Computer Science Principles

Unit 4: Computer Systems and Networking

LECTURE 10: INTERNET (U2)

DR. ERIC CHOU

IEEE SENIOR MEMBER

The Internet

LECTURE 1

Seven Layer Architecture

LESSON 1 [UNIT 2 CODE.ORG]

Web-Domain Name Hosting Topology

STEPHISA TECHNOLOGIES, INC
Developing and Teaching the Future

Copyright © 2000-2008 Stephisa Technologies, Inc. All Rights Reserved.

OSI Model
Data
Layer
Host Layers
Media Layers
Data
Application
Network Process to Application
Data
Presentation
Data representation and Encryption
Data
Session
Interhost communication
Segments
Transport
End-to-End connections and Reliability
Packets
Network
Path Determination and IP (Logical addressing)
Frames
Data Link
MAC and LLC (Physical addressing)
Bits
Physical
Media, Signal and Binary Transmission

Device A
Intermediate node
Intermediate node
Device B
Peer-to-peer protocol (7th layer)
Peer-to-peer protocol (6th layer)
Peer-to-peer protocol (5th layer)
Peer-to-peer protocol (4th layer)
7 Application
7-6 interface
6 Presentation
6-5 interface
5 Session
5-4 interface
4 Transport
4-3 interface
3 Network
3-2 interface
2 Data link
2-1 interface
1 Physical
3rd
2nd
1st
Physical communication

Google
Network process to application
7. Application
Data representation and encryption
6. Presentation
Interhost communication
5. Session
End-to-end connections and reliability
4. Transport
Path determination and logical addressing
3. Network
Physical addressing
2. Data link
Media, signal and binary transmission
1. Physical
MAC
00-0C-F1-56-98-AD
router MAC
DNS , WWW/HTTP
P2P,EMAIL/POP,SMTP
Telnet, FTP
recognizing data
HTML, DOC, JPEG, MP3, AVI
Sockets. Session establishment
in TCP, SIP, RTP
RPC - Named pipes,
TCP, UDP, SCTP, SSL, TLS
IP, IPsec, ICMP, IGMP, OSPF
Ethernet, 802.11, MAC/LLC,VLAN, ATM, HDP
Fibre Channel Frame Relay, HDLC,
PPP, Q.921, Token Ring, ARP
RS-232, RJ45, V.34, 100BASE-TX, SDH, DSL, 802.11

OSI Model LAN Protocols

Logical Internet Model

Application Layer Protocols

- File Transfer Protocol (FTP)
- Hypertext Transfer Protocol (HTTP)
- Simple Mail Transfer Protocol (SMTP)
- Post-Office Protocol (POP3)
- Domain Name System (DNS)

Application Layer Protocols

Uniform Resource Locator:

<scheme>:<hierarchical part>[? query][# fragment]

Important Schemes: ftp, http, mailto, file, app, dns, https, im, info, mms

URL Sample: type http in google
https://**www.google.com**/search?**q**=Application+Layer+Protocols&oq=Application+Layer+Protocols&aqs=chrome.0.69i59.6246j0j8&sourceid=chrome&es_sm=122&ie=UTF-8#q=http

http://www.example.com/2012/samples/first.html

Scheme-Domain Name-----Directory--------Document

HTTP Connection Session Overview

HTTP Connection: Detailed Information

Internet
Wired(LAN)

Internet
WIFI

5G

100 BT and 1GeT

FDMA
power
time
frequency
TDMA
power
time
frequency
CDMA
power
time
frequency

Building a Network

LESSON 2 [UNIT 2 CODE.ORG]

Vocabulary

- **HTML** - HyperText Markup Language: the language of the web, code for web pages
- **HTTP** - HyperText Transfer Protocol: designed for communication on the web, for sending and receiving information
- **ASCII** – American Standard Code for Information Interchange – raw text that any computer can recognize
- **Abstraction** - a simplified representation of something more complex. Abstractions allow you to hide details to help you manage complexity, focus on relevant concepts, and reason about problems at a higher level.
- **Protocol** - A set of rules governing the exchange or transmission of data between devices.

Protocol, Character Set, Data Frame

- One of the most powerful uses of the internet is sending text to people. Since the internet can only send bits around we need a way to encode text with bits...
- If it were up to you, how would you encode text in binary? Quickly, jot down an idea for encoding text.

What's a protocol?

a human protocol and a computer network protocol:

Protocols

Protocols (Rules)

What to communicate, how to communicate and when to communicate.

Key Elements of Protocols

1. Syntax: Structure of format of data
2. Semantics: Meaning of each section of bits.
3. Timing: When to send and how fast?

HTML

- A lot of you have taken web design, and you used this language: HyperText Markup Language, or HTML
- This is the **language of the web**, where the content and formatting of a web page are written.
  Look at the background of a web page for the code.

Protocol called: HTTP

- HyperText Transfer Protocol
- this is the foundation for communication on the web – it was designed to send and receive web page data over the internet.

ASCII

- ASCII makes it all happen.
- ASCII – American Standard Code for Information Interchange
- This is the universally recognized raw text format that any computer can recognize

ASCII

- ASCII encoding is the standard number-to-text encoding scheme used in computers and on the Internet.
- This protocol is 7 bits long and so can encode 128 different symbols, each corresponding to one of the binary numbers between 0 and 127.
- Originally the encoding was designed for older printers and so the codes associated with the numbers 0-31 were designated as "control codes" which instruct the printer to perform certain actions.

ASCII

- ASCII Codes 32-126, however, contain many of the most commonly used symbols you use on the internet, including the lowercase and capital letters, the numbers 0-9, and most commonly-used punctuation marks.
- This printable character set is also used to represent all of the text formatting we use to add variety and emphasis to "plain-text" documents. The ability to represent formatting using only ASCII symbols is the result of cleverly designed protocols.

ASCII

- ASCII codes were originally 7 bits long and so there are 128 possible values.
- 0-31 are “control characters” that are largely defunct and go unused; they were formerly used to control various aspects of machines and printers.
- 127 is the symbol for delete.

ASCII

- Over time, 8 bits became a standard “chunk-size” for encoding information. ASCII made the transition to this 8-bit encoding by just adding an extra 0 to the front of the old 7-bit codes.

The Need for Addressing

LESSON 3 [UNIT 2 CODE.ORG]

Vocabulary

- **IP Address** - A number assigned to any item that is connected to the Internet.
- **Packets** - Small chunks of information that have been carefully formed from larger chunks of information.
- **Protocol** - A set of rules governing the exchange or transmission of data between devices.

Objective

- Explore more deeply how communication between multiple computers can work over the Internet.
- Time to play a simplified game of Battleship!

