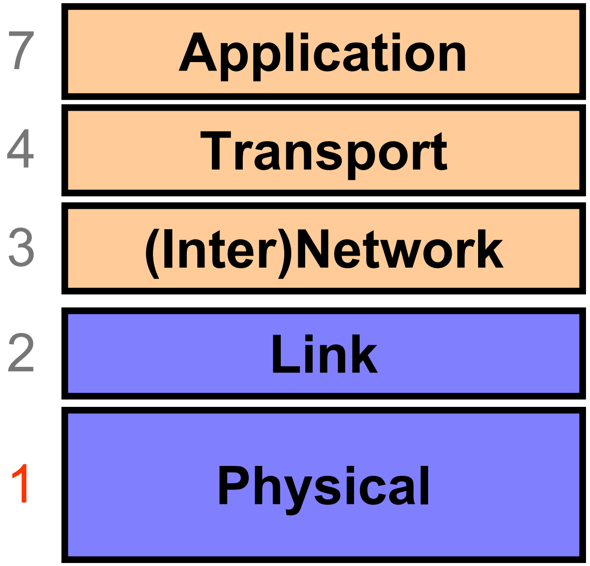
Vertical View of a Single Packet



Internet Layering ("Protocol Stack")



Layer 1: Physical Layer

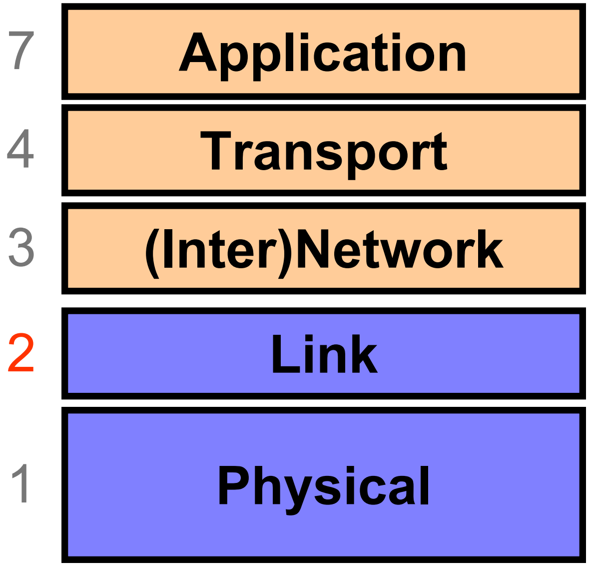


Encoding bits to send them over a single **physical link**

e.g. patterns of

voltage levels / photon intensities / RF modulation

Layer 2: Link Layer



Framing and transmission of a collection of bits into individual messages sent across a single "subnetwork" (one physical technology)

Might involve multiple *physical links* (e.g., modern Ethernet)

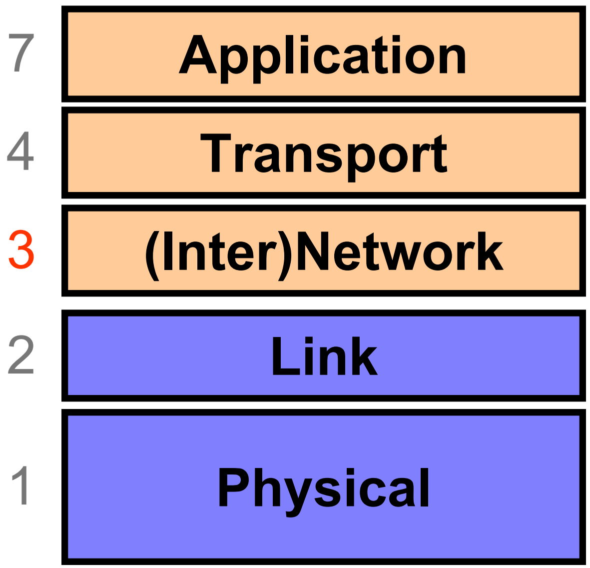
Often technology supports broadcast transmission (**every** "node" connected to subnet receives)

Layer 3: (Inter)Network Layer *(IP)*

Bridges multiple “subnets” to provide *end-to-end* internet connectivity between nodes

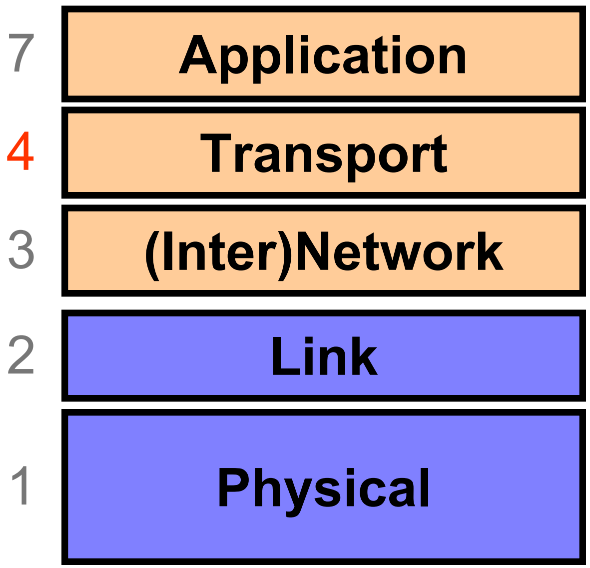
- Provides global addressing

Works across different link technologies



Different **for each Internet “hop”**

Layer 4: Transport Layer



End-to-end communication between processes

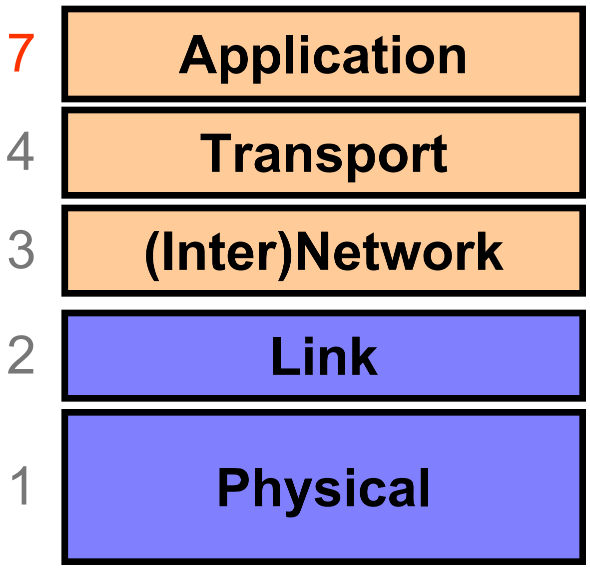
Different services provided:

TCP = reliable *byte stream*

UDP = unreliable *datagrams*

(Datagram = single packet message)

Layer 7: Application Layer



Communication of whatever you wish

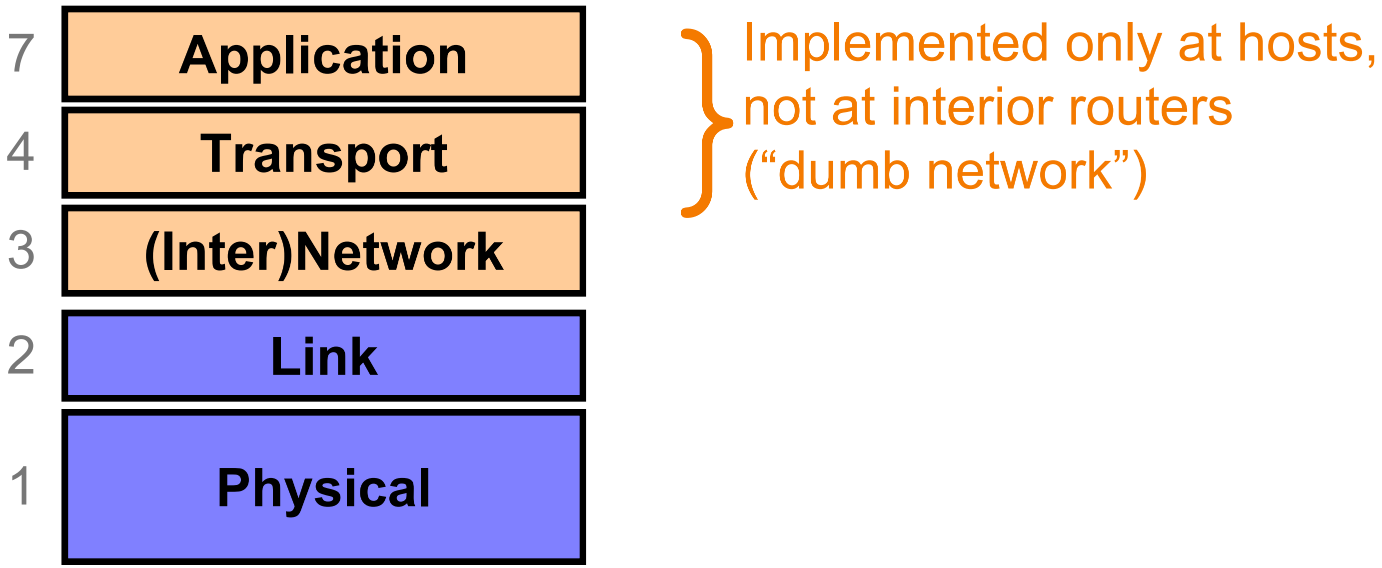
Can use whatever transport(s) is convenient

Freely structured

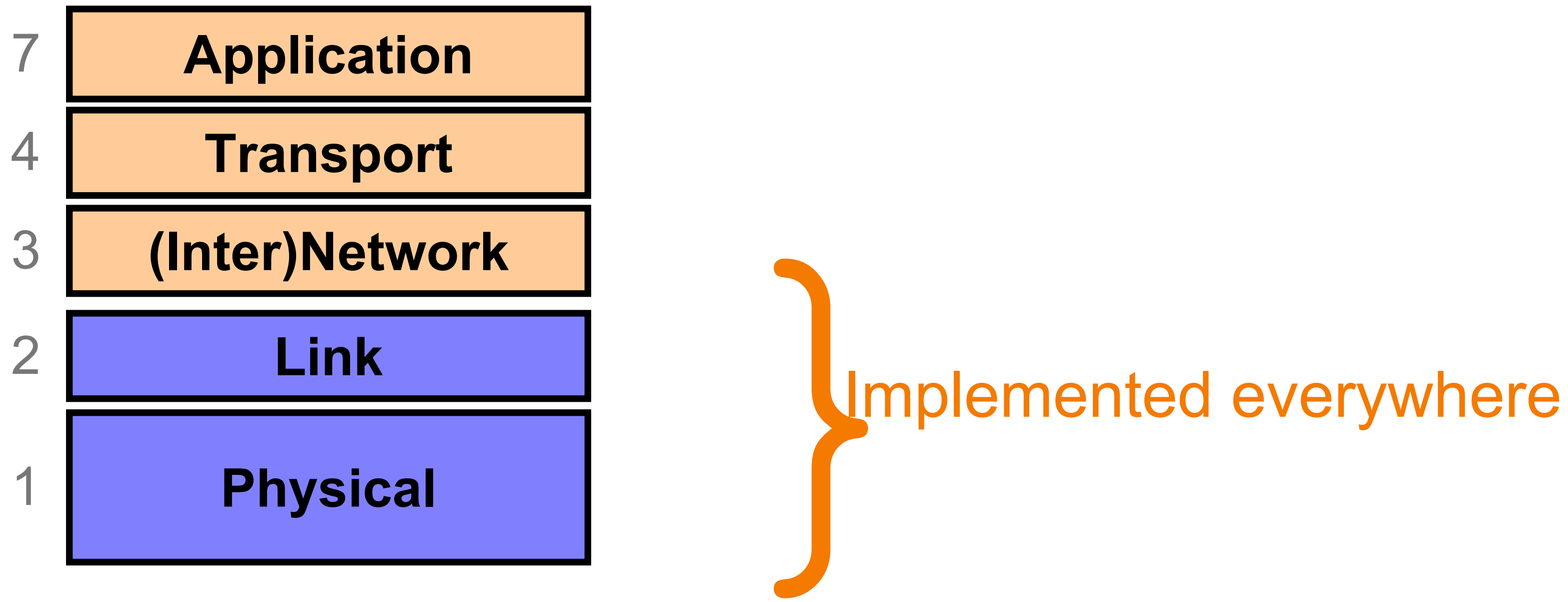
E.g.:

- Skype, SMTP (email), HTTP (Web), Halo, BitTorrent

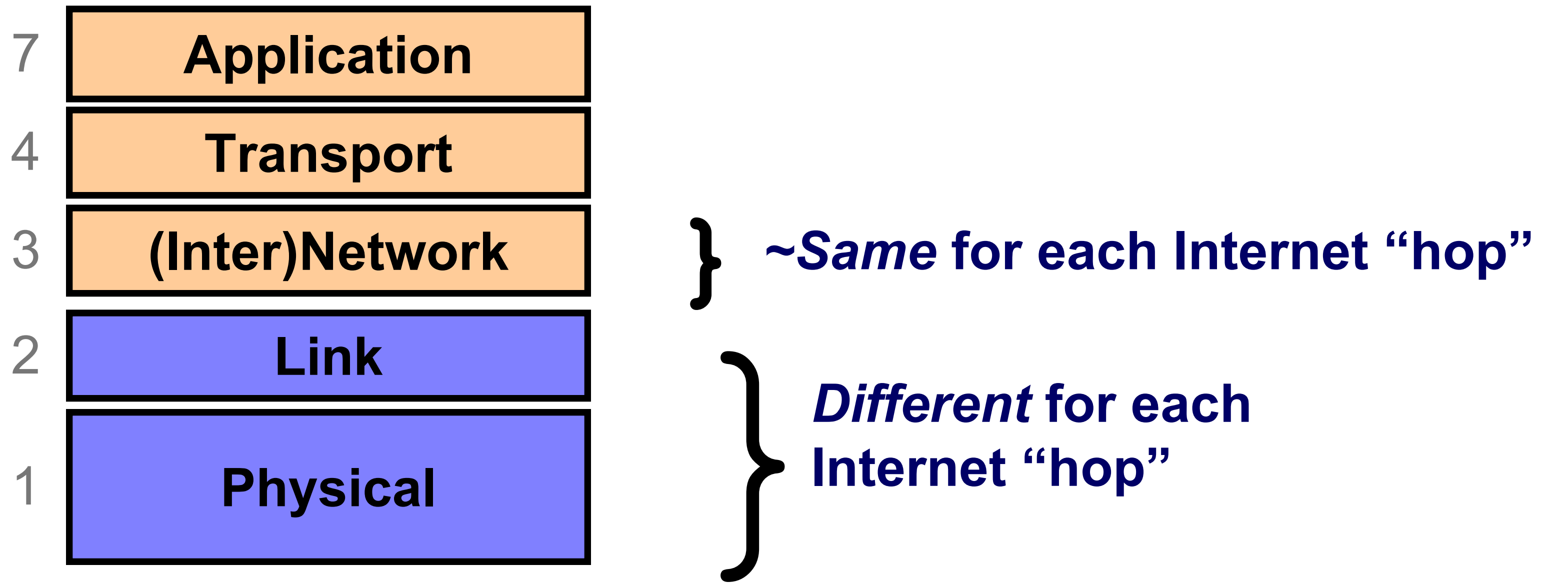
Internet Layering ("Protocol Stack")



Internet Layering ("Protocol Stack")

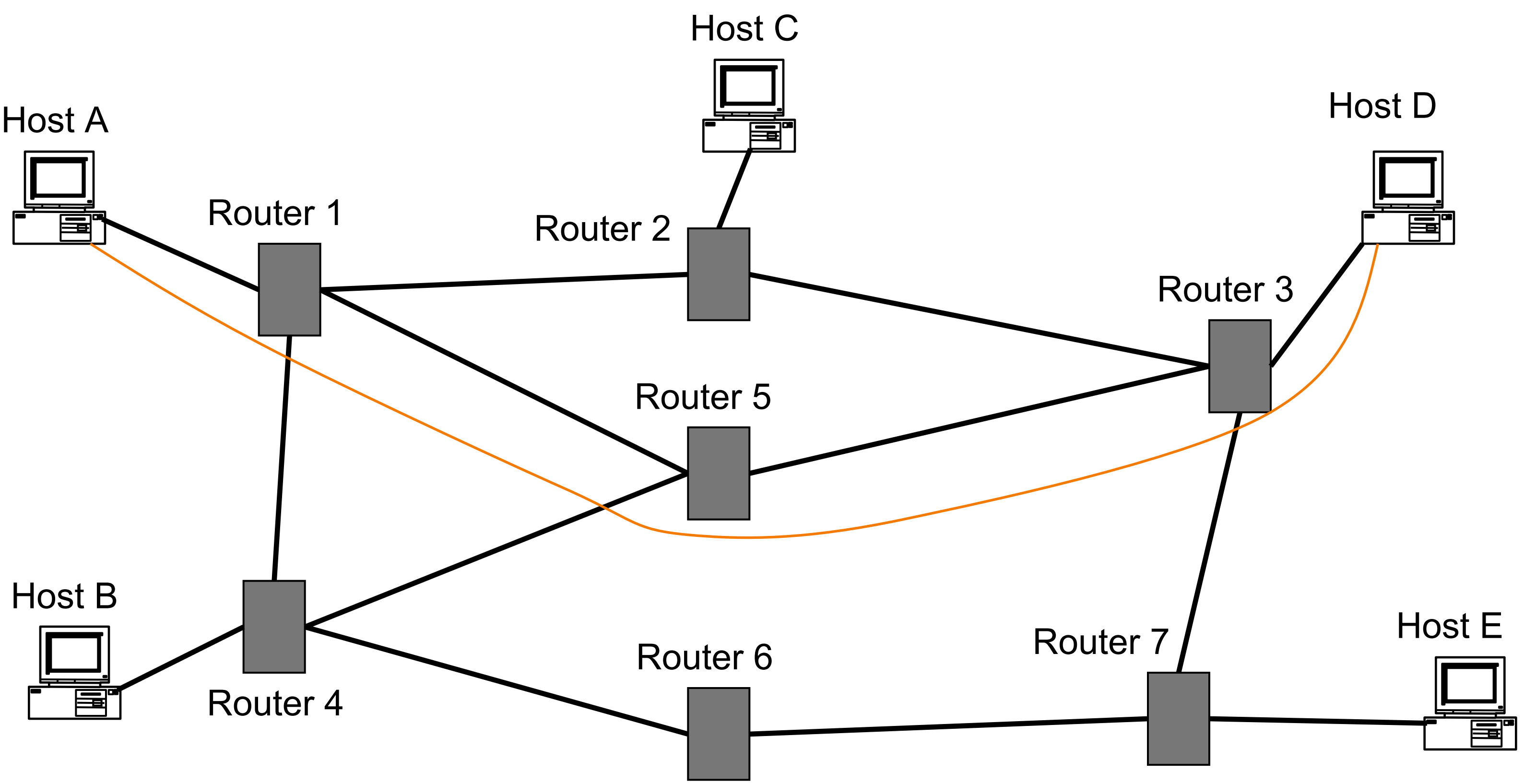


Internet Layering ("Protocol Stack")



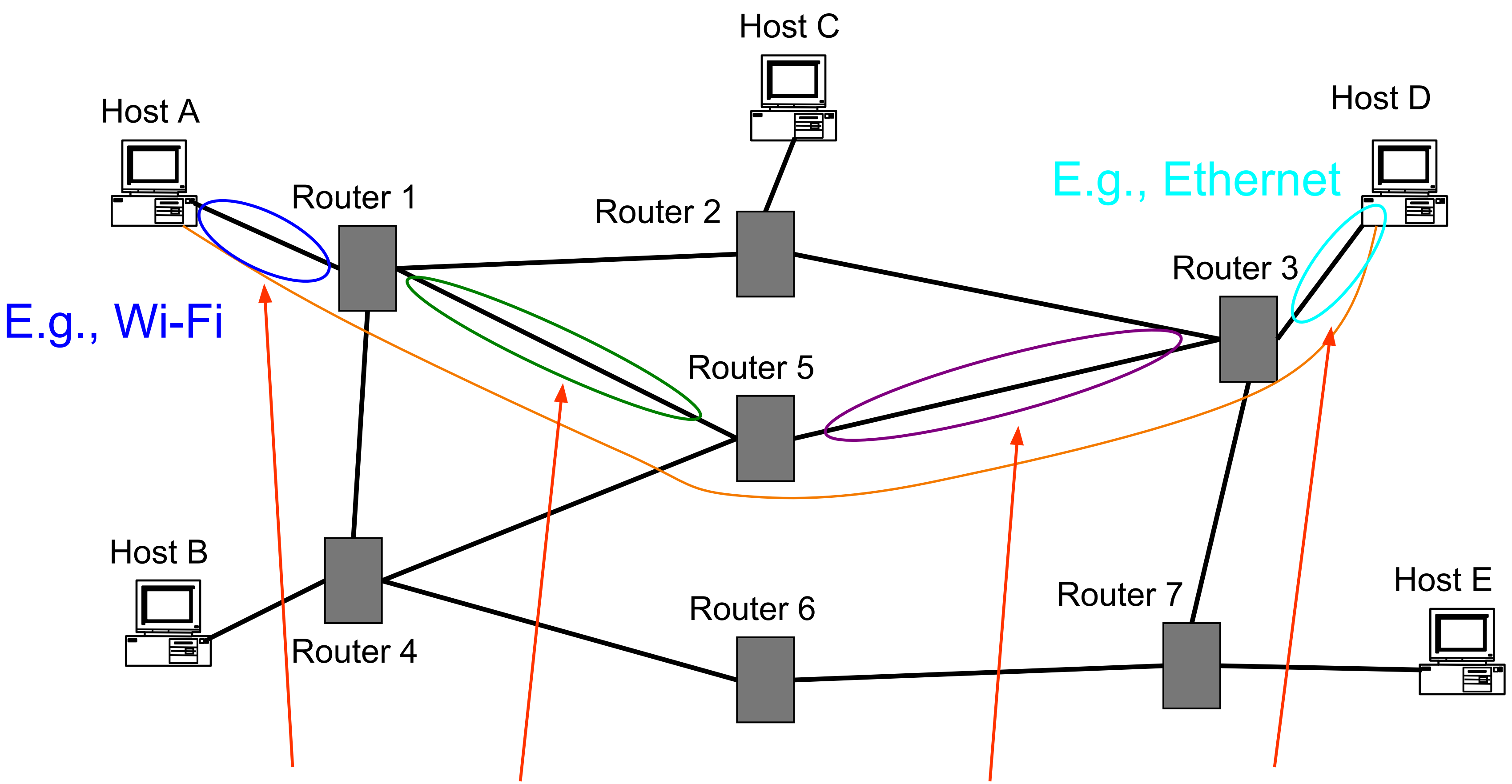
Hop-By-Hop vs. End-to-End Layers

Host A communicates with Host D



Hop-By-Hop vs. End-to-End Layers

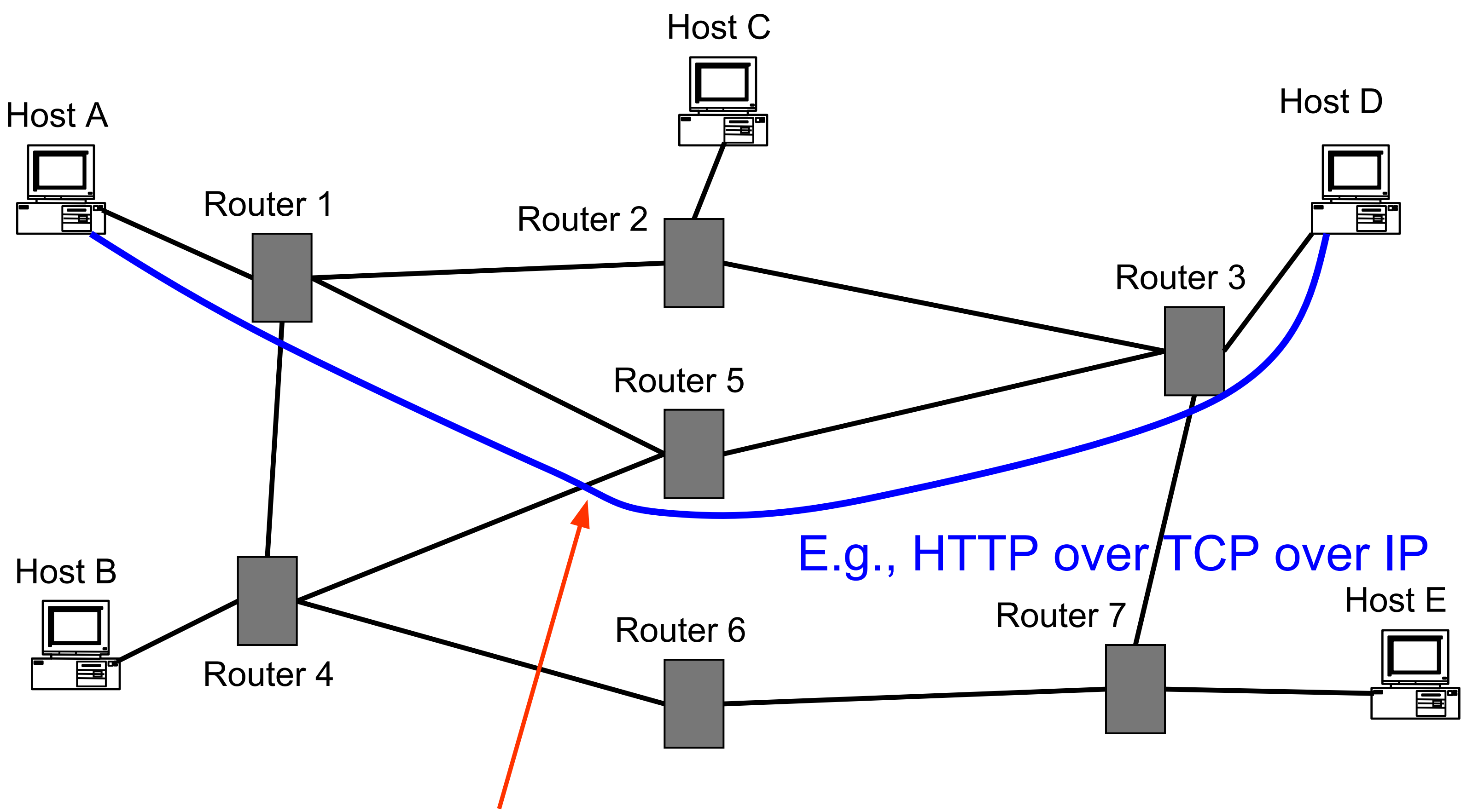
Host A communicates with Host D



***Different* Physical & Link Layers (Layers 1 & 2)**

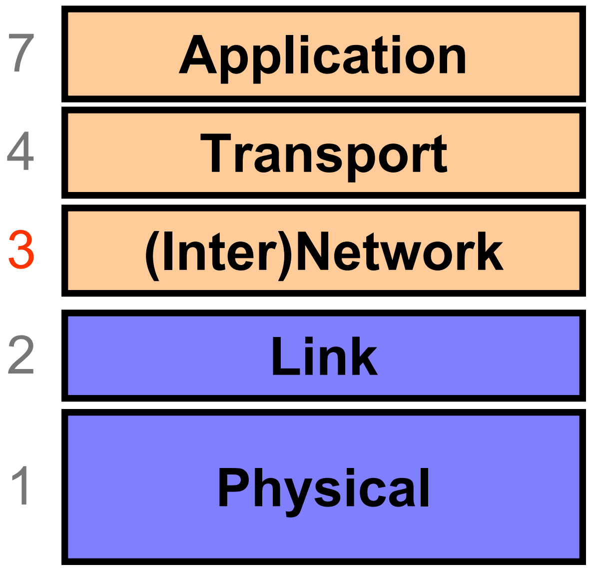
Hop-By-Hop vs. End-to-End Layers

Host A communicates with Host D



Same Network / Transport / Application Layers (3/4/7)
(Routers **ignore** Transport & Application layers)

