Scheme Of Evaluation
Internal Assessment Test 2 – March.2019

Sub:	Cryptography,Network Security & Cyber law							Code:	15CS61
Date:	15/ 04 / 2019	Duration:	90mins	Max Marks:	50	Sem:	VI	Branch:	ISE

Note: Answer Any Five Questions

Question #		Description	Marks Distribution		Max Marks
1	a)	**Perform encryption & decryption using El Gamal algorithm between Alice & Bob.** **Note :p =283, q=47, g=60, r=36, Bob's public key α=216,private key b=7,m=101** • Encryption • Decryption	3M 3M	6M	10 M
	b)	**Draw MAC generation and encryption in CCMP.** • Diagram	4M	4M	
2	a)	**Classify different types of worm in detail.** • Types of worms	10M	10M	10 M
3	a)	**Interpret the working of firewall along with its types and issues.** • Working of firewall • Types • Issues	5M 3M 2M	10M	10 M
4	a)	**Describe Intrusion Detection System with the types.** • IDS • Types	4M 2M	6M	10 M
	b)	**Show how distributed denial of service is detected.** • Formula	4M	4M	
5	a)	**Discuss about SAML assertion with an example.** • SAML explanation • Example	2M 4M	6M	10 M

	b)	**Quote secure electronic records and secure electronic signature.** • Secure electronic records • Secure electronic signature	2M 2M	4M	
6	a)	**Find the session key used by Alice and Bob using Diffie-Hellman key change. (note: prime g = 23, primitive root a=5, secret integer of (XA)=4 & B(XB)=3)** • Finding session key	5M	5M	10 M
	b)	**Quote attribution, acknowledgement & dispatch of electronic records.** • Attribution & acknowledgement • Dispatch of electronic records	3M 2M	5M	
7	a)	**Explain in brief about SHA1 algorithm.** • Overall SHA1 diagram • Per round diagram • Steps	3M 3M 4M	10M	10 M
8	a)	**Whether e-mail from a particular host can be blocked. If yes then block a host NIST to send email to CMRIT web server.** • Whether e-mail can be blocked? • Explanation	1M 9M	10M	10 M

Answers

1. A . Perform encryption & decryption using El Gamal algorithm between Alice & Bob. Note :p =283, q=47, g=60, r=36, Bob's public key α=216,private key b=7,m=101 (6)

- INPUT: Domain parameters (p=283,g=60)

 INPUT: Bob's public key, α=216
 INPUT: encoded message, m=101
1. Alice chooses a random r=36
2. Alice computes $c1=g^k \bmod p=60^{36} \bmod 283=78$
3. Alice computes $c2=(m\ \alpha^k) \bmod p=101 \cdot 216^{36} \bmod p=218$
4. Alice sends ciphertext ($c1$,$c2$)=(78,218) to Bob

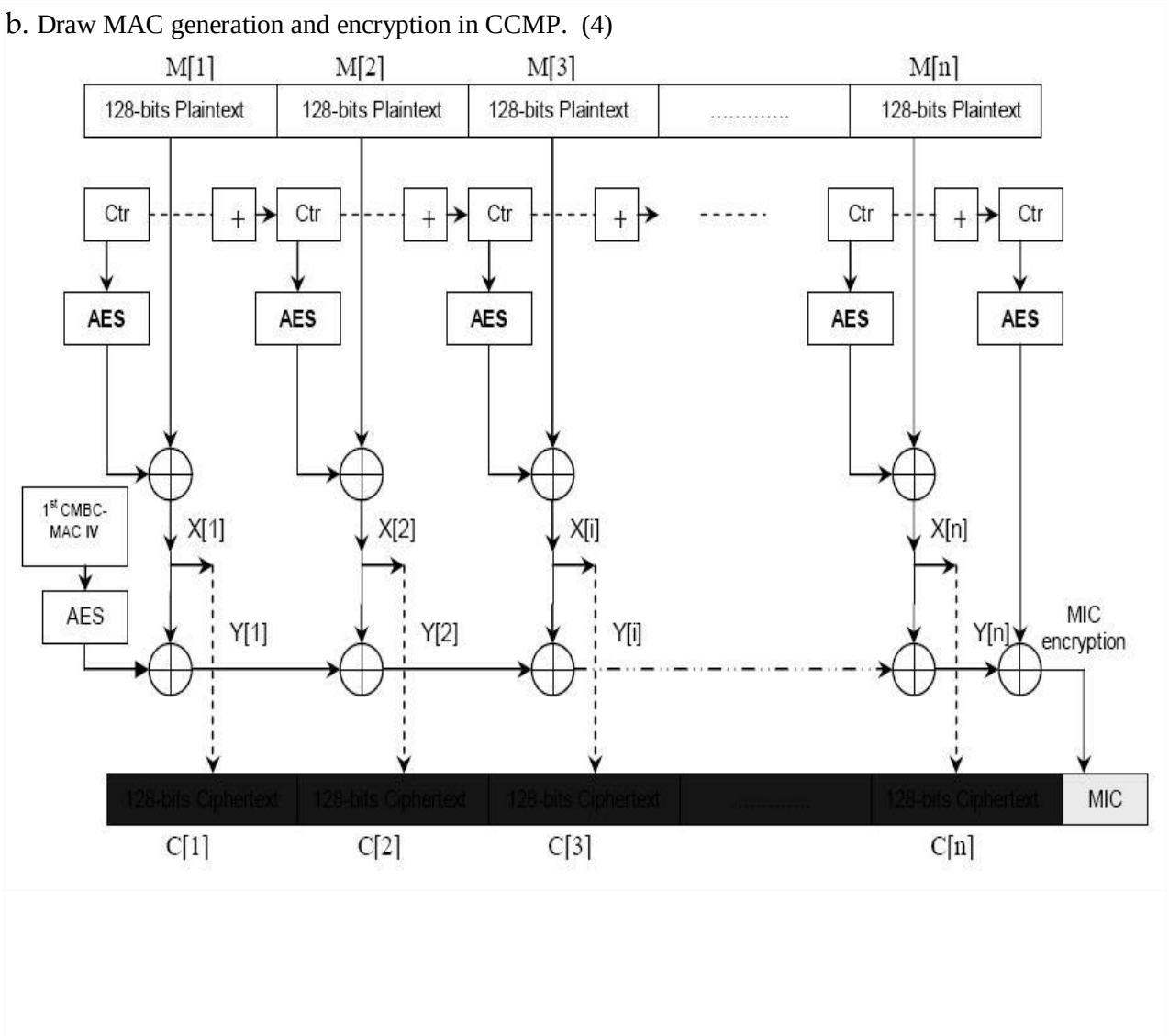
Example of ElGamal decryption by Bob

- INPUT: Domain parameters (p=283,g=60)