Game of Battleship

- The first game is played unplugged, in groups.
- The second game is played using the Internet Simulator, so that multiple students can connect to each other and see each other's messages.
- You must devise a messaging protocol that makes it clear who is sending the message and who the intended recipient is.

1 2 3 4 5 6 7 8 9 10
A
B
C
D
E
F
G
H
I
J

Rules: Game of Battleship

Components:

- 2 game units
- 10 plastic ships
- 4 runners of white pegs
- 2 runners of red pegs
- Label sheet

Each player's fleet contains 5 different ships:

- Carrier (5 holes)
- Battleship (4 holes)
- Cruiser (3 holes)
- Submarine (3 holes)
- Destroyer (2 holes)

Setup

You and your opponent sit facing each other and neighbor can see the other's ocean grid.

Place your fleet of 5 ships on the ocean grid.

Rules for placing ships:

- Place each ship in any horizontal or vertical position but not diagonally.
- Don't place a ship so that any part of it overlaps letters, numbers, the edge of the grid or another ship.
- Don't change the position of any ships once the game has begun.

Game Play

- Decide who will start.
- You and your opponent will alternate turns, calling out one shot per turn to try to hit each other's ships.
- On your turn, pick a target hole and call out its location by **letter** and **number**.

Battleships

Start New Game

Player 1 Board

Search your opponent's board

Computer hit your Battleship!

You hit the computer's Battleship!

Computer's Board

How to play

Game Play

When you call a shot, your opponent must tell you whether your shot is a hit or a miss.

It's a hit!

1. If you call out a shot location that is occupied by a ship on your opponent's ocean grid, your shot is a hit!
2. Your opponent tells you which ship you have hit (cruiser, submarine, battleship, etc...).
3. Record your hit by placing a red peg on your target grid. Your opponent places a red peg in the corresponding hole of the ship on his ocean grid.

Game Play

It's a miss!

1. If you call out a shot location not occupied by a ship on your opponent's ocean grid, its a miss.
2. Record your miss by placing a white peg in the corresponding target hole on your target grid, so you will remember this location and won't call this shot again.

End of the Game

Sinking a Ship

- Once all the holes in any one ship are filled with red pegs, the ship will sink. The owner of that ship must announce which ship was sunk.

End of the Game

- If you are the first player to sink your opponent's entire fleet of 5 ships, you win the game.

Demonstration:

AGAME.COM

- Understand the rule.
- Craft the communication protocol.
- Try to develop a protocol for human communication system. And, internet simulator version

What is needed in the game?

- You devise a **binary protocol** for playing this game which will entail developing an **addressing system** for players, as a well as a formal packet structure for transmitting data about the state of the game.

IP Addresses

- “Internet Protocol” addressing system, commonly known as **IP Addresses**.
- Many computers are connected together by networks. So if a computer sends some bits out, those bits pass through many computers. How does a computer know who those bits are for?
- If a response is necessary, how does it send it back? This is a complicated question, but the first part of the answer is that you need an addressing system, and some kind of fixed structure for messages that everyone agrees on (a **protocol**) so messages can be interpreted properly. (We'll get to routing later).

Activity:

Game of Battleship on Internet Simulator

- The Internet Simulator for this lesson is setup to “broadcast” every message to every person in the group. Since everyone receives any bits that were sent by anyone else, a method for identifying the intended sender and receiver of the message is needed.
- Students will invent a protocol for addressing messages.

Activity:

Game of Battleship on Internet Simulator

- Many network systems, such as local **Ethernet** and **WiFi**, rely on addressing schemes to make sure bits are received by the correct computer based on address and for other computers to simply ignore messages not intended for them.

Activity:

Game of Battleship on Internet Simulator

- We will play a crazy game of Battleship where instead of playing against one other person you will play multiple games against multiple other people simultaneously -- We call this "Broadcast Battleship“
- In our version today, you will play in groups of 3 or 4. To make it easier to track, we've also simplified the playing board to just a 3x3 grid (instead of the classic 10x10)
- And we'll just play with paper and pencil.

Activities:

1. Look at the Rules of Play
2. Play the game, unplugged!
3. We are going to add a new challenge: Your team is going to have to play Battleship without talking. You will only be able to use the **Internet Simulator** to communicate

Video

IP Address and DNS

Fault Tolerant

LECTURE 2

Routers and Redundancy

LESSON 4 [UNIT 2 CODE.ORG]

Vocabulary

- **Network Redundancy** - having multiple backups to ensure reliability during cases of high usage or failure
- **Router** - A type of computer that forwards data across a network

Chicago, IL
New York/Newark
Fairfax County, VA
San Jose/Santa Clara, CA
Dallas/Richardson, TX

Router
Router
Router
Client Computer
Router
Router
Internet Server
Router
Router
Router

Internet Routing

Objectives

- We are exploring the benefits (and potential security concerns) associated with routing traffic across the Internet.
- We will use a version of the Internet Simulator that allows messages to be sent only to an intended recipient, as indicated by the IP address.

Traffic in the Network

- The Internet Simulator also allows students to examine the traffic that goes through all of the (simulated) routers on the network.
- You will discover that messages go through many different routers, may not always take the same path to reach the final destination, and that the routers (and their owners) can see all of this traffic!

Objectives

- The main purpose here is to use the Internet Simulator to dive into examining the network traffic that's generated by students in the class.
- Seeing traffic like this in real life is very hard if not impossible since only engineers of large networks have access to router data on a large scale, and it's also more complicated to understand.

Background Knowledge

- CS Background: A **router** is a computer designed to receive and redirect packets of information based upon the addressing information (e.g. an IP address) contained in the packet.
- Routers will either deliver a packet to its final destination or forward it to one of several other routers it is connected to.

Background Knowledge

- By monitoring current network conditions, a **router** can determine which of these will allow the packet to reach its destination fastest.
- There will often be redundant paths between two locations on the Internet, and so if one path is experiencing traffic or otherwise out of service, additional paths will be available.
- This **redundancy** makes the Internet more reliable and also helps the Internet to scale, accommodating new users (and routers!) as they are connected to the system.

Scenario:

- Imagine you were going to send a letter to a friend living in another state. List the steps you imagine your letter would have to take through the different parts of the postal system.
- Don’t worry if you’re not sure about your answers, just make an educated guess.

Scenario:

- Did you come up with something like this:
- **Example:** Mailbox → Mail Carrier → Local Post Office → Regional Collection Center (of origin) → Regional Collection Center (of destination) → Local Post Office → Mail Carrier → Grandparent's House

Scenario:

- When we send messages through a network we don’t actually want everyone on the network to receive them.
- If we include information about who the message is intended for then we can allow portions of the network to focus on sorting and routing messages, so that they can continue on their way to their intended target.

Scenario:

- In the mail system, mail facilities, post offices, or a mail carrier fills this role.
  In a network of computers, certain computers called “routers” do the same thing, directing messages towards the target computer based on the IP addresses included in the message.