Layer 3: (Inter)Network Layer *(IP)*

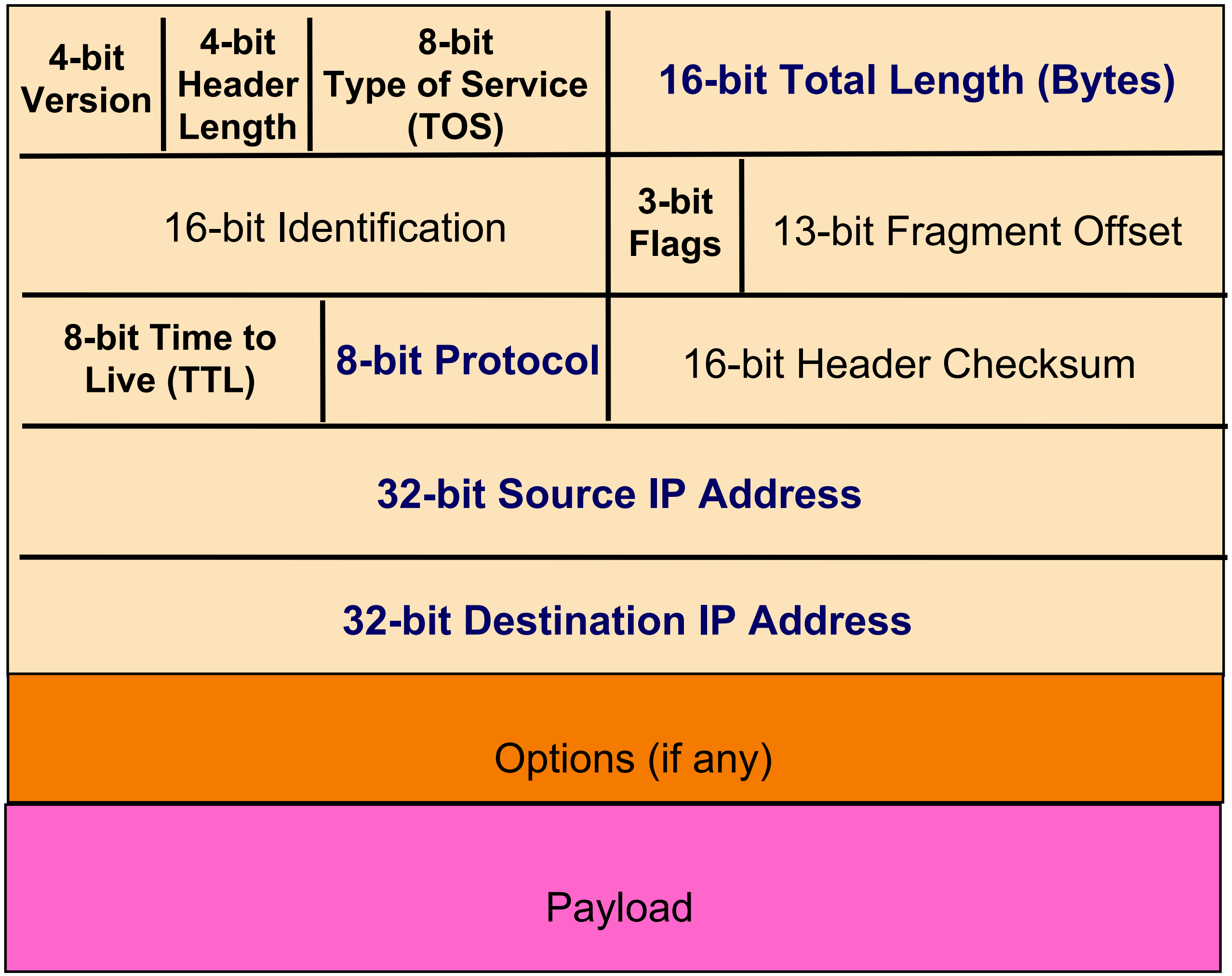


Bridges multiple “subnets” to provide *end-to-end* internet connectivity between nodes

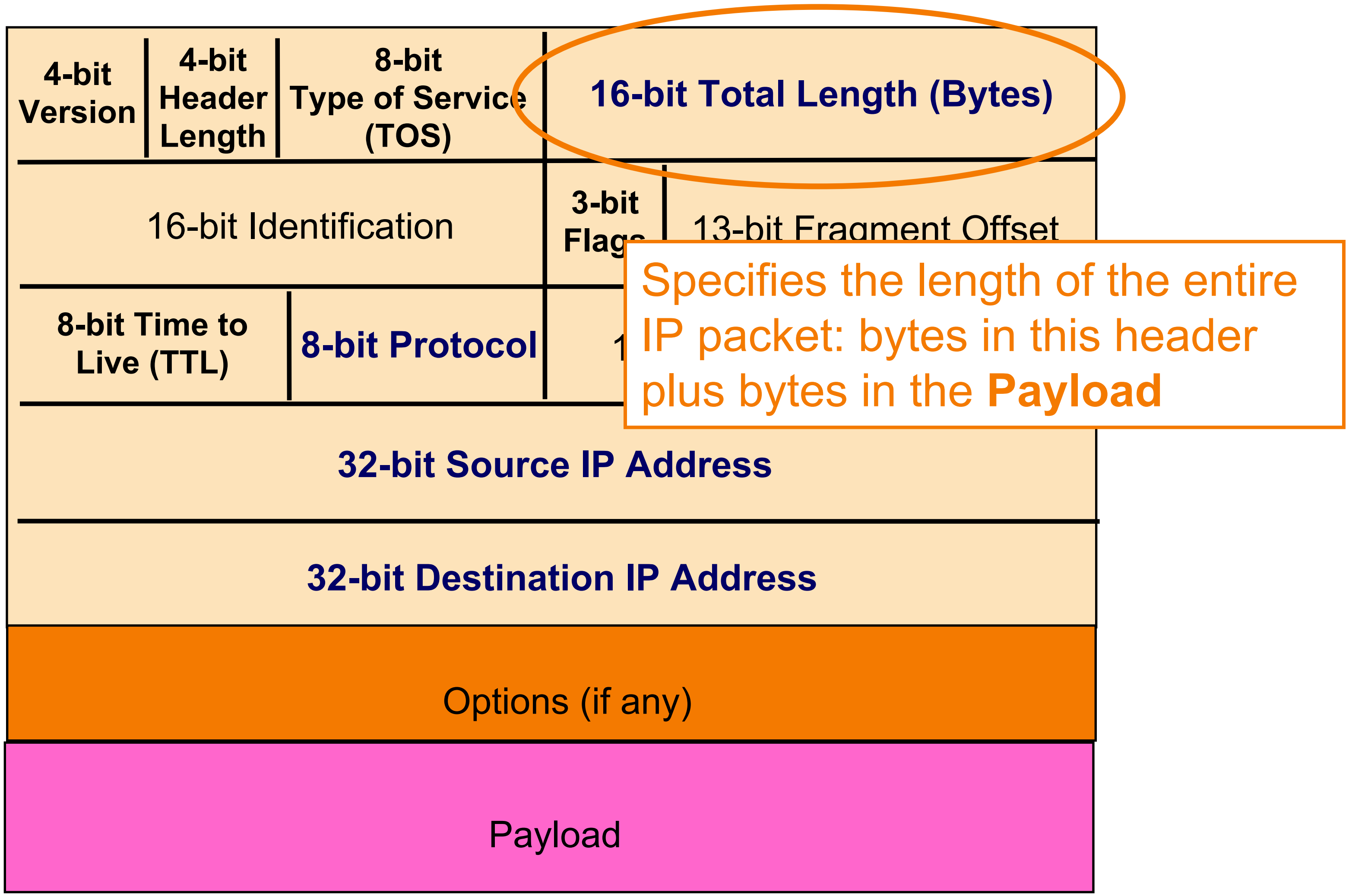
- Provides global addressing

Works across different link technologies

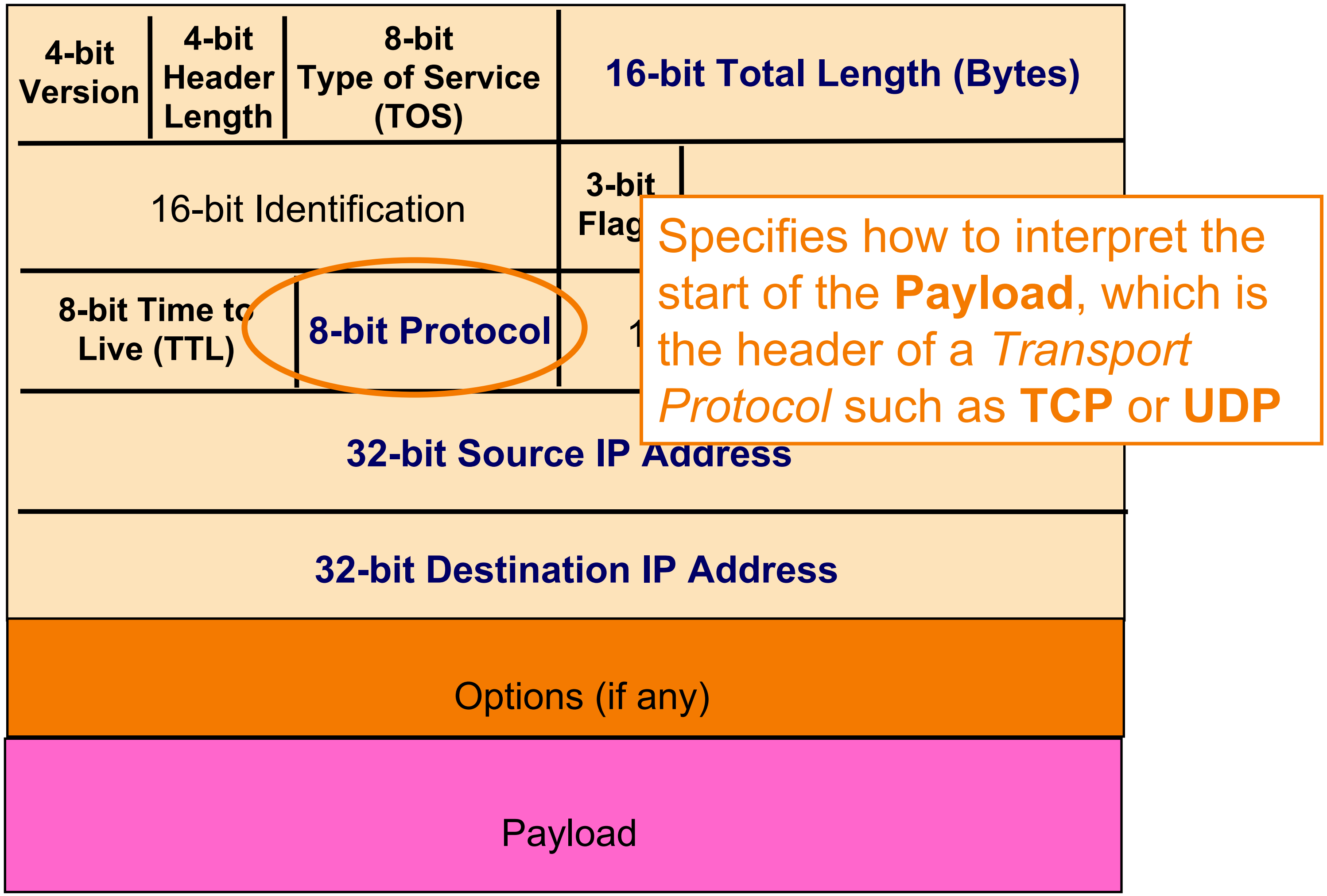
IP Packet Structure



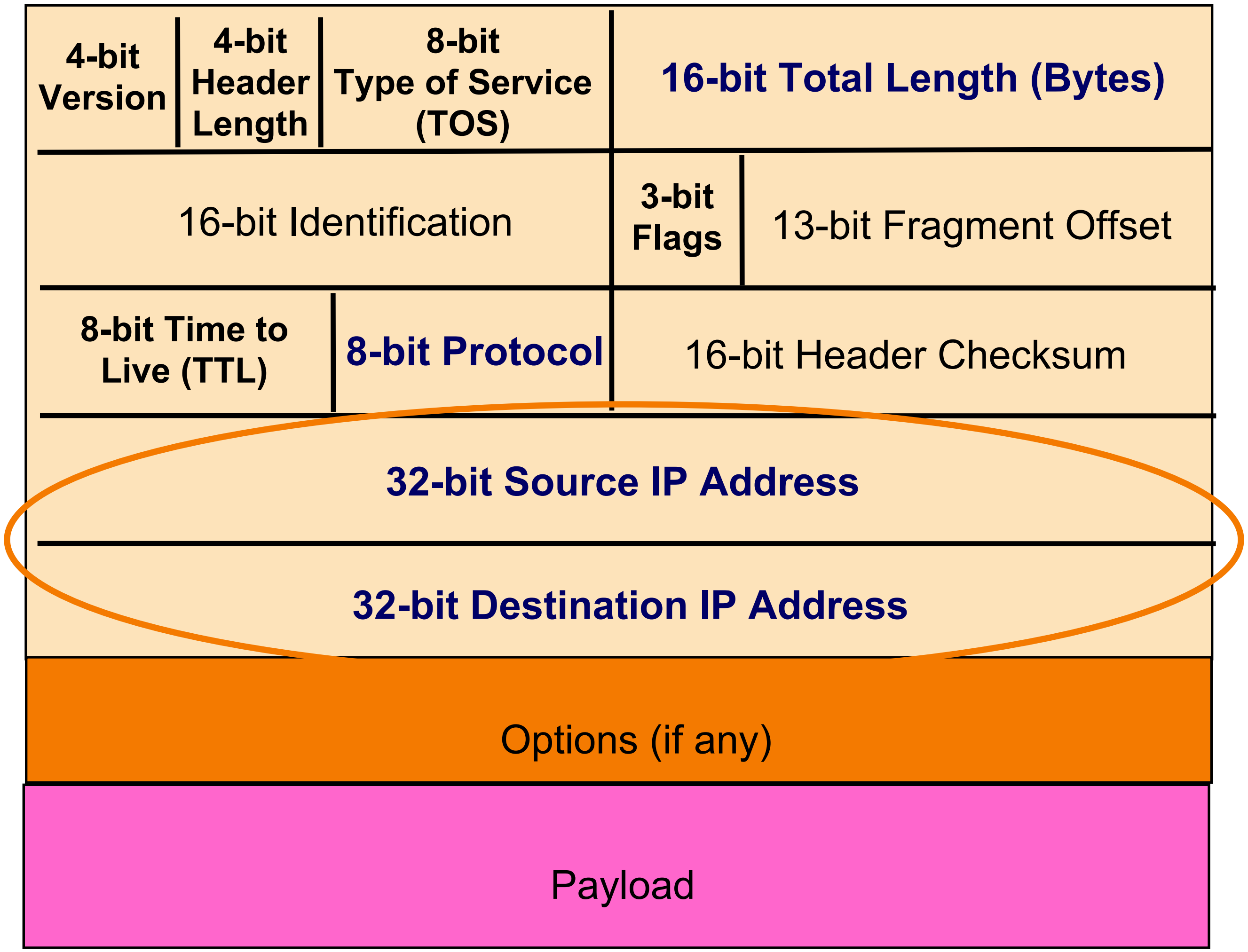
IP Packet Structure



IP Packet Structure



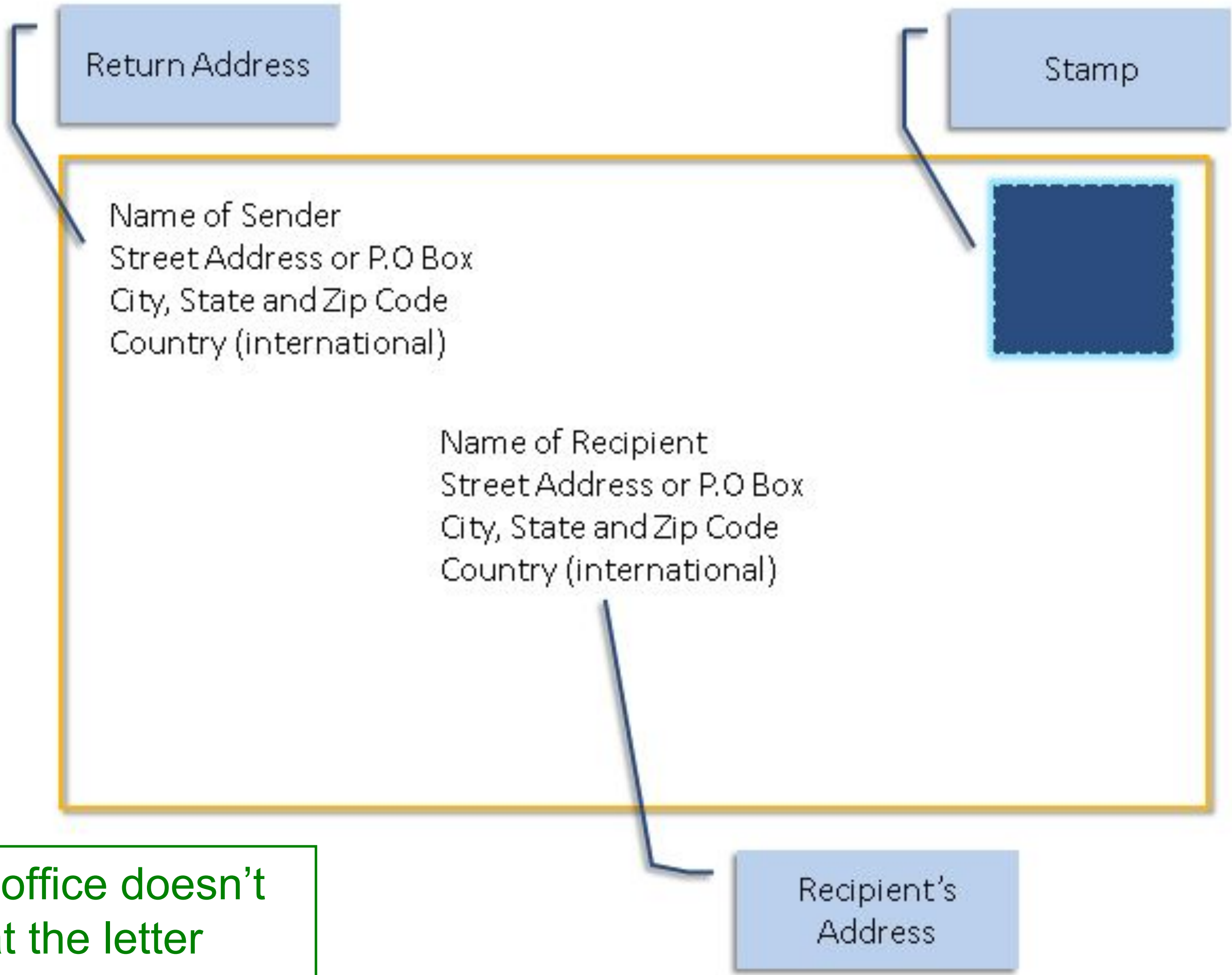
IP Packet Structure



IP Packet Header (Continued)

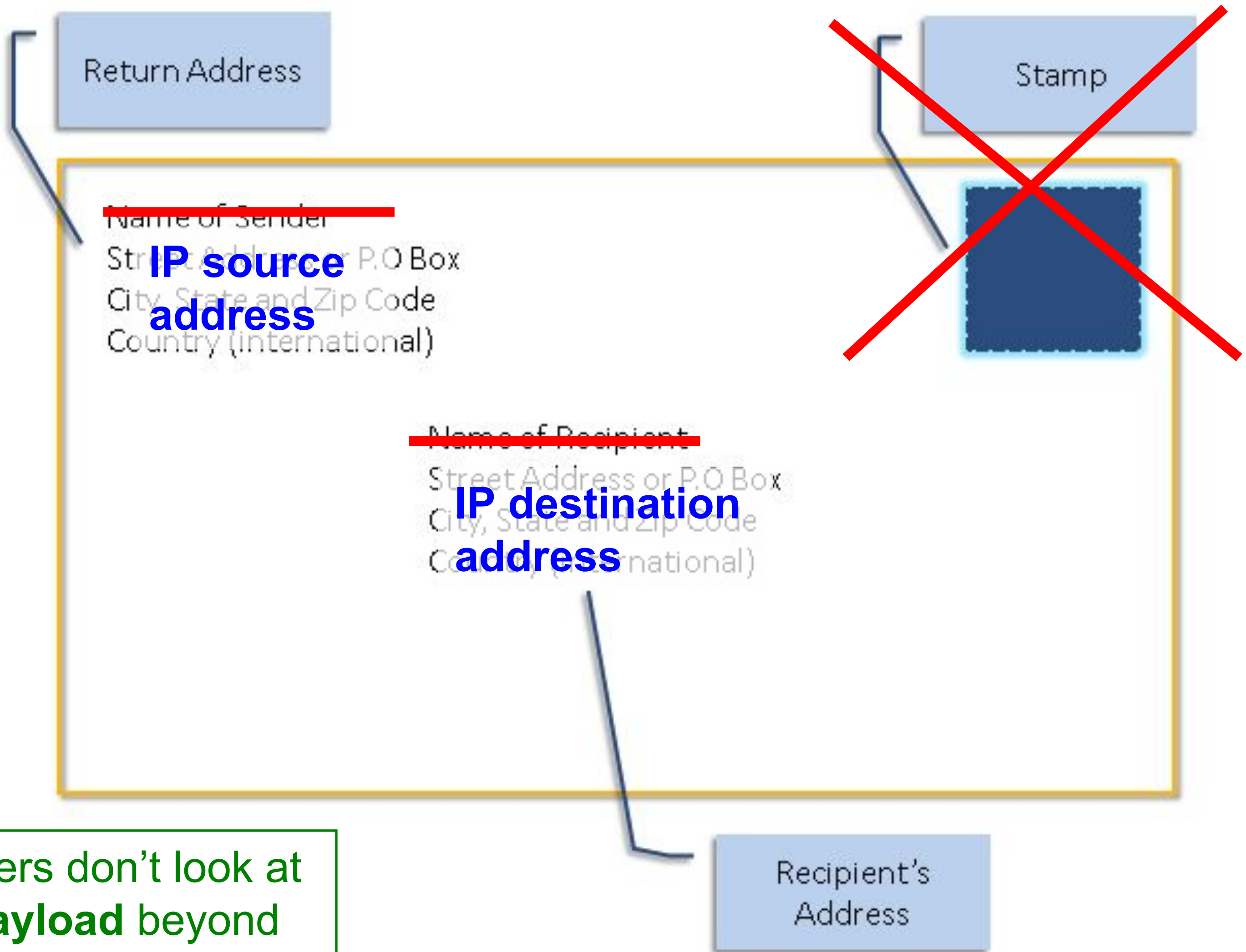
- Two IP addresses
  - Source IP address (32 bits)
  - Destination IP address (32 bits)
- Destination address
  - Unique identifier/locator for the receiving host
  - Allows each node to make forwarding decisions
- Source address
  - Unique identifier/locator for the sending host
  - Recipient can decide whether to accept packet
  - Enables recipient to send a reply back to source

Postal Envelopes:



(Post office doesn't look at the letter inside the envelope)

Analogy of IP to Postal Envelopes:



(Routers don't look at the **payload** beyond the IP header)

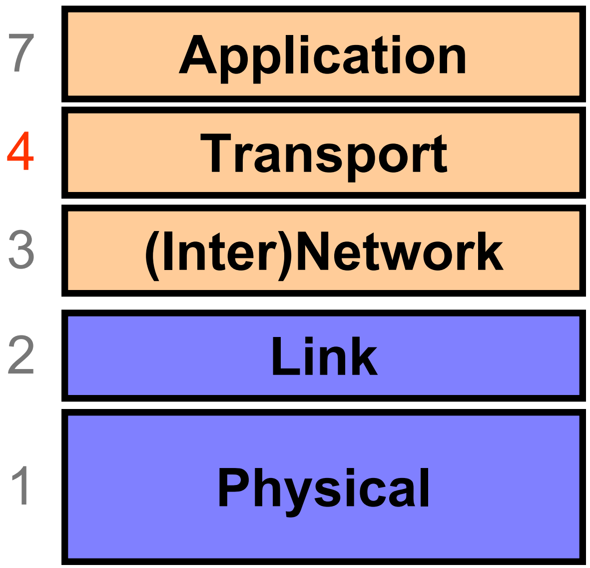
IP: “*Best Effort* ” Packet Delivery

- Routers inspect destination address, locate “next hop” in forwarding table
  - Address = ~unique identifier/locator for the receiving host
- Only provides a “*I’ll give it a try*” delivery service:
  - Packets may be lost
  - Packets may be corrupted
  - Packets may be delivered out of order

"Best Effort" is Lame! What to do?

- It's the job of our Transport (layer 4) protocols to build services our apps need out of IP's modest layer-3 service

Layer 4: Transport Layer



End-to-end communication between processes

Different services provided:

TCP = reliable *byte stream*

UDP = unreliable *datagrams*

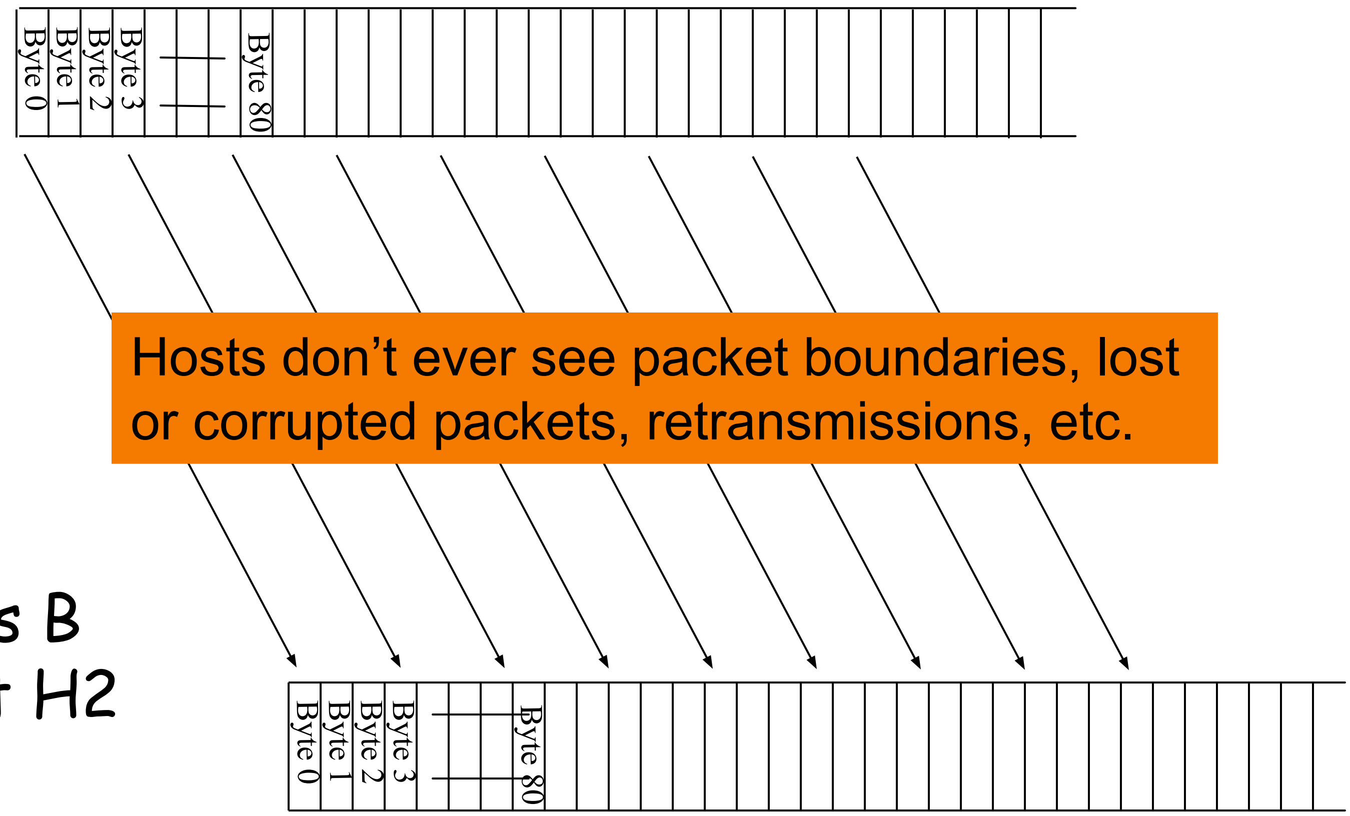
(Datagram = single packet message)

“Best Effort” is Lame! What to do?

- It’s the job of our Transport (layer 4) protocols to build services our apps need out of IP’s modest layer-3 service
- #1 workhorse: TCP (Transmission Control Protocol)
- Service provided by TCP:
  - Connection oriented (explicit set-up / tear-down)
    - End hosts (processes) can have multiple concurrent long-lived communication
  - **Reliable**, in-order, *byte-stream* delivery
    - Robust detection & retransmission of lost data

TCP “Bytestream” Service

Process A on host H1

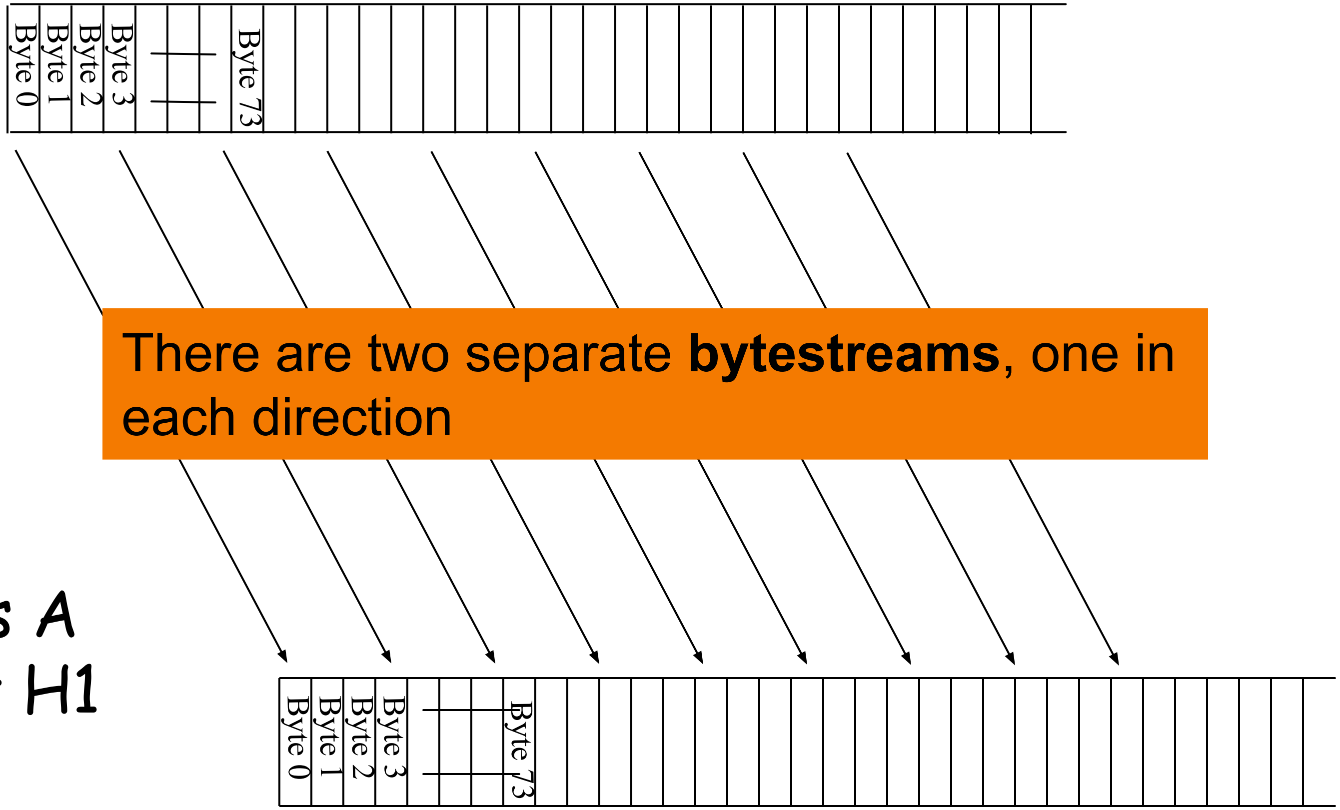


Hosts don’t ever see packet boundaries, lost or corrupted packets, retransmissions, etc.

