RSA Public Key Cryptosystem
provides an adequate level of security for most communication requirements

Originator of a particular RSA coding algorithm

Creation of the Public Key K_R

a. **construct p & q such that**

   i. **$p=3\bullet j + 2$**

   ii. **$q=3\bullet k + 2$**

   where j & k are sufficiently large so that the value K_R constructed below cannot be factored in any reasonable amount of time

b. **compute $K_R = p\bullet q$**

c. **publish K_R but keep p & q private, i.e., secret**

d. **compute $s = (1/3)(2\bullet(p-1)\bullet(q-1)+1)$ and keep it private, i.e., secret**

Sender of coded message

Using the ASCII table, encode the alphabetic ClearText into binary numbers, e.g., the ClearText "CSUN IT" encodes into ASCII as 0100 0011 0101 0011 0101 0101 0100 1110 0010 0000 0100 1001 0101 0100

Group the ASCII code into bundles of predetermined length, e.g.,

(0100 0011) (0101 0011) (0101 0101) (0100 1110) (0010 0000) (0100 1001) (0101 0100)

For the following discussion, assign T_1 = (0100 0011), T_2 = (0101 0011), …, T_7 = (0101 0100)

For each T_i compute the remainder of the whole division $\left(T_i^3 / K_R\right)$; for purposes of the following discussion, assign R_i to the said remainder. For each T_i transmit the remainder R_i to the Receiver.

Receiver of coded message
(also the Originator of the particular RSA coding algorithm being used by the Sender)

Using $s = (1/3)(2\bullet(p-1)\bullet(q-1)+1)$, compute the remainder of the whole division $\left(R_i^s / K_R\right)$; for purposes of the following discussion, assign M_i to the said remainder. In our example, M_1 = (0100 0011) = T_1

For all i, we have that $M_i = T_i$; hence all we have to do is ungroup the M_i and convert them from the ASCII code to ClearText.

RSA encryption provides adequate security for most transactions since the time required to factor K_R into p & q exceeds the life time of any person currently alive, <u>if p and q are of sufficient size</u>, e.g., p & q both exceed 127 digits in length.

Essential Characteristics

a. **Worm – an independent program that makes copies of itself and transmits them over the internet to infect other computer systems.**

b. **Virus – a fragment of a program that embeds itself into another program, thus effectively hiding itself; when the infected program is executed, the virus make copies of itself which are released to infect other programs on the same machine. When infected files are transferred to other machines, the virus is transferred as well, and thus infects the new machine.**

c. **Trojan Horse – a virus that hides in another useful program that performs operations unbeknownst to the user such as recording keystrokes, providing a security hole for system access or illicit access to the communication protocols.**

d. **Phishing –**
    - i. **Email – solicitation to participate in a joint venture required the victim to provide confidential financial &/or personal information that will be used to the detriment of the victim.**
    - ii. **Web-based – spoofing of a website to make it appear to be a valid commercial website, but designed to acquire confidential financial &/or personal information that will be used to the detriment of the victim.**
    - iii. **Face-to-Face – personal solicitation designed to acquire confidential financial &/or personal information that will be used to the detriment of the victim.**

RGB Color Code – selected colors expressed in hexadecimal & decimal values

Red	255:0:0	FF0000
Green	0:255:0	00FF00
Blue	0:0:255	0000FF
White	255:255:255	FFFFFF
Black	0:0:0	000000

- **describe the essential characteristic of all shades of gray**

n:n:n for ((n >= 0) & (n <= 255))
or
xxxxxx for ((x >= 0) & (x <= F))

The RGB for one shade of sea green is 99CC66 or 153:204:102.
The RGB for one shade of sea blue is 6699FF or 102:153:255.