Activity

Internet Transmission with Router

- Today's activity introduces the **newest incarnation of the Internet Simulator**
- Messages now have “To” and “From” addresses in a format similar to IP addresses (just an 8-bit address instead of a full 32 bits)
- This allows messages to be sent to a single intended recipient.

Activity

Internet Transmission with Router

- The Internet Simulator also simulates the routing of messages across a network, with messages possibly being routed across multiple routers in unpredictable sequences before finally being delivered.
- This is done to simulate the way traffic travelling across the Internet is constantly rebalanced in response to over- or under-usage of some channels.
- A message will usually make it to its destination, but we can't know for sure how it will get there.

Video

https://www.youtube.com/watch?v=ewrBalT_eBM

How does Internet works?

Find the New Internet Simulator

Choose a Router:

- Add a router if you need more space.
- Then join a router with a few of the people sitting closest to you.
- Ideally, you'll have 3-4 classmates with you on your router.

Activity

Activity Guide: Routers and Redundancy

Send a quick test message:

- Send a simple "hello" to a friend who is connected to the same router.

Steps:

- Find friend's (small) IP address
- Send a message to that address
- Friend should send a response
- Choose a router

Activity

Activity Guide: Routers and Redundancy

- Join a router with some of your classmates, adding more if you need them. There should be at least three students on each router.
- Messages now include a "To" and "From" IP address in their header. Students should use the **network visualization** to find the IP addresses of some of their classmates.

Activity

Activity Guide: Routers and Redundancy

- **Have a Conversation:**
  Conduct a short conversation (e.g. a simple greeting or a question and answer) with two friends on your router.

- Confirm that both sides are receiving their messages.
  Take a few minutes to practice.
  If your To address is not constructed properly the message will be dropped.

Discussion

- some messages may be dropped
- messages may take different paths to the same location
- routers can theoretically read all traffic moving across them.

Discussion

Read the Traffic:

- Once you have conducted your conversations, pause and demonstrate the ability to view router traffic in the Router tab.
- You should use the data generated here to answer the first set of questions on the Activity Guide.
- Click the “Log Browser” button to view the table of router data.

Discussion

Routing messages functions very differently from **broadcasting**.

- The messages are only sent to the user they are intended for.
- This is just like when you write a letter to a friend, where an address on the outside of the letter tells the mail service how to properly route your letter.

Broadcasting

Routing

Discussion

The Router Logs

- The view we have in this table of traffic wouldn't be available to a normal user on the network -- this is a simulation after all -- but it WOULD be visible to the router.

Discussion

- A router needs to read the header of the message to route it correctly *
- But the contents of the message are right there too, and if it wanted, a router could read all the messages going across it, just like you did.
- This should raises some questions about privacy and security -- we'll talk more about this later.

Network B
Network C
PRACTICAL NETWORKING .NET
Network A
Network D
A
B
Packet Traveling
How data gets from here to there

Circuit Switched Network

Receiver Caller

Packet Switched Network

Receiver Caller

IP Packet Structure

Encapsulation

Host
User
Application
write(fd, buf, len)
User data
Kernel
File
Validate file descriptor.
Sockets
Copy/append buf to socket buffer.
Send socket buffer:
TCP
Create TCP segment according to TCP state.
Compute checksum.
TCP
IP
Add IP header, perform IP routing.
Compute checksum.
IP
TCP
Ethernet
Add Ethernet header, perform ARP.
Eth.
IP
TCP
Driver
Tell NIC to send the packet.
Device
NIC
Fetch the packet from
host memory and send it.
Interrupt the host when send is done.
IFG
Pre.
Eth.
IP
TCP
CRC

Network Topology

Data Flow

Find a Classmate on a Different Router:

- The Internet Simulator can route messages between routers, allowing the entire class to communicate.
- Find two classmates on a different router and ask for their IP addresses. (You'll need to actually talk to one another; you currently won't be able to do it on the Internet Simulator.)

Find a Classmate on a Different Router:

- Have a Conversation:
- Again, you should conduct a short conversation with your two partners, confirming verbally that the messages are being received.
- Troubleshoot errors you may be encountering as before.

Activity

Reading Network Traffic:

- Once you have been able to talk with classmates, look at the "Log Browser" button.
- It is possible to read traffic across the entire network, not merely your router by clicking the “Show all routers” button.
- Open the network traffic and examine the traffic in order to answer the questions listed there on the back page of the activity guide.

Redundancy

- The simulations you used in today's lesson model many features of the actual Internet, most notably its redundant nature (there are many paths between locations).
- Identify these features with the following comments, and note that at the top of the next page you'll find a routing table that shows the different routes taken by the messages.

Source of Redundancy

- Encapsulation
- Retransmission
- Re-routing

Redundancy

- On the Internet Simulator we see messages appearing in the Router Logs multiple times (like in the image to the right).
- This is done to simulate a message passing through multiple routers on its way to its destination.
- A row is created each time it shows up at a new router.

Discussion

- If you trace carefully you'll notice that messages between two people **don't** always visit the same routers along the way.
- This is not a mistake; it’s modeled after the way the actual Internet was designed.
- Why might **the Internet have been designed to be flexible about how messages get from one person to another**? Why go through the trouble of creating multiple paths between users?

Discussion

- **Efficiency:** Communication lines can become backed up with traffic.
- The ability to choose whichever one currently has the least traffic improves the speed of transmission.
- **Reliability:** Having many redundant paths means if one goes down, there are many others that a message can take.

Discussion

- Benefits of redundancy in a routed system, to provide many paths between users of the network.
- Routed messages aren’t seen by all users, but they can still be seen by owners of the router.
- If you control all the routers, you entirely control the way messages travel through the network, and additionally can see all the traffic.

BUFFERING...

Audio/Video source
WebServer
http://webtv.poliba.it
Encoder
Web page content
Multiple levels encoding
L3
L2
L1
L0
VoD Cloud Storage
Streamer
CDN Service
Web Player
Web Player
Web Player
...
iOS
Windows phone
HLS, DASH formats (MPEGTS, fMP4)
Adaptive HTML5 video players

Basic Streaming Process
©2007 HowStuffWorks
Web Server
1
A user visits a Web page hosted on a Web server and finds a file he'd like to see or hear.
2
The Web server sends a message to a streaming media server, requesting the specific file.
Media Server
Client
3
The streaming server streams the file to the user's computer, bypassing the Web server.
4
The client software on the user's computer decodes and plays the file.

veoh
dish
NETFLIX
YouTube
amazon.com
hulu
vudu
mcast

Practice

Do the Assessments and any other materials inside Code Studio.

Homework for Lesson 10:

- Find the news article listed on the handout, and answer the questions.