 INPUT: Bob's private key, a=7
 INPUT: ciphertext ($c1$,$c2$)=(78,218)

 Bob computes $m=(c1^{-a})*c2 \bmod p$ $=78283-7-1 \cdot c2=116 \cdot 218 \bmod p$ $(c1^{-a} = c1^{p-a-1})$ $=101$

b. Draw MAC generation and encryption in CCMP. (4)



2. A) Illustrate different types of worm in detail.

Internet Scanning worms

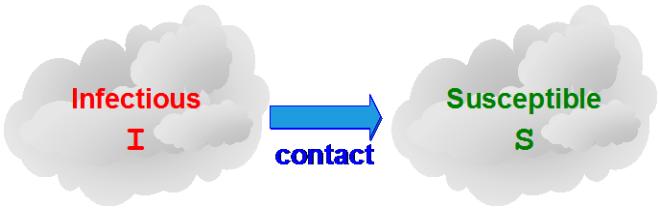
- One characteristic of internet scanning worms is that they are self-activated.
- Case Studies:
- Code Red
- Buffer overflow vulnerability was discovered in the Microsoft IIS web server.
- Slammer
- Buffer overflow vulnerability on the Microsoft SQL server 2000.

Simple Epidemic Model

- "infectious" hosts: continuously infect others.
- "removed" hosts in epidemic area:
  - Recover and immune to the virus.
  - Dead because of the disease.
- "removed" hosts in computer area:
  - Patched computers that are clean and immune to the worm.
  - Computers that are shut down or cut off from worm's circulation.

Epidemic modeling introduction

- Homogeneous assumption:
  - Any host has the equal probability to contact any other hosts in the system.
  - Number of contacts $\propto \mathbf{I} \times \mathbf{S}$



Modelling an Internet Worm

- Simple Epidemic Model

$$S(t) = N - I(t)$$

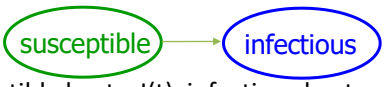
$$dI(t)/dt = \beta I(t)S(t) = \beta I(t)[N - I(t)],$$

$$\alpha = \beta N$$

I(t)= Number of Infectious Hosts at time t
S(t)= Number of Susceptible Hosts at time t
N= number of hosts in the system
β = pair wise infection rate
α = worm's infection rate (average number of probes sent out by an infected host per unit time)

Deterministic epidemic models — Simple epidemic model

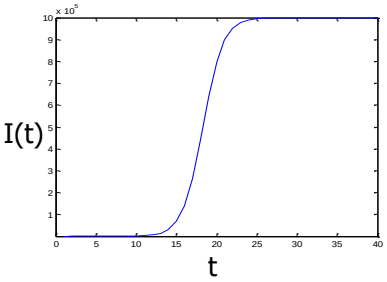
- State transition:



N: population; S(t): susceptible hosts; I(t): infectious hosts

$$dI(t)/dt = \beta S(t) I(t)$$

$$S(t) + I(t) = N$$



- I(t) ↔ S(t) symmetric
- Problems:
  - Constant infection rate β
  - No "removed" state.

Modelling an Internet Worm

- General Epidemic Model
  - Kermack-McKendrick Epidemic Model

$$\begin{aligned} dI(t)/dt &= \beta I(t)S(t) - \gamma I(t) \\ dU(t)/dt &= \gamma I(t) \\ N &= I(t) + U(t) + S(t) \end{aligned}$$

$$\rho \equiv \gamma/\beta.$$

Epidemic threshold theorem – major outbreak occurs if S(0) > ρ

U(t) = number of previously removed ones at time t
gamma = removal rate of infected hosts
ρ = epidemic threshold

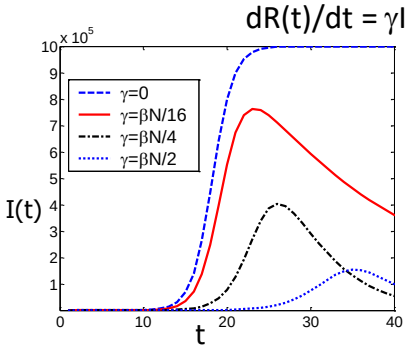
Deterministic epidemic models —Kermack-McKendrick epidemic model

- State transition: susceptible → infectious → removed

R(t): removed from infectious; γ removal rate

$$dI(t)/dt = \beta S(t) I(t) - dR(t)/dt$$

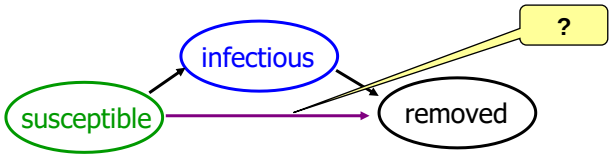
$$dR(t)/dt = \gamma I(t); \quad S(t) + I(t) + R(t) = N$$



- Epidemic threshold:
  - No outbreak if S(0) < γ / β
  - Major Out Break if S(0)> γ / β
- Problems:
  - Constant infection rate β
  - susceptible → removed

Consider human countermeasures

- Human countermeasures:
  - Clean and patch: download cleaning program, patches.
  - Filter: put filters on firewalls, gateways.
  - Disconnect computers.
- Reasons for:
  - Suppress most new viruses/worms from outbreak.
  - Eliminate virulent viruses/worms eventually.
- Removal of both susceptible and infectious hosts.



Topological Worms

- Machines vulnerable to those worms can be represented by graphs
- Types:
- Email worms
- P2P worms

E-mail

- An Email-Worm (also known as a mass-mailer or less commonly an Internet worm) distributes copies of itself in an infectious e-mail attachment. Often, these infected e-mails are sent to e-mail addresses that the worm harvests from files on an infected computer.

P2P Worms

- P2P Worms spread via peer-to-peer file sharing networks (such as Kazaa, Grokster, EDonkey, FastTrack, Gnutella, etc.).
- Most of these worms work in a relative simple way: in order to get onto a P2P network, all the worm has to do is copy itself to the file sharing directory, which is usually on a local machine. The P2P network does the rest: when a file search is conducted, it informs remote users of the file and provides services making it possible to download the file from the infected computer.
- There are also more complex P2P-Worms that imitate the network protocol of a specific file sharing system and responds positively to search queries; a copy of the P2P-Worm is offered as a match.

