KESHAV MEMORIAL INSTITUTE OF TECHNOLOGY

(Approved by AICTE & Govt of T.S and Affiliated to JNTUH)
3-5-1026, Narayanaguda, Hyderabad-29. Ph: 040-23261407

Department Of Information Technology

Audit Form

for

IS ..... Course File 2016-17

Faculty Name: S.V. VASANTHA

SNO	Topic	Audit 1	Audit 2	Audit 3
1	V / M / PEO / POs /PSOs	✓		
2	Course Structure	✓		
3	Course syllabus	✓		
4	Course Outcomes (CO)	✓	Tonomony Levels	
5	Mapping	✓		
6	Academic Calendar	✓		
7	Time table(class &I individual)	Pending		
8	Lesson plan	✓		
9	Topics beyond syllabus (TBS)	✓		
10	Web references	✓		not in proper place
11	Lecture notes	✓		not ✓
12	Power point presentations / Videos			✓ CD ?
13	University Question papers	✓		
14	Internal Question papers with Key	objective key paper pending		Co and Cvcl ?
15	Assignment Question papers	assignment notes pending		
16	Tutorial evidence	✓		
17	Result Analysis to identify			

18	Weak and advanced learners			
19	Result Analysis at the end of the course			
20	Course Assessment	pending		
21	Guest talks, field visits etc.	———		
22	Attendance register	Pending		
23	Course file (Digital form)			

IQAC Committee In charge

KESHAV MEMORIAL INSTITUTE OF TECHNOLOGY

(Approved by AICTE & Govt of T.S and Affiliated to JNTUH)
3-5-1026, Narayanaguda, Hyderabad-29. Ph: 040-23261407

CURRICULUM DELIVERY PROGRESS
AUDIT FORM

Faculty Name: **Academic year:**
Branch: **Class:**

Mid wise split up-of syllabus

Mid-I Units Covered	Periods		Date Of		Remarks
	Required	Taken	Start	Completion	
Unit-I					
Unit-II					

Mid-II Units Covered	Periods		Date Of		Remarks
	Required	Taken	Start	Completion	
Unit-III					
Unit-IV					
Unit -V					

Signature of the Faculty **HOD** **Principal**

KESHAV MEMORIAL INSTITUTE OF TECHNOLOGY
(Approved by AICTE & Govt of T.S and Affiliated to JNTUH)
3-5-1026, Narayanaguda, Hyderabad-29. Ph: 040-23261407

Department Of Information Technology

Vision & Mission of Department

Vision of the Department:

Producing quality graduates trained in the latest software technologies and related tools and striving to make India a world leader in software products and services.

Mission of the Department:

- Mission of the Department: To create a faculty pool which has a deep understanding and passion for algorithmic thought process.
- To impart skills beyond university prescribed to transform students into a well-rounded IT professional.
- To inculcate an ability in students to pursue Information technology education throughout their lifetime by use of multimodal learning platform including e-learning, blended learning, remote testing and skilling.
- Exposure to different domains, paradigms and exposure to the financial and commercial underpinning of the modern business environment through the entrepreneur development cell.
- To encourage collaboration with various organizations of repute for research, consultancy and industrial interactions.
- To create socially conscious and emotionally mature individuals with awareness on India's challenges, opportunities, their role and responsibility as engineers towards achieving the goal of job and wealth creation.

KESHAV MEMORIAL INSTITUTE OF TECHNOLOGY
(Approved by AICTE & Govt of T.S and Affiliated to JNTUH)
3-5-1026, Narayanaguda, Hyderabad-29. Ph: 040-23261407

Department Of Information Technology

PROGRAM OUTCOMES (POs)

1. **Engineering knowledge**: Apply the knowledge of mathematics, science, engineering fundamentals, and an engineering specialization to the solution of complex engineering problems.
2. **Problem analysis**: Identify, formulate, review research literature, and analyze complex engineering problems reaching substantiated conclusions using first principles of mathematics, natural sciences, and engineering sciences.
3. **Design/development of solutions**: Design solutions for complex engineering problems and design system components or processes that meet the specified needs with appropriate consideration for the public health and safety, and the cultural, societal, and environmental considerations.
4. **Conduct investigations of complex problems**: Use research-based knowledge and research methods including design of experiments, analysis and interpretation of data, and synthesis of the information to provide valid conclusions.
5. **Modern tool usage**: Create, select, and apply appropriate techniques, resources, and modern engineering and IT tools including prediction and modelling to complex engineering activities with an understanding of the limitations.
6. **The engineer and society**: Apply reasoning informed by the contextual knowledge to assesssocietal, health, safety, legal and cultural issues and the consequent responsibilities relevant to the professional engineering practice.
7. **Environment and sustainability**: Understand the impact of the professional engineering solutions in societal and environmental contexts, and demonstrate the knowledge of, and need for sustainable development.
8. **Ethics**: Apply ethical principles and commit to professional ethics and responsibilities and norms of the engineering practice.

9. **Individual and team work**: Function effectively as an individual, and as a member or leader in diverse teams, and in multidisciplinary settings.
10. **Communication**: Communicate effectively on complex engineering activities with the engineering community and with society at large, such as, being able to comprehend and write effective reports and design documentation, make effective presentations, and give and receive clear instructions.
11. **Project management and finance**: Demonstrate knowledge and understanding of the engineering and management principles and apply these to one's own work, as a member and leader in a team, to manage projects and in multidisciplinary environments.
12. **Life-long learning**: Recognize the need for, and have the preparation and ability to engage in independent and life-long learning in the broadest context of technological change.

KESHAV MEMORIAL INSTITUTE OF TECHNOLOGY
(Approved by AICTE & Govt of T.S and Affiliated to JNTUH)
3-5-1026, Narayanaguda, Hyderabad-29. Ph: 040-23261407

Department Of Information Technology

PROGRAM SPECIFIC OUTCOMES (PSOs)

PSO1: An ability to analyze the common business functions to design and develop appropriate Information Technology solutions for social upliftments.

PSO2: Shall have expertise on the evolving technologies like Mobile Apps, CRM, ERP, Big Data, etc.

KESHAV MEMORIAL INSTITUTE OF TECHNOLOGY
(Approved by AICTE & Govt of T.S and Affiliated to JNTUH)
3-5-1026, Narayanaguda, Hyderabad-29. Ph: 040-23261407

Vision of the Institution:

To be the fountain head of latest technologies,
producing highly skilled, globally competent engineers.

Mission of the Institution:

- To provide a learning environment that helps students to enhance problem solving skills, be successful in their professional lives and to prepare students to be lifelong learners through multi model platforms and educating them about their professional, and ethical responsibilities.
- To establish Industry Institute Interaction to make students ready for the industry.
- To provide exposure to students to the latest tools and technologies in the area of hardware and software.
- To promote research based projects/activities in the emerging areas of technology convergence.
- To encourage and enable students to not merely seek jobs from the industry but also to create new enterprises
- To induce in the students a spirit of nationalism which will enable the student to develop and understand
  India's problems and to encourage them to come up with effective solutions for the same
- To support the faculty in their endeavors to accelerate their learning curve in order to continue to deliver excellent service to students

Grams: "TECHNOLOGY"
E Mail: dapjntuh@gmail.com

R13

Phone: Off: +91–40–23156115
Fax: +91–40–23158665

JAWAHARLAL NEHRU TECHNOLOGICAL UNIVERSITY HYDERABAD

(Established by Andhra Pradesh Act No.30 of 2008)

Kukatpally, Hyderabad – 500 085, Andhra Pradesh (India)

B. TECH. INFORMATION TECHNOLOGY /COMPUTER SCIENCE AND TECHNOLOGY

I YEAR

Code	Subject	L	T/P/D	C
	English	2	-	4
	Mathematics – I	3	1	6
	Mathematical Methods	3	-	6
	Engineering Physics	3	-	6
	Engineering Chemistry	3	-	6
	Computer Programming	3	-	6
	Engineering Drawing	2	3	6
	Computer Programming Lab	-	3	4
	Engineering Physics / Engineering Chemistry Lab	-	3	4
	English Language Communication Skills Lab	-	3	4
	IT Workshop / Engineering Workshop	-	3	4
	Total	**19**	**16**	**56**

II YEAR I SEMESTER

Code	Subject	L	T/P/D	C
	Probability and Statistics	4	-	4
	Mathematical Foundations of Computer Science	4	-	4
	Data Structures	4	-	4
	Digital Logic Design and Computer Organization	4	-	4
	Electronic Devices and Circuits	4	-	4
	Basic Electrical Engineering	4	-	4
	Electrical and Electronics Lab	-	3	2
	Data Structures Lab	-	3	2
	Total	**24**	**6**	**28**

II YEAR II SEMESTER

Code	Subject	L	T/P/D	C
	Principles of Programming Languages	4	-	4
	Database Management Systems	4	-	4
	Java Programming	4	-	4
	Environmental Studies	4	-	4
	Data Communication	4	-	4
	Design and Analysis of Algorithms	4	-	4
	Java Programming Lab	-	3	2
	Database Management Systems Lab	-	3	2
	Total	**24**	**6**	**28**

III YEAR I SEMESTER

Code	Subject	L	T/P/D	C
	Automata and Compiler Design	4	-	4
	Linux Programming	4	-	4
	Software Engineering	4	-	4
	Operating Systems	4	-	4
	Computer Networks	4	-	4
	Managerial Economics and Financial Analysis	4	-	4
	Operating Systems Lab	-	3	2
	Computer Networks Lab (Through Linux)	-	3	2
	Total	**24**	**6**	**28**

III YEAR II SEMESTER

Code	Subject	L	T/P/D	C
	Web Technologies	4	-	4
	OPEN ELECTIVE Human Values and Professional Ethics Intellectual Property Rights Disaster Management	4	-	4
	Object Oriented Analysis and Design	4	-	4
	Data Warehousing and Data Mining	4	-	4
	Software Testing Methodologies	4	-	4
	Cloud Computing	4	-	4
	Data Mining and Web Technologies Lab	-	3	2
	Advanced English Communication Skills Lab	-	3	2
	Total	**24**	**6**	**28**

IV YEAR I SEMESTER

Code	Subject	L	T/P/D	C
	Information Security	4	-	4
	Design Patterns	4	-	4
	Mobile Application Development	4	-	4
	Information Retrieval Systems	4	-	4
	ELECTIVE – I Wireless Networks and Mobile Computing Image Processing and Pattern Recognition Soft Computing Semantic Web and Social Networks Operations Research	4	-	4
	ELECTIVE – II Software Project Management Computer Graphics Human Computer Interaction Scripting Languages Computer Forensics	4	-	4
	Case Tools and Software Testing Lab	-	3	2
	Mobile Applications Development Lab	-	3	2
	Total	**24**	**6**	**28**

IV YEAR II SEMESTER

Code	Subject	L	T/P/D	C
	Management Science	4	-	4
	ELECTIVE III Web Services E – Commerce Middleware Technologies Ad hoc and Sensor Networks	4	-	4
	ELECTIVE IV Multimedia & Rich Internet Applications Artificial Intelligence Storage Area Networks Machine Learning	4	-	4
	Industry Oriented Mini Project	-	-	2
	Seminar	-	6	2
	Project Work	-	15	10
	Comprehensive Viva	-	-	2
	Total	**12**	**21**	**28**

Note: All End Examinations (Theory and Practical) are of three hours duration.
T Tutorial L – Theory P – Practical/Drawing C Credits

JAWAHARLAL NEHRU TECHNOLOGICAL UNIVERSITY HYDERABAD

IV Year B.Tech. IT/CST-I Sem	L	T/P/D	C
	4	-/-/-	4

(A70522) INFORMATION SECURITY

Objectives:

- Explain the objectives of information security
- Explain the importance and application of each of confidentiality, integrity, authentication and availability
- Understand various cryptographic algorithms.
- Understand the basic categories of threats to computers and networks
- Describe public-key cryptosystem.
- Describe the enhancements made to IPv4 by IPSec
- Understand Intrusions and intrusion detection
- Discuss the fundamental ideas of public-key cryptography.
- Generate and distribute a PGP key pair and use the PGP package to send an encrypted e-mail message.
- Discuss Web security and Firewalls

UNIT – I

Attacks on Computers and Computer Security: Introduction, The need for security, Security approaches, Principles of security, Types of Security attacks, Security services, Security Mechanisms, A model for Network Security

Cryptography: Concepts and Techniques: Introduction, plain text and cipher text, substitution techniques, transposition techniques, encryption and decryption, symmetric and asymmetric key cryptography, steganography, key range and key size, possible types of attacks.

UNIT – II

Symmetric key Ciphers: Block Cipher principles & Algorithms(DES, AES, Blowfish), Differential and Linear Cryptanalysis, Block cipher modes of operation, Stream ciphers, RC4,Location and placement of encryption function, Key distribution **Asymmetric key Ciphers:** Principles of public key cryptosystems, Algorithms(RSA, Diffie-Hellman,ECC), Key Distribution

UNIT – III

Message Authentication Algorithms and Hash Functions: Authentication requirements, Functions, Message authentication codes, Hash Functions, Secure hash algorithm, Whirlpool, HMAC, CMAC, Digital signatures, knapsack algorithm **Authentication Applications:** Kerberos, X.509

Authentication Service, Public – Key Infrastructure, Biometric Authentication

UNIT – IV

E-Mail Security: Pretty Good Privacy, S/MIME **IP Security:**IP Security overview, IP Security architecture, Authentication Header, Encapsulating security payload, Combining security associations, key management

UNIT – V

Web Security: Web security considerations, Secure Socket Layer and Transport Layer *Security, Secure electronic transaction* **Intruders, Virus and Firewalls:** Intruders, Intrusion detection, password management, Virus *and related threats, Countermeasures, Firewall design principles, Types of* firewalls **Case Studies on Cryptography and security:** Secure Inter-branch Payment Transactions, Cross site Scripting Vulnerability, Virtual Elections

TEXT BOOKS:

1. Cryptography and Network Security : William Stallings, Pearson Education,4th Edition.
2. Cryptography and Network Security : Atul Kahate, Mc Graw Hill, 2nd Edition.

REFERENCE BOOKS:

1. Cryptography and Network Security: C K Shyamala, N Harini, Dr T R Padmanabhan, Wiley India, 1st Edition.
2. Cryptography and Network Security : Forouzan Mukhopadhyay, Mc Graw Hill, 2nd Edition.
3. Information Security, Principles and Practice: Mark Stamp, Wiley India.
4. Principles of Computer Sceurity: WM.Arthur Conklin, Greg White, TMH.
5. Introduction to Network Security: Neal Krawetz, CENGAGE Learning.
6. Network Security and Cryptography: Bernard Menezes, CENGAGE Learning.

Outcomes:

- Student will be able to understand basic cryptographic algorithms, message and web authentication and security issues.
- Ability to identify information system requirements for both of them such as client and server.
- Ability to understand the current legal issues towards information security.

Grams: "TECHNOLOGY"
E Mail: dap@jntuh.ac.in
dapjntuh@gmail.com

Phone: Off: +91-40-23156115
Fax: +91-40-23158665

JAWAHARLAL NEHRU TECHNOLOGICAL UNIVERSITY HYDERABAD

(Established by Andhra Pradesh Act No. 30 of 2008)

Kukatpally, Hyderabad – 500 085, Telangana (India)

Dr. B.N. BHANDARI
Ph.D (IIT KGP).
Professor of Elect. & Commn. Engg., &
Director,
Academic & Planning

Lr.No:A1/ Academic Calendar/B. Tech & B. Pharm./2016 Dated: 10.06.2016

To

The Principals of Constituent Colleges.
The Principals of Affiliated Engineering/Pharmacy colleges of JNTUH.

Sir,

Sub:- JNTUH, Hyderabad – Academic & Planning –Approval of Academic Calendar for **II, III and IV years of B. Tech and B. Pharmacy I & II Semester** for the academic year 2016-17 – Communicated.

***

The Academic Calendar for **II, III and IV years of B. Tech and B. Pharmacy I & II Semester** (Regular) for the academic year 2016-17 is approved. The details are as follows:

I Semester:

Description	Period	Duration
Commencement of Class Work	13.06.2016	
First Spell of Instructions	13.06.2016 to 06.08.2016	(8 w)
First Mid Examinations Timings: 10.00 am to 12.00 Noon (Forenoon Session)02.00 pm to 4.00 pm (Afternoon Session)	08.08.2016 to 13.08.2016	(1 w)
Second Spell instructions	16.08.2016 to 04.10.2016	(7 w)
Dussehra Holidays	05.10.2016 to 12.10.2016	(1 w)
Supplementary Examinations	13.10. 2016 to 26.10.2016	(2w)
Second Spell continuation	27.10.2016 to 03.11.2016	(1 w)

Second Mid Examinations Timings: 10.00 am to 12.00 Noon (Forenoon Session)02.00 pm to 4.00 pm (Afternoon Session)	04.11.2016 to 10.11.2016	(1w)
Preparations and Practical Examinations	11.11.2016 to 17.11.2016	(1w)
End semester Examinations	18.11.2016 to 01.12.2016	(2w)

II Semester

Description	Period	Duration
Commencement of class work	**02.12.2016**	
First Spell of Instructions	02.12.2016.to 27.01.2017	(8 w)
First Mid Examinations Timings: 10.00 am to 12.00 Noon (Forenoon Session)02.00 pm to 4.00 pm (Afternoon Session)	28.01.2017 to 04.02.2017	(1w)
Supplementary Examinations	05.02.2017 to 18.02.2017	(2w)
Second Spell of Instructions	19.02.2017 to 14.04.2017	(8 w)
Second Mid Examinations Timings:10.00 am to 12.00 Noon (Forenoon Session) 02.00 pm to 4.00 pm (Afternoon Session)	15.04.2017 to 21.04.2017	(1w)
Preparation and Practical Examinations	22.04.2017 to 28.04.2017	(1 w)
End semester examinations	29.04.2017 to 12.05.2017	(2 w)
Summer Vacation	13.05.2017 to 11.06.2017	(4w)
Commencement of class work for the next academic year 2016-17	**13.06.2017**	

* Dussehra holidays from 05.10.2016 to 12.10.2016 may change subject to the directions from the Government of Telangana

Yours faithfully

DIRECTOR

Copy to:

The Director of Evaluation
The Controller of Examinations.
P.A to VC, Rector and Registrar

KESHAV MEMORIAL INSTITUTE OF TECHNOLOGY

Narayanaguda, Hyderabad-29

DEPARTMENT OF INFORMATION TECHNOLOGY

B.Tech IV Year I Sem. IT Time Table(2016-17)

W.E.F : 13-06-2016

TIME/DAY	9.30-10.20 (1)	10.20-11.10 (2)	11.10-11.25	11.25-12.15 (3)	12.15-1.05 (4)	1.05-1.45	1.45-2.35 (5)	2.35-3.25 (6)	3.25-4.15 (7)
MON	MAD	HCI	SHORT BREAK	IS	IRS	LUNCH	CT & ST LAB		
TUE	IS	HCI		IRS	DP/ IS*		MAD LAB		
WED	DP	BDA /MAD*	SHORT BREAK	IS	IRS		MENTORING& COUNSELLING	IS /DP*	BDA
THU	MAD	BDA		IRS /HCI*	DP		MAD	BDA	HCI
FRI	BDA	MAD/BDA*	SHORT	DP	HCI		BDA	HCI/ IRS*	MAD
SAT	IRS	DP	BREAK	MAD	IS		IRS	IS	HCI

*-

TUTORIAL

SL.NO	SUBJECT	FACULTY
1	INFORMATION SECURITY	MR.G.BALA KRISHNA(IT)
2	DESIGN PATTERNS	MR.NEIL GOGTE(IT)
3	MOBILE APPLICATION DEVELOPMENT	MR.SRINIVAS ADABALA(IT)
4	INFORMATION RETRIEVAL SYSTEM	DR. RAMAKANTA MOHANTHY(IT)
5	BIG DATA ANALYTICS	MRS. REKHA(IT)
6	HUMAN COMPUTER INTERACTION	B.MANASA(IT)
7	CTST LAB (FS-6)	MS.VIJETHA/MS.PRITI SHAH/MS.SHALMILI(IT)
8	MAD LAB (FS-6)	MRS.SRINIVAS ADABALA/MS. B.MANASA/MS.SUNITHA
9	MENTORING & COUNSELLING	MRS.SRINIVAS ADABALA/MS.PRITI SHAH/MS.SHALMILI(IT)

CLASS-INCHARGE **HOD** **PRINCIPAL**

KESHAV MEMORIAL INSTITUTE OF TECHNOLOGY

Narayanaguda, Hyderabad-29

DEPARTMENT OF INFORMATION TECHNOLOGY

B.Tech IV Year I Sem. IT Time Table(2016-17)

INDIVIDUAL TIME TABLE

TIME/DAY	9.30-10.20 (1)	10.20-11.10 (2)	11.10-11.25	11.25-12.15 (3)	12.15-1.05 (4)	1.05-1.45	1.45-2.35 (5)	2.35-3.25 (6)	3.25-4.15 (7)
MON			SHORT BREAK	IS		LUNCH			
TUE	IS				DP/ IS*				
WED			SHORT BREAK	IS			MENTORING & COUNSELLING	IS /DP*	
THU									
FRI			SHORT						
SAT			BREAK		IS			IS	

KESHAV MEMORIAL INSTITUTE OF TECHNOLOGY

DEPARTMENT OF INFORMATION TECHNOLOGY

COURSE – PLAN – One copy to be submitted to the HOD one week before commencement of the semester

Subject Code	Name of the Subject	Class/Sem	Name of the Faculty / Designation	Number of Students	Total Proposed Periods per semester/year	
(A70522)	IS	B.TECH/IT/ IVYR/ I SEM	G BALAKRISHNA Asst. prof.	56	Lectures 56	
Week Number	**Lecture Number**	**Topic**	**Date of Completion**	**Ref.**	**Teaching Methods**	**No.of Classes**
1	1	**UNIT-I** : Introduction to Information Security	15/6/2016	T1,T2	B.B	1
	2	The need for security, Security Approaches	15/6/2016	T2	B.B	1
	3	Principles of Security, Types of Security Attacks, Security Services	16/6/2016	T1,T2	B.B	1
	4	TUTORIAL – 1	16/6/2016		PPT	1
2	5	Security Mechanism, A model for Network Security	22/6/2016	T1	B.B	1
	6	Introduction to Cryptography: Plaint text and Cipher text	22/6/2016	T1,T2	B.B	1
	7	Substitution Techniques, Transposition Techniques	23/6/2016	T1	B.B	1
	8	TUTORIAL – 2	23/6/2016		PPT	1
3	9	Encryption ,decryption, Symmetric and Asymmetric key cryptography	29/6/2016	T1,T2	B.B	1
	10	Steganography, key range and key size, Possible types of attacks	29/6/2016	T1	B.B	1
	11	**UNIT-II:** Block Cipher principles & Algorithms-DES,AES	30/6/2016	T1	B.B	1
	12	TUTORIAL – 3	30/6/2016		PPT	1

Signature of the Coordinator
Date

Signature of the Faculty
Date

*This column has to be filled-up after completion of the lecture/tutorial/practical in the copy kept with the faculty members.

KESHAV MEMORIAL INSTITUTE OF TECHNOLOGY

DEPARTMENT OF INFORMATION TECHNOLOGY

COURSE – PLAN – One copy to be submitted to the HOD one week before commencement of the semester

	13	Blowfish, Differential and Linear Cryptanalysis, Block cipher modes	13/7/2016	T1	B.B	1
4	14	Stream ciphers - RC4	13/7/2016	T1	PPT	1
	15	Location and placement of encryption function	13/7/2016	T1	B.B	1
	16	TUTORIAL – 4	14/7/2016		PPT	1
5	17	Key Distribution, Principles of public key cryptosystems,	14/7/2016	T1	B.B	1
	18	Algorithms-RSA	20/7/2016	T1,T2	B.B	1
	19	Diffie-Hellman Key Exchange	21/7/2016	T1	B.B	1
	20	TUTORIAL – 5	27/7/2016			1
6	21	ECC	28/7/2016	T1	B.B	1
	22	Key Distribution	3/8/2016	T1	B.B	1
	23	**UNIT-III:** Authentication Requirements	4/8/2016	T1	B.B	1
	24	Authentication Functions	4/8/2016	T1	B.B	1
	25	TUTORIAL – 6	17/8/2016		B.B	1
7	26	Message authentication codes	17/8/2016	T1	B.B	1
	27	Hash Functions	18/8/2016	T1	B.B	1
	28	TUTORIAL – 7	18/8/2016		B.B	1
	29	Secure hash algorithm (SHA)	24/8/2016	T1	B.B	1

Signature of the Coordinator
Date

Signature of the Faculty
Date

*This column has to be filled-up after completion of the lecture/tutorial/practical in the copy kept with the faculty members.

I

KESHAV MEMORIAL INSTITUTE OF TECHNOLOGY

DEPARTMENT OF INFORMATION TECHNOLOGY

COURSE – PLAN – One copy to be submitted to the HOD one week before commencement of the semester

8	30	Whirlpool	24/8/2016	T1	B.B	1
	31	HMAC	31/8/2016	T1	B.B	1
	32	CMAC	31/8/2016	T1	B.B	1
	33	Digital signatures, knapsack algorithm	1/9/2016	T1	B.B	1
	34	TUTORIAL – 8	1/9/2016		B.B	1
9	35	Kerberos, X.509,authentication service	7/9/2016	T1	B.B	1
	36	Public key Infrastructure	7/9/2016	T1	B.B	1
	37	Biometic authentication	8/9/2016	T1	B.B	1
10	38	**UNIT-IV:** Pretty Good Privacy	8/9/2016	T1,T2	B.B	1
	39	S/MIME	14/9/2016	T1	B.B	1
	40	TUTORIAL – 9	14/9/2016			1
11	41	IP security architecture	15/9/2016	T1,T2	B.B	1
	42	Authentication Header	15/9/2016	T1	B.B	1
	43	Encapsulating security payload	21/9/2016	T1	B.B	1
	44	Combining security associations	22/9/2016	T1	B.B	1
12	45	Key management	22/9/2016	T1	B.B	1
	46	**UNIT-V:** Web security consideration	28/9/2016	T1,T2	B.B	1

Signature of the Coordinator
Date

Signature of the Faculty
Date

*This column has to be filled-up after completion of the lecture/tutorial/practical in the copy kept with the faculty members.

	47	Secure socket layer and Transport layer security	28/9/2016	T1	B.B	1
13	48	Secure electronic transaction, Intruders	29/9/2016	T1	PPT	1
	49	Intrusion detection, Password management	27/10/2016	T1,T2	PPT	1
	50	Virus and related threats	27/10/2016	T1,T2	PPT	1
	51	Counter measures, Firewall design principles, Types of firewalls	2/11/2016	T1,T2	PPT	1
	52	Case studies: Secure Inter-branch payment transactions	2/11/2016	T1,T2	B.B	1
14	53	Cross site scripting vulnerability	3/11/2016	T1,T2	B.B	1
	54	Virtual Elections	3/11/2016	T1,T2	B.B	1
	55	Revision	4/11/2016		B.B	1
	56	Revision	5/11/2016		B.B	1

Text Books : -

1. Cryptography and Network Security : William Stallings, Pearson Education, 4th Edition
2. Cryptography and Network Security : Atul Kahate, Mc Graw Hill, 2nd Edition

Signature of the Coordinator
Date

Signature of the Faculty
Date

*This column has to be filled-up after completion of the lecture/tutorial/practical in the copy kept with the faculty members.

INFORMATION SECURITY

TOPICS BEYOND SYLLABUS (TBS)

S.NO	TOPIC
1	Information Security in Today's World PO1, PO12
2	Current Trends in Data Security PO12, PSO2
3	Triple DES PO1, PO2
4	RC5 PO1, PO2
5	VPN Security PO2

Information Security in Today's World

Protecting Your PC, Privacy and Self

"The minute you dial in to your Internet service provider or connect to a DSL or cable modem, you are casting your computer adrift in a sea of millions of other computers – all of which are sharing the world's largest computer network, the Internet. Most of those computers are cooperative and well behaved, but some are downright nasty. ***Only you can make sure your computer is ready for the experience.***"

Daniel Appleman, *Always Use Protection, A Teen's Guide to Safe Computing*, (2004 – Apress)

Purpose of This Discussion

- Provide an overview of:
  - What information security is
  - The challenges to InfoSec
  - The latest trends
  - Best practices to help protect your digital assets
  - The need for Information Security professionals
  - CyberWATCH

What Is Information Security?

- Process by which digital information assets are protected
- Topic areas: Policies and procedures, authentication, attacks, remote access, E-mail, Web, wireless, devices, media/medium, secure architectures, IDSes/IPSes, operating systems, secure code, Cryptography, physical security, digital media analysis…

Understanding the Importance of Information Security

- Prevents data theft
- Avoids legal consequences of not securing information
- Maintains productivity
- Foils cyberterrorism
- Thwarts identity theft

Challenges

- A number of trends illustrate why security is becoming increasingly difficult:
  - Speed of attacks
  - Sophistication of attacks
  - Faster detection of weaknesses
  - Distributed attacks
  - Difficulties of patching

Latest Trends

- Identity theft
- Malware
- Patch Management failures
- Distributed Denial of Service

Latest Trends - Identity Theft

- Crime of the 21st century
- Involves using someone's personal information, such as social security numbers, to establish bank or credit card accounts that are then left unpaid, leaving the victim with the debts and ruining their credit rating
- National, state, and local legislation continues to be enacted to deal with this growing problem:
  - **The Fair and Accurate Credit Transactions Act of 2003** is a federal law that addresses identity theft

Latest Trends - Identity Theft - continued

- **Phishing** is a method used by identity thieves to obtain financial information from a computer user
- The word "phishing" was made up by hackers as a cute word to use for the concept of *fishing for information*
- One of the most lucrative forms of spamming
- Often used in conjunction with spoofed Web sites

Latest Trends - Identity Theft - continued

- According to the Identity Theft Resource Center, a victim of identity theft spends an average of more than 600 hours and $1,400 of out-of-pocket expenses restoring their credit by contacting credit bureaus, canceling credit cards, and negotiating with creditors

Latest Trends - Malicious Software (Malware)

- Designed to operate without the computer user's permission
- May change or destroy data
- May operate hardware without authorization
- Can hijack your Web browser
- Might steal information or otherwise aggravate a computer user or organization

Malware: 2006 at a Glance

- 1 in 91 E-mails is viral (2006); down from 1 in 44 (2005)
- New Trojans outweigh Windows viruses & worms 4:1

Top 10 Malware Threats in 2006 – January-June

1. *W32/Sober-Z: 22.4% (at its peak accounted for 1 in every 13 emails)
2. W32/Netsky-P: 12.2% (hardest hitting virus in 2004)
3. W32/Zafi-B: 8.9%
4. *W32/Nyxem-D: 5.9%
5. W32/Mytob-FO: 3.3%
6. W32/Netsky-D: 2.4%
7. W32/Mytob-BE: 2.3%
8. W32/Mytob-EX: 2.2%
9. W32/Mytob-AS: 2.2%
10. W32/Bagle-Zip: 1.9%
11. Others: 36.3%

*Worms

Malware Trends

- Spyware
- Keyloggers
- Rootkits
- Mobile malware
- Combined attack mechanisms

Malware Trends - Spyware

- Advertisement-focused applications that, much like computer worms, install themselves on systems with little or no user interaction
- While such an application may be legal, it is usually installed without the user's knowledge or informed consent
- A user in an organization could download and install a useful (often "free") application from the Internet and in doing so, unwittingly install a spyware component

Malware Trends – Spyware - continued

- Apart from privacy concerns, the greatest issue presented by spyware is its use of your computer's resources and bandwidth
- This translates into lost work as you wait for your computer to finish a task, lost time as you slowly browse the Internet, and can even necessitate a call for service by a technician
- The time and money lost while eradicating spyware often exceeds all other forms of malware and spam combined

Malware Trends - Keyloggers

- Used to capture user's keystrokes:
  - AKA **Keystoke Logging**
- Hardware and software-based
- Useful purposes:
  - Help determine sources of errors on system
  - Measure employee productivity on certain clerical tasks

Malware Trends - Rootkits

- Is a set of software tools intended to conceal running processes, files or system data, thereby helping an intruder to maintain access to a system while avoiding detection
- Often modify parts of the operating system or install themselves as drivers or kernel modules
- Are known to exist for a variety of operating systems
- Are difficult to detect

Malware Trends - Mobile Malware

- Increase in the number of mobile phone viruses being written
- Insignificant compared to the much larger number of viruses being written which target Windows desktop computers

Malware Trends - Combined Attack Mechanisms

- Speed at which malware can spread combined w/a lethal payload
- SPAM with spoofed Web sites
- Trojans installing bot software
- Trojans installing backdoors

Latest Trends - Patch Management Failures

- Shift towards patching versus testing
- In the next few years, it is estimated that 90% of cyber attacks will continue to exploit known security flaws for which a fix is available or a preventive measure known

Latest Trends - Patch Management Failures - continued

- Why? Doesn't scale well and isn't cost-effective:
  - A survey by the Yankee Group found that the average annual cost of patching ranges from $189-$254 per patch for each computer
  - The cost is primarily a result of lost productivity while the patch is applied and for technician installation costs. Patching costs in large organizations can exceed $50 million per year

Latest Trends - SPAM

- January 24, 2004 - Bill Gates predicted that spam would be "a thing of the past" within two years – the threat remains alive
- No end in sight:
  - According to Ferris Research, by 2007, the percentage of spam E-mails will increase to 70% of the total E-mail messages sent

Latest Trends - Vulnerability Exploitation

- Operating system attacks still in vogue:
  - Vista
  - Mac OS X
- Increase in attacks taking advantage of security holes in other products:
  - Desktop tools
  - Alternative Web browsers
  - Media applications
  - Microsoft Office applications

Latest Trends - Ransomware

- Type of malware that encrypts the victim's data, demanding ransom for its restoration
- **Cryptovirology** predates ransomware

Latest Trends - Distributed Denial of Service (DDoS)

- Use hundreds of infected hosts on the Internet to attack the victim by flooding its link to the Internet or depriving it of resources
- A PC becomes a zombie when a **bot**, or automated program, is installed on it, giving the attacker access and control and making the PC part of a zombie network, or **botnet**

Latest Trends - DDoS - continued

- One of the most high profile botnets of 2005 was created by the Zotob worm which achieved worldwide notoriety in August when leading media organizations including ABC, The Financial Times, and The New York Times fell prey to it

Best Practices to Help Protect Your Digital Assets

- Anti-virus software
- Anti-spyware software
- Windows and applications updates
- Security bundles
- Personal firewalls
- Wireless
- Other best practices

Anti-Virus Software

- Install and maintain anti-virus software. Use the software regularly
- Microsoft claims that fewer than 30% of all users have up-to-date anti-virus software installed
- Most AV manufacturers have information and alert pages where you can find "primers" on malware, as well as alerts to the most current threats

Anti-Virus Software Vendors

- McAfee: Virus Scan
- Symantec: Norton Anti-Virus
- Computer Associates: eTrust EZ AntiVirus
- Trend Micro: PC-cillian
- Grisoft: AVG Anti-Virus (**freeware**)
- Alwil Software: Avast! AntiVirus (**freeware**)
- eset: NOD32 (**freeware**)

Anti-Spyware Software

- Install and maintain anti-spyware software
- Use the software regularly
- Sunbelt Software: CounterSpy
- Webroot Software: Spy Sweeper
- Trend Micro: Anti-Spyware
- HijackThis (**freeware**)
- Lavasoft: Ad-Aware SE Personal (**freeware**)
- Spybot: Search & Destroy (**freeware**)
- Microsoft: Windows Defender (**freeware**)

Updating Windows and Other Applications

- Microsoft Update: Web site where users can download updates for various Windows-related products
- For the most part, it's automated
- Check to see it's working properly
- Install vendor-specific patches for applications (e.g., iTunes, Google Desktop)

Security Bundles

- Can include: Anti-virus software, personal firewall software, anti-spyware software, content filtering/parental control, pop-up blockers, anti-spam capabilities
- Can be difficult for the average user to setup:
  - Leads to incorrect configurations providing a false sense of security

Security Bundles - continued

- McAfee: Internet Security Suite
- Symantec: Norton Internet Security
- Computer Associates: eTrust EZ Armor
- Trend Micro: PC-cillian Internet Security
- ZoneAlarm: Internet Security Suite
- F-Secure: Internet Security
- MicroWorld: eScan Internet Security Suite
- Panda Software: Panda Internet Security
- Softwin BitDefender Professional Edition
- eXtendia Security Suite

Personal Firewalls

- Software installed on an end-user's PC which controls communications to and from the user's PC
- Permits or denies communications based on a security policy the user sets
- Use for handheld devices as well (Airscanner, Bluefire)

Personal Firewall Programs

- Zone Labs
- Symantec's Norton Personal Firewall
- Sunbelt's Kerio Personal Firewall
- Tiny Software's Tiny Personal Firewall
- Mac OS X
- Windows XP (with Service Pack 2)

Living in a Wireless World

- By 2007, >98% of all notebooks will be wireless-enabled
- Serious security vulnerabilities have been created by wireless data technology:
  - Unauthorized users can access the wireless signal from outside a building and connect to the network
  - Attackers can capture and view transmitted data (including encrypted data)
  - Employees in the office can install personal wireless equipment and defeat perimeter security measures

Wireless Security Best Practices

- Implement MAC-address filtering
- Turn off unnecessary services (telnet, HTTP)
- Change default SSID/Disable SSID broadcasts
- Change default channel
- Disable DHCP on access point
- Use encryption (usually not enabled by default on most access points
- Change default admin username and password
- Specify the number of clients that can connect to the access point

Other Best Practices

- When not using your PC, turn it off
- View your E-mail as text only; disable the function that automatically views E-mail as HTML
- Do not automatically open attachments
- Do not run software programs of unknown origin
- Delete chain E-mails and junk mail. Do not forward or reply to any of them

Other Best Practices - continued

- Never reply back to an E-mail to "unsubscribe" or to remove yourself from an unknown list. This lets the spammers know that they have reached a live E-mail address and your spam mail will increase
- Back up your critical data and documents regularly – thumb drives and CDs are cheap

The Need for Information Security Professionals

- No matter how hard we try to do the aforementioned, there will still be the need for information security professionals
- Information security personnel are in short supply; those in the field are being rewarded well

The Need for Information Security Professionals – continued

- Security budgets have been spared the drastic cost-cutting that has plagued IT since 2001
- Companies recognize the high costs associated with weak security and have decided that prevention outweighs cleanup
- Regulatory compliance is also driving the need for more qualified professionals

CyberWATCH

- Cybersecurity: Washington Area Technician and Consortium Headquarters
- NSF ATE-funded 4 year project that includes community colleges, four-year schools, high schools, local, state, and federal government agencies, and businesses in the Baltimore, Washington D.C., and Northern Virginia regions

CyberWATCH - continued

- Addressing the challenges and concerns in education and the business industry:
  - The shortage of security professionals
  - A perceived lack of business and team-work skills among IT professionals
  - The lack of a cybersecurity curriculum at many higher education institutions

CyberWATCH - continued

- Professional development for faculty, high school teachers, students, and staff will benefit populations that are traditionally least likely to major in fields requiring a cybersecurity/information security component

CyberWATCH – Getting Involved

- Contact Casey O'Brien at (410) 780-6139

Current Trends in Data Security

1

Data Security

Dorothy Denning, 1982:

- Data Security is the science and study of methods of protecting data (...) from unauthorized disclosure and modification
- Data Security = Confidentiality + Integrity

2

Data Security

- Distinct from *systems* and *network* security
  - Assumes these are already secure
- Tools:
  - Cryptography, information theory, statistics, ...
- Applications:
  - An *enabling* technology

3

Outline

- Traditional data security
- Two attacks
- Data security research today
- Conclusions

4

Traditional Data Security

- Security in SQL = Access control + Views
- Security in statistical databases = Theory

5

[Griffith&Wade'76, Fagin'78]

Access Control in SQL

GRANT privileges ON object TO users
[WITH GRANT OPTION$]

privileges = SELECT | INSERT | DELETE | ...

object = table | attribute

REVOKE privileges ON object FROM users
[CASCADE]

6

Views in SQL

A SQL View = (almost) any SQL query

- Typically used as:

```
CREATE VIEW pmpStudents AS
   SELECT * FROM Students WHERE ...
```

```
GRANT SELECT ON pmpStudents TO DavidRispoli
```

7

Summary of SQL Security

Limitations:

- No row level access control
- Table creator owns the data: that's unfair !

Access control = great success story of the DB commun

... or spectacular failure:

- Only 30% assign privileges to users/roles
  - And then to protect entire tables, not columns

8

Summary (cont)

- Most policies in middleware: slow, error prone:
  - SAP has 10**4 tables
  - GTE over 10**5 attributes
  - A brokerage house has 80,000 applications
  - A US government entity thinks that it has 350K
- Today the database is not at the center of the policy administration universe

[Rosenthal&Winslett'2004]

9

[Adam&Wortmann'89]

Security in Statistical DBs

Goal:

- Allow arbitrary *aggregate* SQL queries
- Hide confidential data

```
SELECT count(*)
FROM   Patients
WHERE age=42
    and sex='M'
    and diagnostic='schizophrenia'
```

OK

```
SELECT name
FROM   Patient
WHERE age=42
    and sex='M'
    and diagnostic='schizophrenia'
```

NO

10

[Adam&Wortmann'89]

Security in Statistical DBs

What has been tried:

- Query restriction
  - Query-size control, query-set overlap control, query monitoring
  - None is practical
- Data perturbation
  - Most popular: **cell combination, cell suppression**
  - Other methods, for continuous attributes: may introduce bias
- Output perturbation
  - For continuous attributes only

11

Summary on Security in Statistical DB

- Original goal seems impossible to achieve
- Cell combination/suppression are popular, but do not allow arbitrary queries

12

Outline

- Traditional data security
- Two attacks
- Data security research today
- Conclusions

13

[Chris Anley, *Advanced SQL Injection In SQL*]

SQL Injection

Your health insurance company lets you see the claims online:

First login: User: fred Password: ********

Now search through the claims :

Search claims by: Dr. Lee

SELECT...FROM...WHERE doctor='Dr. Lee' and patientID='fred'

SQL Injection

Now try this:

Search claims by: Dr. Lee' OR patientID = 'suciu'; --

.....WHERE doctor='Dr. Lee' OR patientID='suciu'; --' and patientID='fred'

Better:

Search claims by: Dr. Lee' OR 1 = 1; --

15

SQL Injection

When you're done, do this:

Search claims by: Dr. Lee'; DROP TABLE Patients; --

16

SQL Injection

- The DBMS works perfectly. So why is SQL injection possible so often ?
- Quick answer:
  - Poor programming: use stored procedures !
- Deeper answer:
  - Move policy implementation from apps to DB

17

Latanya Sweeney's Finding

- In Massachusetts, the Group Insurance Commission (GIC) is responsible for purchasing health insurance for state employees
- GIC has to publish the data:

GIC(**zip, dob, sex**, diagnosis, procedure, ...)

18

Latanya Sweeney's Finding

- Sweeney paid $20 and bought the voter registration list for Cambridge Massachusetts:

GIC(**zip, dob, sex**, diagnosis, procedure, ...)
VOTER(name, party, ..., **zip, dob, sex**)

19

Latanya Sweeney's Finding

zip, dob, sex

- William Weld (former governor) lives in Cambridge, hence is in VOTER
- 6 people in VOTER share his **dob**
- only 3 of them were man (same **sex**)
- Weld was the only one in that **zip**
- Sweeney learned Weld's medical records !

20

Latanya Sweeney's Finding

- All systems worked as specified, yet an important data has leaked
- How do we protect against that ?

Some of today's research in data security address breaches that happen even if all systems work correctly

21

Summary on Attacks

SQL injection:

- A correctness problem:
  - Security policy implemented poorly in the application

Sweeney's finding:

- Beyond correctness:
  - Leakage occurred when all systems work as specified

22

Outline

- Traditional data security
- Two attacks
- Data security research today
- Conclusions

23

Research Topics in Data Security

Rest of the talk:

- Information Leakage
- Privacy
- Fine-grained access control
- Data encryption
- Secure shared computation

24

[Samarati&Sweeney'98, Meyerson&Williams'04]

Information Leakage: k-Anonymity

Definition: each tuple is equal to at least k-1 others

Anonymizing: through suppression and generalization

First	Last	Age	Race
*	Stone	30-50	Afr-Am
John	R*	20-40	*
*	Stone	30-50	Afr-am
John	R*	20-40	*

Hard: NP-complete for supression only
Approximations exists

25

[Miklau&S'04, Miklau&Dalvi&S'05,Yang&Li'04]

Information Leakage: Query-view Security

Have data: TABLE Employee(name, dept, phone)

Secret Query	View(s)	Disclosure? total
S(name)	V(name,phone)	
S(name,phone)	V1(name,dept) V2(dept,phone)	big
S(name)	V(dept)	tiny
S(name) where dept='HR'	V(name) where dept='RD'	none

26

Summary on Information Disclosure

- The theoretical research:
  - Exciting new connections between databases and information theory, probability theory, cryptography [Abadi&Warinschi'05]
- The applications:
  - many years away

27

Privacy

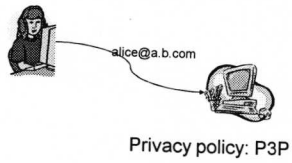
- "Is the right of individuals to determine for themselves when, how and to what extent information about them is communicated to others" [Agrawal'03]
- More complex than confidentiality

28

Privacy

Involves:
- Data
- Owner
- Requester
- Purpose
- Consent

Example: Alice gives her email to a web service

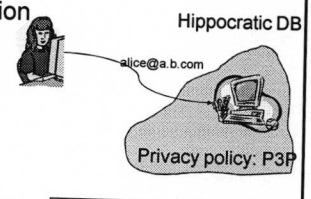


Privacy policy: P3P

29

Hippocratic Databases

DB support for implementing privacy policies.
- Purpose specification
- Consent
- Limited use
- Limited retention
- Protection against:
  - ✓ Sloppy organizations
  - × Malicious organizations

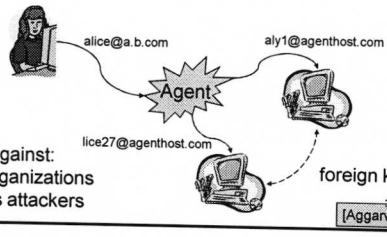


[Agrawal'03, LeFevrey'04]

30

Privacy for Paranoids

- Idea: rely on trusted agents



Protection against:
✓ Sloppy organizations
✓ Malicious attackers

foreign keys ?

[Aggarwal'04]

31

Summary on Privacy

- Major concern in industry
  - Legislation
  - Consumer demand
- Challenge:
  - How to enforce an organization's stated policies

32

Fine-grained Access Control

Control access at the tuple level.

- Policy specification languages
- Implementation

33

Policy Specification Language

No standard, but usually based on parameterized views.

```
CREATE AUTHORIZATION VIEW PatientsForDoctors AS
  SELECT Patient.*
  FROM Patient, Doctor
  WHERE Patient.doctorID = Doctor.ID
     and   Doctor.login = %currentUser
```

Context parameters

34

Implementation

```
SELECT Patient.name, Patient.age
FROM Patient
WHERE Patient.disease = 'flu'
```

⇩

```
SELECT Patient.name, Patient.age
FROM Patient, Doctor
WHERE Patient.disease = 'flu'
      and Patient.doctorID = Doctor.ID
      and Patient.login = %currentUser
```

e.g. Oracle

35

Two Semantics

- The Truman Model = filter semantics
  - transform reality
  - ACCEPT all queries
  - REWRITE queries
  - Sometimes misleading results

```
SELECT count(*)
FROM Patients
WHERE disease='flu'
```

- The non-Truman model = deny semantics
  - reject queries
  - ACCEPT or REJECT queries
  - Execute query UNCHANGED
  - May define multiple security views for a user

[Rizvi'04]

36

Summary of Fine Grained Access Control

- Trend in industry: label-based security
- Killer app: application hosting
  - Independent franchises share a single table at headquarters (e.g., Holiday Inn)
  - Application runs under requester's label, cannot see other labels
  - Headquarters runs Read queries over them
- Oracle's Virtual Private Database

[Rosenthal&Winslett'2004]

37

Data Encryption for Publishing

Scientist wants to publish medical research data on the Web

- Users and their keys:

All authorized users:	K_{user}
Patient:	K_{pat}
Doctor:	K_{dr}
Nurse:	K_{nu}
Administrator :	K_{admin}

- Complex Policies:

Doctor researchers may access trials
Nurses may access diagnostic
Etc...

What is the encryption granularity ?

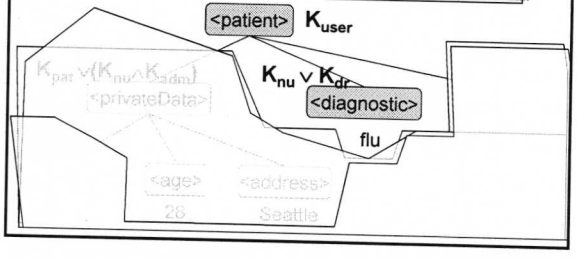
38

[Miklau&S.'03]

Data Encryption for Publishing

An XML tree protection:

Doctor: K_{user}, K_{dr}
Nurse: K_{user}, K_{nu}
Nurse+admin: $K_{user}, K_{nu}, K_{adm}$



Summary on Data Encryption

- Industry:
  - Supported by all vendors: Oracle, DB2, SQL-Server
  - Efficiency issues still largely unresolved
- Research:
  - Hard theoretical security analysis [Abadi&Warinschi'05]

40

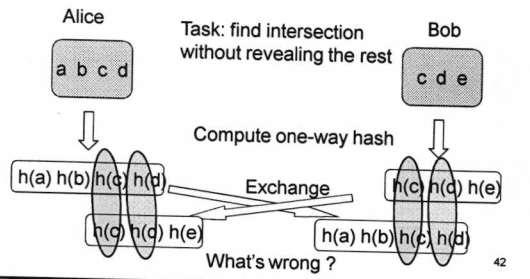
Secure Shared Processing

- Alice has a database DB_A
- Bob has a database DB_B
- How can they compute $Q(DB_A, DB_B)$, without revealing their data ?
- Long history in cryptography
- Some database queries are easier than general case

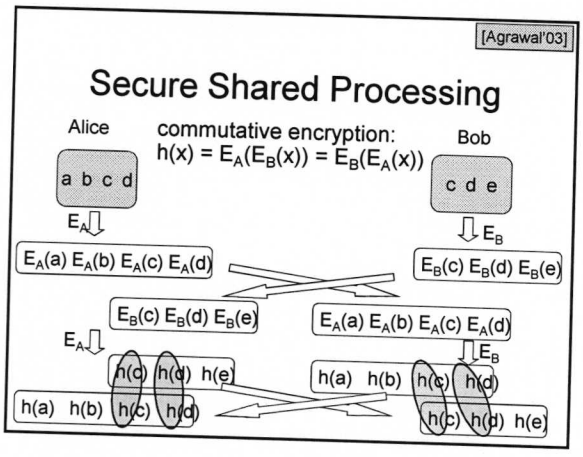
41

[Agrawal'03]

Secure Shared Processing



42



Summary on Secure Shared Processing

- Secure intersection, joins, data mining
- But are there other examples ?

44

Triple DES

- clear a replacement for DES was needed
  - theoretical attacks that can break it
  - demonstrated exhaustive key search attacks
- AES is a new cipher alternative
- prior to this alternative was to use multiple encryption with DES implementations
- Triple-DES is the chosen form

Why Triple-DES?

- why not Double-DES?
  - NOT same as some other single-DES use, but have
- meet-in-the-middle attack
  - works whenever use a cipher twice
  - since $X = E_{K1}[P] = D_{K2}[C]$
  - attack by encrypting P with all keys and store
  - then decrypt C with keys and match X value
  - can show takes $O(2^{56})$ steps

Triple-DES with Two-Keys

- hence must use 3 encryptions
  - would seem to need 3 distinct keys
- but can use 2 keys with E-D-E sequence
  - $C = E_{K1}[D_{K2}[E_{K1}[P]]]$
  - nb encrypt & decrypt equivalent in security
  - if $K1=K2$ then can work with single DES
- standardized in ANSI X9.17 & ISO8732
- no current known practical attacks

Triple-DES with Three-Keys

- although are no practical attacks on two-key Triple-DES have some indications
- can use Triple-DES with Three-Keys to avoid even these
  - $C = E_{K3}[D_{K2}[E_{K1}[P]]]$
- has been adopted by some Internet applications, eg PGP, S/MIME

RC5

- a proprietary cipher owned by RSADSI
- designed by Ronald Rivest (of RSA fame)
- used in various RSADSI products
- can vary key size / data size / no rounds
- very clean and simple design
- easy implementation on various CPUs
- yet still regarded as secure

RC5 Ciphers

- RC5 is a family of ciphers RC5-w/r/b
  - w = word size in bits (16/32/64) nb data=2w
  - r = number of rounds (0..255)
  - b = number of bytes in key (0..255)
- nominal version is RC5-32/12/16
  - ie 32-bit words so encrypts 64-bit data blocks
  - using 12 rounds
  - with 16 bytes (128-bit) secret key

RC5 Key Expansion

- RC5 uses 2r+2 subkey words (w-bits)
- subkeys are stored in array `S[i]`, i=0..t-1
- then the key schedule consists of
  - initializing S to a fixed pseudorandom value, based on constants e and phi
  - the byte key is copied (little-endian) into a c-word array L
  - a mixing operation then combines L and S to form the final S array

RC5 Encryption

- split input into two halves A & B

  $L_0 = A + S[0];$

  $R_0 = B + S[1];$

  for i = 1 to r do

  $L_i = ((L_{i-1} \text{ XOR } R_{i-1}) <<< R_{i-1}) + S[2 \times i];$

  $R_i = ((R_{i-1} \text{ XOR } L_i) <<< L_i) + S[2 \times i + 1];$
- each round is like 2 DES rounds
- note rotation is main source of non-linearity
- need reasonable number of rounds (eg 12-16)

RC5 Modes

- RFC2040 defines 4 modes used by RC5
  - RC5 Block Cipher, is ECB mode
  - RC5-CBC, is CBC mode
  - RC5-CBC-PAD, is CBC with padding by bytes with value being the number of padding bytes
  - RC5-CTS, a variant of CBC which is the same size as the original message, uses ciphertext stealing to keep size same as original

Block Cipher Characteristics

- features seen in modern block ciphers are:
  - variable key length / block size / no rounds
  - mixed operators, data/key dependent rotation
  - key dependent S-boxes
  - more complex key scheduling
  - operation of full data in each round
  - varying non-linear functions

Stream Ciphers

- process the message bit by bit (as a stream)
- typically have a (pseudo) random **stream key**
- combined (XOR) with plaintext bit by bit
- randomness of **stream key** completely destroys any statistically properties in the message
  - $C_i = M_i \text{ XOR StreamKey}_i$
- what could be simpler!!!!
- but must never reuse stream key
  - otherwise can remove effect and recover messages

Stream Cipher Properties

- some design considerations are:
  - long period with no repetitions
  - statistically random
  - depends on large enough key
  - large linear complexity
  - correlation immunity
  - confusion
  - diffusion
  - use of highly non-linear boolean functions

RC4

- a proprietary cipher owned by RSA DSI
- another Ron Rivest design, simple but effective
- variable key size, byte-oriented stream cipher
- widely used (web SSL/TLS, wireless WEP)
- key forms random permutation of all 8-bit values
- uses that permutation to scramble input info processed a byte at a time

RC4 Key Schedule

- starts with an array S of numbers: 0..255
- use key to well and truly shuffle
- S forms **internal state** of the cipher
- given a key k of length l bytes

```
for i = 0 to 255 do
   S[i] = i
j = 0
for i = 0 to 255 do
   j = (j + S[i] + k[i mod l]) (mod 256)
   swap (S[i], S[j])
```

RC4 Encryption

- encryption continues shuffling array values
- sum of shuffled pair selects "stream key" value
- tXOR with next byte of message to en/decrypt

```
i = j = 0
for each message byte M_i
   i = (i + 1) (mod 256)
   j = (j + S[i]) (mod 256)
   swap(S[i], S[j])
   t = (S[i] + S[j]) (mod 256)
   C_i = M_i XOR S[t]
```

RC4 Security

- claimed secure against known attacks
  - have some analyses, none practical
- result is very non-linear
- since RC4 is a stream cipher, must **never reuse a key**
- have a concern with WEP, but due to key handling rather than RC4 itself

IPSEC VPN

What is a VPN?

- VPN is a tunnel
  - data is encrypted and then encapsulated by a VPN gateway
- VPN protects
  - the data from being understood (confidentiality)
  - against spoofing the sender or the recipients' identity (authentication).
- VPN architectures
  - network to network
  - host to network
  - host to host

IPSec Operating Modes

- IPSec Transport Mode
  - Protects the payload only
  - No encapsulation
  - Original IP header preserved
- Host to host
  - Host must be aware of IPSec
  - Provides End-to-end protection
    - From host to host
    - Not just in transit

IPSec Operating Modes

- IPSec Tunnel Mode
  - Encrypts entire message (headers + payload)
  - The IPSec gateway encrypts and encapsulates
  - Adds new headers to send the encrypted packet to the end-point IPSec router
- Could be host-host, host-gateway or gateway-gateway
- Transparent to hosts
- Protects IP address/header

IPSEC Modes and Architectures

- Transport Mode
  - Host-Host
- Tunnel Mode
  - Gateway-Gateway
  - Host-Gateway
  - Host-Host

IPSEC SA – Security Association

- Like a connection
- Uniquely ID'ed by
  - Security Parameters Index (SPI)
    - Local Id number identifies SA
  - IP Destination Address
    - Note, one way
  - Security Protocol
    - AH or ESP

INFORMATION SECURITY

TOPIC WISE WEB LINKS

1	1	**UNIT-I** : Introduction to Information Security	**https://en.wikipedia.org/wiki/Information_security**
	2	The need for security, Security Approaches	http://www.industrial-ip.org/en/knowledge-center/solutions/security-and-compliance/a-layered-approach-to-network-security
	3	Principles of Security, Types of Security Attacks, Security Services	http://www.cs.cornell.edu/courses/cs5430/2015sp/notes/principles.php
	4	Security Mechanism, A model for Network Security	http://w3-o.cs.hm.edu/mediapool/soceanu/pmcio/Network_Security_Vulnerabilities_Threats_Attacks.pdf
	5	Introduction to Cryptography: Plaint text and Cipher text	https://www.cs.jhu.edu/~scheideler/courses/600.471_S05/lecture_6.pdf
	6	Substitution Techniques, Transposition Techniques	http://www.cse.wustl.edu/~jain/cse571-11/ftp/l_02cet.pdf
	7	Encryption ,decryption, Symmetric and Asymmetric key cryptography	https://en.wikipedia.org/wiki/Symmetric and Asymmetric key cryptography
	8	Steganography, key range and key size, Possible types of attacks	https://en.wikipedia.org/wiki/Steganography
2	9	**UNIT-II:** Block Cipher principles & Algorithms-DES,AES	**http://en.citizendium.org/wiki/Block_cipher**
	10	Blowfish, Differential and Linear Cryptanalysis	https://en.wikipedia.org/wiki/Blowfish_(cipher)
	11	Block cipher modes of operation, stream ciphers	https://en.wikipedia.org/wiki/Block_cipher_mode_of_operation
	12	RC4, Location and placement of encryption function	https://en.wikipedia.org/wiki/RC4
	13	Key Distribution, Principles of public key cryptosystems, Algorithms-RSA	https://simple.wikipedia.org/wiki/RSA_(algorithm)
	14	Diffie-Hellman, ECC, Key Distribution	https://en.wikipedia.org/wiki/Key_distribution
3	15	**UNIT-III:** Authentication Requirements, Functions, Message authentication codes	**https://en.wikipedia.org/wiki/Message_authentication_code**
	16	Hash Functions, Secure hash algorithm, Whirlpool, HMAC, CMAC	https://en.wikipedia.org/wiki/Hash-based_message_authentication_code
	17	Digital signatures, knapsack algorithm	https://en.wikipedia.org/wiki/Digital_signature
	18	Kerberos, X.509,authentication service, Public key Infrastructure, Biometic authentication	http://www.umich.edu/~x509/
	19	**UNIT-IV:** Pretty Good Privacy, S/MIME	**https://cryptography.org/getpgp.htm**
	20	IP security architecture, Authentication Header	https://docs.oracle.com/cd/E19683-

INFORMATION SECURITY

TOPIC WISE WEB LINKS

4			01/817-2694/ipsec-ov-1/index.html
	21	Encapsulating security payload	https://tools.ietf.org/html/rfc4303
	22	Combining security associations, key management	https://en.wikipedia.org/wiki/Internet_Security_Association_and_Key_Management_Protocol
5	23	**UNIT-V:** Web security consideration, secure socket layer and Transport layer security	**https://msdn.microsoft.com/en-us/library/windows/desktop/aa380516(v=vs.85).aspx**
	24	Secure electronic transaction, Intruders	http://www3.cs.stonybrook.edu/~liu/cse315/26.pdf
	25	Intrusion detection, Password management	http://www.cse.wustl.edu/~jain/cse571-14/ftp/l_22id.pdf
	26	Virus and related threats	http://www.sciencedirect.com/science/article/pii/0169755289900068
	27	Counter measures, Firewall design principles, Types of firewalls, Secure Inter-branch payment transactions	http://www.academia.edu/9545649/Page_621_1_Firewall_Design_Principles
	28	Cross site scripting vulnerability, Virtual Elections	https://www.netsparker.com/blog/web-security/cross-site-scripting-xss/

(1)

UNIT-I

IS(1) Cryptography & network security — William stallings Pearson education, 4th edition (1). or) Atul Kahal 2nd E

- Information security: before data processing equipment, the security of info is primarily physical (ex: lock & key systems)
- Computer security: with introduction of computers, need for automated-tools for protecting files & other info stored in computer this situation arises especially in time sharing systems
  - Here need is to access info over public telephone n/w, data n/w or internet
- network security: this type of security comes into picture for distributed systems, use of n/w's and communication facilities for carrying data between terminal & comp. or b/w comp & comp

Internet security: data processing equipment with a collection of interconnected n/w's is referred to as an internet

- this focuses on measures to determine, prevent, detect and correct security violations that involve the transmission of information.

Services, Mechanisms and Attacks

To access security needs of an organization effectively and choose correct security protocols and policies

- Security Manager to systematically identify requirements of security and characterizing the approaches to satisfy requirements the three aspects of info security are (ITU-T) international telecommunication union — telecommunication standardization sector (x.
  - Security Attack: action that compromises the security of info owned by an organization
  - Security Mechanism: designed to detect, prevent or recover from security attack
  - Security Service: enhances the security of the data processing &

Services : identification, authorization, signature, notarization, liability, receipts, Endorsement, validation, authenticity, ownership, Registration

Mechanisms: Cryptographic techniques

Attacks : As G.J. Simmons points it, to secure is about how to prevent attacks or detect attacks

Security services

Authentication : assurance given that communicating entity is the one that it claims

- Peer Entity authentication
  used in logical connection to provide confidence in the identity of entity
- Data-origin authentication
  In a connection-less transfer provides assurance that source of received data is as claimed
- Access control

– Prevention of unauthorized use of a resource
ability to limit & control the access to host systems and applications via communication links

– To achieve this each entity should be identified or authenticated so access rights are given.

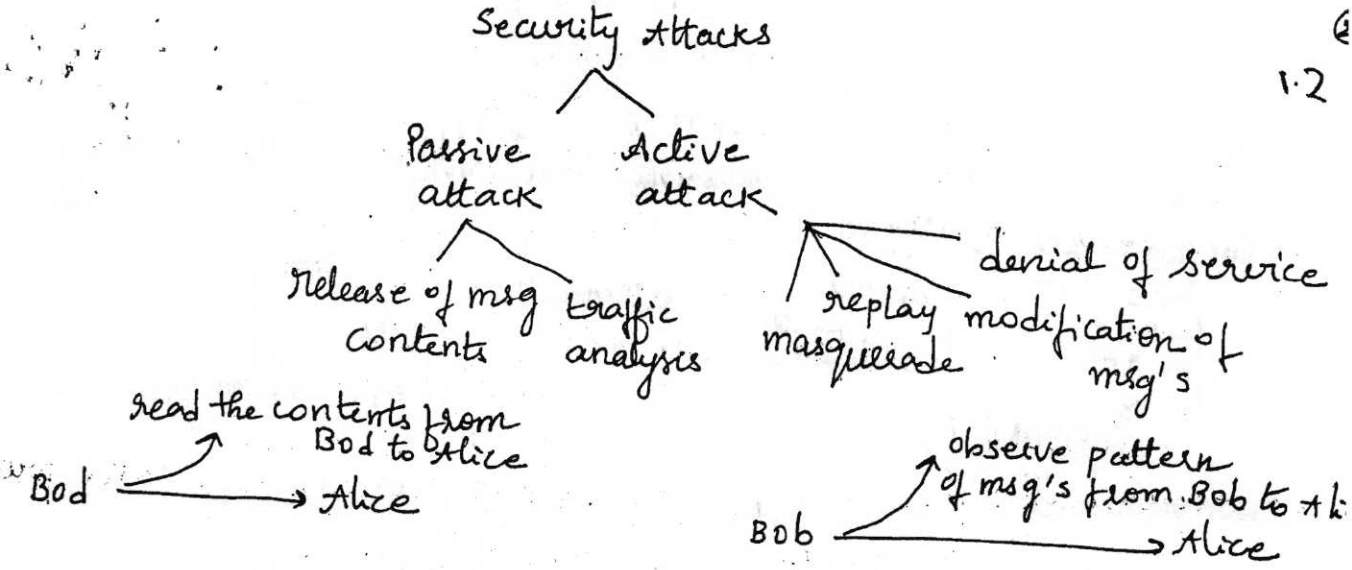
Data confidentiality : protection of data from unauthorized user.

Data integrity : Assurance that data received are exactly as sent by an authorized entity.

The above two apply to a stream of messages, single or selected fields within a message

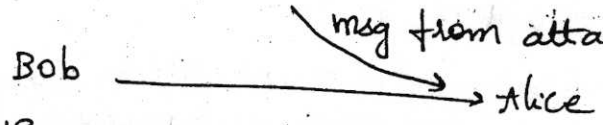
Connection, Connectionless, selective-field and traffic flow.

Nonrepudation : provides protection against denial by one of the entities involved in a communication of having participated in all.



Active

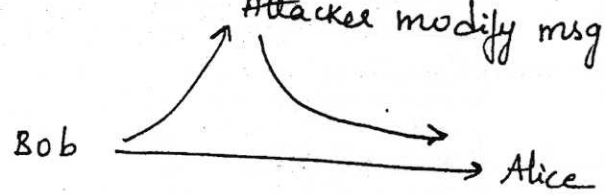
Masquerde : takes place when one entity pretends to be a different entity (spoofing)



Replay : involves the passive capture of a data unit & its subsequent retransmission to produce an unauthorized effect

~~email spoofing~~

Modification of msg's : some portion of legitimate msg is altered or that msg's are delayed or recorded to produce an unauthorized effect

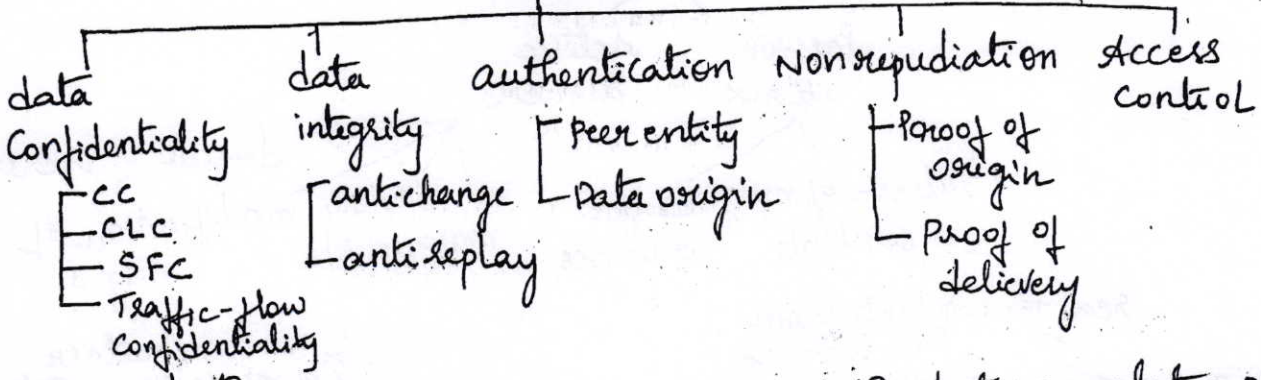


Denial of service : prevents normal use of communication facilites entity may supress all msg's directed to paticular destination

- Another form is distruption of an entire n/w neither by disabling the n/w or by overloading it with msg's as to degrade performance

- Passive are difficult to detect . measure or avoid.

Security services

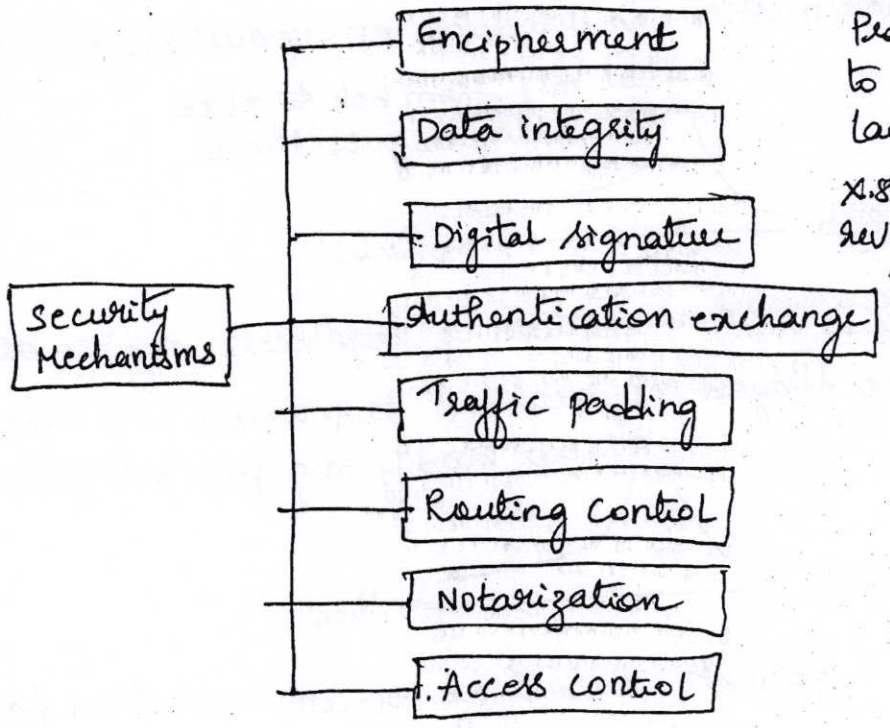


Non repudiation : proof of origin receiver of the data can later prove the identity of the sender if denied

Proof of delievery : the sender of data can later prove that data were delivered to the intended recipient.