Packets

LESSON 6 [UNIT 2 CODE.ORG]

Vocabulary

- **Packets** - Small chunks of information that have been carefully formed from larger chunks of information.
- **TCP** - Transmission Control Protocol - provides reliable, ordered, and error-checked delivery of a stream of packets on the internet. TCP is tightly linked with IP and usually seen as TCP/IP in writing.

Objectives

- YOU WILL develop a protocol for reliably sending a message over an unreliable internet.
- The Internet Simulator has been setup for this lesson to restrict messages to no more than 8 characters each, and messages get dropped messages with some probability on every hop.
- This is a problem-solving lesson. On the real Internet packet sizes are limited, and transmission is unreliable.
- In this lesson we have set up the Internet simulator to restrict packet-size to be very small - 16-bits for the ”to and from” addresses plus only 8 ASCII characters.

Objectives

- Also, the Simulator drops packets pretty regularly. In a set of 10 messages it's very likely one or two will be dropped.
- This is a real problem in the real internet world.

Problem Description

- The problem you have to solve is how to use the 8-ASCII-characters-worth of data to include both a piece of message you're trying to send, as well as information about how many messages (packets) there are in the whole message, and which number this packet is.
- The need for something like the **Transmission Control Protocol TCP**.
- TCP was designed to overcome the inherent unreliability of the Internet.
- A small but non-negligible percentage of packets are lost in transmission because of faults in the infrastructure of the Internet.

Solution

- To constrain the extent of these errors, larger messages are divided into many packets which are individually routed to their recipient.
- The receiving computer will send an acknowledgement confirming the receipt of that packet.

TCP

- If the sending computer does not receive an acknowledgement, it will resend the packet until all packets have been acknowledged.
- Since packets may arrive out of order, additional data must be included to indicate the order in which the packets should be arranged.
- Thus, while individual packets **cannot** be guaranteed to arrive, eventually an entire message can be accurately reconstructed.

TCP State Diagram

Activity: Internet Simulator

- Your friend sent you a message on the Internet, but you never received it.
- Based on what you already know about routers and the physical Internet, list what reasons might explain this fact.

Problems can be

- Wires are cut
- Interference on a radio channel
- Router malfunctions or cannot keep up with traffic being directed to it

Activity

- When we communicate on the Internet, we are not just sending short text messages as we did yesterday.
- We also use the Internet to exchange documents, videos, music, and scientific data, and these files can easily grow to enormous size.

Solution: Packets

- All of this would not be a problem if the Internet were perfectly reliable, but in reality, errors sometimes occur.
- Wires can be cut, routers can be overwhelmed with traffic, and interference with electric or radio signals can cause messages to become corrupted.
- The response to this problem is to split large messages into smaller pieces of information called **packets**.

Solution: Packets

- It turns out that splitting up a message into packets provides many benefits. If a faster route opens up halfway through transmitting a large file, it is easy to reroute later packets in the transmission through that route.
- Splitting up a message into smaller chunks doesn't solve all the problems of unreliability on the Internet. Packets can still be dropped or arrive out of order.

Activity

- Today's challenge is to develop a protocol to reliably send messages even though the network itself is unreliable.
- The NEW version of the Internet Simulator we will be using today has been structured to simulate the unreliability of the Internet.

Activity

- In particular you'll notice a few changes:
  You will only be allowed to send packets containing 8 characters of text!
- Anything larger than 8 characters will be cut off...
  However, you may construct multiple packets prior to sending them, by clicking "Add Packet", and then send them all with one click of the "Send" button.
- Every message has a small chance of being dropped on each “hop” it makes between routers.

Activity Guide

Packets and Making a Reliable Internet

- GET in groups of 2 to 4 members.
  Students (individually or as a group, either is OK)
- Should log into Code Studio and access the Internet Simulator.

Internet Simulator Setup

- We need to Add at least four routers (six or more is ideal; there’s no need to max out the connections on each router
- It's fine if only one or two students are connected to a router)
- Join a DIFFERENT router from your groupmates.

Internet Simulator Setup

- Take some time to get used to the changes introduced to the **Internet Simulator**.
- Exchange messages across routers.
- Try to construct a multi-packet message and send multiple packets at once to someone else try the same thing for a classmate on ANOTHER router and then view the router logs to examine the result of these transmissions.

Activity

- Send a drawing with ASCII text and come up with ways to break it up.
- The wider the drawing is, the more challenging the protocol will be to develop since each individual message is limited to only 8 characters.

```
@@@@          v
@   @      .-, |
@@@@      ((.))/
@   @      / \/
@@@@      /___\
          _/ |_
```

ASCII ART IS FUN

Discussion

You should recognize:

- Packets are dropped with some frequency
- Packets of more than 8 characters are always truncated to just the first 8 characters
- Packets sometimes arrive out of order.

Discussion

- Develope a protocol that will allow you to overcome the unreliability of the network so that a message can be sent and both sender and receiver can be confident the full message was received.

Guidelines for Protocol:

- All communication can only be done through the Internet Simulator.
- The full message sent will be at least 80 characters long - broken into at least 10 packets - and might be entirely random (i.e. there's no way to use human intuition to reconstruct the message).
- The message is not known beforehand.
- The sender and receiver must be confident the full message was successfully transmitted and reconstructed.

Guidelines for Protocol:

- The real problem to solve is to think about what the recipient of the messages should do to inform the sender of what's missing and needs to be re-sent.
- It's tricky because any message the recipient sends back to the sender also stands a chance of being dropped or lost.
- Your protocol needs to **overcome this unreliability** for both the sender or receiver.

Transmission Control Protocol (TCP) Header

20-60 bytes

<table>
<tr><td colspan="4">source port number
2 bytes</td><td colspan="2">destination port number
2 bytes</td></tr>
<tr><td colspan="6">sequence number
4 bytes</td></tr>
<tr><td colspan="6">acknowledgement number
4 bytes</td></tr>
<tr><td>data offset
4 bits</td><td>reserved
3 bits</td><td colspan="3">control flags
9 bits</td><td>window size
2 bytes</td></tr>
<tr><td colspan="4">checksum
2 bytes</td><td colspan="2">urgent pointer
2 bytes</td></tr>
<tr><td colspan="6">optional data
0-40 bytes</td></tr>
</table>

5 Application Layer

The Application layer is the group of applications requiring network communications.

4 Transport Layer (TCP/UDP)

The Transport layer establishes the connection between applications on different hosts.

3 Network Layer (IP)

The Network layer is responsible for creating the packets that move across the network.

2 Data Link Layer (MAC)

The Data Link layer is responsible for creating the frames that move across the network.

1 Physical Layer

The Physical layer is the transceiver that drives the signals on the network.