Web Worms & Case Study

Web worms are executed in browsers which run on diverse h/w/OS platform

- **Cross-Site Scripting (also known as XSS)** is one of the most common application-layer web attacks. XSS vulnerabilities target scripts embedded in a page that are executed on the client-side (in the user's web browser) rather than on the server-side. XSS in itself is a threat that is brought about by the internet security weaknesses of client-side scripting languages, such as HTML and JavaScript. The concept of XSS is to manipulate client-side scripts of a web application to execute in the manner desired by the malicious user. Such a manipulation can embed a script in a page that can be executed every time the page is loaded, or whenever an associated event is performed.
- XSS is the most common security vulnerability in software today. This should not be the case as XSS is easy to find and easy to fix. XSS vulnerabilities can have consequences such as tampering and sensitive data theft.
- **Key Concepts of XSS**

- XSS is a web-based attack performed on vulnerable web applications.
- In XSS attacks, the victim is the user and not the application.
- In XSS attacks, malicious content is delivered to users using JavaScript.
- **Explaining Cross-Site Scripting**
- An XSS vulnerability arises when web applications take data from users and dynamically include it in web pages without first properly validating the data. XSS vulnerabilities allow an attacker to execute arbitrary commands and display arbitrary content in a victim user's browser. A successful XSS attack leads to an attacker controlling the victim's browser or account on the vulnerable web application. Although XSS is enabled by vulnerable pages in a web application, the victims of an XSS attack are the application's users, not the application itself. The potency of an XSS vulnerability lies in the fact that the malicious code executes in the context of the victim's session, allowing the attacker to bypass normal security restrictions.

Mobile Malware

Introduction

APIs can also be used by malware to , for eg, read a confidential document on the smartphone & ship it to the attacker as an MMS attachment

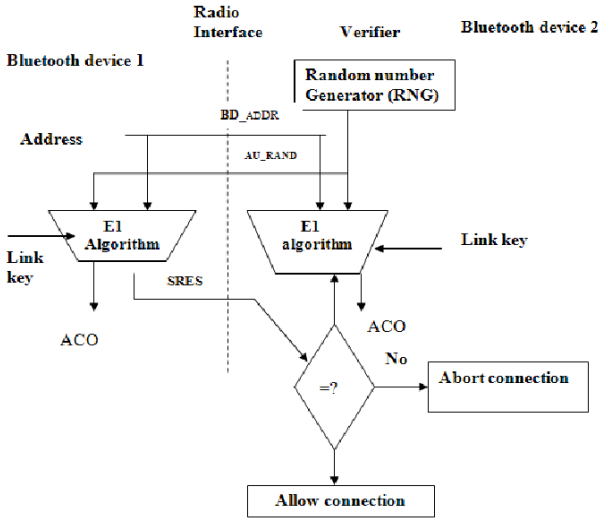
Bluetooth

Vulnerabilites in the bluetooth protocol itself

Discovery & User authorization

- Keep discoverable mode in mobiles
- BD_ADDR(bluetooth address)
- Object Exchange(OBEX) protocol is used to transfer images, business cards
- User authorization is usually required before a file can be accepted by his smartphone.
- Each user selects a PIN which varies b/w 4&16 characters long but 4-characters are typically used.

Link –Layer Security



3 (a) Interpret the working of firewall along with its types and issues. (10)

- Firewall acts as a security guard controlling access b/w internal, protected n/w & an external untrusted n/w based on a given security policy
- Firewall=firmware
- Defence in depth

> **Firewall Functionality (5 marks)**

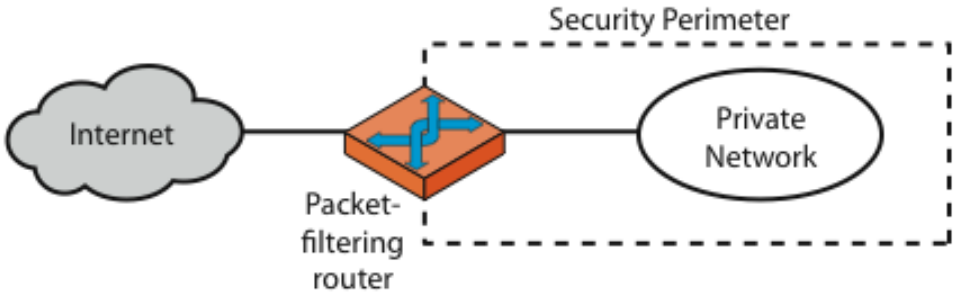
- Access ctrl: configured with a rule set
- Address/ Port Translation: N/w address translation- private to public address translation
- Logging: Log all attacker activity
- Authentication, caching,etc. : web proxy- authenticates internal users attempting to access an external service

Set of policies to access ctrl list

- A rule specifies the action to be taken as a function of
- i) the packet's source IP address & port no'
- ii) the packet's destinationIP address & port no'
- iii) the transport protocol (TCP/UDP)
- iv) the packet's direction- incoming/outgoing
- Permissive Policy:
- Permit all packets except those that are explicitly forbidden

- Restrictive Policy:
- Drop all packets except those that are explicitly permitted