(Prevents either sender or receiver from denying a transmission msg.

Security Mechanisms



mechanisms on specific Protocol & not specific to any paticular protocol layer

x.800 distinguished b/w reversible encipherment mechanisms and irreversible-encipherment mechanisms

1. encipherment : hidding or covering data can provide confidentiality
   2 techniques cryptography & stegnography.
2. data integrity : appends a checkvalue to data for specific process

- Compares the newly checkvalue with the one received, if two checkvalues are same then data integrity has preserved. 13

Digital signature : means by which the sender can electronically sign the data and the receiver can electronically verify the signatu sender uses a process that involves showing that she own's a private key related to the public key that she has announced public receiver uses sender's public key to prove that msg is indeed signed by the sender who claims to have sent the message.

Authentication exchange : Two entities exchange some messages to prove their identity to each other.
She knowns a secret that only she is supposed to known.

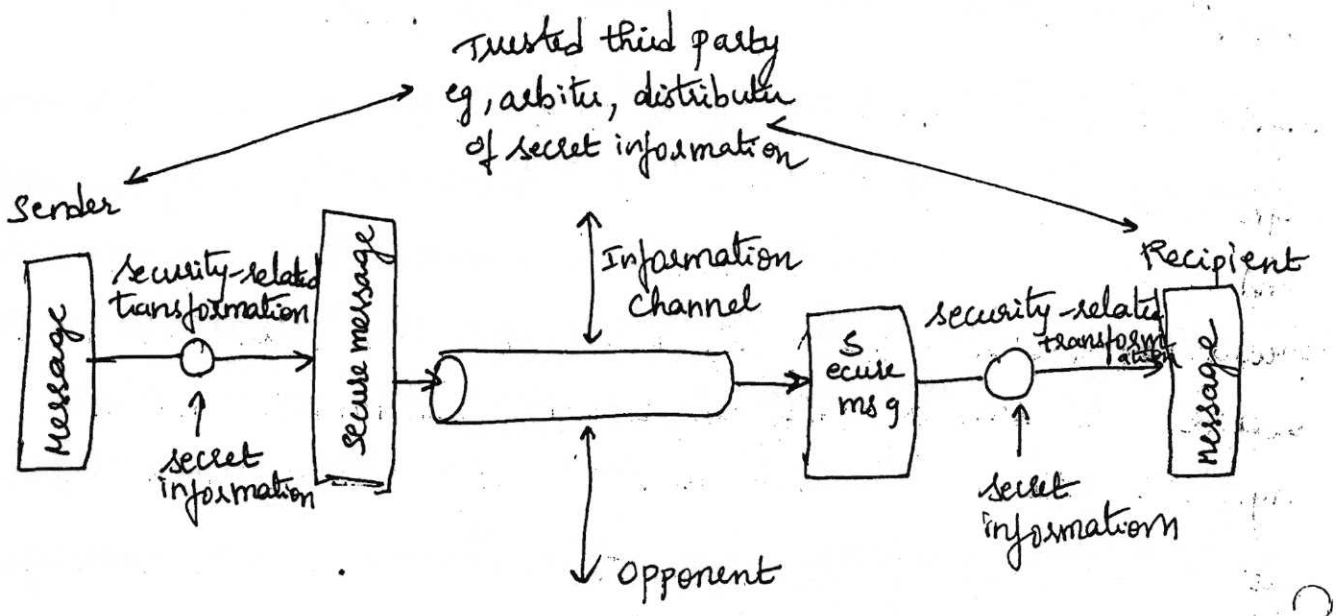
Traffic padding : insert some bogus data into data traffic

Routing control : selecting and continously changing different available routes b/w the sender and receiver

Notarization : selecting third party turstworty to control the communication - between two entities

Access control : uses methods to prove that a user has access right to the data or resources owned by a system.

Model of n/w security



- A message is to be transferred from one party to another across some sort of internet (called principals)
- Two parties who are the principals must cooperate for exchange
- A logical information channel is established by defining a route through the internet from src to destination and by use of protocols
- Security come into play when it is neccessary to protect the info transmission from an opponent who may present a threat to confidentiality, authenticity & so on.

Techniques to provide security have two components

- security related transformation on info to be sent
- some secret information shared by the two principals and it hoped unknown to the opponent

- Trusted third party may be needed to achieve secure transmission

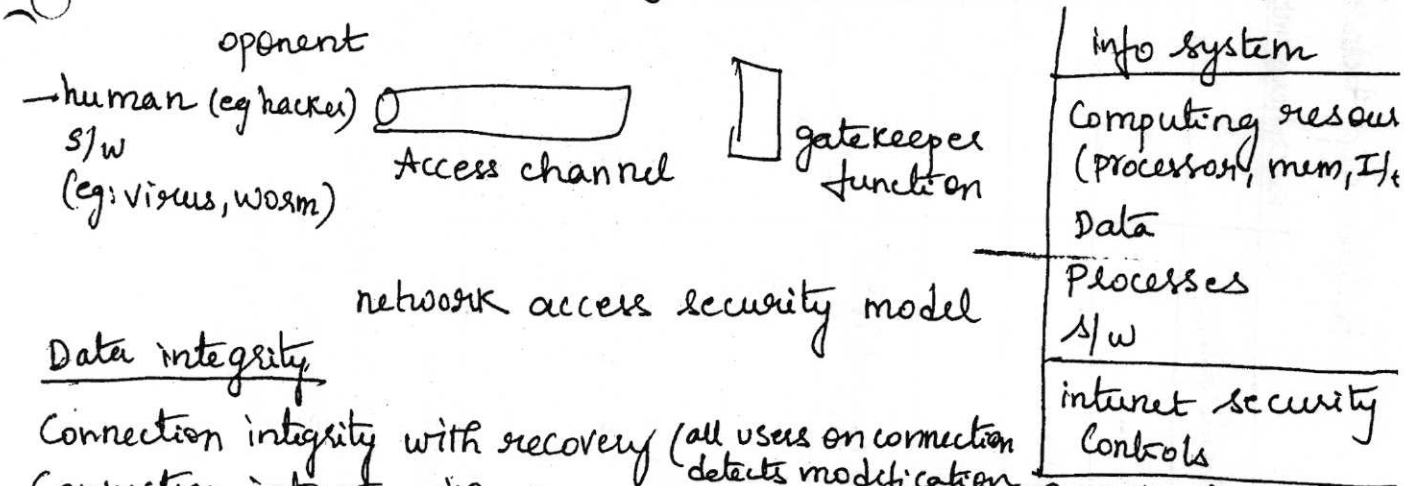
Model shows that there are four basic tasks in designing a paticular security service

1. Design an algorithm

transformation, alg should be in such a way that oponent should not defeat its purpose.

2. Generate the secret info to be used with the alg
3. Develop methods for the distribution and sharing of the secret information
4. Specify a protocol to be used by the two principals that make use of security algorithm a secret info to achieve a paticular security service.

- Another model which reflects a concern for protecting an info system from unwanted access.
- hacker can be someone who simply gets satisfication from breaking and entering a computer system (can be employee who wish to do damage or criminal who exploits Computer assets for financial gain.
- Another type of unwanted access is placement of comp system of logic that exploits vulnerabilities in the system that effect application and utility programas such as editors, compilers.



network access security model

Data integrity

Connection integrity with recovery (all users on connection detects modification with recovery)
Connection integrity without recovery (only detection without recovery)
Selective field connection integrity (selective fields within user data of a data bl
Connectionless integrity (detects modification & replay detection) determines whether selected field modifie

Relationship between security services and Mechanisms

vice	encipherment	DS	Mechanisms Access control	Data integrity	Authentication exchange	TP	R C N
.entity authentication	Y	Y			Y		
ta origin "	Y	Y					
cess control			Y				
fidentiality	Y					Y	
ffic flow confidentiality	Y					Y	
a integrity	Y	Y		Y			
repudiation		Y		Y			Y
aliability				Y	Y		

(21/12/26) Absentees

CSE-B

, 51Y, 52A, 52B, 52C, 52H, 525
P, 52X, 52Z, 532, 533, 535,
1, 53F, 53G, 53L, 5CR, 509

Symmetric encryption principles

five ingredients

- plaintext: original msg or data that is fed into the algorithm as input
- Encryption algorithm: performs various substitutions & transformations on the plaintext (create ciphertext from plaintext)
- Secret key: is also i/p to the alg, the exact substitution & transformations performed by the alg depend on the key.
- Ciphertext: scrambled msg produced as output, it depends on plaintext and secret key
- Decryption algorithm: alg run in reverse, it takes ciphertext & the same secret key and produces the original plaintext.

Two requirements for secure use of symmetric encryption

1. Need of strong encryption algorithm
   (Opponent even if he known algorithm and can access to one or more ciphertext, he is unable to decipher the ciphertext or figure out the key)
2. Sender and receiver must have obtained copies of the secret key in a secure fashion and must keep the key secure.

* Importance of symmetric encryption lies in keeping the key secret not the algorithm
  (i.e it is assumed that it is impractical to decrypt a msg on the basis of the ciphertext plus knowledge of the encryption/decryption algorithm)
- This makes it feasible for widespread use.

Cryptography

1. type of operation used for ~~transforming plaintext to ciphertext~~

Essential elements of a symmetric encryption scheme

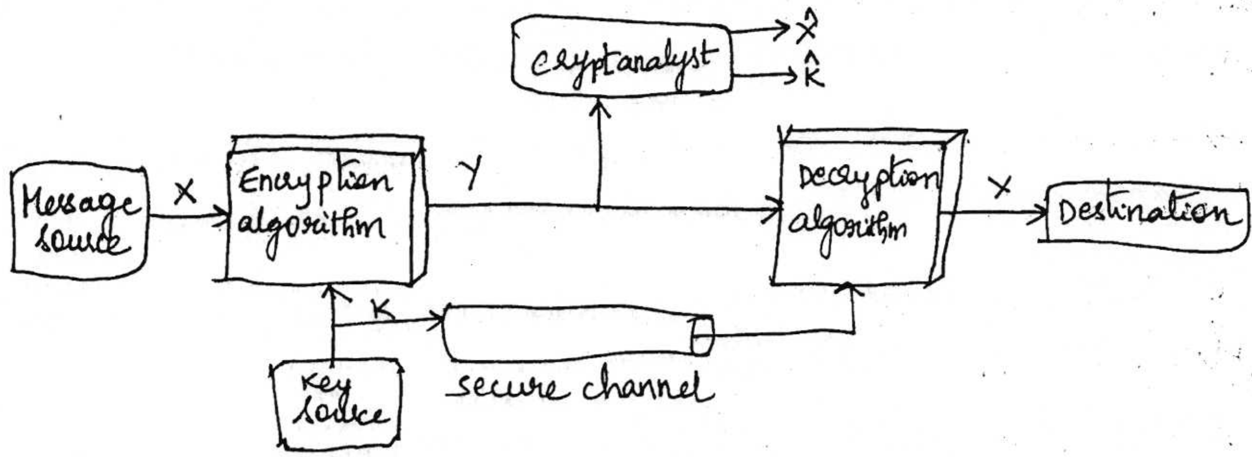


fig: Model of conventional cryptosystem.

- A source produces a message in plaintext, $X = [X_1, X_2 \cdots X_M]$. The M elements of X are letters in some finite alphabet
- Generally alphabet usually consisted of the 26 capital letters. Now-a-days a binary alphabet {0,1} is typically used.
- for encryption key K a key of the form $K = [K_1, K_2 \cdots K_J]$ is generated at msg source. if key generated at msg source it must be provided to the destination by means of some secure channel. alternatively a third party could generate the key and securely deliver it to both source and destination.
- with the msg X and the encryption key K as input the encryption algorithm forms the ciphertext $Y = [Y_1, \cdots Y_2 \cdots Y_N]$ we can write this as $Y = E_K(X)$

8121017742

(17/12/16) absentees presenties
554, 516 / 56P,
562, 55T, 572
574, 559 / 574, dip55
55U, 5L7 ...

Y is produced by using encryption alg E as a function of the plaintext X, with specific function determined by the value of the key K.

— receiver, in possession of the key, able to ~~convert~~ invert the transformation

$$X = D_K(Y)$$

opponent observing Y but not having access to K or X may attempt to recover X or K or both X and K assumed that opponent known encryption (E) and decryption (D) alg's

— If opponent is interested in only this particular message then focus of effort is to recover X by generating a plaintext estimate $\hat{X}$.
if the opponent is interested to read future messages as well, an attempt is made to recover K by generating an estimate $\hat{K}$

Cryptography: means "secret writing", science & art of transforming msg's to make them secure & not subject to attac

Three independent dimensions.

1. Type of operations used for transforming plaintext to ciphertext
All encryption algorithms are based on two general principles
substitution: each elt in plaintext is mapped to another element
transposition: elet's in plaintext are rearranged
fundamental requirement is that no information be lost.

2. number of keys used: if both sender & receiver use same key it is symmetric, single-key, secret key or conventional encryption
• if sender and receiver uses different keys it is asymmetric two key or public key encryption

3. Way in which plaintext is produced block cipher processes input one block of elt's at a time producing an o/p of each i/p blo

Cryptanalysis : Art of breaking codes, helps us to create better secret codes
4 approaches to attacking a encryption scheme.

• Cryptanalysis: cryptanalytic attacks rely on nature of the alg plus perhaps some knowledge of the general characteristics of the plaintext or some samples plaintext-ciphertext pairs

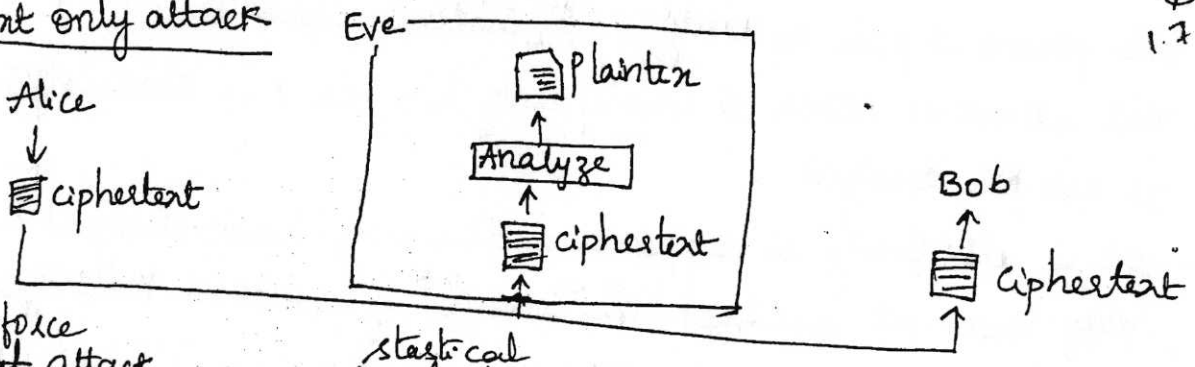
– This attack exploits the characteristics of the algothim to attempt to deduce a specific plaintext or to deduce the key being used if attack succeeds in deducing the key the effect is catastrophic: all future & past msg's encrypted with the key are compromised

• Brute-force attack: attacker tries every possible key on piece of cipher text untill an intelligible translation into plaintext is obtained

– Various types of cryptanalytic attacks based on the amount of information known to the cryptanalyst.

Type of attack	Known to cryptanalyst
1. Ciphertext only	• encryption alg • ciphertext to be decoded
2. Known ciphertext	• encryption alg • Ciphertext to be decoded • one or more plaintext-cipher text pairs formed with the secret key.
3. choosen plaintext	• Encryption alg • ciphertext to be decoded • plaintext msg choosen by crypt analyst together with its corresponding ciphertext generat with the secret key
4. choosen ciphertext	

Ciphertext only attack



Brute-force ~~Statistical~~ attack

Eve use all possible keys., Eve known alg and key domain, eve decrypts ciphertext with every possible key until plaintext makes sense, known bruteforce attack is easier today using a computer. TO prevent the no of possible keys must be very large

~~Brute-force~~ statistical attack

Cryptanalyst benifits from some inherent characteristics of the plaintext language to launch a statistical attack

ex: letter e is most frequently used letter in english text

- Cryptanalyst finds mostly used character in the ciphertext & assumes that the corresponding plaintext character is E after finding few pairs analyst finds the key & use it to decrypt

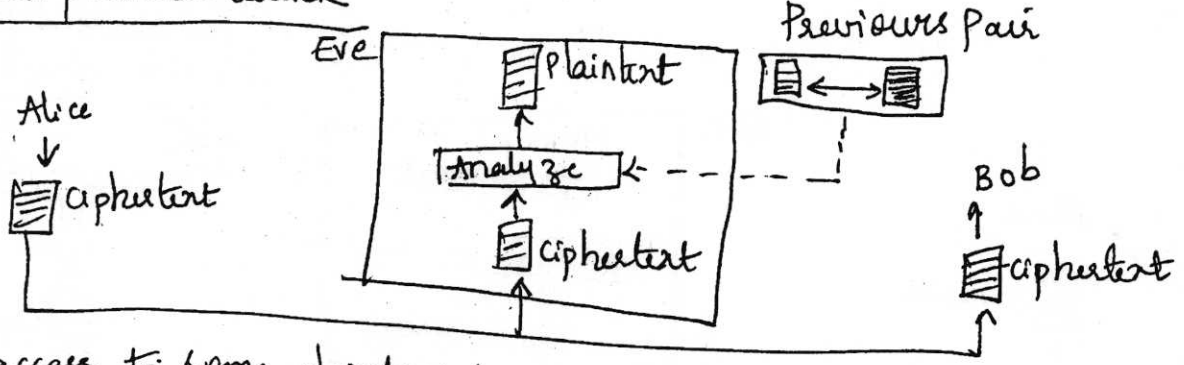
Prevent this ciphertext should hide the characteristic of the lang

pattern attack

some ciphers hide the characteristics of the lang but may create some patterns in the ciphertext cryptanalyst make use of pattern attack to break the ciphers

∴ it is imp to use ciphers that makes the ciphertext look as random as possible

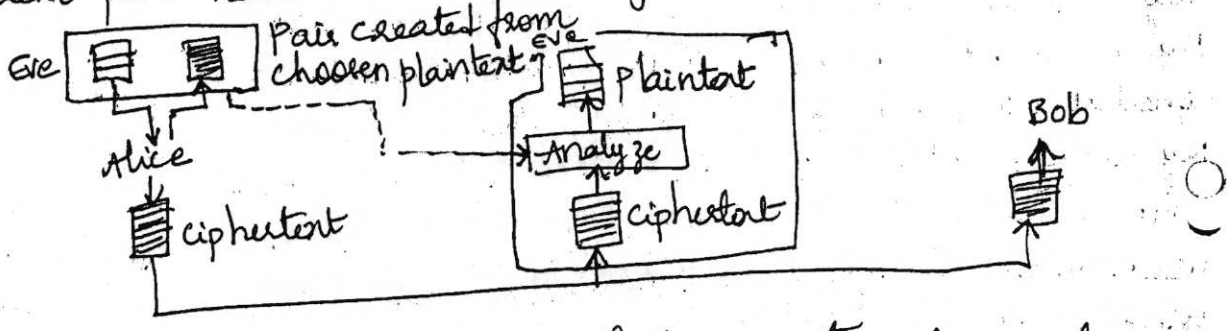
Known plaintext attack



Eve access to some plaintext/ciphertext pairs in addition to the intercepted ciphertext that she wants to break., plaintext/ciphertext pairs are collected earlier.

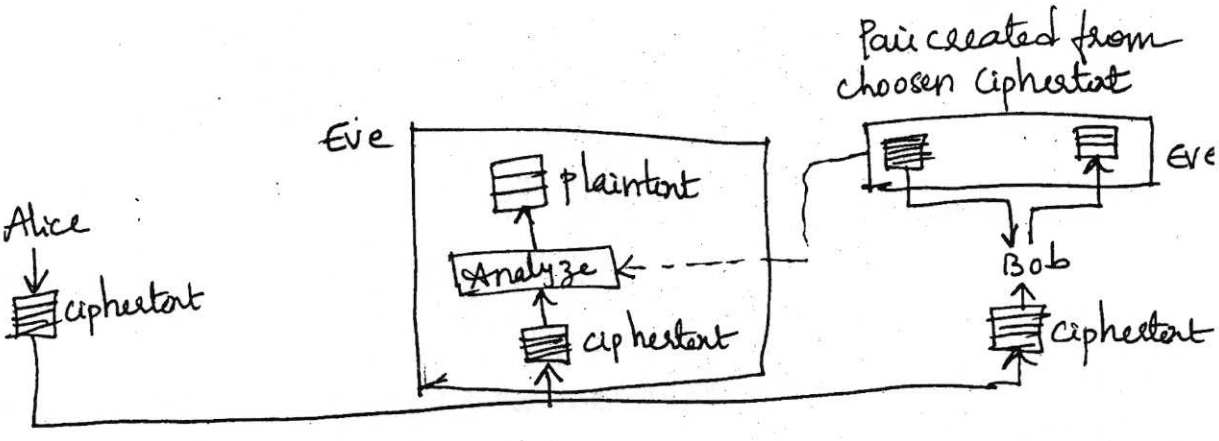
the previous pair to analyze the current ciphertext
- This attack is easier to implement bcz eve has more information to use for analysis
- It is less likely to happen bcz Alice may have changed her key or may have not disclosed the contents of previous messages

Chosen plaintext attack : It is to known plaintext but the plaintext/ ciphertext pairs have been choosen by the attacker herself



This happens if eve has access to Alice's computer, she can choose some plaintext and intercept the created ciphertext, she doesn't have key bcz it is embedded in the s/w used by the sender
- It is much easier to implement but it is much less likely to happen

Choosen-ciphertext Attack : It is to choosen plaintext attack except that eves choosen some ciphertext & decrypts it to form a ciphertext/ plaintext pair, this can happen if eve have access to Bob's comput



4. choosen text

- purpoted ciphertext choosen by cryptanalyst together with decrypted plaintext generated with the secret key.
- same
- same
- plaintext choosen together with ciphertext generated with the secret key
- Pearported (assumed) ciphertext generated wi decrypted plaintext generated with the secret key.

(1) Any encryption scheme is unconditionally secure

If the ciphertext generated by the scheme does not contain enough information to determine uniquely the corresponding plaintext, no matter how much cipher text is available.

i.e how much time an opponent has it is impossible for him or her to decrypt the ciphertext simply bcz required info is not there.

(2) encryption scheme is Computationally secure

- the cost of breaking the cipher exceeds the value of the encrypted information
- The time required to break the cipher exceeds the useful lifetime of the information

key size(bits)	No of alternative keys	Time required at 1 encryption/μs	Time required at 10^6 encryption/μs
32	$2^{32} = 4.3 \times 10^9$	2^{31} μs = 35.8 min	2.15 milliseconds
56	$2^{56} = 7.2 \times 10^{16}$	2^{55} μs = 1142 yrs	10.01 hrs
128	$2^{128} = 3.4 \times 10^{38}$	2^{127} μs = 5.4×10^{24} yrs	
168	$2^{168} = 3.7 \times 10^{50}$	2^{167} μs = 5.9×10^{36} yrs	

Substitution cipher techniques

Caesar cipher : involves replacing each letter of the alphabet with the letter standing three places further down the alphabet

ex: plain: meet me after the party
Cipher: PHHW PH DIWHU WKH SDUWB

plain: a b c d e f g h i j k l m n o p q r s t u v w x y z
Cipher: D E F G H I J K L M N O P Q R S T U V W X Y Z A B C

Numerical equivalent to each letter

a	b	c	d	e	f	g	h	i	j	k	l	m	n	o	p	q	r	s	t	u	v	w	x	y
0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24

alg can be expressed as follows
for each plaintext letter p substitute the ciphertext letter C^3

$$C = E(P) = (P+3) \bmod (26)$$

A shift may be of any amount, so that general Caesar alg is

$$C = E(P) = (P+K) \bmod (26)$$

where K takes a value in the range 1 to 25 decryption alg

$$P = D(C) = (C-K) \bmod (26)$$

Monoalphabetic ciphers

"MEET me"

numerical representation 12 4 4 19 12 4

15 7 7 22 15 7

P H H W P H

e = 3
M = 12
C = 12+3 mod 26
C = 15 mod 26
15 = P

Encryption process
$C = m + e \bmod 26$
Decryption process
$m = C + e \bmod 6$

c = ciphe
m = mes
e =
d =

Polyalphabetic ciphers : each occurrence of a character may have a different substitute, relationship between a character in the plaintext to a character in the ciphertext is one-to-many

Playfair cipher : used by british army durning World War I. the secret key in this cipher is made of 25 alphabet letters arranged in a 5×5 matrix, different letters in the matrix can create many different secret keys

should have ke
rules: no repeati
letters (Keyword,

Secret key =

L	G	D	B	A
Q	M	H	E	C
U	R	N	I/J	F
X	V	S	O	K
Z	Y	W	T	P

K E Y W O
R D A B C
F G H I L
M N Q R S
T U V X Z

NORDKUNKQZPCN

Before encryption if the two letters in a pair are the same, a bogus letter is inserted to seperate them, after inserting bogus letter if no of characters in the plaintext is odd, one extra bogus character is added at the end to make the number of characters even.

Cipher uses three rules for encryption

1. If the two letters in a pair are located in the same row of the secret key, the corresponding encrypted character for each letter is the next letter to the right in the same row.
2. If the two letters are located in same column of the secret key the corresponding encrypted character for each letter is the letter beneth it in the same column.
3. If two letters are not in same row or colu...

but in the same column as the other letter.

$P = P_1 P_2 P_3 \cdots$ $\quad C = C_1 C_2 C_3 \cdots$ $\quad K = [(K_1, K_2), (K_3, K_4) \cdots]$

Encryption: $C_i = K_{P_i}$ $\quad$ Decryption: $P_i = K_{i}$.

ex: "hello"

group letters into two-character pair we get

"he ll o" we need to insert x in between two l's

"he, lx, lo" we have

he → EC $\quad$ lx → QZ $\quad$ lo → BX

Plaintext: hello $\quad$ ciphertext: ECQZBX

the two occurrences of the letter "l" (el) are encrypted as "Q" & "Z"

i.e polyalphabetic cipher.

Hill cipher : invented by Lester S. Hill

- the plaintext is divided into equal-size blocks, the blocks are encrypted one at a time in such a way that each character in the block contributes to the encryption of other characters in the block for this reasons hill cipher comes under block cipher so far studied are stream ciphers.
- In hill cipher the key is a square matrix of size m x m in which m is the size of the block, if we call the key matrix K each element of the matrix is $K_{i,j}$ as shown

$$K = \begin{bmatrix} K_{11} & K_{12} & \cdots & K_{1m} \\ K_{21} & K_{22} & \cdots & K_{2m} \\ \vdots & \vdots & & \vdots \\ K_{m1} & K_{m2} & \cdots & K_{mm} \end{bmatrix}$$

SECRET HE SX SA

KEYWORD

How one block of ciphertext is encrypted, if we call m characters in the plaintext block $P_1, P_2, \ldots P_m$ the corresponding characters in the ciphertext block are $C_1, C_2, \ldots C_m$ then we have

$$C_1 = P_1K_{11} + P_2K_{21} + \cdots + P_mK_{m1}$$
$$C_2 = P_1K_{12} + P_2K_{22} + \cdots + P_mK_{m2}$$
$$C_m = P_1K_{1m} + P_2K_{2m} + \cdots + P_mK_{mm}$$

equations shows each ciphertext character such as C_1 depend on all plaintext characters in the block $(P_1, P_2 \ldots P_m)$ however we should be aware that not all square matrices have multiplicative inverses in Z_{26} so Alice & Bob should be careful in selecting key. Bob will not be able to decrypt the ciphertext sent by Alice if matrix does not have a multiplicative inverse.

* key matrix in the hill cipher needs to have a multiplicative inverse

ex: "code is ready" cipher: "OHKNIHGKLISS"

$$\underset{C}{\begin{bmatrix} 14 & 07 & 10 & 13 \\ 8 & 7 & 6 & 11 \\ 11 & 8 & 18 & 18 \end{bmatrix}} = \underset{P}{\begin{bmatrix} 2 & 14 & 3 & 4 \\ 8 & 8 & 17 & 4 \\ 0 & 3 & 24 & 25 \end{bmatrix}} \underset{K}{\begin{bmatrix} 9 & 7 & 11 & 13 \\ 4 & 7 & 5 & 6 \\ 2 & 21 & 14 & 9 \\ 3 & 23 & 21 & 8 \end{bmatrix}}$$

encryption

$$\underset{P}{\begin{bmatrix} 2 & 14 & 3 & 4 \\ 8 & 18 & 17 & 4 \\ 0 & 3 & 24 & 25 \end{bmatrix}} = \begin{bmatrix} 14 & 7 & 10 & 13 \\ 8 & 7 & 6 & 11 \\ 11 & 8 & 18 & 18 \end{bmatrix} \underset{K^{-1}}{\begin{bmatrix} 2 & 15 & 22 & 3 \\ 15 & 0 & 19 & 3 \\ 9 & 9 & 3 & 11 \\ 17 & 0 & 4 & 7 \end{bmatrix}}$$

decryption.

- Cipher-text only is difficult
- brute force attack is difficult (bcz of $m \times m$ matrix)
- do not preserve the statistics of the plaintext

numbers contained within square bracket

examples $\begin{bmatrix} 2 & 3 \\ 1 & 4 \end{bmatrix}$ 2×4 matrix $\begin{bmatrix} 2 & 3 \\ 1 & 4 \\ 7 & -2 \end{bmatrix}$ 3×2 matrix

Simple Addition

$$\begin{bmatrix} 4 & 2 & 3 \\ 5 & 7 & 6 \end{bmatrix} + \begin{bmatrix} 1 & 5 & 9 \\ 3 & 5 & 4 \end{bmatrix} = \begin{bmatrix} 4+1 & 2+5 & 3+9 \\ 5+3 & 7+5 & 6+4 \end{bmatrix} = \begin{bmatrix} 5 & 7 & 12 \\ 8 & 12 & 10 \end{bmatrix}$$

Simple substraction

$$\begin{bmatrix} 3 & 9 & 4 \\ 2 & 7 & 6 \end{bmatrix} - \begin{bmatrix} 4 & 2 & 1 \\ 3 & 1 & 4 \end{bmatrix} = \begin{bmatrix} 3-4 & 9-2 & 4-1 \\ 2-3 & 7-1 & 6-4 \end{bmatrix} = \begin{bmatrix} -1 & 7 & 3 \\ -1 & 6 & 2 \end{bmatrix}$$

Simple sclar ~~matrix~~ multiplication

$$4\begin{bmatrix} 2 & 2 & 3 & 1 \\ 1 & 2 & 7 & 4 \end{bmatrix} = \begin{bmatrix} 4\times2 & 4\times2 & 4\times3 & 4\times1 \\ 4\times1 & 4\times2 & 4\times7 & 4\times4 \end{bmatrix} = \begin{bmatrix} 8 & 8 & 12 & 4 \\ 4 & 8 & 28 & 16 \end{bmatrix}$$

Simple multiplication

$$\begin{bmatrix} 4 & 5 \\ 3 & 4 \end{bmatrix}\begin{bmatrix} 7 \\ 2 \end{bmatrix} = \begin{bmatrix} (4\times7)+(5\times2) \\ (3\times2)+(4\times2) \end{bmatrix} = \begin{bmatrix} 77+10 \\ 21+8 \end{bmatrix} = \begin{bmatrix} 87 \\ 29 \end{bmatrix}$$

Simple modular matrix

$$\begin{bmatrix} 3 & 7 \\ 5 & 12 \end{bmatrix}\begin{bmatrix} 8 \\ 5 \end{bmatrix} = \begin{bmatrix} (3\times8)+(7\times5) \\ (5\times8)+(12\times5) \end{bmatrix} = \begin{bmatrix} 24+35 \\ 40+60 \end{bmatrix} = \begin{bmatrix} 59 \\ 100 \end{bmatrix}$$

$\begin{bmatrix} 7 \\ 22 \end{bmatrix} \bmod 26$ Dec: $P = CK^{-1} \bmod 26$

Bob and Alice accept this matrix as Key $\begin{bmatrix} 3 & 3 \\ 2 & 5 \end{bmatrix}$

Alice wants to send bob the message HATS

H+A	TS
7, 0	19, 18

To encrypt HA $\begin{bmatrix} 3 & 3 \\ 2 & 5 \end{bmatrix}\begin{bmatrix} 7 \\ 0 \end{bmatrix} = \begin{bmatrix} (7\times3)+(0\times3) \\ (7\times2)+(0\times5) \end{bmatrix} = \begin{bmatrix} 21 \\ 14 \end{bmatrix} \bmod 26$

Encrypt TS $\begin{bmatrix} 3 & 3 \\ 2 & 5 \end{bmatrix}\begin{bmatrix} 19 \\ 18 \end{bmatrix} = \begin{bmatrix} (19\times3)+(18\times3) \\ (19\times2)+(18\times5) \end{bmatrix} = \begin{bmatrix} 111 \\ 38 \end{bmatrix} \bmod 26$

$= \begin{bmatrix} 7 \\ 24 \end{bmatrix} \bmod 26$

21, 14, 7, 24 = VOHY

Decryption Inverse: $\begin{bmatrix} a & b \\ c & d \end{bmatrix}$ becomes $\begin{bmatrix} d & -b \\ -c & a \end{bmatrix}$ Key $\begin{bmatrix} 3 & 3 \\ 2 & 5 \end{bmatrix}$ becomes $\begin{bmatrix} 5 & -3 \\ -2 & 3 \end{bmatrix}$

formula is determinant: $\begin{bmatrix} a & b \\ c & d \end{bmatrix}$ determinant is calculated by ad-bc

determinant of Key $\begin{bmatrix} 3 & 3 \\ 2 & 5 \end{bmatrix} = (3\times5)-(3\times2) = 15-6 = 9$

write

$$K = \begin{bmatrix} 17 & 17 & 5 \\ 21 & 8 & 21 \\ 2 & 2 & 16 \end{bmatrix}$$

WY i te x

(22 17 8) (19 4 23)

$$(22\ 17\ 8)\begin{bmatrix} 17 & 17 & 5 \\ 21 & 8 & 21 \\ 2 & 2 & 16 \end{bmatrix}$$

$= (22\times17)+(17\times21)+(8\times2)$

$(22\times17)+(17\times8)+(8\times2)$

$(22\times5)+(17\times21)+(8\times16)$

(747 696 619) mod 26

(19 20 21) (t v u)

tex ⇒ (453 441 614) = (11 25 18)

Hill cipher

Meet tonight

$\begin{pmatrix} 3 & -2 \\ -1 & 1 \end{pmatrix}$

When 9 is multiplied by its inverse 1 over 9 the answer is 1

$9 \times \frac{1}{9} = 1$

bob now needs to find what to multiply 9 by to get the result 1 mod 26

$9 \times \frac{3}{1} = 1 \bmod 26 \quad (9 \times 3 = 27)$

Now bob can calculate modular 26 of the inverse of the encryption key

$\begin{bmatrix} 5 & -3 \\ -2 & 3 \end{bmatrix}^{-1} = \begin{bmatrix} 5 & 23 \\ 24 & 3 \end{bmatrix}$

modular 26 of 5 is 5
−3 is 23
−2 is 24
3 is 3

then bob multiples this matrix by the 3 from the previous calculations

$3 \begin{bmatrix} 5 & 23 \\ 24 & 3 \end{bmatrix} = \begin{bmatrix} 15 & 69 \\ 72 & 9 \end{bmatrix} \bmod 26$

$= \begin{bmatrix} 15 & 17 \\ 20 & 9 \end{bmatrix}$ → this is decryption key

Bob now has the decryption key & the ciphertext "VOHY"

Decryption key $\begin{bmatrix} 15 & 17 \\ 20 & 9 \end{bmatrix}$ VOHY

VO HY

21, 14 7, 24

Decrypt VO: bob multiplies key by 21 & 14

$\begin{bmatrix} 15 & 17 \\ 20 & 9 \end{bmatrix} \begin{bmatrix} 21 \\ 14 \end{bmatrix} = \begin{bmatrix} (21 \times 15) + (14 \times 17) \\ (21 \times 20) + (14 \times 9) \end{bmatrix} = \begin{bmatrix} 583 \\ 546 \end{bmatrix} \bmod 26$

Decrypt HY:

$\begin{bmatrix} 15 & 17 \\ 20 & 9 \end{bmatrix} \begin{bmatrix} 7 \\ 24 \end{bmatrix} = \begin{bmatrix} (7 \times 15) + (24 \times 17) \\ (20 \times 7) + (24 \times 9) \end{bmatrix} = \begin{bmatrix} 513 \\ 356 \end{bmatrix} \bmod 26$

$= \begin{bmatrix} 7 \\ 0 \end{bmatrix}$

$= \begin{bmatrix} 19 \\ 18 \end{bmatrix}$

So the numerical representation for 7, 0, 19, 18 is "HATS"

A B C D E F G H I J K L M N O P Q R S T U V W X Y Z
0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25

ex:
MATH 12, 0 19, 7 Key $\begin{pmatrix} 3 & 2 \\ 1 & 5 \end{pmatrix}$ using A = 0 working in mod 26.

Polyalphabetic ciphers

— use different monoalphabetic substitutions

features common

1. A set of monoalphabetic substitution rules is used
2. A key determines which particular rule is choosen for a given transformation.

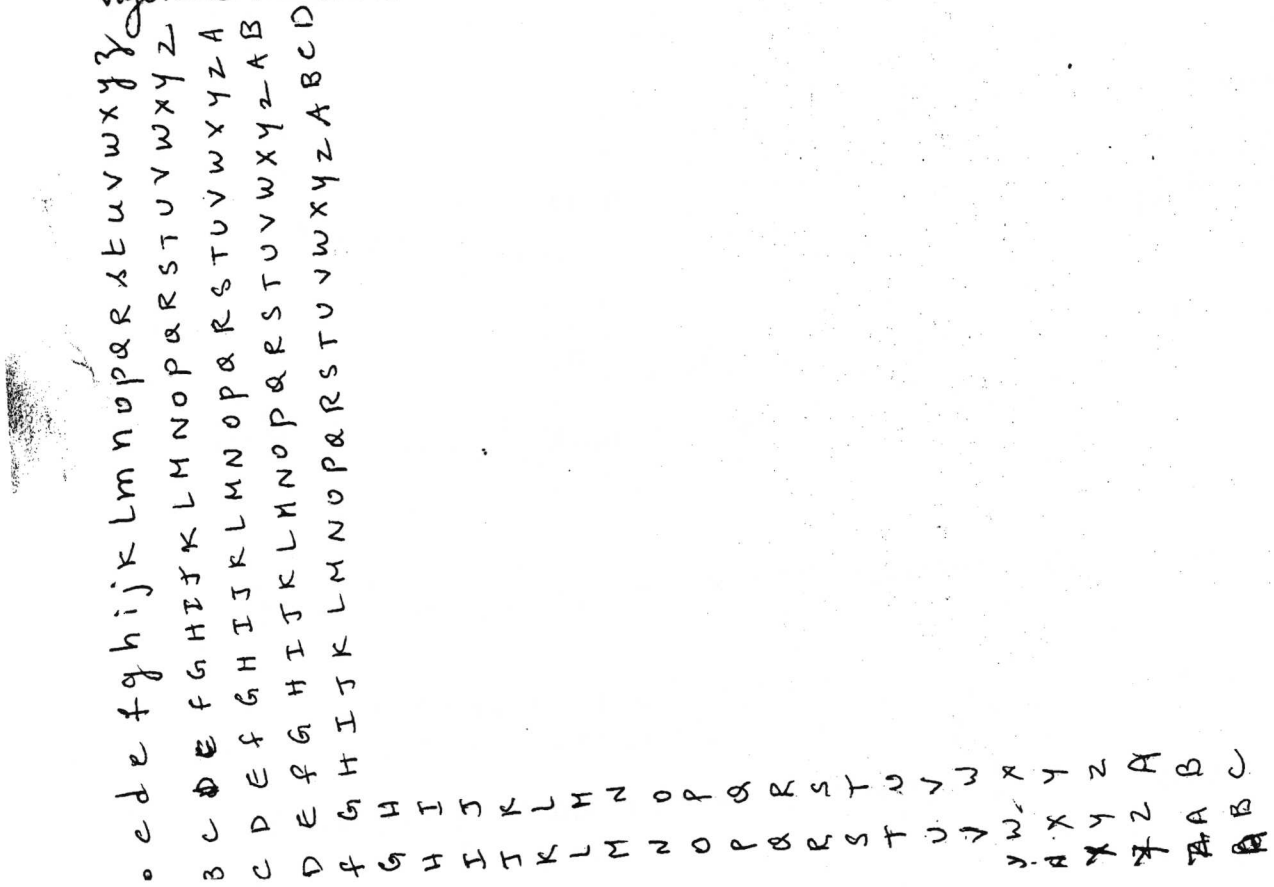
Vigenere cipher — Blaise de Vigenere

the set of monoalphabetic substitutions rules consists of the 26 ceasar ciphers with shifts of 0 through 25.

— each cipher is denoted by a key letter, which is the cipher letter that substitutes for the plaintext letter a.

thus ceaser cipher with shift of three is denoted by d.

Vigenere matrix



Encryption: Given a key letter x and a plaintext letter y the ciphertext letter is at the intersection of the row labeled x and the column labeled y in this case ciphertext is V.

- usually a key is an repeating keyword

ex: key is deceptive message is "we are discovered save yourself"

Key: deceptivedeceptivedeceptive
Plaintext: wearediscoveredsaveyourself
Ciphertext: ZICVTWQNGRZGVTWAVZHCQYGLMGJ

Decryption: is equally simple, key letter again identifies the row position of the ciphertext letter in that row determines the column and plaintext is at the top of that column. (see d in row and go till z is there & stop & see the top what letter is there w)

- the strength of this cipher is that there are multiple ciphertext letters for each plaintext letter, one for each unique letter of the keyword thus the letter frequency info is obscured.

ex:2 "she is listening" using key "PASCAL"
Ciphtext HHWKSWXSLGNTCG

- ✱ Vigenere key stream does not depend on the plaintext characters
- it depends only on position of the character in the plaintext i.e the key stream of ~~length~~ m can be created without knowing what the plaintext is.

- Another way: key is repetition of an initial secret key stream of length m. $(1 \le m \le 26)$

$P = P_1 P_2 P_3 \cdots \quad C = C_1 C_2 C_3 \cdots \quad K = \{(k_1, k_2, \cdots k_m)(k_1, k_2, \cdots k_m) \cdots\}$

Encryption: $C_i = P_i + K_i$ Decryption $P_i = C_i - K_i$

S	h	e	i	s	L	i	s	t	e	n	i	n	g : plaintext
18	7	4	8	18	11	8	18	19	4	13	8	13	6 : P's values
15	0	18	2	0	11	15	0	18	2	0	11	15	0 : key stream
7	7	22	10	18	22	23	10	11	.	.	.		

Key does not depend on the plaintext characters, it depends only on the position of the character in the plaintext (key can be created without knowing what is plaintext)

Joseph Mauborgne

One time pad : The sender changes the key each time she sends a message using another random sequence of integers.

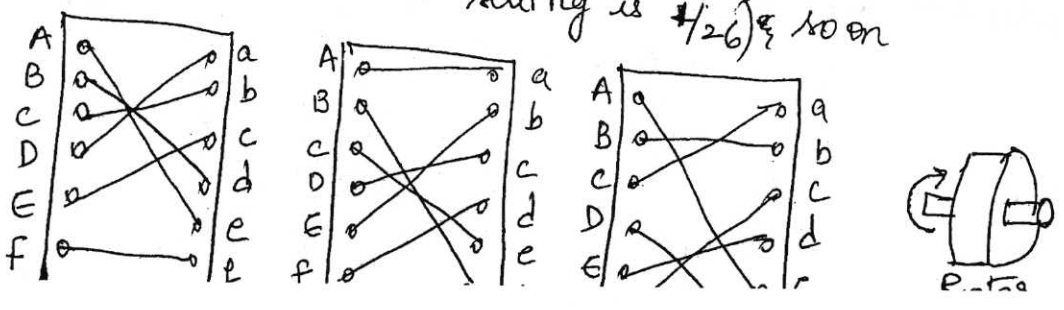
- The idea is used in one time pad invented by Vernam, In this cipher the key has the same length as the plaintext and is choosen completely in random
- It is usually not implemented bcz every the key changes how can alice tell to bob, but there are some occassions when one time pad can be used ex: president of a country needs to send a completely secret message to the president of another country she can send a trusted envoy with the random key before sending the message.

Rotor cipher : it uses idea ~~between~~ behind monoalphabetic substitution but changes the mapping between the plaintext & the ciphertext characters for each plaintext character.

ex: rotor shows only 6 letters but actual rotor use 26 letters

- the rotor is permanently wired. but connection to encryption/decryp characters is provided by brushes.

initial setting of rotor is the secret key between Alice & Bob.
first plaintext character is encrypted using the initial settings
second character is encrypted after the first rotation (at fig. at 1/6 turn, but the actual setting is 1/26) & so on



Before DES the important application of the principle of multiple stages of encryption was a class of systems known as rotor m/c

— basic principle of rotor m/c. The m/c consists of set of independently rotating cylinders through which electricity pluses can flow.

— each cylinder has 26 i/p pins & 26 o/p pins with internal wiring that connects each input pin to a unique o/p pin

— If we associate each i/p & o/p pin with a letter of the alphabet then a single cyclinder defines a monoalphabetic substitution cipher is defined, after 26 letters of plaintext cylinder would be back to the initial position thus we have a polyalphabetic substitution algorithm. with a period of 26.

---

ex: bee is encrypted as BAA if not rotated ; BCA if it is rotated this shows rotor is polyalphabetic cipher bez two occurrences of the same plaintext characters are encrypted ~~letter~~ as different characters.

— rotor cipher is resistant to brute force as the monoalphabetic substitution cipher bez eve still needs to find the first set of mappings among 26! possible ones

— It is resistant to statistical attack than the monoalphabetic substitution cipher bez it does not preserve letter frequency.

Transposition techniques

Rail fence technique

(changes the location of the symbols)

Plaintext is written down as a sequence of diagonals and then read off as a sequence of rows

ex: "meet me after the toga party" with rail fence of depth 2

```
m e m a t r h t g p r y
 e t e t e t e o a a t
```

Cipher: MEMATRHTGPRY
TETETEOAAT

more complex scheme is to write the message in rectangle row by row, and read the message off col by col but permute the order of the columns

- order of columns then becomes the key to the algorithm

ex: Key: 4 3 1 2 5 6 7

Plaintext:

a	t	t	a	c	k	p
o	s	t	p	o	n	e
d	u	n	t	i	l	t
w	o	a	m	x	y	z

Ciphertext: TTNAAPTHTSUOAODWCOIXKNLYPETZ

- More than one stage of transposition, result is more complex permutation that is not easily reconstructed

Key: 4 3 1 2 5 6 7

Input:

t	t	n	a	a	p	t
m	t	s	u	o	a	o
d	w	c	o	i	x	k
n	l	y	p	e	t	z

o/p: NSCYAUOPTTWLTMDNAOIEPAXTTOKZ

To visualize, 28 (7×2) matrix letters

M1)

Original sequence

01 2 3 4 5 6 7 8 9 10 11 12 13 14

15 16 17 18 19 20 21 22 23 24 25 26 27 28

After first transposition

3 10 17 24 04 11 18 25 2 9 16 23 01 8

15 22 5 12 19 26 6 13 20 27 7 14 21 28

After second

17 9 5 27 24 16 12 7 10 2 22 20 3 25

15 13 4 23 19 14 11 1 26 21 18 8 6 28

Keyless transposition ciphers : 2 methods for permutation of characters

- text is written into a table column by column & ~~permutation of ch~~ then transmitted row by row.
- text is written into a table row by row and transmitted column by column.

Keyed transposition ciphers : divide plaintext into groups of predetermined size called blocks and then use a key to permute the characters in each block seperately.

ex: "Enemy attacks to night" ~~key:~~ group five characters

enemy attac·kston ightz

Key: 3 1 4 5 2 ↑ decryption

encryption ↓ 1 2 3 4 5

- 3rd character in plaintext becomes first character in ciphertext

1st second

EEMYN TAACT TKONS HITZG — ciphertext

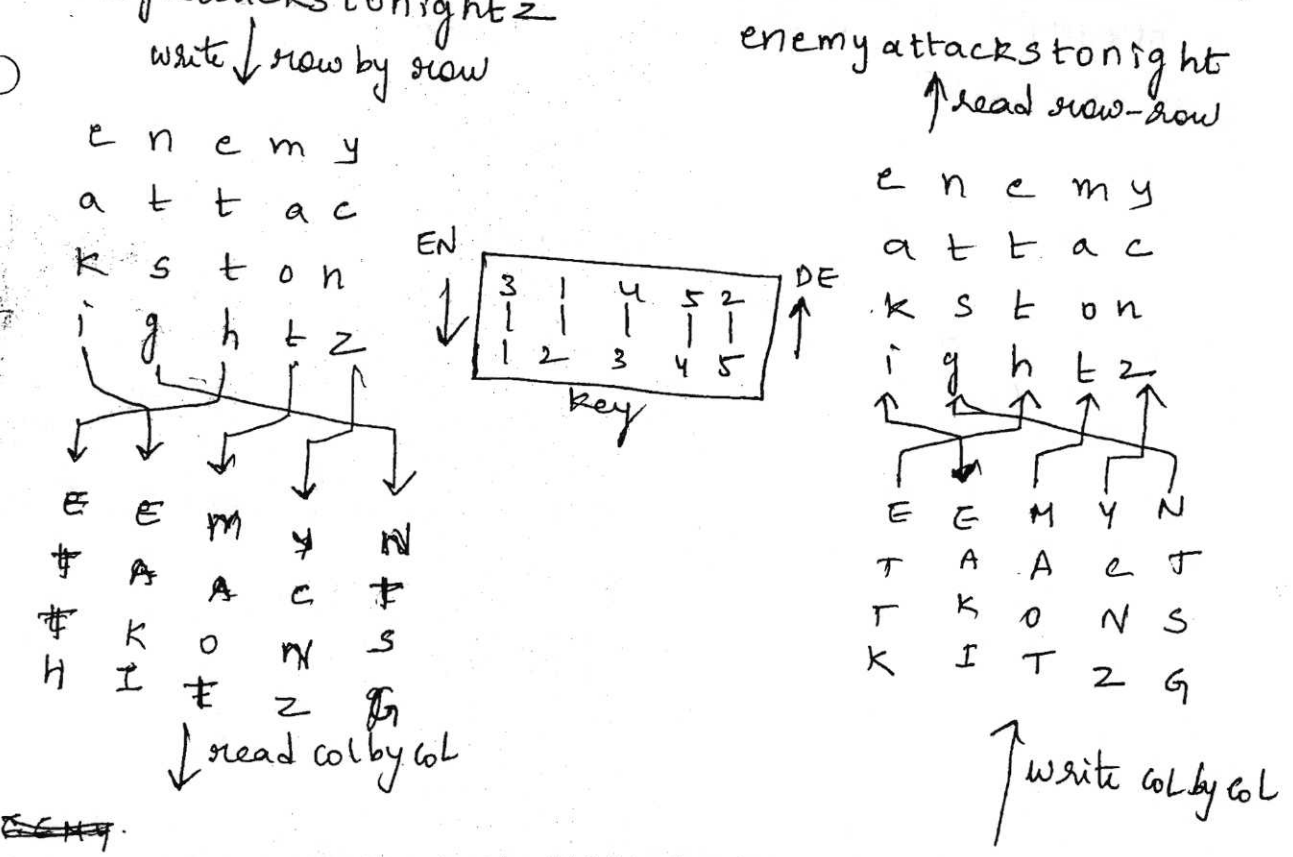
bob uses the key in reverse order to find plaintext.

Combining two approaches

— Encryption & decryption done in three steps

1) Text is written into table row by row
2) Permutation is done by reordering the columns
3) new table is read column by column

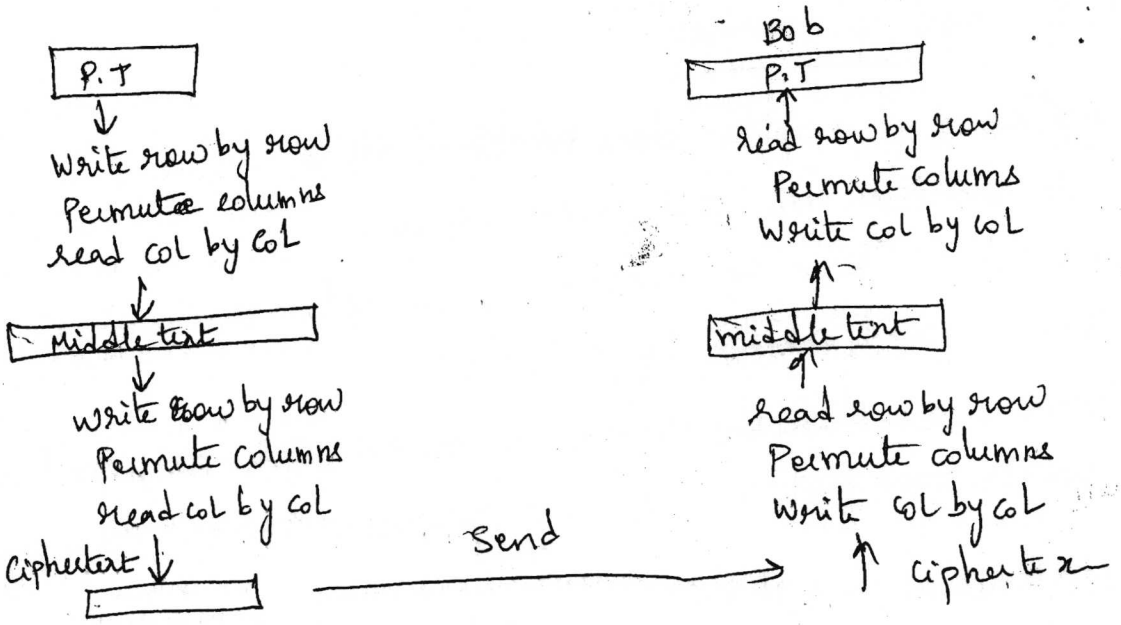
1 & 3 keyless, 2 — blockwise keyed reordering often referred as keyed columnar transposition ciphers or columnar transposition ciphers.



ETTHEAKIMAOTYCTTKONSHITZG → ETTHEAKIMAOTYCTTKONSI
ciphertext transmission Tz

Double transposition cipher

- Repeat twice the algorithm used for encryption & decryption (a different key can be used in each step but normally the same key is used)


P.T
Write row by row
Permute columns
read col by col
Middle text
write row by row
Permute columns
read col by col
Ciphertext
Send
Bob
P.T
read row by row
Permute columns
Write col by col
middle text
read row by row
Permute columns
Write col by col
Ciphertext

Steganography

Plaintext message may be hidden in one of two ways
Steganography : conceal (hide) the existence of the message
Cryptography : render the message unintelligible to outsiders by various transformations of the text

— simple form of steganography but time consuming to construct, is one in which arrangement of words or letters within an apparently- innocuous text spells out the real message.

Some techniques

- Character marking : Selected letters of printed or typewritten text are overwritten in pencil, the marks are ordinarily not visible unless the paper is held at an angle to bright light.
- Invisible ink : Substances can be used for writing but leave no visible trace until heat or some chemical is applied to the paper
- Pin punctures : Small pin punctures on selected letters are ordinarily not visible unless the paper is held up in front of a light.
- Typewriter correction ribbon ; used between lines are typed with a black ribbon. the result of typing with the correction tape are visible only under a strong light.

— It has number of drawbacks when compared to encryption
it requires lot of overhead to hide a relatively few bits of info
Once the system is discovered it becomes virtually worthless
this problem too can be overcome if the insertion methods depends on some key (alternatively msg can be first encrypted & then hidden using steganography)

— Advantages is it can be employeed by parties who have something to lose should the fact of their secret

Block cipher principles

Most symmetric block encryption algorithms in current use are based on a structure referred to as feistel block cipher.

Stream cipher: encrypts digital data stream one bit or one byte at a time ex: autokey vigenere cipher, vernam cipher.

Block cipher: in it a block of plaintext is treated as a whole & used to produce a ciphertext block of equal length.

block size of 64 or 128 bits

- block ciphers are vastly used in network based symmetric cryptographic applications

feistel block cipher structure

- block cipher operates on a plaintext block of n bits to produce a ciphertext block of n bits, there are 2^n possible different plaintext blocks and for encryption to be reversible each must produce a unique ciphertext block.

Reversible mapping

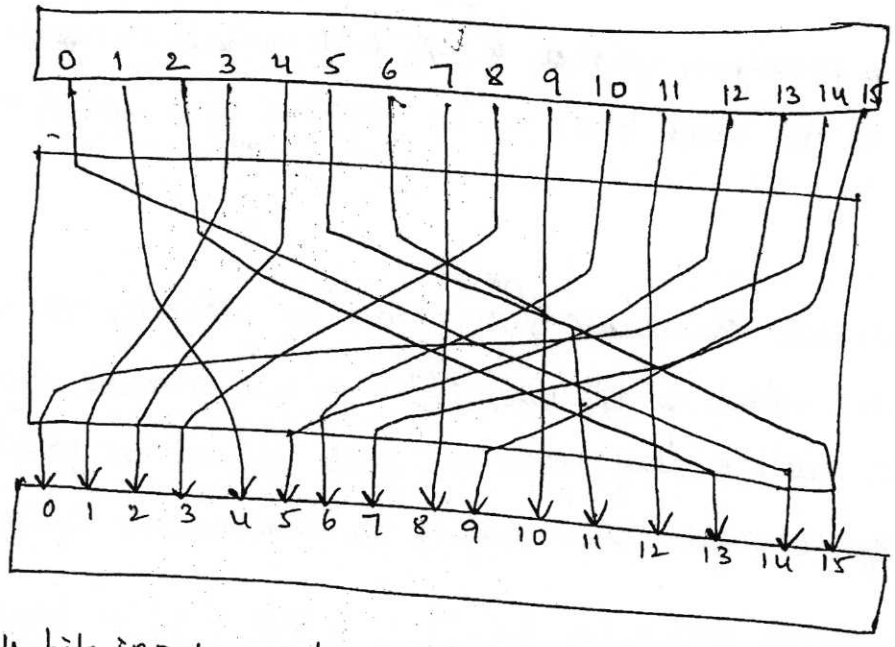
Plaintext	Ciphertext
00	11
01	10
10	00
11	01

Irreversible mapping

Plaintext	Ciphertext
00	11
01	10
10	01
11	01

In reversible mappings the number of different transformations is $2^n!$.

General n-bit n-bit block substitution (n=4)



A 4-bit input produces one of 16 possible input states, which is mapped by the substitution cipher into a unique one of 16 possible output states, it is general form of block cipher and can be used to define any reversible mapping between plaintext and ciphertext Feistel refers to that as the ideal block cipher.

- the problem here is if n is small it is same as substitution cipher, but we have seen such systems are vulnerable to a statistical analysis of the plaintext
- If n is sufficiently large and arbitary reversible substitution between plaintext & ciphertext is allowed then statistical characteristics of the source plaintext are masked to such an extent that this type of cryptanalysis is infeasible.
- but practically not it is not possible, for performance & implementation-point of view, mapping can have key (Paticular reversable mapping) method of defining key the required key length is (4 bits) x (16 rows) = 64 bits
- for an n-bit ideal

for 64-bit block, required key length is $64 \times 2^{64} = 2^{70} \approx 10^{21}$ bits.

- To prevent this problem feistel points out what is needed is an approximation to the ideal block cipher system for large n.

Feistel cipher

- We can approximate the ideal block cipher by utilizing the concept of a product cipher which is the execution of two or more simple ciphers in sequence in such a way that the final result or product is cryptographically stronger than any of the component ciphers
- develop a block cipher with a key length of k bits and a block length of n bits allowing a total of 2^k possible transformations rather than the $2^n!$ transformations available with the ideal block cipher.
- feistel proposed one is a practical application of a proposal by Claude Shannon to develop a product cipher that alternates confusion and diffusion functions.
- Shannons concern was to overcome cryptanalysis based on statistical analysis, because attack has knowledge of characteristics of language (i.e plaintext) ex: if human readable message, if frequency distribution of various letters are known, if these statistics reflect in the ciphertext Cryptanalyst will be able to deduce encryption key, Part of key at least set of key likely to contain exact key.
- Shannon refers strongly ideal cipher, all statistics of the ciphertext are independent of the particular key used.
- Two methods are. diffusion: the statistical structure of plaintext is dissipated into long range statistics of the ciphertext each plaintext digit affect the value

UNIT- II

Feistal cipher: Encryption

(iterated block cipher)

- plaintext and ciphertext consists of fixed-sized blocks
- ciphertext obtained from plaintext by iterating a round function
- Input to round function consists of key and the output of previous round.
- usually implemented in software

Encryption

feistel cipher: is a type of block cipher design, not a specific cipher

- split plaintext block into left and right halves: $P = (L_0, R_0)$

for each round $i = 1, 2, \ldots n$, compute

$$L_i = R_{i-1}$$

$$R_i = L_{i-1} \oplus f(R_{i-1}, K_i)$$

where f is a round function and K_i is subkey

ciphertext: $C = (L_n, R_n)$

Decryption

- Start with ciphertext $C = (L_n, R_n)$
- for each round $i = n, n-1, \ldots 1$, compute
- $R_{i-1} = L_i$

$L_{i-1} = R_i \oplus F(R_{i-1}, K_i)$ where f is round function & K_i is subkey

- Plaintext: $P = (L_0, R_0)$
- formula "works" for any function f
  - but only secure for certain function f.

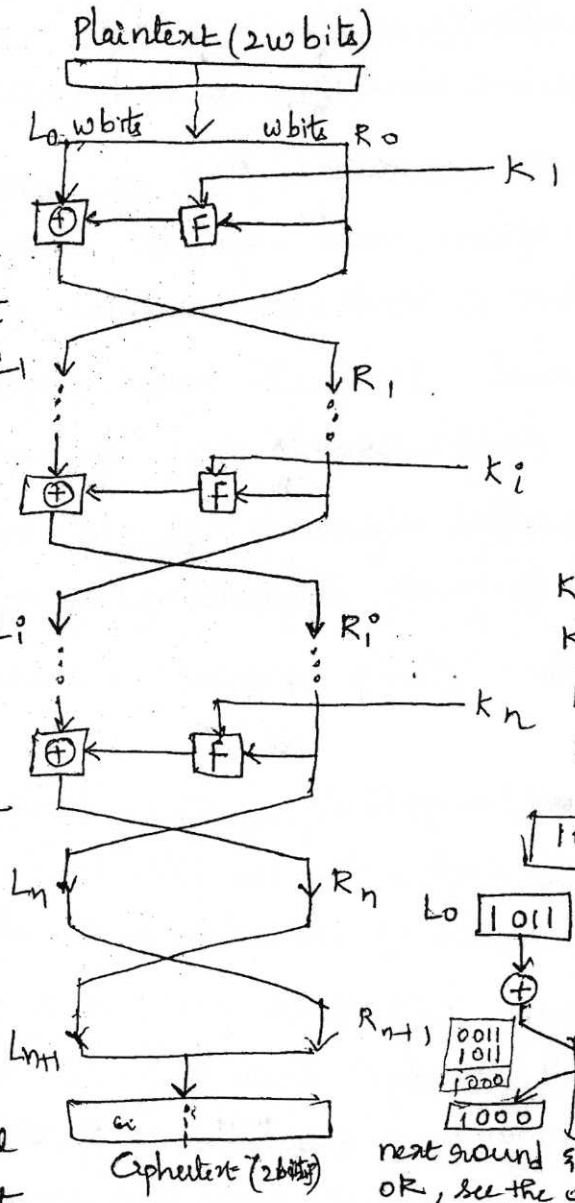
example of diffusion is to encrypt a message $M = m_1, m_2, m_3$ of characters with an averging operations.

$$Y_n = \sum_{i=1}^{k} m_{n+i} \pmod{26}$$

Confusion: Hides the relationship between plaintext, ciphertext and the key. ex: substitution

Diffusion: Spreads the effect on the key over ~~or~~ ciphertext as much as possible. i.e we generate subkeys (The influence of one each plaintext bit is spread over many ciphertexts bits)
ex: permutations

Feistel cipher structure



- The i/p to the encryption algorithm are a plaintext block of length 2w bits and a key K.
- The plaintext block is divided into two halves L_0 and R_0, two halves of the data pass through n rounds of processing & then combine to produce the ciphertext block
- Each round i has as inputs L_{i-1} and R_{i-1} derived from the previous round as well as subkey K_i derived from the overall K. (subkey K_i are different

Exclusive OR

1001 if S
0011 i/p
1010 ge

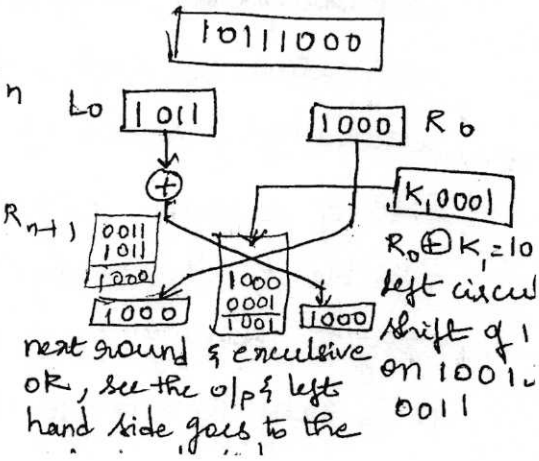
Left shift

1010 become
0101

Subkeys

6 bits of the key
$b_1\ b_2\ b_3\ b_4\ b_5\ b_6$

$K_1 = b_1 b_2 b_3 b_4$
$K_2 = b_3 b_4 b_5 b_6$
$K_3 = b_2 b_3 b_4 b_5$
$K_4 = b_1 b_2 b_5 b_6$



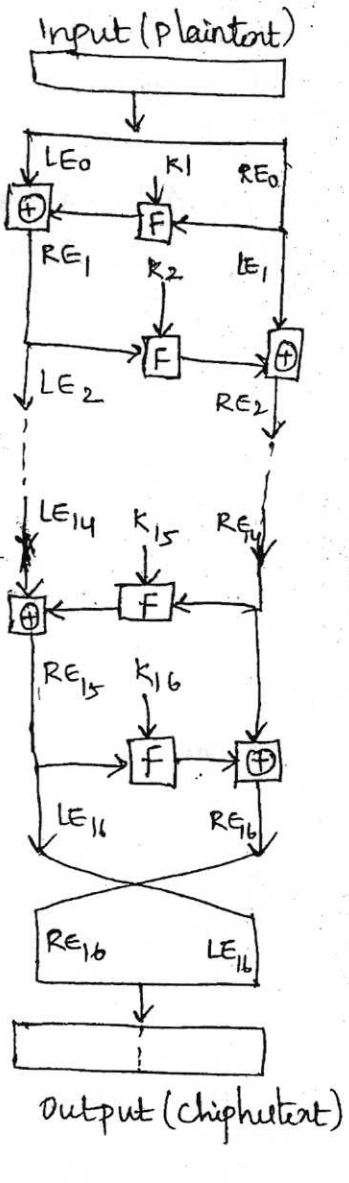
- substitution is performed on the left half of the data, this is done by applying a round function f to the right half of the data and then taking exclusive-OR of the output of that function and the left half of the data
- following substitution a permutation is performed that consists of the interchange of the two halves of the data

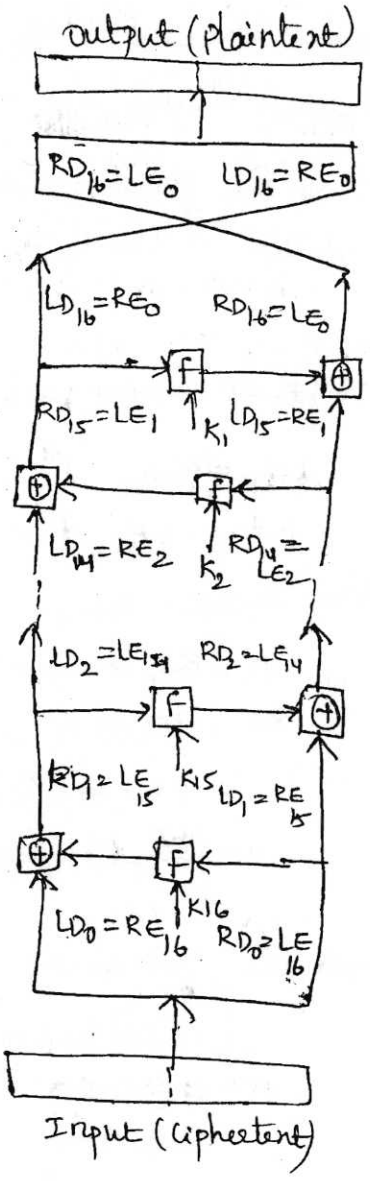
Parameters and design features of feistel network

- Block size: larger block size mean greater security. but reduced encryption/decryption speed for a given algorithm. (universal 64 bit block size has been considered a resonable) (greater diffusion)
- Key size: greater security is key size is large but decrease encryption/decryption speed. greater security is achieved by greater resistance to brute force attacks & greater confusion.
- Number of rounds: In feistel cipher single round offers inadequate security but multiple rounds offer increasing security (16 rounds)
- Subkey generation algorithm: greater complexity in this algorithm should lead to greater difficulty of cryptanalysis.
- Round function: greater complexity means greater resistance to cryptanalysis.
- Fast software encryption/decryption: encryption is embedded in applications in such a way to preclude a hardware implementation accordingly speed of execution of the algorithm becomes a concern.
- Ease of analysis: even if we make our algorithm as difficult as possible to cryptanalyst there is greater benifit in making algorithm easy to analyze i.e if algorithm is concisely & clearly explained it is easy to analyze the algorithm for cryptanalytic vulnerabilities

Decryption algorithm

- use the ciphertext as input to the algorithm, but use the subkey K_i in reverse order. (i.e use K_n in first round, K_{n-1} & so on) until K_1 is used in last round. (this is nice feature because it means we need not implement two different algorithms one for encryption and one for decryption).