Video

The Internet: Packets, Routing, and Reliability

(about 6 minutes)

Connect activity to TCP

- The challenges we encountered in today's activity very closely mirror those that exist on the actual Internet.
- The response was the development of a protocol called the Transmission Control Protocol, or more simply, TCP.
- TCP divides larger messages into **smaller** packets which have ordering information added to their header.
- When a packet arrives at a destination computer, an acknowledgement is sent to the sender, letting them know they don't need to resend that packet.
- Once all the packets have arrived, the ordering information in the headers of the packets allows them to be reordered to create the original message.

Practice

DO your reflections and assessments in **code studio**.

Parallel and Distributed Computing

LECTURE 3

DNS

LESSON 6 [UNIT 2 CODE.ORG]

Vocabulary

- **DNS** - DOMAIN NAME SYSTEM - The service that translates URLs to IP addresses.

Lesson Materials

1. One (only one) IP Address Label
2. Worksheet for names and address
3. Activity Guide – DNS partner questionnaire
4. Activity Guide –Research: DNS in the Real World
5. Homework for Lesson 12

Domain Name Server

- The core idea of this lesson occurs in the unplugged activity that kicks off the lesson, in which students try to keep track of IP addresses that had been randomly assigned to each student in the class, while at the same time the students' IP address changes.
- This leads to identifying the need for an authoritative system for name-to-address mappings, known as the Domain Name System or DNS.

DNS Lookup & Internet Traffic

Domain Name Server

- You will then briefly experiment with a DNS protocol in the Internet Simulator. The activity is similar, in that you will have to grapple with IP addresses changing in real time and use the built in DNS protocol to resolve the issues.

Denial of Service Attacks

- The lesson ends with you doing some rapid research about DNS and some of its vulnerabilities, particularly what are known as **Denial of Service Attacks**.

Purpose

- The basic purpose of this lesson is to show some of the challenges solved, and created by, **DNS**. At its core, the DNS is "simply" a hierarchical system of computers and databases, that maps IP addresses to domain names.
- It enables Internet users to connect human-language locations on the Internet with numeric addresses used by **IP**.
- While distributed and hierarchical, it can be treated in the abstract as a centralized registry of locations on the Internet, allowing users to quickly find locations they are looking for and register themselves so that others may find them.

Activity

- USE THE IP ADDRESSES and THE WORKSHEET.
- For the next 5 minutes, your goal is to complete an accurate list of IP addresses and names for all students in the room. You may only talk to one person at a time, but you may exchange as much information with that person as you want.

IMPORTANT: From time to time, get a NEW IP Address from the collection of addresses.
GO!

Domain Name and IP Addresses

- Why did we switch your IP addresses?
- This simulates the fact that a computer's IP address does not stay the same.
- For example, a person's IP address on their phone changes quite frequently as they move around throughout their day and their phone tries to connect to the Internet from different locations.

Domain Name and IP Addresses

- Do you think the system we just simulated is an efficient way of collecting IP addresses? Are there any inefficiencies you observe? How could it be made better?
- A central list would be better, and the **Internet** has a system for that.

Activity Guide

- Open Code Studio and connect to the Internet Simulator and use the **DNS Partner Questionnaire - Activity Guide.**
- The new configuration of the simulator includes a DNS server.
- A DNS server now appears attached to every router.

DNS Protocol in Experiment

- We no longer can see anybody's IP address. To get an IP address, we have "ask" the DNS server using a text-based protocol.
- Try this: see how to send a request to the **DNS** for someone's address.
- Try the **DNS** protocol to get the address of someone who is attached to their router.

DNS in the Internet Simulator

When you go to the internet simulator now. You will see a “DNS server” attached to the router. In order to communicate with someone else, you must first find their IP address by asking the **DNS**.

1. To begin, click over to the “DNS” tab to see all the hostnames of people on the router. You will see the address of the DNS (always 15) but will not see an address for anyone else on the router.
2. The DNS server responds to a text protocol that will give you someone’s IP address. The protocol is:

   GET <hostnameOfPerson>

For example,

GET madeline4

DNS

- After the DNS has returned an IP address, you can type that IP address into the “To” field, enter a message, and then press “send.”
- You are going to interview/have a conversation with a classmate using only the Internet Simulator. We’ve created a list of interview questions (on the back) and you should both jot down each other’s responses.
- To find the person, you will have to ask the DNS for her IP address. When you have retrieved the IP address, start the interview.

Domain Name Switch

- HOWEVER….As you're working, you both MUST disconnect and reconnect from the simulation. This is to simulate changing IP addresses.
- Do this at least twice during the interview.
- Even though your IP address will change, your hostname will stay the same, so you'll need to re-join a router and ask the DNS for your partner's new IP address in order to continue having your conversation!

ChrisPC DNS Switch Settings

ChrisPC DNS Switch

DNS Settings

DNS Database

Program Settings

About DNS Switch

Visit our Website

Like us on Facebook

DNS Database

Show: All DNS Presets

No	DNS Preset Name	Preferred DNS	Alternative DNS	Category
1	OpenDNS Home	208.67.222.222	208.67.220.220	Regular DNS
2	Google Public DNS	8.8.8.8	8.8.4.4	Regular DNS
3	Comodo SecureDNS 2.0	8.26.56.26	8.20.247.20	Secure DNS
4	Norton ConnectSafe Basic	199.85.126.10	199.85.127.10	Regular DNS
5	Norton ConnectSafe Family	199.85.126.30	199.85.127.30	Family Safe ...
6	Norton ConnectSafe Secure	199.85.126.20	199.85.127.20	Secure DNS
7	OpenNIC Europe	151.236.6.156	5.9.234.2	Secure DNS
8	Swiss Privacy Foundation	77.109.138.45	77.109.139.29	Anonymous...
9	NeuStar DNS Advantage	156.154.70.1	156.154.71.1	Secure DNS
10	OpenDNS Family Shield	208.67.222.123	208.67.220.123	Family Safe ...

Add DNS Preset | Edit DNS Preset | Delete DNS Preset

Protect your privacy & surf anonymously with ChrisPC Anonymous Proxy Pro. Click for details.

ChrisPC DNS Switch 1.00
Developed by Chris P.C. srl

Hide settings

Video

- You may remember (from the IP/DNS video that we saw several lessons ago) that you learned about the Internet system (v DNS) for sharing names and IP addresses. Let's watch that section again!
- Watch the DNS portion of this video as a transition to the next activity.
- Video: **IP and DNS - start at 4:12**

DNS

- Hopefully we all get the basic idea: the **DNS** is the large-scale system that translates human-readable web addresses into their numeric IP addresses so that computers can communicate.
- This system however was not designed to be secure and that has resulted in some major security incidents over time.
- You're now going to learn about some of them and how they work.

Activity Guide

Get in groups of 3-4 people to complete readings.

Use: Research: **DNS in the Real World** - Activity Guide, one copy per student.