What is the color specification that describes the color which is produced by changing the sea green color toward the sea blue color by

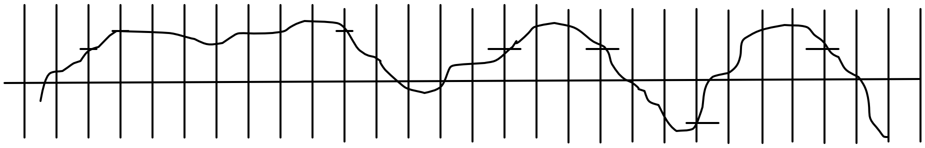
- **reducing the red component of the sea green color by 50% of the difference between the two colors**
- **reducing the green component of the sea green color by 25% of the difference between the two colors**
- **increasing the blue component of the sea green color by 25% of the difference between the two colors**

$\begin{array}{r} 153 \\ \underline{102} \\ 51 \end{array}$	$\begin{array}{r} 204 \\ \underline{153} \\ 51 \end{array}$	$\begin{array}{r} 102 \\ \underline{255} \\ -153 \end{array}$
Δ↓50%	Δ↓25%	Δ↑25%
51 / 2 ≈ 25	51 / 4 ≈ 13	-153 / 4 ≈ -38
$\begin{array}{r} 153 \\ \underline{-25} \\ 128 \end{array}$	$\begin{array}{r} 204 \\ \underline{-13} \\ 191 \end{array}$	$\begin{array}{r} 102 \\ \underline{+38} \\ 140 \end{array}$

Answer: 128:188:140

Describe how analog sound waves are converted to digital representation. What role does the Nyquist Rule play in the conversion process?

The analog sound wave must be sampled at regular intervals, i.e., the amplitude is measured at each interval and the two numbers are stored in digital form, i.e., sequences of bi-tuples <time interval, amplitude>.



Amplitude is a measure of the "Sound Pressure".

The Nyquist Rule states that the <u>Sampling Rate</u> must be at least twice as fast as the fastest frequency being recorded. That means that the distances between the intervals must be ½ of the shortest wave length being recorded.
Since

$$wavelength = \frac{1}{frequency}$$

then the Nyquist Rule can be restated as follows: the sampling intervals must be at least ½ of the shortest wavelength.

Since human perception of sound is limited to approximately 20,000 Hz or 20 GHz, the standardized digital audio recording frequency of 44,100 Hz captures most of the sound which can be heard by the normal human ear.

Describe the following search engine components

a. **Crawler – software robot that receives information about unvisited sites from the index, searches those sites retrieving key words and other significant information concerning the site such as URL's to other sites, and returns that information to the index.**

b. **Query Processor – provides**
   i. **a user front-end to receive a query request, i.e., key words**
   ii. **locate the requested information in the database**
   iii. **report back to the user which web pages contain those key words; the information reported is limited to that information that one of the may crawlers have located.**

c. **Index – consists of a**
   i. **Database of**
      - **searched sites which contains URL's, key words, and other significant information concerning each individual site.**
      - **URL's of identified but unvisited sites**
   ii. **Index manager which receives information from a crawler and classifies it into the proper form for inclusion into the database**

List the three most significant reasons that most web pages not indexed.

a. **web page that is dynamically created, e.g., Amazon.com pages designed for each individual purchaser reflecting that individuals purchase history**
b. **web page consists no text, e.g., only pictures or graphic images**
c. **no external web page contains a URL referencing the web page**
d. **no crawler has yet reached the web page**
e. **web page is password protected**

Describe how Google determines the ranking of pages returned by the search engine.

PageRank – count the links to a page; the more links there are to a page, the more relevant it must be. If page A links to page B, then we may consider it to be a vote by A for B. If many sites vote for B, it must be of great interest. If A has a high rank and it votes for B, then B assumes greater rank than if it was only voted for by lower ranking sites.