Process B
on host H2

Bidirectional communication:

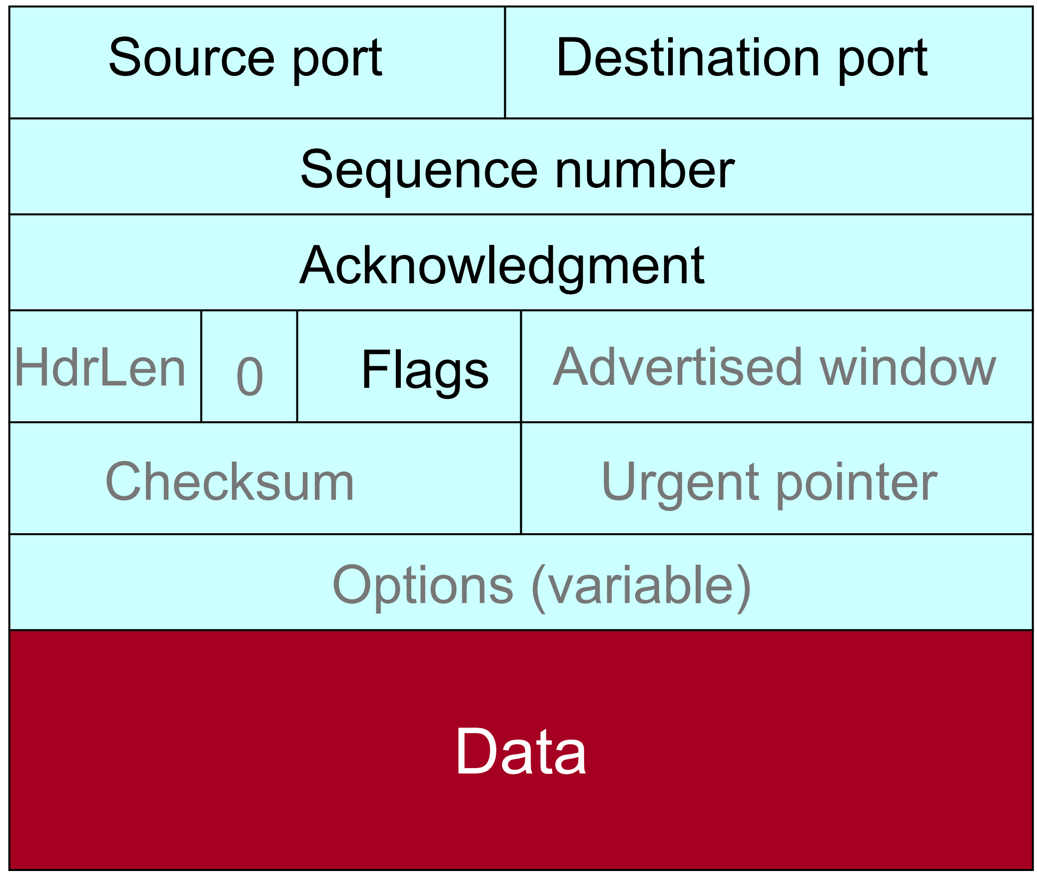
Process B on host H2



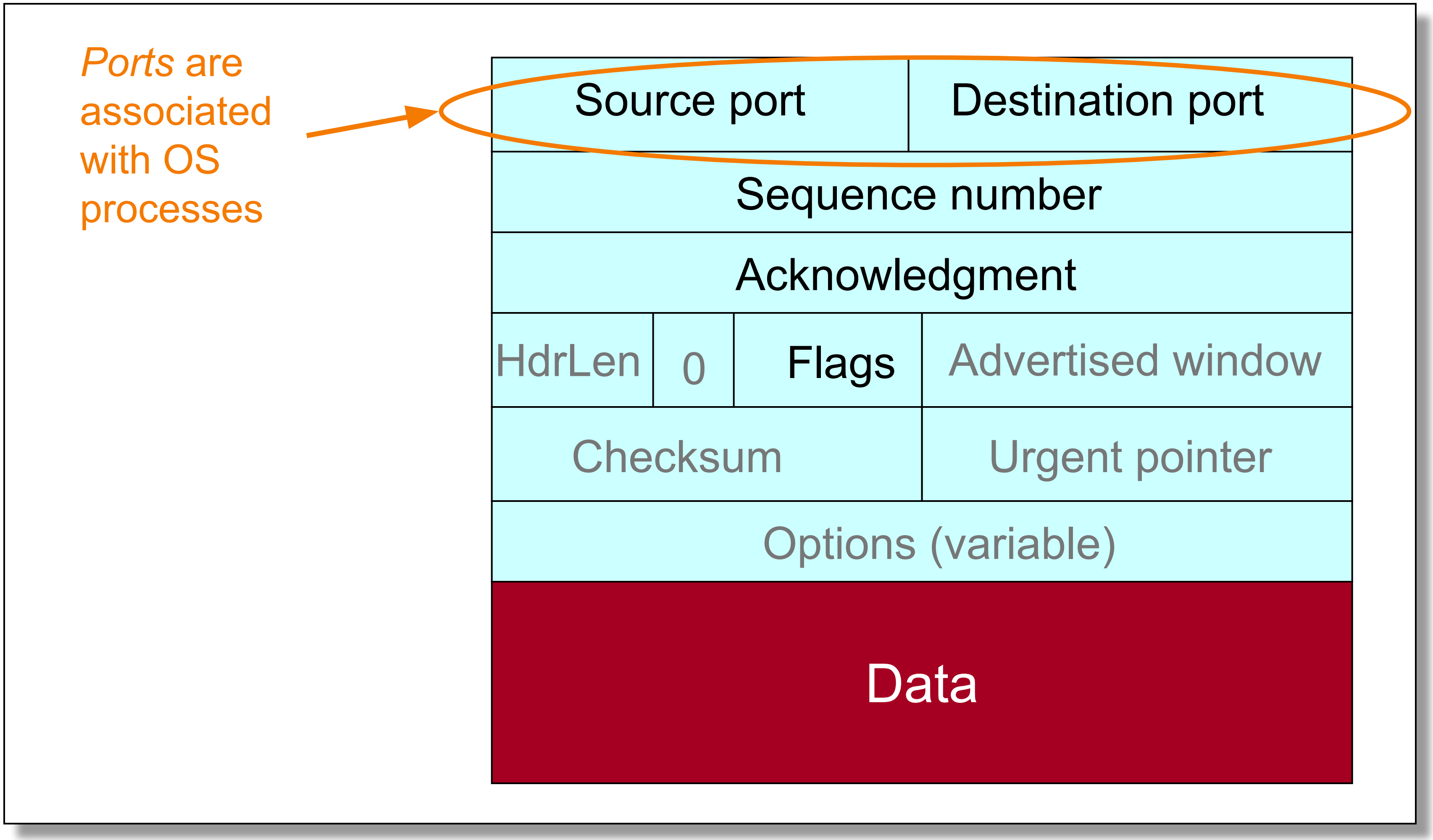
There are two separate **bytestreams**, one in each direction

Process A
on host H1

TCP Header


Source port
Destination port
Sequence number
Acknowledgment
HdrLen
0
Flags
Advertised window
Checksum
Urgent pointer
Options (variable)
Data

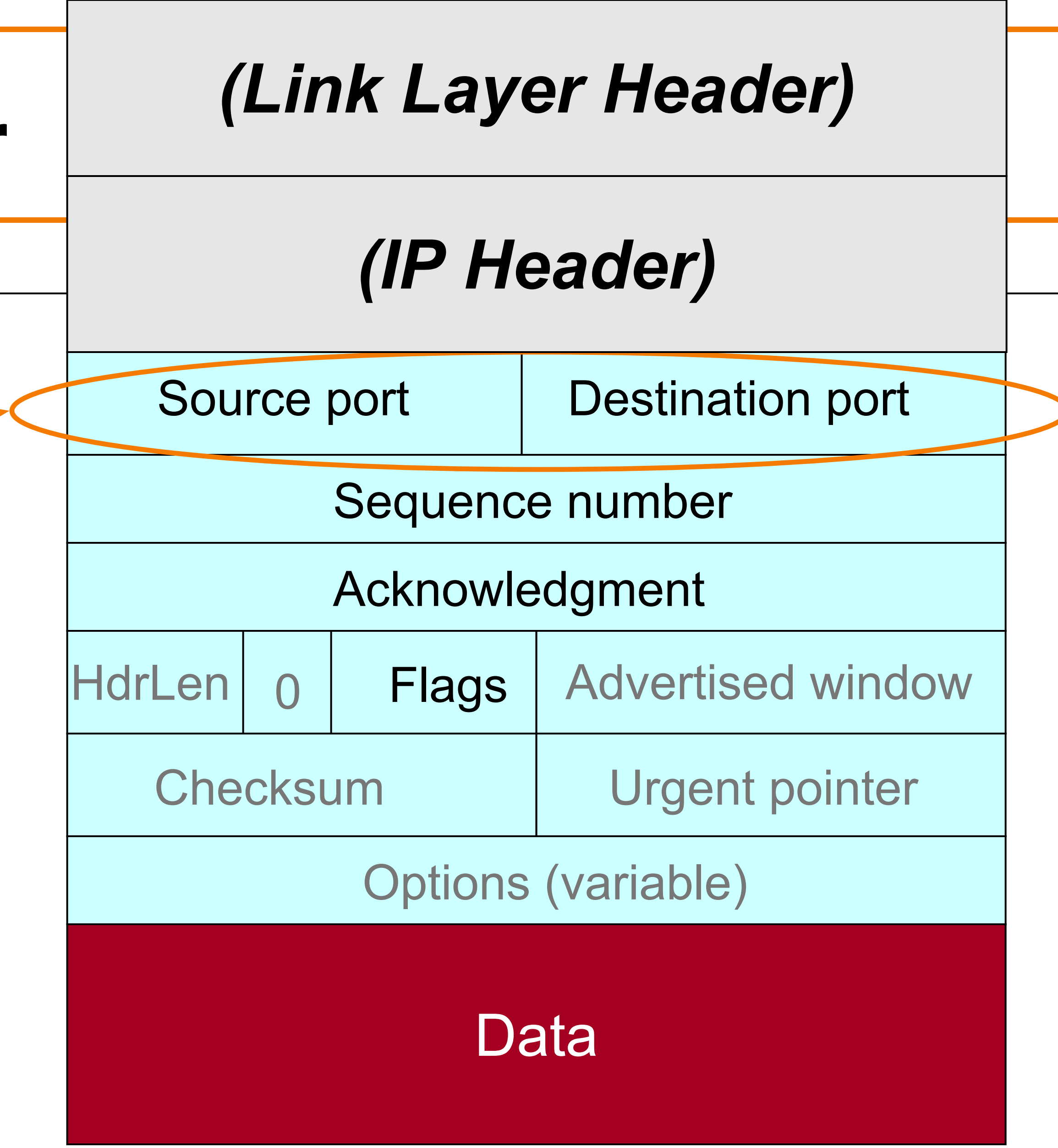
TCP Header



Ports are associated with OS processes
Source port
Destination port
Sequence number
Acknowledgment
HdrLen
0
Flags
Advertised window
Checksum
Urgent pointer
Options (variable)
Data

TCP Header

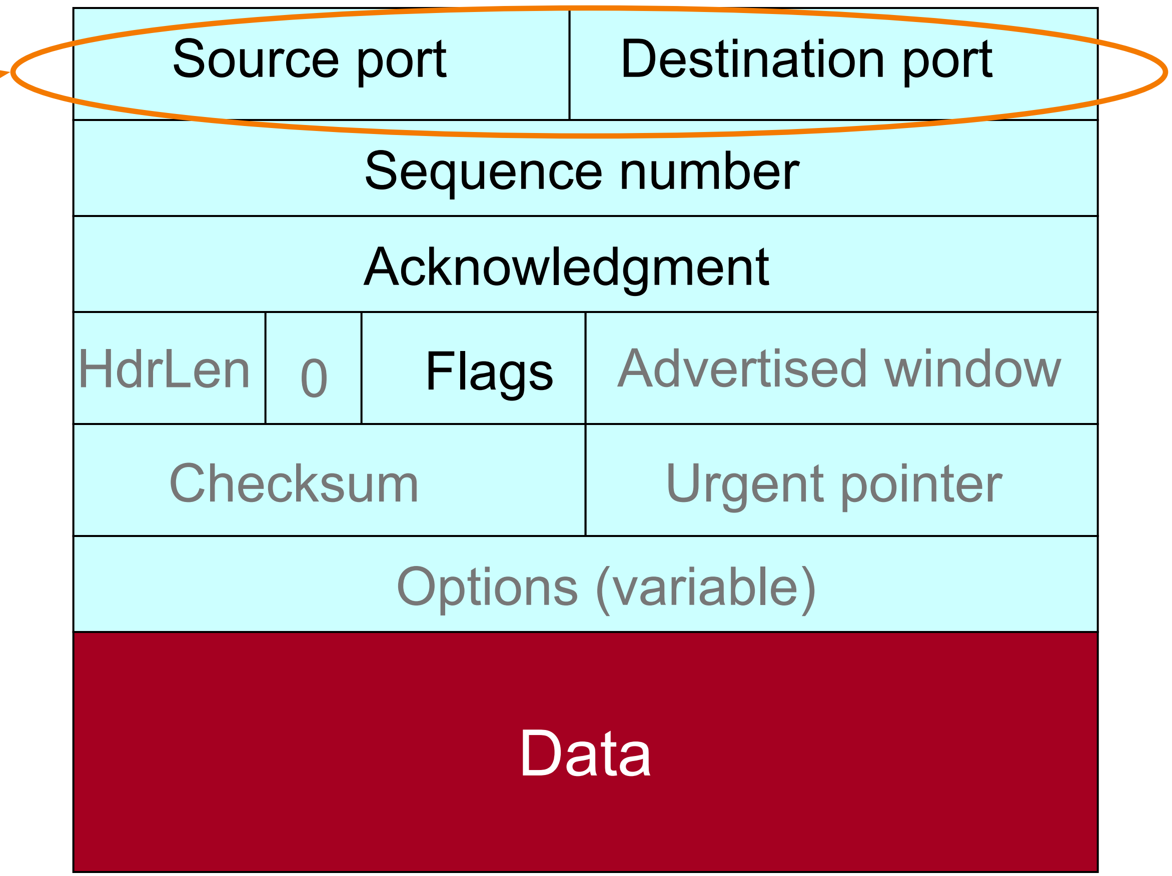
Ports are associated with OS processes



IP source & destination addresses plus TCP source and destination ports uniquely identifies a TCP connection

TCP Header

Ports are associated with OS processes

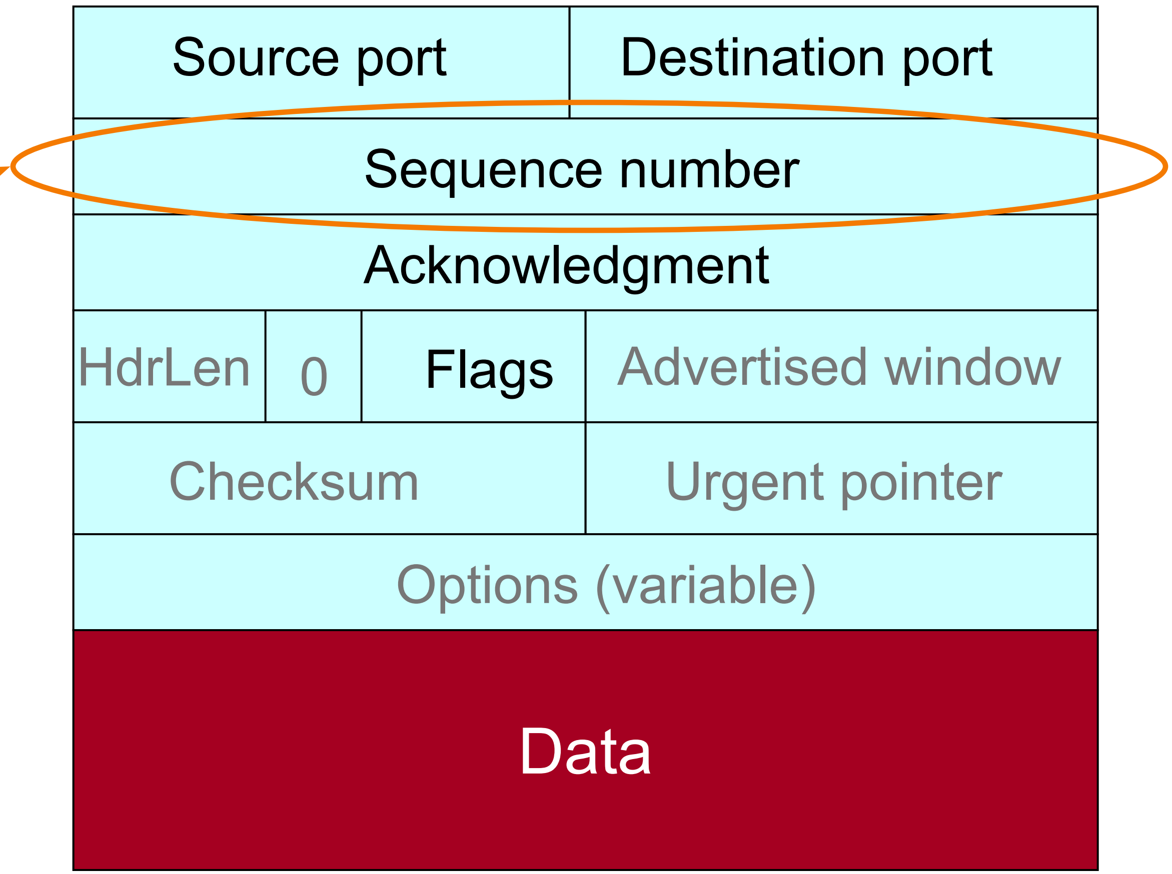


IP source & destination addresses plus TCP source and destination ports uniquely identifies a TCP connection

Some port numbers are "well known" / reserved e.g. port 80 = HTTP

TCP Header

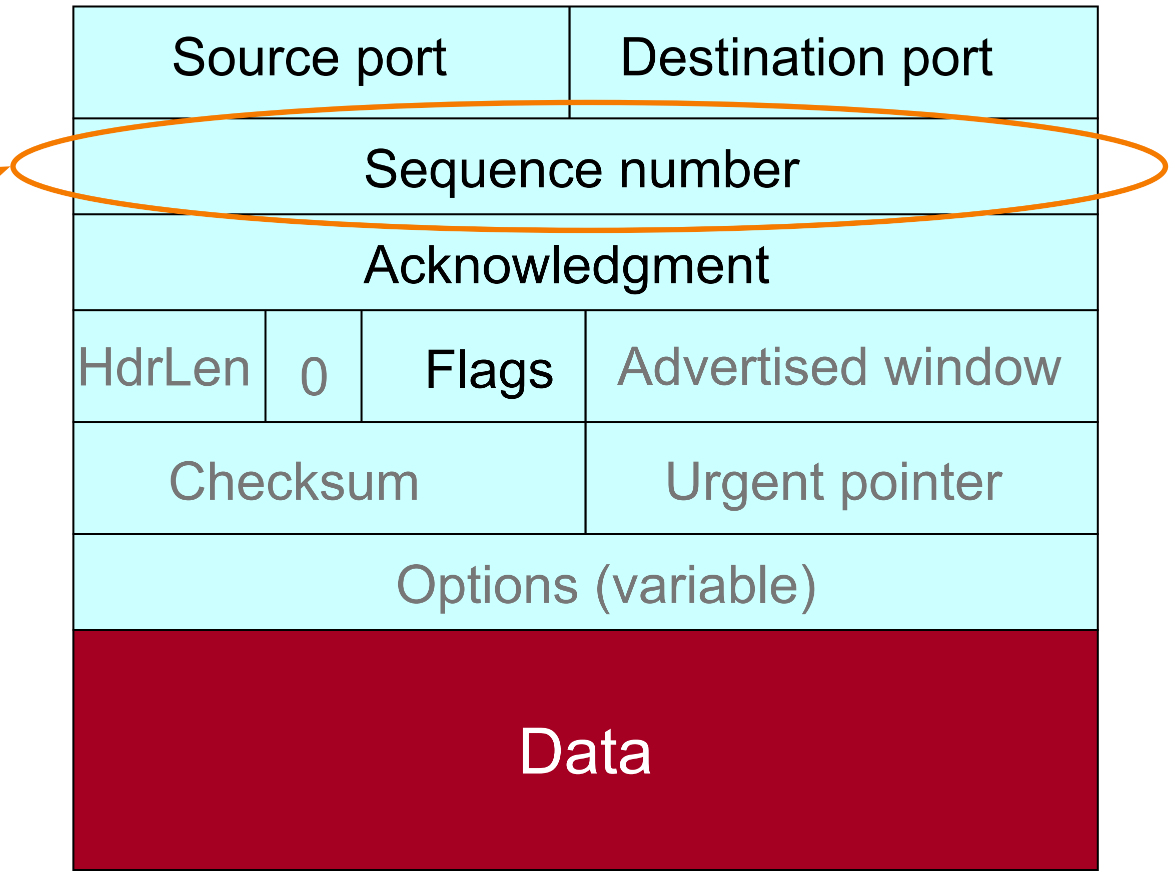
Starting sequence number (byte offset) of data carried in this packet



TCP Header

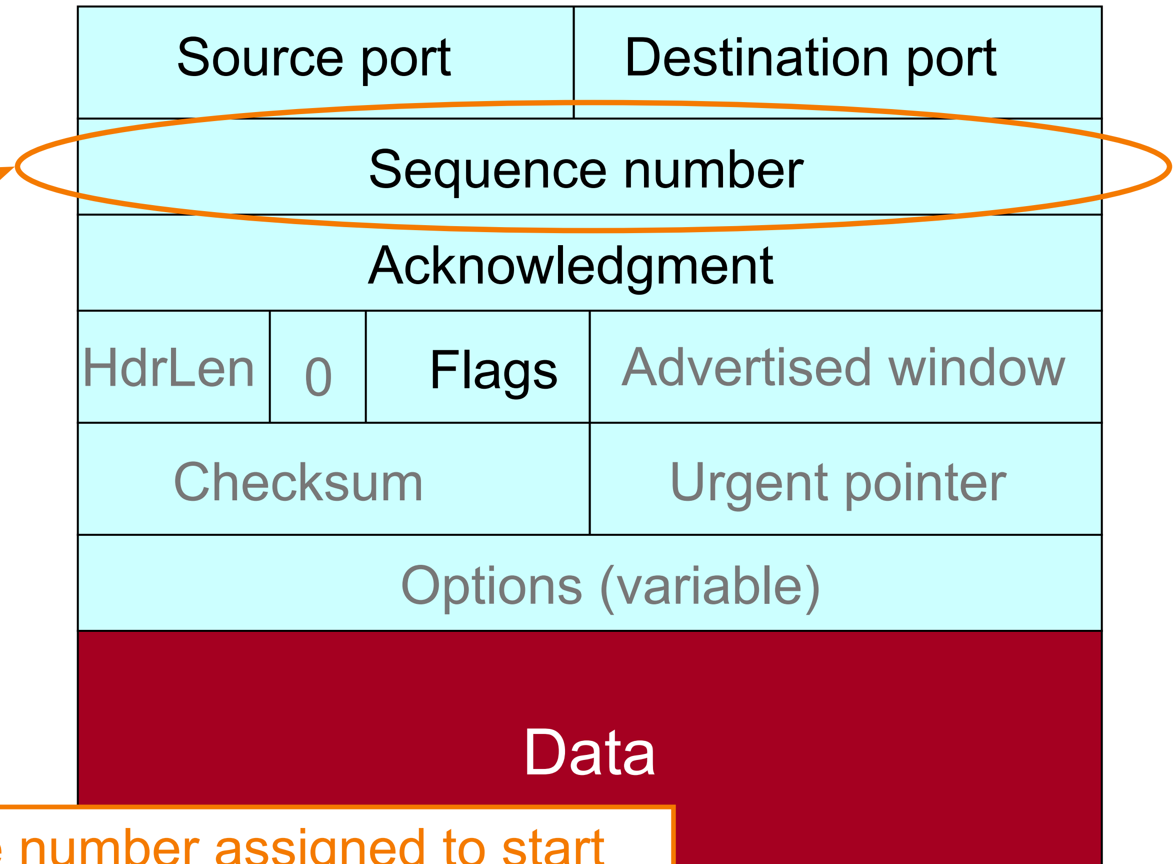
Starting sequence number (byte offset) of data carried in this packet

Byte streams numbered independently in each direction



TCP Header

Starting sequence number (byte offset) of data carried in this packet



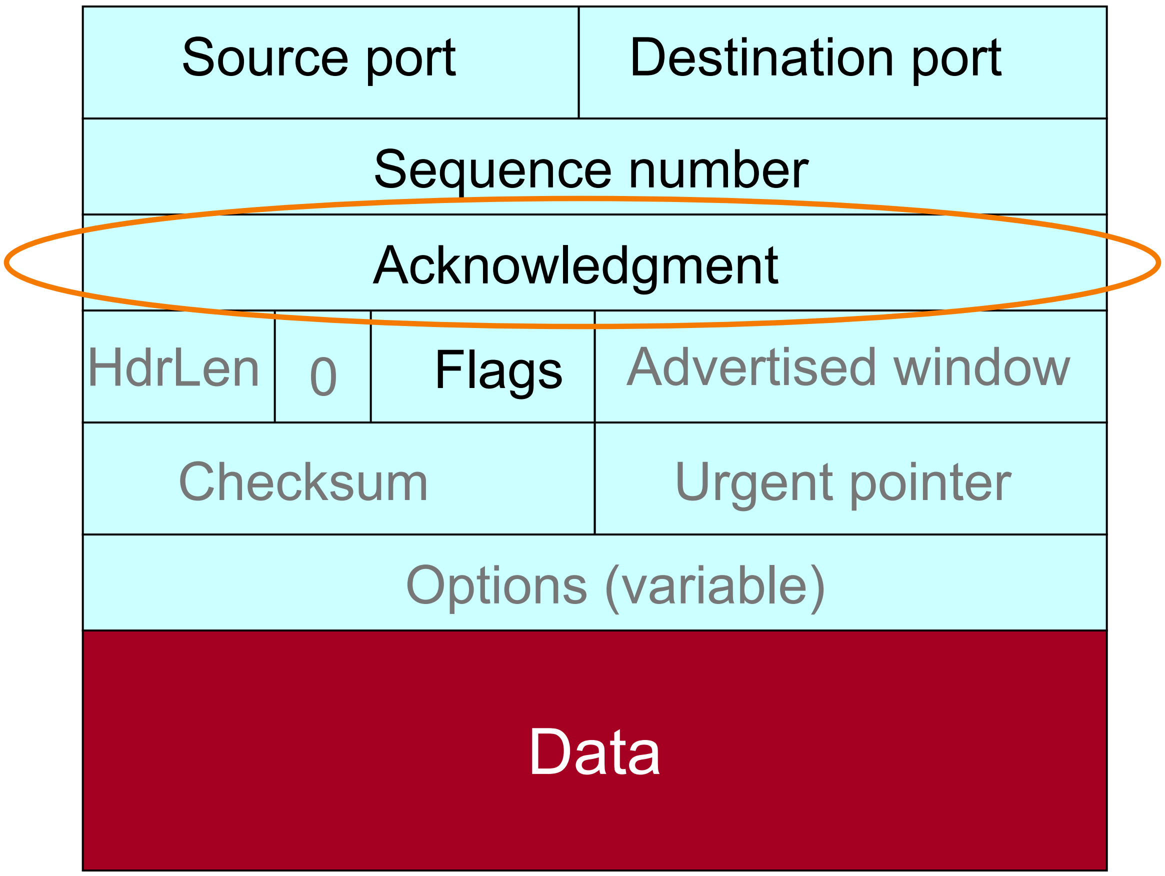
Byte stream numbered independently in each direction

Sequence number assigned to start of byte stream is picked when connection begins; **doesn’t** start at 0

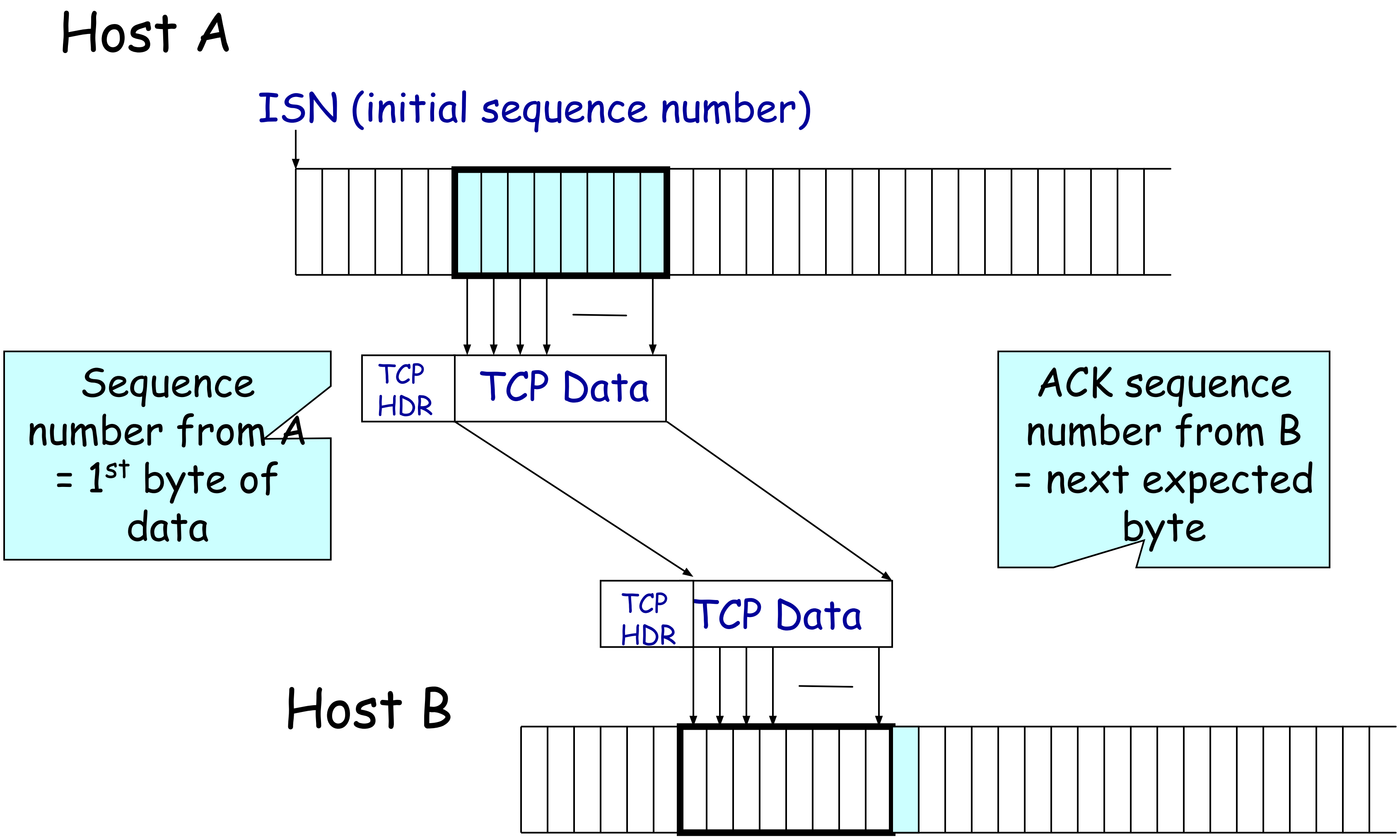
TCP Header

Acknowledgment gives seq # **just beyond** highest seq. received **in order**.