Each person in the group picks an article about **DNS** and **DDoS** attacks

- The list of articles can be found on the first student page (bubble 1) on Code Studio for this lesson. DON’T ALL PICK THE SAME ARTICLE!
- Complete the first page of the worksheet

Activity Guide

So the front page is about your article that you read.

- Flip the paper over, and share with your group members Key points about their articles.
- Find other people in the room who have different articles, and add those points.

Try to get info on as many articles as you can.

Summary

- The Internet is basically a network of computers sending messages to request information and computers replying to messages to satisfy information requests.
- Computers need to identify “from” and “to” for all messages. Computer speak in numbers, not names.
- All communication online is via IP addresses. However, we are more familiar with human readable names, most notably URLs like “**Code.org**” or “**Google.com**.”
- We need a way to translate **human-readable** names into IP addresses.
- It is inefficient for everyone on the Internet to maintain a table of **IP** addresses.

Summary

- The DNS is NOT centralized, but it's not completely autonomous and distributed like routing, either. There is a hierarchical system of servers to maintain an authoritative table that, like a phone book, others can consult when they need to find an address.
- A properly functioning **DNS** system requires collaborative efforts among all users to ensure it is up to date and accurate.

Practice

- Finish up in the Code Studio with Assessments and Reflections.

HTTP

LESSON 6 [UNIT 2 CODE.ORG]

Vocabulary

- **DNS** - The service that translates URLs to IP addresses.
- **HTTP** - HyperText Transfer Protocol - the protocol used for transmitting web pages over the Internet
- **IP Address** - A number assigned to any item that is connected to the Internet.
- **TCP** - Transmission Control Protocol - provides reliable, ordered, and error-checked delivery of a stream of packets on the internet. TCP is tightly linked with IP and usually seen as TCP/IP in writing.
- **URL** - An easy-to-remember address for calling a web page (like www.code.org).

Objectives

- You will be introduced to another high-level protocol of the Internet, **HTTP**. We review the layers of the Internet covered so far, and then watch a video covering high-level protocols of the Internet, most notably **HTTP**.
- You will investigate **HTTP** traffic generated within your own browser by accessing the browser's developer tools and visiting a variety of websites.
- A handout summarizing the structure of **HTTP** is provided to help you understand the components of the **HTTP** requests and responses you will observe. Finally you will see how the layers of the Internet make use of abstraction.

Big Idea

Abstraction

- One of the BIG IDEAS for the College Board is the idea of **Abstraction**

Two Purposes to this lesson:

1. Get a basic understanding of what **HTTP** is and what it's for. **HTTP** like DNS is an ASCII-text based protocol - it's just two computers sending text messages to each other.

   - What makes it a protocol are the rules of the "conversation" the two machines are having.
   - In the case of **HTTP**, it is a **call-and-response protocol** for a client/server relationship, where a client requests a web page or other content (image, sound, video, etc.) from a server. The server looks for it and sends it back.

Second Purpose

2. understand **HTTP** as a "high level" protocol that sits on top of all the other protocols and internet systems we've learned about in the course.

- Text message conversation between the computers is being broken up into **TCP/IP** packets, and all the data gets sent as bits over wires and airwaves, taking different paths, and it gets interpreted reassembled at the end.

Each Internet Layer is a Level of Abstraction

The Internet works in "layers" and this is a perfect example of **abstraction** on the Internet, as one layer makes use of the functionality provided by the layer below it, without worrying about the details of how this functionality is achieved.

HTTP’s Functions

- HTTP doesn't have to worry about anything other than the text protocol of how HTTP works. The network software and devices on your and others' computers handle looking up **addresses**, breaking down **data**, **packeting**, **routing**, **transmission** and **interpretation** and **reassembly**. It's really amazing.

(1) User issues URL from a browser
http://host:port/path/file
(2) Browser sends a request message
GET URL HTTP/1.1
Host: host:port
.....
.....
(3) Server maps the URL to a file or program under the document directory.
(4) Server returns a response message
HTTP/1.1 200 OK
.....
.....
.....
(5) Browser formats the response and displays
Client (Browser)
HTTP (Over TCP/IP)
Server (@ host:port)

How 7-Layer Architecture Works?

- Understand the "Big Picture". You have to be able to recognize that these layers exist and to reasonably explain how they work together for common web page requests.

Layer	Function	Example
Application (7)	Services that are used with end user applications	SMTP,
Presentation (6)	Formats the data so that it can be viewed by the user Encrypt and decrypt	JPG, GIF, HTTPS, SSL, TLS
Session (5)	Establishes/ends connections between two hosts	NetBIOS, PPTP
Transport (4)	Responsible for the transport protocol and error handling	TCP, UDP
Network (3)	Reads the IP address form the packet.	Routers, Layer 3 Switches
Data Link (2)	Reads the MAC address from the data packet	Switches
Physical (1)	Send data on to the physical wire.	Hubs, NICS, Cable

The Internet Protocol **Stack**

- Make sure you have the "**HTTP and Abstraction on the Internet – Resource**" handout
- We want to think of the protocols as working in "**layers**".

This picture in the handout is a simplified version of what's known as the "Internet Protocol Stack." We've studied each layer separately but now you can begin to see how they work together.

UP/Down Stack

Call and Return of Utility Functions

- We imagine outgoing information going **DOWN** the stack and incoming information going **UP** the stack.
- All of this happens in the **network software** on each computer, whether that computer is your phone, a laptop, or a server like a DNS or web server (routers are a special case that only look at the IP layer).
- The important takeaway here is to understand that the system was constructed with layers of abstraction where each layer only needs to concern itself with its specific job, and then hands it off to another layer. This makes the very complicated task of digital network communication possible.

OSI Model

	Data	Layers
Host Layers	Data	Application Network Process to Application
	Data	Presentation Data Representation & Encryption
	Data	Session Inter-Host Communication
	Segments	Transport End-to-End Connections & Reliability
Media Layers	Packets	Network Path Determination & IP
	Frames	Data Link MAC & LLC (Physical Addressing)
	Bits	Physical Media,Signal,& Binary Transmission

5
Application Layer
I want to download a web page from this address: 192.168.1.102
Message
4
Transport Layer (TCP/UDP)
Source Port = 31,244
Destination Port = 80
I want to download a web page from this address: 192.168.1.102
Segment/Datagram
3
Network Layer (IP)
Source IP Addr = 192.168.1.101
Dest IP Addr = 192.168.1.102
Source Port = 31,244
Destination Port = 80
Message
Packet
2
Data Link Layer (MAC)
Source MAC Addr = 00:12:F1:1E:E8:93
Dest MAC Addr = 00:04:A3:4D:1C:73
Src IP Addr
Dest IP Addr
Src Port #
Dest Port #
Message
Frame
1
Physical Layer

Data Flow of the Internet Protocol Suite

HTTP

- At the top layer, for example, DNS just thinks is "speaking DNS" to some other computer - the DNS protocol does not even need to know how the other layers work It just relies on them doing their jobs.
- Have you ever seen the letters "HTTP" anywhere while using the internet?