Describe the following with as much detail as possible:

a) URL

In computing, a **Uniform Resource Identifier** (**URI**) is a compact string of characters used to identify or name a resource on the Internet. The main purpose of this identification is to enable interaction with representations of the resource over a network, typically the World Wide Web, using specific protocols. URIs are defined in schemes defining a specific syntax and associated protocols.

http://python.ecs.csun.edu/compsci/faculty/putnam.html
i.e.,
server software://host name/directory/subdirectory/web page

b) IP Address

An Internet Protocol (**IP**) **address** is a numerical identification (logical address) that is assigned to devices participating in a computer network utilizing the Internet Protocol for communication between its nodes.[1] Although IP addresses are stored as binary numbers, they are usually displayed in human-readable notations,such as 208.77.188.166 (for IPv4), and 2001:db8:0:1234:0:567:1:1 (for IPv6). The role of the IP address has been characterized as follows: "A name indicates what we seek. An address indicates where it is. A route indicates how to get there." [2]

c) Domain Name System/Server

- The **Domain Name System** (**DNS**) is a hierarchical naming system for computers, services, or any resource participating in the Internet. It associates various information with domain names assigned to such participants. Most importantly, it translates human meaningful domain names to the numerical (binary) identifiers associated with networking equipment for the purpose of locating and addressing these devices world-wide. An often used analogy to explain the Domain Name System is that it serves as the "phone book" for the Internet by translating human-friendly computer hostnames into IP addresses. For example, *www.example.com* translates to *208.77.188.166*.
- The Domain Name System makes it possible to assign domain names to groups of Internet users in a meaningful way, independent of each user's physical location. Because of this, World-Wide Web (WWW) hyperlinks and Internet contact information can remain consistent and constant even if the current Internet routing arrangements change or the participant uses a mobile device. Internet domain names are easier to remember than IP addresses such as `208.77.188.166`(IPv4) or `2001:db8:1f70::999:de8:7648:6e8` (IPv6). People take advantage of this when they recite meaningful URLs and e-mail addresses without having to know how the machine will actually locate them.

- The Domain Name System distributes the responsibility for assigning domain names and mapping them to Internet Protocol (IP) networks by designating authoritative name servers for each domain to keep track of their own changes, avoiding the need for a central register to be continually consulted and updated.
- In general, the Domain Name System also stores other types of information, such as the list of mail servers that accept email for a given Internet domain. By providing a world-wide, distributed keyword-based redirection service, the Domain Name System is an essential component of the functionality of the Internet.
- Other identifiers such as RFID tags, UPC codes, International characters in email addresses and host names, and a variety of other identifiers could all potentially utilize DNS [1].
- The Domain Name System also defines the technical underpinnings of the functionality of this database service. For this purpose it defines the DNS protocol, a detailed specification of the data structures and communication exchanges used in DNS, as part of the Internet Protocol Suite (TCP/IP). The context of the DNS within the Internet protocols may be seen in the following diagram. The DNS protocol was developed and defined in the early 1980s and published by the Internet Engineering Task Force (cf. History).

d) TCP/IP

The **Transmission Control Protocol (TCP)** is one of the core protocols of the Internet Protocol Suite. TCP is so central that the entire suite is often referred to as "TCP/IP." Whereas IP handles lower-level transmissions from computer to computer as a message makes its way across the Internet, TCP operates at a higher level, concerned only with the two *end systems*, for example a Web browser and a Web server. In particular, TCP provides reliable, ordered delivery of a stream of bytes from one program on one computer to another program on another computer. Besides the Web, other common applications of TCP include e-mail and file transfer. Among its management tasks, TCP controls message size, the rate at which messages are exchanged, and network traffic congestion.

TCP provides a communication service at an intermediate level between an application program and the Internet Protocol (IP). That is, when an application program desires to send a large chunk of data across the Internet using IP, instead of breaking the data into IP-sized pieces and issuing a series of IP requests, the software can issue a single request to TCP and let TCP handle the IP details.

IP works by exchanging pieces of information called packets. A packet is a sequence of bytes and consists of a *header* followed by a *body*. The header describes the packet's destination and, optionally, the routers to use for forwarding—generally in the right direction—until it arrives at its final destination. The body contains the data which IP is transmitting. When IP is transmitting data on behalf of TCP, the content of the IP packet body is TCP payload.