If sender sends **N** bytestream bytes starting at seq **S** then "ack" for it will be **S+N**.



Sequence Numbers

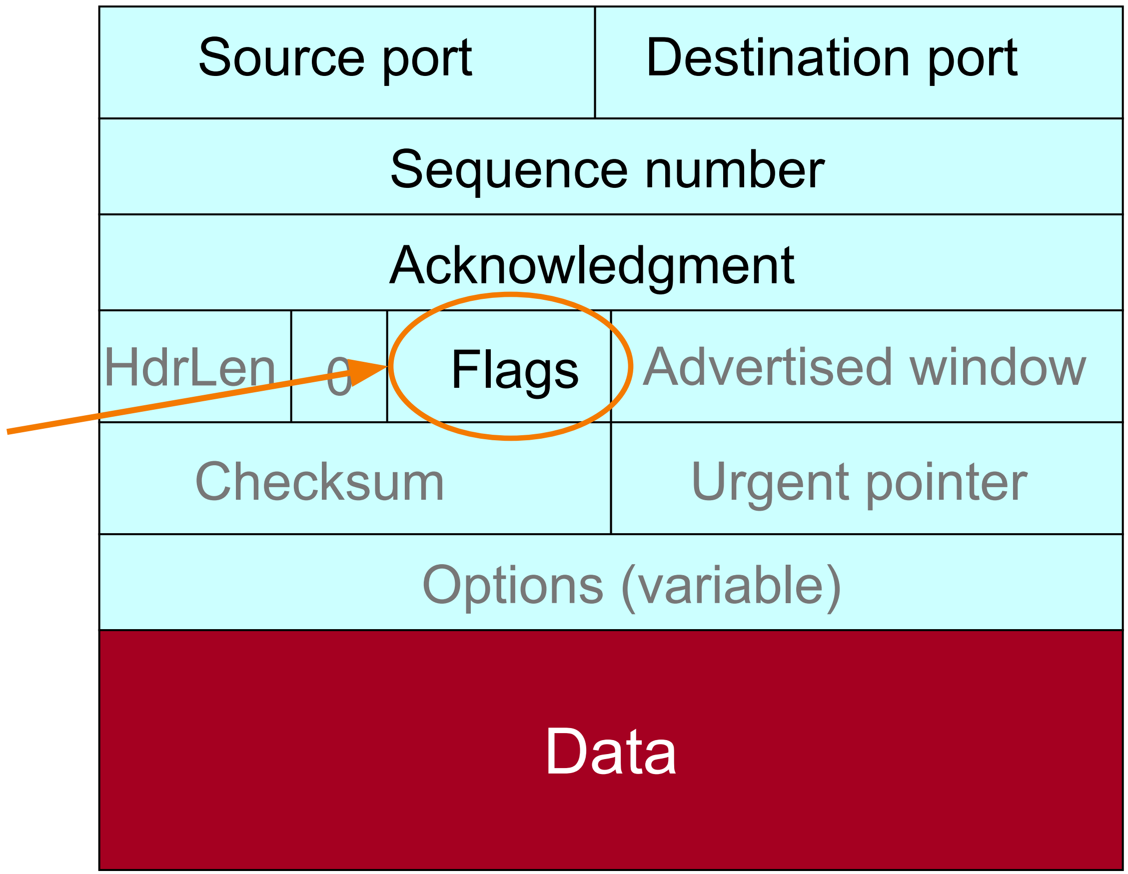


TCP Header

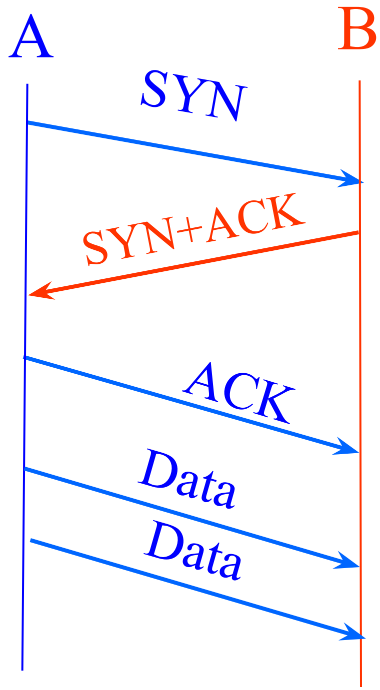
Uses include:

acknowledging data (“**ACK**”)

setting up (“**SYN**”) and closing connections (“**FIN**” and “**RST**”)



Establishing a TCP Connection

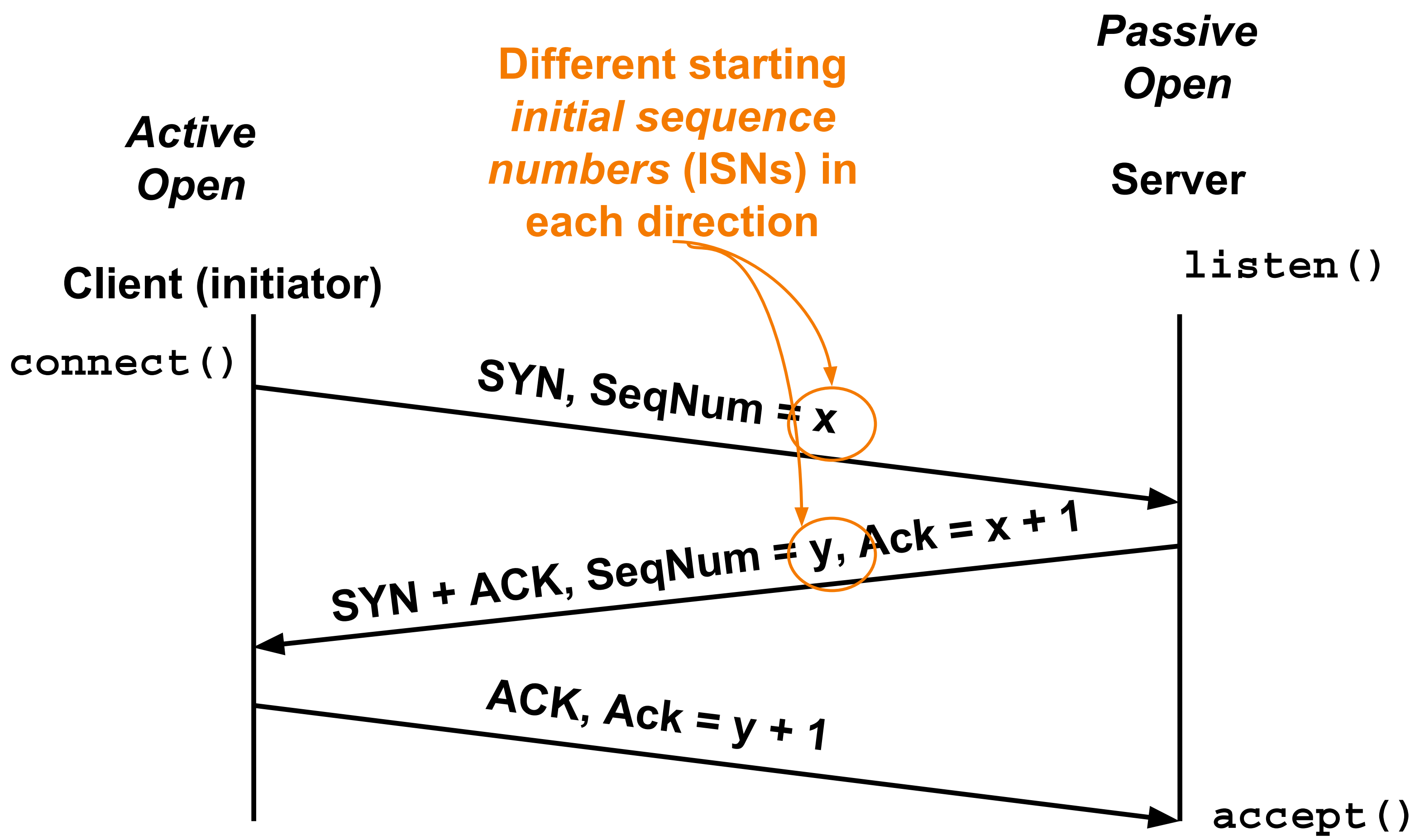


Each host tells its *Initial Sequence Number* (ISN) to the other host.

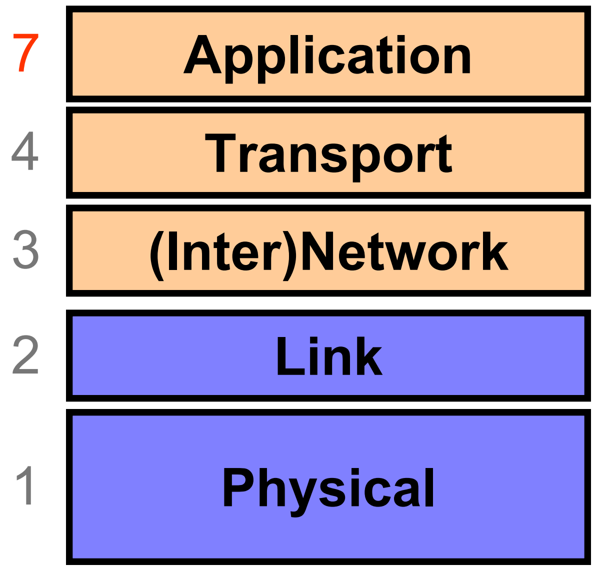
(Spec says to pick based on local clock)

- Three-way handshake to establish connection
  - Host A sends a **SYN** (open; "synchronize sequence numbers") to host B
  - Host B returns a SYN acknowledgment (**SYN+ACK**)
  - Host A sends an **ACK** to acknowledge the SYN+ACK

Timing Diagram: 3-Way Handshaking



Layer 7: Application Layer



Communication of whatever you wish

Can use whatever transport(s) is convenient

Freely structured

E.g.:

Skype, SMTP (email), HTTP (Web), Halo, BitTorrent

Web (HTTP) Request

Method **Resource** **HTTP version** **Headers**

```
GET /index.html HTTP/1.1
Accept: image/gif, image/x-bitmap, image/jpeg, */*
Accept-Language: en
Connection: Keep-Alive
User-Agent: Mozilla/1.22 (compatible; MSIE 2.0; Windows 95)
Host: www.example.com
Referer: http://www.google.com?q=dingbats

```

Blank line

Data (if POST; none for GET)

GET: download data.	POST: upload data.

Web (HTTP) Response

HTTP version **Status code** **Reason phrase** **Headers** **Data**

```
HTTP/1.0 200 OK
Date: Sun, 19 Apr 2009 02:20:42 GMT
Server: Microsoft-Internet-Information-Server/5.0
Connection: keep-alive
Content-Type: text/html
Last-Modified: Sat, 18 Apr 2009 17:39:05 GMT
Set-Cookie: session=44eb; path=/servlets
Content-Length: 2543

<HTML> Some data... blah, blah, blah </HTML>
```

Host Names vs. IP addresses

- Host names
  - Examples: `www.cnn.com` and `bbc.co.uk`
  - Mnemonic name appreciated by humans
  - Variable length, full alphabet of characters
  - Provide little (if any) information about location
- IP addresses
  - Examples: `64.236.16.20` and `212.58.224.131`
  - Numerical address appreciated by routers
  - Fixed length, binary number
  - Hierarchical, related to host location

Networking Attacks: Link-, IP-, and TCP-layer attacks

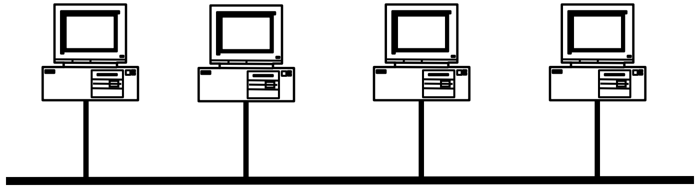
General Communication Security Goals: CIA

- Confidentiality:
  - No one can *read* our data / communication unless we want them to
- Integrity
  - No one can *manipulate* our data / processing / communication unless we want them to
- Availability
  - We can *access* our data / conduct our processing / use our communication capabilities when we want to

No security built in at the network level

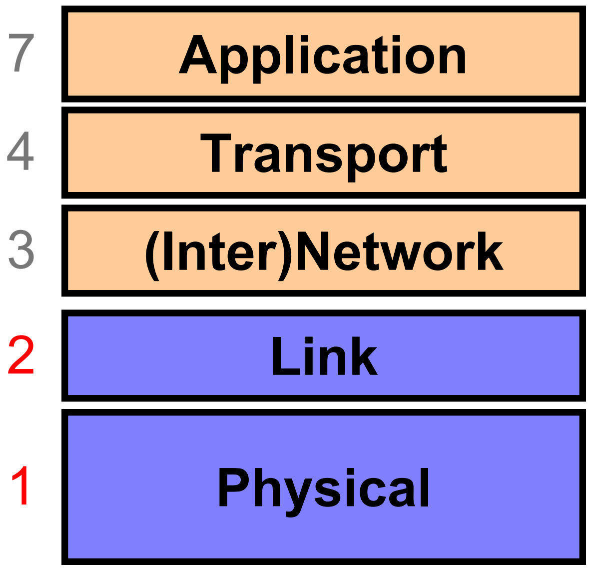
- Everything you have seen in this lecture is just plaintext, to integrity attached to it so an attacker can easily spoof packets at multiple levels
- TLS will give application level security

Link-layer threats



- Confidentiality: eavesdropping (aka sniffing)
- Integrity: injection of spoofed packets
- Availability: delete legit packets (e.g., jamming)

Layers 1 & 2: General Threats?



Framing and transmission of a collection of bits into individual messages sent across a single "subnetwork" (one physical technology)

Encoding bits to send them over a single **physical link**

e.g. patterns of

voltage levels /

photon intensities /

RF modulation

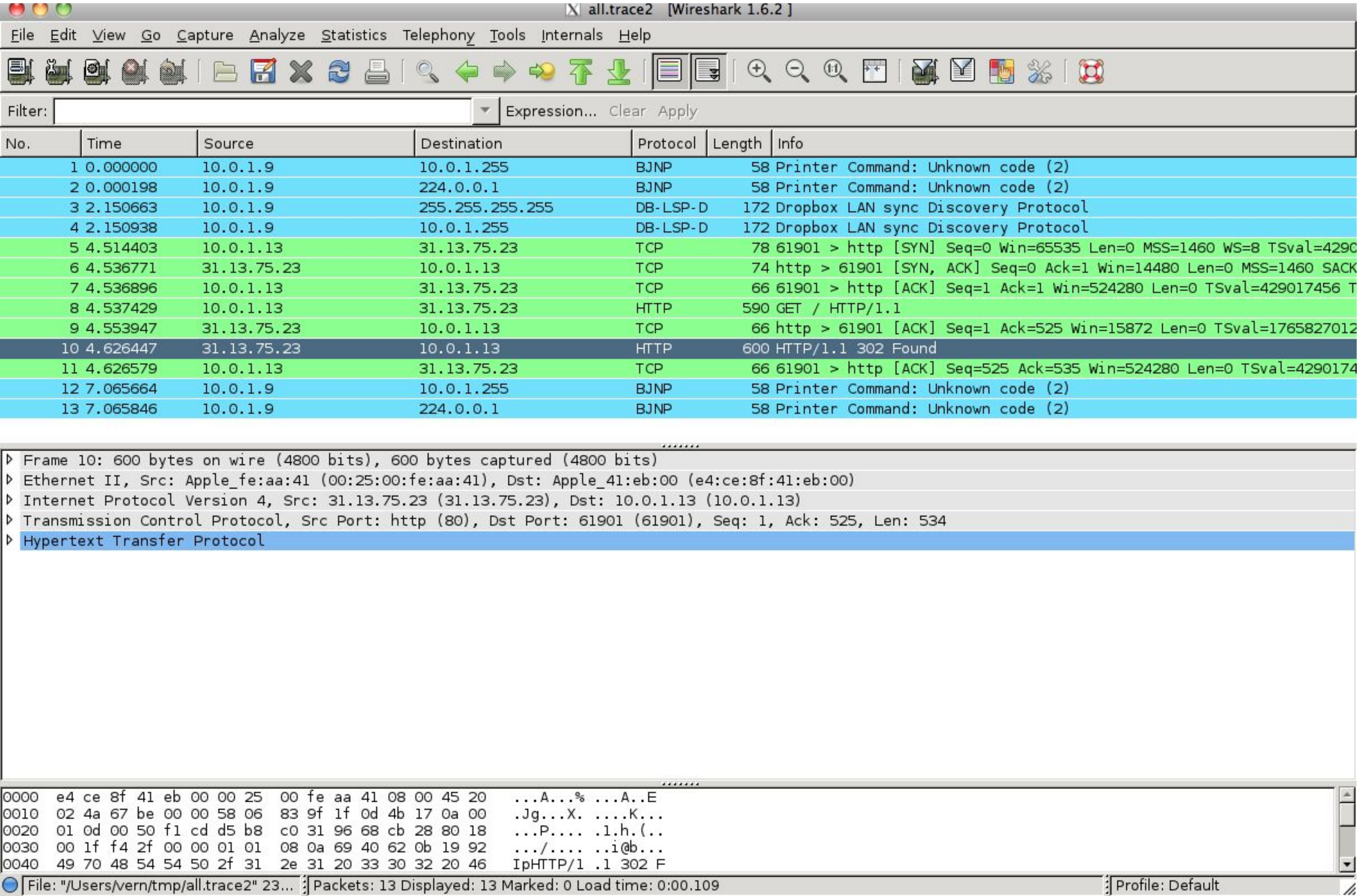
Eavesdropping

- For subnets using broadcast technologies (e.g., WiFi, some types of Ethernet), eavesdropping comes for "free"
  - Each attached system's NIC (= Network Interface Card) can capture any communication on the subnet
  - Some handy tools for doing so
    - tcpdump / windump (low-level ASCII printout)
    - Wireshark (GUI for displaying 800+ protocols)

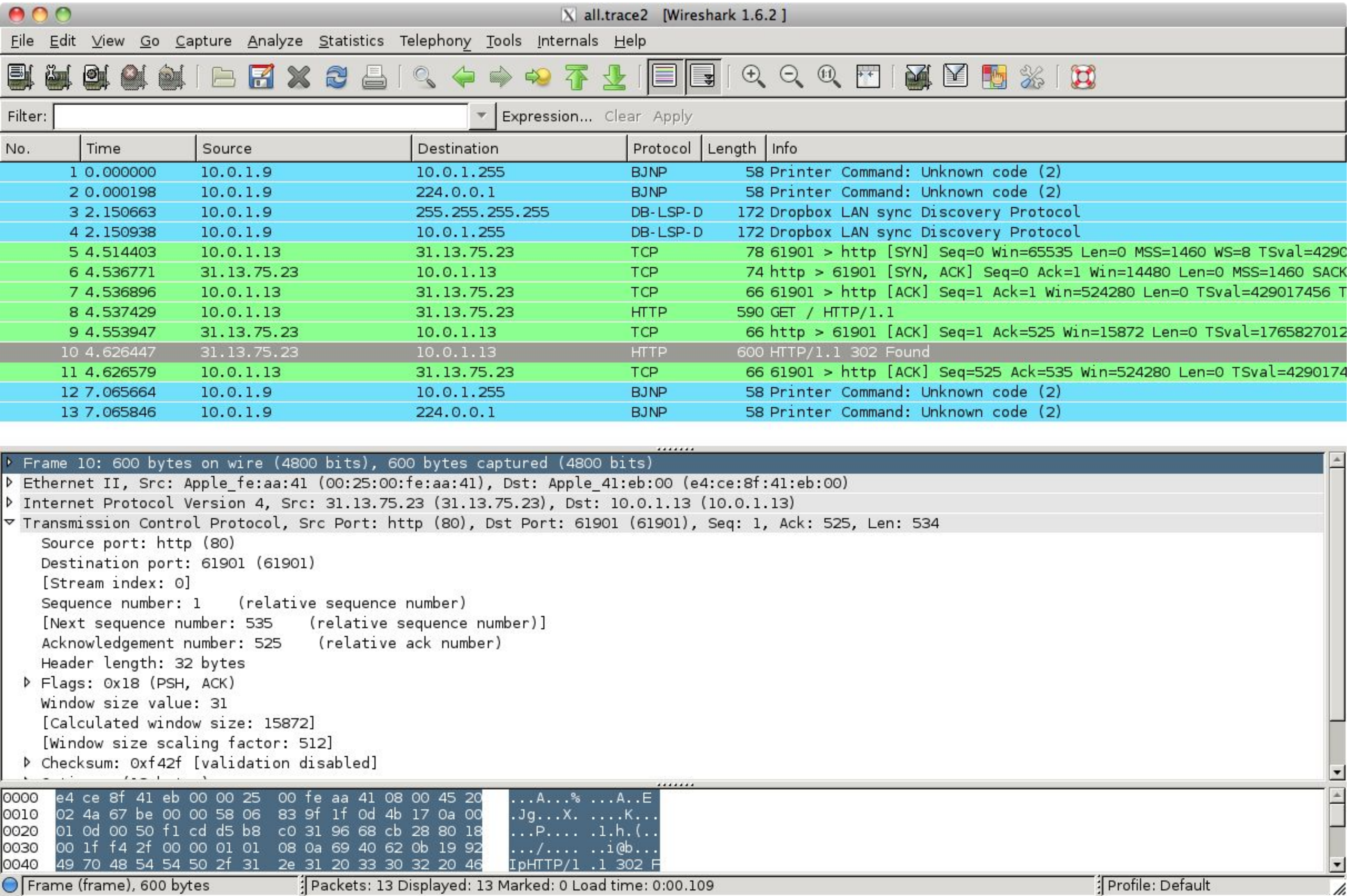
TCPDUMP: Packet Capture & ASCII Dumper

```
demo 2 % tcpdump -r all.trace2
reading from file all.trace2, link-type EN10MB (Ethernet)
21:39:37.772367 IP 10.0.1.9.60627 > 10.0.1.255.canon-bjnp2: UDP, length 16
21:39:37.772565 IP 10.0.1.9.62137 > all-systems.mcast.net.canon-bjnp2: UDP, length 16
21:39:39.923030 IP 10.0.1.9.17500 > broadcasthost.17500: UDP, length 130
21:39:39.923305 IP 10.0.1.9.17500 > 10.0.1.255.17500: UDP, length 130
21:39:42.286770 IP 10.0.1.13.61901 > star-01-02-pao1.facebook.com.http: Flags [S], seq 2
523449627, win 65535, options [mss 1460,nop,wscale 3,nop,nop,TS val 429017455 ecr 0,sack
OK,eol], length 0
21:39:42.309138 IP star-01-02-pao1.facebook.com.http > 10.0.1.13.61901: Flags [S.], seq
3585654832, ack 2523449628, win 14480, options [mss 1460,sackOK,TS val 1765826995 ecr 42
9017455,nop,wscale 9], length 0
21:39:42.309263 IP 10.0.1.13.61901 > star-01-02-pao1.facebook.com.http: Flags [.], ack 1
, win 65535, options [nop,nop,TS val 429017456 ecr 1765826995], length 0
21:39:42.309796 IP 10.0.1.13.61901 > star-01-02-pao1.facebook.com.http: Flags [P.], seq
1:525, ack 1, win 65535, options [nop,nop,TS val 429017456 ecr 1765826995], length 524
21:39:42.326314 IP star-01-02-pao1.facebook.com.http > 10.0.1.13.61901: Flags [.], ack 5
25, win 31, options [nop,nop,TS val 1765827012 ecr 429017456], length 0
21:39:42.398814 IP star-01-02-pao1.facebook.com.http > 10.0.1.13.61901: Flags [P.], seq
1:535, ack 525, win 31, options [nop,nop,TS val 1765827083 ecr 429017456], length 534
21:39:42.398946 IP 10.0.1.13.61901 > star-01-02-pao1.facebook.com.http: Flags [.], ack 5
35, win 65535, options [nop,nop,TS val 429017457 ecr 1765827083], length 0
21:39:44.838031 IP 10.0.1.9.54277 > 10.0.1.255.canon-bjnp2: UDP, length 16
21:39:44.838213 IP 10.0.1.9.62896 > all-systems.mcast.net.canon-bjnp2: UDP, length 16
```

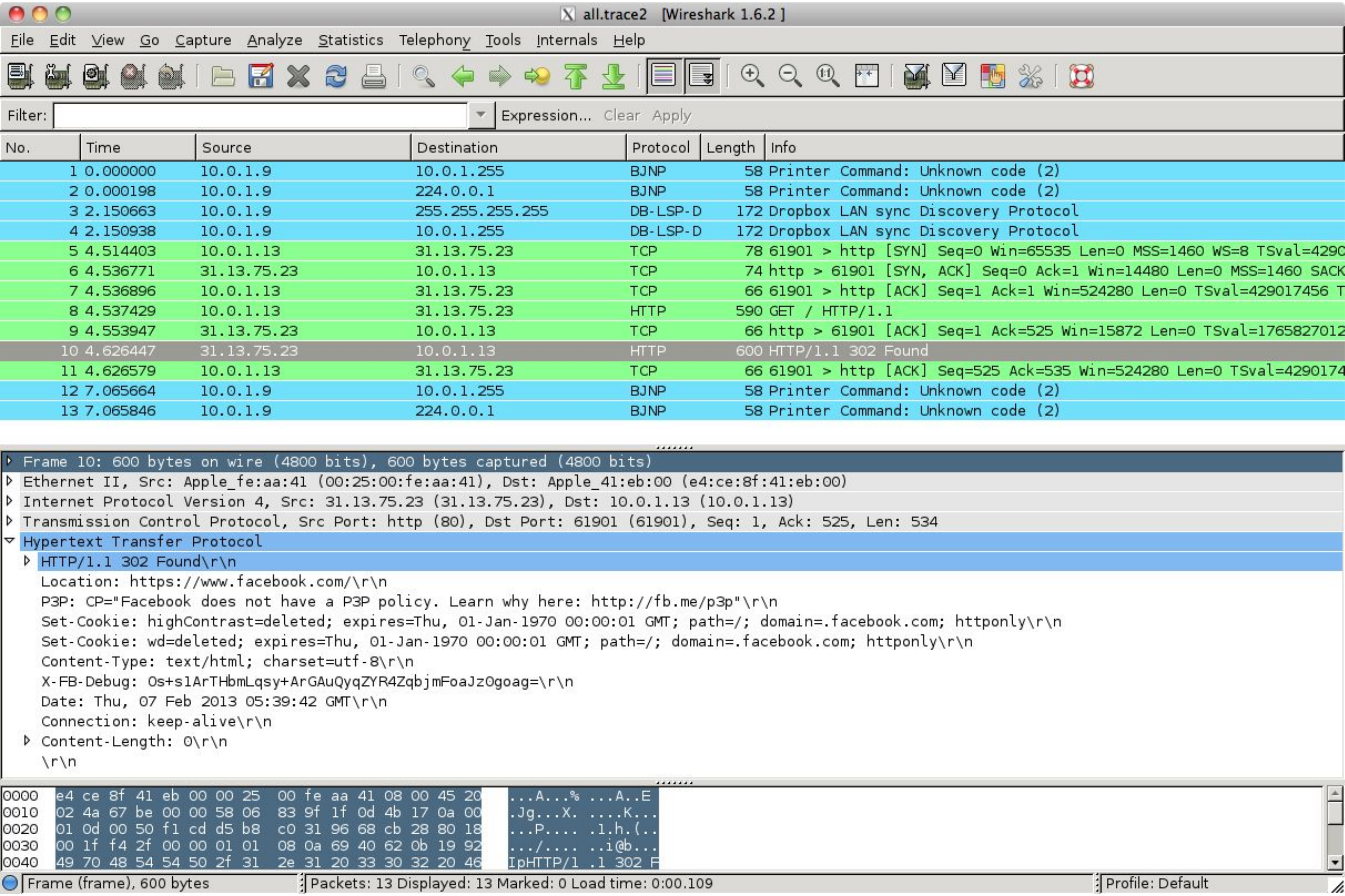
Wireshark: GUI for Packet Capture/Exam.