- LE_i & RE_i for data traveling through the encryption algorithm and LD_i & RD_i for decryption algorithm.
- At each round the intermediate value of the decryption process is equal to the corresponding value of the encryption process with the

another way, let the output of the ith encryption round be $LE_i || RE_i$ then corresponding input to the $(16-i)$th decryption round is $RE_i || LE_i$ or equivalently $RD_{16-i} || LD_{16-i}$

- After last iteration of the encryption process, the two halves of the o/p are swapped so that ciphertext is $RE_{16} || LE_{16}$, the o/p of that round is ciphertext, now take this as input to the same algorithm
- the i/p to first round is $RE_{16} || LE_{16}$ (equal to 32-bit swap of the o/p of the sixteenth round of the encryption process)
- Now output of the first round of the decryption process is equal to a 32-bit swap of the i/p to the sixteenth round of the encryption process

Consider encryption process we see that

$$LE_{16} = RE_{15}$$
$$RE_{16} = LE_{15} \oplus F(RE_{15}, K_{16})$$

on decryption side

$$LD_1 = RD_0 = LE_{16} = RE_{15}$$
$$RD_1 = LD_0 \oplus f(RD_0, K_{16})$$
$$= RE_{16} \oplus f(RE_{15}, K_{16})$$
$$= [LE_{15} \oplus f(RE_{15}, K_{16})] \oplus F(RE_{15}, K_{16})$$

The XOR has the following properties

$$[A \oplus B] \oplus C = A \oplus [B \oplus C]$$
$$D \oplus D = 0$$
$$E \oplus 0 = E$$

thus we have $LD_1 = RE_{15}$ & $RD_1 = LE_{15}$

$\therefore$ o/p of first round of decryption is $LE_{15} || RE_{15}$ (32-bit swap of the i/p to the sixteenth round of the encryption)

In general terms for ith iteration of the encryption algorithm

$$LE_i = RE_{i-1}$$

Rearranging terms

$$RE_{i-1} = LE_i$$

$$LE_{i-1} = RE_i \oplus F(RE_{i-1}, K_i) = RE_i \oplus F(LE_i, K_i)$$

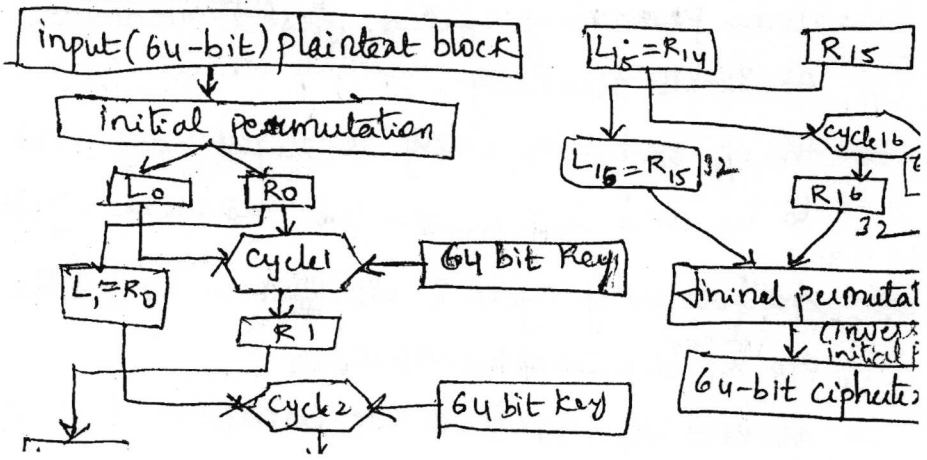
- Thus inputs to the ith iteration as a function of the output and these equations confirm the assignments shown in right hand side
- finally output of the last round of decryption process is $RE_0 \| LE_0$. a 32 bit swap recovers the original plaintext

Data encryption standard

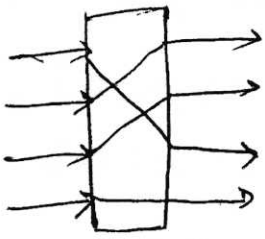
- DES algorithm was developed by IBM based on the lucifer algorithm it has been using before.
- DES is a careful and complex combination of two fundamental building blocks of encryption substution & transposition.
- The algorithm derives the strength from repeated application of these two techniques (16 cycles) one on top of the other.

Product ciphes: Two complementary ciphers can be made more secure by being applied together alternatively

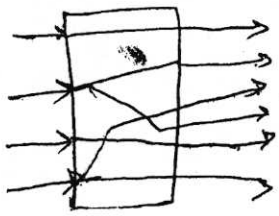
Cycle of substitution and permutation



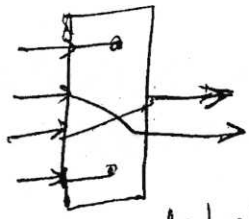
Types of permutations



Permutation



Expansion Permutation



Permuted choice

Bit	Goes to position							
1-8	40	8	48	16	56	24	64	32
9-16	39	7	47	15	55	23	63	31
17-24	38	6	46	14	54	22	62	30
25-32	37	5	45	13	53	21	61	29
33-40	36	4	44	12	52	20	60	28
41-48	35	3	43	11	51	19	59	27
49-56	34	2	42	10	50	18	58	26
57-64	33	1	41	9	49	17	57	25

initial permutation

bit in position 1 move to 40, 2 to 8, 3 to 16, ---8 to 32

- After the initial permutation we break the block into two halves L_0 & R_0 and both are input to cycle. (L_0, R_0 & 64 bit Key)
- Key even though it is a single key, it is processed in different way (we manipulate the key and i/p it into each cycle).
- for paticular cycle we fed in the key, L_0 & R_0 we go through some processing called feistel network processing we get output as 32-bit quantity.
- the o/p 32-bit quantity is right half (R_1) and right half (R_0) is equal to L_1 and L_1 & R_1 are fed into next cycle as well as key and 64 bit key
- so for any cycle i we fed in L_{i-1} & R_{i-1} and key K_i we get o/p R_i (i.e 32-bit output).
- At end of cycle 16 we combile L_{16} & R_{16} each of 32 bit so put

final permutation is according to the table

Bit	Goes to position							
1-8	58	50	42	34	26	18	10	2
9-16	60	52	44	36	28	20	12	4
17-24	62	54	46	38	30	22	14	6
25-32	64	56	48	40	32	24	16	8
33-40	57	49	41	33	25	17	9	1
41-48	59	51	43	35	27	19	11	3
49-56	61	53	45	37	29	21	13	5
57-64	63	55	47	39	39	23	15	7

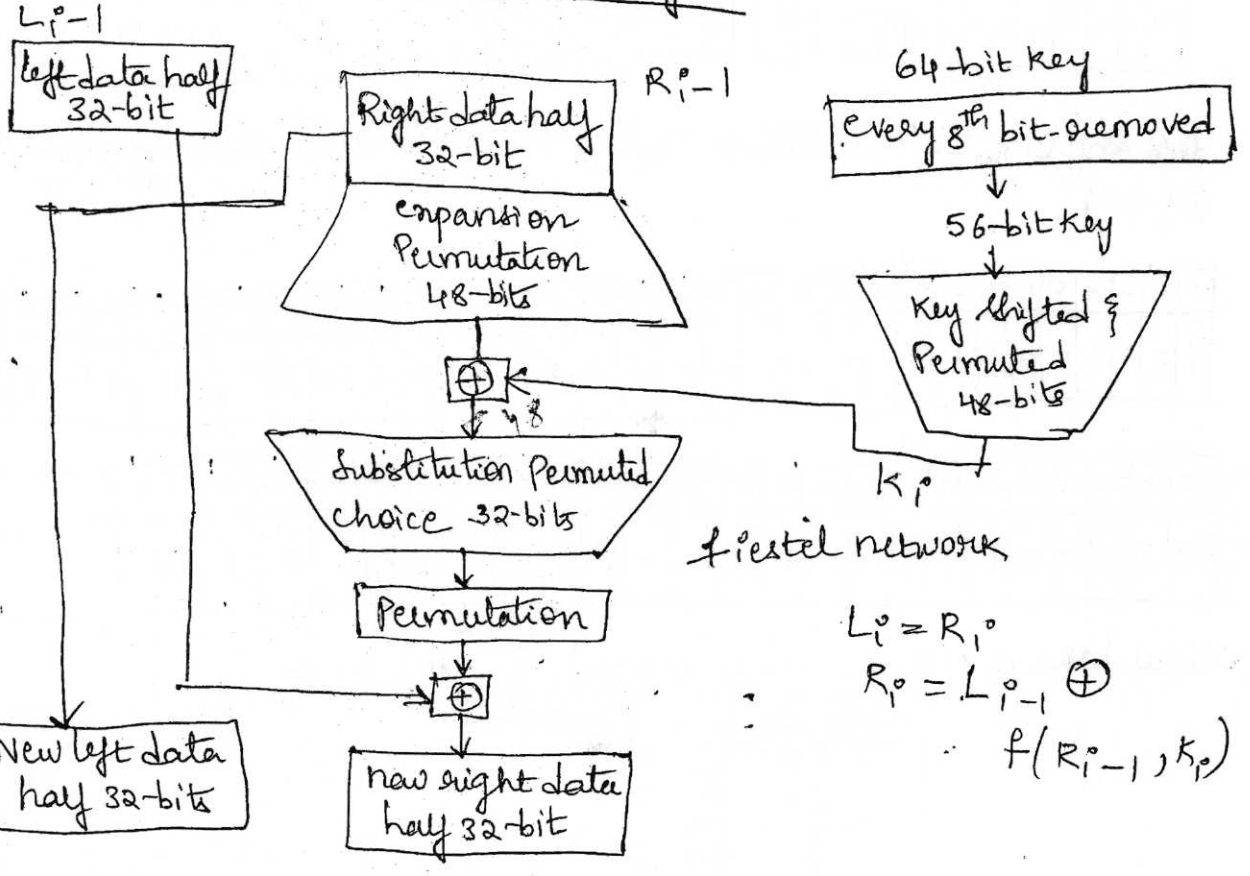
finial permutation.

(in initial permutatio

the output after finial permutation is 64 bit key ! bit is at position so in inverse Permutation, 58 bit replaced to 1 positio

Now what happens inside the cycle

Details of a cycle



feistel network

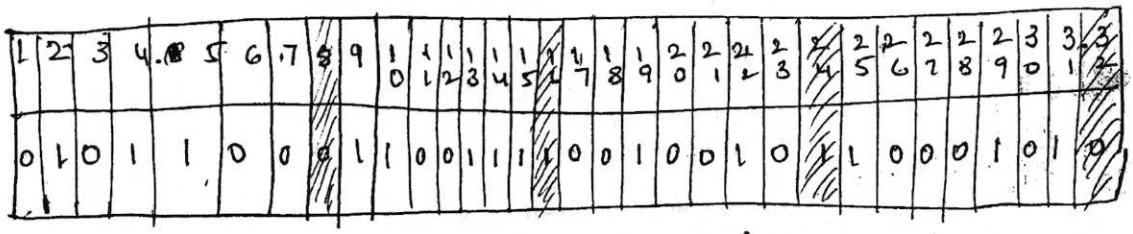
$L_i^\circ = R_{i-1}^\circ$

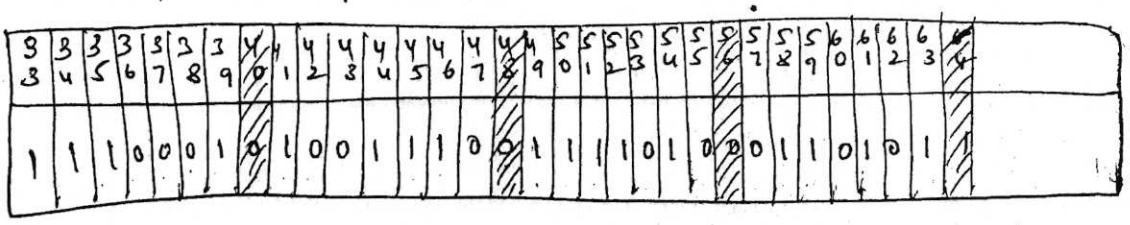
$R_i^\circ = L_{i-1}^\circ \oplus f(R_{i-1}^\circ, K_i^\circ)$

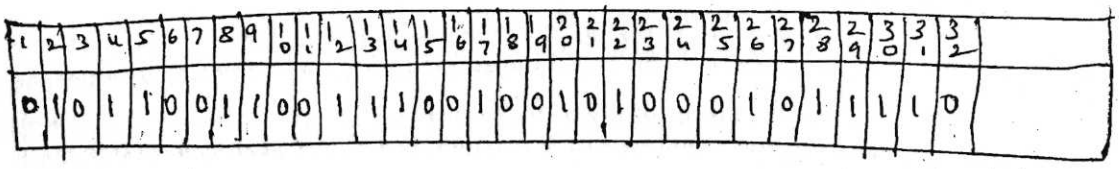
first we remove every 8 bit

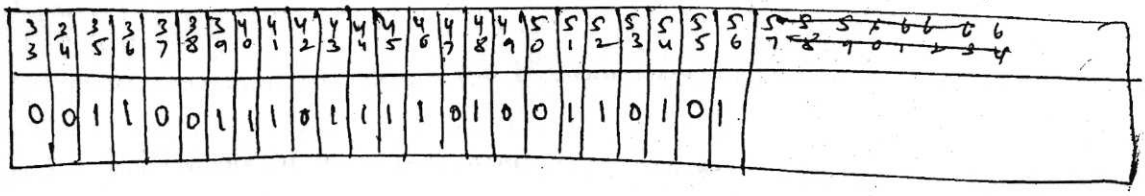
Key shift and permutation (cycle 1)

After every 8th bit removed and split into two 28-bit halves

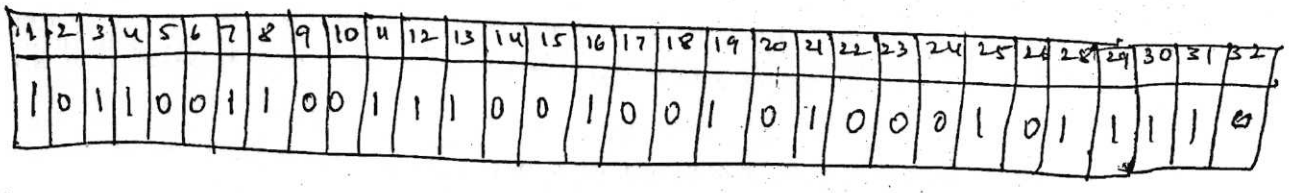


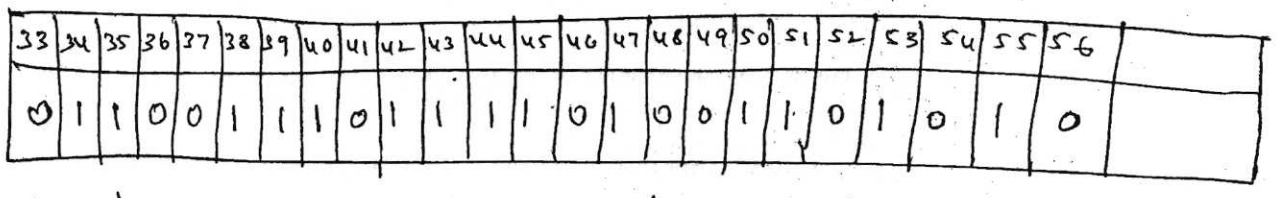






left shifting the two 28 bits halves by 1 bit and putting them together (cycle 1)





How many bits we shift depends on the cycle

Cycle	bits shifted
1	1
2	1
3	2
4	2
5	2
6	2
7	2
8	2

	bits shifted
12	2
13	2
14	2
15	2
16	1

Once shifted goes to permuted choice (kind of permutation where 64 bits are reduced to 48 bits)

Key bit	1	2	3	4	5	6	7	8	9	10	11	12	13	14
Selected for Position	5	24	7	16	6	10	20	18	–	12	3	15	23	1
Key bit	15	16	17	18	19	20	21	22	23	24	25	26	27	28
SFP	9	19	2	–	14	22	11	–	13	4	–	17	21	8
Key bit	29	30	31	32	33	34	35	36	37	38	39	40	41	42
SFP	47	31	27	48	35	41	–	46	28	–	30	32	25	44
Key bit	43	44	45	46	47	48	49	50	51	52	53	54	55	56
SFP	–	37	34	43	29	36	38	45	33	26	42	–	30	40

56 bits to 48 bits

32	1	2	3	4	5
4	5			8	9
8	9			12	13
12	13			16	17
16	17			20	21
20	21			24	25
24	25			28	29
28	29			32	1

– the output of the cycle is 48 bits, Now there are two halves left right (32 bit), we take right half first and feistel network (1 cycle), Input for feistel network is R_{i-1} and K_i and, so what happens is 32 bit right half goes into expansion permut 48 bits

Bit	1	2	3	4	5	6	7	8
moves to position	2,48	3	4	5,7	6,8	9	10	11,13
Bit	9	10	11	12	13	14	15	16
MTP	12,14	15	16	17,19	18,20	21	22	23,25
Bit	17	18	19	20	21	22	23	24
MTP	24,26	27	28	29,31	30,32	33	34	35,37
Bit	25	26	27	28	29	30	31	32
MTP	36,38	39	40	41,43	42,44	45	46	47,1

Expansion permutation 32 bits to 48 bits

After expansion we XOR 48 bit expansion permutation with 48 bits key, simple bit by bit process after this output is 48 bits, Now these 48 bits goes under substitution permuted choice (32 bits)

for substitution we need sBoxes

example

48 bit input

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23
0	1	1	0	0	0	0	1	0	0	1	0	1	1	0	1	0	1	0	1	1	0	1

25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40	41	42	43	44	45	46	47	4
0	0	1	1	1	0	1	1	0	1	0	0	1	1	0	1	0	0	1	1	1	0	0	0

Substitution with permutation

We divide into six bits (Stinson's) (divide into 6bit i.e 6×8 box 48 bits)

	6-bit input	Row value	Column value	s-box result	4 bit output
S-box s1	011101	1(01)	14(1110)	3	0011
S-box s2	010010	0(00)	9(1001)	7	0111
S box s3	110101	3(11)	10(1010)	14	1110
s-box s4	011011	1.(01)	13(1101)	10	1010
s-box s5	001110	0(00)	7 (0111)	6	0110
s-box s6	110100	2(10)	10(1010)	4	0100
s-box s7	110100	2(10)	10(1010)	6	0110
S-box s8	111000	2(10)	12(1100)	15	1111

Row value — is selected from 6-bit put first position & last position bit i.e 01 and the binary value of bits taken are written outside bracket i.e 1(01) and remaining bits in 6bit input after ~~remain~~ removing first & last bit is 1110 the binary value of this is written in column value along with that bits 14(1110)

— The value row value 1 and column value 14 is seen in s-box

we group all 4-bit output

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
0	0	0	1	0	0	1	1	1	1	1	1	0	1	0	1	0

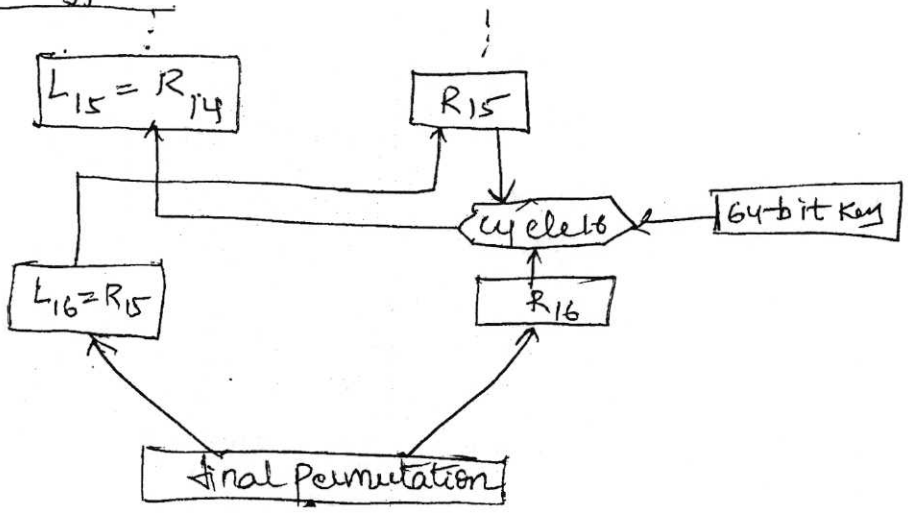
17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32
0	1	1	0	0	1	0	0	0	1	0	0	1	1	1	1

what we get output of S box is 32 bit quantity and that goes to permutation

Bit	Goes to position							
1-8	9	17	23	31	13	28	2	18
9-16	24	16	30	6	26	20	10	1
17-24	8	14	25	3	4	29	11	19
25-32	32	12	22	7	5	27	15	21

- After this X-OR with left half of 32 bit which comes out as output of feistel network and now that is your new right half of 32 bit (R_i) (i.e $R_i = L_{i-1} \oplus F(R_{i-1}, K_i)$ and L_i is simply comes from R_{i-1} (32 bits)

Des decryption

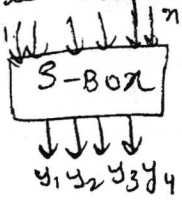


It undergoes permutation ($L_{16}=R_{15}$) same as encryption except that the application of the subkeys is reversed. (o/p 64-bit plaintext)

$$R_{i-1} = L_i$$
$$L_{i-1} = R_i \oplus f(R_{i-1} = L_i, K_i)$$

Properties of S-Box

- rows are permutations
- inputs are non linear combination of the inputs outputs
- change of one bit of the input and half of the output bits change (avalanche effect)
- each o/p bit is dependent on all the input bits



$$y_1 = f_1(x_1, \ldots x_6) \; ; \; y_2 = f_2(x_1, \ldots x_6) \ldots$$

each of o/p is non-linear

not only that y_1, y_2 so on are non-linear but also when you x-or y_1, y_2 so on then also should be non-linear

P-box

The strength of DES

Use of 56-bit Keys : with the use of key length 56 bits there are 2^{56} possible keys (approx: 7.2×10^{16} keys), thus brute force appears impossible

- To search key space one single machine performing one DES encryption per microsecond would take more than thousand years.
- After 1977 diffie and Hellman says that the technology existed to build a parallel machine with 1 million encryption devices each of which could perform one encryption per microsecond this brought the average search time down to about 10 hrs and cost would be $20 million in 1977
- finally DES proved insecure in July 1998 when electronic frontier foundation (EFF) announced that it had broken a DES encryption using special purpose "DES cracker" machine, attacker took less than 3 days. EFF announced a detailed description of the machine enabling other to build their own cracker.
- Week keys, semi week keys and known plaintext attack

Week Keys

- is one which after parity drop operation, consists either of all 0's all 1's or half 0's & half 1's
- four out of the 2^{56} keys are week keys.

Example

Keys before parity drop (64-bits)	Actual Keys (56 bits)
0101 0101 0101 0101	0000000 0000000
1F1F 1F1F 0E0E 0E0E	0000000 FFFFFFF
E0E0 E0E0 F1F1 F1F1	FFFFFFF 0000000
FEFE FEFE FEFE FEFE	FFFFFFF FFFFFFF

- Shift & Permutation in Key scheduling algorithm

Consequences of weak keys

- Round keys created from any of these week keys are the same
- If we encrypt a block with a weak key and subsequently encrypt the result with the same weak key we get the original block (why bcz of pervious result) (inverse alg gives same key) (i.e out of 2^{56} possibilites you are left with 2^4 possibilites)

Semi weak keys $(E_{K_2}(E_{K_1}(P)) = P$

- A semi weak key creates only two different round keys and each of them is repeated eight times
- There are six key pairs that are called semi weak keys
- The round keys created from each pair are the same in different order

Multiple DES

- major criticism against DES is the key length, so we multiple time i.e $2^{56} + 2^{56} = 2^{11^2}$ key

- Avalanche effect

Aim: small change in key or plaintext produces large change in ciphertext

- It is present in DES (good security)

~~ex: P.T — security (1 char 1 byte) so 8 bytes~~

$f(x)$ $\quad f(x \oplus \alpha)$

$|x| = 64$ bits

$x = \langle x_1, x_2, x_3 \rangle$

$\alpha = (0, 1, 0 >$

$x \oplus \alpha = \langle x_1, x_2 \oplus 1, x_3 \rangle$

$= \langle x_1, \bar{x}_2, x_3 \rangle$

$f(x) \oplus f(x \oplus \alpha)$ should be balanced function where $wt(\alpha) = 1$

Nature of DES algorithm

- Cryptanalysis is possible by exploiting the characteristics of the DES alg. focus is on the eight s-boxes that are used in each iteration. because the design criteria for these boxes & indeed for the entire algorithm were not made public
- There is a suspicion that cryptanalysis is possible for an opponent who knowns the weakness of the s-boxes, many unexpected behaviors of the S-Boxes have been discovered but their is ~~on~~ no one so far succeded in discovering the fatal weakness in the S-Boxes.

Timing attacks

- is one in which information about the key or the plaintext is obtained by observing how long it takes a given implementation to perform decryption on various ciphertext
- exploits the fact that an encryption/decryption algorithm often takes slightly different amounts of time on different inputs eg and calculating hamming weight (number of bits equal to one) of the secret key. (but so far that this technique will never be successful against DES)

Differential and Linear cryptanalysis

AES (Advanced encryption standard):

finite fields

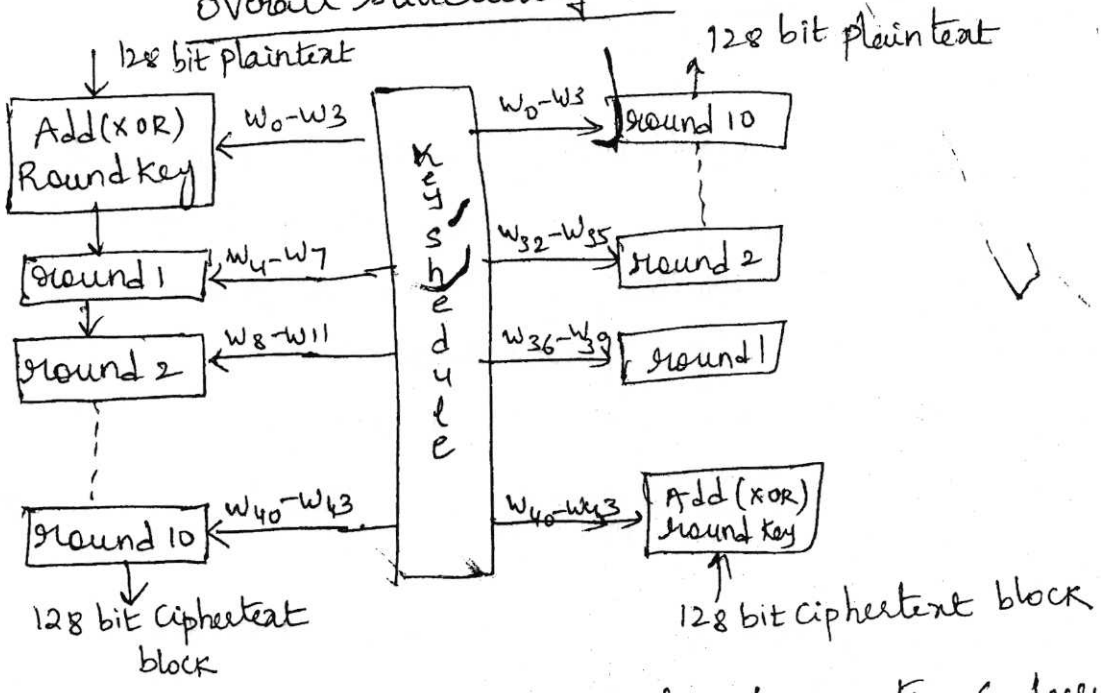
- A finite field is a field with a finite number of elements
- The no of elements in a set is called the order of the field
- A field with order m exists iff m is a prime power i.e, $m = p^n$ for some integer n and with p a prime ~~numb~~ integer
- p is called the characteristics of the finite field
- GF(p) (galva field) : the elements of the fields can be represente as by 0, 1, . . . -p-1
- if p is not prime then multiplications are not defined
- However for finite fields $GF(p^n)$ with $n > 1$, slightly complex represented as polynomials over GF(p).

Polynomial over a field

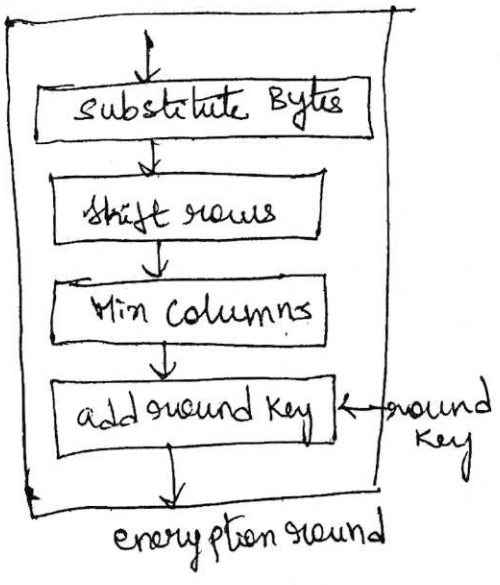
- is an expression of the form $b(x) = b_{m-1}x^{n-1} + b_{m-2}^{n-2} + \cdots + b_0$ x being called ~~poter~~ indeterminate of the polynomial & the $b_i \in F$ the coefficients.
- The degree of a polynomial equals l if $b_j = 0 \; \forall j > l$ and l is the smallest number with this property.
- set of polynomials over a field f is denoted by f(x)
  set of polynomials over a field f which has a degree less than l is denoted by $f[x]_l$

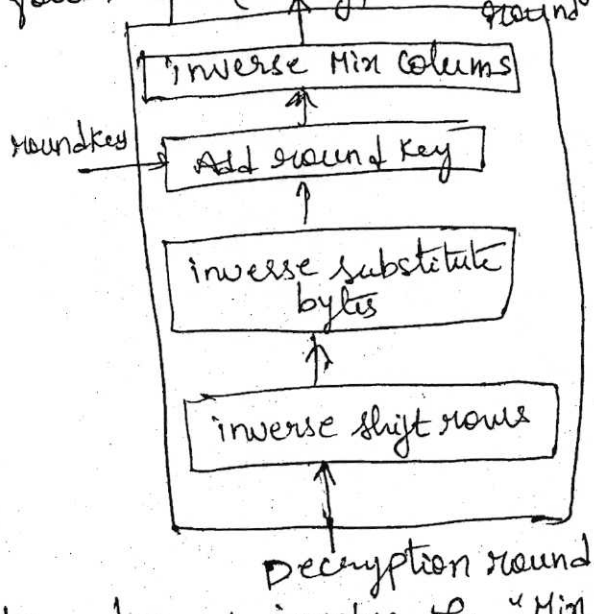
4 (Pg 141) - 1

Overall structure of AES



In each round we have four steps (encryption & decryption)





* : The last round of encryption does not involve the "Mix Columns" step. the last round of decryption does not involve the "inverse mix columns" step.

- last round means round 10 for encryption & round 10 for decryption
- for each round we i/p a state array and get o/p a state array.
- AES input block is 128 bit state array broken down into 4 × 4 matrix and each entry in matrix is byte (so 16 bytes)

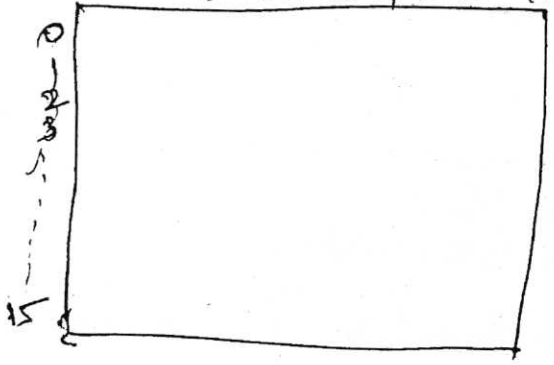
A word consists of 4 bytes (32 bits) each column of the state array is a word. referred to as the state array for each round

byte 0	byte 4	byte 8	byte 12
byte 1	byte 5	byte 9	byte 13
byte 2	byte 6	byte 10	byte 14
byte 3	byte 7	byte 11	byte 15

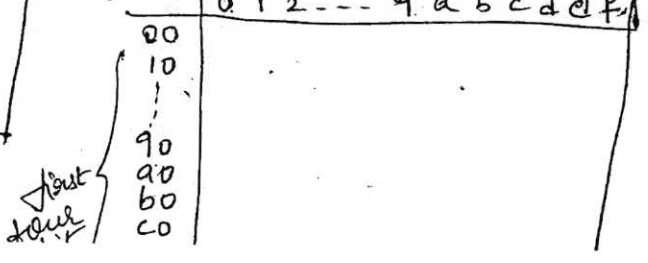
Substitute byte step

- This is a byte-to-byte substitution step using a 16X16 lookup table (whose entry values range from 0 to 255 a byte each)
- same lookup table is used for each byte in all the rounds
  - one lookup table for subbyte: expansencryption
  - A different (but related) lookup table for invsubbytes: decryption
- The substitution lookup tables are developed based on bit scrambling (a kind of randomization) to reduce the correlation between the input bits and the output bits at the byte level
- To find the substitution for an input byte we break the byte into two four bit units (nibble) use the first 4-bit nibble as the row index and the second 4-bit nibble as the column index.

Subbytes lookup table (0-255)



we break the bits into 8 bits quantity first four bits as row and next four bytes as column in lookup



eg: Suppose we want to replace 41 then see 4 in the row i.e 40 and one in the column 1 the intersection value is 83
Same way 9c then 9 in row 90 & c in column i.e de

- Same technique for inverse lookup table for decryption

Shift row step

Shift rows transformation (for encryption)

- the first row is NOT shifted
- the second row is shifted one byte to the left
- the third row is shifted two bytes to the left
- the fourth row is shifted three bytes to the left.

Scrambling: As the bytes of the state array are filled column wise, shifting the rows in the manner indicated above scrambles the byte order of the state array and promotes diffusion.

$$\begin{bmatrix} s_{0,0} & s_{0,1} & s_{0,2} & s_{0,3} \\ s_{1,0} & s_{1,1} & s_{1,2} & s_{1,3} \\ s_{2,0} & s_{2,1} & s_{2,2} & s_{2,3} \\ s_{3,0} & s_{3,1} & s_{3,2} & s_{3,3} \end{bmatrix} \Longrightarrow \begin{bmatrix} s_{0,0} & s_{0,1} & s_{0,2} & s_{0,3} \\ s_{1,1} & s_{1,2} & s_{1,3} & s_{1,0} \\ s_{2,2} & s_{2,3} & s_{2,0} & s_{2,1} \\ s_{3,3} & s_{3,0} & s_{3,1} & s_{3,2} \end{bmatrix}$$

Inverse shift rows step

- Inverse shift rows transformation:
- first row is not shifted
- second row is shifted one byte to the right
- third row is shifted two bytes to the right
- the fourth row is shifted three bytes to the right

$$\begin{bmatrix} s_{0,0} & s_{0,1} & s_{0,2} & s_{0,3} \\ s_{1,0} & s_{1,1} & s_{1,2} & s_{1,3} \\ s_{2,0} & s_{2,1} & s_{2,2} & s_{2,3} \end{bmatrix} \Longrightarrow \begin{bmatrix} s_{0,0} & s_{0,1} & s_{0,2} & s_{0,3} \\ s_{1,3} & s_{1,0} & s_{1,1} & s_{1,2} \\ s_{2,2} & s_{2,3} & s_{2,0} & s_{2,1} \end{bmatrix}$$

Mix Columns

- This step replaces each byte of a column by a function of all the bytes in the same column
- All multiplications are according to the GF(2^8) arithmetic and all additions are XOR operations.
- for encryption, the state matrix is multiplied with the following matrix.

$$\text{Mix Columns}\begin{bmatrix} 02 & 03 & 01 & 01 \\ 01 & 02 & 03 & 01 \\ 01 & 01 & 02 & 03 \\ 03 & 01 & 01 & 02 \end{bmatrix} \times \begin{bmatrix} s_{0,0} & s_{0,1} & s_{0,2} & s_{0,3} \\ s_{1,0} & s_{1,1} & s_{1,2} & s_{1,3} \\ s_{2,0} & s_{2,1} & s_{2,2} & s_{2,3} \\ s_{3,0} & s_{3,1} & s_{3,2} & s_{3,3} \end{bmatrix} = \begin{bmatrix} s'_{0,0} & s'_{0,1} & s'_{0,2} & s'_{0,3} \\ s'_{1,0} & s'_{1,1} & s'_{1,2} & s'_{1,3} \\ s'_{2,0} & s'_{2,1} & s'_{2,2} & s'_{1,2} \\ s'_{3,0} & s'_{3,1} & s'_{3,2} & s'_{1,3} \end{bmatrix}$$

State matrix

for decryption the state matrix is multiplied with the foll matrix

$$\begin{bmatrix} 0E & 0B & 0D & 09 \\ 09 & 0E & 0B & 0D \\ 0D & 09 & 0E & 0B \\ 0B & 0D & 09 & 0E \end{bmatrix} \times \begin{bmatrix} s_{0,0} & s_{0,1} & s_{0,2} & s_{0,3} \\ s_{1,0} & s_{1,1} & s_{1,2} & s_{1,3} \\ s_{2,0} & s_{2,1} & s_{2,3} & s_{2,3} \\ s_{3,0} & s_{3,1} & s_{3,4} & s_{3,3} \end{bmatrix} = \begin{bmatrix} s'_{0,0} & s'_{0,1} & s'_{0,2} & s'_{0,3} \\ s'_{1,0} & s'_{1,1} & s'_{1,2} & s'_{1,3} \\ s'_{2,0} & s'_{2,1} & s'_{2,2} & s'_{2,3} \\ s'_{3,0} & s'_{3,1} & s'_{3,2} & s'_{3,3} \end{bmatrix}$$

finite field arithmetic

- also called Galois field arithmetic
- AES uses GF(2^8) arithmetic: all values are in range of 0-255
- We write all values in hex: a byte is written as two hexadecimal values.
- A binary string is represented as a polynomial
  - $00110110 : x^5 + x^4 + x^2 + x$
  - $10010011 : x^7 + x^4 + x + 1$

Addition (XOR): example

$36 + 93 = 00100110 + 10010011$

$= 1$

> ★ Note: $1+1=0$ hence $x^r + x^r = 0$ for any exponent r

$= (x^5 + \cancel{x^4} + x^2 + \cancel{x} + x^7 + \cancel{x^4} + \cancel{x} + 1)$

$= x^5 + x^2 + x^7 + 1 = x^7 + x^5 + x^2 + 1$

$= 10100101 = a5$

finite field ~~ma~~thematic multiplication

~~(36)×(93)~~ (36) (93) = (0011 0110) (10010011)

$= (x^5 + x^4 + x^2 + x)(x^7 + x^4 + x + 1)$

$= x^{12} + \cancel{x^9} + \cancel{x^6} + \cancel{x^5} + x^{11} + \cancel{x^8} + \cancel{x^5} + x^4 + \cancel{x^9} + \cancel{x^6} + x^3 + \cancel{x^2}$
$+ \cancel{x^8} + x^5 + \cancel{x^2} + x$

$= x^{12} + x^{11} + x^5 + x^4 + x^3 + x = 1100000111010$

If the degree of the resulting polynomial exceeds 7 we need to do an XOR division with the GF(2^8) reducing polynomial

$x^8 + x^4 + x^3 + x + 1 = 100011011$

```
                 1100000111010
100011011 ) ──────────────────
             100011011↓
             ─────────
              100110001
              100011011
             ──────────
              φ  10101010
                 10001101
                ─────────
                  1001001  = 0100 1001
                                4    9
```

7bits put 0 in 1st position to 8bits

Prefix the remainder with sufficient 0's to make it 8bits long

(36) (93) = 49

C-12

ex: (53)(ca) = (01010011) (11001010)

$= (x^6 + x^4 + x + 1)(x^7 + x^6 + x^3 + x)$

$= x^{13} + x^{12} + x^9 + \cancel{x^7} + x^{11} + x^{10} + \cancel{x^7} + x^5 + x^8 + \cancel{x^7} + x^4 + x^2 +$
$\cancel{x^7} + x^6 + x^3 + x$

$= x^{13} + x^{12} + x^{11} + x^{10} + x^9 + x^8 + x^6 + x^5 + x^4 + x^3 + x^2 + x$

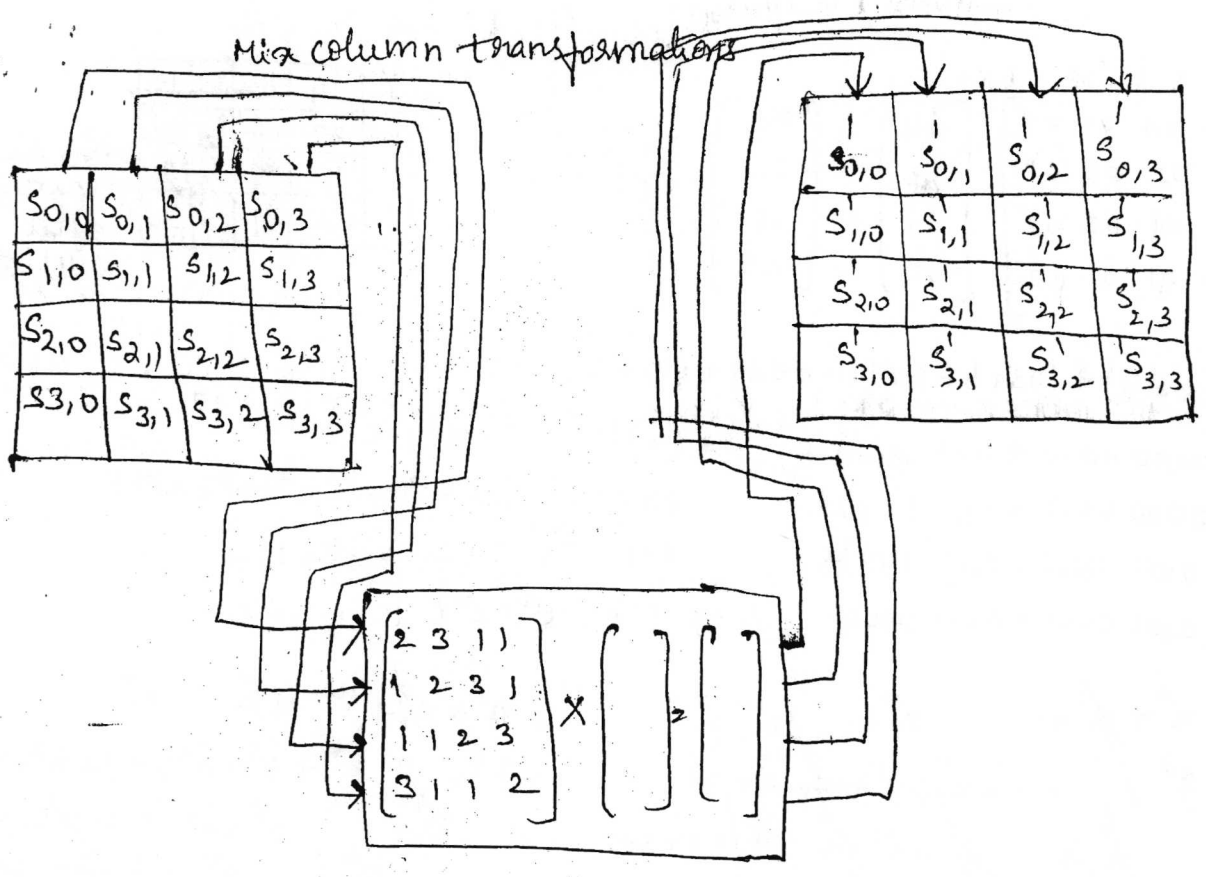
$= 11111101111110$

404

divide : $x^8 + x^4 + x^3 + x + 1$

[illegible]

Mix column transformations



$$\begin{bmatrix} 02 & 03 & 01 & 01 \\ 01 & 02 & 03 & 01 \\ 01 & 01 & 02 & 03 \\ 03 & 01 & 01 & 02 \end{bmatrix} \begin{bmatrix} d4 \\ bf \\ 5d \\ 30 \end{bmatrix} = \begin{bmatrix} 04 \\ 66 \\ 01 \\ e5 \end{bmatrix}$$

d4	e0	b8	1e
27	bf	b4	41
11	98	5d	52
ae	f1	e5	30

this the matrix after sub bytes

$(02 * d4) + (03 * bf) + (01 * 5d) + (01 * 30)$

$= (00000010 * 1101\ 0100) +$
$(00000011 * 1011\ 1111) +$
$(00000001 * 0101\ 1101) +$
$(00000001 * 0011\ 0000)$

$= (x)(x^7 + x^6 + x^4 + x^2) +$
$(x+1)(x^7 + x^5 + x^4 + x^3 + x^2 + x + 1) +$
$(1)(x^6 + x^4 + x^3 + x^2 + 1) +$
$(1)(x^5 + x^4)$

$= x^8 + x^7 + x^6 + x^5 + x^4 + x^3 + x^2 + x^8 + x^6 + x^5 + x^4 + x^3 + x^2 + x + x^7 + x^5 + x^4 + x^3 + x^2 + x + x^6 + x^4 + x^3 + x^2 + 1 + x^5 + x^4$

$= x^8 + x^7 + x^5 + x^4 + x^8 + x^6 + x^5 + x^4 + x^3 + x^2 + x + 1 + x^7 + x^5 + x^4 + x^3 + x^2 + x + 1 + x^6 + x^4 + x^3 + x^2 + 1 + x^5 + x^4$

$= x^2$

$= 00000100$

$= 04$

for second 66

After doing as above we get polynomial expression

$= x^8 + x^6 + x^5 + x^4 + x^3 + x^2 + 1 = 101111101$

exceeds above 7 so reduce using GF(2^8) polynomial

$x^6 + x^4 + x^3 + x + 1 = 100011011$

$$\begin{array}{r|l} 100011011 & 101111101 \\ & 100011011 \\ \hline & 1100110 \end{array}$$

Prefix with sufficient zeros to make the remainder an 8-bit quantity $01100110 = 66$

— Same with inverse mix columns but matrix is

$$\begin{bmatrix} 0E & 0B & 0D & 09 \\ 09 & 0E & 0B & 0D \end{bmatrix}$$

the encryption & decryption matrices are constan ... these two matrices are

Mix columns

$$\begin{bmatrix} ax+by+cz+dt \\ ex+fy+kz+ht \\ ix+jy+kz+lt \\ mx+ny+oz+pt \end{bmatrix} = \begin{bmatrix} a & b & c & d \\ e & f & g & h \\ i & j & k & l \\ m & n & o & p \end{bmatrix} \times \begin{bmatrix} x \\ y \\ z \\ t \end{bmatrix}$$

new matrix Constant old matrix

- Subbytes transformation change the value of the byte based only on original value and an entry in the table, the process does not include the neighboring bytes i.e subbytes is an intrabyte transformat
- The permutation provided by shiftrows transformation exchanges bytes without permuting the bits inside the bytes i.e we can say that shiftrows is a byte exchange transformation
- We need interbyte transformation that changes the bits inside the neighboring bytes (we need to mix bytes to provide diffusion at the bit level)
- Mixing transformation changes the contents of each byte by taking four bytes at a time and combining them to recreate four new bytes.

  Combination process first multiples each byte with a diff constant and then mixes them.

  Mixing provided by matrix multiplication (when we multiply a square matrix by a column matrix the result is a new matrix each elt in new matrix depends on all the four elts of the old matrix after they are multiplied by row values in the constant matrix.

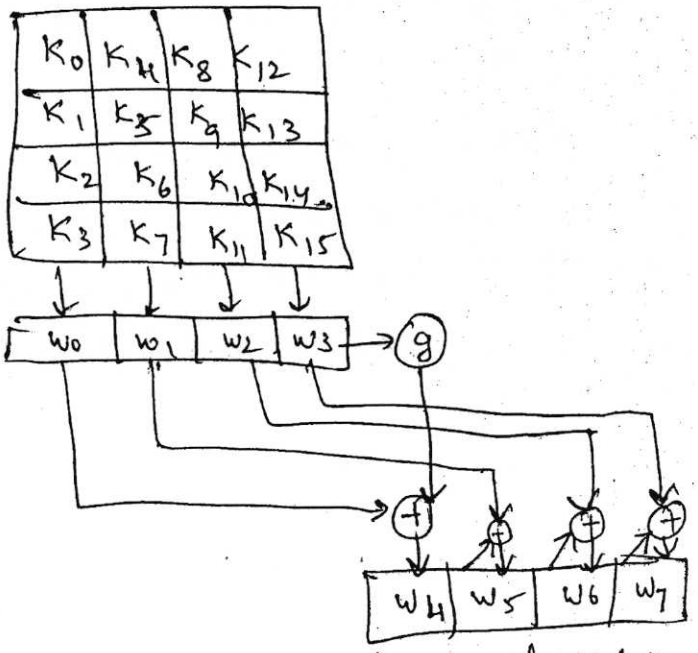
Add round Key Transformation

128 bits of state are bitwise XORed with the 128 bits of the round key operation is viewed as a columnwise operation between the 4 bytes of a state column and one word of the round key

Key expansion

Takes as input a 4 word (16 byte) Key and produces a linear array of 44 words (176 bytes) this is sufficient to provide a 4-word round key for the initial addroundkey stage and each of 10 rounds of the cipher

- Key is copied into first four words of the expanded key remainder of the expanded key is filled in four words at a time w[i] depends on immediately preceding word, w[i-1] and word four positions back, w[i-4]
- The word whose position in the w array is a multiple of 4, a more complex function is used (fun g consists of foll subfunctions)



1. rot word performs a one byte circular left shift on a word

2. Subword performs a byte substitution on each byte of its input word using S-box

3. The result of step 1 & 2 is XORed with a round constant R.con[j]

Round Constant is a word in which the three rightmost bytes are always 0, thus effect of an XOR of a word with Rcon is to only perform an XOR on the leftmost byte of the word

- Rcon is different for each round and defined as

Rcon[j] = (RC[j], 0, 0, 0), with RC[1] = 1, RC[j] = 2 · RC[j-1] & with multiplication difined over the field GF(2^8). Values of RC[j] is hexadecimal

j	1	2	3	4	5	6	7	8	9	10
RC[j]	01	02	04	08	10	20	40	80	1B	36

ex: Round key for round 8

EA D2 73 21 B5 8D BA D2 31 2B F5 60 7F 8D 29 2F

first 4 bytes of round key for round 9

i(decimal)	temp	After rot Word	After subword	RCon(9)	After XOR with Rcon	w[i-4]	w[i] = temp ⊕ w[i-4]
36	7F8D292F	8D292F7F	5DA515D2	1B000000	46A515D2	EAD27321	AC7766F3

w_{i-4}			w_{i-1}	w_i
2b	28	ab	09	
7e	ae	f7	cf	
15	d2	15	4f	
16	a6	88	3c	

Apply S-box for this 4 bytes we get

8a
84
eb

DO XOR with ~~let us say $x^i/3$ of 2^8 so we~~ get the predefined set Rcon we get

01	02	04	08	10	20	40	80	1b	36
00	00	00	00	00	00	00	00	00	00
00	00	00	00	00	00	00	00	00	00
00	00	00	00	00	00	00	00	00	00

2b		8a		01		a0
7e	⊕	84	⊕	00	=	fa
15		eb		00		fe
16		01		9		17

Rcon(4)

- Filling other locations which are not multiple of 4, XOR the 4th previous column with the i−1th column so look w_{i-1} and w_{i-4}

My fill 6th col XOR 5th col with 2nd col

	w_{i-4}			w_{i-1}	w_i
2b	28	ab	09	a0	
7e	ae	f7	cf	fa	
15	d2	15	4f	fe	
16	a6	88	3c	17	

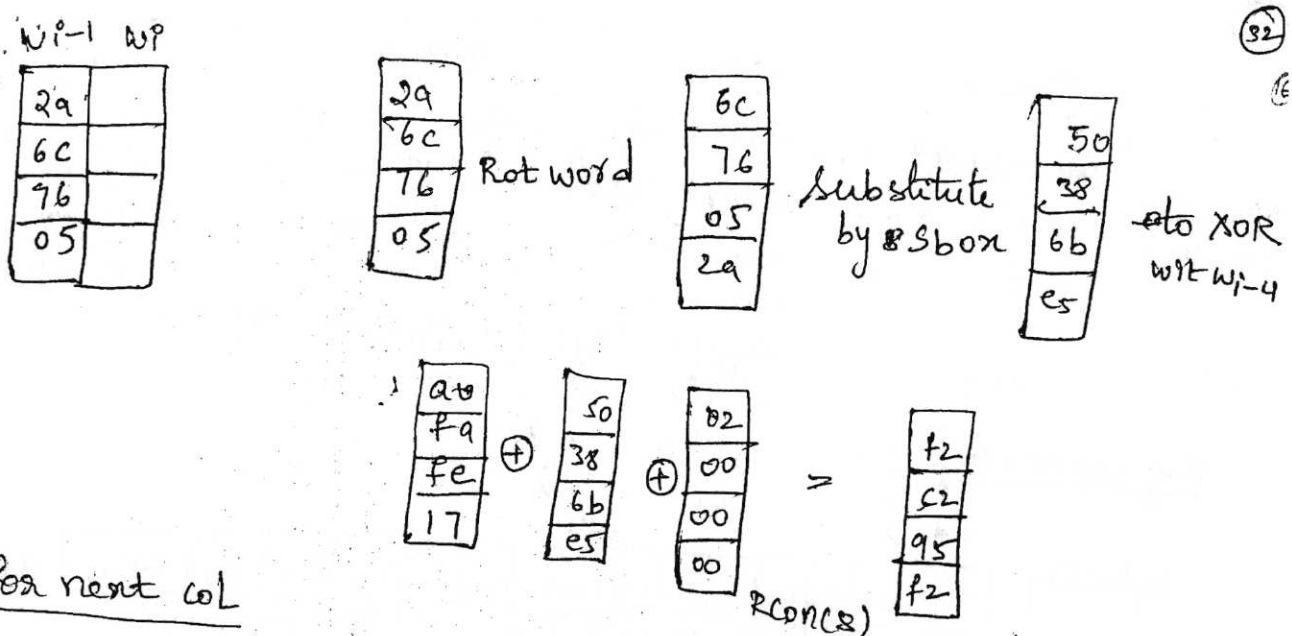
28		a0		88
ae	⊕	fa	=	54
d2		fe		5c
a6		17		b1

|ly

w_{i-4}		w_{i-1}		
ab		88		23
f7	⊕	54	=	a3
15		2c		39
88		b1		39

Now we we want to 8th col since 8 is multiple of 4

w_{i-4}		w_{i-1}		
				2a
09		23		6c
	⊕	a3	=	71



do this until col is not multiple of 4.

So till now

2b	28	ab	09	a0	88	23	2a	f2	79	23	73				
7e	ae	f7	cf	f9	54	a3	6e	e2	96	a3	59				
15	d2	15	4f	fe	2c	39	76	95	b9	39	f6				
16	a6	88	3c	17	b1	39	05	f2	43	39	7f				

Cipher key | round 1 | round 2

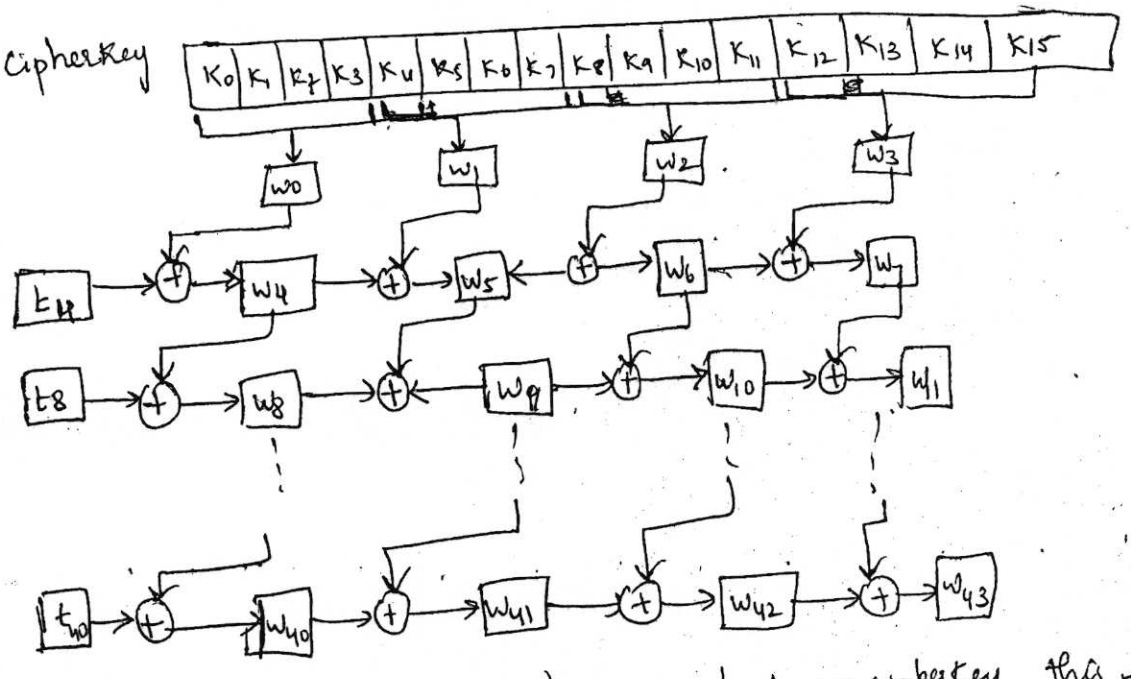
= If the number of rounds is N_r the key expansion routine creates N_r+1 128 bit round key from one single 128 bit cipher key

— Key expansion routine creates round keys word by word where a word is an array of four bytes, the routine creates $4\times(N_r+1)$ words are called $w_0, w_1, w_2, \ldots w_{4(N_r+1)-1}$

— In AES-128 version (10 rounds) there are 44 words, in AES-192 (12 rounds) 52 words & AES-256 (14 rounds) 60 words.

Round	Words			
Pre-round	w_0	w_1	w_2	w_3
1	w_4	w_5	w_6	w_7
2	w_8	w_9	w_{10}	w_{11}
----	----	----	----	----
N_r	w_{4N_r}	w_{4N_r+1}	w_{4N_r+2}	w_{4N_r+3}

Key expansion



1. first four words (w_0, w_1, w_2, w_3) are made from cipherkey, this is thought of as an array of 16 bytes, first four (K_0 to K_3) bytes next four bytes (K_4 to K_7) become w_1 & so on.

2. the rest of words (w_i for $i = 4$ to 43) are made as follows

a) If $(i \bmod 4) \neq 0$ $w_i = w_{i-1} \oplus w_{i-4}$ i.e each word is made from the one at the left and the one at the top.

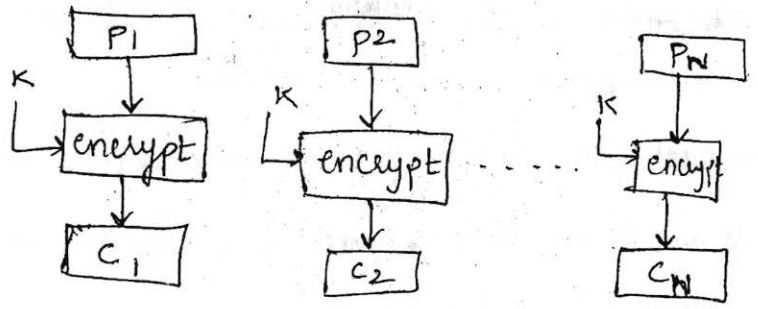
b) If $(i \bmod 4) = 0$ $w_i = t \oplus w_{i-4}$ Hence t a temporary word, is result of applying two routines Subword & Rotword on w_{i-1} and XORing the result with a round Constants Rcon

Block cipher operation

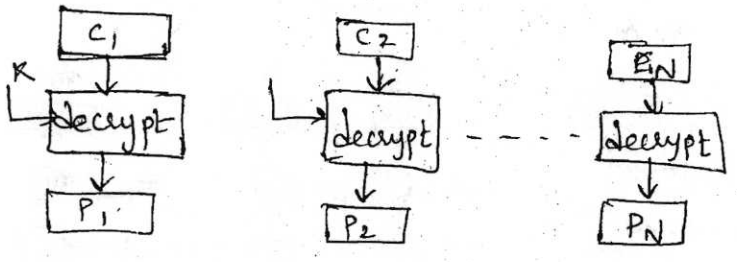
Modes of operation

- block cipher: operates on fixed length b-bit input to produce b-bit ciphertext
- what about encrypting plaintext longer than b bits?
- Break plaintext into b-blocks (padding if neccessary and apply cipher on each block)
- security issues arise: different modes of operation have been developed.

Electronic code break



encryption $C_i = E_K(P_i)$



decryption: $P_i = D_K(C_i)$

Security issues

1. Patterns at the block level are preserved ex: equal blocks in the plaintext became equal blocks in the ciphertext. i.e if eve finds out the ciphertext blocks 1, 5 & 10 are same then he knows that plaintext blocks 1, 5 & 10 are the same.
2. block independency creates opportunities for eve to exchange some ciphertext blocks without knowing the key.

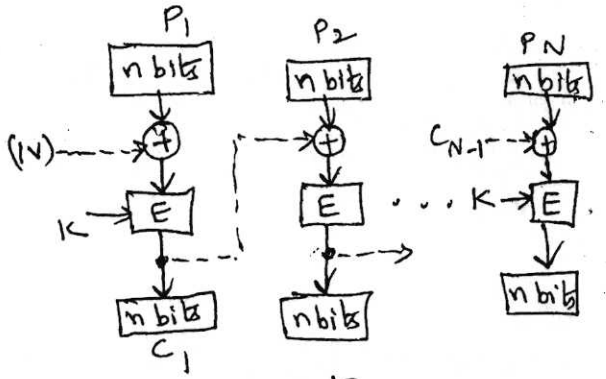
Application: area where the independency of the ciphertext block is useful

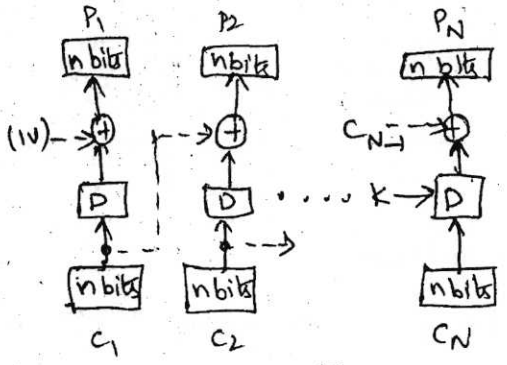
database or decrypted before they are retrieved, order of encryption and decryption of blocks is not important in this mode access to the database can be random if each record is a block or multiple blocks.

2. Cipher block chaining mode (CBC)

each plaintext block is exclusive-ored with the previous ciphertext block before being encrypted. when a block is encrypted the block is sent but a copy of it is kept in memory to be used in the encryption of the next block.

- There is no ciphertext block before the first block in this case a phony block called the initialization vector (IV) is used
- sender and receiver agree upon a specific predetermined IV
- At sender side X-ORing is done before encryption at the receiver site decryption is done before exclusie-oring.





$C_0 = IV$

$C_i = E_K(P_i \oplus C_{i-1})$

Initialization vector (IV): should be known by the sender and receiver. integrity of the vector plays an important role in the security of CBC mode. IV should be kept safe from change.

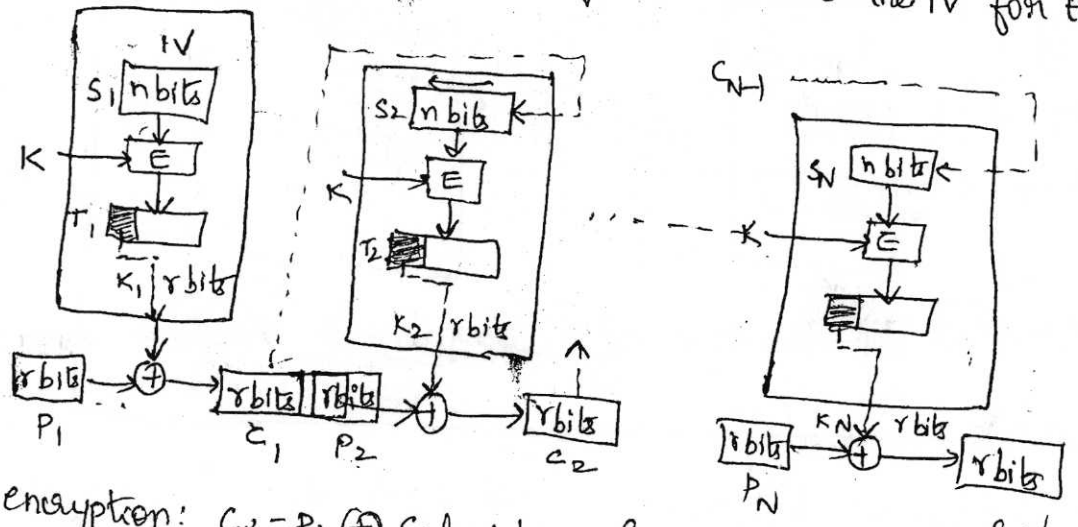
Security issues

- Patterns at block level are not preserved

Cipher feedback (CFB) mode

- In the ~~mode~~ mode the size of the block used is DES or AES is n but the size of the plaintext or ciphertext block is r where $r \leq n$.
- encrypt & decrypt the contents of a shift register S of size n. encryption done by x-oring an r-bit plaintext block with r bits of the shift register
- decryption is done by x-oring an r-bit ciphertext block with r bits of the shift register
- for each block the shift register S_i is made by shifting the shift register S_{i-1} r bits to the left and filling the rightmost r bits with C_{i-1}, S_i is then encrypted to T_i only the rightmost r bits of T_i are exclusive-ored with the plaintext block P_i to make the C_i.

Note: S_1 which is not shifted is set to the IV for the first block.



encryption: $C_i = P_i \oplus \text{SelectLeft}_r \{ E_K [\text{shiftleft}_r (S_{i-1}) \mid C_{i-1})] \}$

decryption: $P_i = C_i \oplus \text{SelectLeft}_r \{ E_K [\text{shiftleft}_r (S_{i-1}) \mid (C_{i-1})] \}$

$C_1 = P_1 \oplus S_s [E(K, IV)]^{or}$ for decryption also encryption

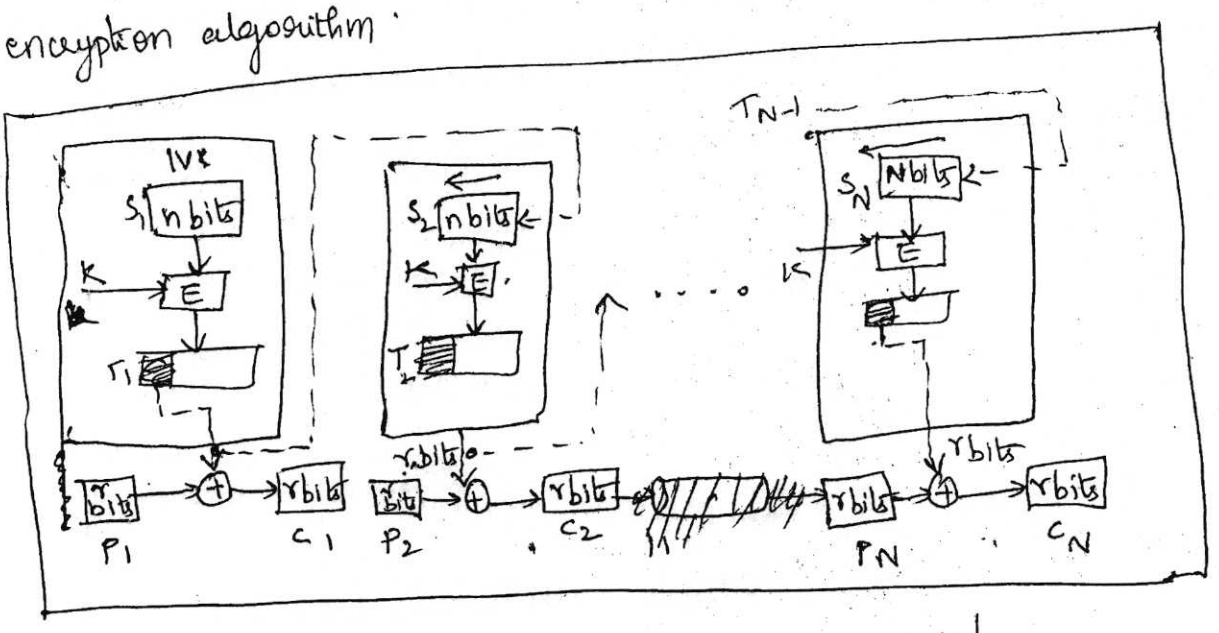
issues 1. Just like CBC patterns at the block level are not preserved
2. More than one message can be encrypted with same key but the value of IV should be changed for each msg.
3. eve and can add some ciphertext block to the end of the ciphertext stream.

Application

No padding bcz the size of the plaintext block is normally fixed (8 for a character 1 for a bit)

output feedback Mode

- each bit in the ciphertext is independent of the previous bit or bits this avoids error propagation.
- If an error occurs in transmission it does not affect the bits that follow, like CFB both the sender & the receiver use the encryption algorithm.