Firewalls – Packet Filters



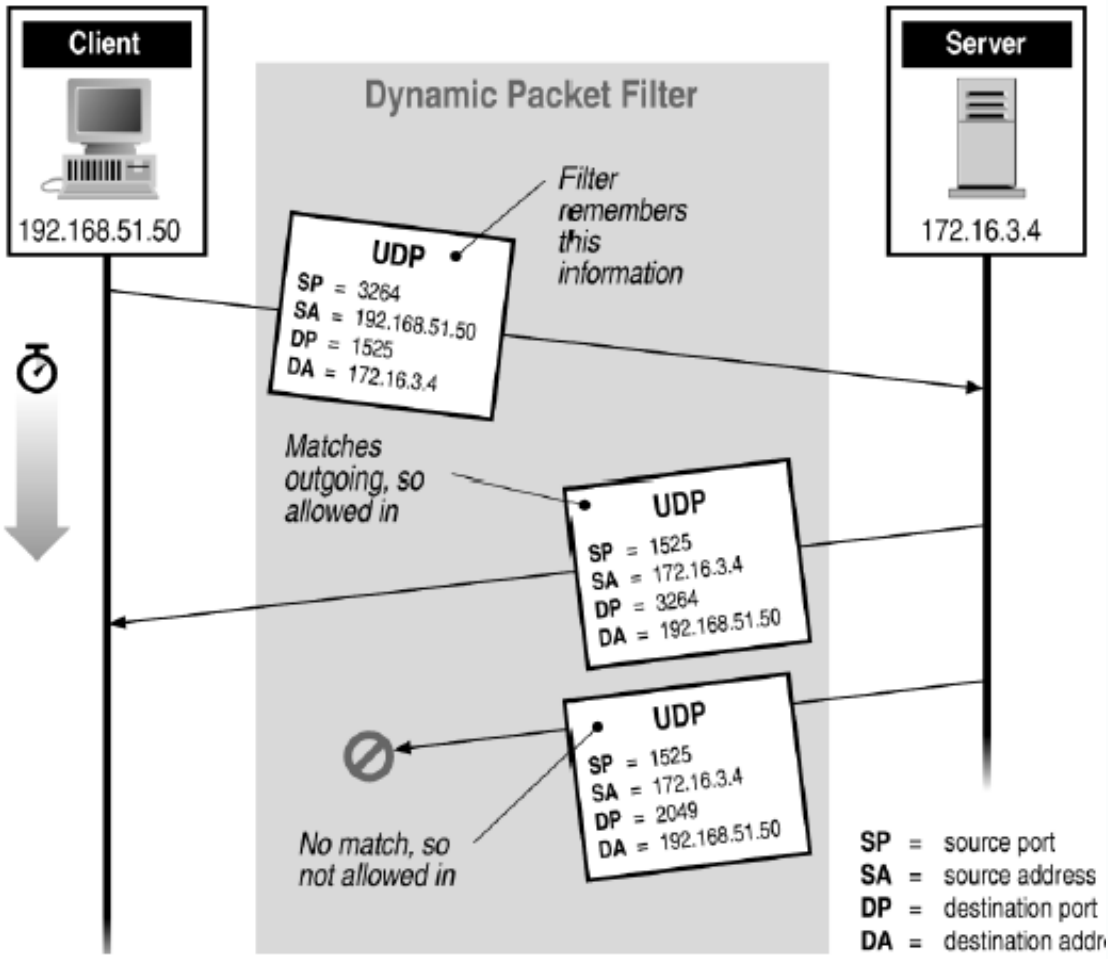
(a) Packet-filtering router

> **Firewalls – Packet Filters (3 marks)**

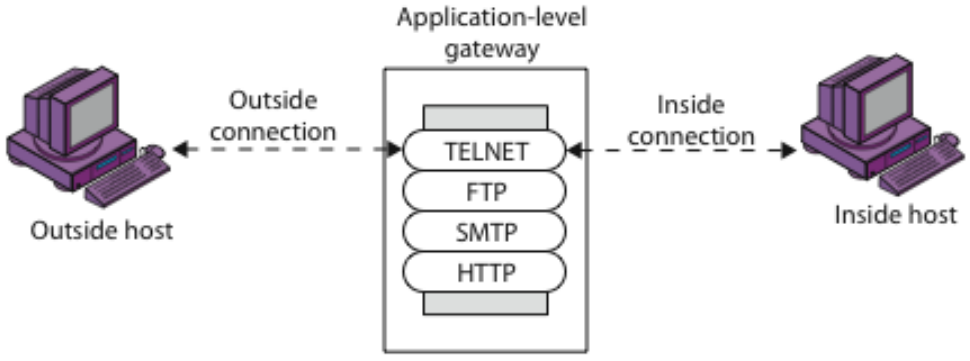
- Simplest of components
- Uses transport-layer information only
  - IP Source Address, Destination Address
  - Protocol/Next Header (TCP, UDP, ICMP, etc)
  - TCP or UDP source & destination ports
  - TCP Flags (SYN, ACK, FIN, RST, PSH, etc)
  - ICMP message type
- Examples
  - DNS uses port 53
    - No incoming port 53 packets except known trusted servers

Firewalls – Stateful Packet Filters

- Traditional packet filters do not examine higher layer context
  - ie matching return packets with outgoing flow
- Stateful packet filters address this need
- They examine each IP packet in context
  - Keep track of client-server sessions
  - Check each packet validly belongs to one
- Hence are better able to detect bogus packets out of context



Firewalls - Application Level Gateway (or Proxy)



(b) Application-level gateway

Application-Level Filtering

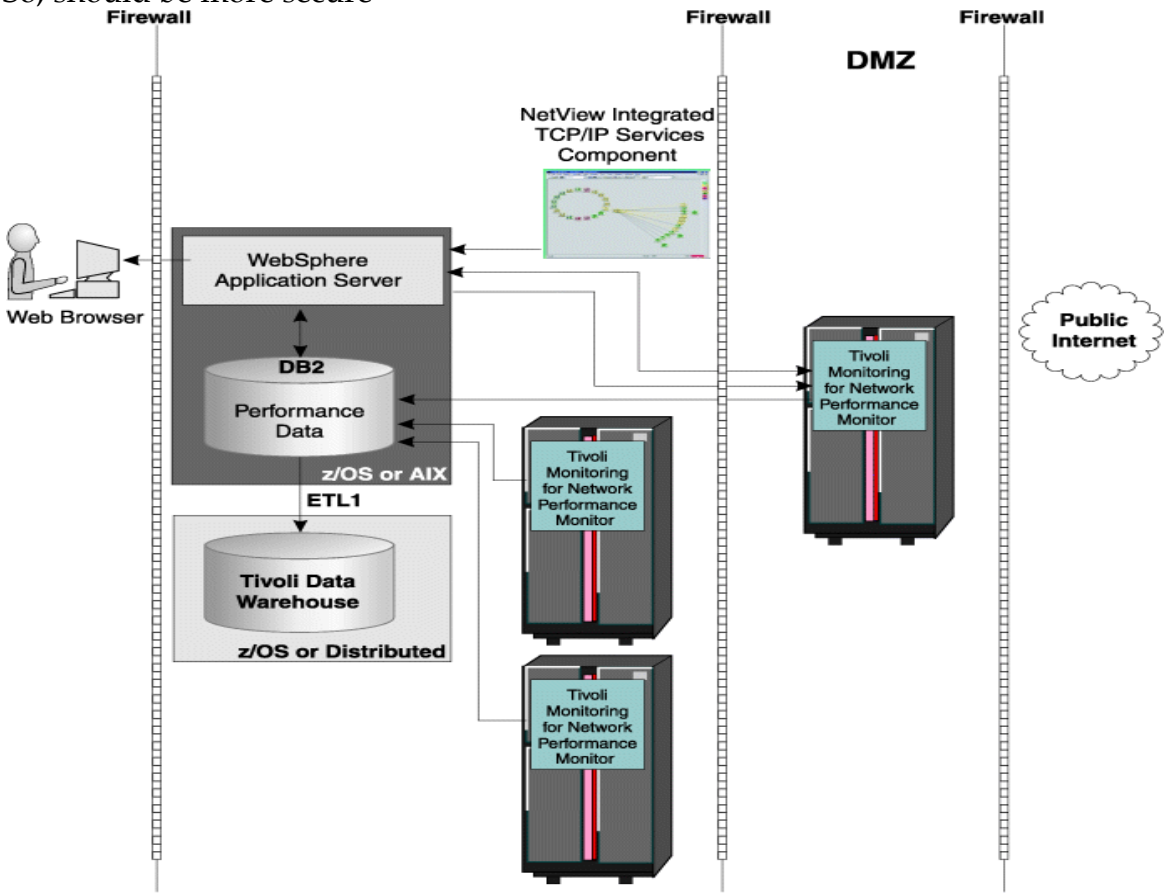
- Has full access to protocol
  - user requests service from proxy