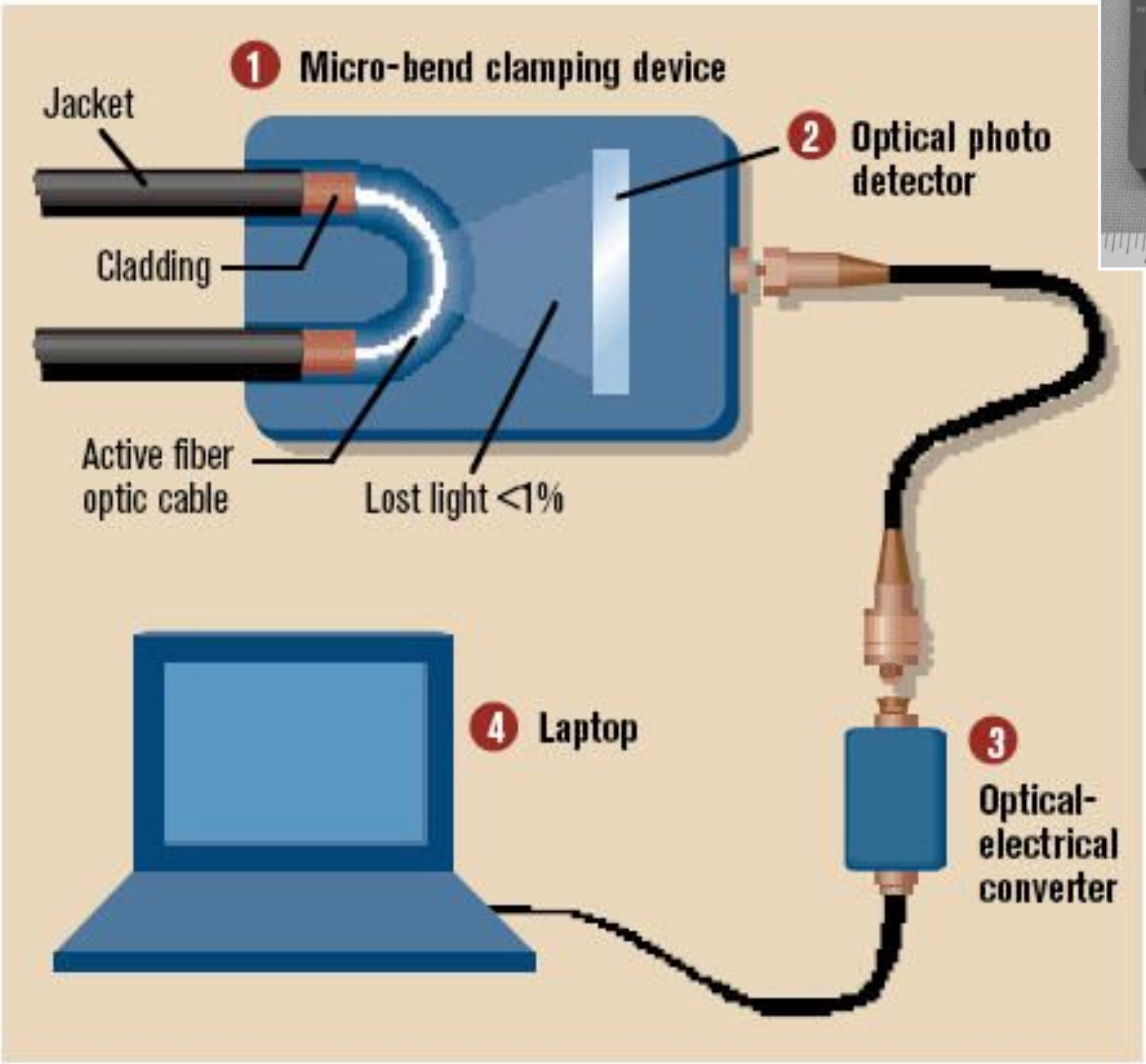
Wireshark: GUI for Packet Capture/Exam.

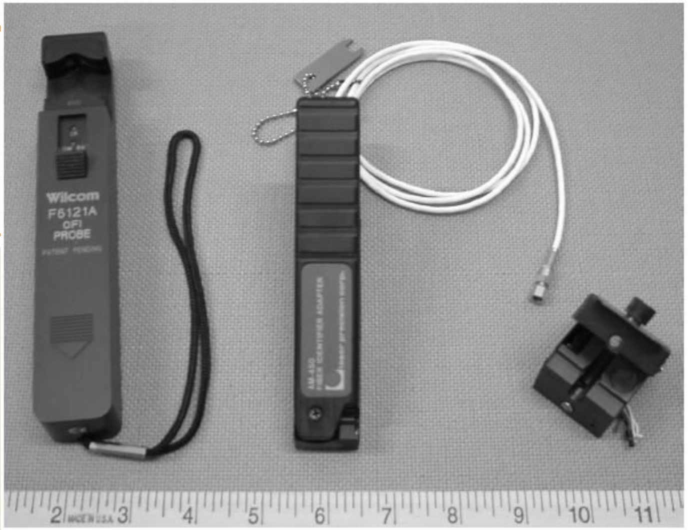


Wireshark: GUI for Packet Capture/Exam.



Stealing Photons





Operation Ivy Bells

By Matthew Carle Military.com

At the beginning of the 1970's, divers from the specially-equipped submarine, USS Halibut (SSN 587), left their decompression chamber to start a bold and dangerous mission, code named "Ivy Bells".

The Regulus guided missile submarine, USS Halibut (SSN 587) which carried out Operation Ivy Bells.

In an effort to alter the balance of Cold War, these men scoured the ocean floor for a five-inch diameter cable carry secret Soviet communications between military bases.

The divers found the cable and installed a 20-foot long listening device on the cable. designed to attach to the cable without piercing the casing, the device recorded all communications that occurred. If the cable malfunctioned and the Soviets raised it for repair, the bug, by design, would fall to the bottom of the ocean. Each month Navy divers retrieved the recordings and installed a new set of tapes.

Upon their return to the United States, intelligence agents from the NSA analyzed the recordings and tried to decipher any encrypted information. The Soviets apparently were confident in the security of their communications lines, as a surprising amount of sensitive information traveled through the lines without encryption.

prison. The original tap that was discovered by the Soviets is now on exhibit at the KGB museum in Moscow.

Link-Layer Threat: Disruption

- If attacker sees a packet he doesn't like, he can jam it (integrity)
- Attacker can also overwhelm link-layer signaling, e.g., jam WiFi's RF (denial-of-service)

Link-Layer Threat: Disruption

- If attacker sees a packet he doesn’t like, he can jam it (integrity)
- Attacker can also overwhelm link-layer signaling, e.g., jam WiFi’s RF (denial-of-service)
- There’s also the heavy-handed approach …

Sabotage attacks knock out phone service

Nanette Asimov, Ryan Kim,Kevin Fagan, Chronicle Staff Writers
Friday, April 10, 2009

PRINT E-MAIL SHARE COMMENTS (477) FONT | SIZE: − +

(04-10) 04:00 PDT SAN JOSE --

Police are hunting for vandals who chopped fiber-optic cables and killed landlines, cell phones and Internet service for tens of thousands of people in Santa Clara, Santa Cruz and San Benito counties on Thursday.

IMAGES



View More Images

MORE NEWS

- Toyota seeks damage control, in public and private 02.09.10
- Snow shuts down federal government, life goes on 02.09.10
- Iran boosts nuclear enrichment, drawing warnings 02.09.10

The sabotage essentially froze operations in parts of the three counties at hospitals, stores, banks and police and fire departments that rely on 911 calls, computerized medical records, ATMs and credit and debit cards.

The full extent of the havoc might not be known for days, emergency officials said as they finished repairing the damage late Thursday.

Whatever the final toll, one thing is certain: Whoever did this is in a world of trouble if he, she or they get caught.

"I pity the individuals who have done this," said San Jose Police Chief Rob Davis.

Ten fiber-optic cables carrying were cut at four locations in the predawn darkness. Residential and business customers quickly found that telephone service was perhaps more laced into their everyday needs than they thought. Suddenly they couldn't draw out money, send text messages, check e-mail or Web sites, call anyone for help, or even check on friends or relatives down the road.

Several people had to be driven to hospitals because they were unable to summon ambulances. Many businesses lapsed into idleness for hours, without the ability to contact associates or customers.

More than 50,000 landline customers lost service - some were residential, others were business lines that needed the connections for ATMs, Internet and bank card transactions. One line alone could affect hundreds of users.

UPDATE LOOK Vancouver's Venues Shine

NEWS | LOCAL BEAT

$250K Reward Out for Vandals Who Cut AT&T Lines

Local emergency declared during outage

By LORI PREUITT

Updated 2:12 PM PST, Fri, Apr 10, 2009

PRINT EMAIL SHARE BUZZ UP! TWITTER FACEBOOK

AT&T is now offering a $250,000 reward for information leading to the arrest of whoever is responsible for severing lines fiber optic cables in San Jose tha left much of the area without phone or cell service Thursday.

John Britton of AT&T said the reward is the largest ever offered by the company.

Link-Layer Threat: Spoofing

- Attacker can inject spoofed packets, and lie about the source address

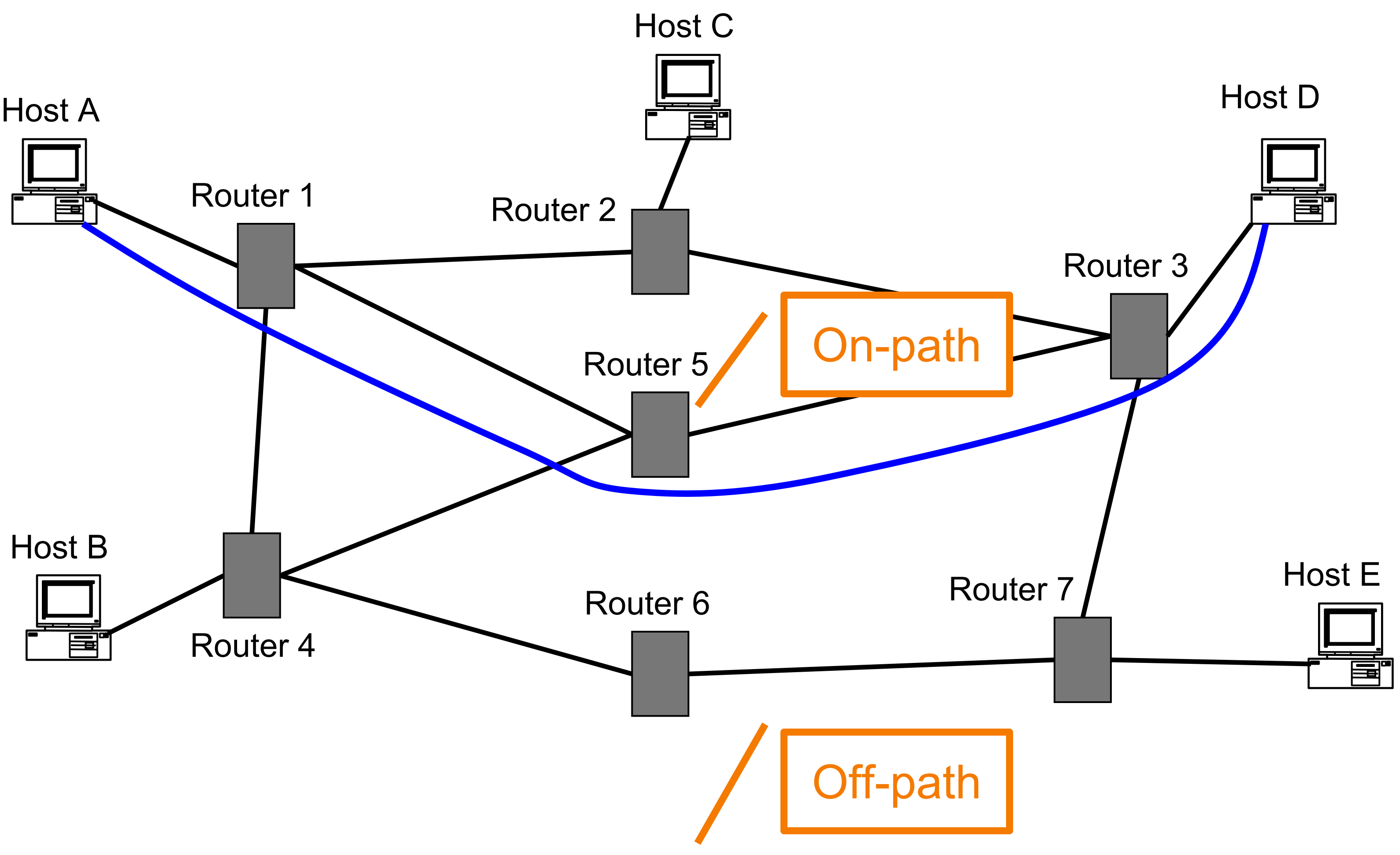
`D`	`C`	`Hello world!`

Physical/Link-Layer Threats: *Spoofing*

- With physical access to a local network, attacker can create any message they like
  - When with a bogus source address: *spoofing*
- When using a typical computer, may require root/administrator to have full freedom
- Particularly powerful when combined with *eavesdropping*
  - Because attacker can understand exact state of victim's communication and craft their spoofed traffic to match it
  - Spoofing w/o eavesdropping = *blind spoofing*

On-path vs Off-path Spoofing

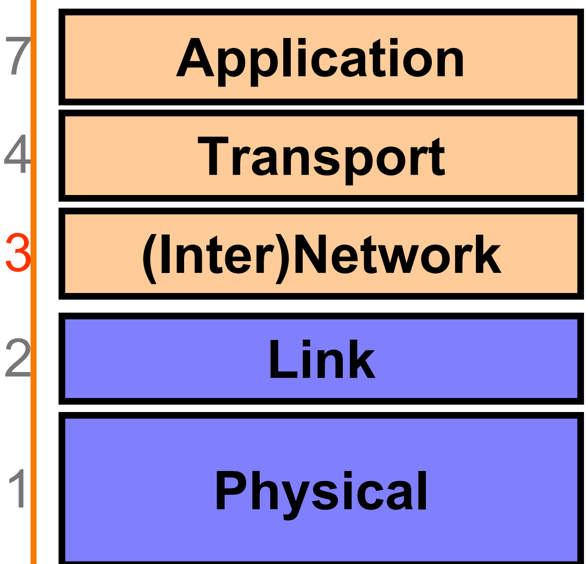
Host A communicates with Host D



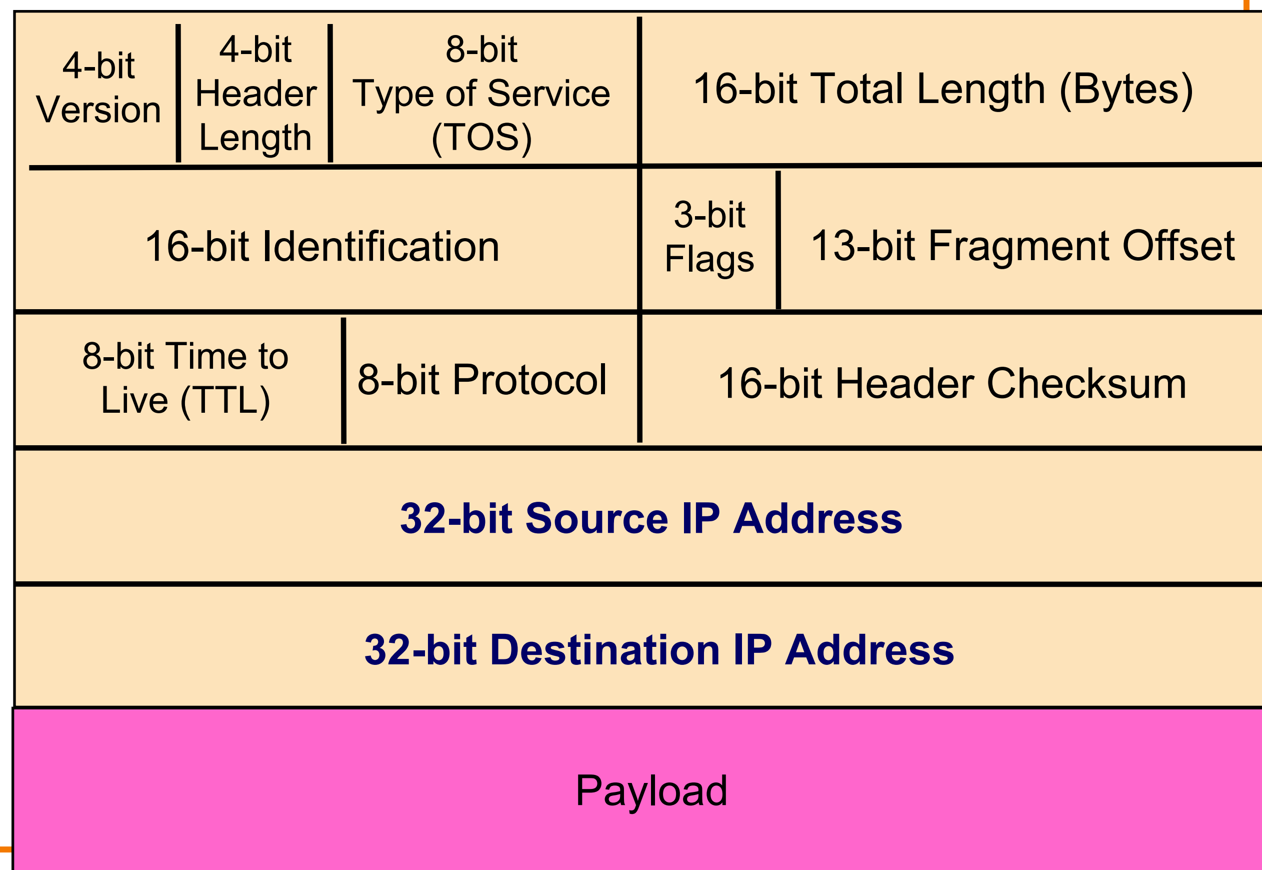
Spoofing on the Internet

- On-path attackers can see victim's traffic ⇒ spoofing is easy
- Off-path attackers can't see victim's traffic
  - They have to resort to blind spoofing
  - Often must guess/infer header values to succeed
    - We then care about work factor: how hard is this
  - But sometimes they can just **brute force**
    - E.g., 16-bit value: just try all 65,536 possibilities!
- When we say an attacker "can spoof", we usually mean "w/ reasonable chance of success"

Layer 3: General Threats?



Bridges multiple "subnets" to provide *end-to-end* internet connectivity between nodes



IP = Internet Protocol

IP-Layer Threats

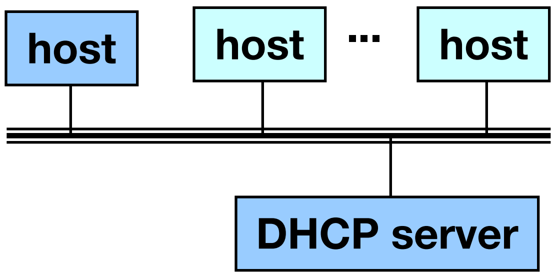
- Can set arbitrary source address
  - "Spoofing" - receiver has no idea who you are
  - Could be blind, or could be coupled w/ sniffing
  - Note: many attacks require two-way communication
    - So successful off-path/blind spoofing might not suffice
- Can set arbitrary destination address
  - Enables "scanning" – brute force searching for hosts
- Can *send like crazy* (flooding)
  - IP has no general mechanism for tracking overuse
  - IP has no general mechanism for tracking consent
  - Very hard to tell where a spoofed flood comes from!
- **If** attacker can manipulate routing, can bring traffic to themselves for *eavesdropping* (not easy)

DNS Service

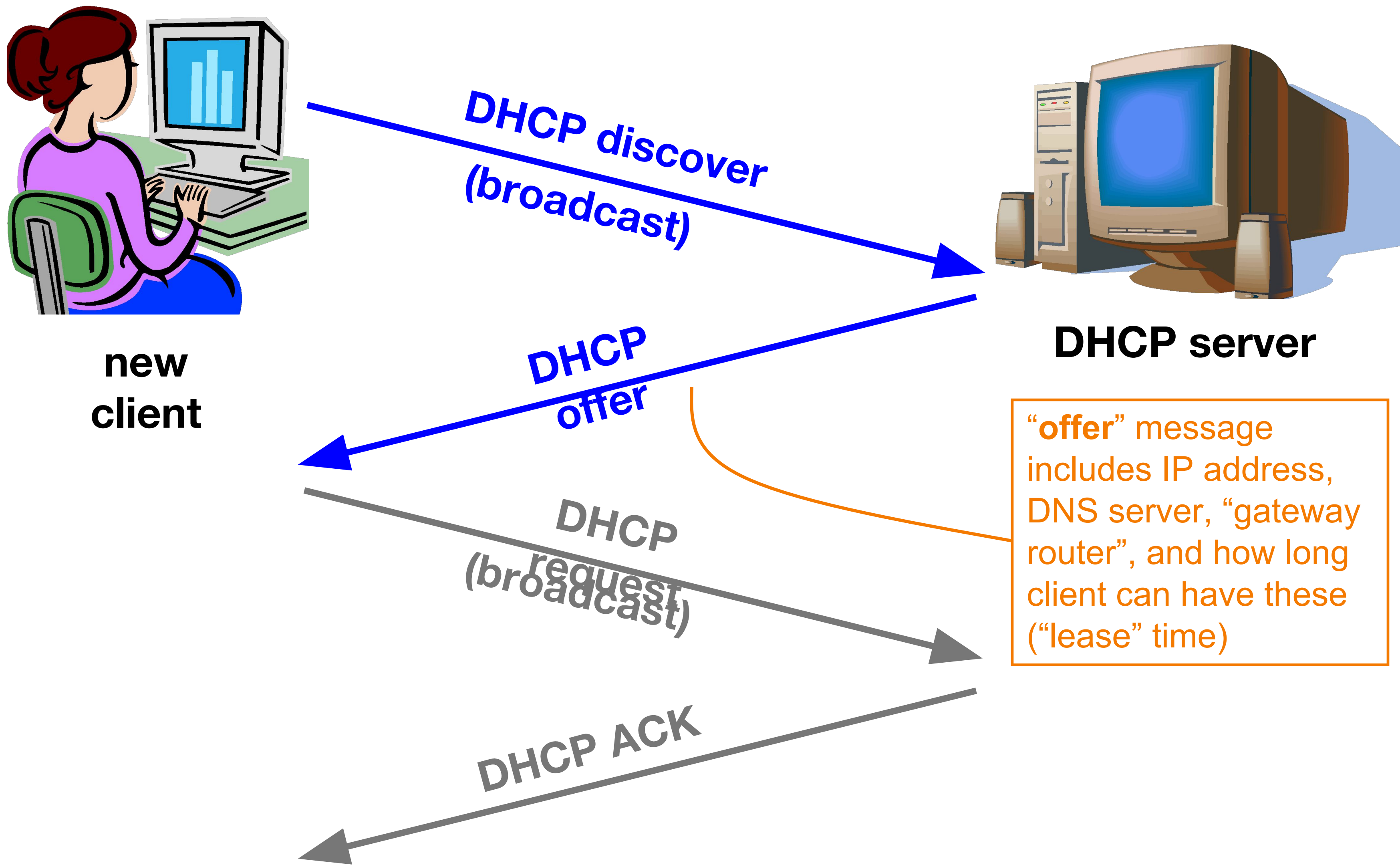
- Runs Domain Name Servers
- Translates domain names google.com to IP addresses
- When user browser wants to contact google.com, it first contacts a DNS to find out the IP address for google.com and then sends a packet to that IP address
- More in future lectures..

LAN Bootstrapping: DHCP

- New host doesn't have an IP address yet
  - So, host doesn't know what source address to use
- Host doesn't know *who to ask* for an IP address
  - So, host doesn't know what destination address to use
- Solution: shout to "**discover**" server that can help
  - Broadcast a server-discovery message (layer 2)
  - Server(s) sends a reply offering an address

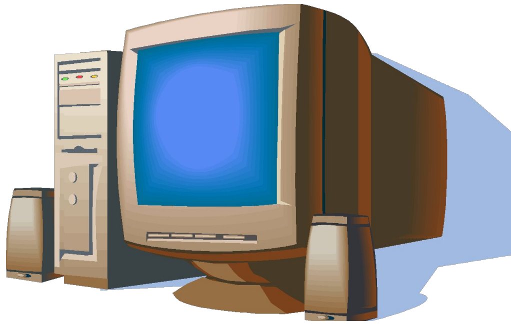


Dynamic Host Configuration Protocol



Dynamic Host Configuration Protocol

new
client



DHCP server

DHCP discover
(broadcast)

DHCP
offer

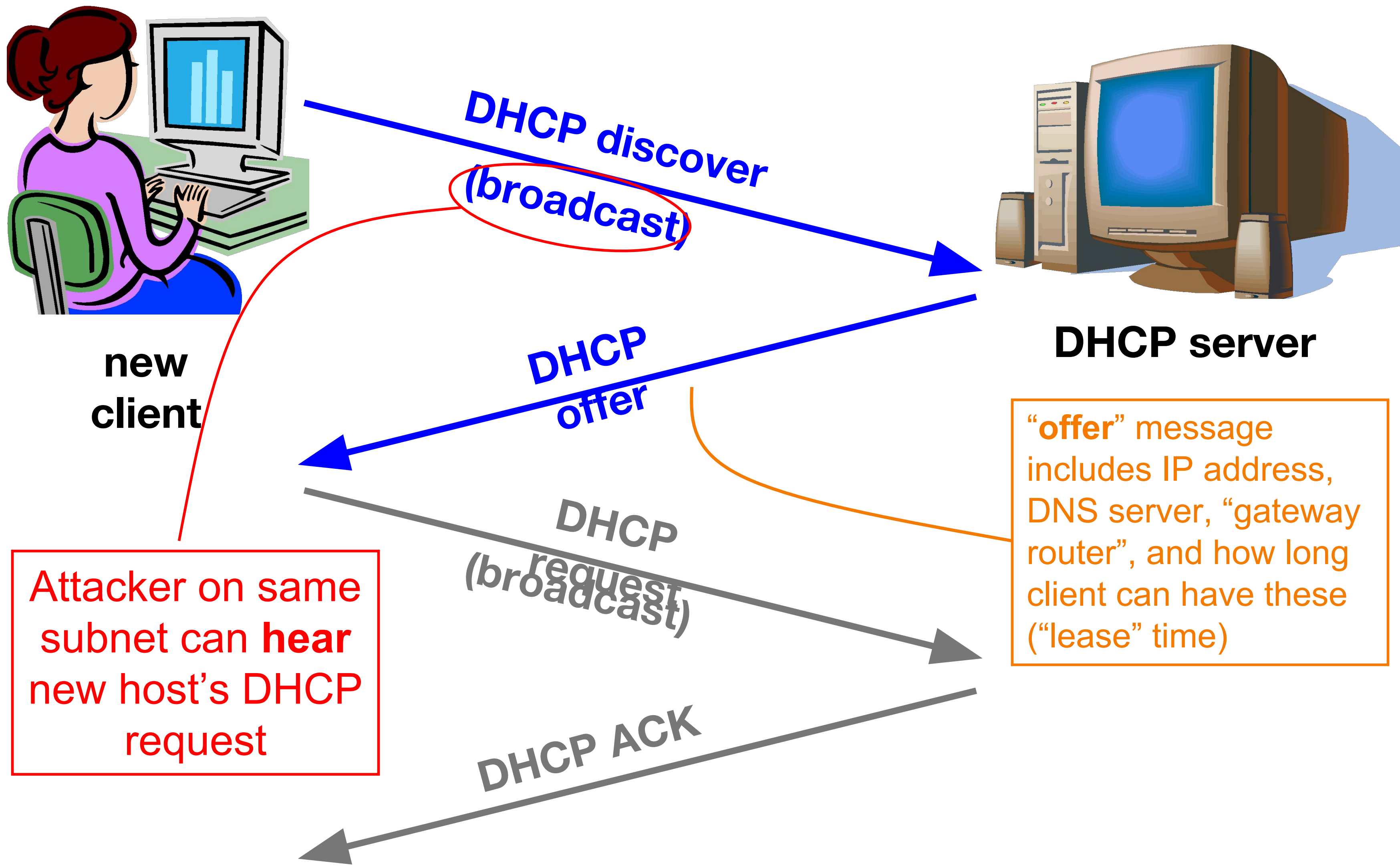
“**offer**” message includes IP address, DNS server, “gateway router”, and how long client can have these (“lease” time)

DHCP
request
(broadcast)

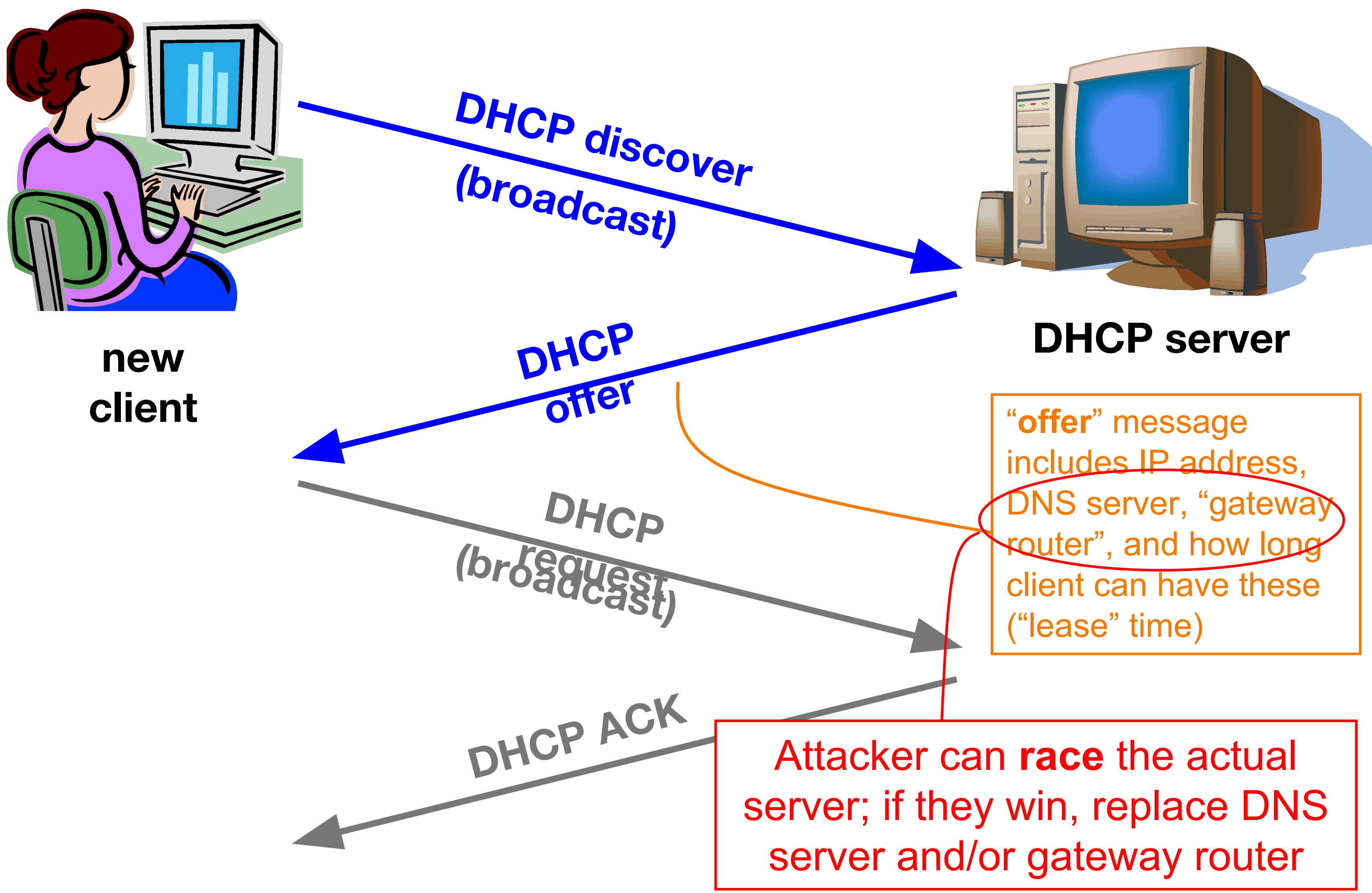
DHCP ACK

Threats?