Due to network congestion, traffic load balancing, or other unpredictable network behavior, IP packets can be lost or delivered out of order. TCP detects these problems, requests retransmission of lost packets, rearranges out-of-order packets, and even helps minimize network congestion to reduce the occurrence of the other problems. Once the TCP receiver has finally reassembled a perfect copy of the data originally transmitted, it passes that datagram to the application program. Thus, TCP abstracts the application's communication from the underlying networking details.

The **Internet Protocol** (**IP**) is a protocol used for communicating data across a packet-switched internetwork using the Internet Protocol Suite (TCP/IP).

IP is the primary protocol in the Internet Layer of the Internet Protocol Suite and has the task of delivering datagrams (packets) from the source host to the destination host solely based on their addresses. For this purpose the Internet Protocol defines addressing methods and structures for datagram encapsulation. The first major version of addressing structure, now referred to as Internet Protocol Version 4 (IPv4) is still the dominant protocol of the Internet, although the successor, Internet Protocol Version 6 (IPv6) is actively deployed worldwide.

Data from an upper layer protocol is encapsulated as packets/datagrams (the terms are basically synonymous in IP). Circuit setup is not needed before a host may send packets to another host that it has previously not communicated with (a characteristic of packet-switched networks), thus IP is a connectionless protocol. This is in contrast to Public Switched Telephone Networks that require the setup of a circuit before a phone call may go through (*connection-oriented* protocol).

Because of the abstraction provided by encapsulation, IP can be used over a heterogeneous network, i.e., a network connecting computers may consist of a combination of Ethernet, ATM, FDDI, Wi-Fi, token ring, or others. Each link layer implementation may have its own method of addressing (or possibly the complete lack of it), with a corresponding need to resolve IP addresses to data link addresses. This address resolution is handled by the Address Resolution Protocol (ARP) for IPv4 and Neighbor Discovery Protocol (NDP) for IPv6.

Number Systems

Decimal	Octal	Binary	Hexadecimal
0	0	0000	0
1	1	0001	1
2	2	0010	2
3	3	0011	3
4	4	0100	4
5	5	0101	5
6	6	0110	6
7	7	0111	7
8	10	1000	8
9	11	1001	9
10	12	1010	A
11	13	1011	B
12	14	1100	C
13	15	1101	D
14	16	1110	E
15	17	1111	F

Conversions

Octal ↔ Binary ↔ Hexadecimal

765_8 ↔ 111 110 101 ↔ 0001 1111 0101 ↔ $1F5_{16}$

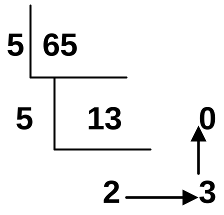
groups of three ↗ **groups of four** ↗

Conversions

Decimal → Base n

Successive Divisions of the Decimal Number by n, preserving the remainders

65_{10} → X_5



65_{10} → 230_5

Base n ➔ Decimal

Polynomial Expansion

230_5 ➔ $2\ 3\ 0_{5}$ ➔ $2*5^2 + 3*5^1 + 0*5^0$ ➔ $50 + 15 + 0$ ➔ 65_{10}

Base

Index

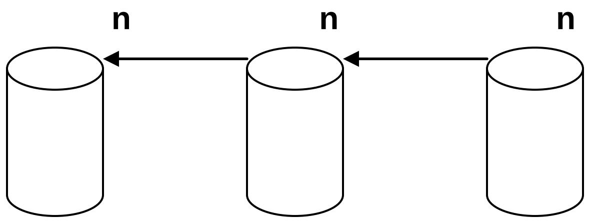
2 1 0

Coefficient * Index$^{\text{Base}}$ + Coefficient * Index$^{\text{Base}}$ + …

Addition

Base n ➔ (1) dump the bucket when it has n stones in it;

(2) add one stone to the bucket on the left



Subtraction

“Take Away”

When bucket is empty for Base n ➔

(1) remove one stone from the bucket on the left

(2) place n stones in the bucket that was empty