- proxy validates request as legal
- then actions request and returns result to user

- Need separate proxies for each service
  - E.g., SMTP (E-Mail)
  - NNTP (Net news)
  - DNS (Domain Name System)
  - NTP (Network Time Protocol)
  - custom services generally not supported

> **Practical Issues (2 marks)**

- Placement of firewalls
- Demilitarized Zone:
- Area b/w two firewalls
- Accessible to both internal n/w & internet
- So, should be more secure



Firewall Configuration

- E.g
- Ruleset for firewall

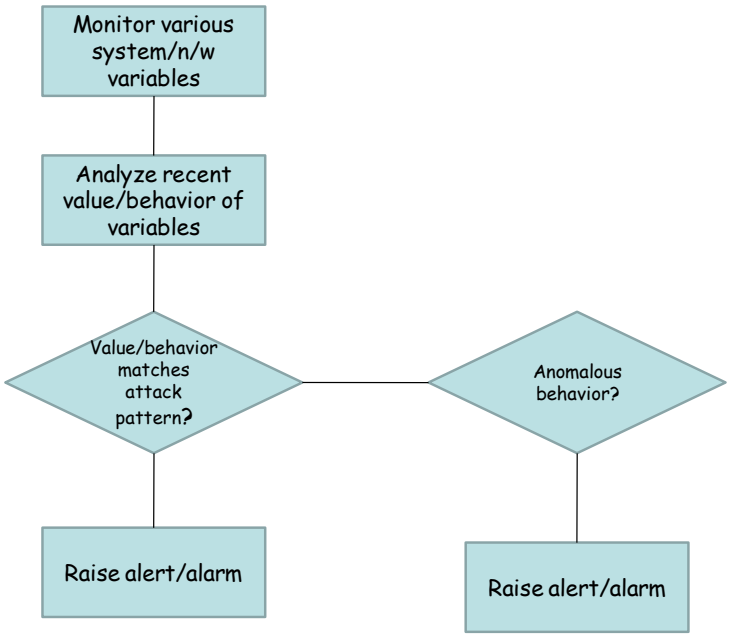
From IP Add	From Port	To IP addr	To Port	Protocol	Action
*	*	Internal	*	*	Drop
User	*	Int_Mail_S	25	SMTP	Accept
User	*	Proxy	80	HTTP	Accept
*	*	DMZ-2	*	*	Drop

4 (a) **Infer Intrusion Detection System with the types.**

> Intrusion:

- The act of gaining unauthorized access to a s/m
- E.g
- Unauthorized login to a s/m
- Worm infection
- Injection of spyware
- Prevention: Diabetes & high blood pressure: Taking healthy food, regular health checkup

Working:



> Types of IDS

- 1. Anomaly vs Signature based IDS
- 2. Host-based vs N/w based IDS
- **Signature Based IDS**
- Compares known threat signatures to observed events to identify incidents.
- This is very effective at detecting known threats but largely ineffective at detecting unknown threats and many variants on known threats.
- Signature-based detection cannot track and understand the state of complex communications, so it cannot detect most attacks that comprise multiple events.
- **Examples:**
- A telnet attempt with a username of "root", which is a violation of an organization's security policy
- An e-mail with a subject of "Free pictures!" and an attachment filename of "freepics.exe", which are characteristics of a known form of malware
- **Anomaly-based detection:** sample network activity to compare to traffic that is known to be normal.
- When measured activity is outside baseline parameters or clipping level, IDPS will trigger an alert.
- Anomaly-based detection can detect new types of attacks.
- Requires much more overhead and processing capacity than signature-based .
- May generate many false positives.

Host-Based IDSs

- Use OS auditing and monitoring mechanisms to find applications taken over by attacker

- Log all relevant system events (e.g., file/device accesses)
- Monitor shell commands and system calls executed by user applications and system programs
    - Pay a price in performance if every system call is filtered

- Problems:
    - User dependent: install/update IDS on all user machines!
    - If attacker takes over machine, can tamper with IDS binaries and modify audit logs
    - Only local view of the attack

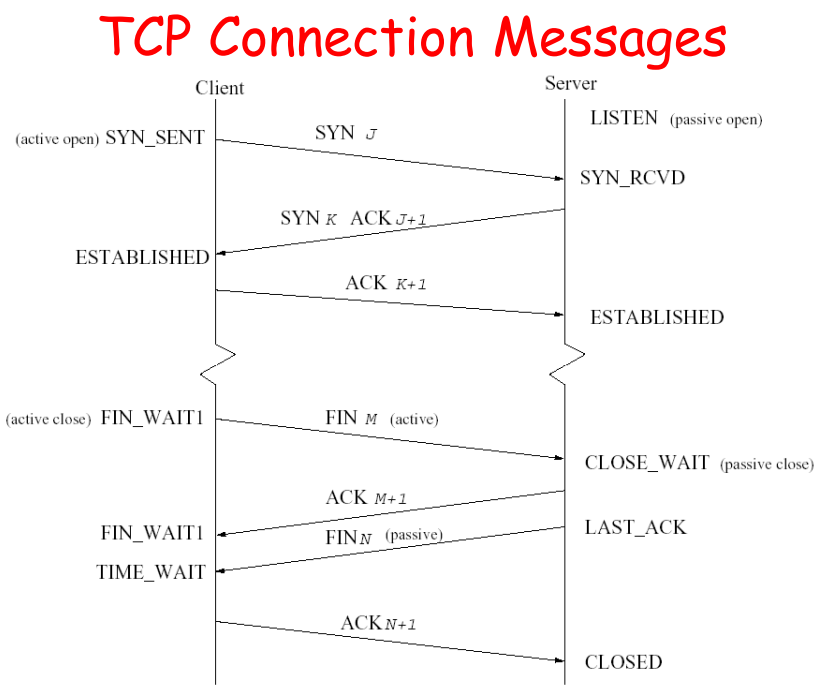
N/w based IDS

- Deploying sensors at strategic locations
    - For example, Packet sniffing via *tcpdump* at routers
- Inspecting network traffic
    - Watch for violations of protocols and unusual connection patterns
    - Look into the packet payload for malicious code
- Limitations
    - Cannot execute the payload or do any code analysis
    - Even DPI gives limited application-level semantic information
    - Record and process huge amount of traffic
    - May be easily defeated by encryption, but can be mitigated with encryption only at the gateway/proxy

b) **Quote how distributed denial of service is detected.**

- **Denial-of-service (DoS) attack** aims at **disrupting the authorized use** of networks, systems, or applications
    - by sending messages which exhaust service provider's resources (network bandwidth, system resources, application resources)
- **Distributed denial-of-service (DDoS) attacks** employ multiple (dozens to millions) compromised computers to perform a coordinated and widely distributed DoS attack
- **Victims** of (D)DoS attacks
    - service-providers (in terms of time, money, resources, good will)
    - legitimate service-seekers (deprived of availability of service itself)
    - Zombie systems(Penultimate and previous layers of compromised systems in DDoS)