  You see it in the URL of most websites http://code.org for example.
- Today we will focus our attention on HTTP, which is a protocol that sits at the same "layer" as **DNS** - right above TCP.

HTTP

- HTTP is an ASCII text based protocol.
- It's somewhat remarkable to note that many "high level" protocols, like HTTP, are just computers sending ASCII text messages back and forth. Each protocol simply defines the rules of the "conversation" between two machines.
- In the case of HTTP it is the protocol used for sending and receiving web pages and other web content.
- Today we'll look under the hood and see HTTP in action.

Video

The Internet: HTML and HTTP

Watch this video in Code Studio.

[about 7 minutes]

Activity

- Get in pairs and use the ”HTTP in Action” Worksheet
- Use the other handout as well:
  HTTP and Abstraction on the Internet - Resource

Activity

- Access the developer tools of their browser.
- Monitor the HTTP traffic generated by visiting a variety of websites.

Record your findings, using the resource as a guide.

Discussion

- What surprised you about the HTTP traffic you observed?
- What does it mean to say that high-level layers of the Internet use low-level layers “abstractly”?
- What other examples of abstraction have we seen in this course?
- Hint: Unit 1 is basically all about abstraction.

Practice

- Worksheet – HTTP In Action
- Assessment in Code Studio
- Reflection in Code Studio

Homework: Unit 1 – Lesson 13

Project: Internet Dilemmas

LESSON 7-8 [UNIT 2 CODE.ORG]

Network Equipment

LECTURE 1

Internet Service Provider

- **An Internet service provider** (ISP) is an organization that provides services for accessing, using, or participating in the Internet. Internet service providers may be organized in various forms, such as commercial, community-owned, non-profit, or otherwise privately owned.
- Internet services typically provided by ISPs include **Internet access**, **Internet transit**, **domain name registration**, **web hosting**, **Usenet service**, and **colocation**.

Application Software

SECTION 7-1

PoP #3
Tier 2 Network
IP Backbone
Transit
Internet
Tier 1 networks
Transit
Peering
IXP
PoP #2
PoP #1
Tier 2 ISP
Transit
Transit
Tier 3 Network
(multi-homed ISP)
Tier 3 Network
(single homed ISP)
PSTN
Central Office
PSTN
Metro-fiber
Ethernet
PSTN
Cable
Operator
DSLAM
Leased line
(T1/E1)
Leased line
(T3/E3)
Leased line
(T1/E1)
Cable plant
ADSL2
ADSL
POTS
Filter
Filter
Filter
PABX
Router
Cable Internet
customer
ADSL customer
(Triple play)
ADSL
customer
Dial-up
customer

DSL vs. Cable vs. Fiber vs. Satellite

SOHO Routers and Networks

SOHO: Small Office or Home

- A SOHO (Small Office/Home Office) usually consists of a business that is privately owned or an individual who is self-employed, so the term usually refers to both a small office space as well as a small number of employees.
- Since the workload for these types of businesses are often primarily on the internet, they require a local area network (LAN), which means their network hardware is structured specifically for that purpose.
- A SOHO network can be a mixed network of wired and wireless computers just like other local networks. Since these types of networks are meant for businesses, they also tend to include printers and sometimes voice over IP (VoIP) and fax over IP technology.
- A SOHO router is a model of broadband router built and marketed for use by such organizations. These are often the same routers used for standard home networking.

abg WiFi n ac
IP Phone
Printer
Internet
Firewall
Wireless Router
Laptop
Desktop Computer

SOHO Routers vs. Home Routers

- While h**ome** networks shifted to predominantly Wi-Fi configurations years ago, SOHO routers continued to feature wired Ethernet. In fact, many SOHO routers did not support Wi-Fi at all.
- Typical examples of Ethernet SOHO routers were common such as the TP-Link TL-R402M (4-port), TL-R460 (4-port), and TL-R860 (8-port).
- Another common feature of older routers was ISDN internet support. Small businesses relied on ISDN for internet connectivity as a faster alternative to dial-up networking.

SOHO LAN

SECTION 7-2

SOHO Routers vs. Home Routers

- Modern SOHO routers require most all the same functions as home broadband routers, and in fact, small businesses use the same models. Some vendors also sell routers with more advanced security and manageability features added, like the ZyXEL P-661HNU-Fx Security Gateway, a DSL broadband router with SNMP support.
- Another example of a popular SOHO router is the Cisco SOHO 90 Series, which is meant for up to 5 employees and includes firewall protection and VPN encryption.

Cable wall jack
Modem
Gateway
Phone

Other Types of SOHO Network Equipment

- Printers that combine the features of a basic printer with copy, scanning, and fax capability are popular with home office professionals. These so-called all-in-one printers include Wi-Fi support for joining to a home network.
- SOHO networks sometimes also operate an intranet web, email, and file server. These servers can be high-end PCs with added storage capacity (multi-drive disk arrays).

SOHO LAN Equipments

- WIFI Adapter
- Power Ethernet/Ethernet
- Modem
- Switch
- Router

Side View

It is this light and the black button underneath it.

WIFI

SECTION 7-3

WIFI: Wireless Ethernet

- The standard for wireless networking within a home or office. Also known as a "Wi-Fi" or "802.11" network, wireless Ethernet is the wireless counterpart to regular, wired Ethernet, which is also the standard for local networks. See 802.11 and wireless LAN.

802.1
Manage-
ment
802.2 LLC
802.3
MAC
802.3
PHY
802.5
MAC
802.5
PHY
802.11
MAC
802.11
PHY
802.15.1
802.15.4
802.16
MAC
+
PHY
Various data rates up to 100 Mbit/s and more