Dynamic Host Configuration Protocol



Dynamic Host Configuration Protocol

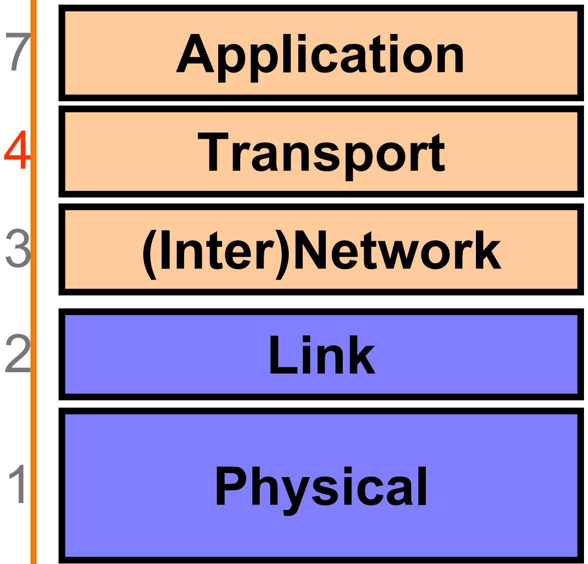


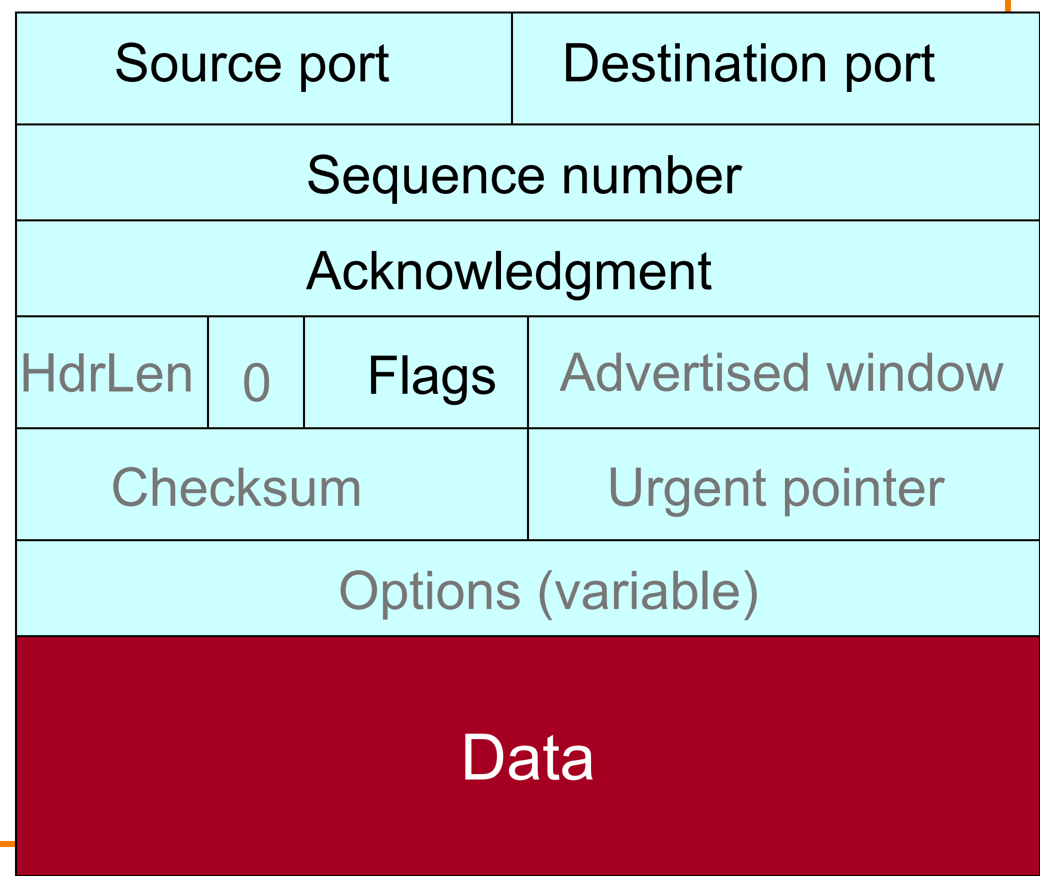
DHCP Threats

- Substitute a fake DNS server
  - Redirect any of a host's lookups to a machine of attacker's choice
- Substitute a fake gateway router
  - Intercept all of a host's off-subnet traffic
    - (even if not preceded by a DNS lookup)
  - Relay contents back and forth between host and remote server and modify however attacker chooses
- An invisible *Man In The Middle* (MITM)
  - Victim host has no way of knowing it's happening
    - (Can't necessarily alarm on peculiarity of receiving multiple DHCP replies, since that can happen benignly)
- How can we fix this?

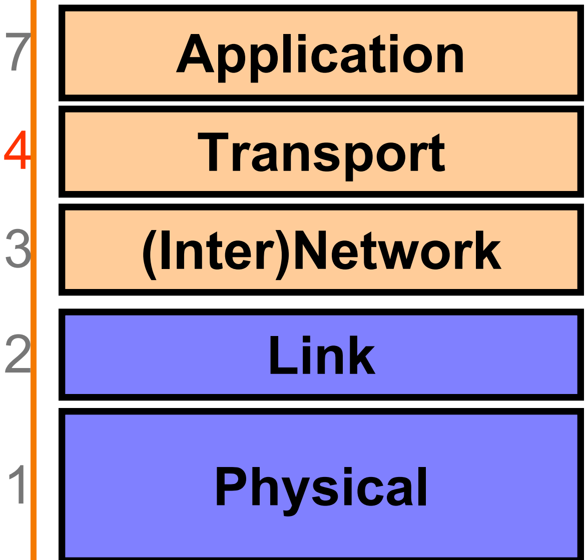
Hard

TCP

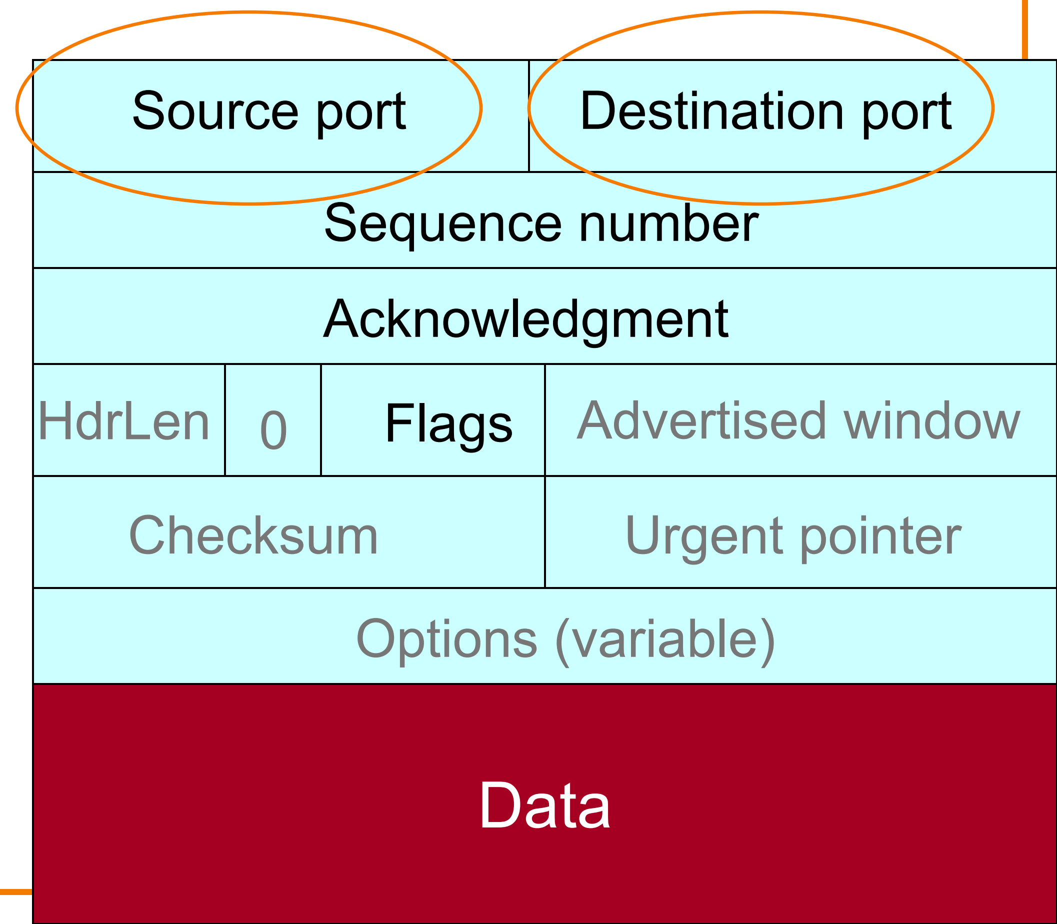

7 Application
4 Transport
3 (Inter)Network
2 Link
1 Physical


Source port
Destination port
Sequence number
Acknowledgment
HdrLen
0
Flags
Advertised window
Checksum
Urgent pointer
Options (variable)
Data

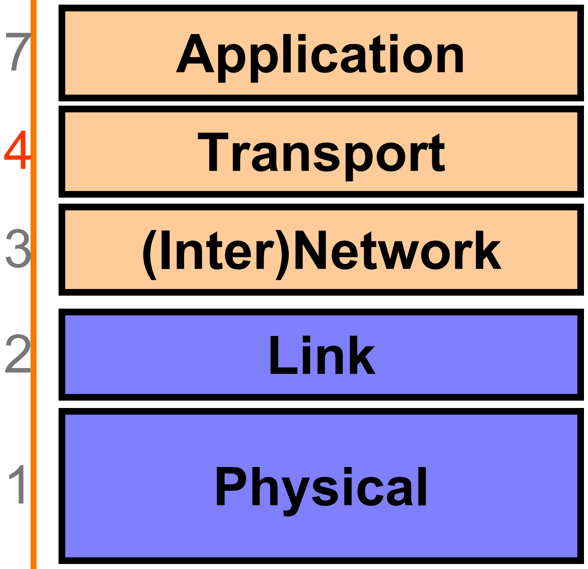
TCP



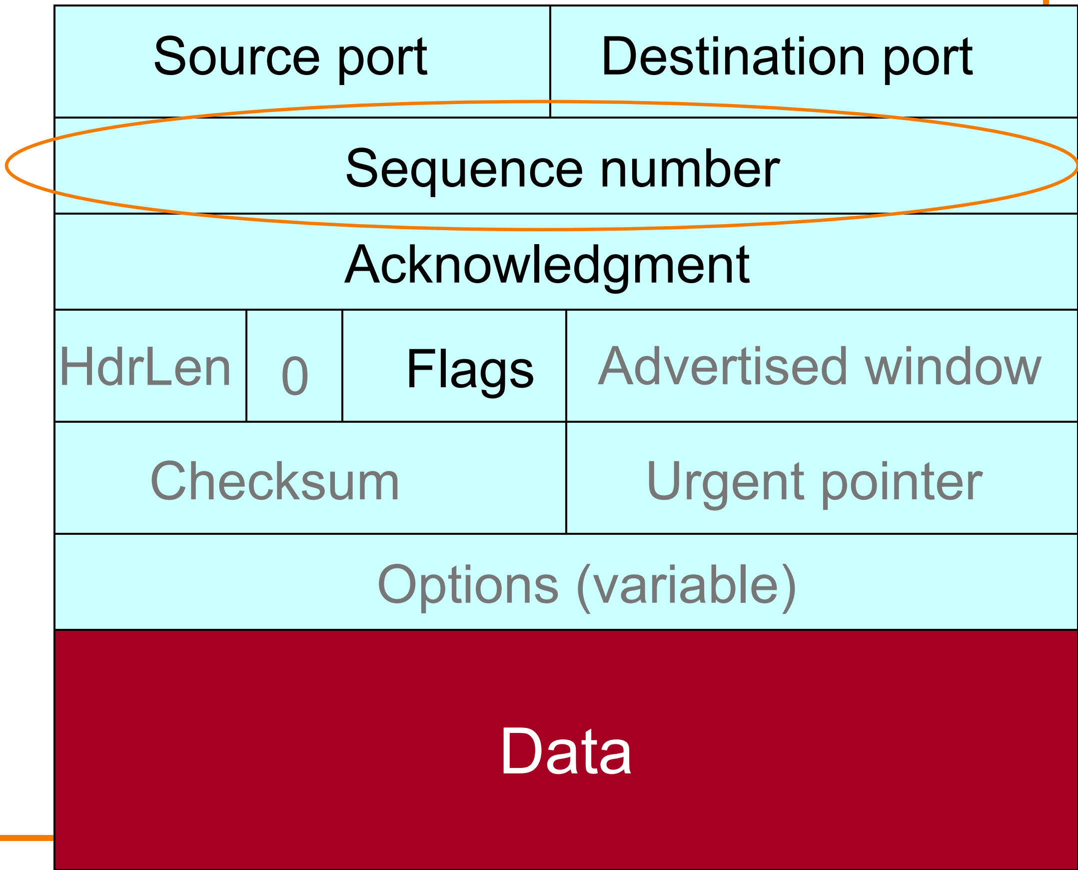
These plus IP addresses define a given connection



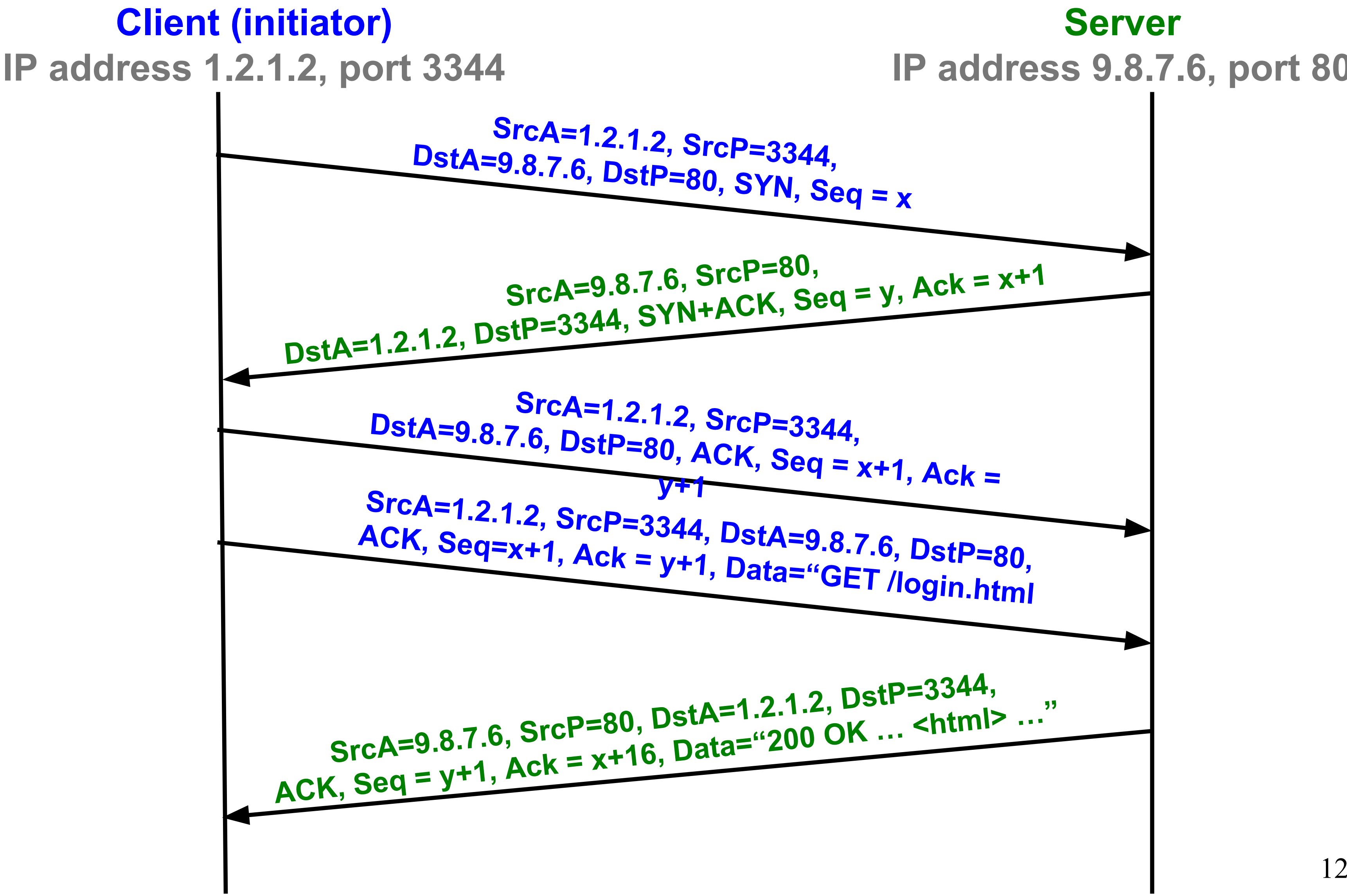
TCP



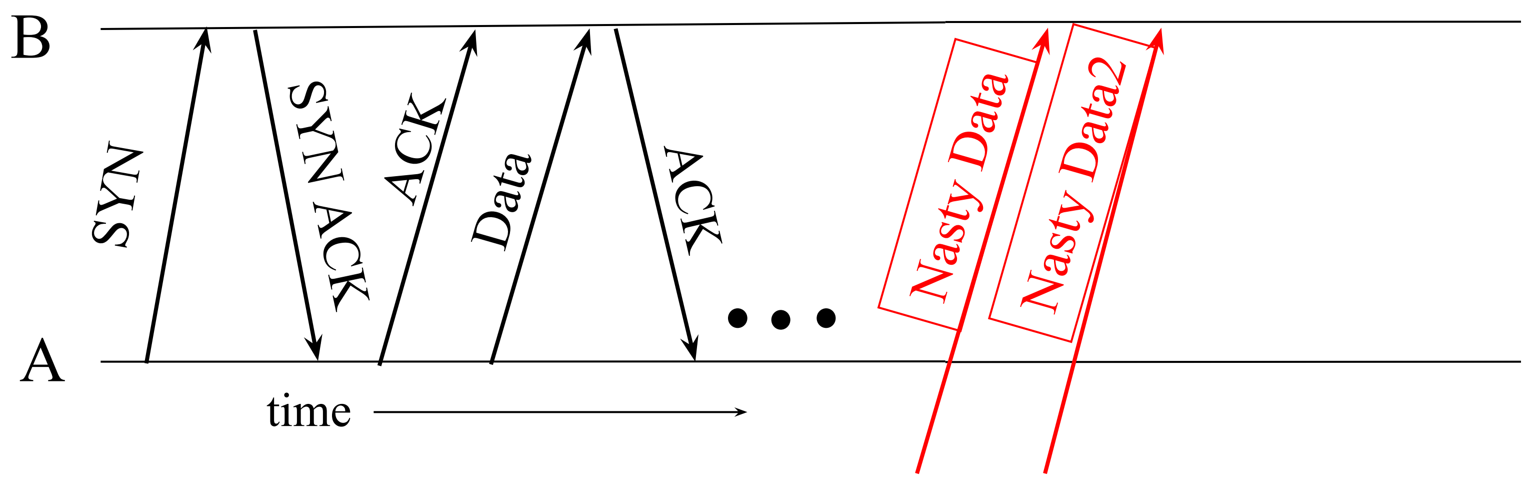
Defines where this packet fits within the sender's bytestream



TCP Conn. Setup & Data Exchange

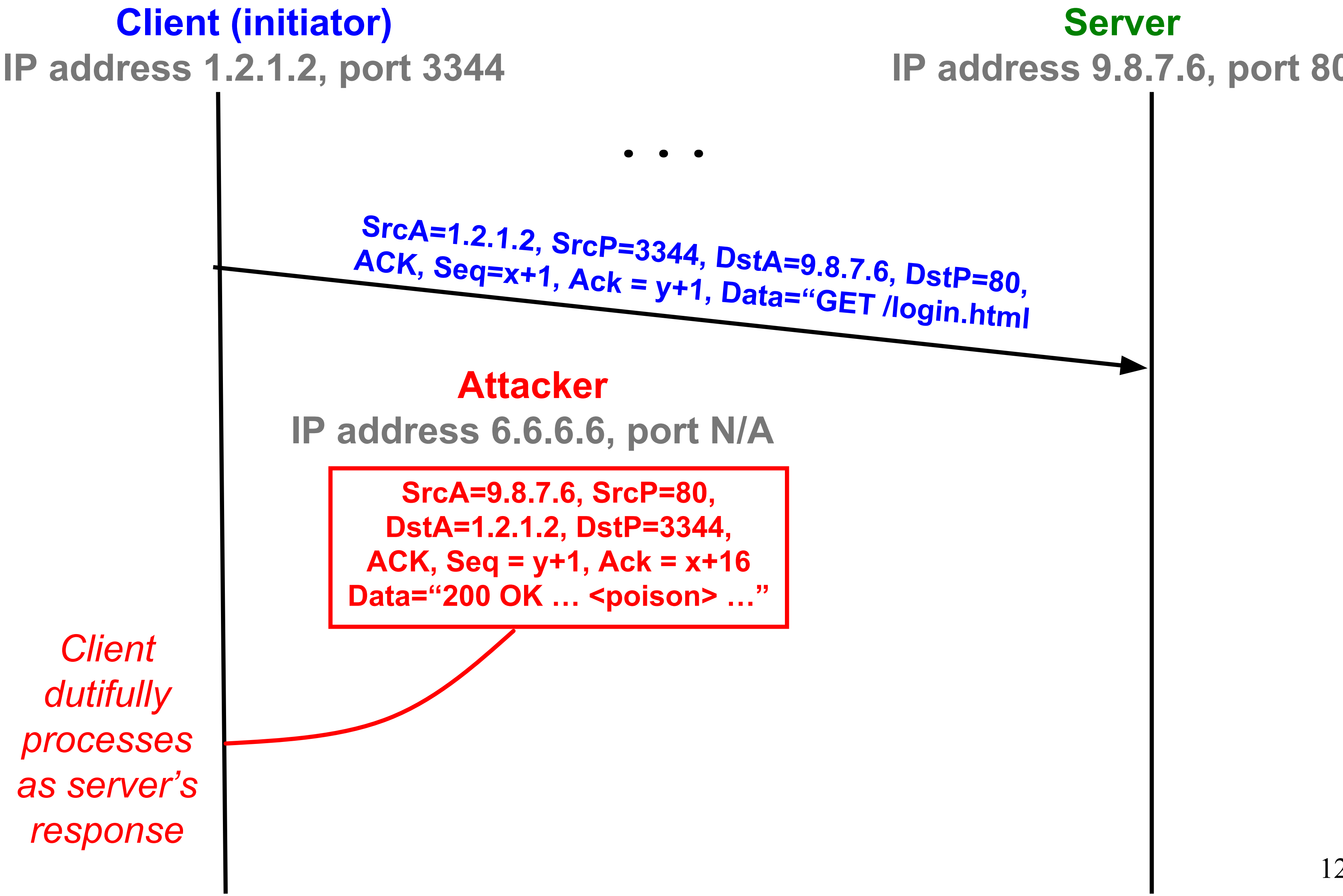


TCP Threat: Data Injection

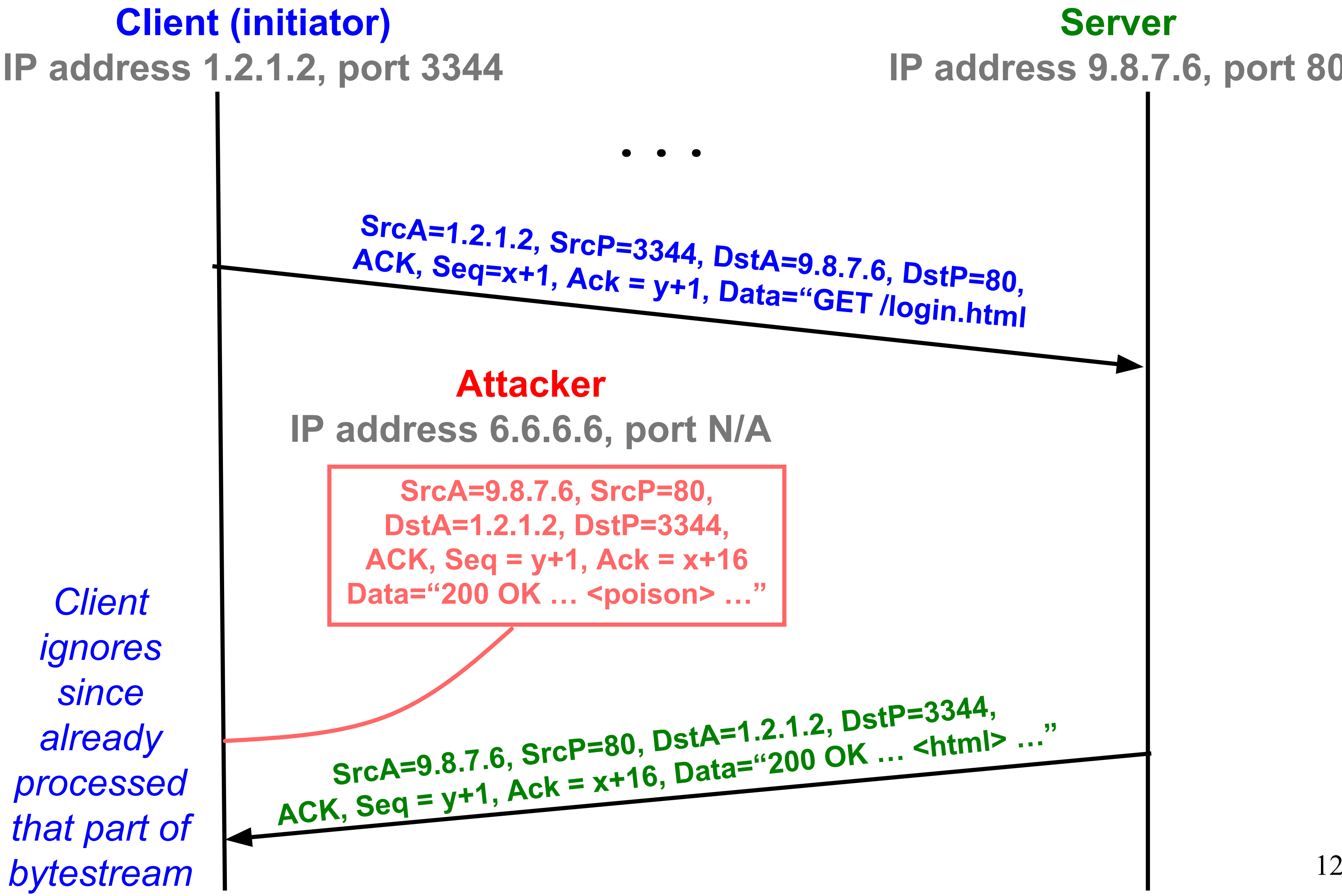


- If attacker knows ports & sequence numbers (e.g., on-path attacker), attacker can inject data into any TCP connection
  - Receiver B is *none the wiser!*
- Termed TCP connection hijacking (or "*session hijacking*")
  - In general means to take over an already-established connection!
- We are toasted if an attacker can see our TCP traffic!
  - Because then they immediately know the port & sequence numbers