Detection

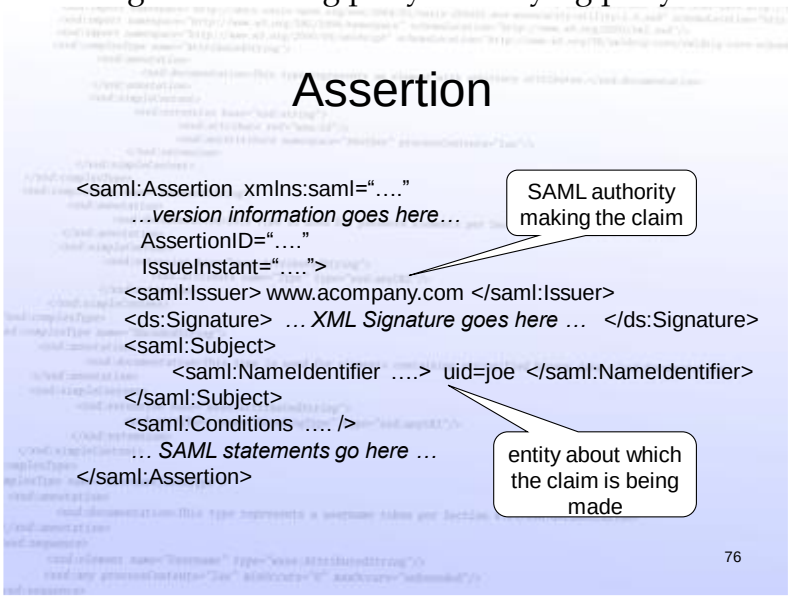


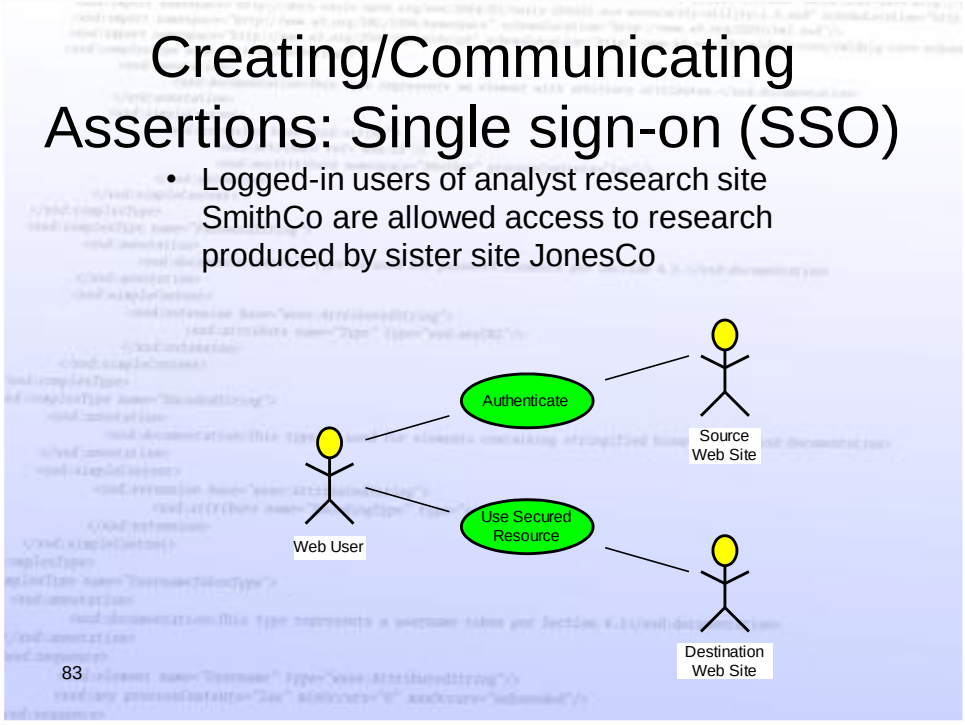
- Utilize SYN-FIN pair behavior
- Or SYNACK – FIN
- Can be both on client or server side
- However, RST violates SYN-FIN behavior
  - Passive RST: transmitted upon arrival of a packet at a closed port (usually by servers)
  - Active RST: initiated by the client to abort a TCP connection (e.g., Ctrl-D during a telnet session)
    - Often queued data are thrown away
  - So SYN-RST_{active} pair is also normal
- Si= # of SYN packet arrivals in the i-th observation interval
- Fi= # of FIN packet arrivals in the i-th observation interval
- Di=normalized difference b/w # of SYN & FIN packets in the i-th observation interval, i.e Di=Si-Fi/Fi
- T= Threshold for detection
- Time series, D1,D2,D3,…

Raise an alert if the most recently computed detection variable Di exceeds the threshold i.e Di>T1

5 (a) **Discuss about SAML assertion with an example.**

- Permissions management data is currently handled in mostly proprietary ways, among tightly coupled modules in a single security domain.
- Web is loosely coupled, consisting of many security domains. A standard is needed to govern the transfer of assertions between domains.
- Cookies don't do it –
  - Cookie (signed with server's private key) can be used for re-authentication at a particular server, but is of no use at a different server
- Cross domain authentication currently requires proprietary SSO software
- SAML intended as a Web standard that will supersede proprietary software
- SAML must be used in the context of a trust relationship between asserting and relying parties
- **Example**: statement "Bill has access to resource *X*" may be of no use unless we know that Bill is at the other end of the line
- Trust relationship is established using mechanisms such as SSL, signatures, encryption, etc.
- this security framework is not part of SAML
  Assertion
- A set of statements (claims) made by a SAML authority (asserting party)
  - **Authentication statement**: subject was authenticated using a particular technique at a particular time
  - **Attribute statement**: particular attribute values are associated with the subject
  - **Authorization decision statement**: subject is authorized to perform certain actions
  - E.g SP1 asserting party SP2 relying party





> E.g

- Signing into some tour websites (Smart tours) filter the range of flights according to the requirements
- Which in turn will direct to some authorized website (Jet Air)

(b) **Quote secure electronic records and secure electronic signature.**

Where any security procedure has been applied to an electronic record at a specific point of time. Then such record shall be deemed to be a secure electronic record from such point of time to the time of verification.