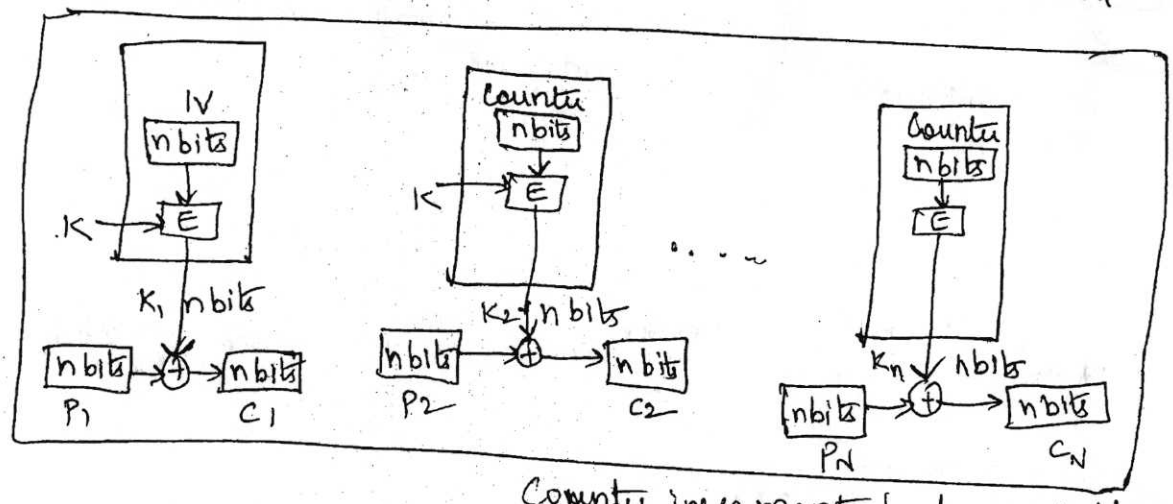


Cons issues: Pattern at block level are not preserved
Any change in the ciphertext affects the plaintext encrypted at the receiver side

Counter (CTR) Mode

- No feedback. the pseudorandomness in the key stream is achieved using a counter, An n bit counter is initialized to a predetermined value (IV) and incremented based on a predefined rule (mod 2^n)
- To provide better randomness the increment value can depend on the block number to be incremented.

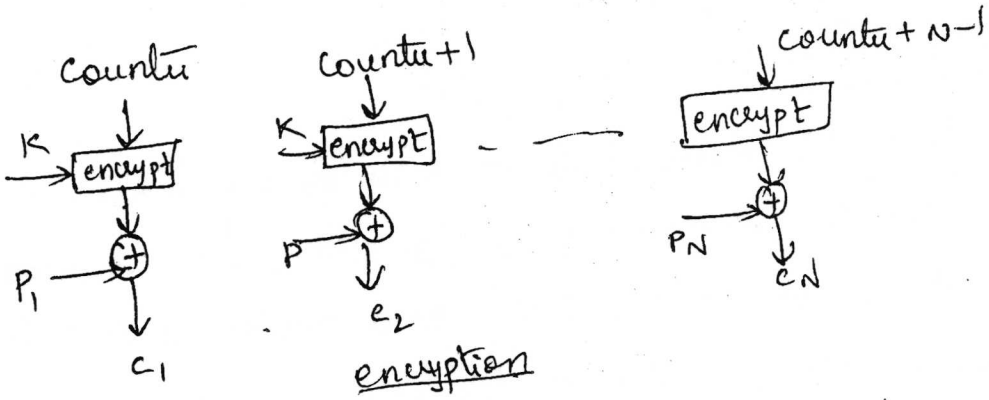
encryption: $C_i = P_i \oplus E_{K_i}(\text{counter})$ Decryption: $P_i = C_i \oplus E_{K_i}(\text{counter})$



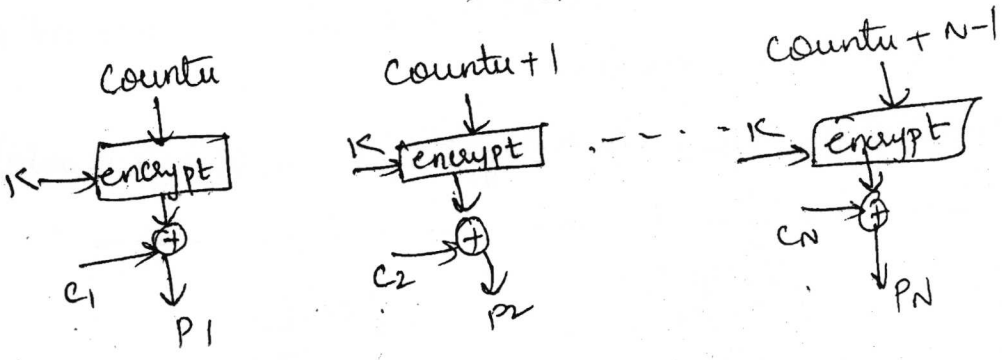
Counter incremented for each block

- CTR has increased recently with applications to ATM (asynchronous transfer mode) network security & IPsec
- counter value must be different for each plaintext block that is encrypted
- the counter is encrypted and then XORed with the plaintext block to produce ciphertext block
- for decryption the same sequence of counter values is used with each encrypted counter XORed with a ciphertext block to recover the corresponding plaintext block.

adv: H/w & S/w efficiency, preprocessing, Random access, provable security, simplicity


counter
K
encrypt
P_1
C_1
counter+1
K
encrypt
P
e_2
counter+N-1
encrypt
P_N
C_N

encryption


counter
K
encrypt
C_1
P1
counter+1
K
encrypt
C_2
P2
counter+N-1
K
encrypt
C_N
P_N

- H/w efficiency: CTR mode can be done parallely on multiple blocks of P.T or C.T, for chaining modes the alg must complete the computation on one block before beginning on the next block this limits max throughput

- S/w efficiency: S/w that supports parallel feature such as aggressive pipelining, multiple instruction dispatch per clock cycle, large no of registers, SIMB instructions can be efficiently utilized

- Preprocessing: encution of encryption alg does not depend on the i/p of P.T & C.T, if sufficient memory is available & security is ~~transmitte~~ maintained, preprocessing can be used to prepare the o/p of the encryption boxes that fed into the XOR functions

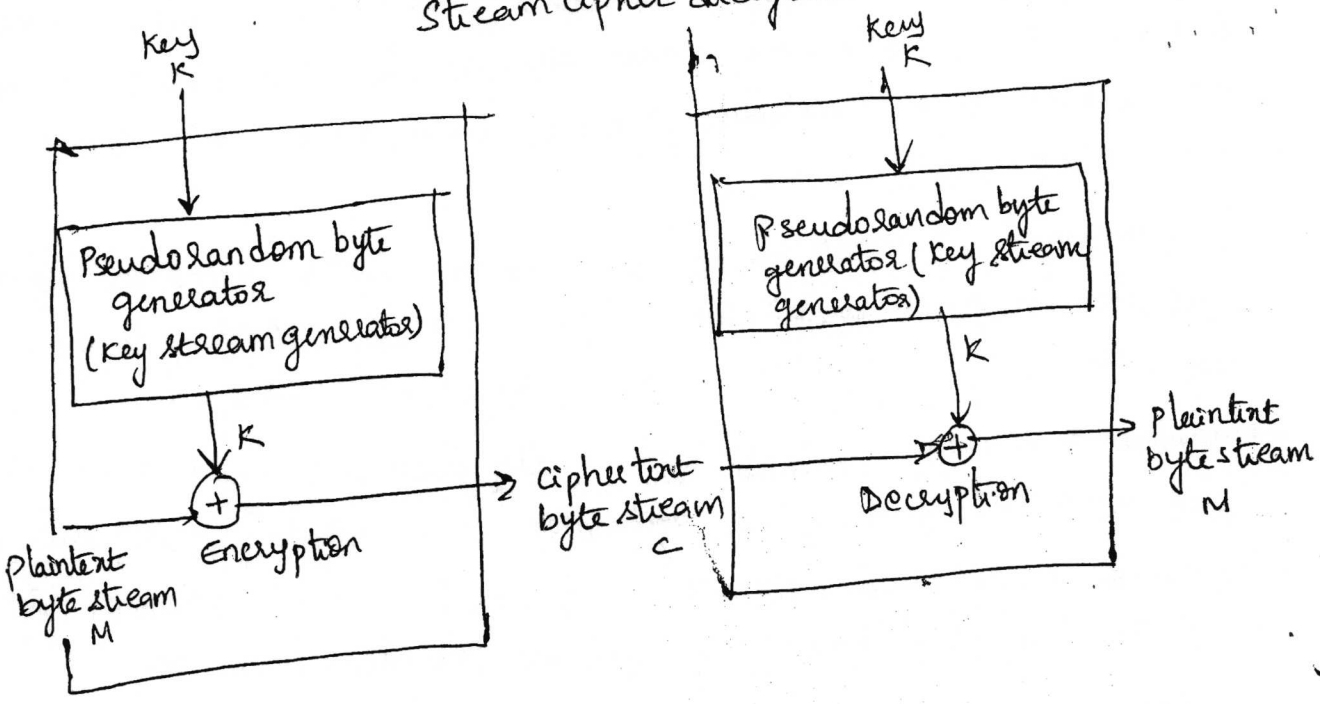
- Random access: ith block of P.T or C.T can be processed in random access fashion, with chaining mode ith block can access until the i-1 block are computed

- Provable security: CTR is as secure as other modes

- Simplicity: Unlike ECB & CBC modes CTR mode requires only the implementation of the encryption alg & not the decryption alg. this matters more when decryption algorithm differs substantially from encryption alg as it does for AES.

Mode	Applications
ECB	• Secure transmission of single values eg: an encryption key
CBC	• General purpose block oriented transmission • authentication
CFB	• General purpose stream oriented transmission • authentication
OFB	• Stream oriented transmission over noisy channel (eg: satellite communication)
CTR	• General purpose block oriented transmission • useful for high speed requirements.

Stream cipher diagram



The RSA ALGORITHM

RSA (Ron Rivest, Adi Shamir & Len Adleman) : it is block cipher in which the plaintext and ciphertext are integers between 0 & $n-1$ for some n., size of n is 1024 bits or 309 decimal digits i.e $n < 2^{1024}$

Description of the algorithm

- This algorithm uses expression with exponentials.
- Plaintext is encrypted in blocks with each block having a binary value less than some number n i.e block size must be less than or equal to $\log_2(n)$
- In practise block size is i bits where $2^i < n \leq 2^{i+1}$

encryption for some plaintext block M & ciphertext block C.

$$C = M^e \bmod n$$

$$M = C^d \bmod n = (M^e)^d \bmod n = M^{ed} \bmod n$$

Sender & receiver ~~must~~ ~~know~~ must known the value of n Sender knowns the value of e and only the receiver must known the value of d. thus it is public key encryption algorithm with a public key of $PU = \{e, n\}$ and a private key of $PR = \{d, n\}$

- Algorithm to satisfy public key encryption following requirements must be met.

1. It is possible to find values of e, d, n such that $M^{ed} \bmod n = M$ for all $M < n$.
2. It is relatively easy to calculate $M^e \bmod n$ & $C^d \bmod n$ for all values of $M < n$.
3. It is infeasible to determine d given e and n.

- We consider 1 step., the relation $M^{ed} \bmod n = M$ hold if e & d are multiplicative inverse modulo of

for p, q prime, $\phi(pq) = (p-1)(q-1)$ relationship between e & d can be expressed as $ed \bmod \phi(n) = 1$

it is equivalent to saying

$ed \equiv 1 \bmod \phi(n)$

$d \equiv e^{-1} \bmod \phi(n)$ i.e e, d are multiplicative inverses mod $\phi(n)$ according to modular arthematic that is true only if d is relatively prime to $\phi(n)$ equivalently $\gcd(\phi(n), d) = 1$

— Now state the RSA scheme, the requirements are

p, q, two prime numbers

$n = pq$

e, with $\gcd(\phi(n), e) = 1$; $1 < e < \phi(n)$

$d \equiv e^{-1} \bmod (\phi(n))$

Private key consists of $\{d, n\}$, Public key consists of $\{e, n\}$, suppose that user A has published its public key & that user B wishes to send the message M to A then B calculates $C = M^e \bmod n$ & transmits c. on receiving ciphertext user A decrypts by calculating $M = C^d \bmod n$

<u>example</u>

1. Select two prime numbers, $p = 17$ & $q = 11$
2. Calculate $n = pq = 17 \times 11 = 187$
3. Calculate $\phi(n) = (p-1)(q-1) = 16 \times 10 = 160$
4. Select e such that e is relatively prime $\phi(n) = 160$ & less than $\phi(n)$: we choose $e = 7$
5. determine d such that $de \equiv 1 \pmod{160}$ & $d < 160$ then correct value is $d = 23$, bez $23 \times 7 = 161 = 10 \times 160 + 1$

calculate d using Euclid's algorithm

— resulting keys are public key $PU = \{7, 187\}$ & private key $PR = \{23, 187\}$

Now imagine that Alice wants to send the plaintext 5 to Bod she uses the public exponent 13 to encrypt 5

Plaintext: 5 $c = 5^{13} \bmod 77$

$c = 26$

Bob receives the C.T 26 & uses the private key 37 to decipher the ciphertext

C.T : 26 $P = 26^{37} \bmod 77$

$= 5$

2	26
2	13-0
2	6-1
2	3-0
	1-1

fast modular exponentiation algorithm

b_i	1	1	0	1	
c	1	3	6	12	
f	5	48	71	26	

$c \leftarrow 0$; $f \leftarrow 1$
for $i \leftarrow k$ downto 0
do $c \leftarrow 2 \times c$
$f \leftarrow (f \times f) \bmod n$
if $b_i = 1$
then $c \leftarrow c + 1$
$f \leftarrow (f \times a) \bmod n$
return f

$c = 0$; $f = 1$ $k = 5$
$i = 5$ to 0
$n = 13$
$a = 5$

$c \leftarrow 2 \times c$
$= 2 \times 0 = 0$

$f = (1 \times 1) \bmod n = 1 \bmod 13 = 1$
$f = 1$

if $b_i = 1$ true so $c \leftarrow c + 1$

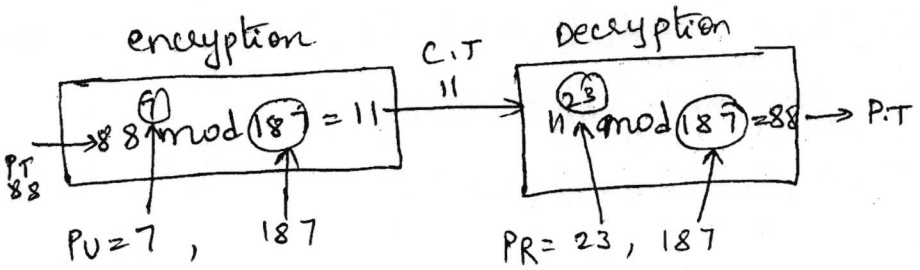
$0 + 1 \Rightarrow c = 1$

$f = (f \times a) \bmod n = (1 \times 5) \bmod 13$
$= 5 \bmod 13 = 3$

b_i	1	0	0	1	0	1
c	1	2	4	9	18	37
f	26	60	58	69	64	5

2	37
2	18-1
2	9-0
2	4-1
2	2-0
2	1-1

2	26
2	13-0
2	6-1
2	3-0
	1-1



choose d such that $de \equiv 1 \pmod{160}$

e relative prime to $\phi(n)$ -

$g(\phi n, e) = 1$.

$g(160,$

$p = 7, q = 11$

$n = 7 \times 11 = 77$

$\phi(n) = 6 \times 10 = 60$

$ed \bmod \phi(n) = 1$ So e should be relative prime and it should be between 1 & 60

1, 3, 5, 7, 11, 13, 17, 19, 23, 29, 37

find gcd(3,60) — No gcd(5,60)

Euclid(a,b)

1. $A \leftarrow a$; $B \leftarrow b$
2. if $B = 0$ return $A = gcd(a,b)$
3. $R = A \bmod B$
4. $A \leftarrow B$
5. $B \leftarrow R$
6. goto 2

$A = 5$	$A = 13$
$B = 60$	$B = 60$
~~gcd(5,60)~~	$R = 13 \% 60$
$R = 5 \% 60$	$R = 8$
$= 0$	$A = 60$
$A = 60$	$B = 8$
$B = 0$	~~gcd~~ $60 \% 8$
~~gcd(60,0) = 60~~	

~~gcd(13,60)~~

$$\boxed{gcd(a,b) = gcd(b, a \bmod b)}$$

If e to be 13 then d = 37 Note that $ex d \bmod 60 = 1$ (they are inverse of each other)

$d \equiv e^{-1} \bmod (\phi(n))$ (choose d such that multiplying e and d and doing mod with $\phi(n)$ should be 1).

$= e^{-1} \bmod (60)$

$= \frac{1}{13} \bmod (60)$

13 × 37 = 481

UNIT – III

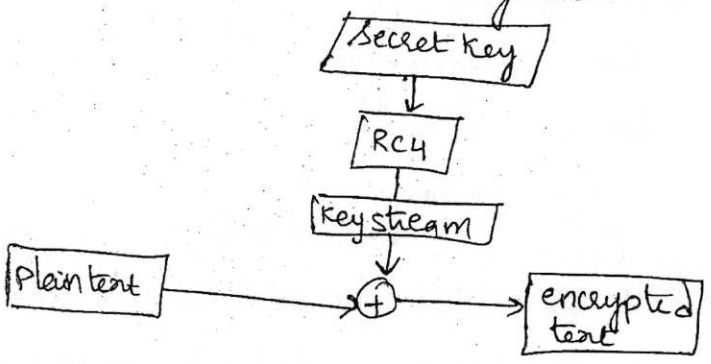
RC4 Basics

IS (3)

3.1

- A symmetric key encryption algorithm invented by Ron Rivest.
- Normally uses 64 bit and 128 bit key sizes.
- Most popular implementations is in WEP (wired equivalent privacy) protocol for 802.11 wireless networks and in SSL.
- Cryptographically very strong yet very easy to implement
- Consists of 2 parts: key scheduling algorithm (KSA) & pseudo-random generator algorithm

RC4 block diagram



Key is input to a pseudorandom bit generator that produces a stream of 8 bit numbers that are apparently random., pseudo stream is one that is unpredicatable without knowledge of the input key, the output of the generator called a keystream is combined one byte at a time with the plaintext stream using XOR operation

ex: if next byte generated by generator is 01101100 & next plaintext is

Key → 01101100
⊕ P.T → 11001100
C.T → 10100000
} encryption

Decryption

10100000 → C.T
⊕ 01101100 → Key
11001100

Steps of algorithm

- Initialize an array of 256 bytes
- Run the KSA on them
- Run it

initialization of S

```
char s[256];
int i;
for (i=0; i<256; i++)
    s[i]=i
```

After this the array would like this

S[] = {0,1,2,3, ... 254,255}

A temporary vector T is also created, if length of the key K is 256 bytes then K is transferred to T. otherwise for a key of length keylen bytes the first keylen elements of T are copied from K & K is replaced repeated as many times as neccessary to fill out T

```
/* initialization */
for (i=0 to 255 do
  S[i] = i
  T[i] = K[i mod keylen];
```

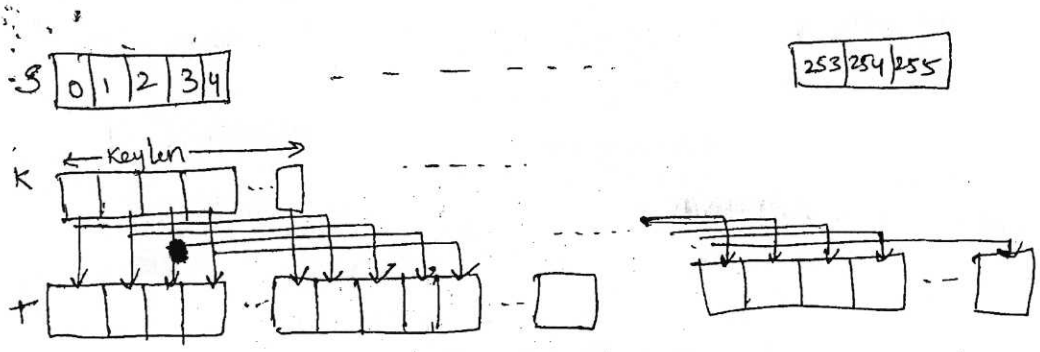
Now use T to produce the initial permutation of S, start from S[0] to S[255] . for each S[i] swapping S[i] with another byte in S according to scheme dicatated by T[i]

```
/* initial permutation of S */
j=0;
for (i = 0 to 255 do
  j = (j + S[i] + T[i]) mod 256;
  swap (S[i], S[j]);
```

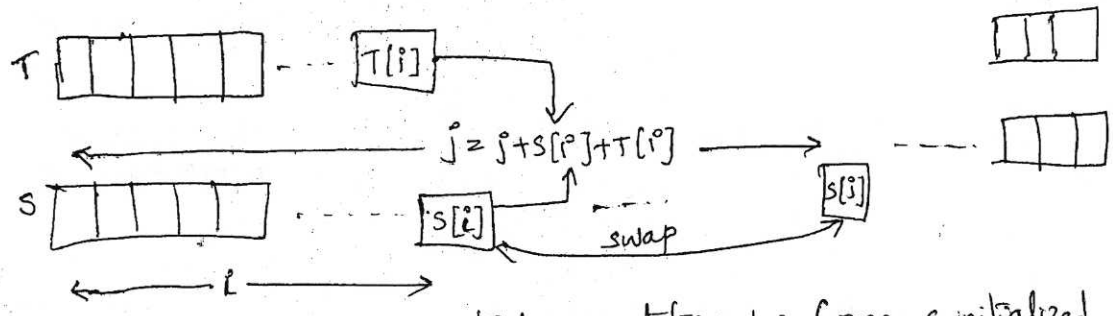
Steam generation : once the S vector is initialized, the input Key is no longer used, steam generation involves cycling all the elts of S[i], swap S[i] with another byte in S after A[255] reached process continue starting over again at S[0].

```
i, j = 0;
While(True)
  i = (i+1) mod 256;
  j = (j + S[i]) mod 256;
  swap (S[i], S[j]);
  t = (S[i] + S[j]) mod 256;
  K = S[t]
```

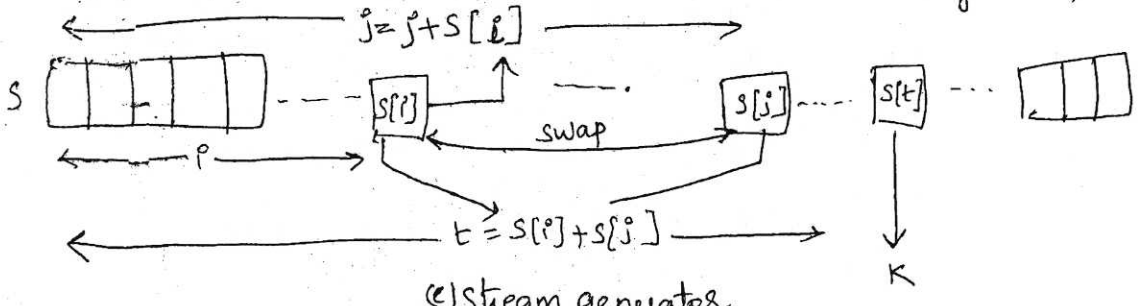
To encrypt XOR the value of K with the next byte of plaintext, To decrypt



(a) Initial state of S & T



(b) initial permutation of S (once S initialized, i/p key is no longer used)



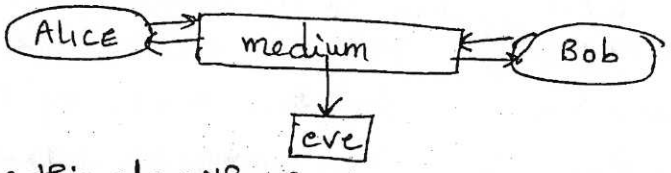
(c) Stream generator

RC4 is same as one time pad, there we used genuine random number here we use pseudorandom number stream.

CSE-D 9/2/17

516, 566, 5T, 62, 6P, 72, 6E, 59, 5X, 67, 69, 6N, 517, 5M
5L, 6M, 575

Diffie-Hellman Key Exchange Algorithm



Why use this algorithm?

Suppose Alice & bob wants to exchange some key across a media. where this medium is evesdrop by eve i.e he is able access all the information send through media, inspite of this problem alice & Bob can transfer the key without eve getting the key.

The algorithm

- Select a sufficiently large prime number q.
- Select an integer α.
- This 'α' should be a primitive root of q
- That means it should satisfy the foll:

The values of

$\alpha \bmod q, \alpha^2 \bmod q, \alpha^3 \bmod q, \ldots \alpha^{q-1} \bmod q$ should all be distinct i.e they should be different.

eg: 3 is a primitive root of 7 i.e 3^2 is primitive root of 7, 3^3 is also so on. up 3^6 is distinict.

Actual Key exchange

α and q are selected

The 2 numbers are made public

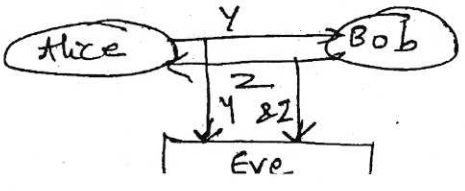
Alice

1. Select a random number M and keeps it with himself
2. Calculates $Y = \alpha^M \bmod q$

Bob

1. Selects a random number c and keeps it with himself
2. Calculates $Z = \alpha^c \bmod q$.

Now Alice sends his calculated value to bod & bod sends his calculated value to Alice, but eve have access to medium & he can easly get Y & Z value



E (4) (0) (G

A
same colour(4)
Rand " (R)
Y & R

B
(4) same col
(b) Ran "

The Key 3.5

Alice

Alice uses z to calculate key K_1 by the following way using his secret key M (uses previously selected secret key)

$$K_1 = z^M \bmod q$$

Bob

bob uses y to calculate y to calculate key K_2 by the foll way using his secret key c

$$K_2 = y^c \bmod q$$

But it is observed that $K_1 = K_2$ thus both of them get the key even after sharing certain keys publicly.

Why couldn't the spyfi eve find the key inspite of having y & z bcz α was the primitive root of q i.e

- So, $Y = \alpha^M \bmod q$ becomes a discrete log problem.
- In discrete log problem if you know Y, α & q then it is close to impossible to find M (find by brute force but if values are large then it is impossible)

Example

Public: $\alpha = 3$ $q = 353$

Alice

$M = 97$ (private)

$Y = \alpha^M \bmod q$

$Y = 40$ (public)

Bob

$c = 233$ (private)

$z = \alpha^c \bmod q$

$z = 248$ (public)

Now both exchange these keyes

40 →

← 248

Calculating Key

Alice

$K_1 = z^M \bmod q$

$z = 248$

Bob

$K_2 = Y^c \bmod q$

$Y = 40$

after calculation $K_1 = K_2 = 160$

Even ($\alpha = 3, q = 17$)

$3 \bmod 17, b, 12$

Ex2

Alice

$3 \bmod 17$

Bob

$3 \bmod 17, (13) \rightarrow b$

$b \quad 13$

Alice

$12 \equiv 3^{13} \bmod 17$ same as $3^{13^{15}} \bmod 17$

Bob

$b \equiv 3^{15} \bmod 17$

same as $3^{15^{13}} \bmod 17$

The discrete logorithm

Consider a finite mathematical group $(G, \cdot)$ '.' is the corresponding operator i.e multiplication operator for an element $\alpha \in G$ having order n. Let $\langle \alpha \rangle = \{\alpha^i : 0 \le i < n-1\}$ (α^n is not included bcz α^n is back to 1)

Order means if I take α and if I multiply n times then I get unity of this group

- The discrete logarithm problem (DLP) is to find the unique integer i, $0 \le i \le n-1$ such that $\boxed{\alpha^i = \beta}$ (i.e finding logarithm of β with respect to α)
- we denote this integer by (referred to as discrete log)

$$\boxed{\log_\alpha \beta}$$

- Dlp is the inverse of exponentiation operation
  - exponentation is easy to compute (by the square & multiply alg)
  - However in the group property choosen computation of discrete log is believed to be difficult
  - Thus the exponentation is a potential one way problem have application in pubic key cryptography.

One such application is Elgamal cryptosystem

- Let p be a prime such that Dlp in (Z_p^*) is hard (Z_p^* is multiplicative group zero is excluded) (assumption using this we see possible crypto system)
- Let $\alpha \in Z_p^*$ be a primitive element (means if we gone on multiple then we get all elts in the group primitive element)

Define $P(\text{plaintext}) = Z_p^*$ and $C(\text{Ciphertext}) = Z_p^* \times Z_p^*$

$K(\text{key}) = \{p, \alpha, \underbrace{a}_{(PK)}, \beta) \; \alpha^a \equiv \beta \pmod p\}$

(even if α & β are known by adversary

for a given key $K = \{p, \alpha, a, \beta\}$ and for a secret random number $\gamma \in Z_{p-1}$ define

3.7

$$e_K(x, \gamma) = (y_1, y_2)$$

where

$$y_1 = \alpha \bmod p$$

and $y_2 = x\beta^{\gamma} \bmod p$ (multiplicative masking)

For $y_1, y_2 \in Z_p^*$ define $d_K(y_1, y_2) = y_2 (y_1^a)^{-1}$

$\rightarrow$ $y_2 = x\beta^{\gamma} \pmod p$ | $\beta = \alpha^a \pmod p$

$y_1 = \alpha^{\gamma} \pmod p$ |

receiver does we he get y_2 he calculates x

$$y_1^a = (\alpha^{\gamma})^a = (\alpha^a)^{\gamma}$$

$$= \beta^{\gamma}$$

then receiver computes inverse of β^{γ}

$$(\beta^{\gamma})^{-1} \text{ (makes with } y_2)$$

$$(x\beta^{\gamma})(\beta^{\gamma})^{-1} = x$$

for adversary to get the value of x he must assume a

working of algorithm

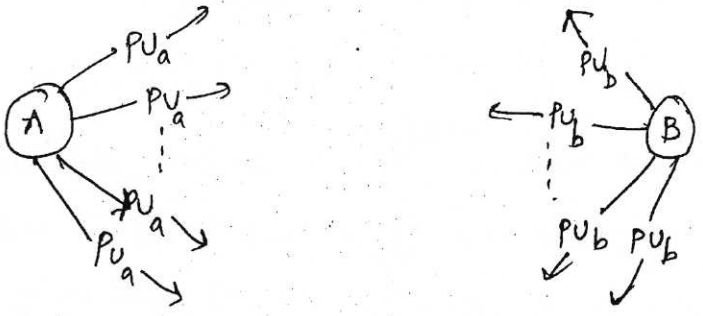
- plaintext x is masked by multiplying it by β^{γ} yielding y_2
- The value α^{γ} is also transmitted as a part of the ciphertext
- bob who has the secret a can compute β^{γ} by raising α^{γ} to a, then he obtains x by dividing y_2 with β^{γ}
- ElGamal is randomized algorithm, C.T depends on both the P.T x and the random value γ chosen by alice (encryption)
- The same P.T can be mapped into $p-1$ chipertexts depending on the choice of γ

∴ Use of public key cryptograph is in this regard
- the distribution of public keys
- the use of public key encryption to distribute secret keys.

Distribution of public keys

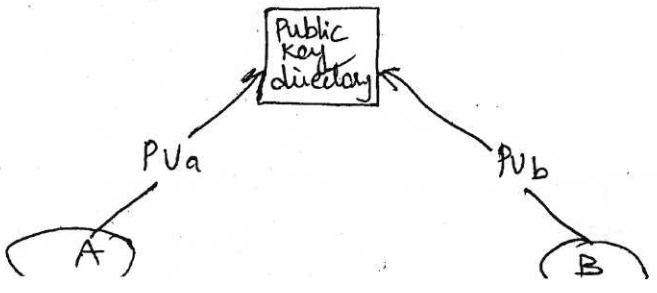
- Several techniques for the distribution of public keys
  - public announcement
  - Publicly available directory
  - Public key authority
  - Public key certificates.

Public announcement : any participant can send his or her public key to any other participant or broadcast the key to the community at large



Weakness : Anyone can forge such a public announcement i.e some user can pretend to be user A & send a public key to another Participant or boroadcast such a public key

Pubticly Availial directory : Security can be achieved by maintaining a publicly available dynamic directory of public keys ; maintenance and distribution of the public directory would have to be the responsibilty of some trusted entity or organization



1. authority mantains a directory with a {name, public key} entry for each participant

3.9

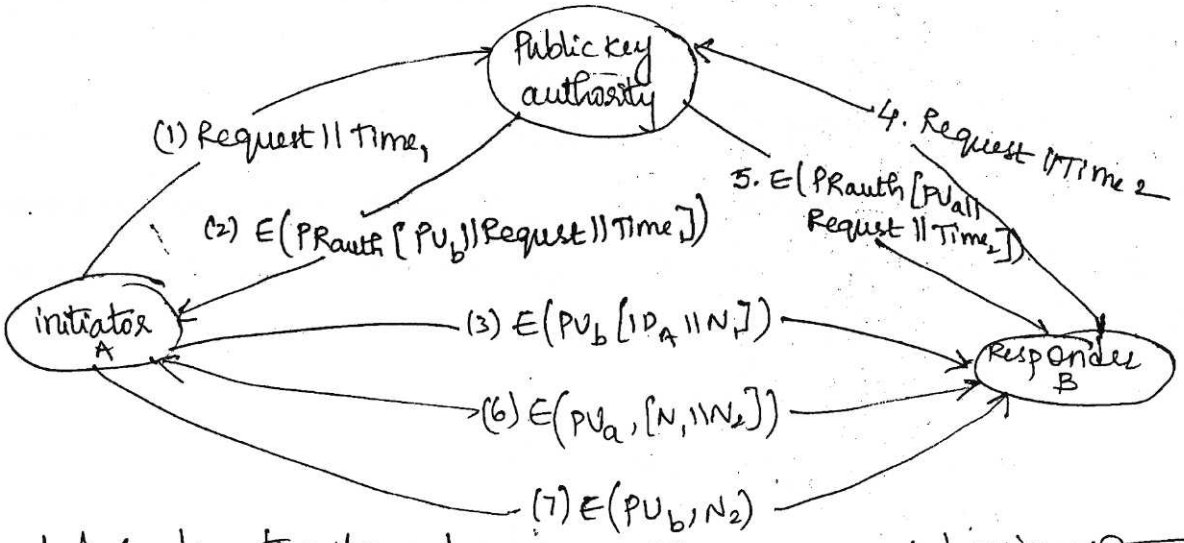
2. each participant registers a public key with the directory authority registration would have to be in person or by some form of secure authenticated communication
3. A participant may replace the existing key with a new one at any time
4. Participants could also access the directory electronically

Weakness

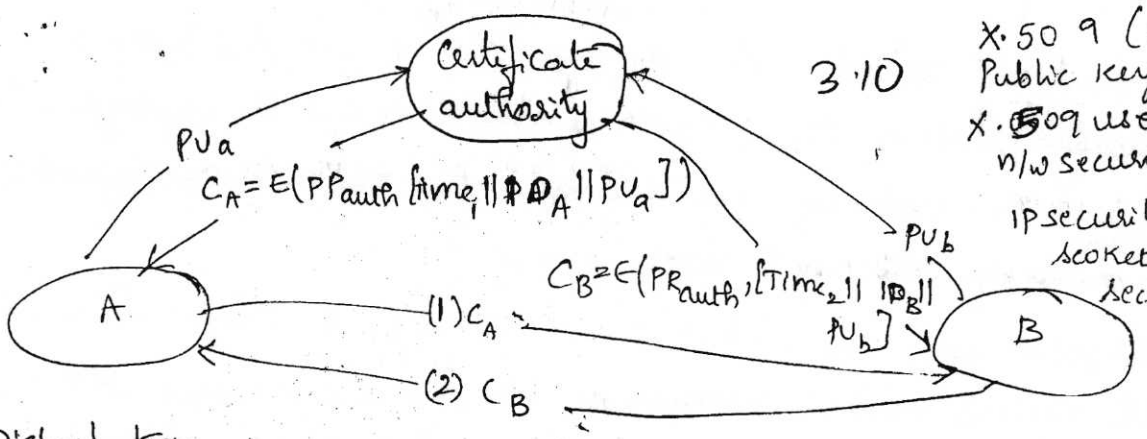
If adversary succeeds in obtaining or computing the private key of the directory authority, then public keys of authorites are known and eavesdrop on messages sent to any participant.

Public key Authority

As before the scenario assumes that a central authority maintains a dynamic directory of public keys of all participants, in addition each Participant reliably known a public key for the authority, with only authority knowing the corresponding private key



1. A sends a timestamped message ~~that is encrypted using the authority's private key PRauth~~ to the public key authority containing a request



3.10

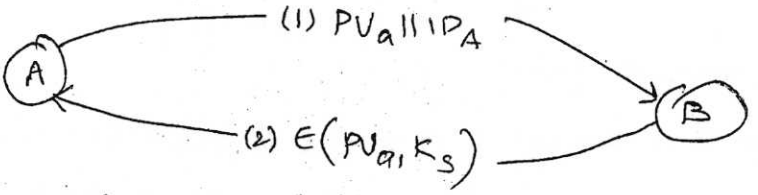
X.509 (format Public key certificat
X.509 used in mos
n/w security applic
n
IP security, secure
sockets layer (s
secure electro
transcati
(SET)

Distribution of Secret Keys using Public Key Cryptography

Simple Secret Key distribution (by Merkle)

If A wishes to communicate with B the following procedure is employed

1. A generate a public/private key pair $\{PU_a, PR_a\}$ and transmit a msg to B consisting of PU_a and identifier of A, ID_A
2. B generates a secret key K_s, & transmit it to A encrypted with A's Public key
3. A computes $D(PR_a, E(PU_a, K_s))$ to recover the secret key bcz only A can decrypt the msg only A & B will know the identity of K_s
4. A discards PU_a & PR_a & B discards PU_a



Insecure against an adversary who can intercept msg's & then either rely the intercepted message or substitute another msg such an attack is known as a man-in-the-middle attack

1. A generates a public/private key pair $\{PU_a, PR_a\}$ & transmits a msg intended for B consisting of PU_a and an identifier of A, ID_A
2. E intercepts the msg, creates its own public/private key pair $\{PU_e, PR_e\}$ and transmits $PU_e || ID_A$ to B
3. B generates a secret key K_s & transmits $E(PU_e, K_s)$

2. The authority responds with a message that is encrypted using the authority's private key PR_{auth}. Thus A is able to decrypt the message using the authority's public key. ∴ A is assured that the message originated with the authority. The msg includes following
   - B's public key, PU_b which A can use to encrypt messages desired for B
   - the original request to enable A to match this response with the corresponding earlier request & to verify that the original request was not altered before reception by the authority
   - The original timestamp so A can determine that this is not an old message from the authority containing a key other than B's current public key.

3. ~~A stores B's public key from the authority in the same manner as A retrieved B's public key~~

3 A stores B's public key & also uses it to encrypt a message to B containing an identifier of A (ID_A) & a nonce (N_1) used to identify this transaction uniquely

4.5. B retrieves A's public keys ~~have~~ from the authority in the same manner as A retrieved B's public key

~~At this~~ A & b delivered public keys securely

6. B sends a msg to A encrypted with PU_a & containing A's nonce (N_1) as well as a new nonce generated by B (N_2) bcz only B could have decrypted message (3) & the presence of N_1 in msg (6) assures A that the correspondent is B

7. A returns N_2, encrypted using B's public key to assure B that its correspondent is A

Public key certificates

use must appeal to the authority for a public key for every other [illegible] maintaining of directory of names

Requirements for a hash function

- Purpose of hash function is to produce a "fingerprint" of a file message or other block of data
- Hash func must have the following properties
  1. H can be applied to a block of data of any size
  2. H produces ~~a fixed-length~~ a fixed length o/p
  3. $H(x)$ is relatively easy to compute for any given x, making both H/w and S/w implementations practical
  4. for any given hash value h it is computationally infeasible to find x such that $H(x)=h$ referred to the literature as <u>One-way Property</u>
  5. for any given block x it is computationality infeasible to find $y \neq x$ such that $H(y)=H(x)$ referred to as <u>Weak collision resistance</u>
  6. It is computationally infeasible to find any pair (x, y) such that $H(x)=H(y)$ this is sometimes referred to as <u>Strong collision resistance</u>

- first three properties are requirements for the partical applications of a hash function to msg authentication.
- 4th; one way property states that it is easy to generate a code given a msg but virtually impossible to generate a msg given a code. (it is imp if authentication technique involves the use of a secret value) (fig a)
- 5th propcety guarantees that an alternative msg hashing to the same value as a given msg cannot be found (fig b & c) prevents forgery when an encrypted hash code is used.
- 6th refers to how resistant the hash func is to a type of attack known as the birthday ...

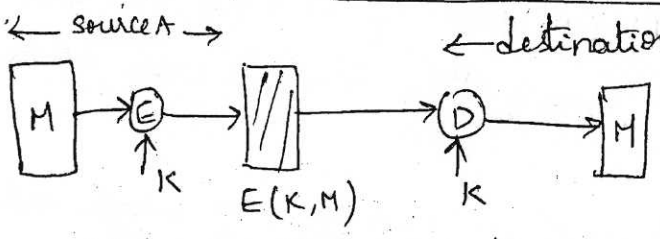
CSE-D (20/2/17)

5G, 5K, 5L, 5M, 5Q, 5R, 5T, 61, 62, 63, 65, 68, 69, 6A, 6B, 6C, 6D
6E, 6F, 6G, 6M, 6N, 6P, 6Q, 6R, 6S, 6V, 73, 76, 78, 517, 519

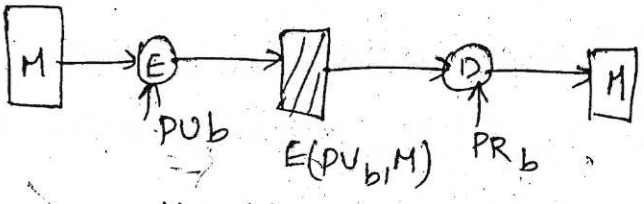
CSE-B (20/2/17)

52V, 524, 51U, 52R, 525, 51W, 52F, 52N

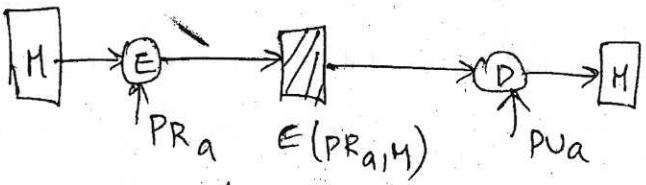
Basic use of message encryption



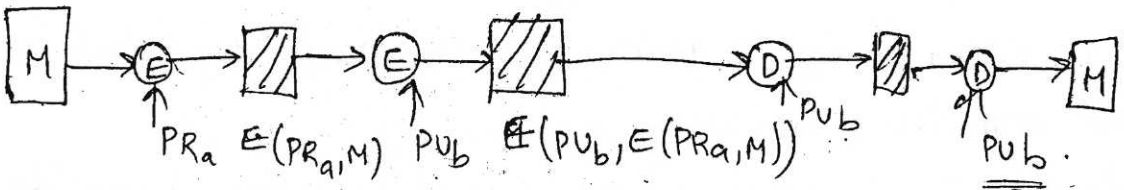
(a) symmetric encryption : confidentiality & authentication



(b) public key encryption : confidentiality



(c) public key encryption : confidentiality authentication & signature



(d) Confidentiality & authention & signature

Authentication requirements

1. disclosure
2. Traffic analysis
3. Masquerade
4. Content modification
5. Sequence modification (b/w insertion, deletion & reording)
6. Timing modification (delay or replay of msg)
7. source repudation
8. destination

Authentication functions

1. Message encryption : C.T serves as its authenticator

Message authentication code (MAC) func of the msg & a secret key that produces a fixed-length value that serves as the authenticator

3.14

Hash function: A func that maps a msg of any length into a fixed length hash value which serves as the authentication

11.1(a) msg M transmitted from source A to destination B is encrypted using a secret key K shared by A & B if no party knowns the key then confidentiality

11.1(b) Source A uses PU_b of dest B to encrypt M bez only B has the corresponding private key PR_b only B can decrypt the msg (no conf bez opponent can use B's public key to encrypt a msg claiming to be A)

11.1(c) to provide authentication A uses its private key to encrypt the msg and B uses A's public key to decrypt

(11.1 d) : A can encrypt M first using its private key which provides digital signature & then using B's public key

(dis public key alg which is complex must be executed four times rather than two in each communication)

MAC : involves the use of secret key to generate a small fixed size block of data known as a cryptographic checksum or MAC it is appended to the msg

- This technique assumes two parties say A & B share a common secret key K
- when A wants send msg to B, calculate the MAC as a func of the msg and the key : MAC = C(K, M) where

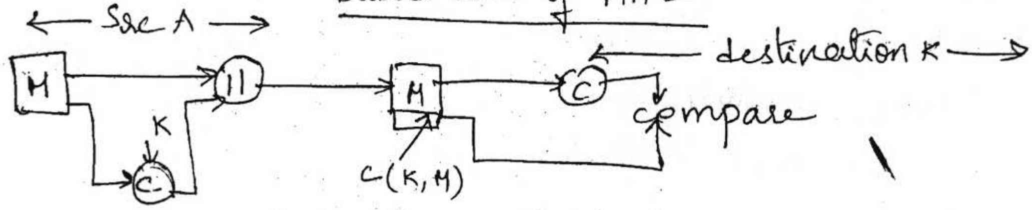
M = i/p msg
C = MAC func
K = shared secret key
MAC = message authentication code.

- msg & MAC are send to recepient, recipient perform same calculation, using same secret key to generate a new MAC. Received MAC is compared to the calculated MAC then

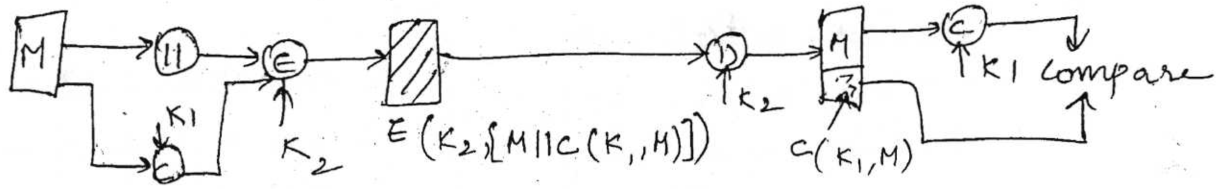
3.15

1. attacker cannot alter the MAC
2. no one else knowns the secret key
3. attacker cannot alter the sequence number

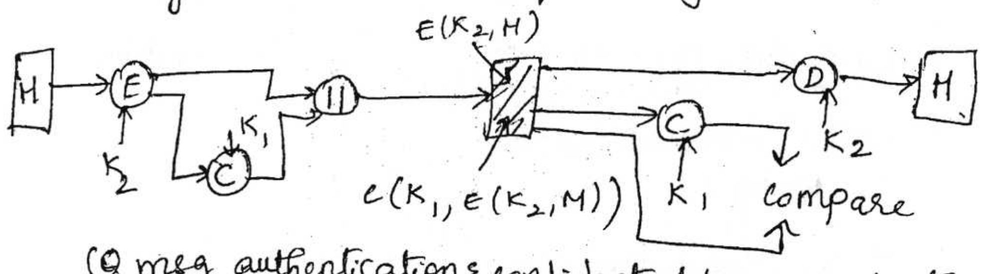
Basic uses of MAC



(a) msg authentication



(b) msg authentication & confidentiality; authentication tied to P.T



(c) msg authentication & confidentiality; authentication tied to C.T

- A MAC func is $||^{ly}$ to encryption. one difference is that the mac algorithm need not be reversible (no decryption)
- MAC func is many to one func, domain of the fun consists of msg of some arbitary length whereas the range consists of all possib MAC's and all possible keys
- If an n-bit MAC is used then there are 2^n possible MAC's whereas there are N possible messages with N>>:
- further with a k-bit key there are 2^k possible keys.

CSE-D 14/2/17 507, 509, 510, 514, 524, 522, 24, 525, 530

CSE-D 15/2/17

ex: 100-bit msg and 10-bit MAC then there are a total of 2^{100} different messages but only 2^{10} different MAC's so on average each MAC value is generated by a total of $2^{100}/2^{10} = 2^{90}$ different messages.

3.16

ex:2 If a 5 bit key is used then there are $2^5 = 32$ different mappings from the set of messages to the set of MAC values

Message authentication

→ $A \rightarrow B: H||C(K,M)$
- provides authentication
  - only A & B share K

→ $A \rightarrow B: E(K_2, [M||C(K,M)])$
- provides authentication
  - only A & B share K_1
- provides confidentiality
  - only A & B share K_2

} message authentication & confidentiality : authentication tied to p.t

→ $A \rightarrow B: E(K_2, M)||C(K_1, E(K_2, M))$
- provides authentication
  - using K_1
- provides confidentiality
  - using K_2

- MAC does not provide digital signature because both sender & receiver share the same key.

Situations in which MAC is used

- no of applications in which same msg is broadcast to a no of destination ex: notification to users that the n/w is unavailable or alarm
- exchange in which one side has a heavy load & cannot afford the time to decrypt all incoming messages, authentication done by selective basis
- Authentication of comp prog in p.t is an attractive service (comp prog can

∴ Mac is a many-to-one function. how would an opponent attempt to discovery a key

3.17

- If Confidentiality is not employed then opponent has access to P.T msg and their associated MACs
- Suppose $K > n$ (Key length > MAC size) then a given a known M_1 & MAC_1 with $MAC_1 = C(K, M)$ Cryptanalyst can perform $MAC_i = C(K_i, M_1)$ for all possible key values K_i atleast one key is guaranteed to produce a match of $MAC_i = MAC_1$ Note a total of 2^K MACs will be produced but there are only $2^n < 2^K$ diff MAC values thus a no of keys will produce the correct MAC & the opponent has no way of knowing which is the correct key. on avg a tot of $2^K/2^n = 2^{(K-n)}$ keys will produce a match.

Thus the opponent must iterate the attack

- Round 1

  Given: $M_1, MAC_1 = C(K, M_1)$

  Compute $MAC_i = C(K_i, M_1)$ for all 2^K keys

  Number of matches $\approx 2^{(K-n)}$

- Round 2

  Given: $M_2, MAC_2 = C(K, M_2)$

  Compute: $MAC_i = C(K_i, M_2)$ for the $2^{(K-n)}$ keys resulting from Round 1

  Number of matches $\approx 2^{(K-2\times n)}$ & so on

on avg α rounds will be needed if $K = \alpha \times n$

Ex: if an 80 bit key & MAC is 32 bits long then

1st round 2^{48} possibilities

2nd " 2^{16} "

3rd " single key which one must be the one used by the sender.

- If keylength is less than or equal to MAC length then it is likely that

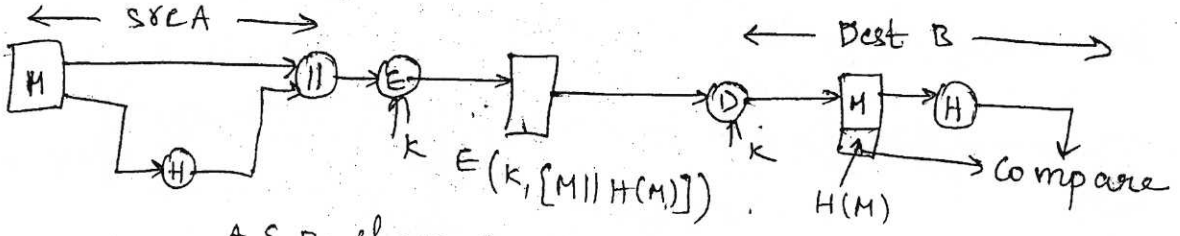
It is possible that more than one key will produce such a match in which case the opponent would need to perform the same test on a new pair.

3.18

Hash function

A variation on the MAC is the one-way hash function, a hash func accepts a variable-size message M as input & produces a fixed-size output referred to as a hash code H(M), unlike MAC a hash code does not use a key but is a function only of the i/p msg. the hash code is referred to as a message digest or hash value.

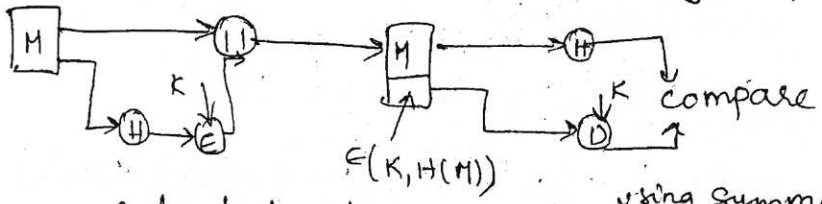
- Hash code is a function of all the bits of the msg & provides an error detecting capability : A change of any bit or bits in the msg results in a change to the hash code.



A & B share secret key msg must have come from A & has not been altered.

a) Confidentiality is provided

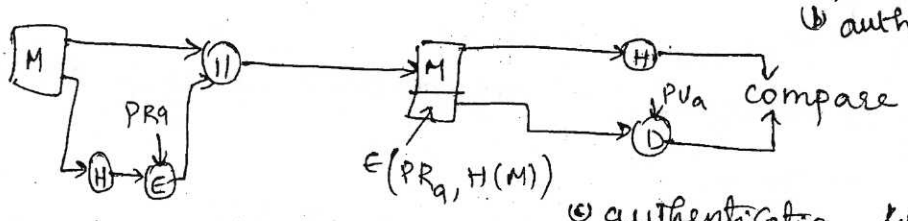
b)



only hash code is encrypted using symmetric encryption. this reduces the processing burden for those applications that do not require confidentiality

b) authentication

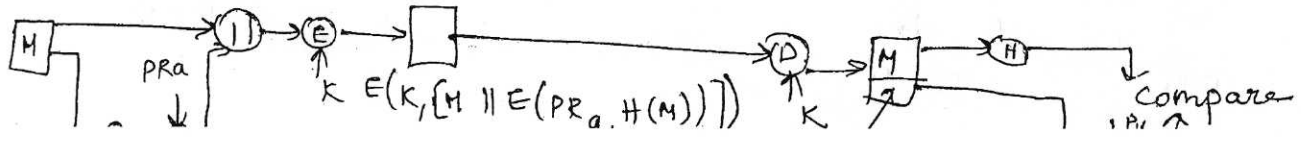
c)



c) authentication, signature

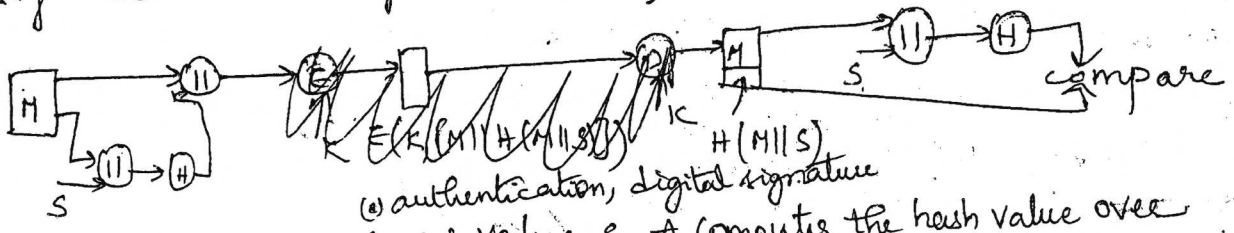
only hash code is encrypted using public key encryption & using sender's private key, provides signature bcz only the sender could have produced the encrypted hash code

d)



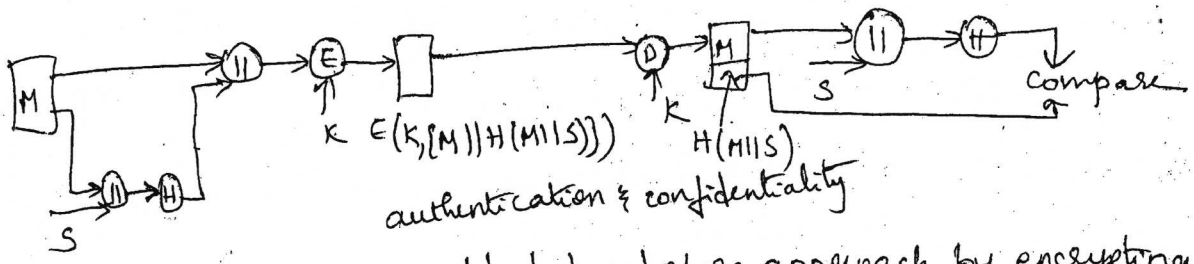
If confidentiality & digital signature is desired then the msg plus the private key encrypted hash code can be encrypted using a symmetric secret key.

3.20

e) 

(e) authentication, digital signature

share common secret value S. A computes the hash value over the concatenation of M & S & appends the resulting hash value to M bcz B possess S it can recompute the hash value to verify bcz secret value itself is not sent

f) 

Confidentiality can be added for before approach by encrypting the entire msg plus the hash code.

a) $A \rightarrow B: E(K, [M \| H(M)])$

b) $A \rightarrow B: M \| E(K, H(M))$

c) $A \rightarrow B: M \| E(PR_a, H(M)]$

d) $A \rightarrow B: E(K, [M \| E(PR_a, H(M))])$

e) $A \rightarrow B: (M \| H(M \| S)$

f) $A \rightarrow B: E(K, [M \| H(M \| S])$

Requirements of HAC

when entire msg is encrypted for confidentiality using symmetric or assymmetric encryption

16/2/17 (Presentation)

6B, 5H, 4D, 3A,
6F, 6P, 5A, 5Q
7B, 5E, 6E, 4D
5A 6Q, 5E, 4I
5S4, 5K, 6B

(presentation) CSE-B 17/2/17

533, 526, 507, 52W, 5213, 2K
525, 528, 534

9949111498

Simple hash function

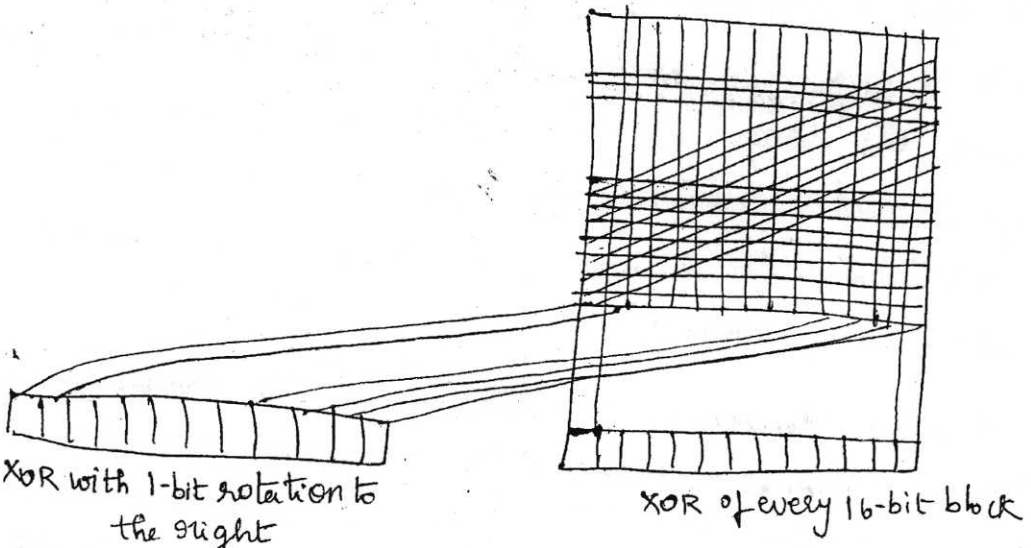
- The i/p msg is viewed as a sequence of n-bit blocks
- i/p is processed one block at a time in an iterative fashion to produce an n-bit hash function.
- One of the simplest hash functions is the bit-by-bit exclusive-OR of every block this is expressed as follows

$$C_i = b_{i1} \oplus b_{i2} \oplus \cdots \oplus b_{im}$$

C_i = ith bit of the hash code $1 \leq i \leq n$
m = number of n bit blocks in the i/p
b_{ij} = ith bit in jth block
$\oplus$ = XOR operation.

This operation produces a simple parity for each bit position & is known as longitudinal redundancy check.

- To improve matters is to perform a one bit circular shift or rotation on the hash value after each block is processed.
  1. Initially set the n-bit hash value to zero
  2. process the current hash value to the left by one bit
     a. Rotate the current hash value to the left by one bit
     b. XOR the block into the hash value



These has the effect of randomizing the

3.22

Birthday Attacks

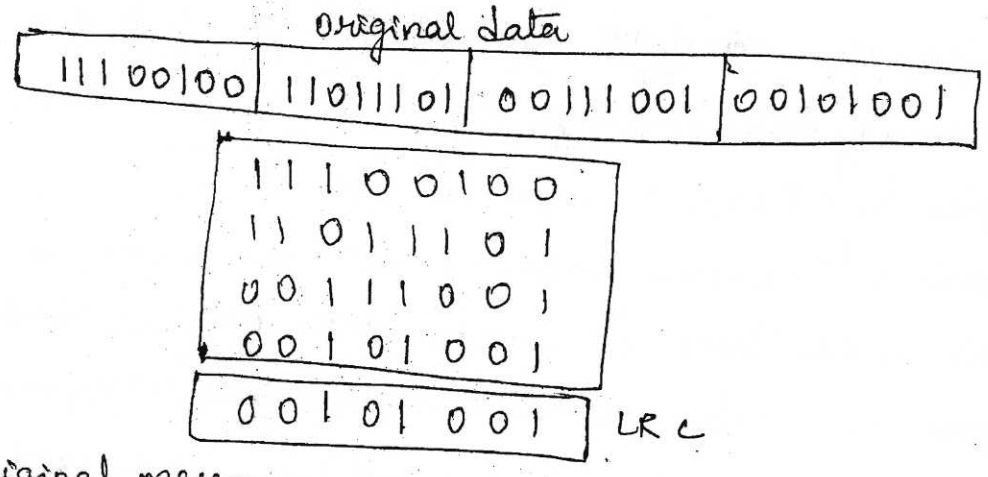
Suppose that a 64 bit hash code is used. one might think that this is quite secure ex: if an encrypted hash code C is transmitted with the corresponding unencrypted message M then an opponent would need to find M' such that H(M') = H(M) to substitute another msg and fool the receiver. on avg an opponent would have to try about 2^{63} msg's to find one that matches the hash code of the intercepted message.

a different sort of attack is possible based on the birthday Paradox (contradiction, inconsistency). Yuval proposed the following strategy.

1. The source A is prepared to "sign" a msg by appending the appropriate m-bits hash code and encrypting that hash code with A's private key (this e)
2. The opponent generates $2^{m/2}$ variations on the message all of which convey essentially the same meaning. the opponent prepares an equal number of messages all of which are variations on the fraudent messages to the substituted for the real one.
3. The two sets of msg's are compared to find a pair of messages that produces the same hash code, probability of success by the birthday contradictor is greater than 0.5. if no match is found aditional valid & fraudent msgs are generated until a match is found.
4. The opponent offers the valid variation to A for signature the signature can then be attached to the fraudent variation for transmission on the intended recipient, bez the two variations have the same hash code they will produce the same signature

Simple hash function using Bitwise XOR

	Bit 1	Bit 2	---	Bit n
Block 1	b_{11}	b_{21}	---	b_{n1}
Block 2	b_{12}	b_{22}	---	b_{n2}
---	--	--	--	--
Block m	b_{1m}	b_{2m}	---	b_{mm}
Hash code	c_1	c_2	---	c_n



Original message 7391743

Multiply 7×3	21
discard first digit	1
1×9	9
9×1	9
9×7	63
discard first digit	3
3×4	12
discard	2
2×3	6

Message digest is 6

General structure of secure hash function

- hash functions use iterative structure
  - Process message in blocks (including length)
- attacks focus on collisions in function f

Conclusion: the length of the hash code should be substantial.

Another version of the birthday attack can be used even if the opponent has access to only one message and its valid signature & cannot obtain multiple signings 3.24

Assume the opponent intercepts a msg with a signature in the form of an encrypted hash code & that unencrypted hash code is m bits long..

- calculate the unencrypted hash code G
- construct any desired message in the form $Q_1, Q_2, \cdots Q_{N-2}$
- Compute $H_i = E(Q_i, H_{i-1})$ for $1 \leq i \leq (N-2)$
- Generate $2^{m/2}$ random blocks for each block X compute $E(X, H_{N-2})$ generate an additional $2^{m/2}$ random blocks for each block Y compute $D(Y, G)$ where D is the decryption func corresponding to E
- Based on the birthday paradox with high probability there will be an X & Y such that $E(X, H_{N-2}) = D(Y, G)$
- form the msg $Q_1, Q_2, \cdots Q_{N-2}, X, Y$ this msg has the hash code G and therefore can be used with the intercepted encrypted signature

this form of attack is known as a meet-in-middle attack

Hash function of MD family uses following boolean functions

Bitwise boolean functions

3.25

$XOR(x,y,z) = x \oplus y \oplus z$

$MAJ(x,y,z) = xy \oplus xz \oplus yz$

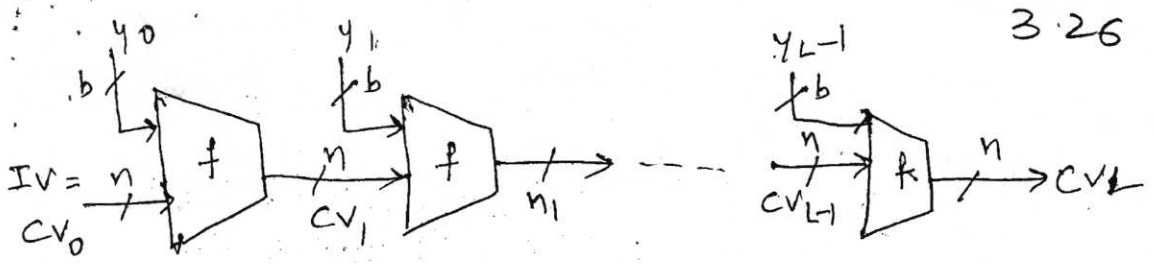
$IF(x,y,z) = xy \oplus xz \oplus z$

MD4 hash function

- was designed by Ron rivest in 1990
- i/p messages : one 512-bit i/p msg block is denoted by $M = (m_0, m_1, \ldots m_{15})$ each message word m_K consists of 32 bits which are denoted by m_{Kj} where $0 \leq j \leq 31$.
- Register words : 32 bit register word are denoted by a_i where i is the number of the compression step with $0 \leq i \leq 47$ and the register bits are indexed with $a_{i,j}$, $0 \leq j \leq 31$. each register word a_i is computed according to an update rule, four o/p registers after each step i are grouped to $(a_{i-3}, a_i, a_{i-1}, a_{i-2})$
- Boolean functions : bitwise boolean functions used in step i is denoted by $f_i(x,y,z)$ x,y,z are 32-bit words and boolean func's IF, MAJ & XOR are applied

Description : MD alg compresses an i/p with max length of 2^{64} to a 128 bit hash value

- size of one msg block is 512 bit (i/p msg is padded to fit this msg block size)
- first padding always appends a single 1 bit to the end of the msg then 0 bits are appended until the msg length is congruent to 448 modulo 512
- finally 64 bit representation of the msg length before padding was applied is appended



IV = initial value
CVi = chaining variable
Yi = ith i/p block
f = compression algorithm

L = no of i/p blocks
n = length of hash code
b = length of i/p block

Can use block ciphers as hash functions (hash alg uses repeated use of compression func)

- using $H_0 = 0$ and zero pad of final block
- Compute: $H_i = E_{M_i}[H_{i-1}]$
- using final block as the hash value
- similar to CBC but without a key

Resulting hash is too small (64-bit)

- Due to direct birthday attack
- due to "meet-in-middle attack

Description of MD Hash family

- Hash functions of MD family are iterated hash functions & follow MD design principle
- Hash function share a common structure of the compression func.
- The compression function consists of two major parts
  - message expansion
  - evaluation of a number of similar operations (these steps are grouped together into 3-5 rounds) after last step of the compression function the i/p chaining variables are added to the output

CSE-D (21/2/17)

2E, 36, 2P, 509, 514, 507

508, 53N, 3.11

(CSE-D (22/2/17) Presentations

3G, H, D, Q, R, T, U, W, X

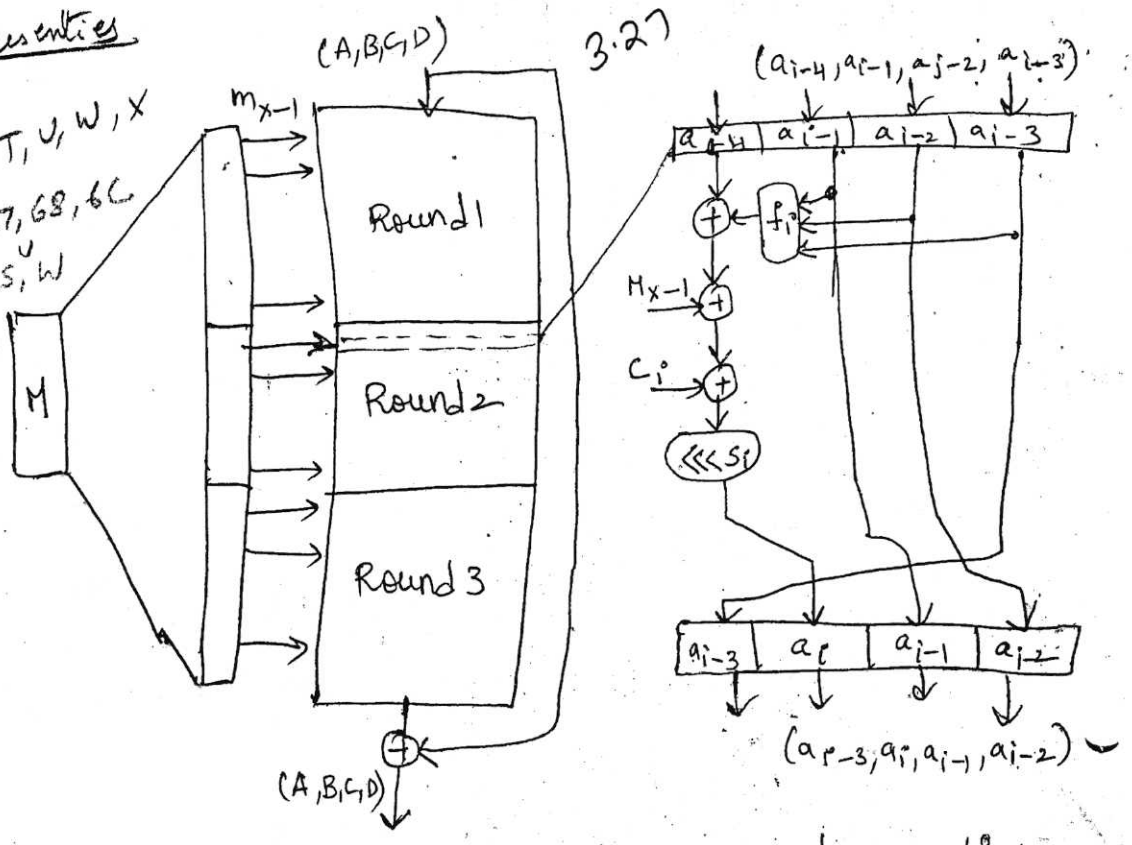
Y, 62, 65 64, 67, 68, 6C

D, E, G, H, ID, S, W

Y, 2, 71, 72, 74

516

3.27



Msg expansion ensures that each msg block is used more than once during one iteration of the compression function.