TCP *Data Injection*



TCP *Data Injection*



TCP Threat: Disruption

- Q: Is it possible for an on-path attacker to shut down a TCP connection if they can see our traffic?
- A: YES: they can infer the port and sequence numbers – they can insert fake data, too! (Great Firewall of China)

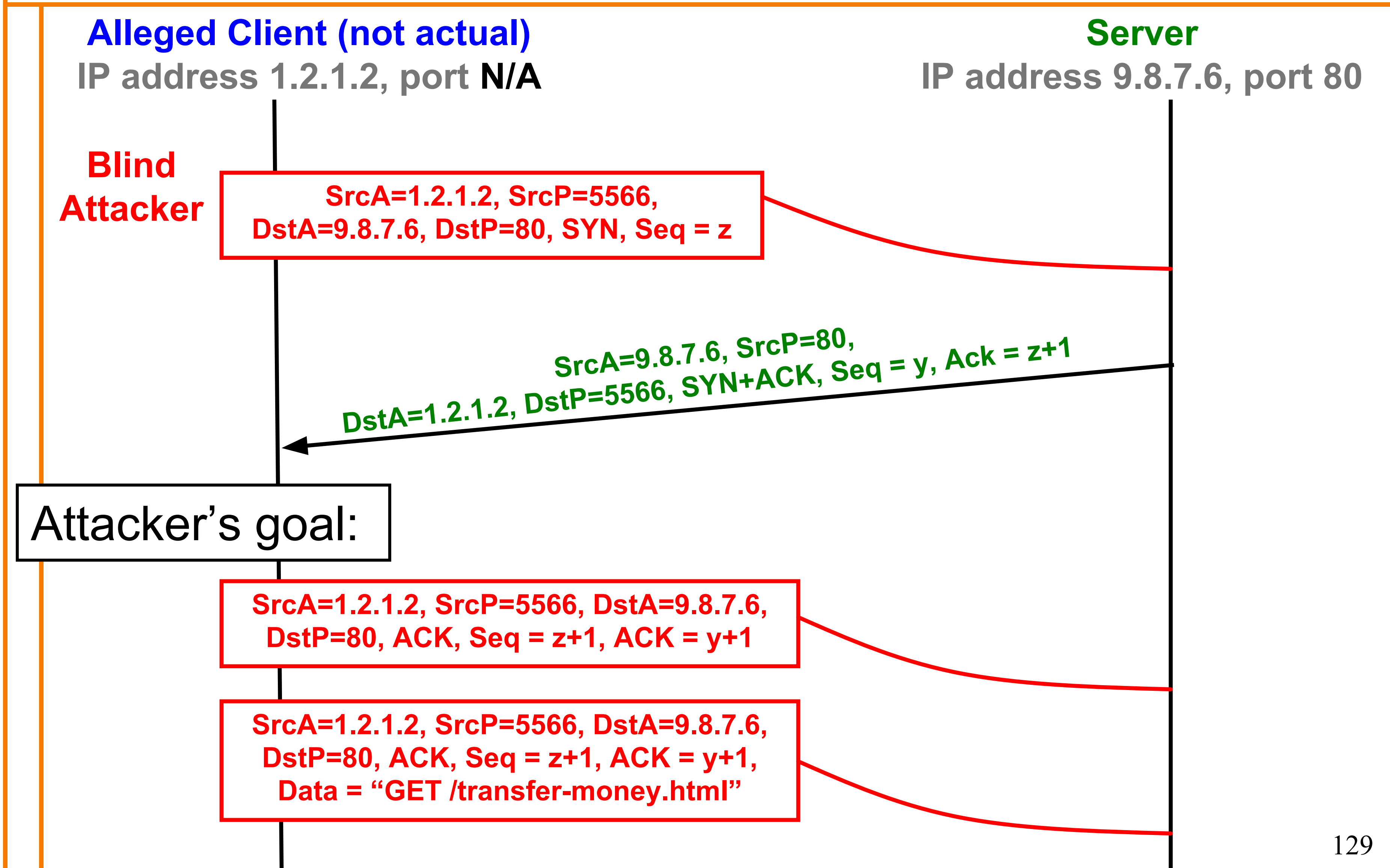
TCP Threat: Blind Hijacking

- Q: Is it possible for an off-path attacker to inject into a TCP connection even if they can't see our traffic?
- A: YES: if somehow they can infer or guess the port and sequence numbers

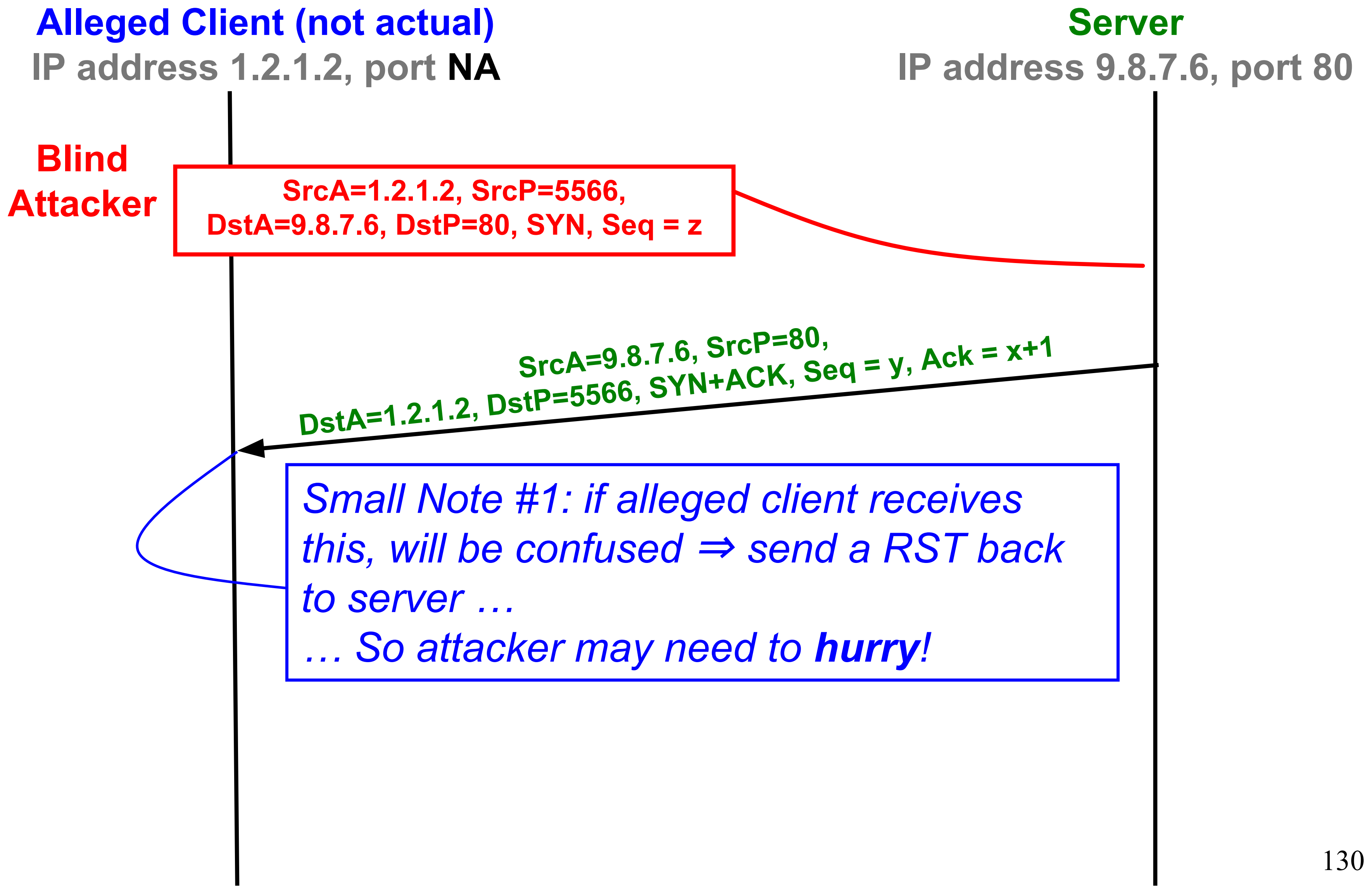
TCP Threat: Blind Spoofing

- Q: Is it possible for an off-path attacker to create a fake TCP connection, even if they can't see responses?
- A: YES: if somehow they can infer or guess the TCP initial sequence numbers
- Why would an attacker want to do this?
  - Perhaps to leverage a server's trust of a given client as identified by its IP address
  - Perhaps to frame a given client so the attacker's actions during the connections can't be traced back to the attacker

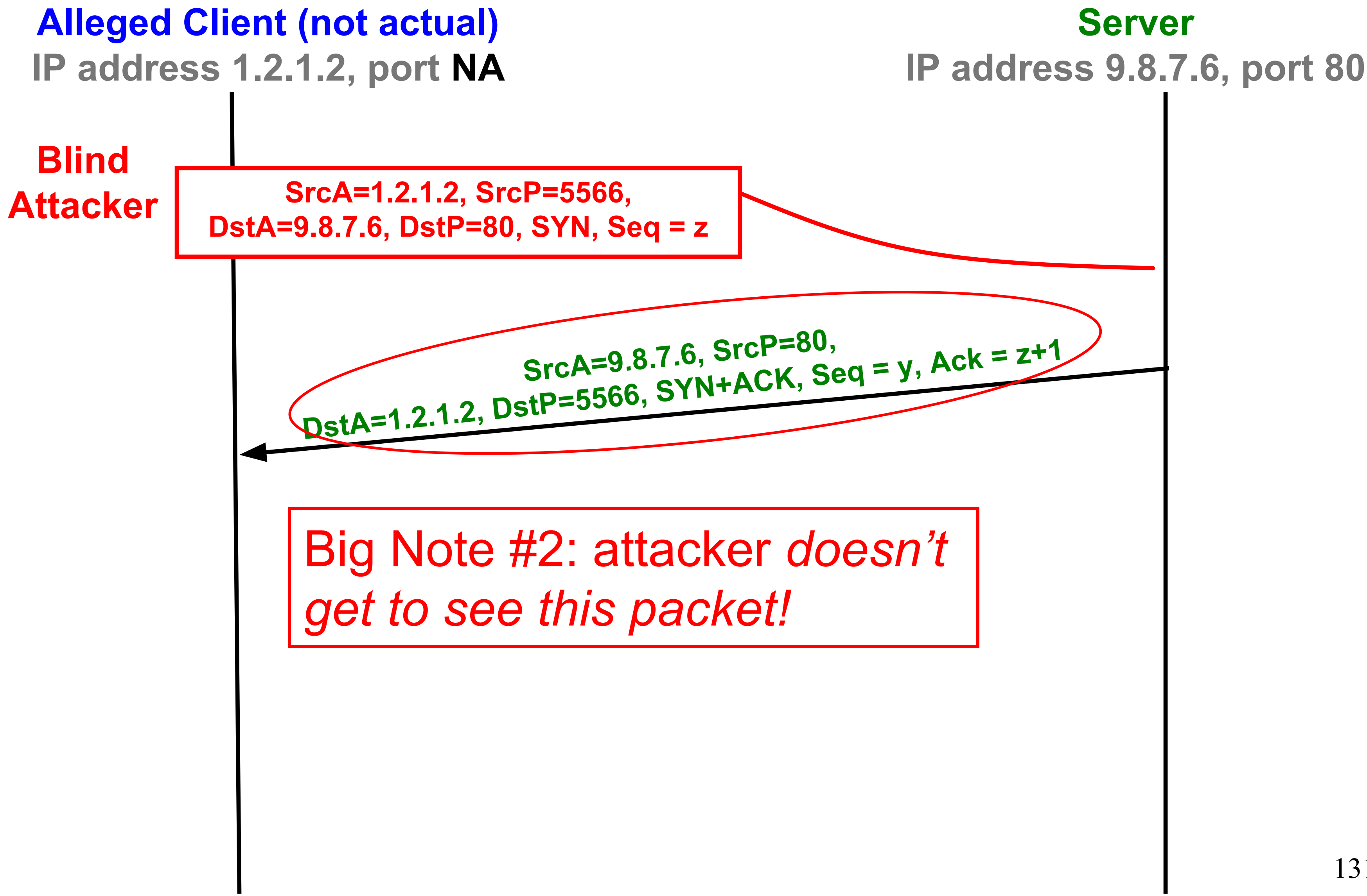
Blind Spoofing on TCP Handshake



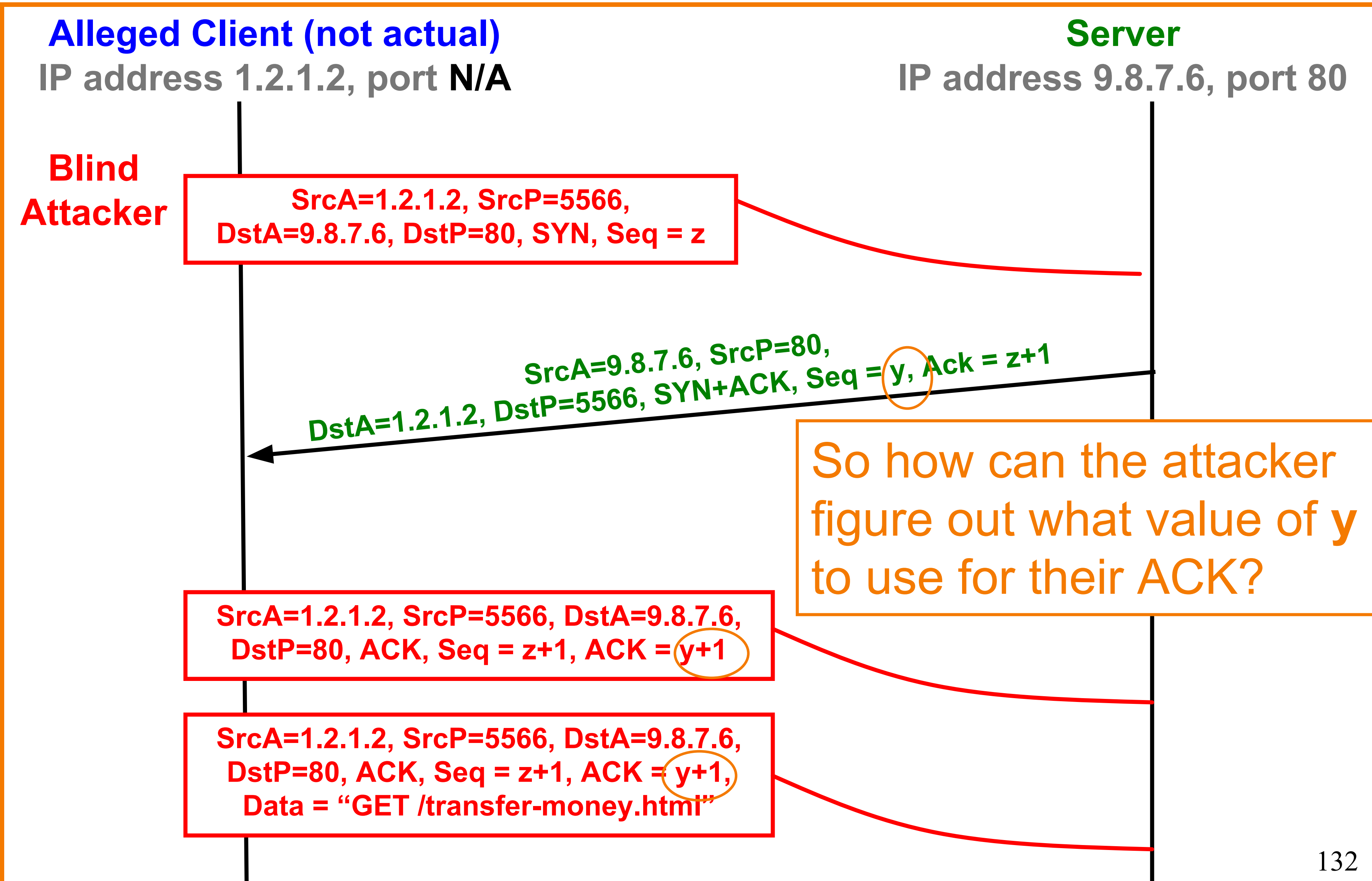
Blind Spoofing on TCP Handshake



Blind Spoofing on TCP Handshake

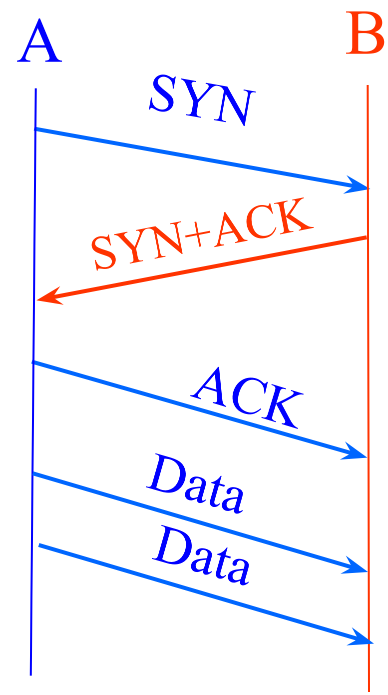


Blind Spoofing on TCP Handshake



So how can the attacker figure out what value of **y** to use for their ACK?

Reminder: Establishing a TCP Connection



Each host tells its *Initial Sequence Number* (ISN) to the other host.

(Spec says to pick based on local clock)

Hmm, any way for the attacker to know ***this?***

How Do We Fix This?

Use a (Pseudo)-*Random* ISN

Sure – make a non-spoofed connection *first*, and see what server used for ISN y then!

Summary of TCP Security Issues

- An attacker who can observe your TCP connection can manipulate it:
  - Forcefully **terminate** by forging a RST packet
  - **Inject** (*spoof*) data into either direction by forging data packets
  - Works because they can include in their spoofed traffic the correct sequence numbers (both directions) and TCP ports
  - *Remains a major threat today*

Summary of TCP Security Issues

- An attacker who can observe your TCP connection can manipulate it:
  - Forcefully **terminate** by forging a RST packet
  - **Inject** (*spoof*) data into either direction by forging data packets
  - Works because they can include in their spoofed traffic the correct sequence numbers (both directions) and TCP ports
  - *Remains a major threat today*
- If attacker could predict the ISN chosen by a server, could "blind spoof" a connection to the server
  - Makes it appear that host ABC has connected, and has sent data of the attacker's choosing, when in fact it hasn't
  - *Undermines any security based on trusting ABC's IP address*
  - Allows attacker to "frame" ABC or otherwise avoid detection
  - **Fixed** (mostly) today by choosing **random** ISNs

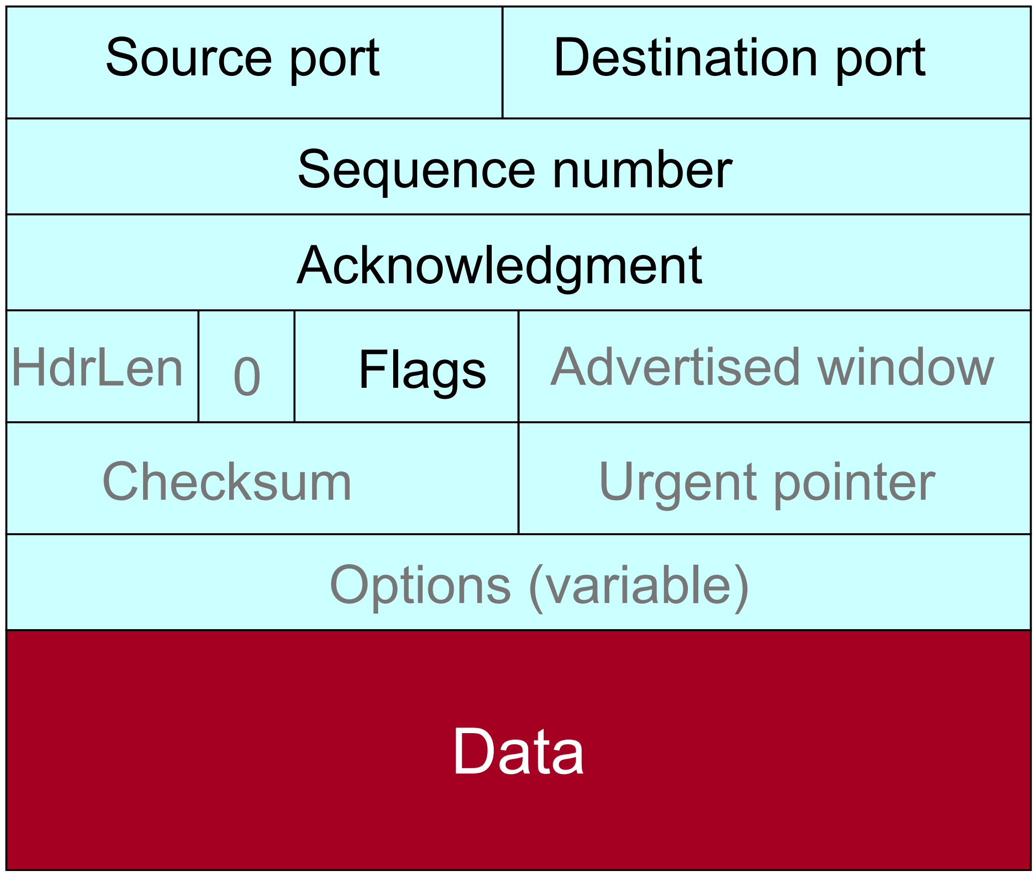
Summary of IP security

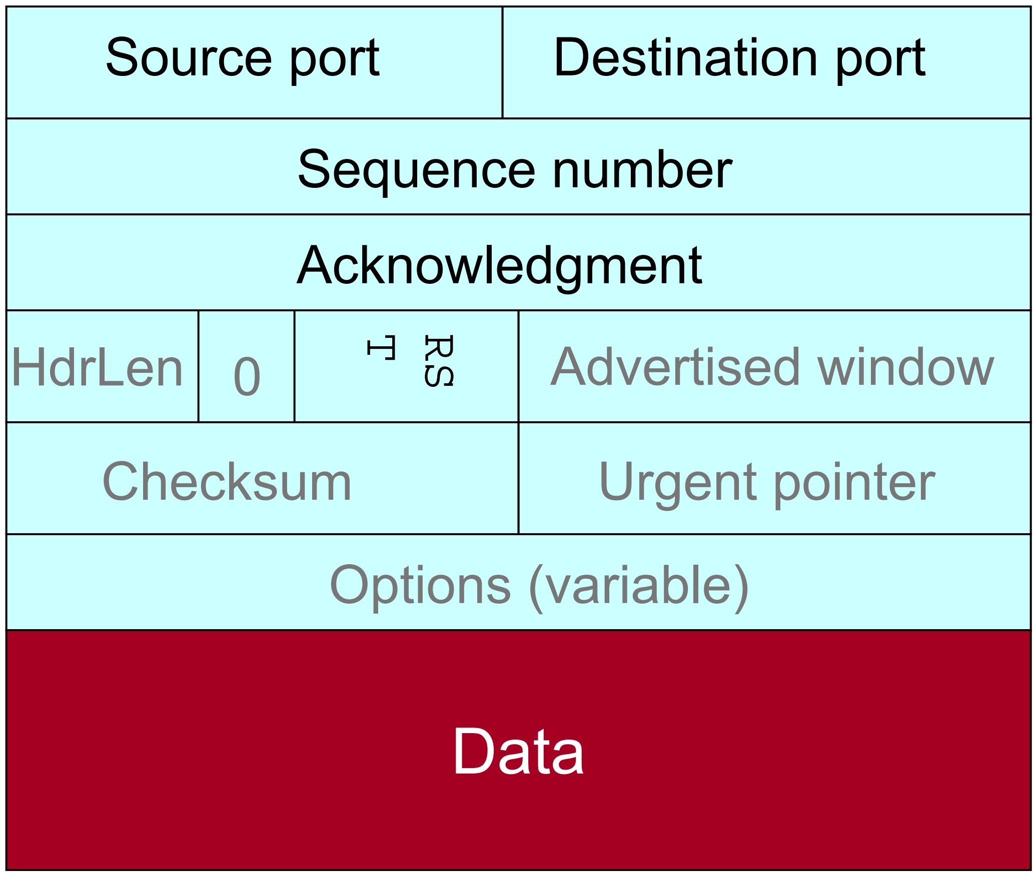
- No security against on-path attackers
  - Can sniff, inject packets, mount TCP spoofing, TCP hijacking, man-in-the-middle attacks
  - Typical example: wireless networks, malicious network operator
- More security against off-path attackers
  - TCP is more secure than UDP and IP

Extra Material

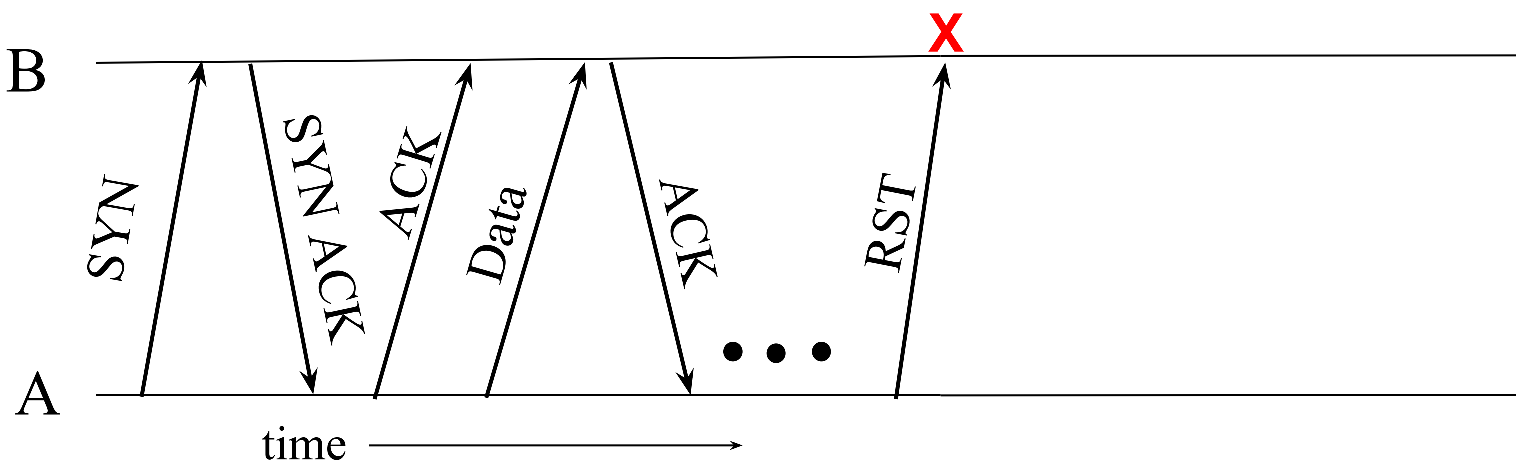
TCP Threat: Disruption

- Normally, TCP finishes ("closes") a connection by each side sending a `FIN` control message
  - Reliably delivered, since other side must ack
- But: if a TCP endpoint finds unable to continue (process dies; info from other "peer" is inconsistent), it abruptly terminates by sending a **RST** control message
  - Unilateral
  - Takes effect immediately (no ack needed)
  - Only accepted by peer if has correct* sequence number


Source port
Destination port
Sequence number
Acknowledgment
HdrLen
0
Flags
Advertised window
Checksum
Urgent pointer
Options (variable)
Data


Source port
Destination port
Sequence number
Acknowledgment
HdrLen
0
Advertised window
Checksum
Urgent pointer
Options (variable)
Data