Secure digital signature.

If, by application of a security procedure agreed to by the parties concerned, it can be verified that a digital signature, at the time it was affixed, was-

- Unique to the subscriber affixing it.
- Capable of identifying such subscriber.
- Created in a manner or using a means under the exclusive control of the subscriber and is linked to the electronic record to which it relates in such a manner that if the electronic record was altered the digital signature would be invalidated. Then such digital signature shall be deemed to be a secure digital signature.

Security procedure.

The Central Government shall for the purposes of this Act prescribe the security procedure having regard to commercial circumstances prevailing at the time when the procedure was used, including-

- the nature of the transaction.
- the level of sophistication of the parties with reference to their technological capacity.
- the volume of similar transactions engaged in by other parties.
- the availability of alternatives offered to but rejected by any party.
- the cost of alternative procedures, and
- the procedures in general use for similar types of transactions or communications.

6 (a) **Find the session key used by Alice and Bob using Diffie-Hellman key change. (note: prime g = 23, primitive root a=5, secret integer of (XA)=4 & B(XB)=3)**

1. Alice and Bob agree to use a modulus $p = 23$ and base a = 5 (which is a primitive root modulo 23).
2. Alice chooses a secret integer $a = 4$, then sends Bob $A = g^a \bmod p$
   - $A = 5^4 \bmod 23 = 4$
3. Bob chooses a secret integer $b = 3$, then sends Alice $B = g^b \bmod p$
   - $B = 5^3 \bmod 23 = 10$
4. Alice computes $s = B^a \bmod p$
   - $s = 10^4 \bmod 23 = 18$
5. Bob computes $s = A^b \bmod p$
   - $s = 4^3 \bmod 23 = 18$
6. Alice and Bob now share a secret (the number 18).

Both Alice and Bob have arrived at the same value s, because, under mod p.

b) Quote attribution, acknowledgement & dispatch of electronic records. (5)

Attribution of electronic records.

An electronic record shall be attributed to the originator -

- if it was sent by the originator himself.
- by a person who had the authority to act on behalf of the originator in respect of that electronic record, or
- by an information system programmed by or on behalf of the originator to operate automatically.

Acknowledgment of receipt.

- Where the originator has not agreed with the addressee that the acknowledgment of receipt of electronic record be given in a particular form or by a particular method, an acknowledgment may be given by -
    - any communication by the addressee, automated or otherwise, or
    - any conduct of the addressee, sufficient to indicate to the originator that the electronic record has been received.
- Where the originator has stipulated that the electronic record shall be binding only on receipt of an acknowledgment of such electronic record by him, then unless acknowledgment has been so received, the electronic record shall be deemed to have been never sent by the originator.
- Where the originator has not stipulated that the electronic record shall be binding only on receipt of such acknowledgment, and the acknowledgment has not been received by the originator within the time specified or agreed or, if no time has been specified or agreed to within a reasonable time, then the originator may give notice to the addressee stating that no acknowledgment has been received by him and specifying a reasonable time by which the acknowledgment must be received by him and if no acknowledgment is received within the aforesaid time limit he may after giving notice to the addressee, treat the electronic record as though it has never been sent.

Time and place of dispatch and receipt of electronic record.

- Save as otherwise agreed to between the originator and the addressee, the dispatch of an electronic record occurs when it enters a computer resource outside the control of the originator.
- Save as otherwise agreed between the originator and the addressee, the time of receipt of an electronic record shall be determined as follows, namely :-
    - if the addressee has designated a computer resource for the purpose of receiving electronic records -
        - receipt occurs at the time when the electronic, record enters the designated computer resource, or
        - if the electronic record is sent to a computer resource of the addressee that is not the designated computer resource, receipt occurs at the time when the electronic record is retrieved by the addressee.
    - if the addressee has not designated a computer resource along with specified timings, if any, receipt occurs when the electronic record enters the computer resource of the addressee.
- Save as otherwise agreed to between the originator and the addressee, an electronic record is deemed to be dispatched at the place where the originator has his place of business, and is deemed to be received at the place where the addressee has his place of business.
- The provisions of sub-section (2) shall apply notwithstanding that the place

where the computer resource is located may be different from the place where the electronic record is deemed to have been received under sub-section (3).

- For the purposes of this section -
- if the originator or the addressee has more than one place of business, the principal place of business, shall be the place of business.
- if the originator or the addressee does not have a place of business, his usual place of residence shall be deemed to be the place of business.
- "Usual place of residence", in relation to a body corporate, means the place where it is registered.

7. **A . Explain in brief about SHA1 algorithm.**

SHA originally designed by NIST & NSA in 1993 was revised in 1995 as SHA-1 US standard for use with DSA signature scheme standard is FIPS 180-1 1995, also Internet RFC3174 the algorithm is SHA, the standard is SHS based on design of MD4 with key differences produces 160-bit hash values 2005 results on security of SHA-1 raised concerns on its use in future applications

Secure Hash Algorithm Std-SHA-512

- Digest Length=***160 bit***
  - I/P Text=512 bit
  - Sub Block size=32bit
  - 512/32=16 total Sub blocks
  - No. Of Rounds=4
  - Iteration per round=***20***
  - Chaining Variable = 5*32=160
  - K[t] constant= ***Where t=0 to 79***
  - O/P-> four 32 bit blocks

1. ***Padding:*** Length of the message is 64 bits short of multiple of 512 after padding.
2. ***Append*** a 64-bit ***length*** value of original message is taken.
3. ***Divide the input into 512-bit blocks***
4. ***Initialise IV*** 5-word (160-bit) buffer (A,B,C,D,E) to

   (***A***=01 23 45 67,

   B=89 AB CD EF,

C=FE DC BA 98,

D=76 54 32 10,

E=C3 D2 E1 F0)

5. ***Process Blocks*** now the actual algorithm begins. message in 16-word (512-bit) chunks:

   Copy IV into single register for storing temporary intermediate as well as the final results.

   Divide the current 512-bit blocks into 16 sub-blocks, each consisting of 32 bits.

   - Has No. Of Rounds=4, each round consisting of 20 ***bit /step iteration*** operations on message block & buffer
   - expand 16 words into 80 words(20*4) by mixing & shifting.K[t] constant= ***Where t=0 to 79***
   - Form new buffer value by adding output to input.