Two diff msg expansions

1) round wise permutation : In this msg words are not changed but rather used in different order in each round.

2) recursive msg expansion : designed to increase the diffusion of the msg words. (all i/p of each steps depend on all msg words)

- In each step of compression func a no of registers are updated by compressing one word of the expanded message, operations of one step are very similar in every specific hash function & consists of following basic operations.

  1. Bitwise boolean functions
  2. integer addition modulo 2^w
  3. Bit shifts & rotations.

- each step of the compression function differs only in the use of different parameters or a different boolean function

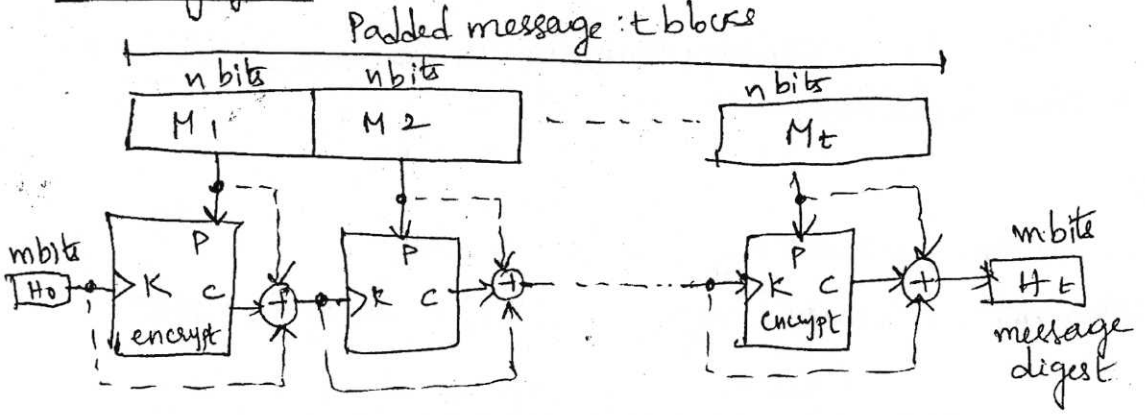
Whirlpool

- designed by Vincent Rijmen & paulo S.L.M Barreto it is endored by New european schemes for signature, integrity & encryption (NESSIE)
- Iterated cryptographic hash func based on miyaguchi-preneel scheme that uses symmetric key block ciphe in place of compression function.
- Preparation: whirlpool is a block cipher based hash func intended to provide security & performance that is comparable, if not better than that found in xion block ciphes based hash func such as SHA

features

- Code length is 512 bits
- overall structure of the hash func is one that has been shown to be resistant to the usual attacks on block cipher based hash codes

security goals



to make alg stronger against attack the P.T, the C.K & the C.T are all exclusive-ored together to create new digest

- before starting algorithm msg needs to be prepared for ~~preparati~~ processing, whirlpool requires that the length of the original msg be less than 2^{256} bits

- Padding is single 1-bit followed by any no of zero's to make the length of the padding an odd multiple of 256 bits 3.29
- after padding a block of 256 bits is added to define the length of the original message. (this block is unsigned integer)
- after padding and adding the length field the augmented msg size is an even multiple of 256 bits or a multiple of 512 bits.
- whirlpool creates a digest of 512 bits from a multiple 512-bit block msg. The 512 bit digest, Ho is initialization to all 0's, this value becomes the cipher key for encryption the first block.
- C.T resulting from encrypting each block becomes the cipher key for the next block - after being X-ORed with previous cipher key & Plaintext block. the msg digest is the final 512-bit ciphertext after the last exclusive-or operation

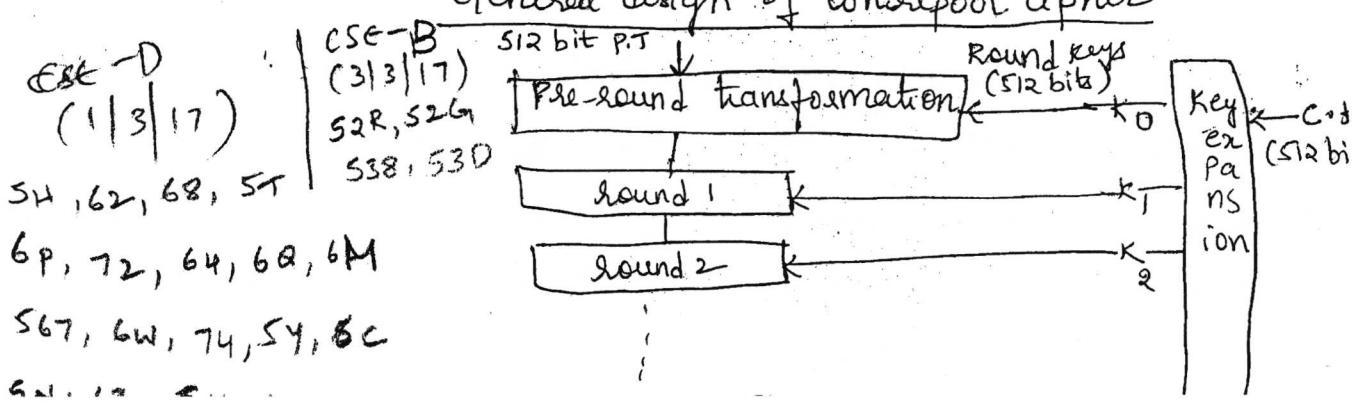
whirlpool cipher

is non feistel cipher like AES designed as a block cipher to be used in a hash alg. whirlpool cipher is compared with AES cipher & their differences.

Rounds

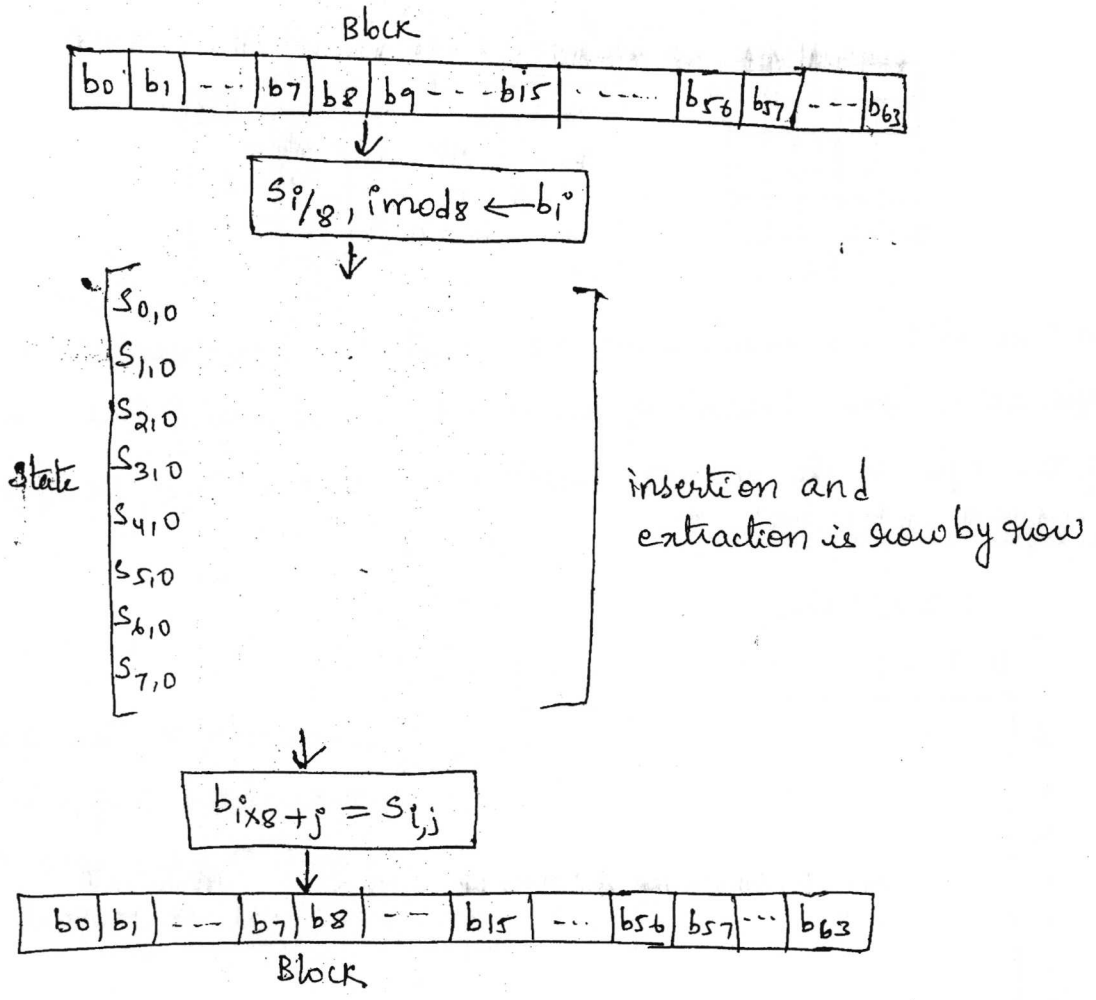
- uses 10 rounds, block & key size are 512 bits
- The cipher uses 11 roundkeys K_0 to K_{10} each of 512 bits

General design of whirlpool cipher

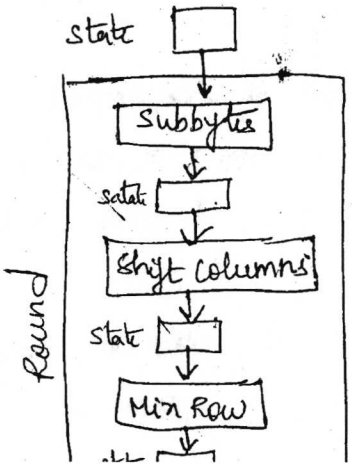


States & blocks : Size of state is 512 bits (A block is considered as a row matrix of 64 bytes, a state is considered as a square matrix 8×8 bytes)

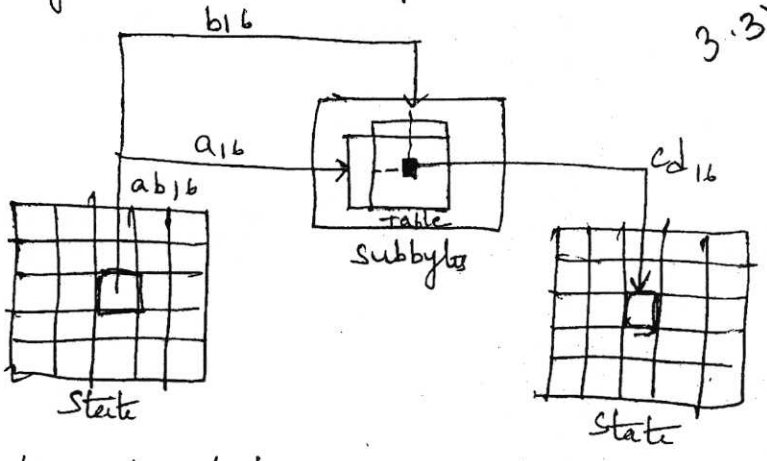
- Unlike AES the block-to-state or state-to-block transformation is done row by row



Structure of each round : each round uses four transformations



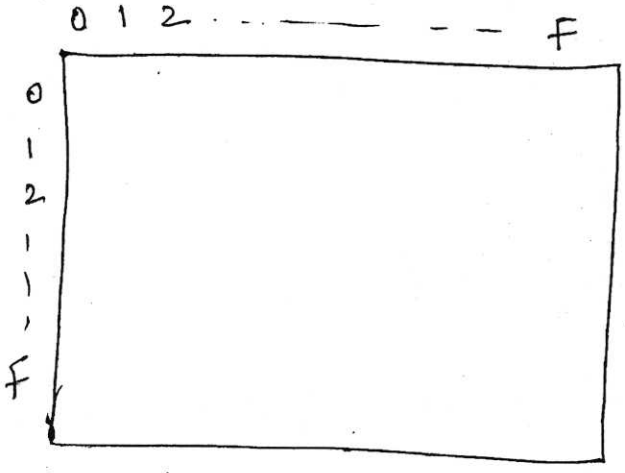
SubBytes : like AES subbytes provide a nonlinear transformation. A byte is represented as two hexadecimal digits, left defines row & the right define the column of the substitution table.



3.3)

In this state is treated as 8x8 matrix, transformation is done ~~e~~ byte at a time, contents of each byte are changed but the arrangment of the bytes in the matrix remain the same (i.e 64 different byte to byte transformations)

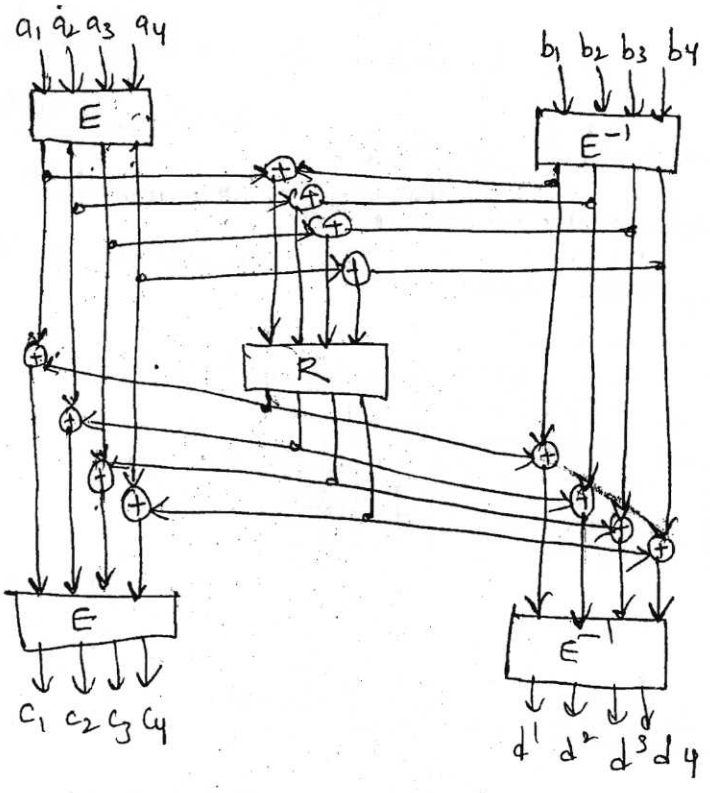
S(Box) table

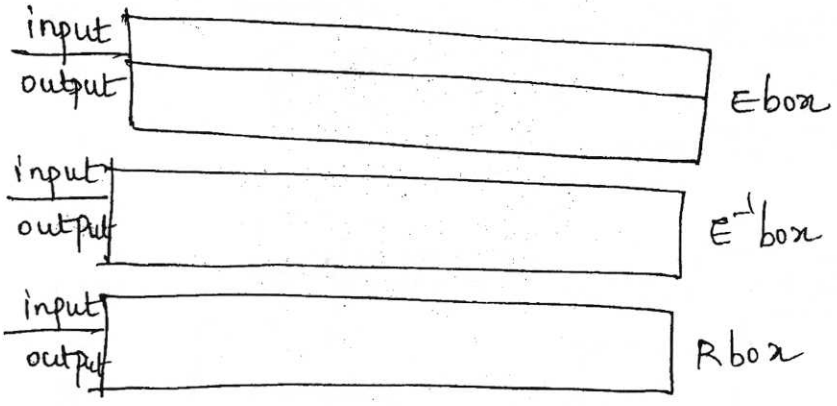


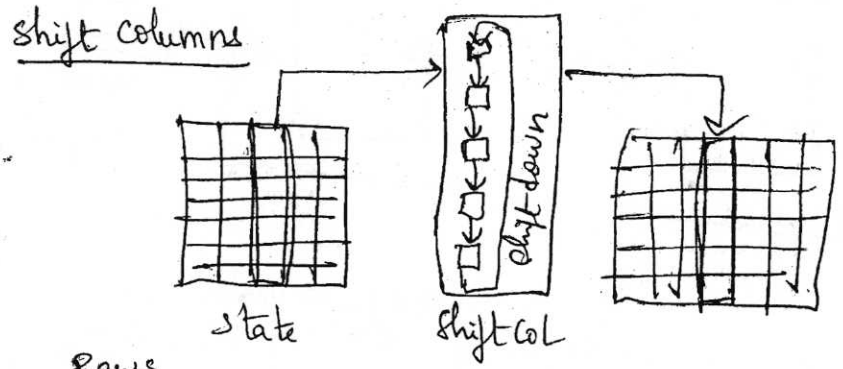
The entries in this table can be calculated algebraically using the GF(2^4) field with the irrecidual polynomial (x^4+x+1)

Each hexadecimal digit in a byte is the input to a minibox ($\in$, $\in^{-1}$) the result are fed into another minibox R. the $\in$ boxes calculate the exponential of i/p hexadecimal; R box uses a pseudorandom number generator.

$\in^{-1}$ is just inverse of $\in$ box where the roles of input & output are changed (the input/output) values for boxes are also tabulated in





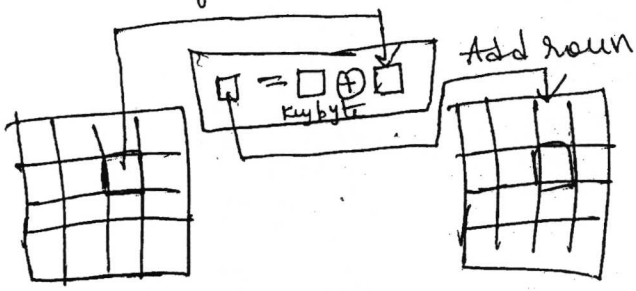


CSE – D
Presentes 2|3|17
5 6, J, H, M, N, P, Q, T
U, W, Y, 62, 63, 6, 8, 9
6C, D, E, G, N, P, S
U, W, Y, Z, 71, 2, 4, 5
8, 516, 517, 519

Mix ~~Columns~~ Rows (Mix row has same effect as Mix col in AES) It diffuses the bits. Mix rows bytes are interpreted as 8 bit words with coefficient GF(2) multiplication of bytes is done in GF(2^8) but modulus is different from the one used in AES. Whirlpool cipher uses (0x11D) or ($x^8 + x^4 + x^3 + x^2 + 1$) as the modulus

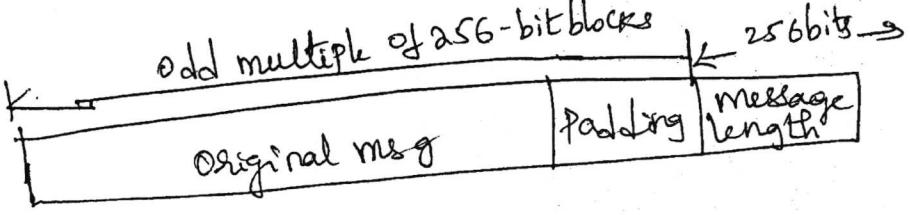
3.3

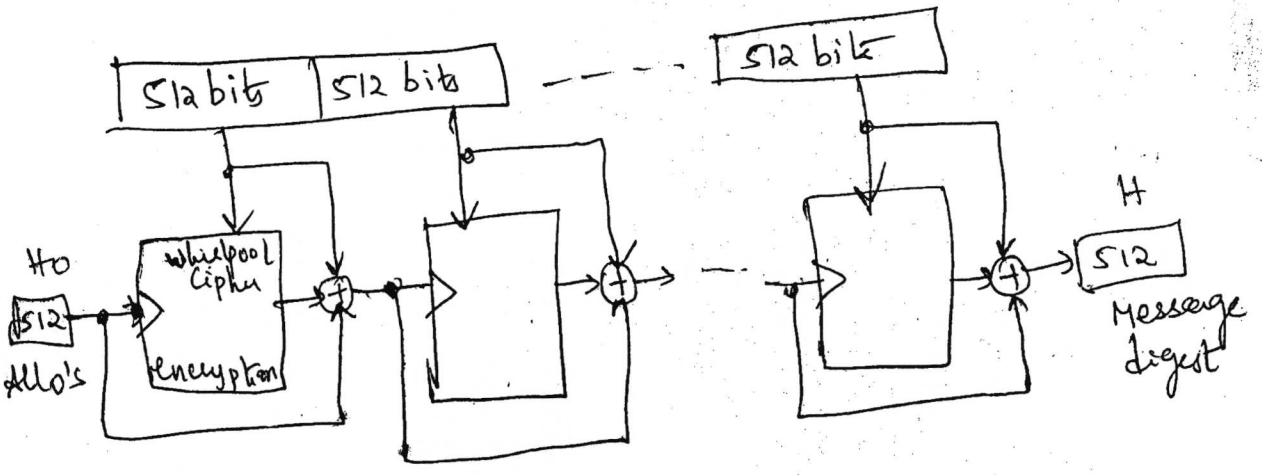
Round key : Whirlpool cipher is done byte by byte because each round key is also a state of an 8x8 matrix. a byte from data is added state is added in GF(2^8) field to the corresponding byte in the round-key state result is the new byte in the new state



Key expansion

Whirlpool hash function





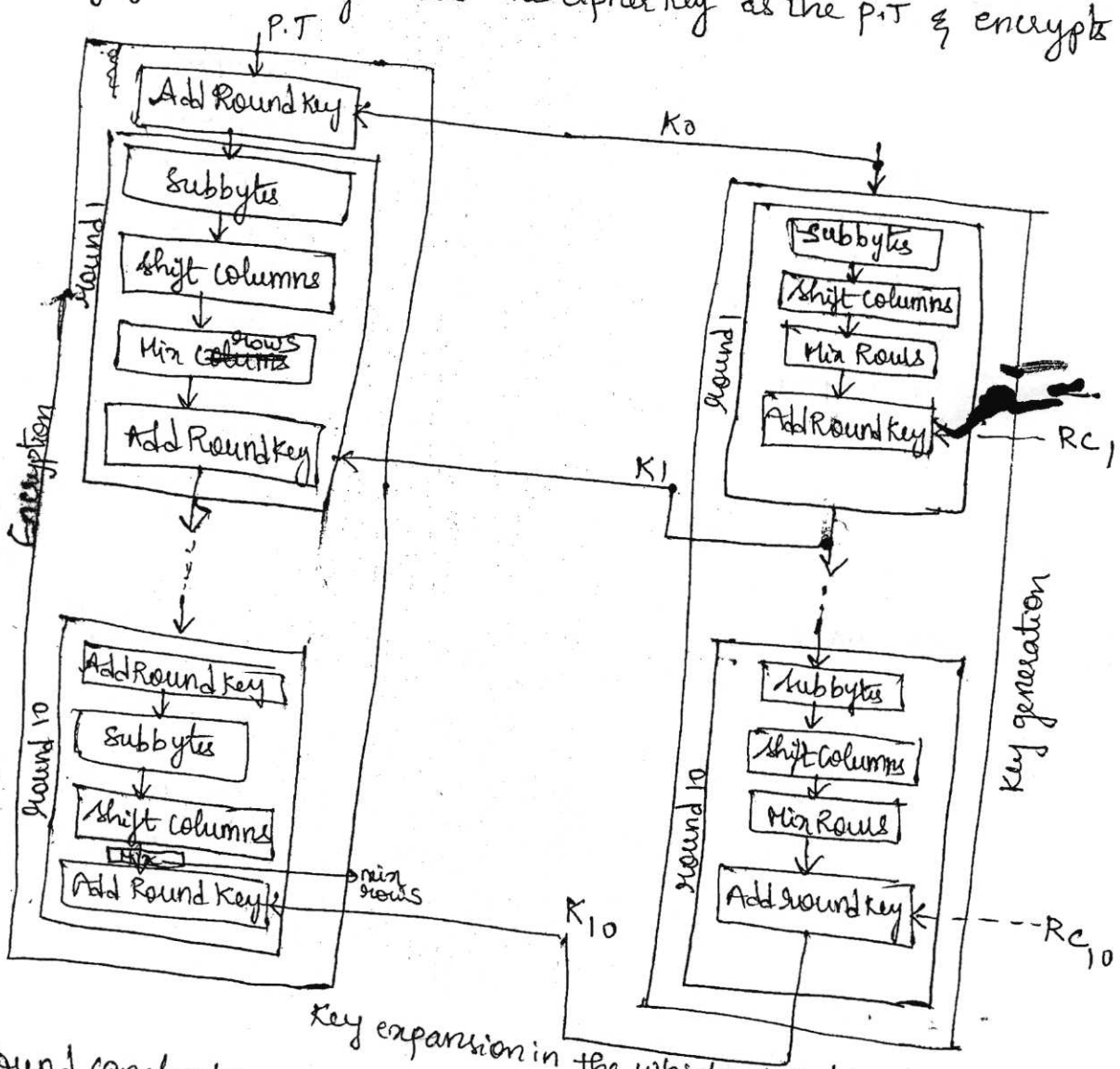
Key expansion

different from the alg in AES, instead of creating round keys. using a new alg for Whirlpool uses a copy of the encryption alg (without pre-round) to create the round keys. the o/p of each round in the encryption alg is the round key for that round.

Where do round keys for the key expansion alg comes from

for the key-expansion alg i.e key expansion alg uses constants as round keys & encryption algorithm uses the output of each round of the key-expansion alg as the round keys

— Key generation alg treats the cipher key as the P.T & encrypts it.



Round constants : Each round RC_i is a 8×8 matrix where only the first row has non zero values. the rest of the entries are all 0's the values for the first row in each constants matrix can be calculated using the subbytes transformation

$RC_{round}[row, column] = Subbytes(8(round-1)+column)$ if row = 0

$RC_{round}[row, column] = 0$ if row ≠ 0

— i.e RC_1 uses the first ...

HMAC objectives

- It is design not an algorithm which can be used for any hash function. like HMAC-MD5, HMAC-AES

3.35

- use hash functions without modifications.
- Allow for easy replace ability of embedded hash function
- perserve original performance of hash func without significant degradation
- uses and handles keys in a simple way
- Has well understood cryptographic analysis of authentication mechanism strength.

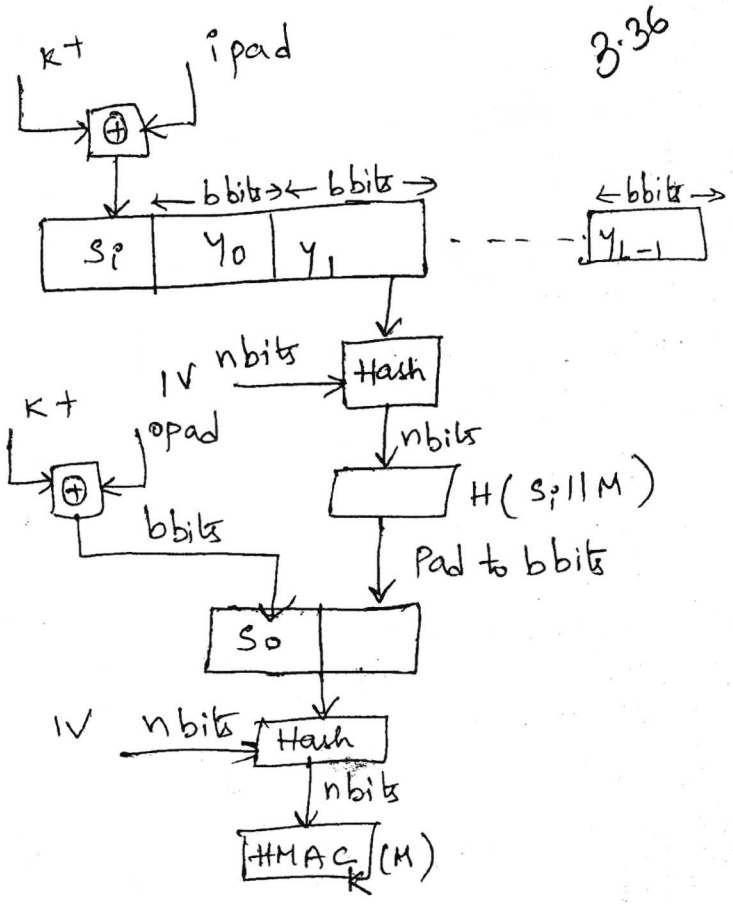
HMAC algorithm

$$HMAC_K(M) = H[K^+ \oplus opdad] \,||\, H[K^+ \oplus ipad \,||\, M])$$

H =
IV =
M =
Y_i =
L =
b =
n = 1
K =
K^+ =
ipad =
opad =

1. Append zeros to the left end of K to create b-bit string K^+ (eg if K is of length 160 bits and b = 512, then K will be appended with 44 zero bytes 0x00)
2. XOR K^+ with ipad to produce the b-bit block S_i
3. Append M to S_i
4. Apply H to the stream generated in step 3
5. XOR K^+ with opad to produce the b-bit block s
6. Append H to

3.36



K^+ — secret key
Pad it with b bits
K - 1024 compress to
512 — no of bits
ipad — fixed
36 repeated $\frac{b}{8}$ times
opad (5c repeated
64 times)

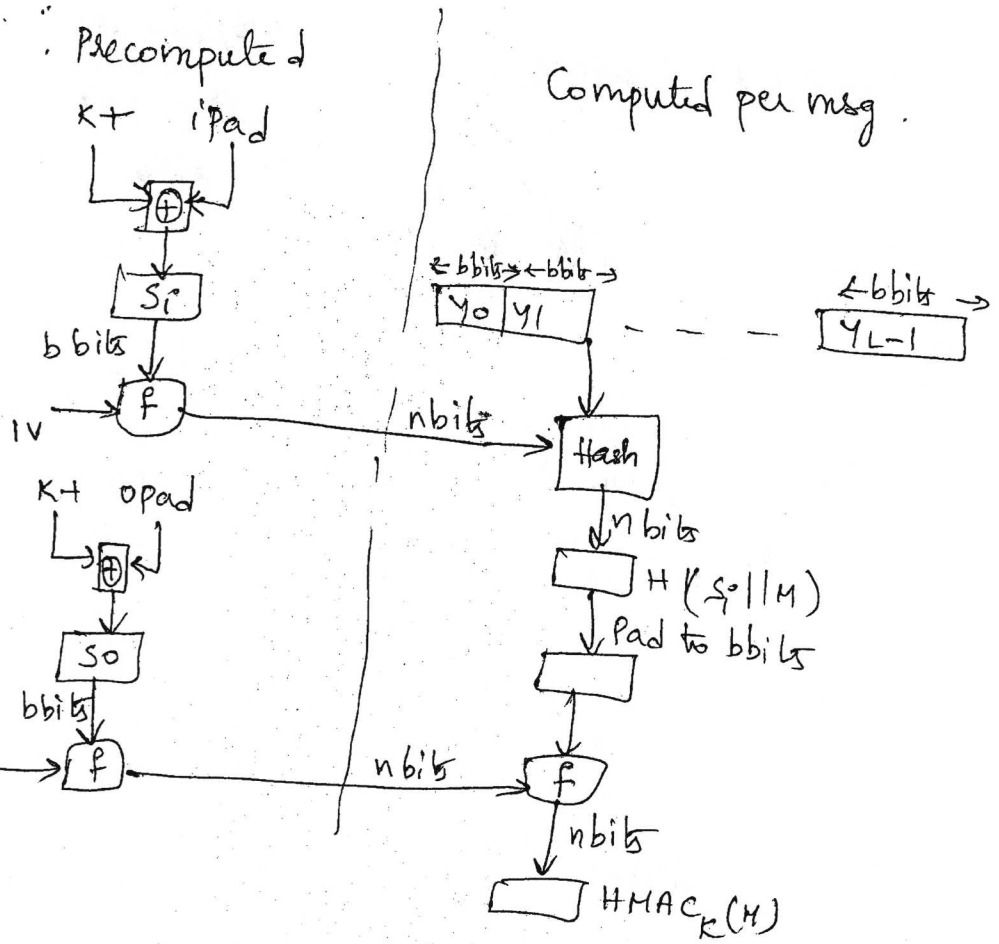
XOR with ipad results in flipping one-half of the bits of K
III^{ly} XOR " opad " " " " " " " " but
a different set of bits, in effect by passing S_i & S_o through the compression function of the hash alg. (we have pseudorandomly generated two keys from K)

A more efficient implementation is possible two quantities are Precomputed

$$f(IV, (K^+ \oplus ipad))$$
$$f(IV, (K^+ \oplus opad))$$

where f (cv, block) is the compression function for the hash func which takes as arguments a chaining variable of n bits & a block b bits & produces a chaining variable of n bits

— These quantities only need to be computed initially & every time the key changes, precomputed quantities substitute for the initial value (IV) in the hash function



27/2/17
CSE-D (presentics)
5G, 5J, 5K, P, Q, R, S
T, W, X, 4, 61, 2, 4, 5, 6
7, 8, 6B, C, D, E, G, H
GF, P, S, U, Z, 7, 72
4, 8, 516, 519

27/2/17 CSE-B
515, 52W, 52V, 507
52Z, 52E, 2P, 536
509, 5CR, 527, 537
525, 538, 52F, 52L

CMAC (data authentication algorithm defined in fips PUB 113 also known as CBCMAC (cipher block chaining message authentication code)

3.38

- this is adopted in government & industry
- this MAC is secure under a reasonable set of security criteria with few restriction. only messages of one fixed length of mn bits are processed where n is the cipher block size & m is a fixed +ve integer.

ex: given CBCMAC of a one block message say X say $T = MAC(K, X)$

- the adversary immediately knowns the CBCMAC for the two block message $X \| (X \oplus T)$ since this is once again T.
- Black & Rogaway demonstated that this limitation could be overcome using three keys : one key of length k to be used at each step of the cipher block chaining & two key of length n where k is the key length and n is the cipher block length.
- The proposed construction is refined by Iwata & Kurosawa so that the two n-bit keys could be derived from the encryption key rather than being provided seperately.
- this is adopted by NIST Cipher block message authentication code (CMAC)
- first operation of CMAC: when the message is an integer multiple n of the cipher block length b. for AES b=128 & triple DES b=64
- The msg is divided into n blocks $M_1, M_2, \ldots M_n$. the alg makes use of a kbit encryption key K and n-bit constant K_1, for AES key size is 128, 192, 256 bits, triple DES key size is 112 or 168 bits CMAC is calculated as follows

$$C_1 = E(K, M_1)$$

$$C_2 = E(K, M_2 \oplus C_1)$$

$$C_n = E(K, [M_n \oplus C_{n-1} \oplus K_1])$$

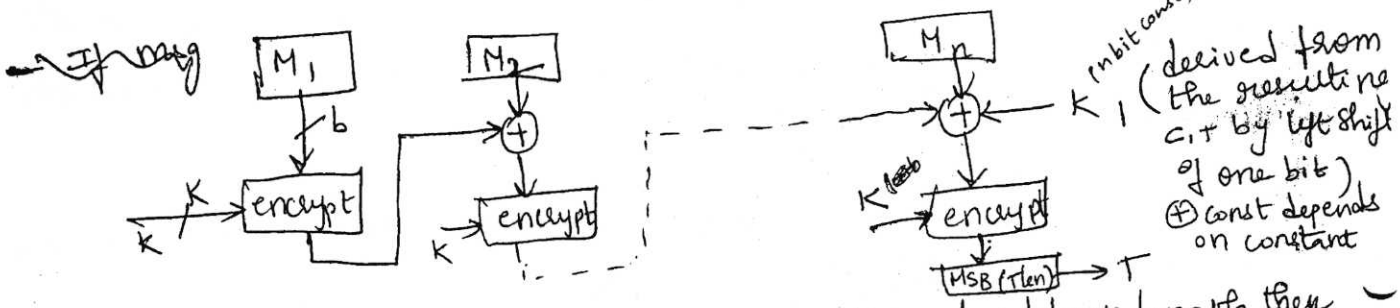
$$T = MSB_{Tlen}(C_n)$$

339

where T = MAC also referred as tag

Tlen = bit length of T

$MSB_s(X)$ = the s leftmost bits of the bit string X



If msg is not an integer multiple of the cipher block length then the final block is padded to the right (least significant bits) with a 1 & as many 0s as neccessary, so that final block is also of length b. The CMAC operation then proceeds as before except that a different n-bit key K_2 is used instead of K_1

- the two n-bit keys are derived from the K-bit encryption key as follows

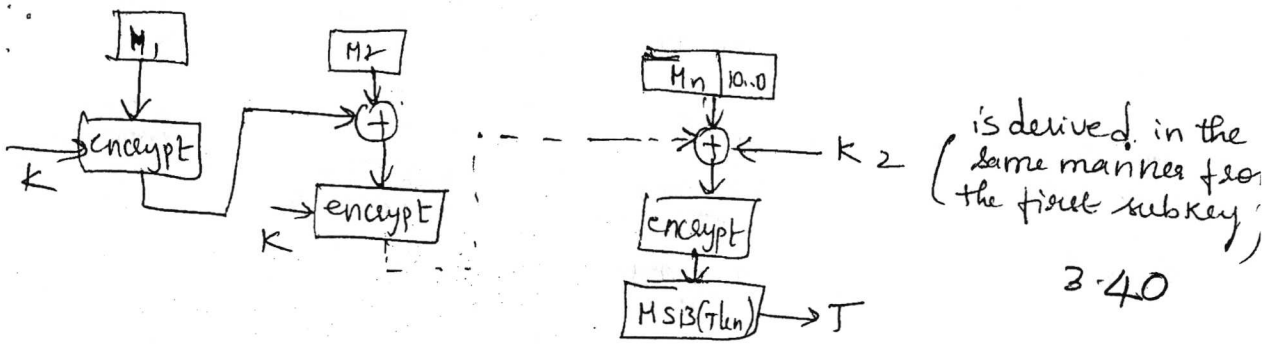
$$L = E(K, 0^n)$$

$$K_1 = L \cdot x$$

$$K_2 = L \cdot x^2 = (L \cdot x) \cdot x$$

where multiplication (·) is done in the finite field $GF(2^n)$ & x & x^2 are first & second order polynomials that are elements of $GF(2^n)$ thus binary representation of x consists of n-2 zero followed by 10; the binary representation of x^2 consists of n-3 zeros followed by 100.

- the finite field is defined w.r.t an irreducible polynomial that is lexographically first among all such polynomials with the min possible number of nonzero terms.
- For two approved block sizes the polynomials are $x^{64} + x^4 + x^3 + x + 1$ &



3.40

- To generate K_1 & K_2 the block cipher is applied to the block that consists entirely of 0bits first subkey is derived from the resulting ciphertext by a left shift of one bit & conditionally by XORing a constant that depends on the block size
- Second subkey is derived in the same manner from the first subkey

Kerberos (named after Greek three headed dog, which is there at the gate of Hades)

The problem that Kerberos addresses is this: Assume an open distributed environment in which users at workstations wish to access services on servers distributed throughout the network.

- Rather than building in elaborate authentication protocols at each server Kerberos provides a centralized authentication server whose function is to authenticate users to servers & servers to users
- Kerberos relies exclusively on symmetric encryption making no use of
- Public key encryption.

Threats exist

- user pretend to be another use
- User may alter the n/w addr
- replay attack

Motivation

Distributed architecture consisting of dedicated user workstations & distributed or centralized servers.

- three approaches to secure.

1. rely on each individual client workstation to assure the identity of its user and rely on each server to enforce a security policy based on user identification.
2. require that client systems authenticate themselves to servers
3. require the user to prove identity for each service invoked

Requirement of Kerberos

1. Secure: Kerberos should be strong enough that a potential opponent does not find it to be the weak link.
2. Reliable: one system able to backup anoth...

Transparent : Ideally user should not be aware ~~of supporting large numbers of clients & servers~~ that authentication is taking place beyond the requirement to enter a password 3.42

Scalable : System should be capable of supporting large number of clients & servers

— Kerberos is that of a trusted third party authentication service that uses a protocol based on that proposed by Needham and Schroeder.

Kerberos version 4

— makes use of DES

A simple authentication dialogue

— In a unprotected n/w environment any client can apply to any server for services their is an obvious security risk here is an opponent can pretend to be an client, to counter threat server can required to undertake this task for each client/server interaction but in open environment this places a substantial burden on each server

— An alternative is to use an authentication server (As) that knowns passwords of all users & stores these in a centralized database

— As shares a unique secret key with each server, these keys distributed physically in some other secure manner.

hypothetical dialogue

C → AS : $ID_c || P_c || ID_V$

AS → C : Ticket (user authentication) (ID, n/w addrs, server's ID)

C: client AS: authentication server V = server ID_c = identifier of user on c

IDV: identifier of V P_c = password of user on c

AD_c = n/w addr of c

K_v = secret encryption key shared by AS & V

|| = concatenation

3.43

- Ticket is encrypted using the secret key shared by AS & this server ticket is send to c bcz ticket is encrypted it can be altered by c with ticket c can apply to V for service, c sends msg to V
- Containing c's ID & ticket. V decrypts the ticket & verifies that the user ID in the ticket is same as the unencrypted user ID in the msg. if two match the server considers the user authenticated & grants the request service.

CSE-D 20/2/17 (Presenties)

5H, 5Q, R, T, 62, 65, 66, 68, 6D, 6E, 6G, H, P, 72, 79

CSE-B

2G, 39, 22, 2H, 2L, 2F, 1U, 3C, 1Z, 2W, 2N, 1X, 507, 524

3A, 509, 2Z, 36

A more Secure Authentication Dialogue

first problem is to minimize the number of times that a user has to enter a password, every time the user logs on it should enter password instead we can have for a single logon session the workstation can store the mail server ticket after it is received and use it on behalf of the user for multiple accesses to the mail server.

- new ticket for every different service
- second problem involved a plaintext transmission of the password opponent could capture the password and use any service

To solve we introduce a scheme for avoiding plaintext password and a new server known as ticket granting server (TGS).

once per user logon session

(clients requests TGT on behalf of user)

(1) C → AS : $ID_c || ID_{tgs}$

2) AS → C : $E_{Kc}[Ticket_{tgs}]$

once per type of service

(3) C → TGS : $ID_c || ID_v || Ticket_{tgs}$

(4) TGS → C : Ticket

once per service session

(5) C → V : $ID_c || Ticket_v$

$Ticket_{tgs} = E_{K_{tgs}}[ID_c || AD_c || ID_{tgs} || TS_1 || lifetime_1]$

$Ticket_v = E_{Kv}[ID_c || AD_c || ID_v || TS_2 || lifetime_2]$

- first user requests a ticket granting ticket from the AS. this ticket is saved by the client module in the user workstation each time the user requires access to new service. the client applies to the TGS using the ticket to authenticate itself. The TGS then grants a ticket for particular service the client saves each service granting ticket and uses it to authenticate its user to a server each time a particular service is requested.

(1) client requests a TGT on behalf of user by sending its user's ID to the AS along with TGS ID

... with a ticket that is encrypted with a key that is ... response

it prompts the user for psw generate key & attempt to decrypt the incoming msg, if correct psw is supplied the ticket is successfully recovered.

3.45

(3) client requests a service granting ticket on behalf of the user. for this client transmits a msg to the TGS containing the user's ID and TGT.

4) The TGS decrypts the incoming ticket and verifies the success of the decryption by presence of its ID. TGS compares the user ID and n/w address with the incoming info to authenticate the user if user permitted access to V the TGS issue a ticket to grant access to the ~~requires~~ requested service

5) The client requests access to a service on behalf of the user for this client transmits a msg to the server containing the user's ID and the service granting ticket, server authenticate by using the contents of the ticket

Kerberos realms & multiple Kerberi

A full service Kerberos environment consisting of a Kerberos server a number of clients & number of application servers requires the following.

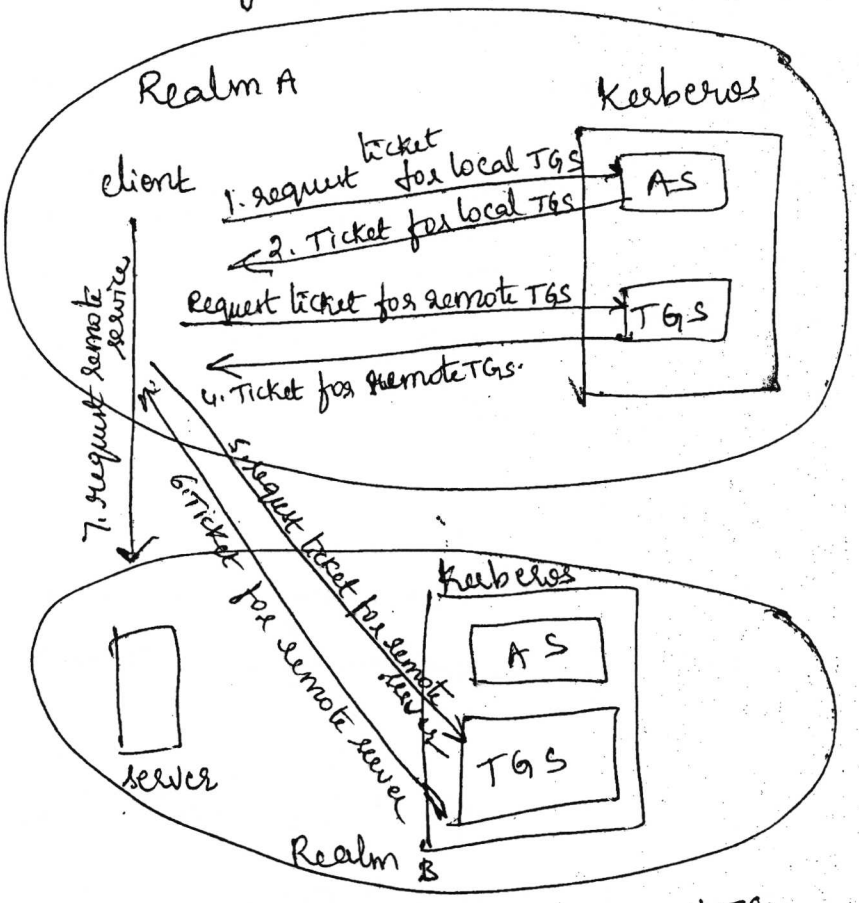
1. Kerberos server must have the user ID & hashed psw of all participating users in its database. All users are registered with the Kerberos server.
2. The Kerberos server must share a secret key with each server All servers are registered with the Kerberos server

Such environment is referred to as a realm. To have users & servers

- users in one realm may need access to servers in other realms and some servers may be willing to provide service to users from other realms provided that those users are authenticated.

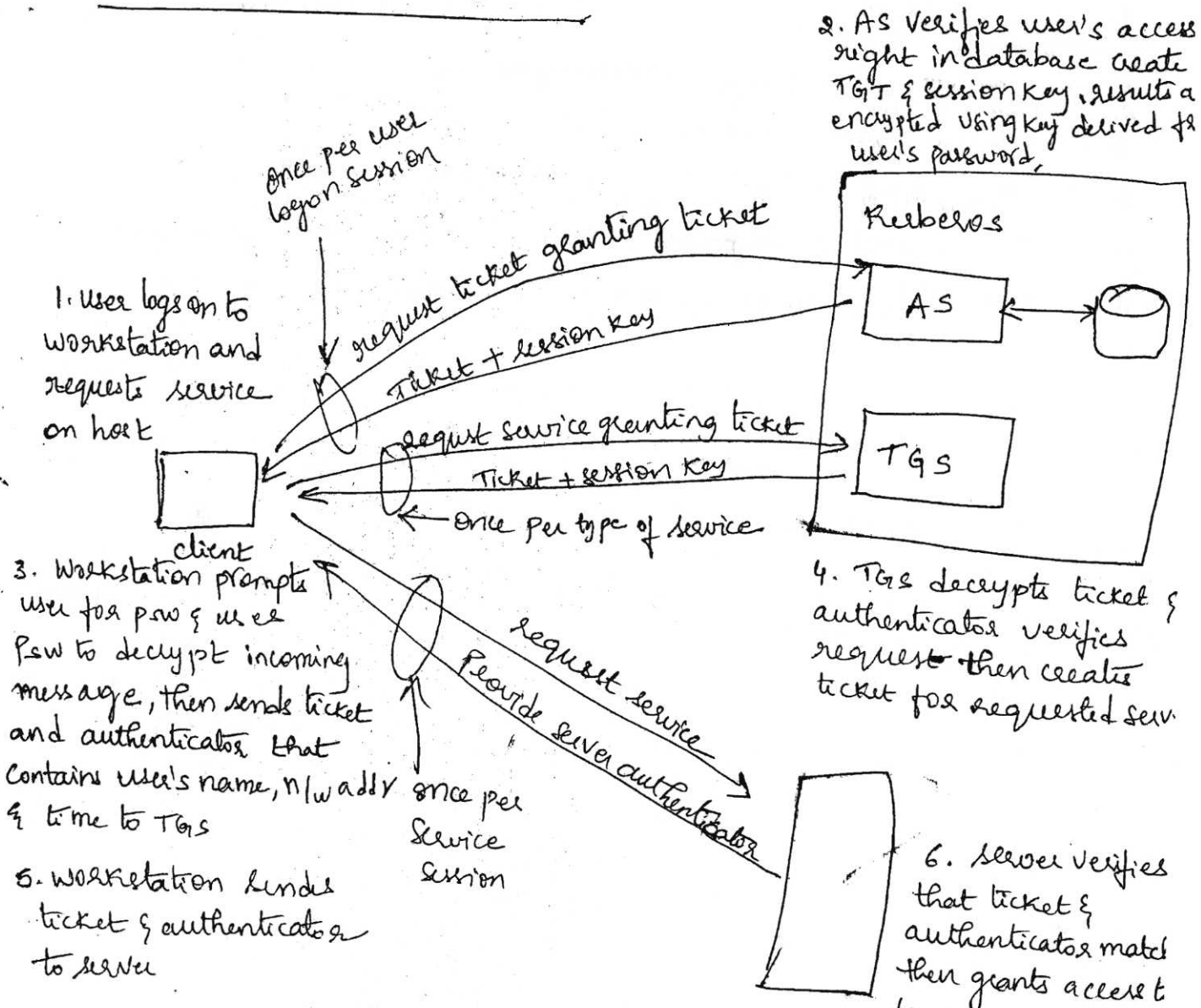
– Kerberos provide a mechanism for supporting such interrealm authentication for two realms to support inter realm authentication a third requirement is added

3. Kerberos server in each interoperating realm shares a secret key with the server in the other realm. two Kerberos servers are registered with each other.



1. $C \rightarrow AS$: $ID_c \| ID_{tgs} \| TS_1$
2. $AS \rightarrow C$: $E_{Kv}[K_{c,tgs} \| ID_{tgs} \| TS_2 \| lifetime_2 \| Ticket_{tgs}]$
3. $C \rightarrow TGS$: $ID_{tgsrem} \| Ticket_{tgs} \| authenticator_c$
4. $TGS \rightarrow C$: $E_{Kc,tgs}[K_{c,tgsrem} \| ID_{tgsrem} \| TS_4 \| Ticket_{tgsrem}]$
5. $C \rightarrow TGS_{rem}$: $ID_{vrem} \| Ticket_{tgsrem} \| authenticator$
6. $TGS \rightarrow C$: $E_{...}[K_{c,vrem} \| ID_{vrem} \| TS_6 \| Ticket_{vrem}$

Actual Kerberos overview



Summary of Kerberos Version 4 message exchanges

a) Authentication service exchange: to obtain ticket granting ticket

1) $C \rightarrow AS: ID_c \| ID_{tgs} \| TS_1$

2) $AS \rightarrow C: E_{K_c}[K_{tgs} \| ID_{tgs} \| TS_2 \| lifetime_2 \| ticket_{tgs}]$

$Ticket_{tgs} = E_{K_{tgs}}[K_{c,tgs} \| ID_c \| AD_c \| ID_{tgs} \| TS_2 \| lifetime_2]$

b) T-G-S exchange: to obtain service granting ticket

3) $C \rightarrow TGS: ID_v \| Ticket_{tgs} \| authenticator$

4) $TGS \rightarrow C: E_{K_{c,tgs}}[K_{c,v} \| ID_v \| TS_4 \| Ticket_v]$

$Ticket_{tgs} = E_{K_{tgs}}[K_{c,tgs} \| ID_c \| AD_c \| ID_{tgs} \| TS_2 \| lifetime_2]$

$Ticket_v = E_{K_v}[K_{c,v} \| ID_c \| AD_c \| ID_v \| TS_4 \| lifetime_4]$

$Authenticator = E_{K_{c,tgs}}[ID_c \| AD_c \| TS_3]$

c) Client/ Server ...

Kerberos version 5

1. Version 5 has a longer ticket lifetime
2. " allows tickets to be renewed
3. " can accept any symmetric key alg
4. " uses a different protocol for describing data types
5. " has more overhead than version 4.

CSC-B

521, 2B, 2K, 1X, 5N, 2E, 24
2A, 3B, 28, 2H, 37, 27, 29
22, 2F, 39, 3D

SHA-1 Compression function

each round is of the form

$$A,B,C,D,E \leftarrow (E + f(t,B,C,D) + S^5(A) + W_t + K_t),\ A,\ S^{30}(B),\ C,$$

A, B, C, D, E = five words of the buffer

t = step number ; $0 \leq t \leq 79$

$f(t,B,C,D)$ = primitive logical function for step t

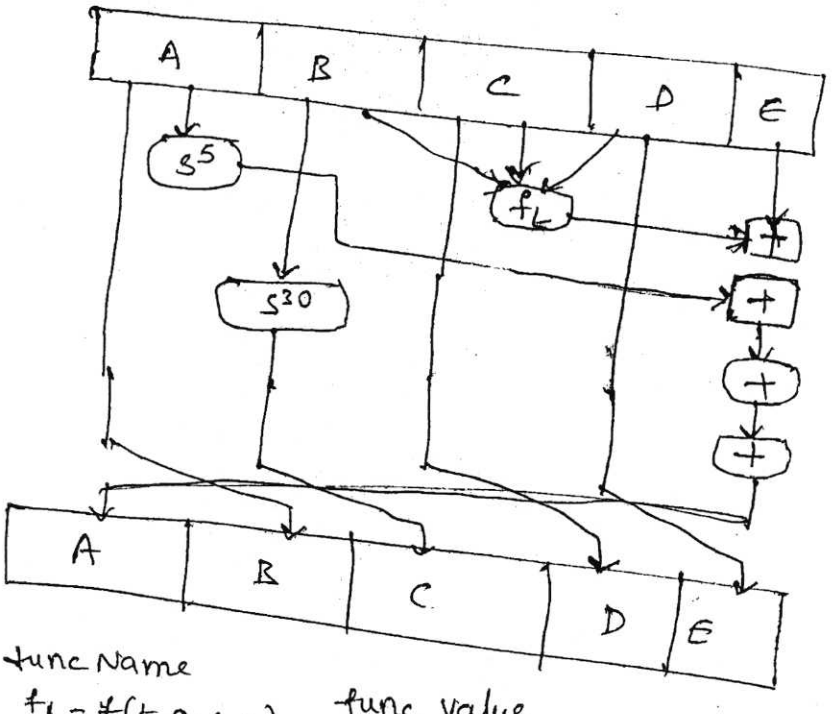
S^K = circular left shift of the 32-bit argument by K bits

W_t = a 32 bit word derived from the current 512 bit i/p

K_t = an additive constant ; four distinct values are used as defined previously.

$+$ = addition modulo 2^{32}

— each primitive func takes three 32 bit word as i/p & produces a 32 bit word o/p. each func performs a set of bitwise logical operations i.e the n^{th} bit of the o/p is a function of the n^{th} bit of the three i/p's



Step	Func Name	Func value
$(0 \leq t \leq 19)$	$f_1 = f(t,B,C,D)$	
$(20 \leq t \leq 39)$	$f_2 = f(t,B,C,D)$	$(B \wedge C) \vee (\bar{B} \wedge D)$
$(40 \leq t \leq 59)$	$f_3 = f(t,B,C,D)$	$B \oplus C \oplus D$

CSE-D

571, 564, 565, 56X, 52, 56B, 661, 55M, 55W, 74, 55P, 56N, 56D, 64, 550

SHA-1 (Secure hash function) 3.50

- Was developed by the National institute of standards & technology (NIST) and published as a federal information processing standard in 1993.
- SHA is based on the MD4 algorithm and its design closely models MD_4
- It takes input a msg with a maximum length of less than 2^{64} bits and produced as output a 160 bit msg digest, i/p is processed in 512 bit blocks
- Overall processing of a msg follows the structure shown for MD5 with a block length of 512 bits and a hash length and chaining variable length of 160 bits

Step1 : Appending padding bits : msg is padded so that its length is congruent to 448 modulo 512 ∴ no of padding bits is in the range of 1 to 512

Step 2 : Append length : A block of 64 bits is appended to the msg

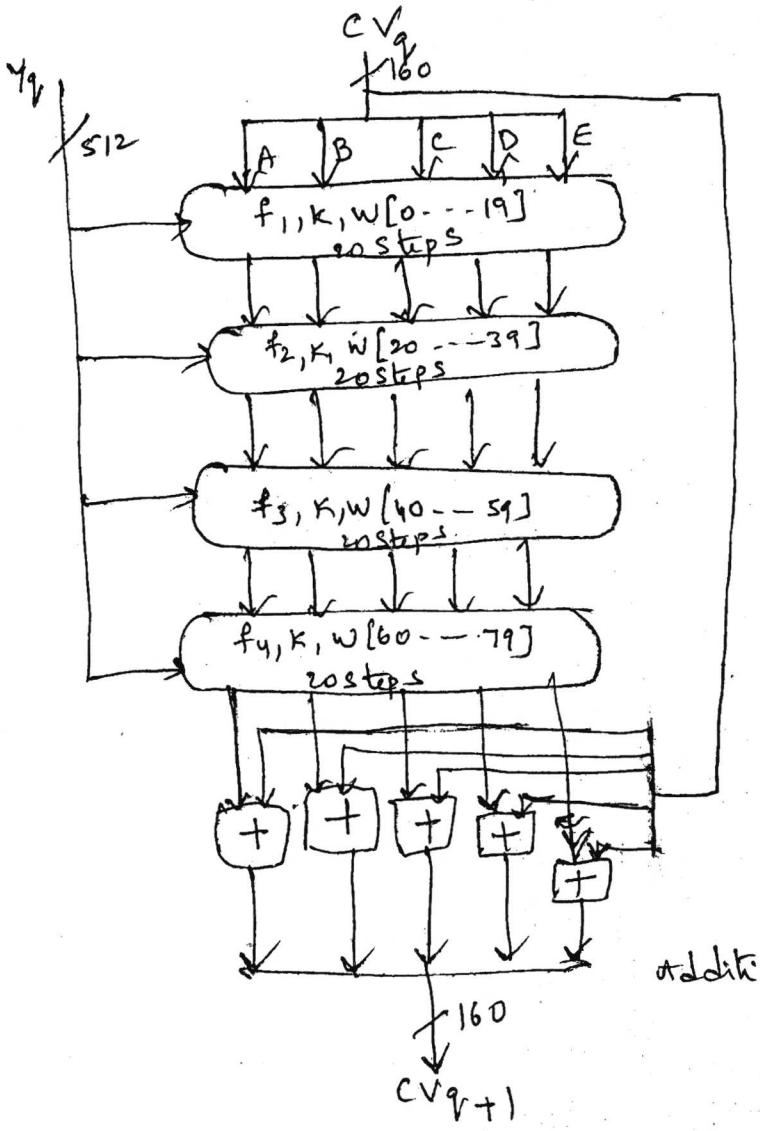
Step 3 : Initialize MD buffer : A 160 bit buffer is used to hold intermediate and final results of the hash function. buffer can be represented as five 32-bit registers (A, B, C, D, E) these registers are initialized to the following 32 bit integers

32 bit integers	32 bit string initilization Values appear as follows
A = 67452301	word A = 67 45 23 01
B = EFCDAB89	word B = EF CD AB 89
C = 98BADCFE	word C = 98 BA DC FE
D = 10325476	word D = 10 32 54 76
E = C3D2E1F0	word E = C3 D2 E1 F0

Step 4 : process msg in 512 bit blocks : heart of alg is a module Consists of four rounds of processing of 20 steps each, four rounds m'ly structure but each uses a diff primitive logical function we refer as f_1, f_2, f_3 & f_4

- each round takes as input the current 512 bit block being processed (Y_q) & the 160 bit buffer value ...

3.51



addition (+) is mod 2^{32}

Each round makes use of an additive constant K_t where $0 \leq t \leq 79$ indicates one of the 80 steps across five rounds

- O/p of 4th round added as i/p to first round (CV_q) to produce CV_{q+1} addition is done independently for each of the five words in the buffer with each of the corresponding word in CV_q using addition modulo 2^{32}

Steps output After all L 512-bit block have been processed the o/p from the Lth stage is the 160 bit msg digest.

we summarize the behavior of SHA-1 as follows

$$CV_0 = IV$$
$$CV_{q+1} = SUM_{32}(CV_q, ABCDE_q)$$
$$MD = CV_L$$

where IV = initial value of the ABCD buffer

ABCDE

E-mail & IP Security

Pretty Good Privacy (PGP)

Operational Description

The PGP has five services

1. authentication
2. confidentiality
3. compression
4. e-mail compatability
5. segmentation.

Authentication

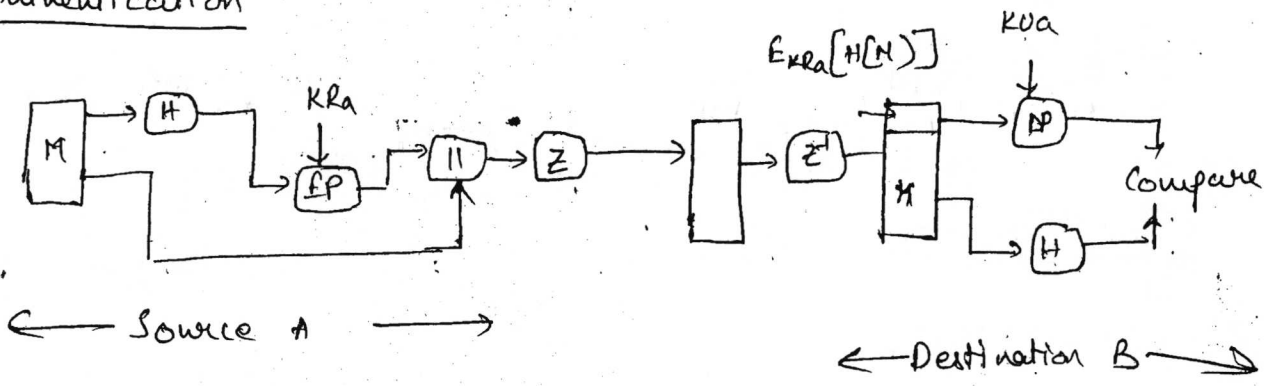


Figure describes the digital signature service provided by PGP. The sequence is as follows.

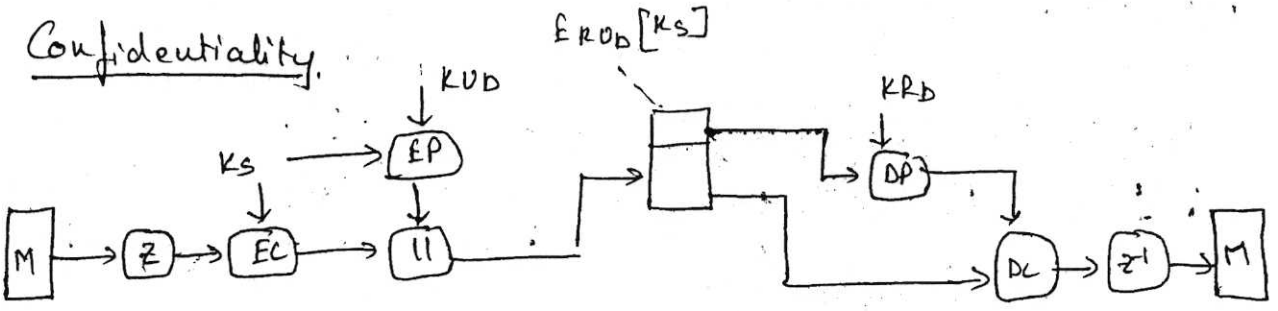
1) The sender creates a message.
2) SHA-1 is used to generate a 160-bit hash code of the message.
3) The hash code is ~~generated~~ encrypted with RSA using the sender's private key and the result is prepended to the message.

4) The receiver uses RSA with the sender's public key to decrypt and recover the hash code.

5) The receiver generates a new hash code for the message and compares it with the decrypted hash code. If the two match, the message is accepted as authentic.

In PGP, the combination of SHA-1 and RSA provides an effective digital signature.

Confidentiality.



The confidentiality is provided by PGP by encrypting messages to be transmitted or to be stored locally as files.

For this PGP uses CAST-128 encryption algorithm (or) IDEA (or) 3DES algorithms, in CFB mode.

The figure describes how PGP offers confidentially in the following steps.

① The sender generates a message and a random 128-bit number to be used as a session key for this message only.

② The message is encrypted, using CAST-128 (or IDEA or 3DES) with the session key.

③ The session key is encrypted with RSA, using receipient's public key, and is prepended to the message.

④ The receiver uses RSA with its private key to decrypt and recover the session key.

⑤ The session key is used to decrypt the message.

Compression

PGP compresses the message after applying signature but before encryption.

The compression function (Z) and Z^{-1} for decompression

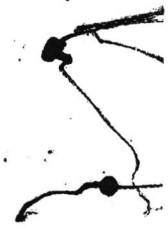
The compression used is ZIP.

A signature is generated before compression for

1) It is preferable to sign an uncompressed message so that one can store only uncompressed message together

together with signature for future verification.

2) Message encryption is applied after encryption to strengthen cryptographic security.



E-mail compatability

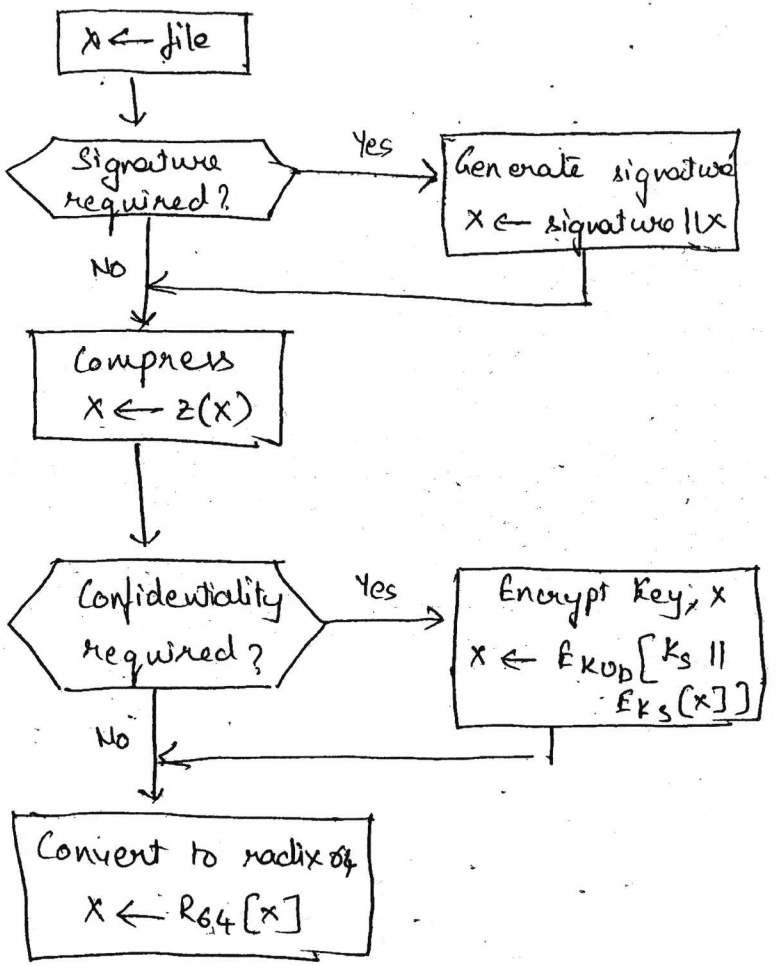
The electronic mail systems only permit the use of blocks consisting of ASCII text. PGP provides the service of converting the raw 8-bit binary stream to a stream of printable ASCII characters.

→ The scheme used to do this is radix 64 conversion.