Abrupt Termination

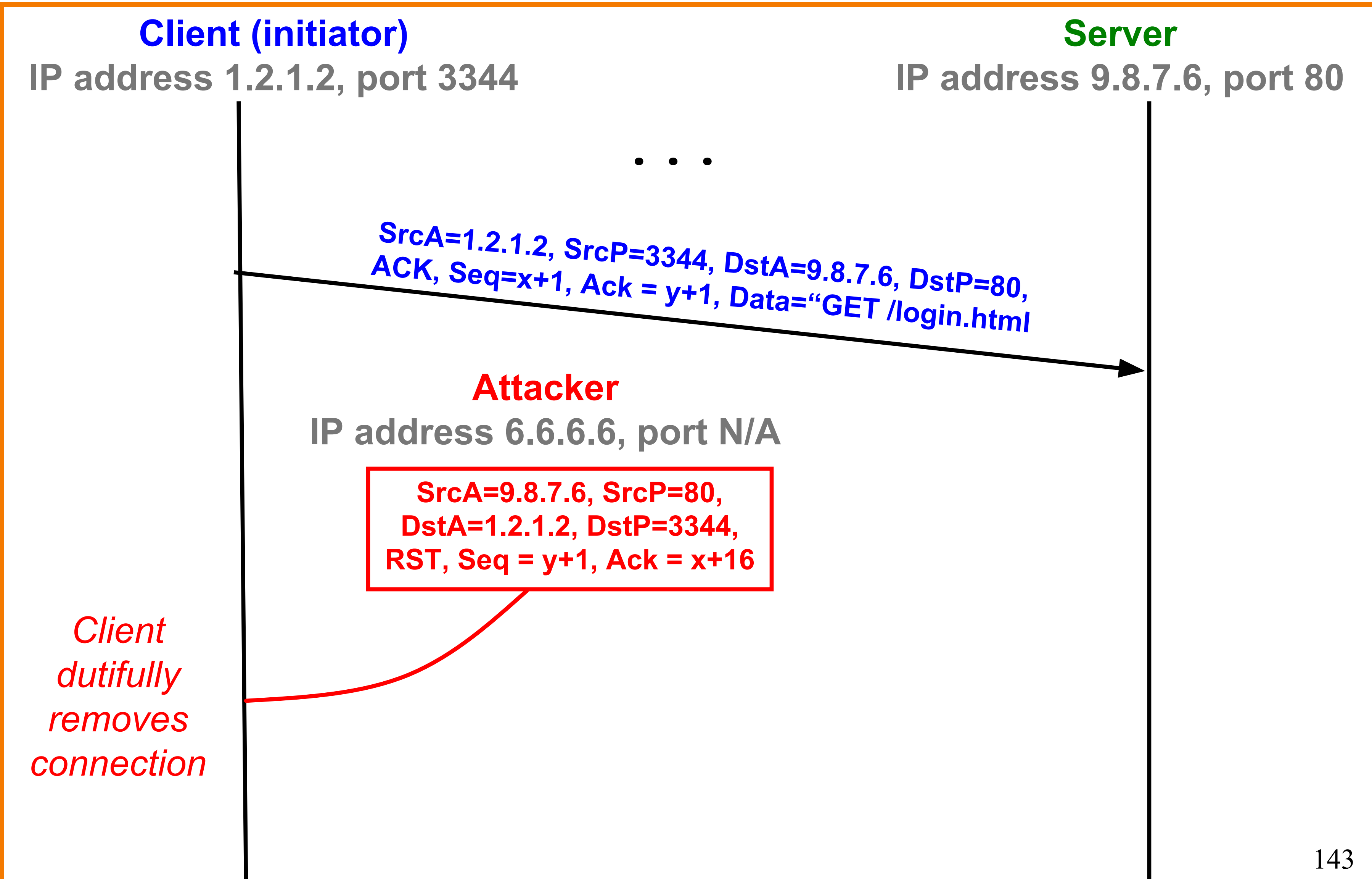


- A sends a TCP packet with RESET (**RST**) flag to B
  - E.g., because app. process on A crashed
  - (Could instead be that B sends a RST to A)
- Assuming that the sequence numbers in the **RST** fit with what B expects, That's It:
  - B's user-level process receives: `ECONNRESET`
  - No further communication on connection is possible

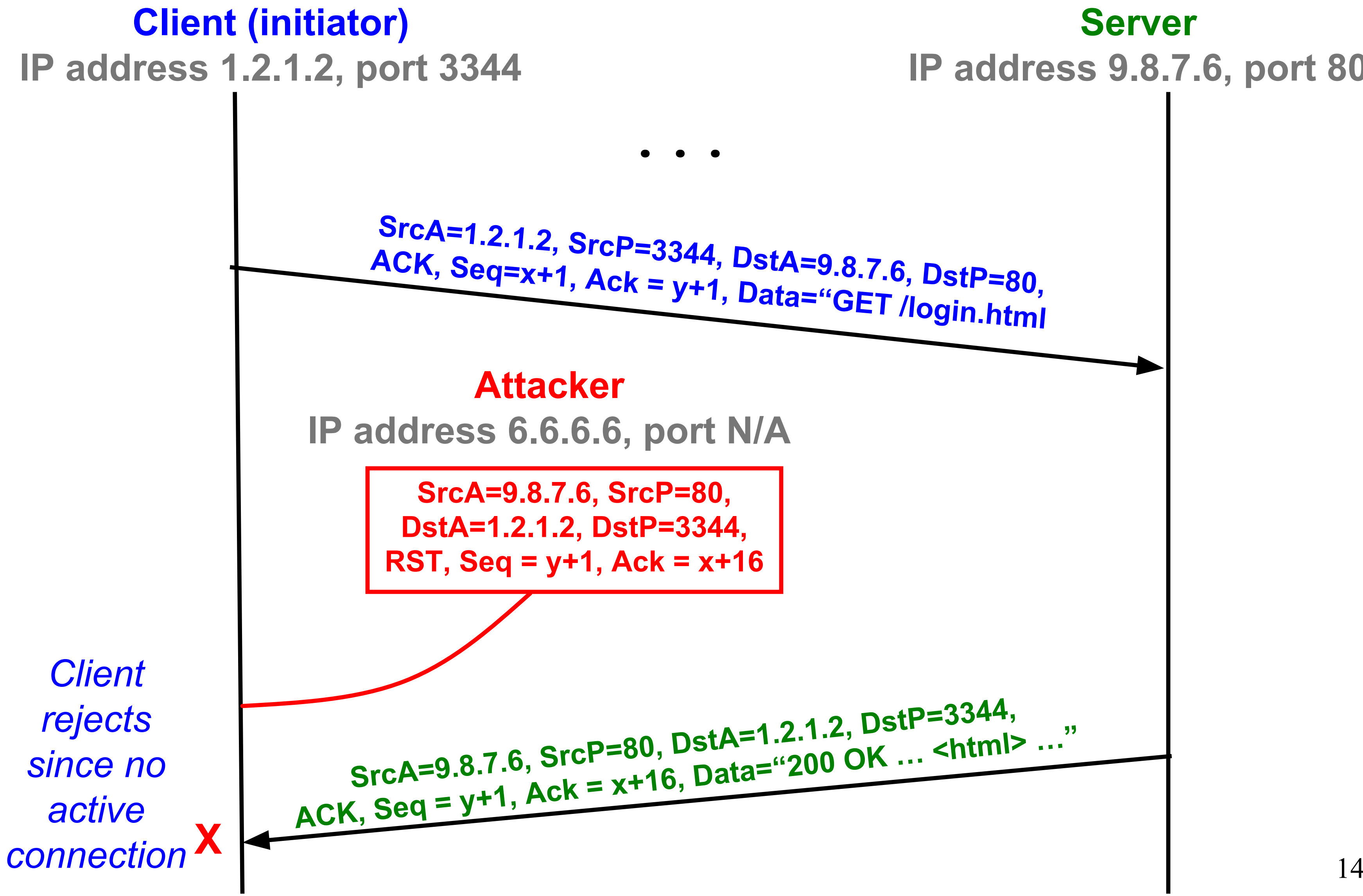
TCP Threat: Disruption

- Normally, TCP finishes ("closes") a connection by each side sending a `FIN` control message
  - Reliably delivered, since other side must ack
- But: if a TCP endpoint finds unable to continue (process dies; info from other "peer" is inconsistent), it abruptly terminates by sending a `RST` control message
  - Unilateral
  - Takes effect immediately (no ack needed)
  - Only accepted by peer if has correct* sequence number
- So: if attacker knows ports & sequence numbers, can disrupt any TCP connection

TCP *RST Injection*



TCP *RST Injection*



Threats to Comm. Security Goals

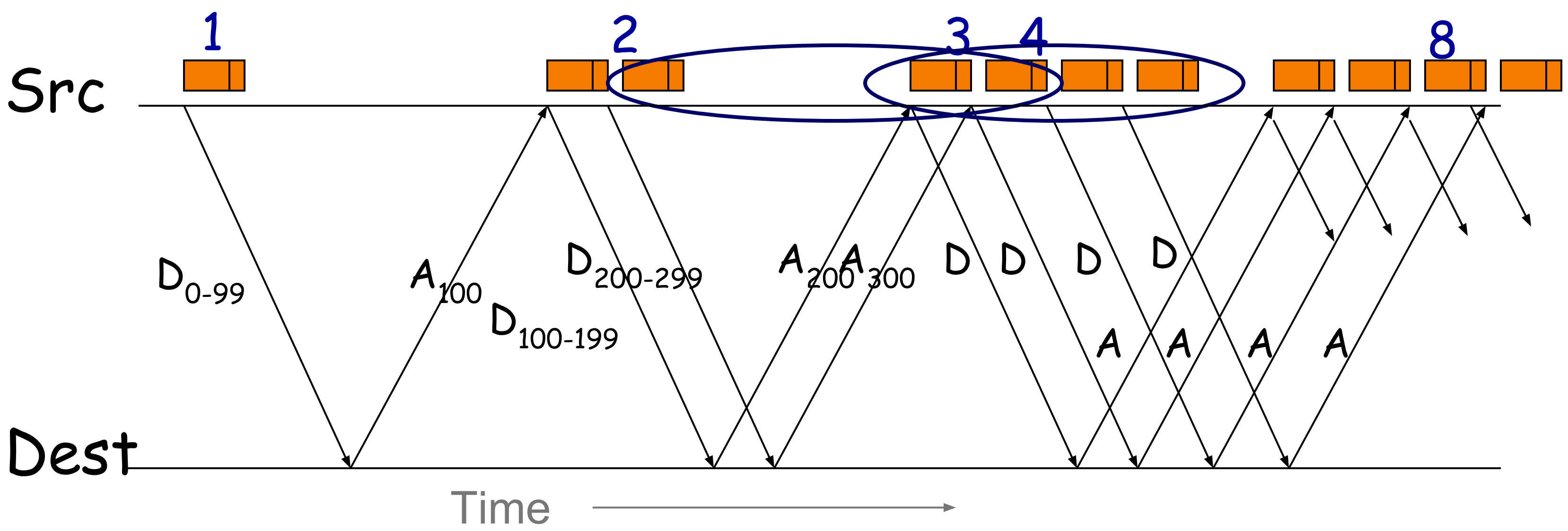
- Attacks can subvert each type of goal
  - Confidentiality: eavesdropping / theft of information
  - Integrity: altering data, manipulating execution (e.g., code injection)
  - Availability: *denial-of-service*

- Attackers can also combine different types of attacks towards an overarching goal
  - E.g. use eavesdropping (confidentiality) to construct a spoofing attack (integrity) that tells a server to drop an important connection (denial-of-service)

TCP's Rate Management

Unless there's loss, TCP doubles data in flight every "round-trip". All TCPs expected to obey ("fairness").

Mechanism: for each arriving ack for new data, increase allowed data by 1 maximum-sized packet

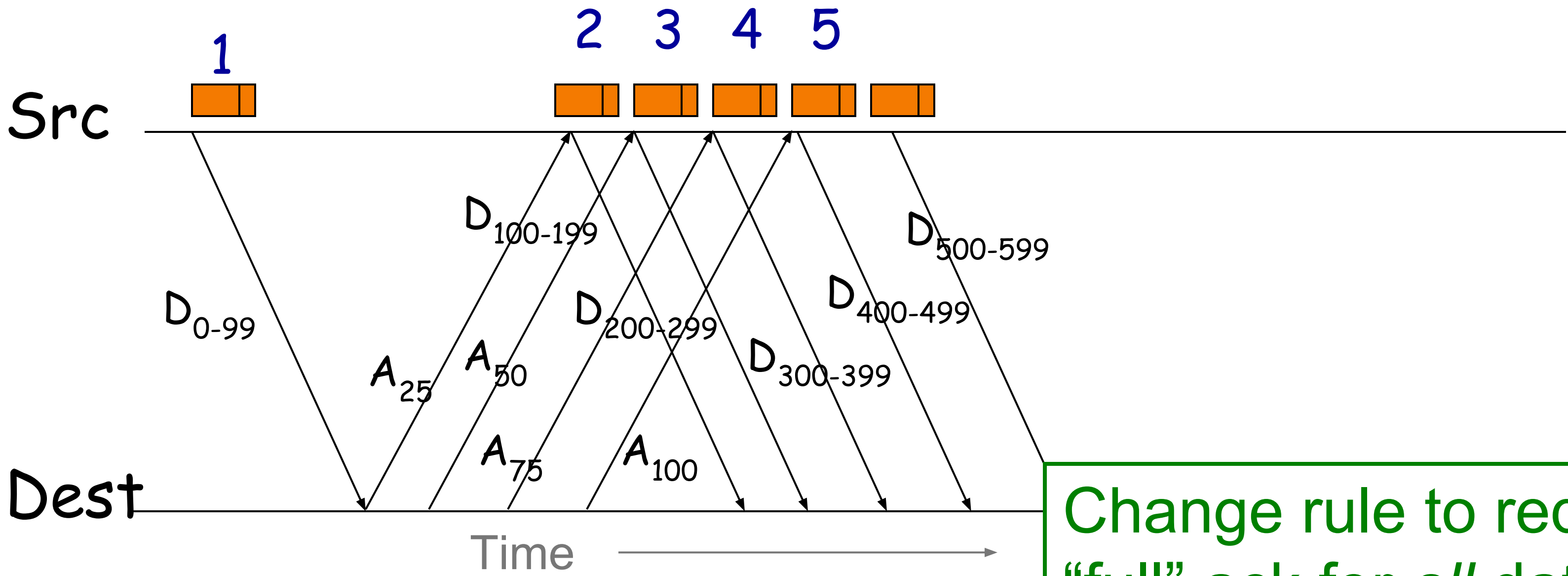


E.g., suppose maximum-sized packet = 100 bytes

Protocol Cheating

How can the destination (receiver) get data to come to them faster than normally allowed?

ACK-Splitting: each ack, even though **partial**, increases allowed data by one maximum-sized packet



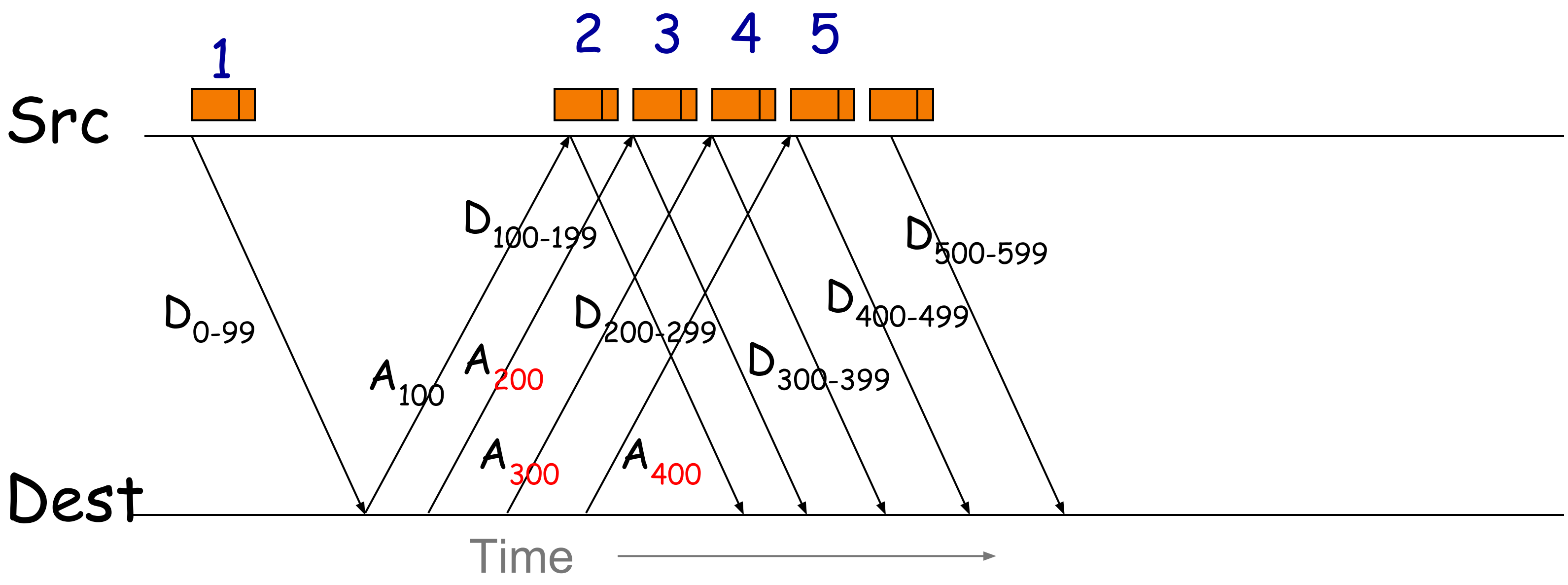
How do we defend against this?

Change rule to require "full" ack for *all* data sent in a packet

Protocol Cheating

How can the destination (receiver) *still* get data to come to them faster than normally allowed?

Opportunistic ack'ing: acknowledge data not yet seen!



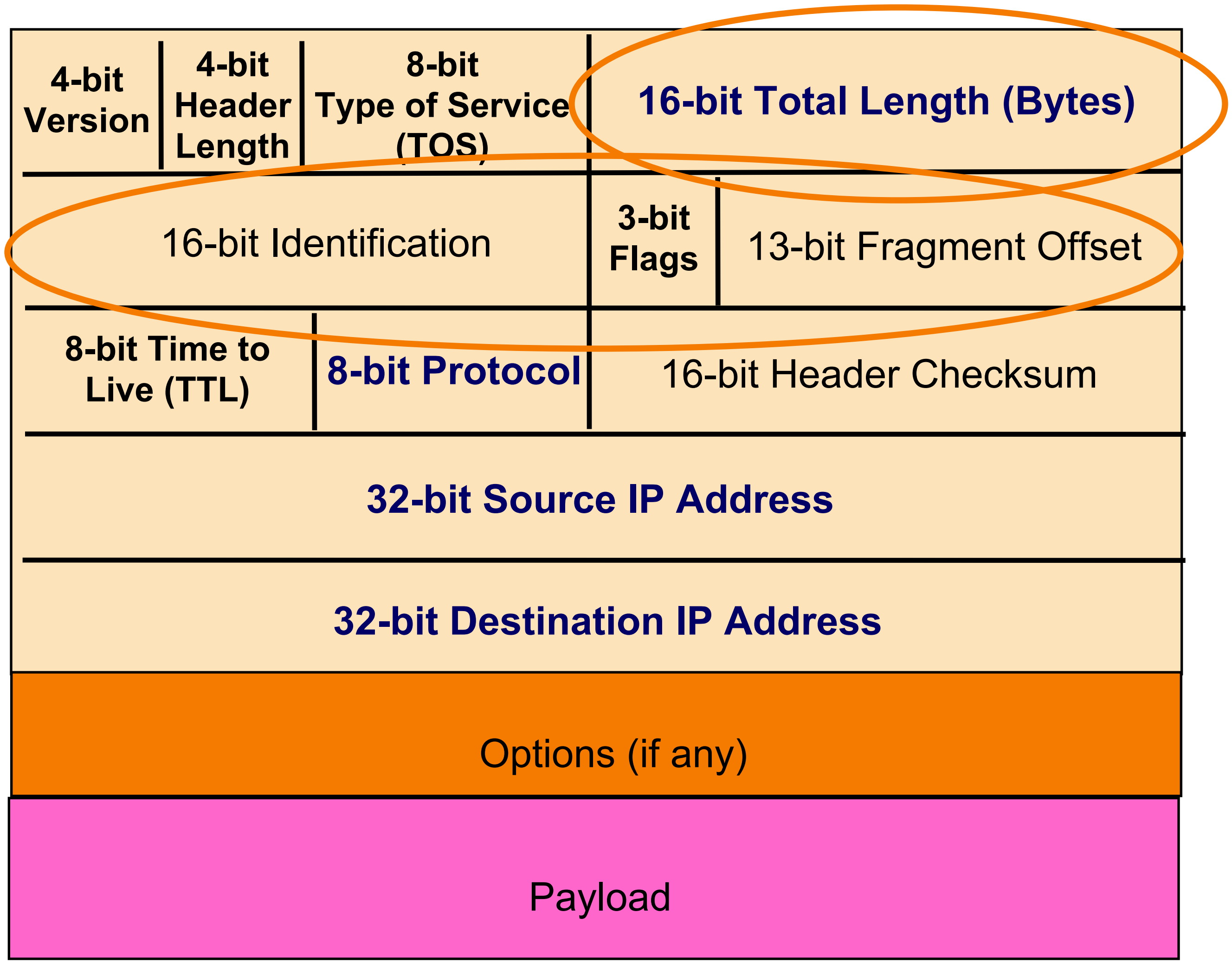
How do we defend against *this*?

Keeping Receivers Honest

- Approach #1: if you receive an ack for data you haven't sent, kill the connection
  - Works only if receiver acks too far ahead
- Approach #2: follow the "round trip time" (RTT) and if ack arrives too quickly, kill the connection
  - Flaky: RTT can vary a lot, so you might kill innocent connections
- Approach #3: make the receiver prove they received the data

  Note: a *protocol* change
  - Add a **nonce** ("random" marker) & require receiver to include it in ack. Kill connections w/ incorrect nonces
    - o (nonce could be function computed over payload, so sender doesn't explicitly transmit, only implicitly)

IP Packet Structure



IP Packet Header Fields (Continued)

- Total length (16 bits)
  - Number of bytes in the packet
  - Maximum size is 65,535 bytes (2^{16} -1)
  - … though underlying links may impose smaller limits
- Fragmentation: when forwarding a packet, an Internet router can split it into multiple pieces ("fragments") if too big for next hop link
- End host reassembles to recover original packet
- Fragmentation information (32 bits)
  - Packet identifier, flags, and fragment offset
  - Supports dividing a large IP packet into fragments
  - … in case a link cannot handle a large IP packet

Example: E-Mail Message Using MIME

MIME version

method used to encode data

type and subtype

encoded data

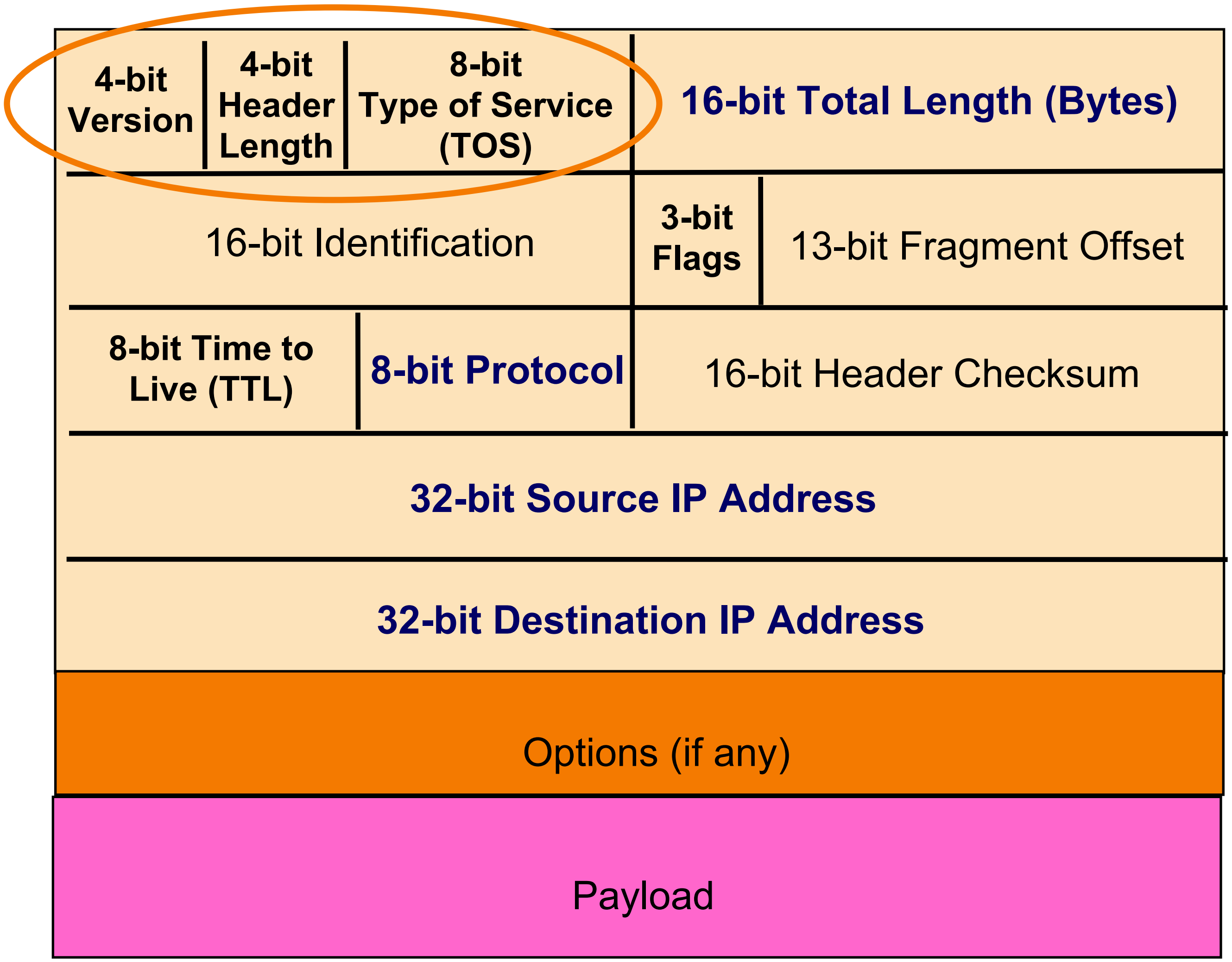
```
From: jrex@cs.princeton.edu
To: feamster@cc.gatech.edu
Subject: picture of my cat
MIME-Version: 1.0
Content-Transfer-Encoding: base64
Content-Type: image/jpeg

Base64 encoded data ….
JVBERi0xLjMNJeLjz9MNMSAwI
.........................
......base64 encoded data
```

Example With Received Header

Return-Path: <casado@cs.stanford.edu>
Received: from ribavirin.CS.Princeton.EDU (ribavirin.CS.Princeton.EDU [128.112.136.44])
by newark.CS.Princeton.EDU (8.12.11/8.12.11) with SMTP id k04M5R7Y023164
for <jrex@newark.CS.Princeton.EDU>; Wed, 4 Jan 2006 17:05:37 -0500 (EST)
Received: from bluebox.CS.Princeton.EDU ([128.112.136.38])
by ribavirin.CS.Princeton.EDU (SMSSMTP 4.1.0.19) with SMTP id M2006010417053607946
for <jrex@newark.CS.Princeton.EDU>; Wed, 04 Jan 2006 17:05:36 -0500
Received: from smtp-roam.Stanford.EDU (smtp-roam.Stanford.EDU [171.64.10.152])
by bluebox.CS.Princeton.EDU (8.12.11/8.12.11) with ESMTP id k04M5XNQ005204
for <jrex@cs.princeton.edu>; Wed, 4 Jan 2006 17:05:35 -0500 (EST)
Received: from [192.168.1.101] (adsl-69-107-78-147.dsl.pltn13.pacbell.net [69.107.78.147])
(authenticated bits=0)
by smtp-roam.Stanford.EDU (8.12.11/8.12.11) with ESMTP id k04M5W92018875
(version=TLSv1/SSLv3 cipher=DHE-RSA-AES256-SHA bits=256 verify=NOT);
Wed, 4 Jan 2006 14:05:32 -0800
Message-ID: <43BC46AF.3030306@cs.stanford.edu>
Date: Wed, 04 Jan 2006 14:05:35 -0800
From: Martin Casado <casado@cs.stanford.edu>
User-Agent: Mozilla Thunderbird 1.0 (Windows/20041206)
MIME-Version: 1.0
To: jrex@CS.Princeton.EDU
CC: Martin Casado <casado@cs.stanford.edu>
Subject: Using VNS in Class
Content-Type: text/plain; charset=ISO-8859-1; format=flowed
Content-Transfer-Encoding: 7bit

IP Packet Structure



IP Packet Header Fields

- Version number (4 bits)
  - Indicates the version of the IP protocol
  - Necessary to know what other fields to expect
  - Typically "4" (for IPv4), and sometimes "6" (for IPv6)
- Header length (4 bits)
  - Number of 32-bit words in the header
  - Typically "5" (for a 20-byte IPv4 header)
  - Can be more when IP options are used
- Type-of-Service (8 bits)
  - Allow packets to be treated differently based on needs
  - E.g., low delay for audio, high bandwidth for bulk transfer

Sample Email (SMTP) interaction

```
S: 220 hamburger.edu
C: HELO crepes.fr
S: 250  Hello crepes.fr, pleased to meet you
C: MAIL FROM: <alice@crepes.fr>
S: 250 alice@crepes.fr... Sender ok
C: RCPT TO: <bob@hamburger.edu>
S: 250 bob@hamburger.edu ... Recipient ok
C: DATA
S: 354 Enter mail, end with "." on a line by itself
C: From: alice@crepes.fr
C: To: hamburger-list@burger-king.com
C: Subject: Do you like ketchup?
C:
C: How about pickles?
C: .
S: 250 Message accepted for delivery
C: QUIT
S: 221 hamburger.edu closing connection
```

Email header

Email body

Lone period marks end of message