   6. output hash value is the final buffer value

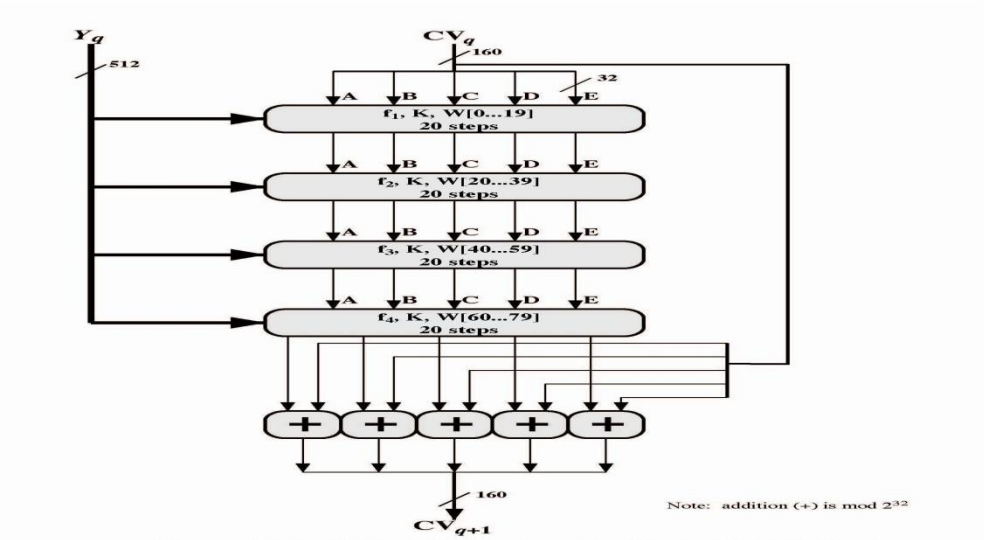
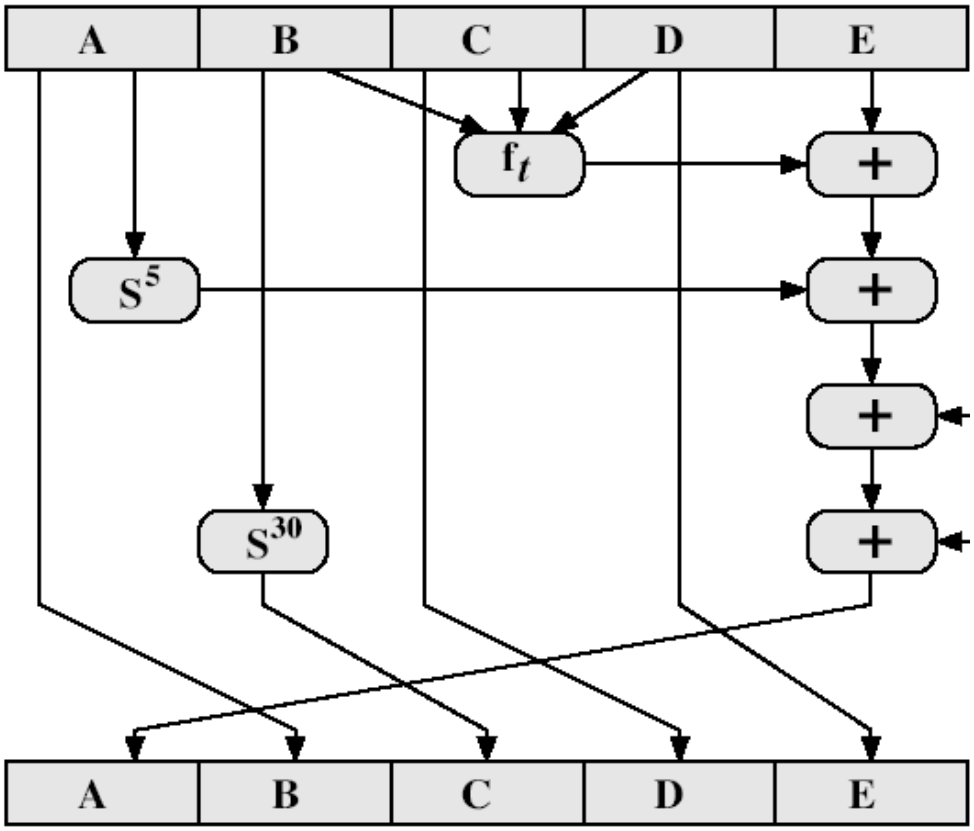


Figure 12.5 SHA-1 Processing of a Single 512-bit Block (SHA-1 Compression Function)



ABCDE=(F[t]+E+S5(A)+W[t]+K[t]),>>>Shift right by 1 bit for next iteration

each round has 20 steps which replaces the 5 buffer words thus:

(A,B,C,D,E) <-(E+f(t,B,C,D)+(A<<5)+W_t+K_t),A,(B<<30),C,D)

ABCDE refer to the 5 words of the buffer

t is the step number

f(t,B,C,D) is nonlinear function for round

W_t is derived from the message block

K_t is a constant value

S^t circular left shift of 32 bit sub-block by t bits

Process F(t) in each SHA-1 round

- where g can be expressed as:

ROUND 1: (b AND c) OR ((NOT b) AND (d)) same as MD5

ROUND 2: b XOR c XOR d

ROUND 3: (b AND c) OR (b AND d) OR (c AND d)

ROUND 4: b XOR c XOR d

Creation of 80-word input W_t

Adds redundancy and interdependence among message blocks

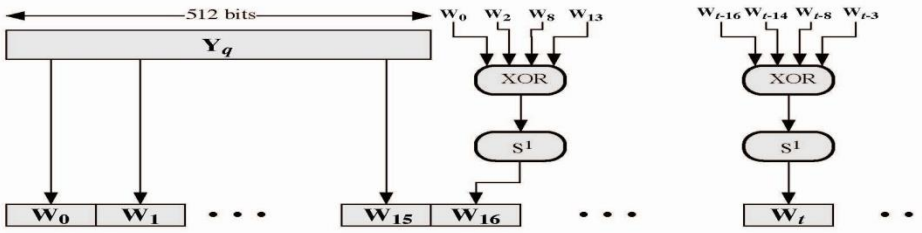


Figure 12.7 Creation of 80-word Input Sequence for SHA-1 Processing of

8.a. Whether e-mail from a particular host can be blocked. If yes then block a host NIST to send email to CMRIT web server.
Yes, e-mail from a particular host can be blocked.

From IP Add	From Port	To IP addr	To Port	Protocol	Action
NIST	25	Internal	*	SMTP	Drop