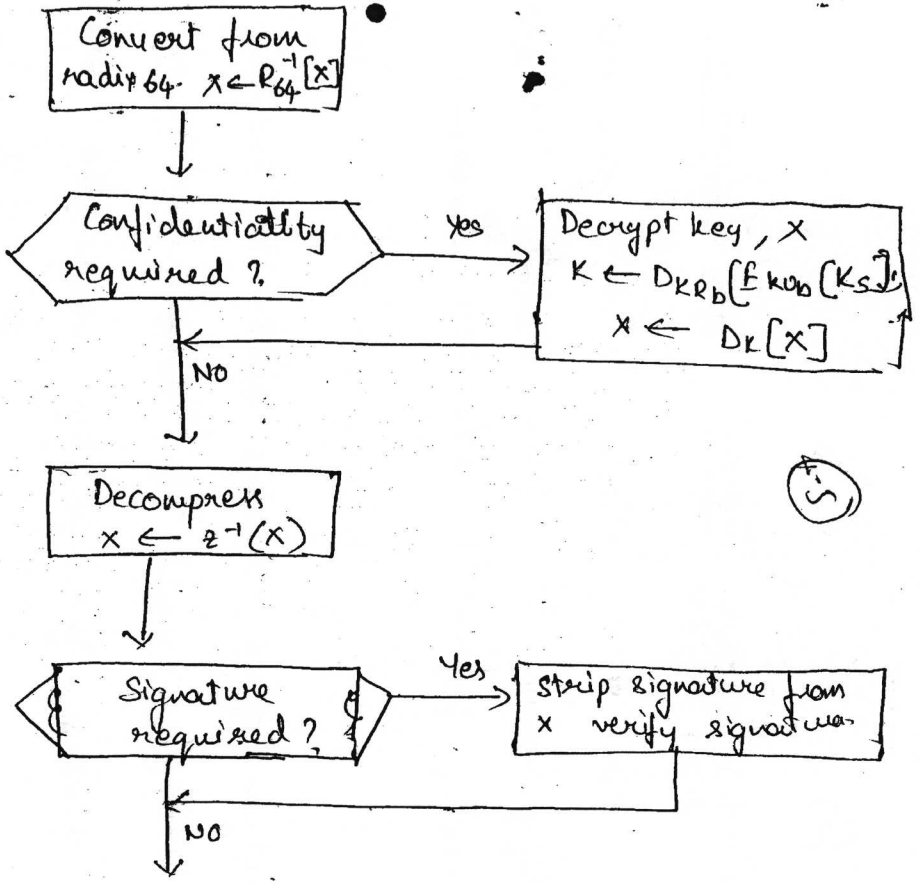
→ Each group of three octets of binary data is mapped into four ASCII characters. This format also appends a CRC to detect transmission errors.

Segmentation and Reassembly

PGP automatically subdivides a message that is too large into segments that are small enough to send via e-mail. The following figure shows the procedure of ~~segmentation and reassembly in PGP~~ transmission and reception of messages in PGP.



(fig) Transmission of messages in PGP



(fig) Reception of messages in PGP.

S/MIME (Secure/Multipurpose Internet Mail Extension)

- It is an extension of MIME internet e-mail standard, based on technology from RSA Data Security. The background of traditional e-mail format is RFC 822 standard.

RFC 822

→ The overall structure of a message ~~that~~ in RFC 822 is very simple. A message consists of some number of header lines (the header) followed by unrestricted text (the body). The header is seperated from the body by a blank line.

→ A header line usually consists of a blank line followed by a colon, followed by the keywords arguments.

→ The most frequently used keywords are From, To, Subject and Date.

e.g.; Date: Tue, 16 Jan 1998 10:37:17 (EST)
From: "william stallings" <ws@shore.net>
Subject: The Syntax in RFC 822
To: Smith@other-host.com
Cc: Jones@Yet-Another-host.com
← Blank line.
Hello, This section begins the actual message body.

MIME :- The MIME specification includes the following contents.

MIME Header fields

MIME-Version :- Must have a parameter value 1.0.

Content-Type :- Describes the data contained in the body with sufficient detail that the receiving user agent can pick an appropriate agent to represent data.

Content-Transfer-Encoding :- indicates the type of transformation that has been used to represent the body of the message in a way that is acceptable for mail transport.

Content-ID :- Used to identify MIME entities uniquely in multiple contexts.

Content-Description :- A text description of the object with the body ; this is useful when the object is not readable (e.g. audio data)

MIME Content Types

Type	SubType	Description
Text	Plain	Unformatted text (ASCII)
	Enriched	Provides greater format flexibility.

Type	Sub-type	Description
Multipart	Mixed	The different parts are independent but are to be transmitted together.
	Parallel	Differs from mixed only in that no order is defined for delivering the parts.
	Alternative	The different parts are alternative versions of the same information.
	Digest	Similar to mixed, but the default type/sub-type of each part is message.
Message	rfc822	The body is itself an encapsulated message that conforms to RFC 822
	Partial	Used to allow fragmentation of lar mail items, in a way that is transpa to the receiptant.
	External-body	Contain a pointer to a object that exists elsewhere.
Image	jpeg	The image is in JPEG format
	gif	The image is in GIF format.
Video	mpeg	MPEG format.
Audio	basic	Single-channel 8-bit ISDN

Application PostScript Adobe Postscript

Octet-Stream General binary data consisting of 8-bit bytes.

MIME Transfer Encodings

7 bit The data are all represented by short lines of ASCII characters.

8 bit The lines are short, but there may be non-ASCII characters (octets with the high-order bit set)

binary Not only may non-ASCII characters be present, but the lines are not necessarily short enough for SMTP transport.

quoted-printable. Encodes the data in such a way that if the data being encoded are mostly ASCII text, the encoded form of

base 64 Encodes data by mapping 6-bit blocks of input to 8-bit blocks of output, all of which are printable ASCII characters.

x-token A named nonstandard encoding.

Example of MIME message structure

MIME-Version : 1.0
From : NM <nm @ bell.com>
To : AK <ak @ imsoft.com>
Subject : A multipart example.
Content-type : multipart / mixed
boundary : unique-boundary-1.

This is a preamble of the multipart message

-- unique-boundary-1

Some text appears here

-- unique-boundary-1

S/MIME Functionality

S/MIME provides the following functions.

- Enveloped data : This consists of encrypted content of any type and encrypted-content encryption keys for one or more receipients.

- Signed data :- A digital signature is formed by taking the message digest of the content to be signed, and encrypting that with the private key of the signer. The content and plus signature are then encoded using base64 encoding.

Clear-signed data :- In this case only the digital signature is encoded using base 64. As a result, recipients without S/MIME capability can view the message content, although they cannot verify the signature.

Signed and enveloped data:- Signed-only and encrypted-only entities may be nested, so that encrypted data may be signed and signed-data (or) clear-signed data may be encrypted.

Cryptographic algorithms used in S/MIME

MUST: The definition is an absolute requirement of the specification. An implementation must include this feature or function to be ~~incom~~ in conformance with the specification.

SHOULD: There may exist valid reasons in particular circumstances to ignore this feature or function, but it is recommended the implementation include the feature or function.

Function	Requirement
Create a message digest to be used in forming a digital signature	MUST support SHA-1 Receiver SHOULD support MD5 for backward compatibility.

Function	Requirement
Encrypt message digest to form digital signature	

The services are as follows

	AH	ESP (encryption only)	ESP (encryption plus authentication)
Access control	✓	✓	✓
Connectionless integrity	✓		✓
Data origin authentication	✓		✓
Rejection of replayed packets	✓	✓	✓
Confidentiality		✓	✓
Limited traffic flow confidentiality		✓	✓

Security associations

- Association is a one way relationship b/w a sender & a receiver that affords security services to the traffic carried on it.
- If peer relationship is needed two-way secure exchange then two security associations are required

Security associations are uniquel identified by three parameters

- Security parameter index (SPI): A bit string assigned to this SA and having local significance only. SPI is carried in AH and ESP headers to enable receiving system to select the SA under which received packet is processed
- IP destination addr: addr of destination endpoint of the SA
- Security protocal identifier: Indicates whether association is an AH or ESP

SA parameters

- Sequence number counter: A 32 bit value used to generate the sequence num field in AH or ESP
- Sequence counter overflow: flag indicating whether overflow of the

Anti Replay window: determine whether an inbound AH or ESP packet is a replay.

AH information: authentication alg, keys, keys lifetimes & related parameters

ESP info: encryption & authentication alg, keys, initialization values, key lifetime

lifetime of this security associations: Time interval or byte counter after which an SA must be replaced with a new SA

IPsec protocol mode: Tunnel or Transport or wildcard

Path MTU: An observed path max transmission unit

SA Selectors

IP traffic is related to specific SAs in the security policy database (SPD) Contains entries each defines a subset of IP traffic & points to an SA for that traffic

- each SPD entry is defined by a set of IP & upper layer protocol field value called selectors
- these selectors are used to filter outgoing traffic in order to map it into a particular SA

following selectors determine an SPD entry

1) destination IP addr
2) src IP addr
3) user ID
4) data sensitivity level: (secret info flow)
5) Transport layer protocol:
6) IPsec protocol (AH or ESP or AH/ESP)
7) src & dest ports
8) IPV6 class
9) IPV6 flow label
10) [illegible]

Transport and Tunnel mode

Transport mode : provides protection primarily for upper layer protocols Protection also extends to the payload

- This mode is used for end-to-end communication b/w two hosts When host runs AH or ESP over IPV4 the payload is the data that normally follow IP header . for IPV6 payload is the data that follow the IP header and any IPV6 extension headers
- ESP in transport mode encrypts & optionally authenticates the IP Payload but not the IP header.
- AH in transport mode authenticates the IP payload and selected portions of the IP header

Tunnel mode

- Provides protection to the entire IP packet
- To achieve this after AH ~~and~~ or ESP fields are added to the IP packet entire packet plus security fields is treated as the payload of new outer IP packet with a new outer IP header
- entire inner packet travels through a tunnel from one point to an IP network to another no routers along way can examine the inner ~~Packet~~ IP header bez original packet (inner packet) is encapsulated the new larger packet may have totally diff Src & destination address .
- Tunnel mode is use when one or both ends of an security ~~associat~~ associations is a security gateway such as firewall or router that implements IPsec.

ex : Host A generates an IP packet with the destination addr of host B on another n/w , this packet is routed from originating host to a firewall at the boundary of A's n/w , this firewall filters all outgoing packets to determine need of IPsec processing . if Packet

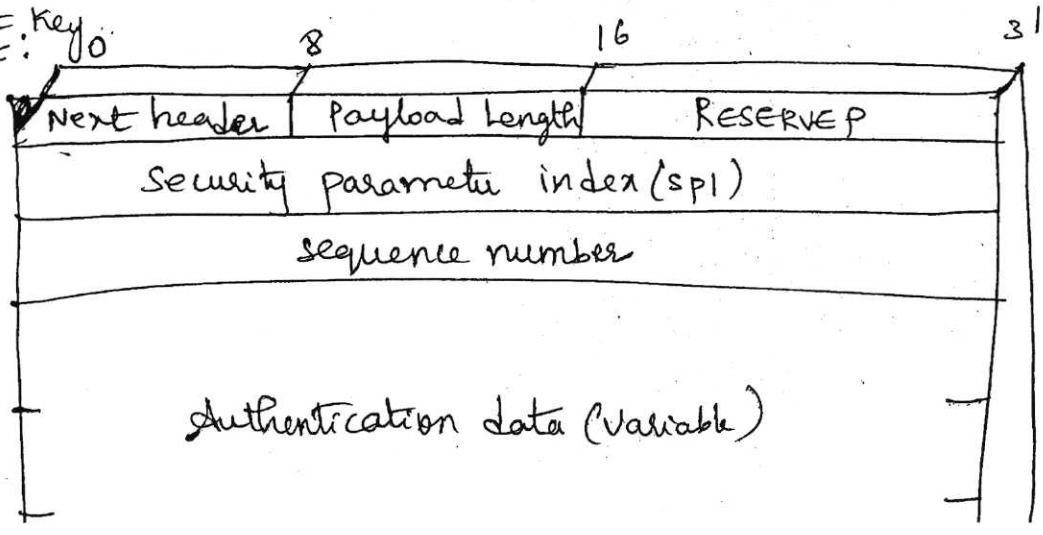
the packet with an outer ip header (src ip addr of this outer ip packet is this firewall) & destination addr may be firewall that forms the boundary to B's local n/w, this packet now routed to B's firewall., At B's firewall the outer ip header is stripped off & the inner packet is delivered to B.

Esp in tunnel mode encrypts and optionally authenticates the entire inner ip packet including the inner ip header.

AH in tunnel mode authenticates the entire inner ip packet & selected Portions of the outer ip header

Authentication Header

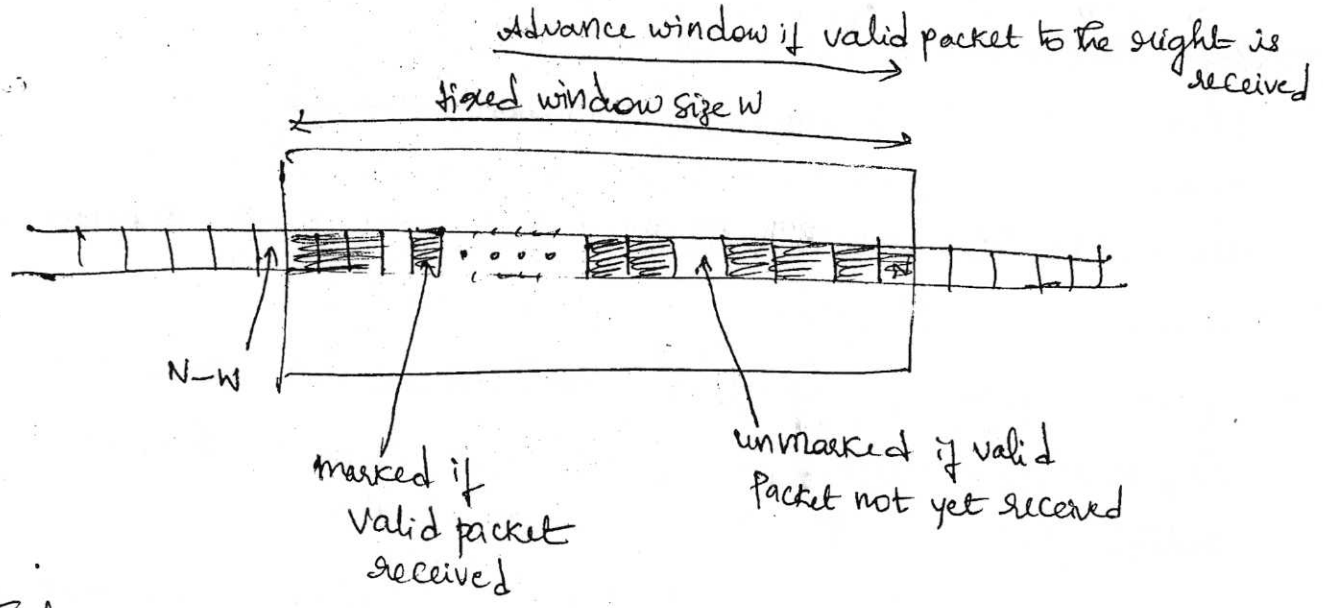
- provides support for data integrity and authentication of ip packets
- data integrity feature ensures that undetected modification to a packet's content in transit is not possible.
- authentication feature enables end system or n/w device to authenticate the user or application & filter traffic accordingly it also prevents addr spoofing attack.
- AH also guards against replay attack.
- authentication is based on HAC. hence two parties must share a secret key



- any packet with a sequence number in the range from N-W+1 to N that has been correctly received the corresponding slot in the window is marked

Inbound processing proceeds as follows

1. If received packet falls within window & new the MAC is checked if the packet is authenticated the corresponding slot is marked
2. If received packet is to the right of the window and is new the MAC is checked. If the packet is authenticated the window is advanced so that seq num is the right edge of window. and corresponding slot is marked
3. If received packet is to left to the window or if authentication fails the packet is discarded : this is auditable event



Integrity check value

Authentication data field holds a value referred as integrity check value. The ICV is a msg authentication code or trusted version of a code produced by a MAC algorithm.

- HMAC-MD5-96 } both uses HMAC alg first with MD_5 hash code &
  HMAC-SHA-1-96 } second with SHA-1 hash code. for both full HMAC value is calculated but then truncated by using

Next header (8 bits): identify type of header immediately following this header

Payload length (8bit): length of authentication header in 32 bit words minus 2

Reserved (16 bits): for future use

Security parameter index (32 bits): identifies a security association

Sequence Number (32 bits): A monotonically increasing counter value

Authentication data: A variable length field that contains the integrity check value (ICV) or MAC for this packet.

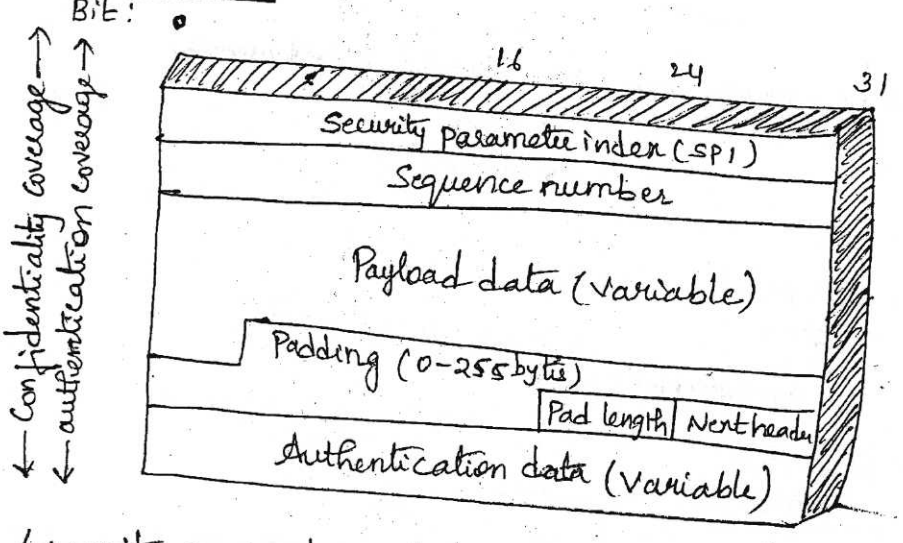
Anti Replay service

- A replay attack is one in which an attacker obtain a copy of an authenticated packet & later transmits it to the intended destination
- how sequence number is generated by the sender.
  When an new SA is established the sender initializes a sequence number counter to 0. each time that a pack is sent on this SA the sender increments the counter & places the value in sequence number field. thus first value to be used is 1
- if anti replay is enabled the sender must not allow the seq number to cycle $2^{32}-1$ back to zero otherwise there would be multiple packets with the same sequence number.
- if the limit $2^{32}-1$ is reached the sender should terminate this SA with a new key.
- bcz IP is an connectionless unreliable service the protocol does not guarantee that packets will be delivered in order and does not guarantee that all packets will be delivered
- therefore ipsec authentication document at receiver side implement a window size W (default W=64)

Encapsulating security payload

This payload provides confidentiality services

Esp format



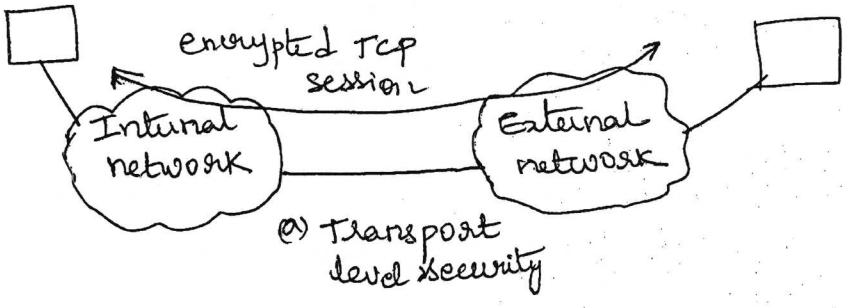
- Security parameter index (32 bits) : identifies a security association
- Sequence number (32-bits) : increasing counter value, it provides an anti replay function
- Payload data (Variable) : transport level segment or IP packet that is protected by encryption.
- Padding (0-255) : if an encryption algorithm requires the plaintext to be a multiple of some number of bytes the padding field is used to expand the plaintext to the required length.
- The Esp format requires that the pad length and next header fields be right aligned within a 32 bit word
- additional padding may be added to provide partial traffic flow confidentiality

- Next header (8 bits) : identifies the type of data contained in the Payload data field by identifying the first header in that payload
- Authentication data : Variable length field contains the ICV Computed over the Esp packet minus the authentication data field

Encryption and authentication algorithms

- Three-key triple DES
- RC5
- IDEA
- Three-key triple IDEA
- CAST
- Blowfish

Transport and Tunnel mode



(a) Transport level security

Transport mode ESP

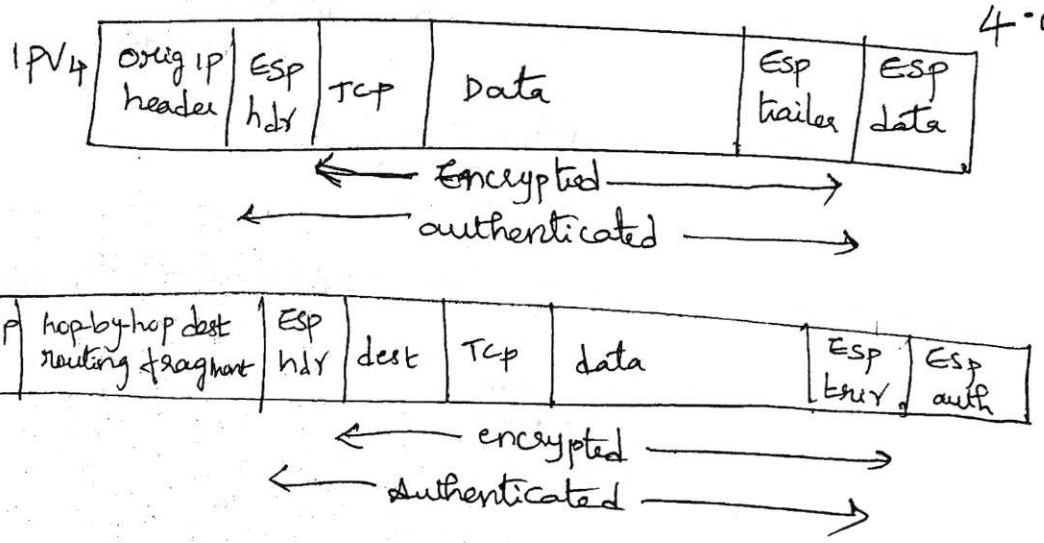
is used to encrypt and optionally authenticate the data carried by IP

Using IPV4

ESP header is inserted into the IP packet immediately prior to the transport layer header & an ESP trailer is placed after the IP packet, if authentication is selected the ESP authentication data field is added after the ESP trailer. entire transport layer segment plus the ESP header are encrypted.

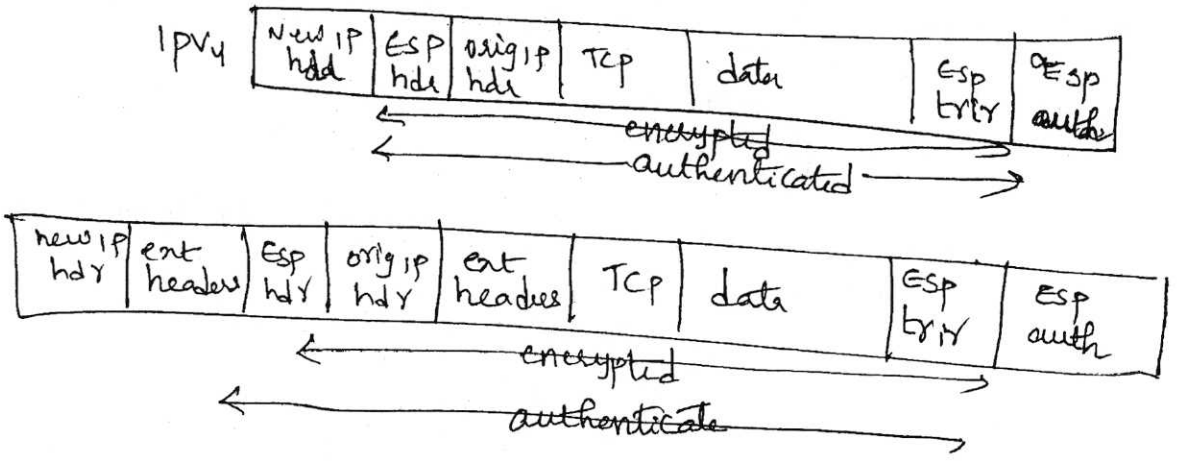
Using IPV6

ESP is viewed as end-to-end payload, it is not examined by intermediate routers, therefore the ESP header appears after the IPV6 base header and the hop-by-hop routing & fragment extension headers. destination options extension header could



Tunnel mode : is used to encrypt an entire IP packet, for this mode Esp header is prefixed to the packet and then the packet plus the Esp trailer is encrypted.

— because the IP header contains the destination addr & possibly source routing directives and hop-by-hop option information it is not possible simply to transmit the encrypted IP packet prefixed by the Esp header. intermidate routers would be unable to process such a packet therefore it is necessary to encapsulate the entire block with a new IP header that will contain sufficient information for routing but not for traffic analysis



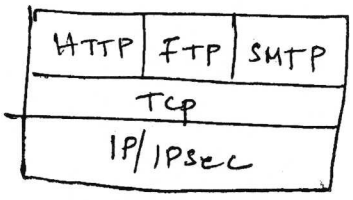
UNIT-V

Web security considerations threats

	Threats	consequences	countermeasures
Integrity	• modification of user data • Trojan horse browser • modification of message traffic in transit	• loss of info • compromise of machine • vulnerability to all other threats	cryptographic checksums
Confidentiality	• eavesdropping on the net • Theft of info from server • Theft of data from client • info abt n/w config • info abt which client talks to server	• loss of info • Loss of privacy	encryption, web proxies
Denial of service	• Killing of user threats • flooding m/c with bogus request • filling up disk or mem • Isolating m/c by DNS attacks	• Disruptive • Annoying • Prevent user from getting work done	• difficult to prevent
Authentication	• Impersonation of legitimate ~~data~~ users • Data forgery	• Misrepresentation of user • Belief that false info is valid	Cryptographic techniques.

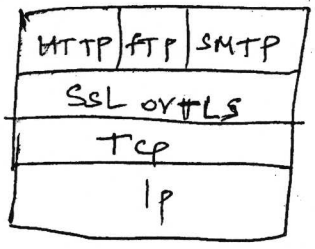
Web traffic security approaches

- One way is to use ipsecurity



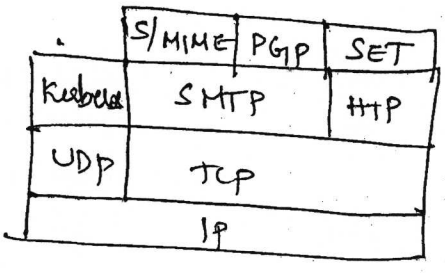
adv of using ipsec is that it is transparent to end users and applications to provide a general purpose solution

- Implement security just above TCP

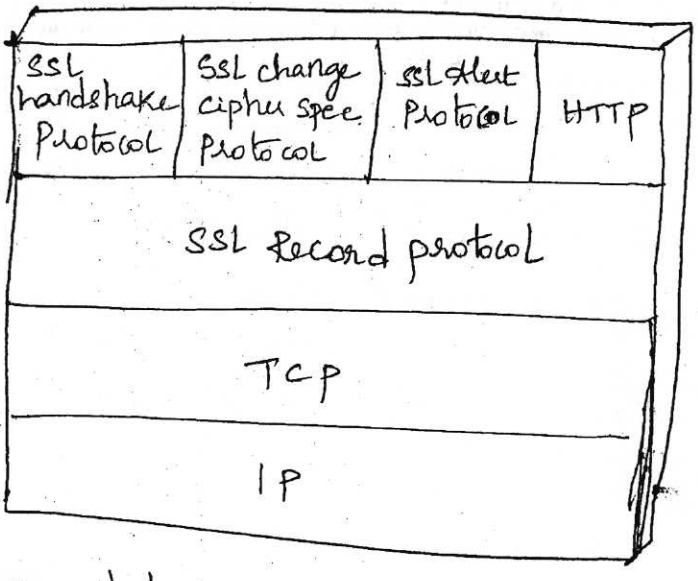


Secure socket layer and the follow on internet standard known as Transport layer security, At this level two implementation choices SSL or TLS ex: Netscape & microsoft explorer.

- Application specific security services are embedded within the particular application.. adv of this approach can be tailored to the specific needs of a given application ex: SET (secure electronic transaction)



SSL architecture

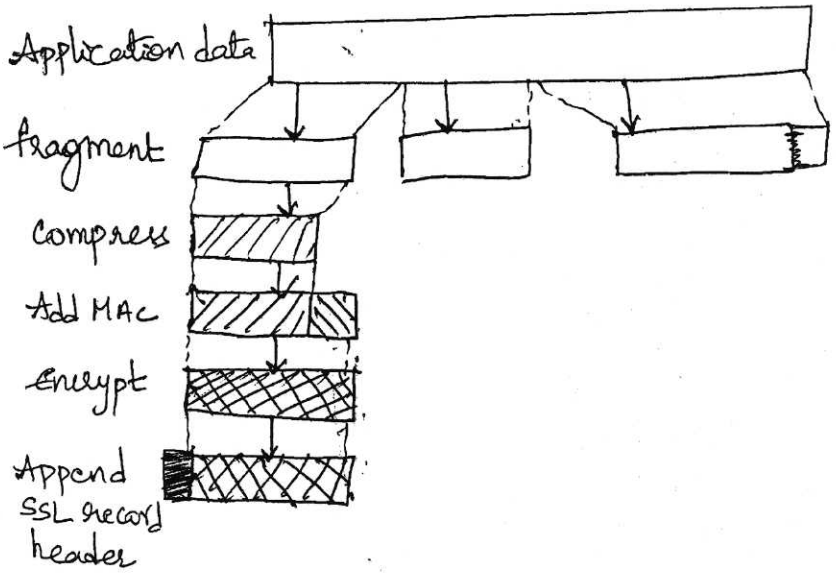


- SSL is designed to make use of TCP to provide a reliable end-to-end secure service
- SSL is not a single protocol but rather two layers of protocols
- SSL Record protocol provides basic security services to various higher layer protocols. particularly HTTP which provides the transfer service for web client/server interaction can operate on top of SSL
- Three higher level protocols are defined as part of SSL the handshake protocol, the change cipher spec protocol, the alert Protocol.
- Two important SSL concepts are the SSL session & the SSL connection

Connection: is a transport that provides a suitable type of service, for SSL such connections are peer-to-peer relationships Connection are transient. every connection is associated with one session

Session: is an association b/w a client & a server sessions are created by handshake protocol, session define a set of Cryptographic security parameters which can be shared among multiple connections

SSL Record protocol



The record protocol takes an application message to be transmitted fragments the data into manageable blocks, optionally compresses the data applies a MAC encryption adds a header & transmits the resulting unit in a TCP segment received data are decrypted verified, decompressed and reassembled and then delivered to higher level users.

- first step is fragmentation: each upper layer is fragmented into blocks of 2^{14} bytes or less Next Compression is optionally applied, Compression must be lossless & may not increase the content length by more than 1024 bytes In SSLV3 no Compression alg is specified so default compression alg is null next step is to compute a MAC over Compressed data for this a shared key is used calculation is defined as

hash(MAC_write_secret || pad_2 || hash (MAC_write_secret || pad_1 || seq_num || SSLCompressed.type || SSLCompressed.type || SSLCompressed.length || SSLCompressed.fragment))

where || = concatination

MAC-Secret-Key = Shared secret key

hash = cryptographic hash alg either MD5 or SHA-1

Pad-1 = the byte 0X36 (00110110) repeat 48 times (384 bits) for MD_5 and 40 times (320 bits) SHA-1

Pad-2 = the byte 0X5C (0101 1100) repeated 48 times for MD_5 & 40 times for SHA-1

Seq-num = the seq num for this msg

SSLcompressed.type = high level protocol used to process this fragmentation

SSLcompressed.length = length of the compressed fragment

SSLcompressed.fragment = the compressed fragment.

- Next compressed msg plus the HAC are encrypted using symmetric encryption; it may not increase content length by more than 1024 bytes so that total length may not exceed $2^{14}+2048$

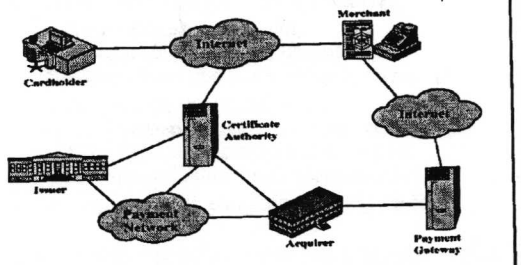
for encryption alg are permitted

Block cipher		Stream cipher	
Algorithm	Key Size	Algorithm	Key Size
· AES	128,256	RC4-40	40
IDEA	128	RC4-128	128
RC2-40	40		
DES-40	40		
DES	56		
3DES	168		
fortezza	80		

Secure Electronic Transactions (SET)

- open encryption & security specification
- to protect Internet credit card transactions
- developed in 1996 by Mastercard, Visa etc
- not a payment system
- rather a set of security protocols & formats
  - secure communications amongst parties
  - trust from use of X.509v3 certificates
  - privacy by restricted info to those who need it

SET Components



SET Transaction

1. customer opens account
2. customer receives a certificate
3. merchants have their own certificates
4. customer places an order
5. merchant is verified
6. order and payment are sent
7. merchant requests payment authorization
8. merchant confirms order
9. merchant provides goods or service
10. merchant requests payment

Dual Signature

- customer creates dual messages
  - order information (OI) for merchant
  - payment information (PI) for bank
- neither party needs details of other
- but **must** know they are linked
- use a dual signature for this
  - signed concatenated hashes of OI & PI

$DS=E_{KRc}[H(H(PI) \| H(OI))]$

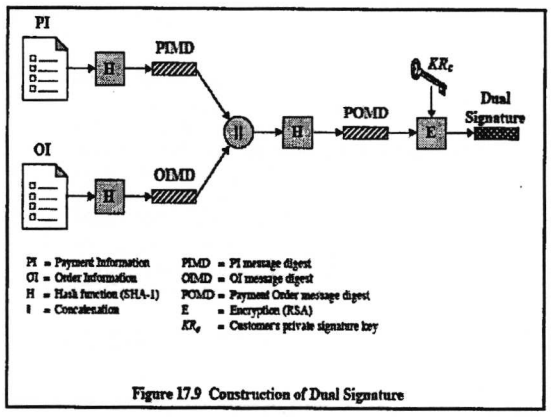
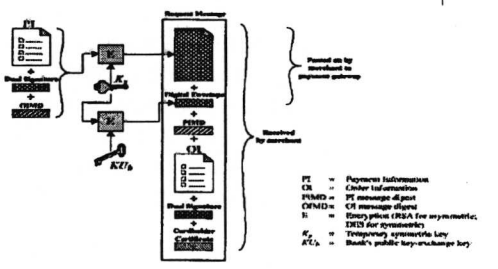
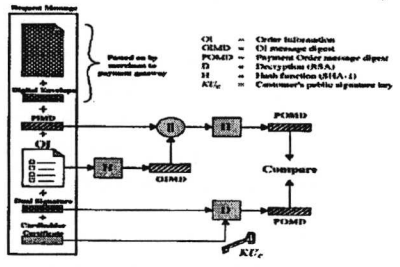


Figure 17.9 Construction of Dual Signature

Purchase Request – Customer



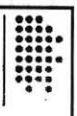
Purchase Request – Merchant



Purchase Request – Merchant

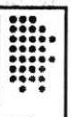
1. verifies cardholder certificates using CA sigs
2. verifies dual signature using customer's public signature key to ensure order has not been tampered with in transit & that it was signed using cardholder's private signature key
3. processes order and forwards the payment information to the payment gateway for authorization (described later)
4. sends a purchase response to cardholder

Payment Gateway Authorization



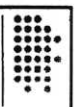
1. verifies all certificates
2. decrypts digital envelope of authorization block to obtain symmetric key & then decrypts authorization block
3. verifies merchant's signature on authorization block
4. decrypts digital envelope of payment block to obtain symmetric key & then decrypts payment block
5. verifies dual signature on payment block
6. verifies that transaction ID received from merchant matches that in PI received (indirectly) from customer
7. requests & receives an authorization from issuer
8. sends authorization response back to merchant

Payment Capture



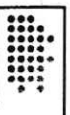
- merchant sends payment gateway a payment capture request
- gateway checks request
- then causes funds to be transferred to merchants account
- notifies merchant using capture response

Intruders



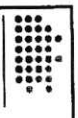
- significant issue for networked systems is hostile or unwanted access
- either via network or local
- can identify classes of intruders:
  - masquerader
  - misfeasor
  - clandestine user
- may seem benign, but still cost resources
- varying levels of competence

Intrusion Techniques



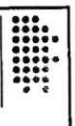
- aim to increase privileges on system
- basic attack methodology
  - target acquisition and information gathering
  - initial access
  - privilege escalation
  - covering tracks
- key goal often is to acquire passwords
- so then exercise access rights of owner

Password Guessing



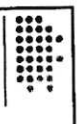
- one of the most common attacks
- attacker knows a login (from email/web page etc)
- then attempts to guess password for it
  - try default passwords shipped with systems
  - try all short passwords
  - then try by searching dictionaries of common words
  - intelligent searches try passwords associated with the user (variations on names, birthday, phone, common words/interests)
  - before exhaustively searching all possible passwords
- check by login attempt or against stolen password file
- success depends on password chosen by user
- surveys show many users choose poorly

Password Capture



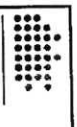
- another attack involves **password capture**
  - watching over shoulder as password is entered
  - using a trojan horse program to collect
  - monitoring an insecure network login (eg. telnet, FTP, web, email)
  - extracting recorded info after successful login (web history/cache, last number dialed etc)
- using valid login/password can impersonate user
- users need to be educated to use suitable precautions/countermeasures

Intrusion Detection



- inevitably will have security failures
- so need also to detect intrusions so can
  - block if detected quickly
  - act as deterrent
  - collect info to improve security
- assume intruder will behave differently to a legitimate user
  - but will have imperfect distinction between an attack and normal use of resources

Approaches to Intrusion Detection



- statistical anomaly detection
  - threshold: events frequency, independent of user
  - profile based: a profile of activity for each user
- rule-based detection
  - anomaly: based on usage pattern
  - penetration identification: using expert systems
- SAD: to define normal behavior
  RBD: to define improper behavior

Audit Records

- fundamental tool for intrusion detection
- native audit records
  - part of all common multi-user O/S
  - already present for use
  - may not have info wanted in desired form
- detection-specific audit records
  - created specifically to collect wanted info
  - at cost of additional overhead on system

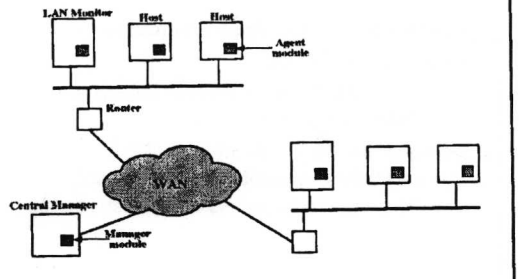
Base-Rate Fallacy

- practically an intrusion detection system needs to detect a substantial percentage of intrusions with few false alarms
  - if too few intrusions detected -> false security
  - if too many false alarms -> ignore / waste time
- this is very hard to do
- existing systems seem not to have a good record

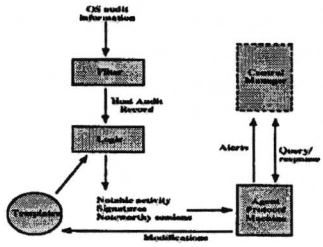
Distributed Intrusion Detection

- traditional focus is on single systems
- but typically have networked systems
- more effective defense has these working together to detect intrusions
- issues
  - dealing with varying audit record formats
  - integrity & confidentiality of networked data
  - centralized or decentralized architecture

Distributed Intrusion Detection - Architecture



Distributed Intrusion Detection – Agent Implementation

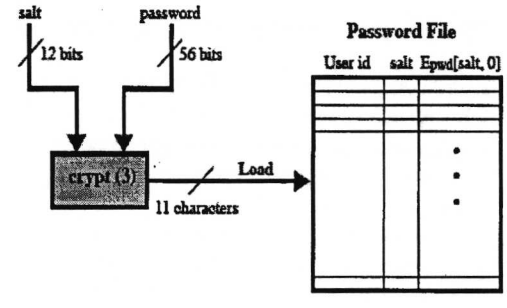


Honeypots

- decoy systems to lure attackers
  - away from accessing critical systems
  - to collect information of their activities
  - to encourage attacker to stay on system so administrator can respond
- are filled with fabricated information
- instrumented to collect detailed information on attackers activities
- may be single or multiple networked systems

Password Management

- front-line defense against intruders
- users supply both:
  - login – determines privileges of that user
  - password – to identify them
- passwords often stored encrypted
  - Unix uses multiple DES (variant with salt)
  - more recent systems use crypto hash function



(a) Loading a new password

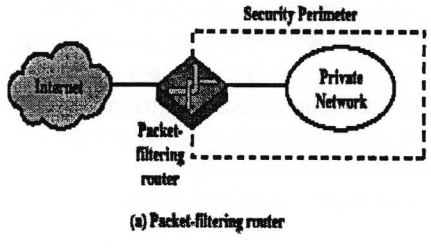
What is a Firewall?

- a **choke point** of control and monitoring
- interconnects networks with differing trust
- imposes restrictions on network services
  - only authorized traffic is allowed
- auditing and controlling access
  - can implement alarms for abnormal behavior
- is itself immune to penetration
- provides **perimeter defence**

Firewall Limitations

- cannot protect from attacks bypassing it
  - eg sneaker net, utility modems, trusted organisations, trusted services (eg SSL/SSH)
- cannot protect against internal threats
  - eg disgruntled employee
- cannot protect against transfer of all virus infected programs or files
  - because of huge range of OS & file types

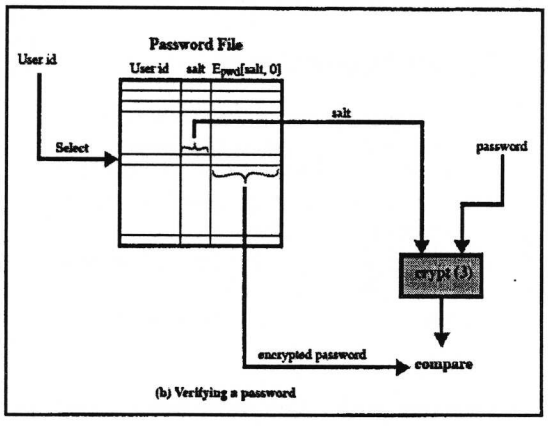
Firewalls – Packet Filters



(a) Packet-filtering router

Firewalls – Packet Filters

- simplest of components
- foundation of any firewall system
- examine each IP packet (no context) and permit or deny according to rules
- hence restrict access to services (ports)
- possible default policies
  - that not expressly permitted is prohibited
  - that not expressly prohibited is permitted



Managing Passwords

- need policies and good user education
- ensure **every** account has a default password
- ensure users change the default passwords to something they can remember
- protect password file from general access
- set technical policies to enforce good passwords
  - minimum length (>6)
  - require a mix of upper & lower case letters, numbers, punctuation
  - block know dictionary words

Managing Passwords

- may reactively run password guessing tools
  - note that good dictionaries exist for almost any language/interest group
- may enforce periodic changing of passwords
- have system monitor failed login attempts, & lockout account if see too many in a short period
- do need to educate users and get support
- balance requirements with user acceptance
- be aware of **social engineering** attacks

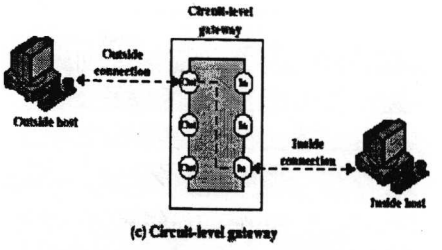
Motivation of Firewall

- seen evolution of information systems
- now everyone want to be on the Internet
- and to interconnect networks
- has persistent security concerns
  - can't easily secure every system in org
- need "harm minimisation"
- a **Firewall** usually part of this

Firewalls - Application Level Gateway (or Proxy)

- use an application specific gateway / proxy
- 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
  - some services naturally support proxying
  - others are more problematic
  - custom services generally not supported

Firewalls - Circuit Level Gateway



(c) Circuit-level gateway

Firewalls - Circuit Level Gateway

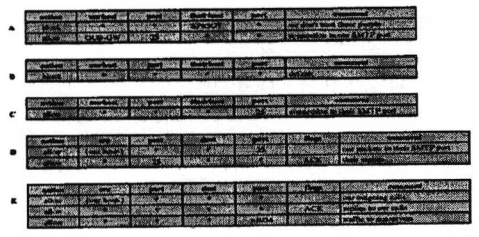
- relays two TCP connections
- imposes security by limiting which such connections are allowed
- once created usually relays traffic without examining contents
- typically used when trust internal users by allowing general outbound connections
- SOCKS commonly used for this

Bastion Host

- highly secure host system
- potentially exposed to "hostile" elements
- hence is secured to withstand this
- may support 2 or more net connections
- may be trusted to enforce trusted separation between network connections
- runs circuit / application level gateways
- or provides externally accessible services

Firewalls – Packet Filters

Table 20.1 Packet-Filtering Examples



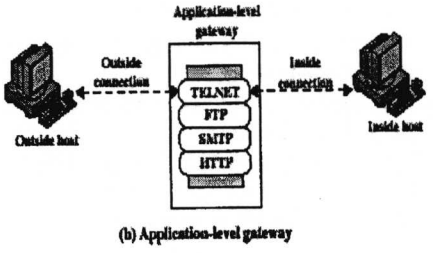
Attacks and Countermeasures

- IP address spoofing
  - fake source address to be trusted
  - add filters on router to block
- source routing attacks
  - attacker sets a route other than default
  - block source routed packets
- tiny fragment attacks
  - force header info into a separate packet fragment
  - either discard or reassemble before check

Firewalls – Stateful Packet Filters

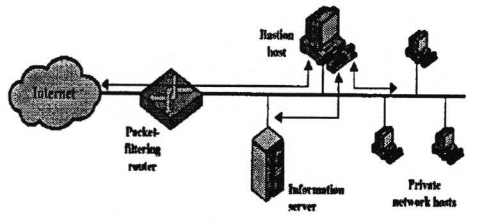
- examine each IP packet in context
  - keeps tracks of client-server sessions
  - checks each packet validly belongs to one
- better able to detect bogus packets out of context

Firewalls - Application Level Gateway (or Proxy)



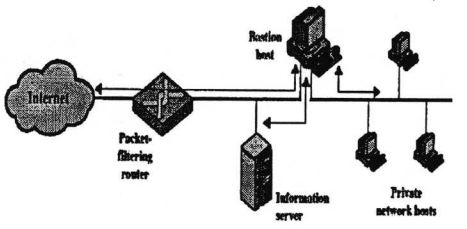
(b) Application-level gateway

Firewall Configurations



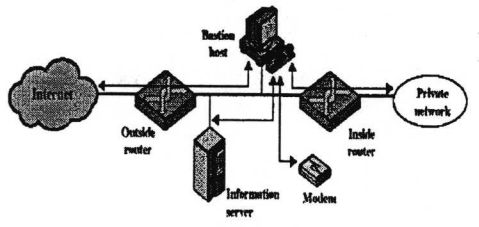
(a) Screened host firewall system (single-homed bastion host)

Firewall Configurations



(b) Screened host firewall system (dual-homed bastion host)

Firewall Configurations



(c) Screened-subnet firewall system

Summary

- have considered:
  - Email security
  - IP security
  - Web security
  - Intrusion detection
  - Firewalls

Virus Countermeasures

- best countermeasure is prevention
- but in general not possible
- hence need to do one or more of:
  - **detection-of viruses in infected system**
  - **identification-of specific infecting virus**
  - **removeal-restoring system to clean state**

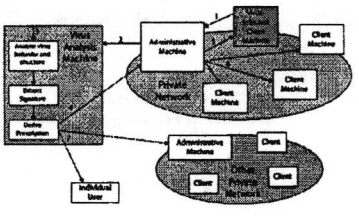
Anti-Virus Software

- first-generation
  - scanner uses virus signature to identify virus
  - or change in length of programs
- second-generation
  - uses heuristic rules to spot viral infection
  - or uses crypto hash of program to spot changes
- third-generation
  - memory-resident programs identify virus by actions
- fourth-generation
  - packages with a variety of antivirus techniques
  - eg scanning & activity traps, access-controls
- arms race continues

Advanced Anti-Virus Techniques

- generic decryption
  - use CPU simulator to check program signature & behavior before actually running it
- digital immune system (IBM)
  - general purpose emulation & virus detection
  - any virus entering org is captured, analyzed, detection/shielding created for it, removed

Digital Immune System



Behavior-Blocking Software

- integrated with host O/S
- monitors program behavior in real-time
  - eg file access, disk format, executable mods, system settings changes, network access

  for possibly malicious actions
  - if detected can block, terminate, or seek ok
- has advantage over scanners
- but malicious code runs before detection

Distributed Denial of Service Attacks (DDoS)

- Distributed Denial of Service (DDoS) attacks form a significant security threat
- making networked systems unavailable
- by flooding with useless traffic
- using large numbers of "zombies"
- growing sophistication of attacks
- defense technologies struggling to cope

Email Virus

- spread using email with attachment containing a macro virus
- triggered when user opens attachment
- or worse even when mail viewed by using scripting features in mail agent
- hence propagate very quickly
- usually targeted at Microsoft Outlook mail agent & Word/Excel documents
- need better O/S & application security

Worms

- replicating but not infecting program
  - typically spreads over a network
- using users distributed privileges or by exploiting system vulnerabilities
- widely used by hackers to create **zombie PC's, subsequently used for further attacks, esp DoS**
- major issue is lack of security of permanently connected systems, esp PC's

Worm Operation

- worm phases like those of viruses:
  - -dormant
  - -propagation
    - search for other systems to infect
    - establish connection to target remote system
    - replicate self onto remote system
  - -triggering
  - -execution

Morris Worm

- best known classic worm
- released by Robert Morris in 1988
- targeted Unix systems
- using several propagation techniques
  - simple password cracking of local pw file
  - exploit bug in finger daemon
  - exploit debug trapdoor in sendmail daemon
- if any attack succeeds then replicated self

Recent Worm Attacks

- new spate of attacks from mid-2001
- Code Red -used MS IIS bug
  - probes random IPs for systems running IIS
  - had trigger time for denial-of-service attack
  - 2nd wave infected 360000 servers in 14 hours
- Code Red 2 -installed backdoor
- Nimda -multiple infection mechanisms
- SQL Slammer -attacked MS SQL server
- Sobig.f -attacked open proxy servers
- Mydoom -mass email worm + backdoor

Worm Techology

- multiplatform
- multiexploit
- ultrafast spreading
- polymorphic
- metamorphic
- transport vehicles
- zero-day exploit

Zombie

- program which secretly takes over another networked computer
- then uses it to indirectly launch attacks
- often used to launch distributed denial of service (DDoS) attacks
- exploits known flaws in network systems

Viruses

- a piece of self-replicating code attached to some other code
  - similar to biological virus
- both propagates itself & carries a payload
  - carries code to make copies of itself
  - as well as code to perform some covert task

Virus Operation

- virus phases:dormant –waiting on trigger event
  - propagation –replicating to programs/disks
  - triggering –by event to execute payload
  - execution –of payload
- details usually machine/OS specific
  - exploiting features/weaknesses

Virus Structure

```
program V :=
{goto main;
1234567;
    subroutine infect-executable :={loop:
    file := get-random-executable-file;
    if (first-line-of-file = 1234567) then goto loop
    else prepend V to file; }
    subroutine do-damage := {whatever damage is to be done}
    subroutine trigger-pulled := {return true if condition holds}
    main: main-program :={infect-executable;
    if trigger-pulled then do-damage;
    goto next;}
next:
}
```

Types of Viruses

- can classify on basis of how they attack
- parasitic virus
- memory-resident virus
- boot sector virus
- stealth
- polymorphic virus
- metamorphic virus

Macro Virus

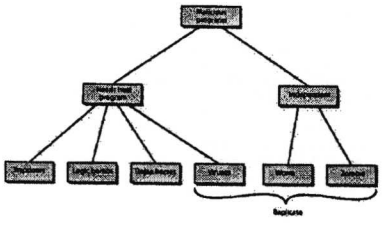
- **macro codeattached to some data file**
- interpreted by program using file
  - eg Word/Excel macros
  - esp. using auto command & command macros
- code is now platform independent
- is a major source of new viral infections
- blur distinction between data and program files
- classic trade-off: "ease of use" vs "security"
- have improving security in Word etc
- are no longer dominant virus threat

Virus and related threats – Countermeasures

Viruses and Other Malicious Content

- computer viruses have got a lot of publicity
- one of a family of **malicious software**
- effects usually obvious
- have figured in news reports, fiction, movies (often exaggerated)
- getting more attention than deserve
- are a concern though

Malicious Software



Backdoor or Trapdoor

- secret entry point into a program
- allows those who know access bypassing usual security procedures
- have been commonly used by developers
- a threat when left in production programs allowing exploited by attackers
- very hard to block in O/S
- requires good s/w development & update

Logic Bomb

- one of oldest types of malicious software
- code embedded in legitimate program
- activated when specified conditions meteg
  - -presence/absence of some file
  - -particular date/time
  - -particular user
- when triggered typically damage system
  - -modify/delete files/disks, halt machine, etc

Trojan Horse

- program with hidden side-effects
- which is usually superficially attractive
  - -eg game, s/w upgrade etc
- when run performs some additional tasksallows
  - -attacker to indirectly gain access they do not have directly
- often used to propagate a virus/worm or install a backdoor
- or simply to destroy data

Virus Countermeasures

- best countermeasure is prevention
- but in general not possible
- hence need to do one or more of:
  - **detection-of viruses in infected system**
  - **identification-of specific infecting virus**
  - **removeal-restoring system to clean state**

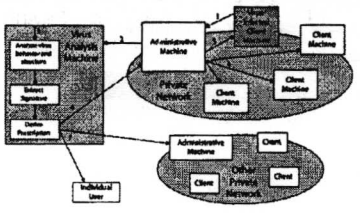
Anti-Virus Software

- first-generation
  - scanner uses virus signature to identify virus
  - or change in length of programs
- second-generation
  - uses heuristic rules to spot viral infection
  - or uses crypto hash of program to spot changes
- third-generation
  - memory-resident programs identify virus by actions
- fourth-generation
  - packages with a variety of antivirus techniques
  - eg scanning & activity traps, access-controls
- arms race continues

Advanced Anti-Virus Techniques

- generic decryption
  - use CPU simulator to check program signature & behavior before actually running it
- digital immune system (IBM)
  - general purpose emulation & virus detection
  - any virus entering org is captured, analyzed, detection/shielding created for it, removed

Digital Immune System



Behavior-Blocking Software

- integrated with host O/S
- monitors program behavior in real-time
  - eg file access, disk format, executable mods, system settings changes, network access

  for possibly malicious actions
  - If detected can block, terminate, or seek ok
- has advantage over scanners
- but malicious code runs before detection

Distributed Denial of Service Attacks (DDoS)

- Distributed Denial of Service (DDoS) attacks form a significant security threat
- making networked systems unavailable
- by flooding with useless traffic
- using large numbers of "zombies"
- growing sophistication of attacks
- defense technologies struggling to cope

via another site's dynamically generated Web pages. The attacker's target is not a Website, but rather its users (i.e. clients or browsers).

The idea of CSSV is quite simple to understand and is based on exploiting the scripting technologies, such as JavaScript, VBScript or JScript. Let us understand how this works. Consider the following Web page containing a form as shown in Fig. 10.9, in which the user is expected to enter her postal address. Suppose that the URL of the site sending this page is www.test.com and when the user submits this form, it would be processed by a server-side program called as address.asp. We would typically expect the user to enter the house number, street name, city, postal code and country, etc. However, imagine that the user enters the following weird string, instead:
<SCRIPT>Hello World</SCRIPT>

As a result, the URL submitted would be something like www.test.com/address.asp?address=
<SCRIPT>Hello World </SCRIPT>.

Now suppose that the server-side program address.asp does not validate the input sent by the user and simply sends the value of the field address to the next Web page. What would this translate to? It would mean that the next Web page would receive the value of address as <SCRIPT>Hello World</ SCRIPT>.
As we know, this would most likely treat the value of the address field as a script, which would be executed as if it is written in a scripting language, such as JavaScript etc on the Web browser. Therefore, the user would get to see *Hello World.*

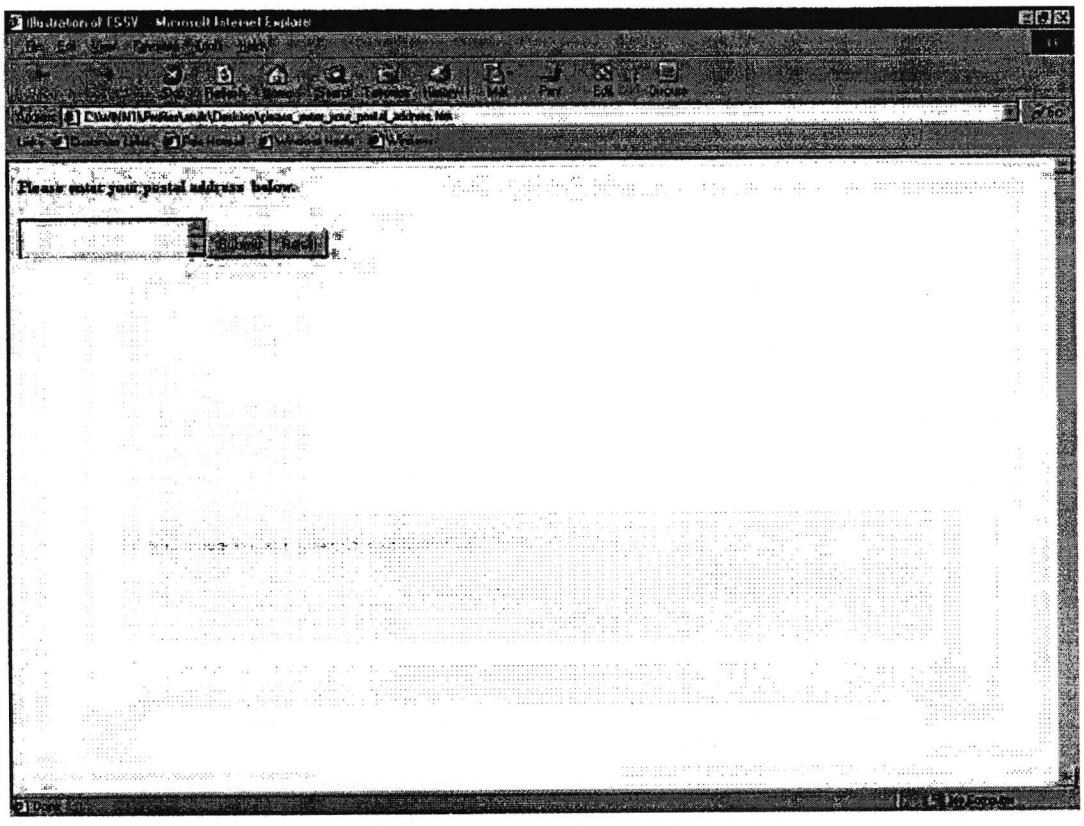


Fig. 10.9 *Sample HTML form*

Obviously, no serious damage is done. However, extrapolate this possibility to other situations where a user can actually send damaging scripts to the server. This can cause the same or another client to receive a Web page whose contents/look-and-feel are changed. In a more damaging case, the confidential information entered by a user could also be captured and sent to another user and so on. How can this be done?
When a JavaScript program gets downloaded on a browser through a CSSV attack, the JavaScript, in turn, can call up the services of an ActiveX control. An ActiveX control is a small program that gets downloaded from the server to the client and executes on the client. The ActiveX control can write to the disk or read from it and perform many such tasks. Once downloaded to the client via the malicious

CASE STUDIES

10.4 Secure Inter-branch Payment Transactions

Points for classroom discussions

1. *What is the technology to achieve non-repudiation? How is this guaranteed?*
2. *How is the problem of key distribution resolved in PKI?*
3. *Why are cryptographic toolkits required?*
4. *How can smart cards be used in cryptography?*

General Bank Of India (GBI) has implemented an Electronic Payment System called as *EPS* in about 1200 branches across the country. This system transfers payment instructions between two computerized branches of GBI. A central server is maintained at the EPS office located in Mumbai. The branch offices connect to the Local VSAT of a private network by using dial-up connection. The local VSAT has a connectivity established with the EPS office. GBI utilizes its proprietary messaging service called as *GBI-Transfer* to exchange payment instructions. Currently, EPS has minimal data security. As the system operates in a closed network, the current security infrastructure may suffice the need. The data moving across the network is in encrypted format.

Current EPS Architecture EPS is used to transmit payment details from the payer branch to the payee branch via the central server in Mumbai. Fig. 10.5 depicts the flow, which is also described stepby-step.

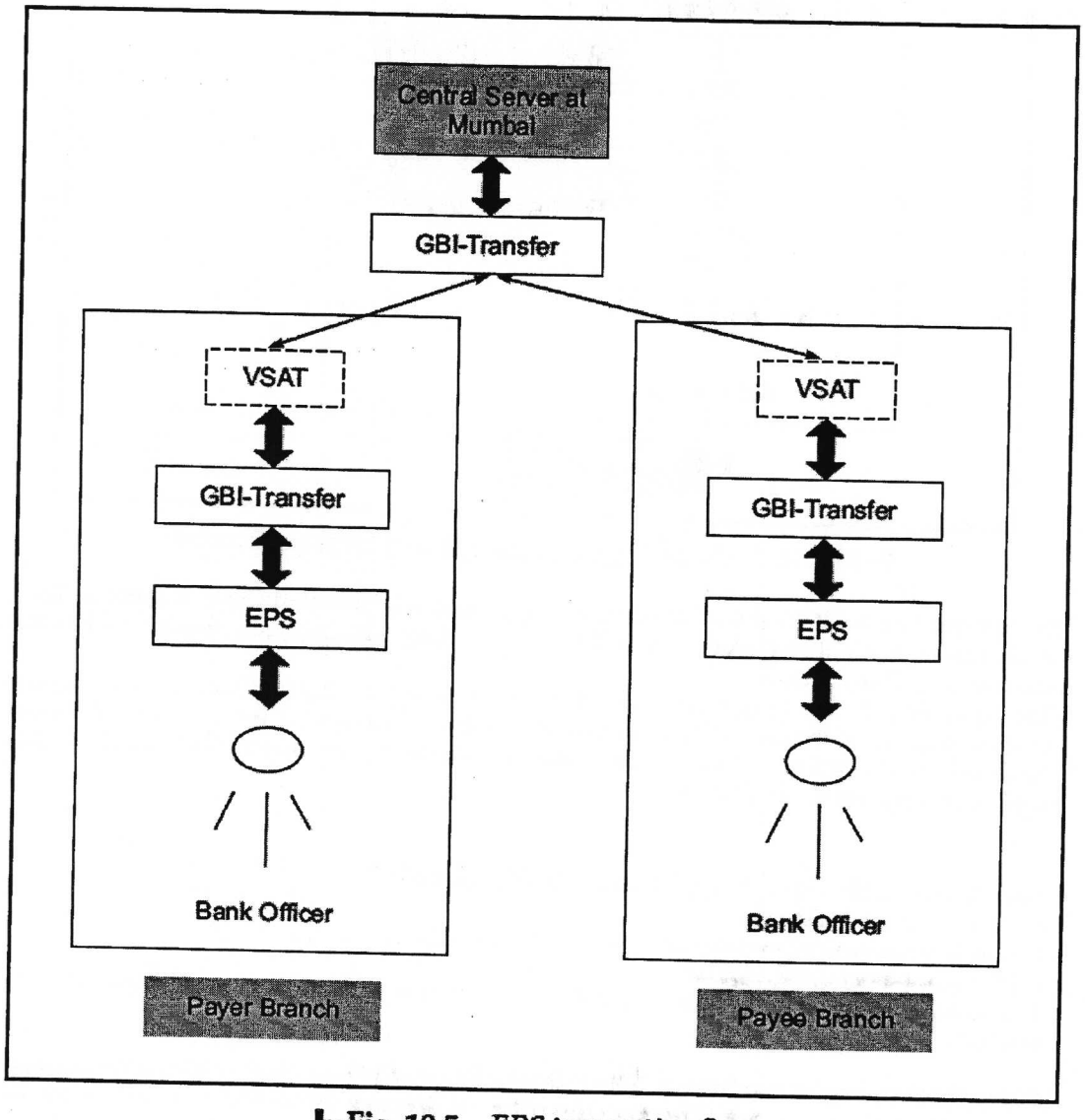


Fig. 10.5 *EPS transaction flow*

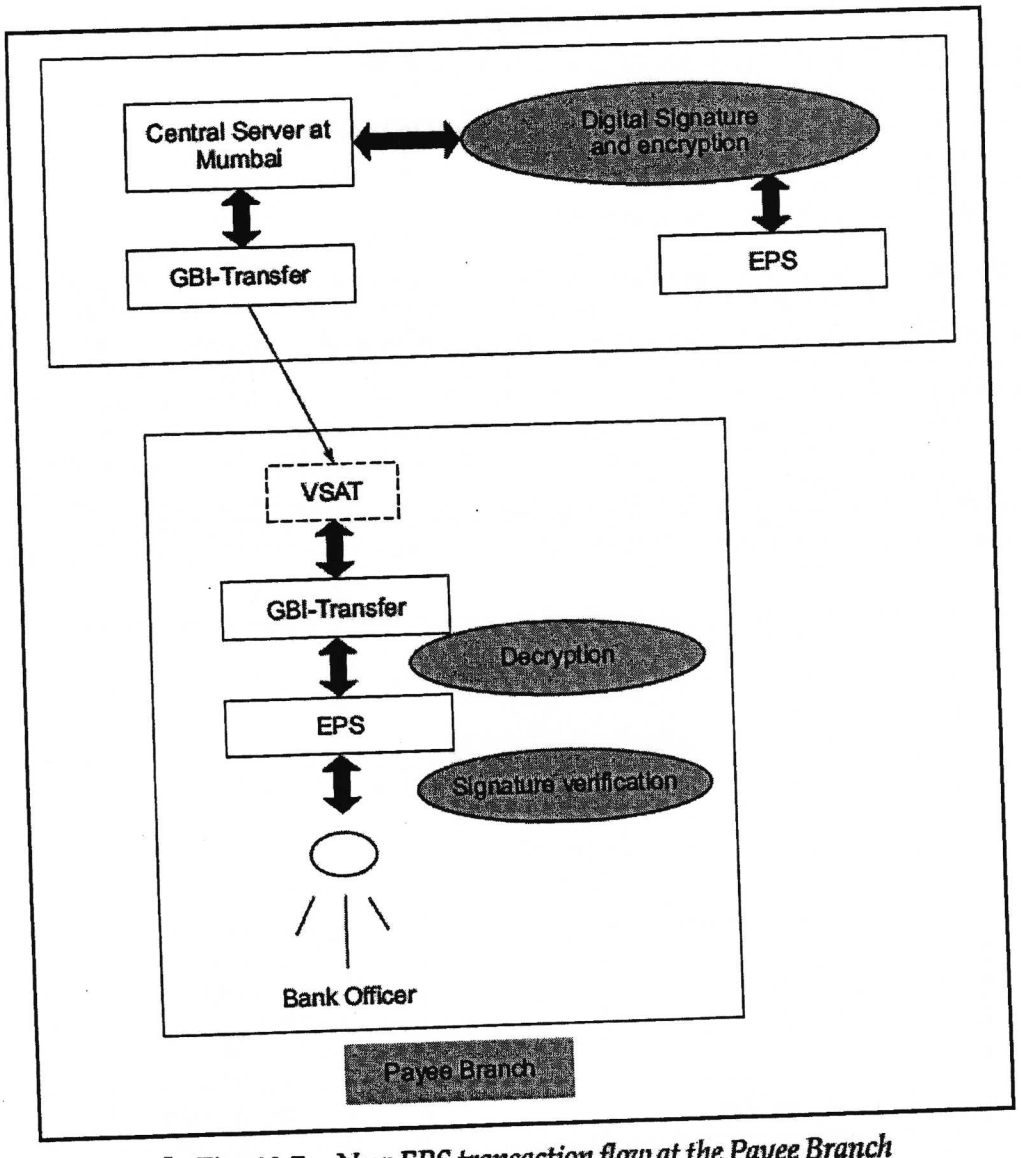


Fig. 10.7 *New EPS transaction flow at the Payee Branch*

In the Payee Leg, the PM software at the EPS central office will generate a Credit Request for the *Payee Bank*. This request will be digitally signed. The signature along with the Credit Request will be encrypted and sent to the *Payee Branch*.
The *Payee Branch* will decrypt the Credit Request and verify the digital signature. If the signature is verified successfully, the transaction is entered into database. Otherwise, it gets rejected and the status of the same is sent to EPS central Office. The Credit Response to the EPS central office can also be digitally signed and encrypted in a similar fashion.

10.7 Cross Site Scripting Vulnerability (CSSV)

Points for classroom discussions
1. *What is the purpose of scripting technologies on the Internet?*
2. *What can prevent CSSV attacks?*
3. *What sort of testing can the creators of a Web site perform in order to guard against possible CSSV attacks?*

Cross Site Scripting Vulnerability (CSSV) is a relatively new form of attacks that exploits inadequate validations on the server-side. The term *Cross Server Scripting Vulnerability (CSSV)* is actually not completely correct. However, this term was coined when the problem was not completely understood and has stuck ever since. Cross-site scripting happens when malicious tags and/or scripts attack a Web browser

A typical payment transfer takes the following steps:
1. A data-entry person in the *Payer Branch* enters transaction details through the EPS interface.
2. A Bank Officer checks the validity of the transaction through the EPS interface.
3. After validating the transaction, the Bank Officer authorizes the transaction. Authorized transaction is stored in a local Payment Master (PM) database.
4. Once the transaction is stored in PM, a copy of the same is encrypted and stored in a file. This transaction file is stored in OUT directory.
5. The *GBI-Transfer* application looks for any pending transactions (i.e. for the presence of any files in the OUT directory) by a polling mechanism and if it finds such transactions, it sends all these files one-by-one to the EPS central office located in Mumbai by dialing the local VSAT.
6. The local VSAT gets connectivity to the EPS central office and the transaction is transferred and stored in the IN directory at the EPS central office.
7. The interface program at the EPS central office collects the file pending in the IN directory and sends it to the PM application at that office.
8. In order to send the Credit Request to PM, the transaction headers are changed. The transaction with changed headers in encrypted format is then placed in OUT directory of the EPS central office.
9. The GBI-Transfer application at the EPS central office collects the transactions pending in the OUT directory and sends them to the *Payee Bank* through the VSAT.
10. The transaction is transferred and stored in the IN directory of the *Payee Branch.*
11. The interface program at the *Payee Branch* collects the transaction and posts it in PM.
12. PM marks the credit entry and returns back an acknowledgement of the same. The acknowledgement is placed in OUT directory of the *Payee Branch.*
13. The acknowledgement is picked by GBI-Transfer at the *Payee Branch* and sent to the EPS central office through the VSAT.
14. The EPS central office receives the credit acknowledgement and forwards it to *Payer Branch.*
15. The *Payer Branch* receives the credit acknowledgement receipt. This completes the transaction.