	802.11 (Legacy)	802.11b (Legacy)	802.11a (Legacy)	802.11g (Legacy)	802.11n (HT)	802.11ac (VHT)	802.11ax (HE)
Year Ratified	1997	1999	1999	2003	2009	2014	2019 (Expected)
Operating Band	2.4 GHz/IR	2.4 GHz	5 GHz	2.4 GHz	2.4/5 GHz	5 GHz	2.4/5 GHz
Channel BW	20 MHz	20 MHz	20 MHz	20 MHz	20/40 MHz	20/40/80/160 MHz	20/40/80/160 MHz
Peak PHY Rate	2 Mbps	11 Mbps	54 Mbps	54 Mbps	600 Mbps	6.8 Gbps	10 Gbps
Link Spectral Efficiency	0.1 bps/Hz	0.55 bps/Hz	2.7 bps/Hz	2.7 bps/Hz	15 bps/Hz	42.5 bps/Hz	62.5 bps/Hz
Max # SU Streams	1	1	1	1	4	8	8
Max # MU Streams	NA	NA	NA	NA	NA	4 (DL only)	8 (UL & DL)
Modulation	DSSS, FHSS	DSSS, CCK	OFDM	OFDM	OFDM	OFDM	OFDM, OFDMA
Max Constellation / Code Rate	DQPSK	CCK	64-QAM, 3/4	64-QAM, 3/4	64-QAM, 5/6	256-QAM, 5/6	1024-QAM, 5/6
Max # OFDM tones	NA	NA	64	64	128	512	2048
Subcarrier Spacing	NA	NA	312.5 kHz	312.5 kHz	312.5 kHz	312.5 kHz	78.125 kHz

Wi-Fi Connectivity Speed Timeline

Gigabit Wi-Fi As Primary Access

DSRC = Dedicated Short-Range Communications

Wired Ethernet

SECTION 7-4

Ethernet Cables and How They Work

- An Ethernet cable is one of the most common forms of network cable used on wired networks. Ethernet cables connect devices within a local area network, like PCs, routers, and switches.
- These physical cables are limited by the distance that they can stretch and still carry proper signals and by their durability.
- These limits are one reason there are different types of Ethernet cables optimized to perform certain tasks in particular situations.

Original IEEE	IEEE Shorthand Name	Informal Name(s)	Speed	Typical Cabling
802.3i	10BASE-T	Ethernet	10 Mbps	UTP
802.3u	100BASE-T	Fast Ethernet (Fast E)	100 Mbps	UTP
802.3z	1000BASE-X	Gigabit Ethernet (Gig E, GbE)	1000 Mbps	Fiber
802.3ab	1000BASE-T	Gigabit Ethernet (Gig E, GbE)	1000 Mbps	UTP
802.3ae	10GBASE-X	10 GbE	10 Gbps	Fiber
802.3an	10GBASE-T	10 GbE	10 Gbps	UTP
802.3ba	40GBASE-X	40GbE (40 GigE)	40 Gbps	Fiber
802.3ba	100GBASE-X	100GbE (100 GigE)	100 Gbps	Fiber

IEEE 802.3 PHYs
Rate (b/s)
1T
100G
10G
1G
100M
10M
1M
10^{-1}
1
10
10^2
10^3
10^4
10^5
Distance (m)
Key:
- Backplane
- Twin-axial
- Multimode Fibre
- Voice grade copper
- Co-axial
- Twisted pair
- Single-mode Fibre
- Single twisted pair
- Point to Multipoint Fibre
- Deprecated

Cat 7
50ft

FCoE

Connection Diagram

Power Line Ethernet

SECTION 7-5

Power-line Communication

- Power-line communication (PLC) carries data on a conductor that is also used simultaneously for AC electric power transmission or electric power distribution to consumers.
- It is also known as power-line carrier, power-line digital subscriber line (PDSL), mains communication, power-line telecommunications, or power-line networking (PLN).

Power-line Communication

- A wide range of power-line communication technologies are needed for different applications, ranging from home automation to Internet access which is often called broadband over power lines (BPL). Most PLC technologies limit themselves to one type of wires (such as premises wiring within a single building), but some can cross between two levels (for example, both the distribution network and premises wiring).
- Typically transformers prevent propagating the signal, which requires multiple technologies to form very large networks. Various data rates and frequencies are used in different situations.

Power
Cables
Router
Power Line Ethernet
To VoIP Configuration
Internet
VoIP Phone

Bedroom
MegaPlug™ AV Powerline Ethernet Network Adapter
Desktop
Den/Office
INTERNET
Router
MegaPlug™ AV Powerline Ethernet Network Adapter
Living Room
Gaming Console
DVR
HDTV
MegaPlug™ AV 4-Port Switch

VPN

SECTION 7-6

What Is a VPN? - Virtual Private Network

- A **virtual private network**, or VPN, is an encrypted connection over the Internet from a device to a network.
- The **encrypted** connection helps ensure that sensitive data is safely transmitted. It prevents unauthorized people from eavesdropping on the traffic and allows the user to conduct work remotely.
- VPN technology is widely used in corporate environments.

Existing Servers and PCs
Virtual Hub
SoftEther
VPN Client
Home
over TCP/UDP
VPN Tunnel
VPN Tunnel
over TCP/UDP
SoftEther
VPN Client
Hotel room
VPN Tunnel
Physical Ethernet
Adapter
(Local Bridging)
Physical Ethernet Switch
Headquarters
VPN Tunnel
Cascading
VPN Tunnel
Built-in VPN clients
Smartphones
Remote
Virtual Hub
Physical Ethernet
Adapter
(Local Bridging)
Physical Ethernet Switch
Branch Offices

Anti-Virus and Computer Security

LECTURE 8

Firewall

- In computing, a **firewall** is a network security system that monitors and controls incoming and outgoing network traffic based on predetermined security rules. A firewall typically establishes a barrier between a trusted internal network and untrusted external network, such as the Internet.
- Firewalls are often categorized as either **network firewalls** or **host-based firewalls**. Network firewalls filter traffic between two or more networks and run on network hardware. Host-based firewalls run on host computers and control network traffic in and out of those machines.

User configuration
L3-L7 connectivity policies
Microsoft Threat Intelligence
Known malicious IPs and FQDNs
Spoke 1
Spoke2
Spoke VNets
Central VNet
Azure Firewall
Threat intel, NAT, network and application traffic filtering rules allows inbound/outbound access
Traffic is denied by default
Azure to on-prem traffic filtering
On-premises

Powershell
Network Controller
Northbound Interface (REST APIs)
Distributed Firewall Manager
Southbound Interface
Policies
Policies
Host 1
Host 2
VM1
VM2
VM1
VM3
VM2
VM3
VNICs
VNICs
VSwitch
VSwitch
NIC
NIC
NIC
NIC
Gateway

Malware

- **Malware**, or “malicious software,” is an umbrella term that describes any malicious program or code that is harmful to systems.
- Hostile, intrusive, and intentionally nasty, malware seeks to invade, damage, or disable computers, computer systems, networks, tablets, and mobile devices, often by taking partial control over a device’s operations. Like the human flu, it interferes with normal functioning.
- **Malware** is all about making money off you illicitly. Although malware cannot damage the physical hardware of systems or network equipment (with one known exception—see the Google Android section below), it can steal, encrypt, or delete your data, alter or hijack core computer functions, and spy on your computer activity without your knowledge or permission.

WORMS
ROOTKITS
SPYWARE
TROJANS
CRIMEWARE
MALWARE
VIRUSES
ADWARE

Antivirus
free
COMODO
new