Requirements to Enhance EPS As GBI is in the process of complete automation and setting up connectivity over the Internet or a private network, they need to ensure stringent security measures, which demand the usage of a Public Key Infrastructure (PKI) framework.

As a part of implementing security, GBI wants the following aspects to be ensured:
- Non-repudiation (Digital Signatures)
- Encryption – 128-bit (Upgrade to the current 56-bit encryption)
- Smart card support for storing sensitive data & on-card digital signing
- Closed loop Public Key Infrastructure

Proposed Solution Since providing cryptographic functionalities require the usage of a cryptographic toolkit, it is assumed that GBI will implement an appropriate Certification Authority (CA) infrastructure and a PKI infrastructure offering. The transaction will be digitally signed and encrypted/decrypted at the Payer and Payee branches, as well as at the EPS central office. The signing operation can be performed on the system or on external hardware like a smart card. On the server side, a provision of automated signing without any manual intervention will be provided.

The transaction flow described earlier would now be split into two legs:
- The Payer Leg (*Payer Branch* to the EPS central office)
- The Payee Leg (EPS central office to the *Payee Branch*)

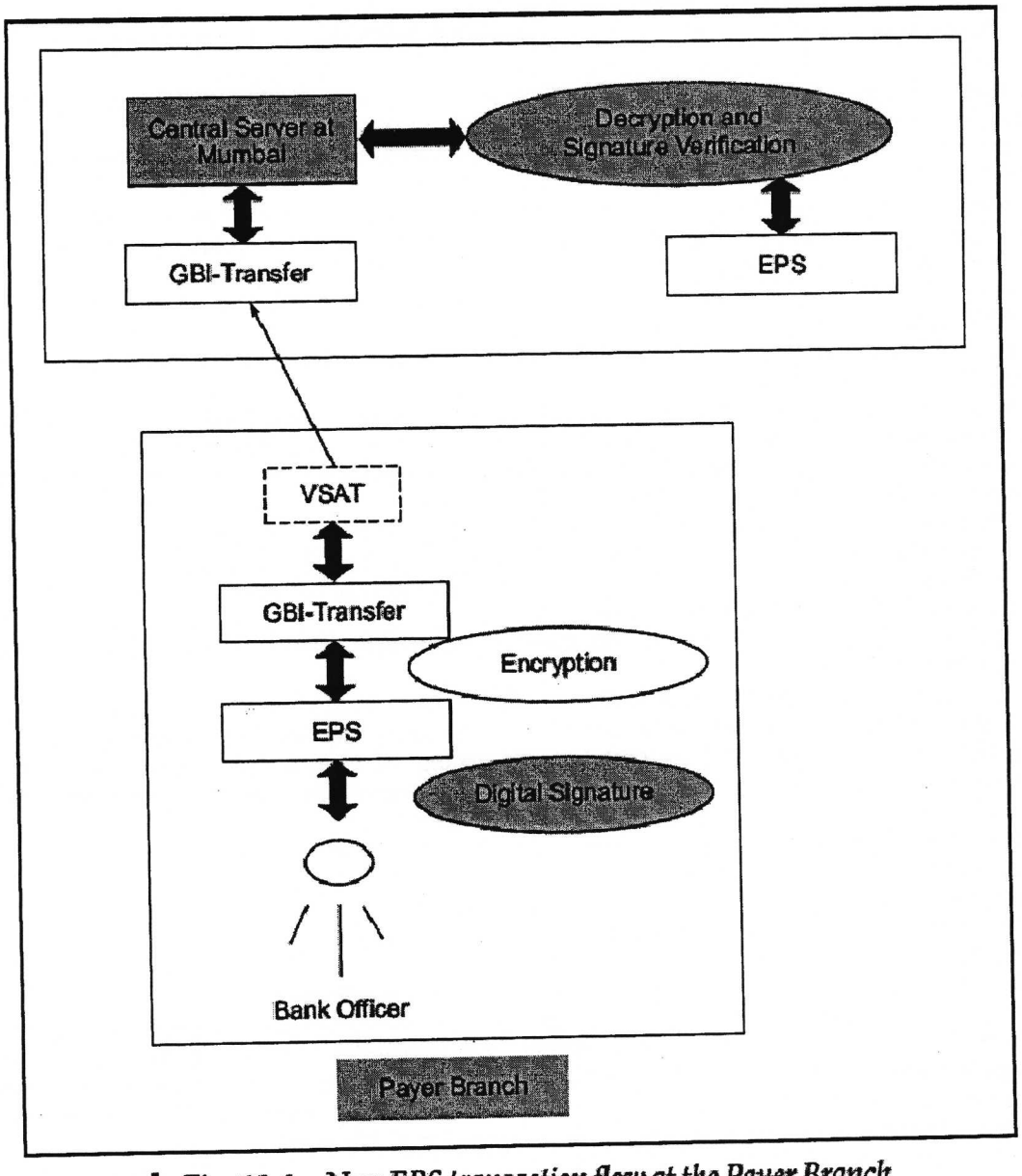


Fig. 10.6 *New EPS transaction flow at the Payer Branch*

The architecture for the Payer Leg is shown in Fig. 10.6. As shown, after verifying the transaction, the EPS Officer authorizes the transaction at the *Payer Branch*. Internally, the application digitally signs the transaction. This signature, along with the transaction data is stored in the local PM Database and then encrypted and placed in the IN directory. For signature and encryption, a cryptographic toolkit is required at the *Payer Branch*. The signed-and-encrypted transaction is sent to the EPS central office in the same way as before.

The encrypted file is decrypted at EPS central office. Before storing the transaction in the database, the digital signature is verified using an appropriate cryptographic toolkit. The verification process may also check the status of the user's digital certificate by either CRL or OCSP check. If the status of the certificate is invalid, the transaction will be rejected, otherwise it will be stored in the local PM database.

On the Payee Leg, the EPS central office will create a Credit Request as before, sign and encrypt it with the bank officer's digital certificate. This signed-and-encrypted request will be forwarded to the *Payee Branch*. The flow is shown in Fig. 10.7.

Code No: 56030 **R09**

JAWAHARLAL NEHRU TECHNOLOGICAL UNIVERSITY, HYDERABAD
B. Tech III Year II Semester Examinations, May - 2015
NETWORK SECURITY
(Common to CSE, IT)

Time: 3 hours Max. Marks: 75

Answer any five questions
All questions carry equal marks

- - -

1.a) Define Authentication, Confidentiality, Non-repudiation, and Availability.
b) Write notes on route table modification and UDP Hijacking. [5+10]

2. Explain cipher block modes of operation with neat diagrams. [15]

3.a) What requirements must a public key cryptosystem fulfill to be a secure algorithm?
b) Define digital signature, digital certificate. What are the advantages of digital signature, digital certificate? [7+8]

4. Explain PGP in detail. [15]

5.a) Explain IPSec Document Overview with a neat diagram.
b) Explain about different payload types of ISAKMP. [7+8]

6.a) List and briefly define the parameters that define an SSL connection state.
b) What services are provided by SSL record Protocol?
c) List and define the principal categories of SET participants. [4+4+7]

7.a) What are the two common techniques used to protect a password file?
b) What is the role of encryption and compression in the operation of a Virus?
c) List and briefly explain about types of intruders. [4+6+5]

8.a) List four techniques used by a Firewall to control access and enforce a security policy.
b) Briefly explain access control list and capability ticket. [8+7]

---ooOoo---

Code No: 56030 **R09**

JAWAHARLAL NEHRU TECHNOLOGICAL UNIVERSITY HYDERABAD
B. Tech III Year II Semester Examinations, May - 2016
NETWORK SECURITY
(Common to CSE, IT)

Time: 3 hours **Max. Marks: 75**

Answer any five questions
All questions carry equal marks

- - -

1.a) Explain various security attacks.
b) What is UDP hijacking? [8+7]

2. Explain HMAC algorithm. [15]

3.a) Explain about Kerberos.
b) Compare Symmetric algorithms with public key algorithms. [7+8]

4.a) Explain the fields in PGP message format.
b) What are the S/MIME messages? [7+8]

5.a) Draw and explain fields in AH header.
b) Write short notes on key management. [7+8]

6. Explain about secure Electronic Transaction. [15]

7.a) Write about SNMPV1 community facility.
b) Write short notes on intruders. [7+8]

8.a) Explain about trusted systems.
b) What are the limitations of firewalls? [7+8]

--ooOoo--

R13

Code No: 117DY

JAWAHARLAL NEHRU TECHNOLOGICAL UNIVERSITY HYDERABAD
B. Tech IV Year I Semester Examinations, March - 2017
INFORMATION SECURITY
(Information Technology)

Time: 3 Hours **Max. Marks: 75**

Note: This question paper contains two parts A and B.
Part A is compulsory which carries 25 marks. Answer all questions in Part A. Part B consists of 5 Units. Answer any one full question from each unit. Each question carries 10 marks and may have a, b, c as sub questions.

Part- A (25 Marks)

1.a) List and explain about different security services defined by X.800? [2]
b) Briefly define Caesar cipher? Apply Caesar cipher technique for a given plain text: network security and cryptography? [3]
c) Explain briefly about the different block cipher modes of operation – CFB, OFB, CTR? [2]
d) Which of the four different stages involved in each round of AES? Explain it with neat diagrams. [3]
e) What are the four requirements were defined for Kerberos? [2]
f) What are the basic uses of message authentication? [3]
g) What is meant by Secure Electronic Transaction? [2]
h) Explain about confidentiality and message Integrity? [3]
i) Explain about Logic bombs and Trojan Horses? [2]
j) What is Digital Immune System? [3]

Part-B (50 Marks)

2.a) Briefly define substitution technique. Apply play fair cipher technique for a given keyword: monarchy.
b) Briefly explain Vernam cipher with an example. [5+5]

OR

3.a) Write a short notes on Playfair Cipher. Construct a Playfair matrix with the key "largest" and encrypt the message "Must see you over Cadogan West".
b) With a neat diagram, explain about a model for network security. [5+5]

4.a) With a neat diagram, explain briefly about the data encryption standard algorithm? And Briefly discuss about the strength of data encryption standard algorithm?
b) Explain in detail about public key cryptosystems. [5+5]

OR

5.a) With a neat diagram, explain about the multiple encryptions (Triple DES with two and three keys)?
b) What is the difference between Double and Triple DES? [6+4]

6.a) In what order should be the signature function and the confidentiality function be applied to a message, and why?

b) Describe the digital certificates? [5+5]

OR

7.a) Discuss the techniques of public key certificates for distribution of public keys?

b) X.509 includes three alternative authentication procedure what are these three procedure explain them in brief? [5+5]

8.a) Explain the general format of Pretty Good Privacy Message.

b) Explain Radix-64 format? Compare PGP and S/MIME in Radix-64 conversion. [5+5]

OR

9.a) What are the MIME Specifications?

b) Why does PGP generates a signature before applying compression and how it generates signature. Explain? [5+5]

10.a) Discuss briefly about the various components of SET systems?

b) Explain in detail about packet-filtering router with a neat diagram? [5+5]

OR

11.a) Explain briefly the 4 techniques used in guessable password.

b) Write short notes on Firewalls? [5+5]

--ooOoo--

Code No: 56030 **R09**

JAWAHARLAL NEHRU TECHNOLOGICAL UNIVERSITY, HYDERABAD
B. Tech III Year II Semester Examinations, May - 2015
NETWORK SECURITY
(Common to CSE, IT)

Time: 3 hours Max. Marks: 75

Answer any five questions
All questions carry equal marks

- - -

1.a) Define Authentication, Confidentiality, Non-repudiation, and Availability.
b) Write notes on route table modification and UDP Hijacking. [5+10]

2. Explain cipher block modes of operation with neat diagrams. [15]

3.a) What requirements must a public key cryptosystem fulfill to be a secure algorithm?
b) Define digital signature, digital certificate. What are the advantages of digital signature, digital certificate? [7+8]

4. Explain PGP in detail. [15]

5.a) Explain IPSec Document Overview with a neat diagram.
b) Explain about different payload types of ISAKMP. [7+8]

6.a) List and briefly define the parameters that define an SSL connection state.
b) What services are provided by SSL record Protocol?
c) List and define the principal categories of SET participants. [4+4+7]

7.a) What are the two common techniques used to protect a password file?
b) What is the role of encryption and compression in the operation of a Virus?
c) List and briefly explain about types of intruders. [4+6+5]

8.a) List four techniques used by a Firewall to control access and enforce a security policy.
b) Briefly explain access control list and capability ticket. [8+7]

---ooOoo---

Code No: 56030 **R09**

JAWAHARLAL NEHRU TECHNOLOGICAL UNIVERSITY HYDERABAD
B. Tech III Year II Semester Examinations, May - 2016
NETWORK SECURITY
(Common to CSE, IT)

Time: 3 hours **Max. Marks: 75**

Answer any five questions
All questions carry equal marks

- - -

1.a) Explain various security attacks.
b) What is UDP hijacking? [8+7]

2. Explain HMAC algorithm. [15]

3.a) Explain about Kerberos.
b) Compare Symmetric algorithms with public key algorithms. [7+8]

4.a) Explain the fields in PGP message format.
b) What are the S/MIME messages? [7+8]

5.a) Draw and explain fields in AH header.
b) Write short notes on key management. [7+8]

6. Explain about secure Electronic Transaction. [15]

7.a) Write about SNMPV1 community facility.
b) Write short notes on intruders. [7+8]

8.a) Explain about trusted systems.
b) What are the limitations of firewalls? [7+8]

--ooOoo--

R13

Code No: 117DY

JAWAHARLAL NEHRU TECHNOLOGICAL UNIVERSITY HYDERABAD
B. Tech IV Year I Semester Examinations, March - 2017
INFORMATION SECURITY
(Information Technology)

Time: 3 Hours **Max. Marks: 75**

Note: This question paper contains two parts A and B.
Part A is compulsory which carries 25 marks. Answer all questions in Part A. Part B consists of 5 Units. Answer any one full question from each unit. Each question carries 10 marks and may have a, b, c as sub questions.

Part- A (25 Marks)

1.a) List and explain about different security services defined by X.800? [2]
b) Briefly define Caesar cipher? Apply Caesar cipher technique for a given plain text: network security and cryptography? [3]
c) Explain briefly about the different block cipher modes of operation – CFB, OFB, CTR? [2]
d) Which of the four different stages involved in each round of AES? Explain it with neat diagrams. [3]
e) What are the four requirements were defined for Kerberos? [2]
f) What are the basic uses of message authentication? [3]
g) What is meant by Secure Electronic Transaction? [2]
h) Explain about confidentiality and message Integrity? [3]
i) Explain about Logic bombs and Trojan Horses? [2]
j) What is Digital Immune System? [3]

Part-B (50 Marks)

2.a) Briefly define substitution technique. Apply play fair cipher technique for a given keyword: monarchy.
b) Briefly explain Vernam cipher with an example. [5+5]

OR

3.a) Write a short notes on Playfair Cipher. Construct a Playfair matrix with the key "largest" and encrypt the message "Must see you over Cadogan West".
b) With a neat diagram, explain about a model for network security. [5+5]

4.a) With a neat diagram, explain briefly about the data encryption standard algorithm? And Briefly discuss about the strength of data encryption standard algorithm?
b) Explain in detail about public key cryptosystems. [5+5]

OR

5.a) With a neat diagram, explain about the multiple encryptions (Triple DES with two and three keys)?
b) What is the difference between Double and Triple DES? [6+4]

6.a) In what order should be the signature function and the confidentiality function be applied to a message, and why?

b) Describe the digital certificates? [5+5]

OR

7.a) Discuss the techniques of public key certificates for distribution of public keys?

b) X.509 includes three alternative authentication procedure what are these three procedure explain them in brief? [5+5]

8.a) Explain the general format of Pretty Good Privacy Message.

b) Explain Radix-64 format? Compare PGP and S/MIME in Radix-64 conversion. [5+5]

OR

9.a) What are the MIME Specifications?

b) Why does PGP generates a signature before applying compression and how it generates signature. Explain? [5+5]

10.a) Discuss briefly about the various components of SET systems?

b) Explain in detail about packet-filtering router with a neat diagram? [5+5]

OR

11.a) Explain briefly the 4 techniques used in guessable password.

b) Write short notes on Firewalls? [5+5]

--ooOoo--

KESHAV MEMORIAL INSTITUTE OF TECHNOLOGY
Narayanguda, Hyderabad.
IV B.Tech – I – Semester –R13- Mid Internal Examinations I- AUG – 2016

Subject: IS
Branch / Section: IT **Duration: 60 Min.** **Max. Marks: 10**

Answer any TWO from the following Questions

1. Write about DES with neat diagram and procedure in each round?
   (or)
   Explain about AES algorithm in detail?

2. Explain the procedure involved in RSA algorithm? Solve the following problem.
   The public of a given user e=7 and N=187, what is the private key of the user?

3. Write down the differences between Symmetric key and Asymmetric key cryptography

4. Explain about key distribution in detail

Code No: A70522 **Set No. 1**

JAWAHARLAL NEHRU TECHNOLOGICAL UNIVERSITY HYDERABAD

IV B.Tech. I Sem., I Mid-Term Examinations, AUG– 2016

INFORMATION SECURITY

Objective Exam

Name: _____ **Hall Ticket No.** | | | | | | | | | | |

Answer All Questions. All Questions Carry Equal Marks. Time: 20 Min. Marks: 10.

I Choose the correct alternative:

1. _____attacks attempts to alter the system resources. [ ]
   A) passive B) active C) both A&B D)Models Views controller

2. ____ prevents either the sender or receiver from denying a transmitted message. [ ]
   A)confidentiality B)integrity C)authentication D)non-repudiation

3. In RSA algorithm plain text M= [ ]
   A)m^dmod n B)c^dmod n C)m^emod n D)c^emod n

4. Symmetric block cipher consists of a sequence of rounds, with each round perform [ ]
   A) Substitutions B) Permutations C) Both D) None

5. In AES Algorithm with the key length of 128 bits no of rounds performed is [ ]
   A) 10 B)12 C)18 D)16

6. _____could breach security and cause harm [ ]
   A)Security Service B)Security Attack C) Security Mechanism D) All

7. __is a technique in which the letters of the plain text are replaced by other letters or symbols[ ]
   A) Transposition B)substitution C)Permutation D) A & B

8. _____ is the original message that is fed into the algorithm as input [ ]
   A) Secret Key B)Cipher Text C) Plain Text D) All

9. In DES The substitution consist of set of _____ S-Boxes [ ]
   A) 6 B)4 c)16 d)8

10. _____ Enables two users to exchange the key securely [ ]
    A) Diffie – Hellman B) RSA C) DSS D) DES

Cont...2

II FIIL IN THE BLANKS

11. RSA stands for_____.

12. DES algorithm uses block length of 64 bits of plain text and key length of _____ bits.

13. _____service is the protection of data from passive attacks

14. A method of breaking cipher text by trying all possible keys called _____.

15. Restoring the plain text from cipher text called_____.

16. In Diffie – Hellman generation of secret key by user A , K = _____.

17. The message which is encrypted by receiver public key, is decrypted with _____ key.

18. In RSA $\varphi(n)$ = _____ .

19. _____ takes place when one entity pretends to be a different entity.

20. In _____ key cipher two different related keys are used.

-000-

Code No: A70522 **Set No. 2**

JAWAHARLAL NEHRU TECHNOLOGICAL UNIVERSITY HYDERABAD

IV B.Tech. I Sem., I Mid-Term Examinations, AUG– 2016

INFORMATION SECURITY

Objective Exam

Name: _____ Hall Ticket No. | | | | | | | | | | |

Answer All Questions. All Questions Carry Equal Marks. Time: 20 Min. Marks: 10.

I Choose the correct alternative:

1. Symmetric block cipher consists of a sequence of rounds, with each round perform [ ]
   A) Substitutions B) Permutations C) Both D) None

2. In AES Algorithm with the key length of 128 bits no of rounds performed is [ ]
   A) 10 B)12 C)18 D)16

3. _____attacks attempts to alter the system resources. [ ]
   A) passive B) active C) both A&B D)Models Views controller

4. ____ prevents either the sender or receiver from denying a transmitted message. [ ]
   A) confidentiality B)integrity C)authentication D)non-repudiation

5. In RSA algorithm plain text M= [ ]
   A)m^dmod n B)c^dmod n C)m^emod n D)c^emod n

6. In DES The substitution consist of set of _____ S-Boxes [ ]
   A) 6 B)4 c)16 d)8

7. _____ Enables two users to exchange the key securely [ ]
   A) Diffie – Hellman B) RSA C) DSS D) DES

8. _____could breach security and cause harm [ ]
   A)Security Service B)Security Attack C) Security Mechanism D) All

9. __is a technique in which the letters of the plain text are replaced by other letters or symbols[ ]
   A) Transposition B)substitution C)Permutation D) A & B

10. _____ is the original message that is fed into the algorithm as input [ ]
    A) Secret Key B)Cipher Text C) Plain Text D) All

Cont...2

II FIIL IN THE BLANKS

11. A method of breaking cipher text by trying all possible keys called _____.

12. Restoring the plain text from cipher text called_____.
13. RSA stands for_____.

14. DES algorithm uses block length of 64 bits of plain text and key length of _____ bits.

15. _____service is the protection of data from passive attacks
16. _____ takes place when one entity pretends to be a different entity.

17. In _____ key cipher two different related keys are used.

18. In Diffie – Hellman generation of secret key by user A , K = _____.

19. The message which is encrypted by receiver public key, is decrypted with _____ key.

20. In RSA $\varphi(n)$ = _____ .

-000-

Code No: A70522

Set No. 3

JAWAHARLAL NEHRU TECHNOLOGICAL UNIVERSITY HYDERABAD

IV B.Tech. I Sem., I Mid-Term Examinations, AUG– 2016

INFORMATION SECURITY

Objective Exam

Name: _____ **Hall Ticket No.** | | | | | | | | | | |

Answer All Questions. All Questions Carry Equal Marks. Time: 20 Min. Marks: 10.

I Choose the correct alternative:

1. In RSA algorithm plain text M= [ ]
   A)m^dmod n B)c^dmod n C)m^emod n D)c^emod n
2. Symmetric block cipher consists of a sequence of rounds, with each round perform [ ]
   A) Substitutions B) Permutations C) Both D) None
3. In AES Algorithm with the key length of 128 bits no of rounds performed is [ ]
   A) 10 B)12 C)18 D)16
4. __is a technique in which the letters of the plain text are replaced by other letters or symbols[ ]
   A) Transposition B)substitution C)Permutation D) A & B
5. _____ is the original message that is fed into the algorithm as input [ ]
   A) Secret Key B)Cipher Text C) Plain Text D) All
6. _____could breach security and cause harm [ ]
   A)Security Service B)Security Attack C) Security Mechanism D) All
7. In DES The substitution consist of set of _____ S-Boxes [ ]
   A) 6 B)4 c)16 d)8
8. _____ Enables two users to exchange the key securely [ ]
   A) Diffie – Hellman B) RSA C) DSS D) DES
9. _____attacks attempts to alter the system resources. [ ]
   A) passive B) active C) both A&B D)Models Views controller
10. ____ prevents either the sender or receiver from denying a transmitted message. [ ]
    A)confidentiality B)integrity C)authentication D)non-repudiation

Cont...2

II FIIL IN THE BLANKS

11. _____service is the protection of data from passive attacks

12. A method of breaking cipher text by trying all possible keys called _____.

13. Restoring the plain text from cipher text called_____.

14. The message which is encrypted by receiver public key, is decrypted with _____ key.

15. In RSA $\varphi(n)$ = _____ .

16. In Diffie – Hellman generation of secret key by user A , K = _____.

17. _____ takes place when one entity pretends to be a different entity.

18. In _____ key cipher two different related keys are used.

19. RSA stands for_____.

20. DES algorithm uses block length of 64 bits of plain text and key length of _____ bits.

-000-

Code No: A70522

Set No. 4

JAWAHARLAL NEHRU TECHNOLOGICAL UNIVERSITY HYDERABAD

IV B.Tech. I Sem., I Mid-Term Examinations, AUG– 2016

INFORMATION SECURITY

Objective Exam

Name: _____ **Hall Ticket No.** | | | | | | | | | | |

Answer All Questions. All Questions Carry Equal Marks. Time: 20 Min. Marks: 10.

I Choose the correct alternative:

1. Symmetric block cipher consists of a sequence of rounds, with each round perform [ ]
   A) Substitutions B) Permutations C) Both D) None

2. In AES Algorithm with the key length of 128 bits no of rounds performed is [ ]
   A) 10 B)12 C)18 D)16

3. _____attacks attempts to alter the system resources. [ ]
   A) passive B) active C) both A&B D)Models Views controller

4. ____ prevents either the sender or receiver from denying a transmitted message. [ ]
   A)confidentiality B)integrity C)authentication D)non-repudiation

5. In RSA algorithm plain text M= [ ]
   A)m^dmod n B)c^dmod n C)m^emod n D)c^emod n

6. _____ is the original message that is fed into the algorithm as input [ ]
   A) Secret Key B)Cipher Text C) Plain Text D) All

7. In DES The substitution consist of set of_____ S-Boxes [ ]
   A) 6 B)4 c)16 d)8

8. _____ Enables two users to exchange the key securely [ ]
   A) Diffie – Hellman B) RSA C) DSS D) DES

9. _____could breach security and cause harm [ ]
   A)Security Service B)Security Attack C) Security Mechanism D) All

10. __is a technique in which the letters of the plain text are replaced by other letters or symbols[ ]
    A) Transposition B)substitution C)Permutation D) A & B

Cont...2

II FIIL IN THE BLANKS

11. A method of breaking cipher text by trying all possible keys called _____.

12. Restoring the plain text from cipher text called_____.

13. RSA stands for_____.

14. DES algorithm uses block length of 64 bits of plain text and key length of _____ bits.

15. _____service is the protection of data from passive attacks

16. In RSA $\varphi(n)$ = _____ .

17. _____ takes place when one entity pretends to be a different entity.

18. In _____ key cipher two different related keys are used.

19. In Diffie – Hellman generation of secret key by user A , K = _____.

20. The message which is encrypted by receiver public key, is decrypted with _____ key.

-000-

Code No: A70522 **Set No. 1**

JNUTH IV B.Tech. I Sem., I Mid-Term Examinations, AUG– 2016
INFORMATION SECURITY Objective Exam Key

1. B
2. D
3. B
4. C
5. A
6. B
7. B
8. C
9. D
10. A

11. Ron Rivest, Adi Shamir, and Leonard Adleman
12. 56
13. Confidentiality
14. Brute force cryptanalysis
15. Decryption
16. $K= y_B^{xA} \bmod q$
17. Private key
18. $\varphi(n) = (p - 1)(q - 1)$
19. masquerade
20. Asymmetric

MID INTERNAL EXAM KEY

Information security: before data processing equipment, the security of info is primarily physical (ex: lock & key systems)

- Computer security: with introduction of computers, need for automated-tools for protecting files & other info stored in compu this, situation arises especially in time sharing systems
- Here need is to access info over public telephone n/w, data n or internet
- network security: this type of security comes into picture for distributed systems, use of n/w's and communication facilitie for carrying data between terminal & comp or b/w comp & co
- Internet security: data processing equipment with a collection of interconnected n/w's is referred to as an internet
- this focuses on measures to determine, prevent, detect and cor security violations that involve the transmission of informatio

Services, Mechanisms and Attacks

To access security needs of an organization effectively and choose correct security protocols and policies

- security Manager to systematically identify requiremen security and characterizing the approaches to satisfy require the three aspects of info security are (ITU-T) international communication union – telecommuni
- Security Attack: action that compromises the security of inf owned by an organization
- Security Mechanism: designed to detect, prevent or recover fron ty attack
- service: enhances the security of the data processing is of an organization, services counter security atta

Services : identification, authorization, signature, notarization, [illegible] receipts, Endorsement, validation, authenticity, ownership, Registrat

Mechanisms: Cryptographic techniques

Attacks : As G.J. Simmons points it, to secure is about how to prev attacks or detect attacks

Security services

Authentication : assurance given that communicating entity is the one that it claims

- Peer Entity authentication
  used in logical connection to provide confidence in the identity of entity

- Data-Origin authentication
  In a connection-less transfer provides assurance that source of received data is as claimed

- Access control

– Prevention of unauthorized use of a resource
ability to limit & control the access to host systems and applicat via communication links

– To achieve this each entity should be identified or authenticat so access rights are given.

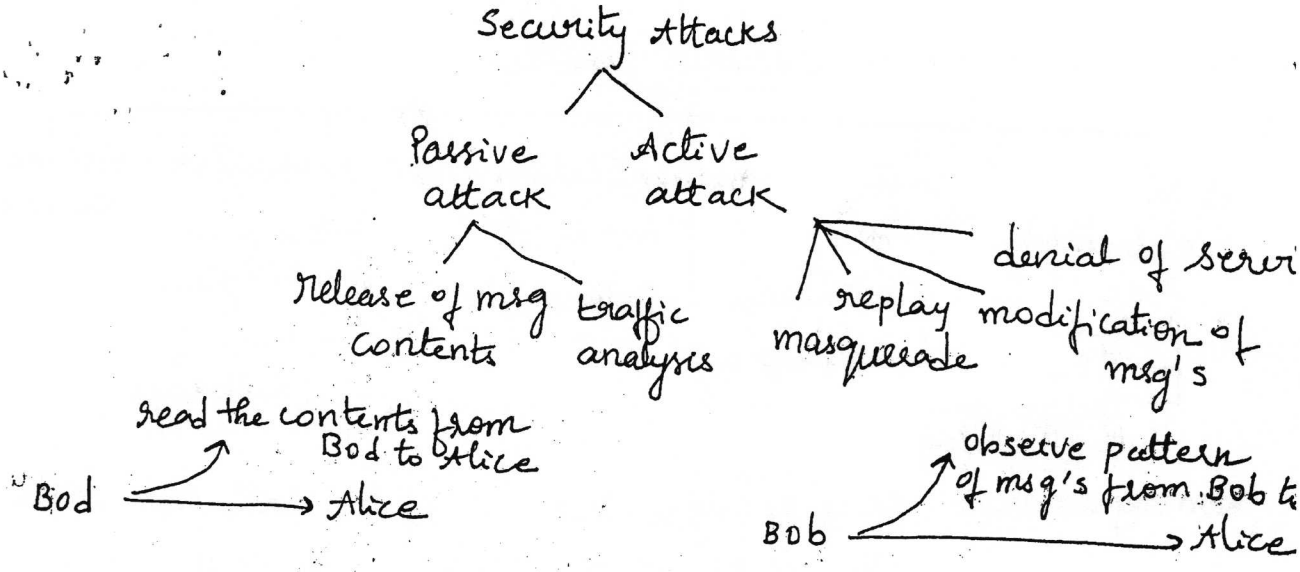
Data confidentiality : protection of data from unauthorized user.

Data integrity : Assurance that data received are exactly as sent by an authorized entity.

The above two apply to a stream of messages, single or selected fields within a message

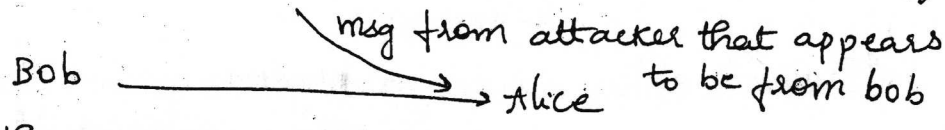
Connection, Connectionless, selective-field and traffic flow.

Nonrepudation : provides protection against denial by one of the entities involved in a communication of having participated in all



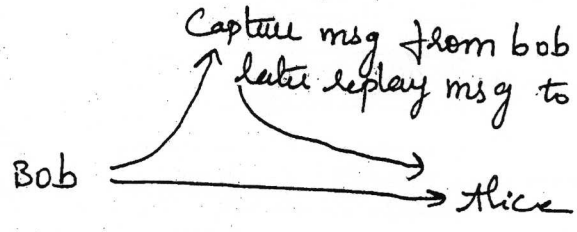
Active

- Masquerade : takes place when one entity pretends to be a diff entity (spoofing)

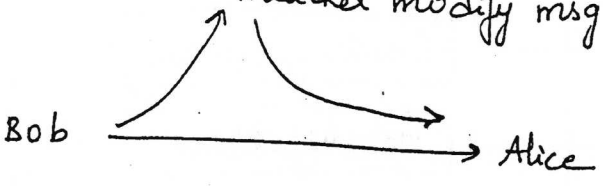


Replay : involves the passive capture of a data unit & its subsequent retransmission to produce an unauthorized effec

~~email snooping~~



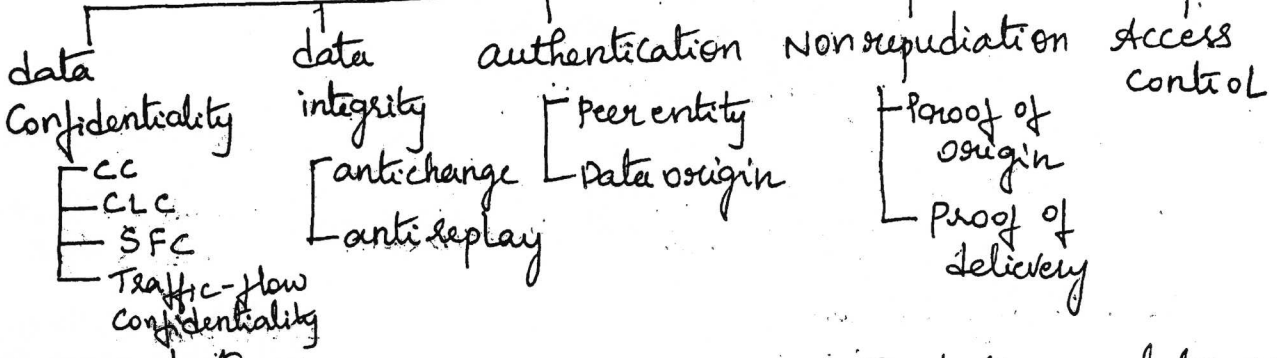
- Modification of msg's : some portion of legitimate msg is altere or that msg's are delayed or recorded to produce an unauth ized- effect



Denial of service : prevents normal use of communication facili entity may supress all msg's directed to paticular destinatio Another form is distruption of an entire n/w neither by disabling the n/w or by overloading it with msg's as to degrade peforma

- Passive are diffi[illegible] to detect

Security services

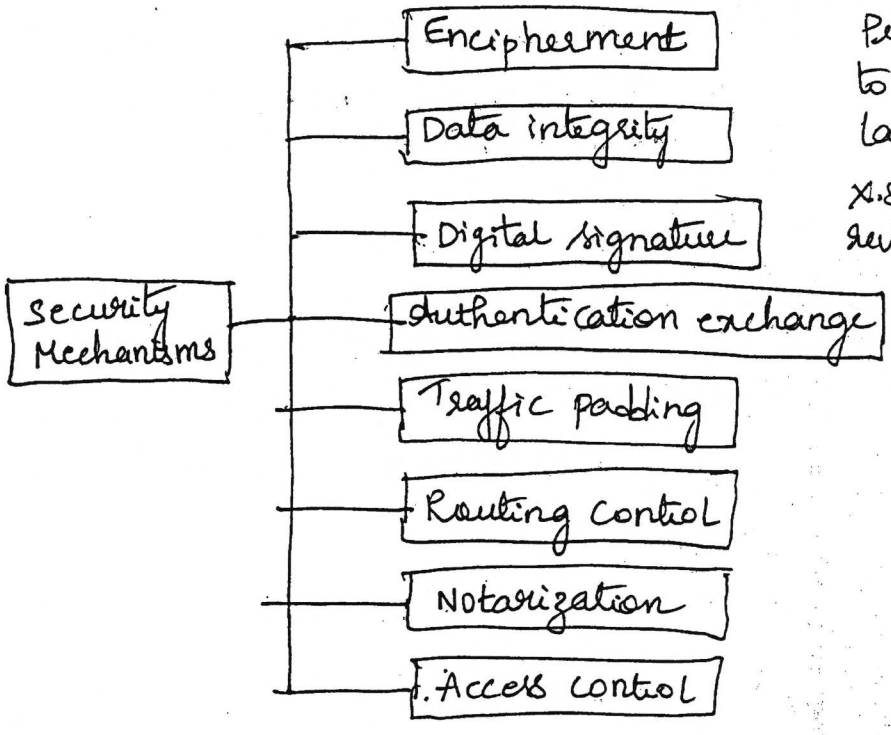


Non repudiation : proof of origin receiver of the data can later prove the identity of the sender if denied

Proof of delievery : the sender of data can later prove that data were delivered to the intended recipient.

(Prevents either sender or receiver from denying a transmission msg.

Security Mechanisms



mechanisms on specific Protocol & not specific to any paticular protoc layer

x.800 distinguished b/w reversible encipherment mechanisms and irre ersable - encipherment mechanisms

1. encipherment : hidding or covering data can provide confidentiality 2 techniques cryptography & stegnography
2. data integrity : appends a checkvalue to data for specific process

Compares the newly checkvalue with the one received, if two checkvalues are same then data integrity has preserved.

Digital signature: means by which the sender can electronica sign the data and the receiver can electonically verify the sign sender uses a process that involves showing that she own's a private key related to the public key that she has announced pu receiver uses sender's public key to prove that msg is indeed signed by the sender who claims to have sent the message.

- Authentication exchange: Two entities exchange some messages to prove their identity to each other.
  She knowns a secret that only she is supposed to known.

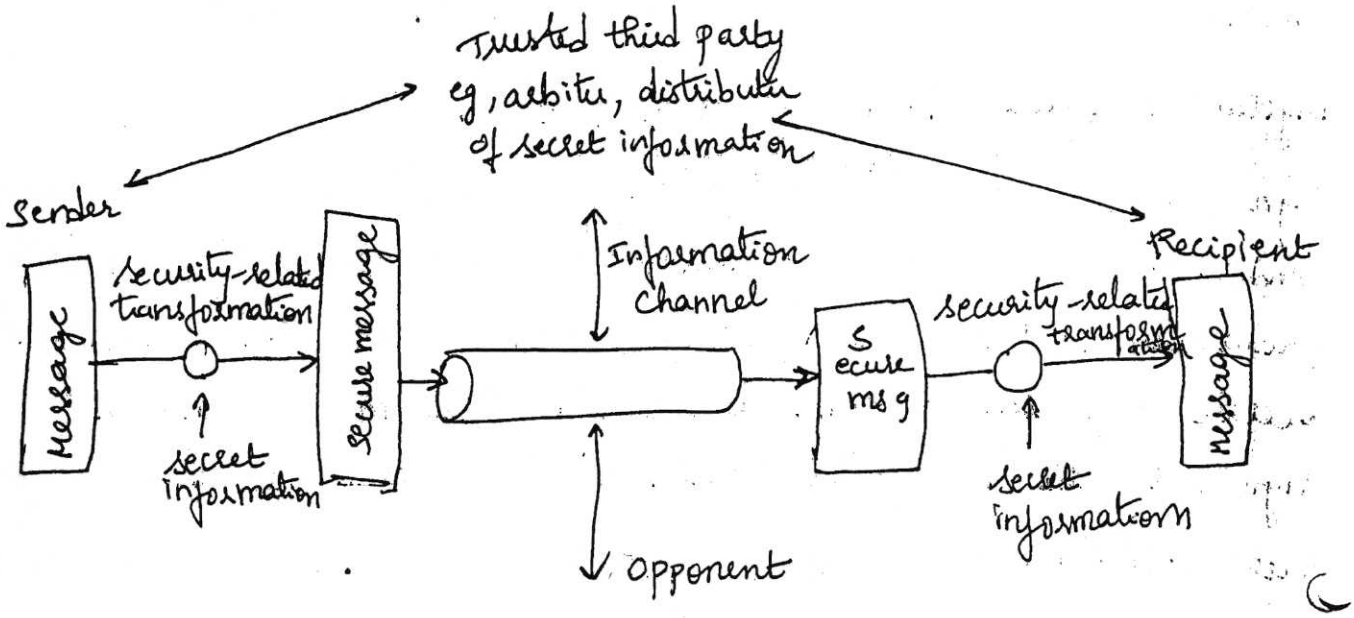
Traffic padding: insert some bogus data into data traffic

Routing control: selecting and continously changing different availiable routes b/w the sender and receiver

- Notarization: selecting third party turstworty to control the co munication- between two entities

Access control: uses methods to prove that a user has access right to the data or resources owned by a system.

Model of n/w security



- A message is to be transferred from one party to another across some sort of internet (called principals)
- Two parties who are the principals must cooperate for exchange
- A logical information channel is established by defining a route through the internet from src to destination and by use of protocol
- Security come into play when it is neccessary to protect the info transmission from an opponent who may present a threat to confidentiality, authenticity & so on.

Techniques to provide security have two components

- Security related transformation on info to be sent
- Some secret information shared by the two principals and it hoped unknown to the opponent

- Trusted third party may be needed to achieve secure transmission

Model shows that there are four basic tasks in designing a paticular security service

50, I, L, N, O, V
not these
Y, Z, G1, 65, 66
67, 6A, 6B, 6C
6f, 65, 6L, 6p

KESHAV MEMORIAL INSTITUTE OF TECHNOLOGY

Narayanguda, Hyderabad.

IV B.Tech – I – Semester –R13- Mid Internal Examinations II- NOV – 2016

Subject: IS

Class & Branch :IV IT **Duration: 60 Min.** **Max. Marks: 10**

Answer any TWO from the following Questions

1. a)Explain HMAC algorithm with neat diagram.
   b) Explain X.509 certificate format in detail
   (or)
   Explain about Kerberos version 4 in detail?

2. Explain PGP in detail.
   (or)
   Write in detail about S/MIME.

3. Discuss about IP security architecture.
   (or)
   Write in detail about OAKLEY key distribution protocol?

4. Write short notes on any two of the following .
   a) Viruses b) Firewalls c) SET d) Intruders

Code No: A70522

Set No. 1

JAWAHARLAL NEHRU TECHNOLOGICAL UNIVERSITY HYDERABAD

IV B.Tech. I Sem., II Mid-Term Examinations, Nov– 2016

INFORMATION SECURITY

Objective Exam

Name: _____ **Hall Ticket No.**_____

Answer All Questions. All Questions Carry Equal Marks. Time: 20 Min. Marks: 10.

I **Choose the correct alternative:**

1. Masquerade is an attack on [ ]

a. Data retrieval b. Authentication c. Non-repudiation d. Data access

2. IPSec is provided at the layer [ ]

a. Below transport Layer b. Below network layer c. At the application layer d. At the physical layer

3. ESP in transport mode encrypts and optionally authenticates which of the following[ ]

a. IP header b. IP payload c. Both IP header and payload d. None

4. Which of the following defines payloads for exchanging key generation and authentication data?[ ]

a. UDP b. HTTP c. ISAKMP d. TCP

5. SSL is implemented over which layer[ ]

a. TCP b. IP c. HTTP d. FTP

6. In which of the following schemes the system checks to see if a password selected by a user is allowable and, if not rejects it. [ ]

a. Proactive b. Reactive c. Active d. Underactive

7. Which of the following malicious programs are independent [ ]

a. Logic bombs b. Trapdoors c. Worm d. Trojan horses

8. What type of protocol is Oakley [ ]

a. Routing protocol b. Transport Protocol c. Key exchange protocol d. Communication Protocol

9. The size of the salt value used in modifying the DES [ ]

Algorithm in UNIX systems is

a. 8 bit b. 12 bit c. 16 bit d. 32 bit

10. Which is a virus that mutates with every infection [ ]

a. Memory-resident virus b. Parasitic virus c. Polymorphic virus d. Stealth Virus

II. FILL IN THE BLANKS

11. _____is an individual who accesses data, programs, or resources for which such access is not authorized.

12. _____ is an open source software package for e-mail security.

13. Email compatibility PGP uses _____ conversion.

14. S/MIME stands for_____.

15._____ header provides data integrity and authentication of IP packets.

16. An SSL _____ is an association between client and server.

17._____ is designed to protect credit card transactions on the internet.

18._____ as identified as intruders.

19. _____ defines a set of rules to decide the behavior of an intruder.

20. A _____ is a program that can replicate itself and send copies from computer to computer across network connections.

Code No: A70522

Set No. 2

JAWAHARLAL NEHRU TECHNOLOGICAL UNIVERSITY HYDERABAD

IV B.Tech. I Sem., II Mid-Term Examinations, Nov– 2016

INFORMATION SECURITY

Objective Exam

Name: _____ **Hall Ticket No.**_____

Answer All Questions. All Questions Carry Equal Marks. Time: 20 Min. Marks: 10.

I Choose the correct alternative:

1. Which of the following defines payloads for exchanging key generation and authentication data? [ ]

a. UDP b. HTTP c. ISAKMP d. TCP

2. SSL is implemented over which layer [ ]

a. TCP b. IP c. HTTP d. FTP

3. In which of the following schemes the system checks to see if a password selected by a user is allowable and, if not rejects it. [ ]

a. Proactive b. Reactive c. Active d. Underactive

4. Which of the following malicious programs are independent [ ]

a. Logic bombs b. Trapdoors c. Worm d. Trojan horses

5. What type of protocol is Oakley [ ]

a. Routing protocol b. Transport Protocol c. Key exchange protocol d. Communication Protocol

6. Masquerade is an attack on [ ]

a. Data retrieval b. Authentication c. Non-repudiation d. Data access

7. IPSec is provided at the layer [ ]

a. Below transport Layer b. Below network layer c. At the application layer d. At the physical layer

8. ESP in transport mode encrypts and optionally authenticates which of the following

a. IP header b. IP payload c. Both IP header and payload d. None [ ]

9. Which is a virus that mutates with every infection [ ]

a. Memory-resident virus b. Parasitic virus c. Polymorphic virus d. Stealth Virus

10. The size of the salt value used in modifying the DES [ ]

Algorithm in UNIX systems is

a. 8 bit b. 12 bit c. 16 bit d. 32 bit

II. FILL IN THE BLANKS

11. S/MIME stands for_____.

12._____ header provides data integrity and authentication of IP packets.

13. An SSL _____ is an association between client and server.

14._____ is designed to protect credit card transactions on the internet.

15. A _____ is a program that can replicate itself and send copies from computer to computer across network connections.

16. _____is an individual who accesses data, programs, or resources for which such access is not authorized.

17. _____ is an open source software package for e-mail security.

18. Email compatibility PGP uses _____ conversion.

19._____ as identified as intruders.

20. _____ defines a set of rules to decide the behavior of an intruder.

Code No: A70522

Set No. 3

JAWAHARLAL NEHRU TECHNOLOGICAL UNIVERSITY HYDERABAD

IV B.Tech. I Sem., II Mid-Term Examinations, Nov– 2016

INFORMATION SECURITY

Objective Exam

Name: _____ **Hall Ticket No.**_____

Answer All Questions. All Questions Carry Equal Marks. Time: 20 Min. Marks: 10.

I **Choose the correct alternative:**

1. The size of the salt value used in modifying the DES [ ]
Algorithm in UNIX systems is

a. 8 bit b. 12 bit c. 16 bit d. 32 bit

2. Which is a virus that mutates with every infection [ ]

a. Memory-resident virus b. Parasitic virus c. Polymorphic virus d. Stealth Virus

3. SSL is implemented over which layer [ ]

a. TCP b. IP c. HTTP d. FTP

4. In which of the following schemes the system checks to see if a password selected by a user is allowable and, if not rejects it. [ ]

a. Proactive b. Reactive c. Active d. Underactive

5. ESP in transport mode encrypts and optionally authenticates which of the following[ ]

a. IP header b. IP payload c. Both IP header and payload d. None

6. Which of the following defines payloads for exchanging key generation and authentication data? [ ]

a. UDP b. HTTP c. ISAKMP d. TCP

7. Which of the following malicious programs are independent [ ]

a. Logic bombs b. Trapdoors c. Worm d. Trojan horses

8. What type of protocol is Oakley [ ]

a. Routing protocol b. Transport Protocol c. Key exchange protocol d. Communication Protocol

9. Masquerade is an attack on [ ]

a. Data retrieval b. Authentication c. Non-repudiation d. Data access

10. IPSec is provided at the layer [ ]

a. Below transport Layer b. Below network layer c. At the application layer d. At the physical layer

II. FILL IN THE BLANKS

11. _____ defines a set of rules to decide the behavior of an intruder.

12. A _____ is a program that can replicate itself and send copies from computer to computer across network connections.

13. _____is an individual who accesses data, programs, or resources for which such access is not authorized.

14._____ header provides data integrity and authentication of IP packets.

15. An SSL _____ is an association between client and server.

16. _____ is an open source software package for e-mail security.

17. Email compatibility PGP uses _____ conversion.

18. S/MIME stands for_____.

19._____ is designed to protect credit card transactions on the internet.

20._____ as identified as intruders.

Code No: A70522

Set No. 4

JAWAHARLAL NEHRU TECHNOLOGICAL UNIVERSITY HYDERABAD

IV B.Tech. I Sem., II Mid-Term Examinations, Nov– 2016

INFORMATION SECURITY

Objective Exam

Name: _____ **Hall Ticket No.**_____

Answer All Questions. All Questions Carry Equal Marks. Time: 20 Min. Marks: 10.

I **Choose the correct alternative:**

1. Which of the following defines payloads for exchanging key generation and authentication data? [ ]

a. UDP b. HTTP c. ISAKMP d. TCP

2. SSL is implemented over which layer[ ]

a. TCP b. IP c. HTTP d. FTP

3. In which of the following schemes the system checks to see if a password selected by a user is allowable and, if not rejects it. [ ]

a. Proactive b. Reactive c. Active d. Underactive

4. Masquerade is an attack on [ ]

a. Data retrieval b. Authentication c. Non-repudiation d. Data access

5. IPSec is provided at the layer [ ]

a. Below transport Layer b. Below network layer c. At the application layer d. At the physical layer

6. ESP in transport mode encrypts and optionally authenticates which of the following[ ]

a. IP header b. IP payload c. Both IP header and payload d. None

7. Which of the following malicious programs are independent [ ]

a. Logic bombs b. Trapdoors c. Worm d. Trojan horses

8. What type of protocol is Oakley [ ]

a. Routing protocol b. Transport Protocol c. Key exchange protocol d. Communication Protocol

9. Which is a virus that mutates with every infection [ ]

a. Memory-resident virus b. Parasitic virus c. Polymorphic virus d. Stealth Virus

10. The size of the salt value used in modifying the DES [ ]

Algorithm in UNIX systems is

a. 8 bit b. 12 bit c. 16 bit d. 32 bit

II. FILL IN THE BLANKS

11. S/MIME stands for_____.

12._____ header provides data integrity and authentication of IP packets.

13. An SSL _____ is an association between client and server.

14._____ is designed to protect credit card transactions on the internet.

15._____ as identified as intruders.

16. _____is an individual who accesses data, programs, or resources for which such access is not authorized.

17. _____ is an open source software package for e-mail security.

18. Email compatibility PGP uses _____ conversion.

19. A _____ is a program that can replicate itself and send copies from computer to computer across network connections.

20. _____ defines a set of rules to decide the behavior of an intruder.

Code No: A70522 **Set No. 1**

JNUTH IV B.Tech. I Sem., II Mid-Term Examinations, NOV– 2016
INFORMATION SECURITY Objective Exam Key

1. B
2. A
3. B
4. C
5. A
6. A
7. C
8. C
9. B
10. C

11. Unauthorized user

12. PGP

13. Radix 64

14. Secure/Multipurpose Internet Mail Extensions
15. Authentication

16. Connection/session

17. SET

18. Attackers

19. Rule based anomaly detection

20. Worm

Keshav Memorial Institute of Technology

DEPARTMENT OF INFORMATION TECHNOLOGY

Assignment 1

IV B Tech - I Semester

Branch: **IT** Acad. Year : 2016-2017

Subject: INFORMATION SECURITY Max Marks: **5M**

Answer all questions. All questions carry equal marks

1. Write about DES with neat diagram and procedure in each round? (course outcome: 1 taxonomy level: 2)

2. Explain about AES algorithm in detail? (course outcome: 1 taxonomy level: 2)

3. Explain the procedure involved in RSA algorithm? Solve the following problem. The public of a given user e=7 and N=187, what is the private key of the user? (course outcome: 1 taxonomy level: 5)

4. Define security attack? Explain in detail about the various types of attacks. (course outcome: 2 taxonomy level: 2)

5. Write down the differences between Symmetric key and Asymmetric key cryptography (course outcome: 2 taxonomy level: 2)

6. Define security mechanism? Explain various mechanisms in details? (course outcome: 2 taxonomy level: 1)

7. Consider the Plain text "KMIT STUDENTS". Convert to cipher text using following Techniques (Solve any three) (course outcome: 1 taxonomy level: 3)
   a) Ceaser Cipher b) Rail – Fence c) Vernam Cipher d) play fair

8. Explain about key distribution in detail (course outcome: 2 taxonomy level: 1)

Keshav Memorial Institute of Technology

DEPARTMENT OF INFORMATION TECHNOLOGY

Assignment 2

IV B Tech - I Semester

Branch: **IT** Acad. Year : 2016-2017

Subject: INFORMATION SECURITY Max Marks: **5M**

Answer all questions. All questions carry equal marks

1. Explain SHA -512 Algorithm in detail (course outcome: 2 taxonomy level: 2)
2. Explain HMAC algorithm with neat diagram (course outcome: 2 taxonomy level: 2)
3. Explain X.509 certificate format in detail (course outcome: 2 taxonomy level: 1)
4. Explain about Kerberos version 4 in detail. (course outcome: 2 taxonomy level: 2)
5. Explain PGP in detail (course outcome: 2 taxonomy level: 2)
6. Write in detail about S/MIME (course outcome: 2 taxonomy level: 1)
7. Discuss about IP security architecture (course outcome: 4 taxonomy level: 1)
8. Write in detail about OAKLEY key distribution protocol? (course outcome: 4 taxonomy level: 2)
9. Explain how security is provided to web (course outcome: 3 taxonomy level: 3)
10. Write short notes on any two of the following . (course outcome: 4 taxonomy level: 2)
    a) Viruses b) Firewalls c) SET d) Intruders e) Digital Signature

E-mail & IP Security

Pretty Good Privacy (PGP)

Operational Description

The PGP has five services

1. authentication
2. confidentiality
3. compression
4. e-mail compatability
5. segmentation.

Authentication

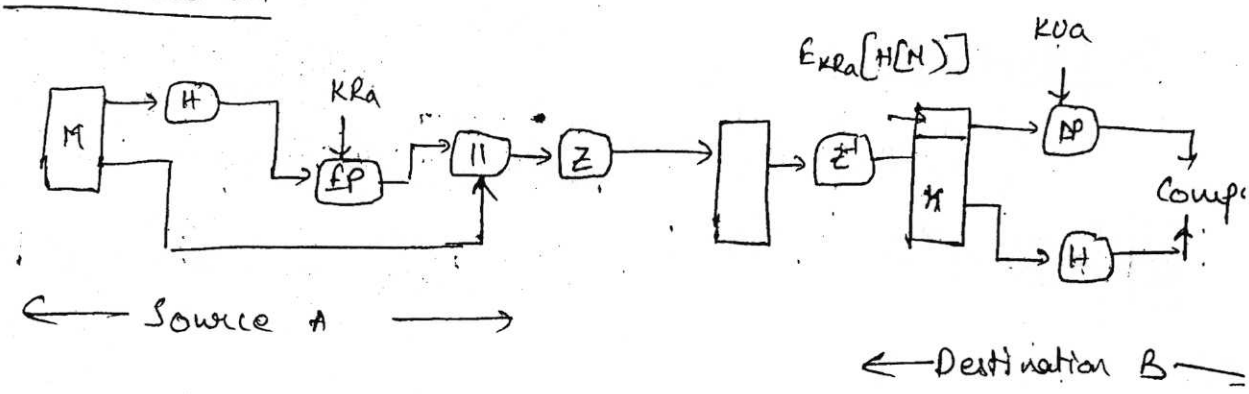


Figure describes the digital signature service provided by PGP. The sequence is as follows.

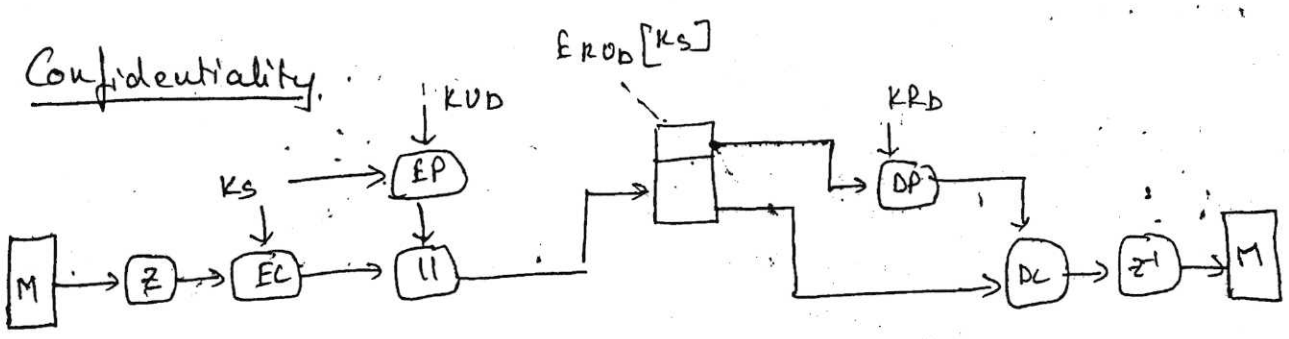
1) The sender creates a message.
2) SHA-1 is used to generate a 160-bit hash code of the message.
3) The hash code is generated encrypted with RSA using the sender's private key and the result is prepended to the message.

4) The receiver uses RSA with the sender's public key to decrypt and recover the hash code.

5) The receiver generates a new hash code for the message and compares it with the decrypted hash code. If the two match, the message is accepted as authentic.

In PGP, the combination of SHA-1 and RSA provides an effective digital signature.

Confidentiality.



The confidentiality is provided by PGP by encrypting messages to be transmitted or to be stored locally as files.

For this PGP uses CAST-128 encryption algorithm (or) IDEA (or) 3DES algorithms, in CFB mode.

The figure describes how PGP offers confidentially in the following steps.

① The sender generates a message and a random 128-bit number to be used as a session key for this message.

② The message is encrypted, using CAST-128 (or IDEA or 3 with the session key.

③ The session key is encrypted with RSA, using receipient public key, and is prepended to the message.

④ The receiver uses RSA with its private key to decry and recover the session key.

⑤ The session key is used to decrypt the message.

Compression

PGP compresses the message after applying signature but before encryption.

The compression function (Z) and Z^{-1} for decompressic The compression used is ZIP.

A signature is generated before compression for

1) It is preferable to sign an uncompressed message so that one can store only uncompressed messa

together with signature for future verification.

2) Message encryption is applied after encryption to strengthen cryptographic security.

E-mail compatability

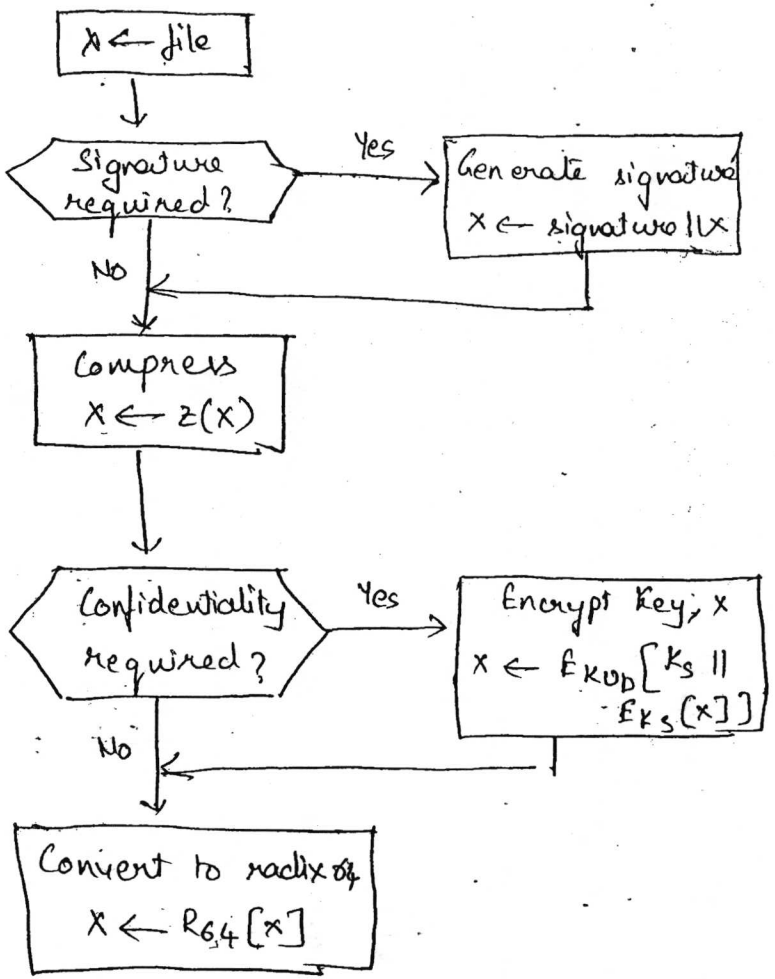
The electronic mail systems only permit the use of blocks consisting of ASCII text. PGP provides the service of converting the raw 8-bit binary stream to a stream of printable ASCII characters.

→ The scheme used to do this is radix 64 conversion.

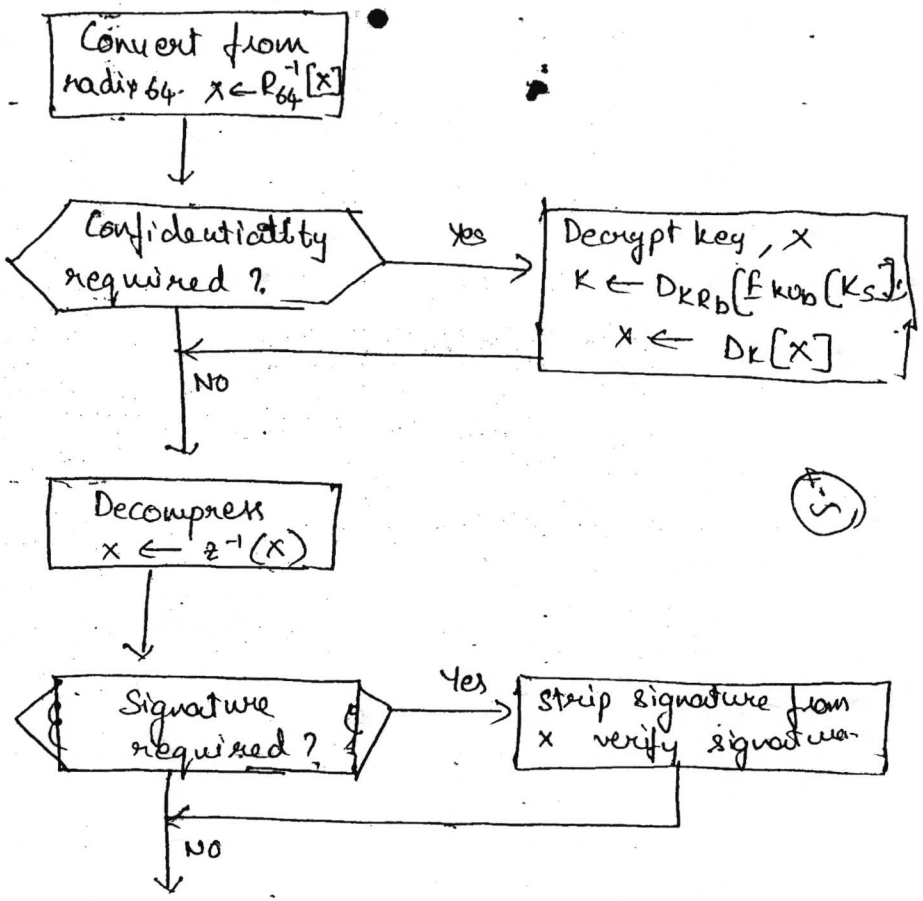
→ Each group of three octets of binary data is mapped into four ASCII characters. This format also appends a CRC to detect transmission errors.

Segmentation and Reassembly

PGP automatically subdivides a message that is too large into segments that are small enough to send via e-mail. The following figure shows the procedure of ~~segmentation and reassembly in PGP~~ transmission and reception of messages in PGP.



(fig) Transmission of messages in PGP



(fig) Reception of messages in PGP.

S/MIME (Secure/Multipurpose Internet Mail Extension)

- It is an extension of MIME internet e-mail standard, based on technology from RSA Data Security. The backgro of traditional e-mail format is RFC 822 standard.

RFC 822

→ The overall structure of a message ~~that~~ in RFC 822 is very simple. A message consists of some number of header lines (the header) followed by unrestricted text (the body). The header is seperated from the body by a blank line.

→ A header line usually consists of a blank line follow by a colon, followed by the keywords arguments.

→ The most frequently used keywords are From, To, Subject and Date.

e.g.; Date: Tue, 16 Jan 1998 10:37:17 (EST)
From: "william stallings" <ws@shore.net>
Subject: The Syntax in RFC 822
To: Smith @ 0ther-host.com
Cc: Jones @ Yet-Another-host.com
← Blank line.
Hello, This section begins the actual

MIME :- The MIME specification includes the following contents.

MIME Header fields

MIME-Version :- Must have a parameter value 1.0.

Content-Type :- Describes the data contained in the bo with sufficient detail that the receiving user agent can pick an appropriate agent to represent data.

Content-Transfer-Encoding :- indicates the type of transfo that has been used to represent the body of the mess in a way that is acceptable for mail transport.

Content-ID :- Used to identify MIME entities unique in multiple contexts.

Content-Description :- A text description of the object with the body ; this is useful when the object is not readable (e.g. audio data)

MIME Content Types

Type	SubType	Description
Text	Plain	Unformatted text (ASCII)
	Enriched	Provides greater format

Flexibility...

Type	Sub-type	Description
Multipart	Mixed	The different parts are independent but are to be transmitted together.
	Parallel	Differs from mixed only in that no order is defined for delivering the parts.
	Alternative	The different parts are alternative versions of the same information.
	Digest	Similar to mixed, but the default type/sub-type of each part is message.
Message	rfc 822	The body is itself an encapsulated message that conforms to RFC 822
	Partial	Used to allow fragmentation of la mail items, in a way that is transpa to the receiptant.
	External-body	Contains a pointer to a object that exists elsewhere.
Image	jpeg	The image is in JPEG format.
	gif	The image is in GIF format.
Video	mpeg	MPEG format.
Audio	basic	Single-channel 8-bit ISDN

Application	PostScript	Adobe Postscript
	Octet-Stream	General binary data consisting of 8-bit bytes.

MIME Transfer Encodings

7 bit	The data are all represented by short lines of ASCII characters.
8 bit	The lines are short, but there may be non-ASCII characters (octets with the high-order bit set
binary	Not only may non-ASCII characters be present, but the lines are not necessarily short enough for SMTP transport.
quoted-printable	Encodes the data in such a way that if the data being encoded are mostly ASCII text, the encoded form of
base64	Encodes data by mapping 6-bit blocks of input to 8-bit blocks of output, all of which are printable ASCII characters.
x-token	A named nonstandard encoding.

Example of MIME message structure

MIME-Version : 1.0
From : NM <nm @ bell.com>
To : AK <ak @ imsoft.com>
Subject : A multipart example.
Content-type : multipart / mixed
boundary : unique-boundary-1.

This is a preamble of the multipart message

-- unique-boundary-1

Some text appears here

-- unique-boundary-1

S/MIME Functionality

S/MIME provides the following functions.

- Enveloped data : This consists of encrypted content of any type and encrypted-content encryption keys for one or more receipients.

- Signed data : A digital signature is formed by taking the message digest of the content to be signed. and encrypting that with the private key of the signer. The content and plus signature are then encoded using base64 encoding.

HHAC objectives

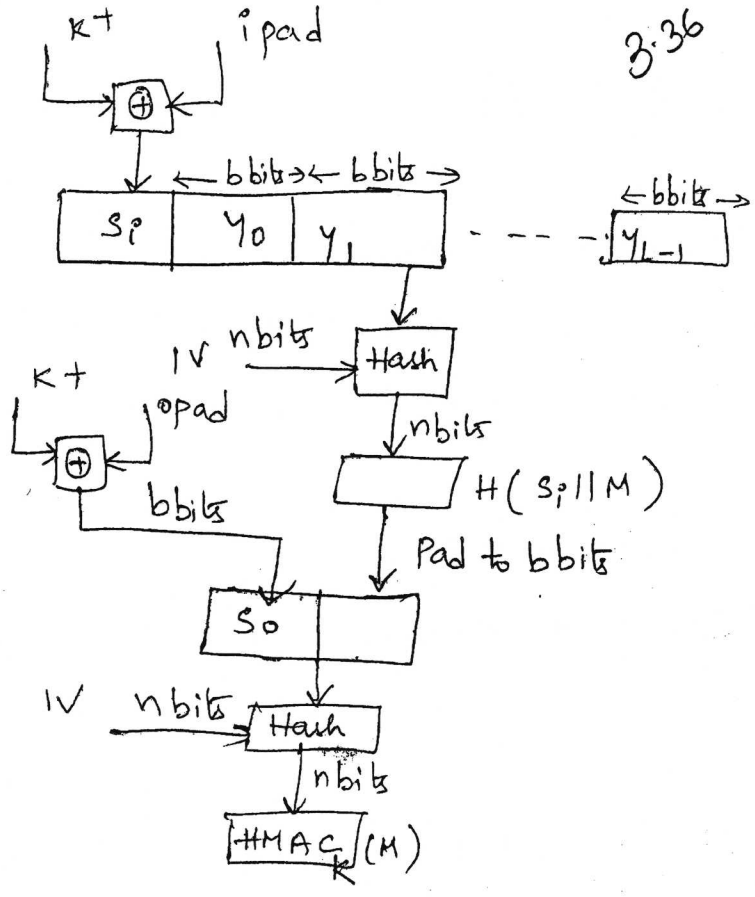
- It is design not an algorithm which can be used for any hash function. like HHAC-MD5, HMAC-AES 3.35
- use hash functions without modifications.
- Allow for easy replace ability of embedded hash function
- perserve original performance of hash func without significar degradation
- uses and handles keys in a simple way
- Has well understood cryptographic analysis of authentication mechanism stiength.

HHAC algorithm.

$$HMAC_K(M) = H\left[K^+ \oplus opdad\right] \| H\left[K^+ \oplus ipad \| M\right]$$

H =
IV =
M =
Y_i =
L =
b =
n = 1
K =
K^+ =
ipad =
Opad =

1. Append zeros to the left end of K to create b-bit string K^+ (eg if K is of length 160 bits and b = 512, then K will be appended with 44 zero bytes 0x00)
2. XOR K^+ with ipad to produce the b-bit block S_i
3. Append M to S_i
4. Apply H to the stream generated in step 3
5. XOR K^+ with opad to produce the b-bit block S_o



K+ — secret key
Pad it with b bits
K — 1024 compress to 512 — no of bits
ipad — fixed
36 repeated $\frac{b}{8}$ times
opad (5c repeated 64 times)

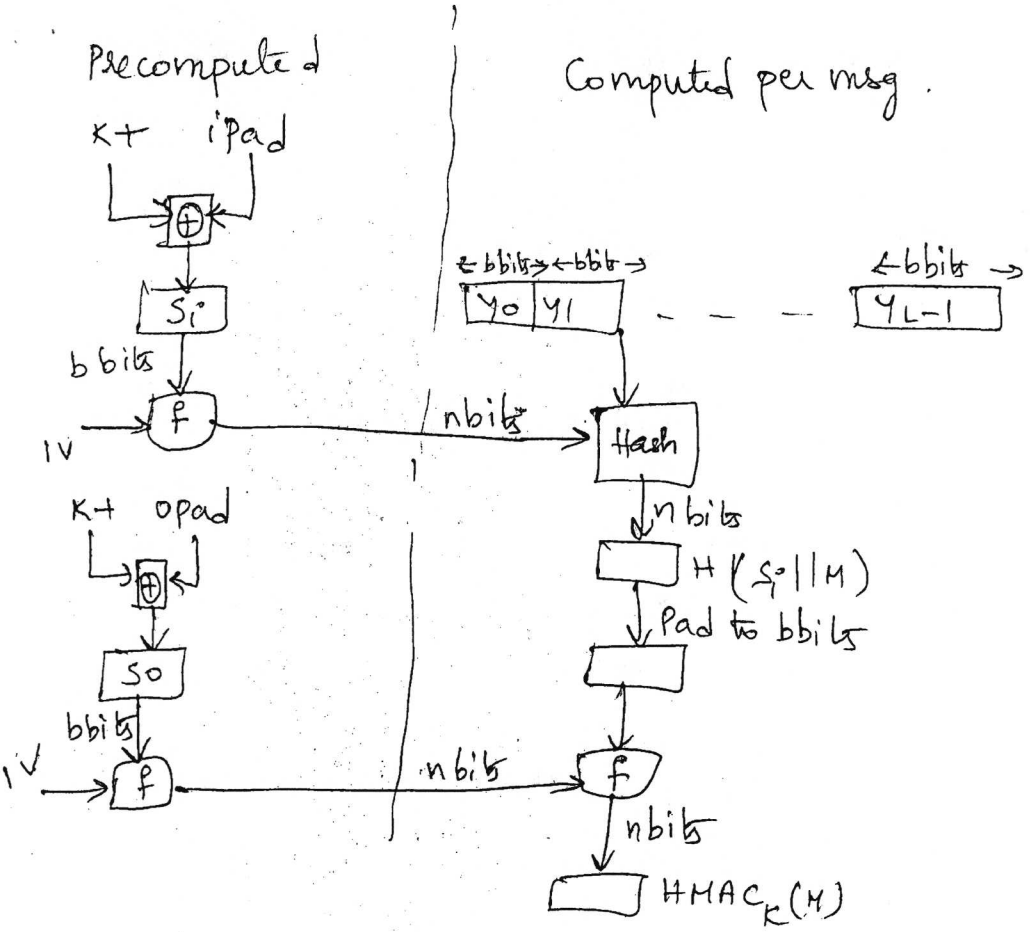
XOR with ipad results in flipping one-half of the bits of K
III^{ly} XOR " opad " " " " " " " " " but a different set of bits, in effect by passing S_i & S_o through the compression function of the hash alg. (we have pseudorandomly generated two keys from K)

A more efficient implementation is possible two quantities are Precomputed

$$f(IV, (K^+ \oplus ipad))$$
$$f(IV, (K^+ \oplus opad))$$

where f (cv, block) is the compression function for the hash func which takes as arguments a chaining variable of n bits & a block b bits & produces a chaining variable of n bits

— These quantities only need to be computed initially & every time the key changes, precomputed quantities substitute for the initial value (IV) in the hash function



27/2/17
CSE-D (presentics)
5G, 5J, 5K, P, Q, R, S
T, W, X, 4, 61, 2, 4, 5, 6
7, 8, 6B, C, D, E, G, H
G5, P, S, U, Z, 7', 72
4, 8, 516, 519

27/2/17 CSE-B
515, 52W, 52V, 507
52Z, 52E, 2P, 536
509, 5CR, 527, 537
525, 538, 52F, 52L

AES (Advanced encryption standard):

finite fields

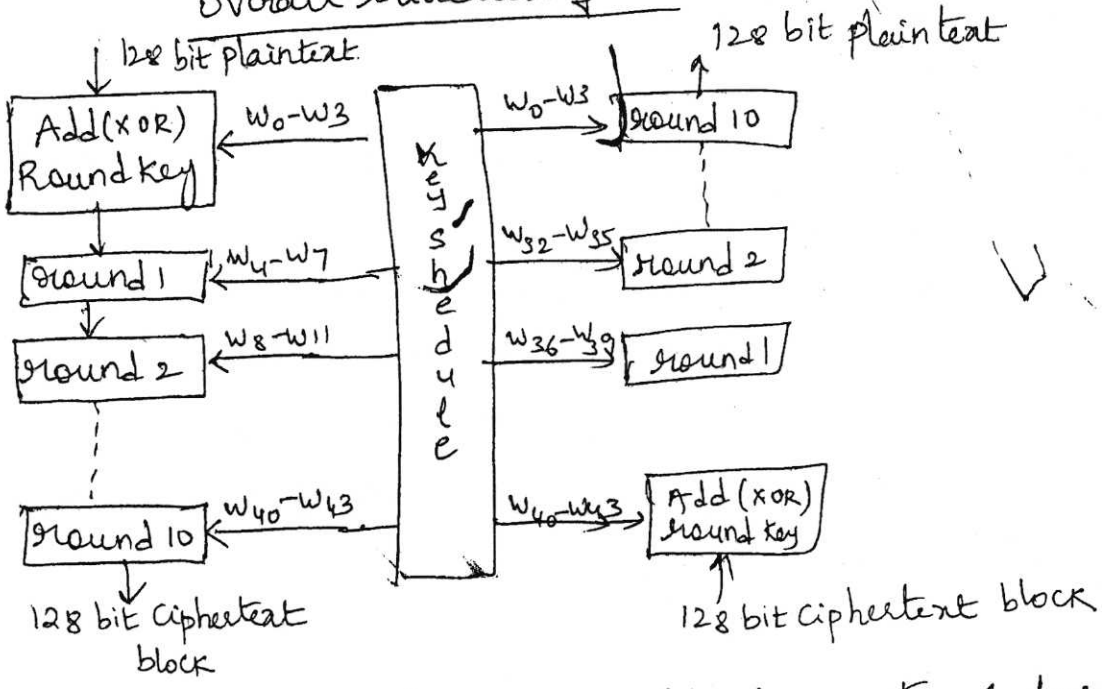
- A finite field is a field with a finite number of elements
- The no of elements in a set is called the order of the field
- A field with order m exists iff m is a prime power i.e, $m = p^n$ for some integer n and with p a prime ~~numb~~ integer
- p is called the characteristics of the finite field
- GF(P) (galwa field) : the elements of the fields can be represented as by 0, 1, . . . p−1
- if p is not prime then multiplications are not defined
- However for finite fields $GF(p^n)$ with $n > 1$, slightly complex represented as polynomials over GF(p).

Polynomial over a field

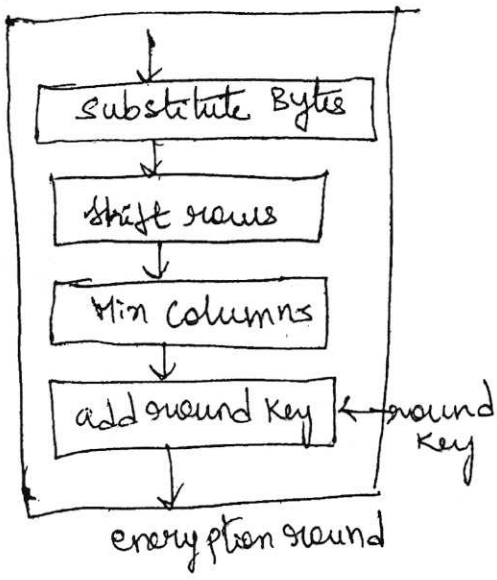
- is an expression of the form $b(x) = b_{m-1}x^{n-1} + b_{m-2}^{n-2} + \cdots + b_0$ x being called ~~inter~~ indeterminate of the polynomial & the $b_i \in F$ the coefficients.
- The degree of a polynomial equals l if $b_j = 0 \; \forall j > l$ and l is the smallest number with this property.
- set of polynomials over a field f is denoted by f(x)
  set of polynomials over a field f which has a degree less than l is denoted by $f[x]_l$

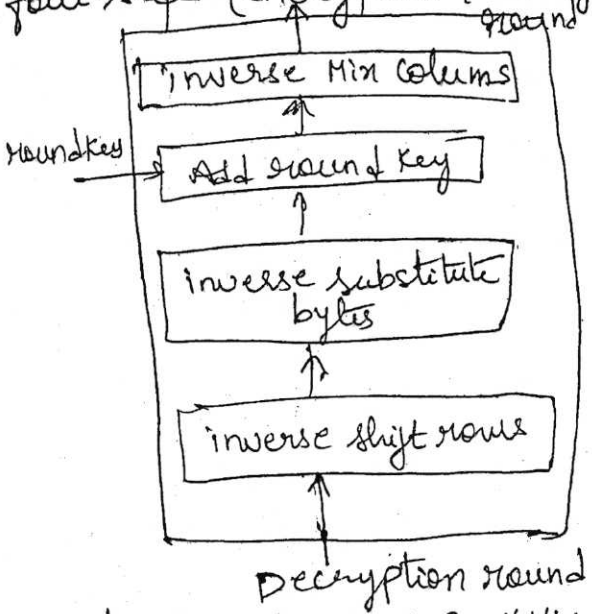
4 (Pg 141) - 1

Overall structure of AES



In each round we have four steps (encryption & decryption)





* : The last round of encryption does not involve the "Mix Columns" step. the last round of decryption does not involve the "inverse mix columns" step.

- last round means round 10 for encryption & round 10 for decryption
- for each round we i/p a state array and get o/p a state array.
- AES input block is 128 bit state array broken down into 4 × 4 matrix and each entry in matrix is byte (so 16 bytes)

A word consists of 4 bytes (32 bits) each column of the state ass is a word.

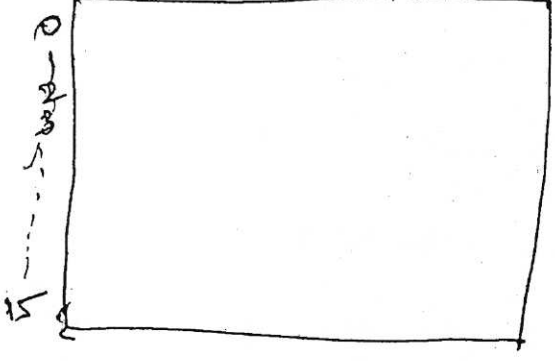
referred to as the state array for each round

byte 0	byte 4	byte 8	byte 12
byte 1	byte 5	byte 9	byte 13
byte 2	byte 6	byte 10	byte 14
byte 3	byte 7	byte 11	byte 15

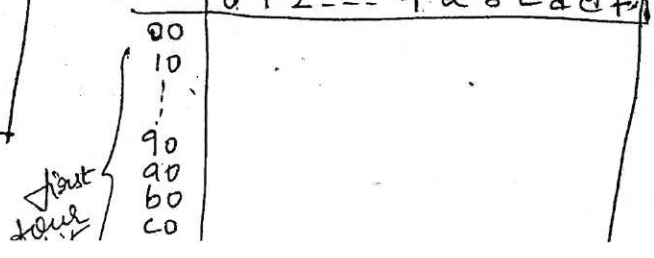
Substitute byte step

- This is a byte-to-byte substitution step using a 16X16 lookup table (whose entry values range from 0 to 255 a byte each)
- same lookup table is used for each byte in all the round
  - one lookup table for subbyte: expansion encryption
  - A different (but related) lookup table for invsubbytes decryption
- The substitution lookup tables are developed based on bit scrambling (a kind of randomization) to reduce the correlation between the input bits and the output bits at the byte level
- To find the substitution for an input byte we break the byte into two four bit units (nibble) use the first 4-bit nibble as the row index and the second 4-bit nibble as the colum index.

Subbytes lookup table (0-255)



We break the bits into 8 bits quantity first four bits as row and next four bytes as column in lookup



ex: suppose we want to replace 41 then see 4 in the row i.e 40 and one in the column 1 the intersection value is 83
same way 9c then 9 in row 90 & c in column i.e de

- same technique for inverse lookup table for decryption

Shift row step

Shift rows transformation (for encryption)

- the first row is NOT shifted
- the second row is shifted one byte to the left
- the third row is shifted two bytes to the left
- the fourth row is shifted three bytes to the left.

Scrambling: As the bytes of the state array are filled column wise, shifting the rows in the manner indicated above scrambles the byte order of the state array and promotes diffusion.

$$\begin{bmatrix} s_{0,0} & s_{0,1} & s_{0,2} & s_{0,3} \\ s_{1,0} & s_{1,1} & s_{1,2} & s_{1,3} \\ s_{2,0} & s_{2,1} & s_{2,2} & s_{2,3} \\ s_{3,0} & s_{3,1} & s_{3,2} & s_{3,3} \end{bmatrix} \Longrightarrow \begin{bmatrix} s_{0,0} & s_{0,1} & s_{0,2} & s_{0,3} \\ s_{1,1} & s_{1,2} & s_{1,3} & s_{1,0} \\ s_{2,2} & s_{2,3} & s_{2,0} & s_{2,1} \\ s_{3,3} & s_{3,0} & s_{3,1} & s_{3,2} \end{bmatrix}$$

Inverse shift rows step

: Inverse shift rows transformation:
- first row is not shifted
- second row is shifted one byte to the right
- third row is shifted two bytes to the right
- the fourth row is shifted three bytes to the right

$$\begin{bmatrix} s_{0,0} & s_{0,1} & s_{0,2} & s_{0,3} \\ s_{1,0} & s_{1,1} & s_{1,2} & s_{1,3} \\ s_{2,0} & s_{2,1} & s_{2,2} & s_{2,3} \end{bmatrix} \Longrightarrow \begin{bmatrix} s_{0,0} & s_{0,1} & s_{0,2} & s_{0,3} \\ s_{1,3} & s_{1,0} & s_{1,1} & s_{1,2} \\ s_{2,2} & s_{2,3} & s_{2,0} & s_{2,1} \end{bmatrix}$$

Mix Columns

- This step replaces each byte of a column by a function of all the bytes in the same column
- All multiplications are according to the GF(2^8) arithmetic and all additions are XOR operations.
- For encryption, the state matrix is multiplied with the follow matrix.

$$\text{Mix columns}\begin{bmatrix} 02 & 03 & 01 & 01 \\ 01 & 02 & 03 & 01 \\ 01 & 01 & 02 & 03 \\ 03 & 01 & 01 & 02 \end{bmatrix} \times \begin{bmatrix} s_{0,0} & s_{0,1} & s_{0,2} & s_{0,3} \\ s_{1,0} & s_{1,1} & s_{1,2} & s_{1,3} \\ s_{2,0} & s_{2,1} & s_{2,2} & s_{2,3} \\ s_{3,0} & s_{3,1} & s_{3,2} & s_{3,3} \end{bmatrix} = \begin{bmatrix} s'_{0,0} & s'_{0,1} & s'_{0,2} & s'_{0,3} \\ s'_{1,0} & s'_{1,1} & s'_{1,2} & s'_{1,3} \\ s'_{2,0} & s'_{2,1} & s'_{2,2} & s'_{2,3} \\ s'_{3,0} & s'_{3,1} & s'_{3,2} & s'_{3,3} \end{bmatrix}$$

State matrix

For decryption the state matrix is multiplied with the following matrix

$$\begin{bmatrix} 0E & 0B & 0D & 09 \\ 09 & 0E & 0B & 0D \\ 0D & 09 & 0E & 0B \\ 0B & 0D & 09 & 0E \end{bmatrix} \times \begin{bmatrix} s_{0,0} & s_{0,1} & s_{0,2} & s_{0,3} \\ s_{1,0} & s_{1,1} & s_{1,2} & s_{1,3} \\ s_{2,0} & s_{2,1} & s_{2,2} & s_{2,3} \\ s_{3,0} & s_{3,1} & s_{3,2} & s_{3,3} \end{bmatrix} = \begin{bmatrix} s'_{0,0} & s'_{0,1} & s'_{0,2} & s'_{0,3} \\ s'_{1,0} & s'_{1,1} & s'_{1,2} & s'_{1,3} \\ s'_{2,0} & s'_{2,1} & s'_{2,2} & s'_{2,3} \\ s'_{3,0} & s'_{3,1} & s'_{3,2} & s'_{3,3} \end{bmatrix}$$

Finite field arithmetic

- also called Galois field arithmetic
- AES uses GF(2^8) arithmetic: all values are in range of 0-255
- We write all values in hex: a byte is written as two hexadecimal values.
- A binary string is represented as a polynomial
  - $00110110 : x^5 + x^4 + x^2 + x$
  - $10010011 : x^7 + x^4 + x + 1$

Addition (XOR): example

$36 + 93 = 00110110 + 10010011$

★ Note: $1+1=0$ hence $x^i + x^i = 0$ for any exponent i

$= (x^5 + x^4 + x^2 + x + x^7 + x^4 + x + 1)$

$= x^5 + x^2 + x^7 + 1 = x^7 + x^5 + x^2 + 1$

$= 10100101 = a5$

finite field ~~arithmetic~~ multiplication

~~(36)(93)~~ $(36)(93) = (00110110)(10010011)$

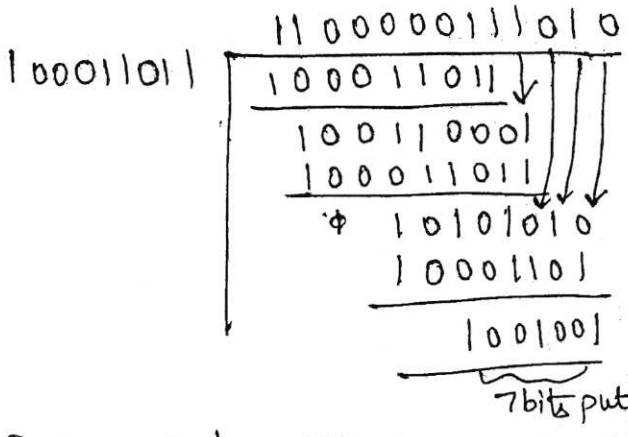
$= (x^5 + x^4 + x^2 + x)(x^7 + x^4 + x + 1)$

$= x^{12} + x^9 + x^8 + x^5 + x^{11} + x^8 + x^5 + x^3 + x^7 + x^4 + x^2 + x^3 + x$
$+ x^8 + x^5 + x^4 + x$

$= x^{12} + x^{11} + x^5 + x^4 + x^3 + x = 1100000111010$

If the degree of the resulting polynomial exceeds 7 we need to do an XOR division with the GF(2^8) reducing polynomial

$x^8 + x^4 + x^3 + x + 1 = 100011011$



Prefix the remainder with sufficient 0's to make it 8bits long

$(36)(93) = 49$

c-12

ex: $(53)(ca) = (01010011)(11001010)$

$= (x^6 + x^4 + x + 1)(x^7 + x^6 + x^3 + x)$

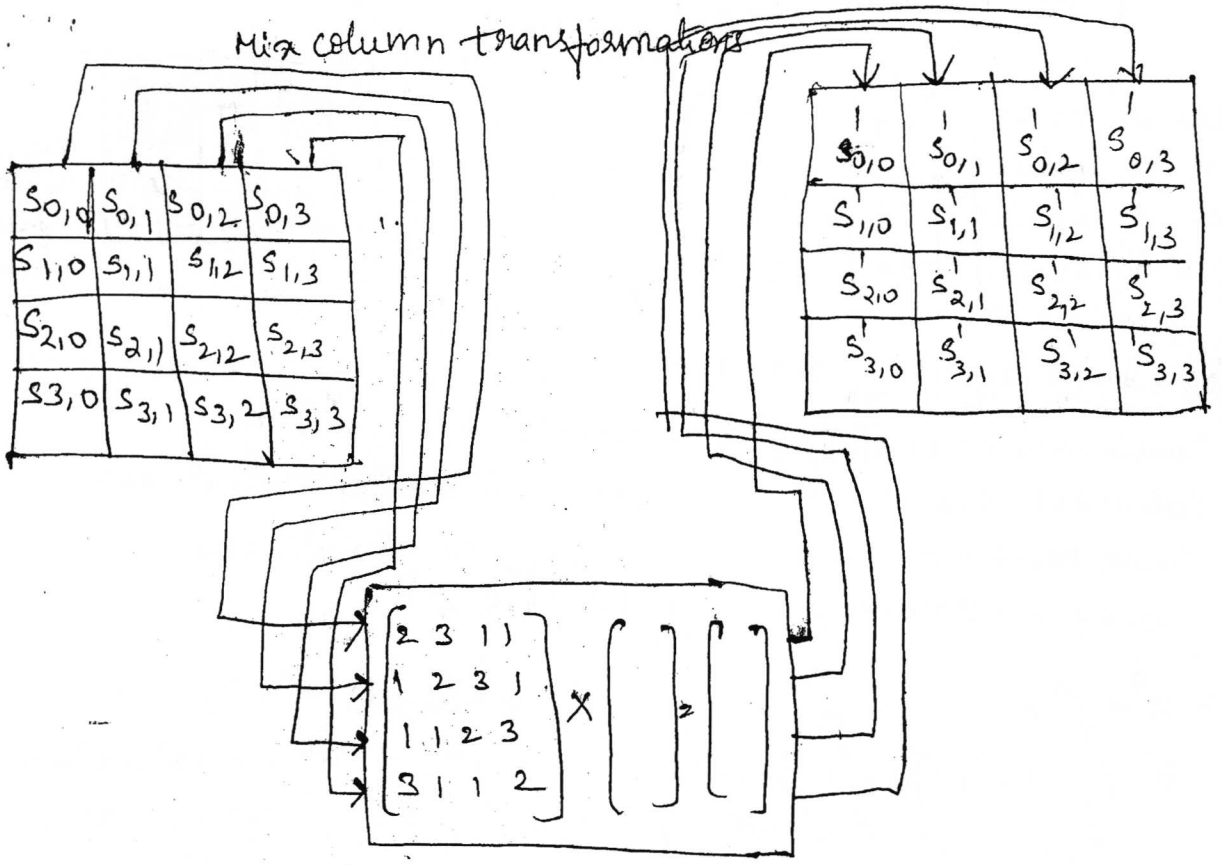
$= x^{13} + x^{12} + x^9 + x^7 + x^{11} + x^{10} + x^7 + x^5 + x^8 + x^7 + x^4 + x^2 +$
$x^7 + x^6 + x^3 + x$

$= x^{13} + x^{12} + x^{11} + x^{10} + x^9 + x^8 + x^6 + x^5 + x^4 + x^3 + x^2 + x$

$= 11111101111110$

divide: $x^8 + x^4 + x^3 + x + 1$

Mix column transformations



$$\begin{bmatrix} 02 & 03 & 01 & 01 \\ 01 & 02 & 03 & 01 \\ 01 & 01 & 02 & 03 \\ 03 & 01 & 01 & 02 \end{bmatrix} \begin{bmatrix} d4 \\ bf \\ 5d \\ 30 \end{bmatrix} = \begin{bmatrix} 04 \\ 66 \\ 01 \\ e5 \end{bmatrix}$$

d4	e0	b8	1e
27	bf	b4	41
11	98	5d	52
ae	f1	e5	30

this the matrix after sub bytes

$(02 * d4) + (03 * bf) + (01 * 5d) + (01 * 30)$

$= (00000010 * 1101\ 0100) +$
$(00000011 * 1011\ 1111) +$
$(00000001 * 0101\ 1101) +$
$(00000001 * 0011\ 0000)$

$= (x)(x^7 + x^6 + x^4 + x^2) +$
$(x+1)(x^7 + x^5 + x^4 + x^3 + x^2 + x + 1) +$
$(1)(x^6 + x^4 + x^3 + x^2 + 1) +$
$(1)(x^5 + x^4)$

$= x^8 + x^7 + x^6 + x^4 + x^2$
$x^8 + x^6 + x^5 + x^4 + x^2 + x$
$x^7 + x^5 + x^4 + x^3 + x$
$x^6 + x^4 + x^3 + x^2 + 1$
$x^5 + x^4$

$= x^8 + x^7 + x^5 + x^3$
$x^8 + x^6 + x^5 + x^4 + x^3 + x^2 + x + 1$
$x^7 + x^5 + x^4 + x^3 + x^2 + x + 1$
$x^6 + x^4 + x^3 + x^2 + 1$
$x^5 + x^4$

$= x^2$

$= 00000\cdot0100$

$= 04$

for second 66

After doing as above we get polynomial expression

$= x^8 + x^6 + x^5 + x^4 + x^3 + x^2 + 1 = 101111101$

exceeds above 7 so reduce using GF(2^8) polynomial

$x^6 + x^4 + x^3 + x + 1 = 100011011$

$$\begin{array}{r|l} 100011011 & 101111101 \\ & 100011011 \\ \hline & 1100110 \end{array}$$

Prefix with sufficient zeros to make the remainder an 8-bit quantity $01100110 = 66$

— Same with inverse mix columns but matrix is

$\begin{bmatrix} 0E & 0B & 0D & 09 \\ 09 & 0E & 0B & 0D \end{bmatrix}$ the encryption & decryption matrices are constan ... these two matrices are

Mix columns

$$\begin{bmatrix} ax+by+cz+dt \\ ex+fy+kz+ht \\ ix+jy+kz+lt \\ mx+ny+oz+pt \end{bmatrix} = \begin{bmatrix} a & b & c & d \\ e & f & g & h \\ i & j & k & l \\ m & n & o & p \end{bmatrix} \times \begin{bmatrix} x \\ y \\ z \\ t \end{bmatrix}$$

new matrix Constant old matrix

- Sub bytes transformation change the value of the byte based only on original value and an entry in the table, the process does not include the neighboring bytes i.e sub bytes is an intrabyte transform
- The permutation provided by shift rows transformation exchange bytes without permuting the bits inside the bytes i.e we can say that shift rows is a byte exchange transformation
- We need interbyte transformation that changes the bits inside the neighboring bytes (we need to mix bytes to provide diffusion at the bit level)
- Mixing transformation changes the contents of each byte by taking four bytes at a time and combining them to recreate four new bytes.

  Combination process first multiplies each byte with a diff constant and then mixes them.

  Mixing provided by matrix multiplication (when we multiply a square matrix by a column matrix the result is a new or matrix each elt in new matrix depends on all the four elts of the old matrix after they are multiplied by row values in the constan matrix.

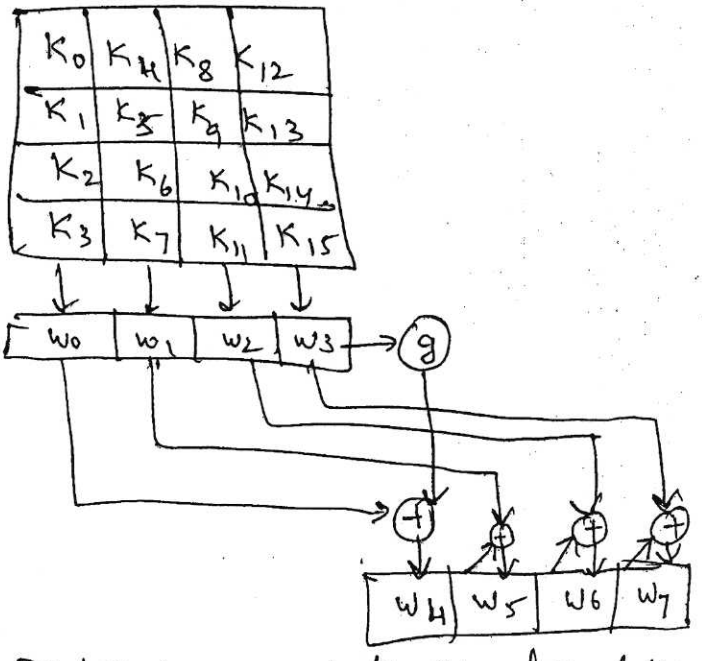
Add round Key Transformation

128 bits of state are bitwise XORed with the 128 bits of the round key operation is viewed as a columnwise operation between the 4 bytes of a state column and one word of the round key

Key expansion

Takes as input a 4 word (16 byte) Key and produces a linear array of 44 words (176 bytes) this is sufficient to provide a 4-word round key for the initial addroundkey stage and each of 10 rounds of the cipher

- Key is copied into first four words of the expanded key remainder of the expanded key is filled in four words at a time w[i] depends on immediately preceding word, w[i-1] and word four positions back, w[i-4]
- The word whose position in the w array is a multiple of 4, a more complex function is used (the g consists of four subfunctions)



1. rot word performs a one byte circular left shift on a word

2. subword performs a byte substitution on each byte of its input word using S-box

3. The result of step 1 & 2 is XORed with a round constant R.con

Round constant is a word in which the three rightmost bytes are always 0, thus effect of an XOR of a word with Rcon is to or Perform an XOR on the leftmost byte of the word

- Rcon is different for each round and defined as

$Rcon[j] = (RC[j], 0, 0, 0)$, with $RC[1] = 1$, $RC[j] = 2 \cdot RC[j-1]$ & with multiplication difined over the field $GF(2^8)$. Values of RC[j] is hexadecimal

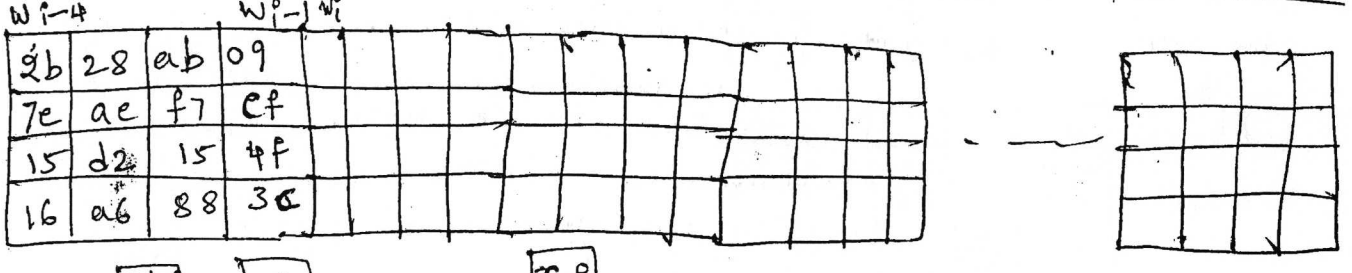
	1	2	3	4	5	6	7	8	9	10
RC[j]	01	02	04	08	10	20	40	80	1B	36

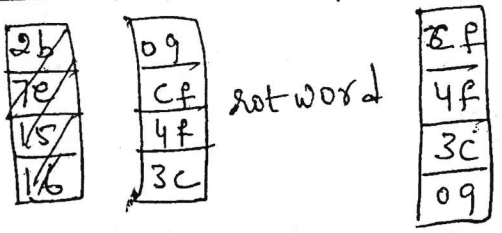
ex: round key for round 8

EA D2 73 21 B5 8D BA D2 31 2B F5 60 7F 8D 29 2F

first 4 bytes of round key for round 9

i(decimal)	temp	After rot word	After subword	Rcon(a)	After XOR with Rcon	w[i-4]	w[i] = w[i-4] ⊕ w[i-1]
36	7F8D292F	8D292F7F	5DA515D2	1B000000	46A515D2	EAD27321	ACC7





Apply S-box for this 4 bytes we get

8a
84
eb

DO XOR with ~~let us say $x^i/3$ of 2^8 so the get~~ the predefined set Rcon we get

01	02	04	08	10	20	40	80	1b	36
00	00	00	00	00	00	00	00	00	00
00	00	00	00	00	00	00	00	00	00
00	00	00	00	00	00	00	00	00	00

2b		8a		01		a0
7e		84		00		fa
15	⊕	eb	⊕	00	=	fe
16		01		01		17

Rcon(4)

- Filling other locations which are not multiple of 4, XOR the 4th previous column with the $i-1$th column so look w_{i-1} and w_{i-4}

Now fill 6th col XOR 5th col with 2nd col

w_{i-4}				w_{i-1}	w_i
2b	28	ab	09	a0	
7e	ae	f7	cf	fa	
15	d2	15	4f	fe	
16	a6	88	3c	17	

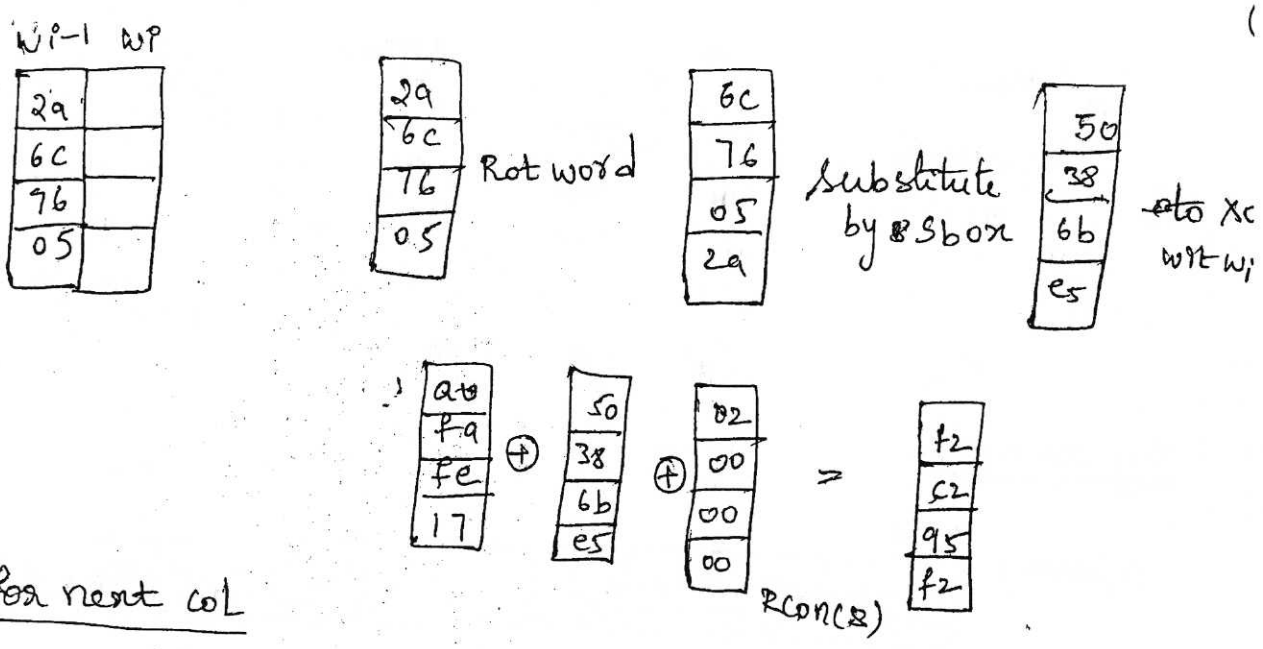
28		a0		88
ae		fa		54
d2	⊕	fe	=	5c
a6		17		b1

Now

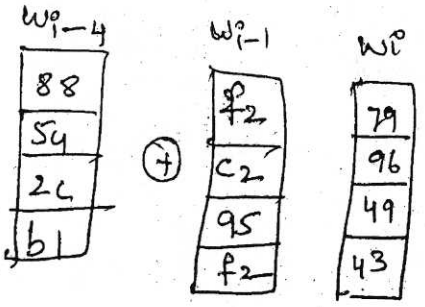
w_{i-4}		w_{i-1}		
ab		88		23
f7	⊕	54	=	a3
15		2c		39
88		b1		39

Now we we want to 8th col since 8 is multiple of 4

w_{i-4}		w_{i-1}		
09		23		2a
cf	⊕	a3	=	6c
				71



for next col



do this until col is not multiple of 4.

So till now

2b	28	ab	09	a0	88	23	2a	f2	79	23	73				
7e	ae	f7	cf	fa	54	a3	6c	e2	96	a3	59				
15	d2	15	4f	fe	2c	39	76	95	b9	39	f6				
16	a6	88	3c	17	b1	39	05	f2	43	39	7f				

cipher key | round 1 | round 2

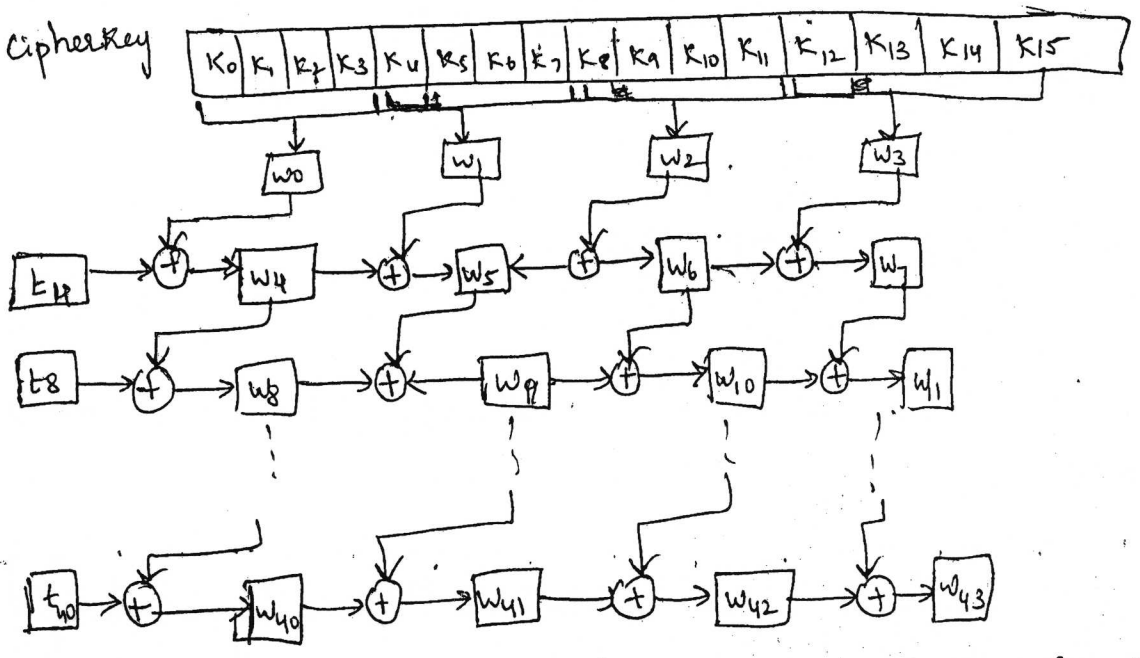
= If the number of rounds is N_r the key expansion routine creat N_r+1 128 bit round key from one single 128 bit cipher key

— Key expansion routine creates round keys word by word where a word is an array of four bytes, the routine creates $4\times(N_r+1)$ wor are called $w_0, w_1, w_2, \ldots w_{4(N_r+1)-1}$

— In AES-128 version (10 rounds) there are 44 words, in AES-192 (12 roun 52 words, AES-256 (14 rounds) 60 words.

Round	Words			
Pre-round	w_0	w_1	w_2	w_3
1	w_4	w_5	w_6	w_7
2	w_8	w_9	w_{10}	w_{11}
...	...			
N_r	w_{4N_r}	w_{4N_r+1}	w_{4N_r+2}	w_{4N_r+3}

Key expansion



1. First four words (w_0, w_1, w_2, w_3) are made from ciphertext, this is thought of as an array of 16 bytes, first four (K_0 to K_3) bytes next four bytes (K_4 to K_7) become w_1 & so on.

2. The rest of words (w_i for $i = 4$ to 43) are made as follows

a) If $(i \bmod 4) \neq 0$ $w_i = w_{i-1} \oplus w_{i-4}$ i.e each word is made from the one at the left and the one at the top.

b) If $(i \bmod 4) = 0$ $w_i = t \oplus w_{i-4}$ Hence t a temporary word, is result of applying two routines SubWord & RotWord on w_{i-1} and XORing the result with a round Constants Rcon

Rearranging terms

$$RE_{i-1} = LE_i$$

$$LE_{i-1} = RE_i \oplus F(RE_{i-1}, K_i) = RE_i \oplus F(LE_i, K_i)$$

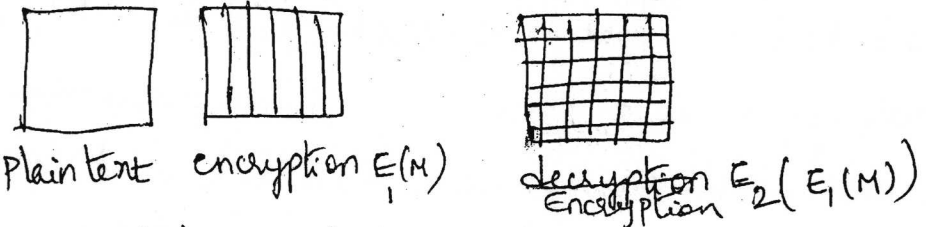
— Thus inputs to the ith iteration as a function of the output and th equations confirm the assignments shown in right hand side

— finally output of the last round of decryption process is $RE_0 || LE_0$ a 32 bit swap recovers the original plaintext

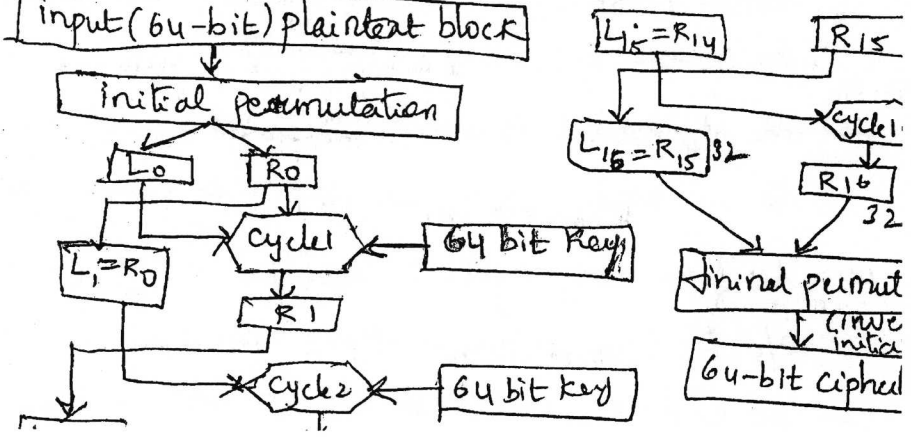
Data encryption standard

- DES algorithm was developed by IBM based on the lucifier algorithm it has been using before.
- DES is a careful and complex combination of two fundamental building blocks of encryption substitution & transposition.
- The algorithm derives the strength from repeated application of these two techniques (16 cycles) one on top of the other.

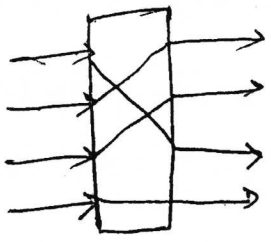
Product cipher: Two complementary ciphers can be made more secure by being applied together alternatively



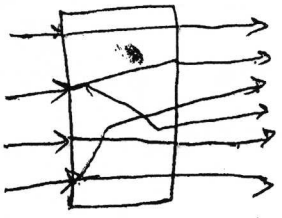
Cycle of substitution and permutation



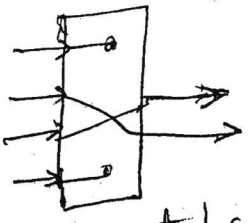
Types of permutations



Permutation



expansion Permutation



Permuted choice

Bit	Goes to position							
1-8	40	8	48	16	56	24	64	32
9-16	39	7	47	15	55	23	63	31
17-24	38	6	46	14	54	22	62	30
25-32	37	5	45	13	53	21	61	29
33-40	36	4	44	12	52	20	60	28
41-48	35	3	43	11	51	19	59	27
49-56	34	2	42	10	50	18	58	26
57-64	33	1	41	9	49	17	57	25

initial permutation

bit in position 1 move to 40, 2 to 8, 3 to 16, --- 8 to 32

- After the initial permutation we break the block into two halves L_0 & R_0 and both are input to cycle. (L_0, R_0 & 64 bit Key)
- Key even though it is a single key, it is processed in different way (we manipulate the key and i/p it into each cycle).
- for paticular cycle we fed in the Key, L_0 & R_0 we go through some processing called feistel network processing we get output as 32-bit quantity.
- the o/p 32-bit quantity is right half (R_1) and right half (R_0) is equal to L_1 and L_1 & R_1 are fed into next cycle as well as key and 64 bit Key
- so for any cycle i we fed in L_{i-1} & R_{i-1} and Key K_i we get o/p R_i (i.e 32-bit output).
- At end of cycle 16 we combile L_{16} & R_{16} each of 32 bit so put

final permutation is according to the table

Bit	Goes to position							
1-8	58	50	42	34	26	18	10	2
9-16	60	52	44	36	28	20	12	4
17-24	62	54	46	38	30	22	14	6
25-32	64	56	48	40	32	24	16	8
33-40	57	49	41	33	25	17	9	1
41-48	59	51	43	35	27	19	11	3
49-56	61	53	45	37	29	21	13	5
57-64	63	55	47	39	39	23	15	7

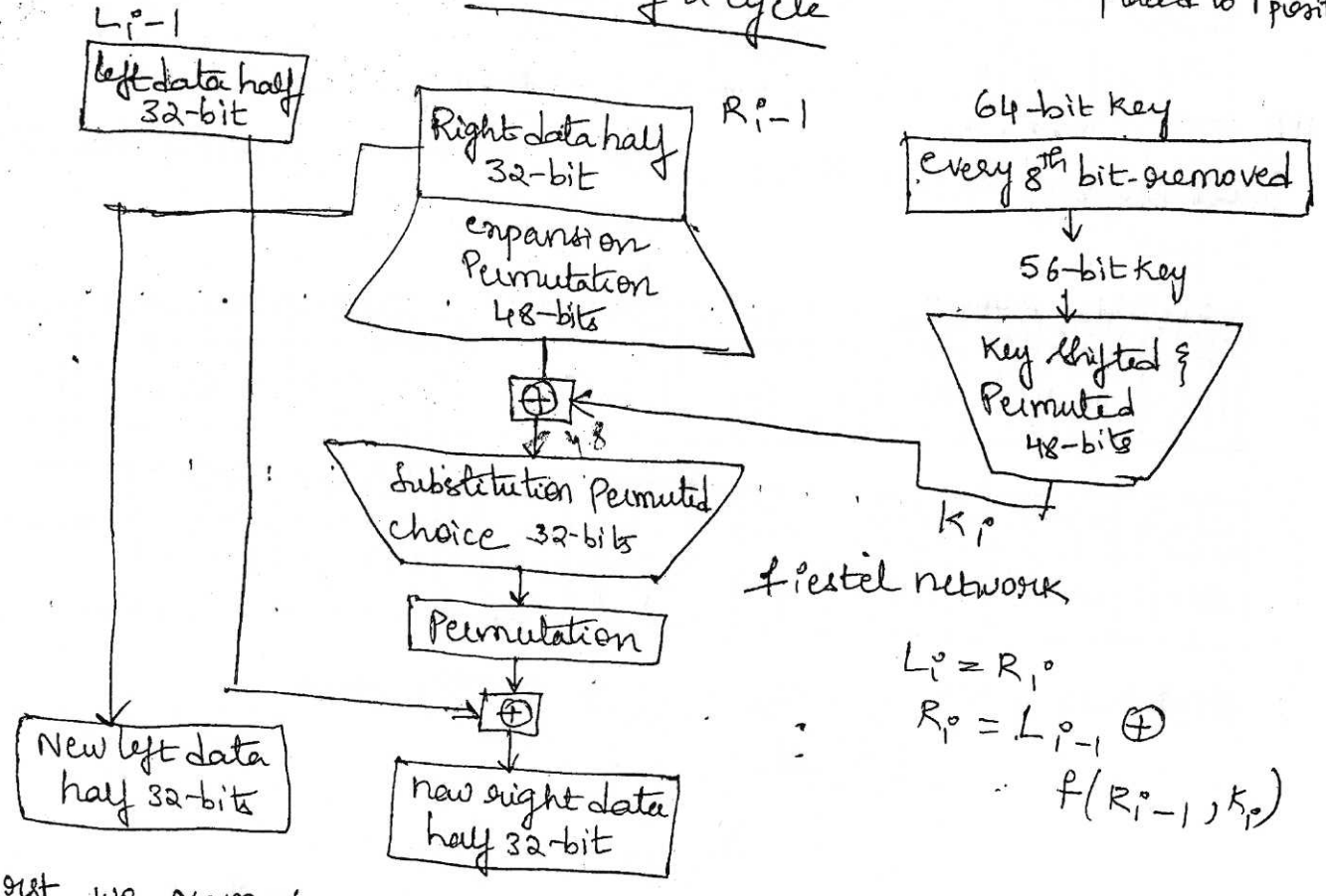
finial permutation.

(in intial permutat
64 bit key ! bit is at position
so in inverse
Permutation, 58 bi
replaced to 1 posit

- the output after finial permutation is

Now what happens inside the cycle

Details of a cycle



Fiestel network

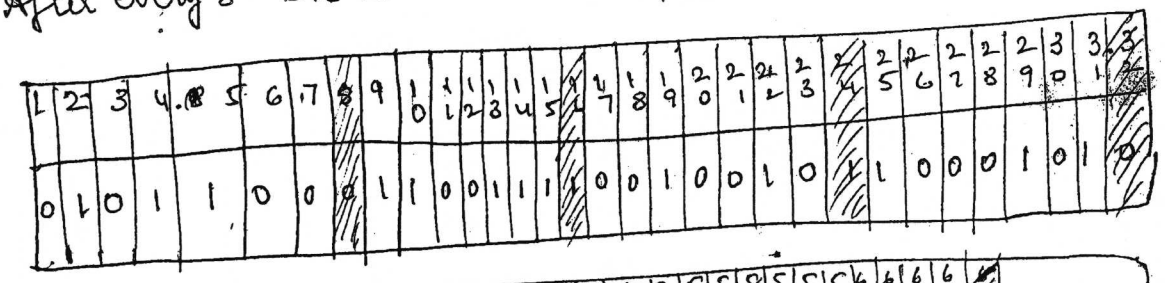
$L_i^\circ = R_1^\circ$

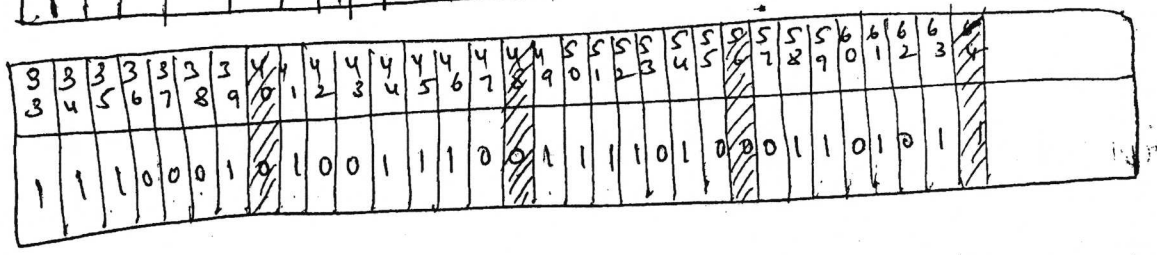
$R_i^\circ = L_{i-1}^\circ \oplus f(R_{i-1}^\circ, K_i^\circ)$

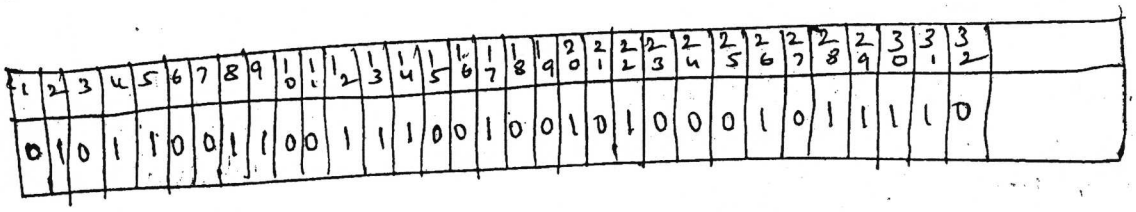
first we remove every 8 bit

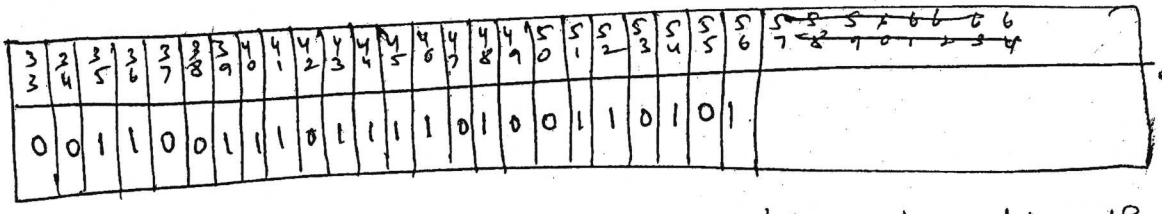
Key shift and permutation (cycle 1)

After every 8th bit removed and split into two 28-bit halves

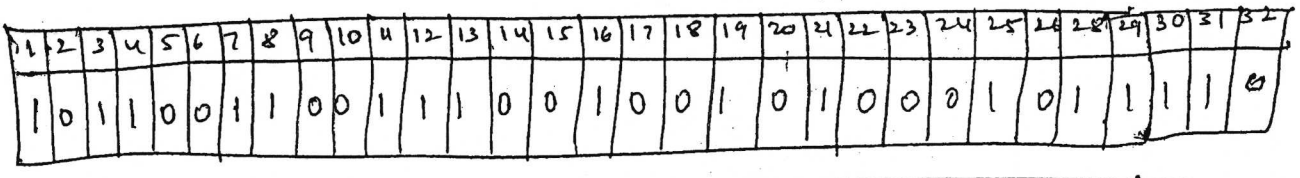


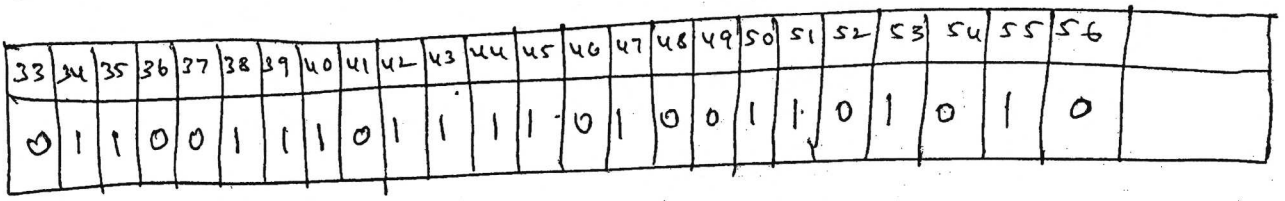






left shifting the two 28 bits halves by 1 bit and putting them together (cycle 1)





How many bits we shift depends on the cycle

cycle	bits shifted
1	1
2	1
3	2
4	2
5	2
6	2
7	2
8	2
9	1
10	2
11	2
12	2
13	2
14	2
15	2
16	1

once shifted goes to permuted choice (kind of permutatio where 64 bits are reduced to 48 bits)

Key bit	1	2	3	4	5	6	7	8	9	10	11	12	13	14
Selected for Position	5	24	7	16	6	10	20	18	–	12	3	15	23	1
Key bit	15	16	17	18	19	20	21	22	23	24	25	26	27	28
Sfp	9	19	2	–	14	22	11	–	13	4	–	17	21	8
Key bit	29	30	31	32	33	34	35	36	37	38	39	40	41	42
Sfp	47	31	27	48	35	41	–	46	28	–	30	32	25	44
Key bit	43	44	45	46	47	48	49	50	51	52	53	54	55	56
Sfp	–	37	34	43	29	36	38	45	33	26	42	–	30	40

56 bits to 48 bits

32	1	2	3	4	5
4	5	6	7	8	9
8	9	10	11	12	13
12	13	14	15	16	17
16	17	18	19	20	21
20	21	22	23	24	25
24	25	26	27	28	29
28	29	30	31	32	1

– the output of the cycle is 48bits, Now there are two halves left right (32 bit), we take right half first and feistel network (i.e cycle), Input for feistel network is R_{i-1} and K_i and, what happens is 32 bit right half goes into expansion permu 48 bits

Bit moves to position	1	2	3	4	5	6	7	8
	2,48	3	4	5,7	6,8	9	10	11,13
Bit	9	10	11	12	13	14	15	16
MTp	12,14	15	16	17,19	18,20	21	22	23,25
Bit	17	18	19	20	21	22	23	24
MTp	24,26	27	28	29,31	30,32	33	34	35,37
Bit	25	26	27	28	29	30	31	32
MTp	36,38	39	40	41,43	42,44	45	46	47,1

Expansion permutation 32 bits to 48 bits

After expansion we XOR 48 bit expansion permutation with 48 bits key, simple bit by bit process after this output is 48 bits, Now these 48 bits goes under substitution permuted choice (3, bit)

for substitution we need sBoxes

example

48 bit input

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23
0	1	1	0	0	0	0	1	0	0	1	0	1	1	0	1	0	1	0	1	1	0	1

25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40	41	42	43	44	45	46	47
0	0	1	1	1	0	1	1	0	1	0	0	1	1	0	1	0	0	1	1	1	0	0

Substitution with permutation

We divide into six bits (divide into 6bit i.e 6x8 = 48 bits)

(Stinson's)

	6-bit input	Row value	Column value	s-box result	4 bit output
S-box S1	011101	1(01)	14(1110)	3	0011
S-box S2	010010	0(00)	9(1001)	7	0111
S-box S3	110101	3(11)	10(1010)	14	1110
S-box S4	011011	1(01)	13(1101)	10	1010
S-box S5	001110	0(00)	7(0111)	6	0110
S-box S6	110100	2(10)	10(1010)	4	0100
S-box S7	110100	2(10)	10(1010)	6	0110
S-box S8	111000	2(10)	12(1100)	15	1111

Row value — is selected from 6-bit put first position & last position bit i.e 01 and the binary value of bits taken are written outside bracket i.e 1(01) and remaining bits in 6bit input after removing first & last bit is 1110 the binary value of this is written in column value along with that bits value

14(1110)

and column value 14 is seen in s-box

we group all 4-bit output

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
0	0	0	1	0	0	1	1	1	1	1	1	0	1	0	1	0

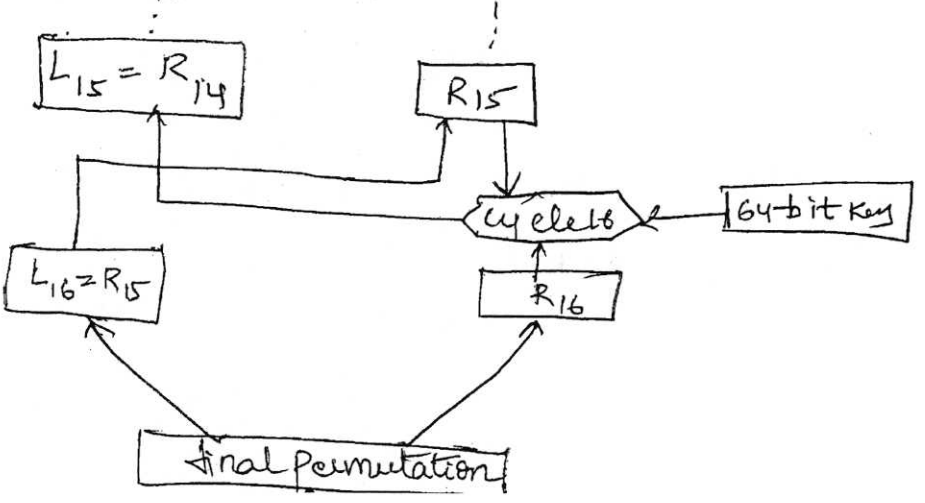
17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32
0	1	1	0	0	1	0	0	0	1	0	0	1	1	1	1

what we get output of S box is 32 bit quantity and that goes to permutation

Bit	Goes to position							
1-8	9	17	23	31	13	28	2	18
9-16	24	16	30	6	26	20	10	1
17-24	8	14	25	3	4	29	11	19
25-32	32	12	22	7	5	27	15	21

- After this X-OR with left half of 32 bit which comes out as output of feistel network and now that is your new right half of 32 bit (R_i) (i.e $R_i = L_{i-1} \oplus f(R_{i-1}, K_i)$ and L_i is simply comes from R_{i-1} (32 bits)

Des decryption



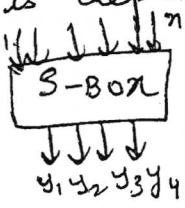
It undergoes permutation ($L_{16} = R_{15}$) same as encryption except that the application of the subkeys is reversed.

(o/p 64-bit plaintext)

$$R_{i-1} = L_i$$
$$L_{i-1} = R_i \oplus f(R_{i-1} = L_i, K_i)$$

Properties of S-Box

- rows are permutations
- inputs are non linear combination of the ~~inputs~~ outputs
- change of one bit of the input and half of the output bits change (avalanche effect)
- each o/p bit is dependent on all the input bits



$$y_1 = f_1(x_1, \ldots x_6) \; ; \; y_2 = f_2(x_1, \ldots x_6) \ldots$$

each of o/p is non-linear

~~Not~~ not only that y_1, y_2 so on are non-linear but also when you X-OR y_1, y_2 so on then also should be non-linear

P-box

INFORMATION SECURITY

TUTORIAL TOPICS

S.NO	Tutorial	Topic
1	TUTORIAL – 1 (TBS-1)	Information Security in Today's World
2	TUTORIAL – 2 (TBS-2)	Current Trends in Data Security
3	TUTORIAL – 3 (TBS-3)	Triple DES
4	TUTORIAL – 4 (TBS-4)	RC5
5	TUTORIAL – 5 (QUIZ)	UNIT-1
6	TUTORIAL – 6 (Test)	UNIT-1&2
7	TUTORIAL – 7 (QUIZ)	Hash Function and MAC
8	TUTORIAL – 8 (TBS -5)	VPN Security
9	TUTORIAL – 9 (Doubts clarification)	PGP

KESHAV MEMORIAL INSTITUTE OF TECHNOLOGY

Department of Information Technology

Result Analysis to identify Weak and advanced learners

Name of the faculty : G BALAKRISHNA **Academic Year:** 2016-17

Branch & Section: IT **Exam:** INTERNAL & EXTERNAL

Subject: INFORMATION SECURITY **Year: IV** **Semester: I**

Sr No	H.T. No	Name	Internal exam marks Average	Student Performance based on Internal exam marks	External Exam Marks	External Exam Total Marks	Overall Rank
1	13BD1A1201	A THODESHWARI	23	A	57	B	A
2	13BD1A1202	AVADHANAM SUBHAKEERTHI	22	A	40	C	B
3	13BD1A1203	B SRAVYA	15	B	1	D	D
4	13BD1A1204	B SRINIVAS	21	A	26	D	C
5	13BD1A1205	BADHURI NIKHIL SAI	19	B	47	C	B
6	13BD1A1206	BODLA SAI KRISHNA	20	A	39	D	C
7	13BD1A1207	BYALYA PHANI RAJ	18	B	13	D	D
8	13BD1A1208	CHAITANYA THAKUR	19	B	36	D	C
9	13BD1A1209	CHIMALAPATI SRIKAR	16	B	16	D	D
10	13BD1A1210	D AKHIL	19	B	39	D	C
11	13BD1A1211	DODLAPATI SHAILAJA	24	A	67	A	A
12	13BD1A1212	G NIDHI RAO	23	A	50	B	B
13	13BD1A1213	G SANJAY	19	B	26	D	C
14	13BD1A1214	GOPAL HULSURE	20	A	42	C	B
15	13BD1A1215	GOUNI TEJASWINI	12	C	13	D	D
16	13BD1A1216	GUNDETI ANITHA	19	B	12	D	D
17	13BD1A1217	HEMA NEEHARIKA P	19	B	10	D	D
18	13BD1A1218	JANARDHAN DESAI	11	C	10	D	D
19	13BD1A1219	K KESAVA KAUSHIK	16	B	26	D	C
20	13BD1A1220	K SIMON JOSEPH	21	A	32	D	C
21	13BD1A1222	KASHI SRAVAN	18	B	31	D	C
22	13BD1A1223	KOTAMARTHY ABIJITH KISHAN	19	B	35	D	C
23	13BD1A1224	M D ABDUL HAFEEZ SHAREEF	18	B	10	D	D
24	13BD1A1225	M SANDEEP	18	B	26	D	C
25	13BD1A1226	MANDAVALLI DURGA LAVANYA	22	A	55	B	B
26	13BD1A1227	MANGU TEJASWINI	19	B	32	D	C
27	13BD1A1228	MEELA PAVAN KUMAR	22	A	55	B	B
28	13BD1A1229	MOYYA ANEESHA	21	A	26	D	C
29	13BD1A1230	N SANDHYA	22	A	49	C	B
30	13BD1A1231	NADARGULU SAI RAGHAVA	22	A	59	B	A

31	13BD1A1232	NIYATI SHAH	21	A	52	B	B
32	13BD1A1234	PANDILLA DIVYA	21	A	51	B	B
33	13BD1A1235	PATLURI SUSMITHA PRIYADARSH	18	B	18	D	D
34	13BD1A1236	PEDDI SOUMYA	22	A	54	B	B
35	13BD1A1237	PIYUSHI V V	18	B	48	C	B
36	13BD1A1238	PRAHARSHITA KRISHNA	20	A	41	C	B
37	13BD1A1239	PURAM HARITHKUMAR	15	B	27	D	C
38	13BD1A1240	R SUSHEEL	17	B	29	D	C
39	13BD1A1241	RAHUL PATIL	20	A	54	B	B
40	13BD1A1242	RANGU SUPRIYA	18	B	39	D	C
41	13BD1A1243	REGONDA PRIYANKA	21	A	41	C	B
42	13BD1A1244	S PRIYANKA	20	A	50	B	B
43	13BD1A1245	S SACHIT REDDY	21	A	27	D	C
44	13BD1A1246	SAGI KAUMUDI	21	A	50	B	B
45	13BD1A1247	SAINI DIVYA SREE	18	B	41	C	C
46	13BD1A1248	SAMUDRALA LAXMI SINDHU	20	A	35	D	C
47	13BD1A1249	SODARI SHIREESHA	18	B	29	D	C
48	13BD1A1250	SONIA JAISWAL	18	B	35	D	C
49	13BD1A1251	SRI RAM SAI TEJA	22	A	49	C	B
50	13BD1A1252	SRIYASRI A	19	B	48	C	B
51	13BD1A1253	TEDDU POOJA	18	B	12	D	D
52	13BD1A1254	UPPALAPATI HARIKA	18	B	19	D	D
53	13BD1A1255	V ANUSHA	23	A	55	B	B
54	13BD1A1256	V GREESHMA	17	B	13	D	D
55	13BD1A1257	VANAM SRAVAN PARASAR	15	B	17	D	D
56	13BD1A1258	VINAYAK MAITREYEE	20	A	58	B	B
57	13BD1A1259	YADLA YESHASWI	19	B	33	D	C
58	13BD1A1260	YALLA AKHIL	18	B	41	C	C

KESHAV MEMORIAL INSTITUTE OF TECHNOLOGY

(Approved by AICTE & Govt of T.S and Affiliated to JNTUH)
3-5-1026, Narayanaguda, Hyderabad-29. Ph: 040-23261407

Department Of Information Technology

IV B.Tech I Sem Regular Examination 2016-17

SUBJECT	FACULTY NAME	STUDENTS APPEARED	STUDENTS PASSED	STUDENTS FAILED	PASS %
DP	MR.NEIL GOGTE	58	40	18	69.0
HCI	B.MANASA	58	49	9	84.5
IRS	DR. RAMAKANTA MOHANTHY	57	50	8	87.7
IS	MR.G.BALA KRISHNA	58	45	13	77.6
MAD	MR.SRINIVAS ADABALA	58	43	15	74.1
BDA	MRS. REKHA	58	32	26	55.2
CTST-L	MS.VIJETHA/MS.PRITI SHAH/MS.SHALMILI	58	58	0	100.0
MAD-L	MRS.SRINIVAS ADABALA/MS. B.MANASA/MS.SUNITHA	58	58	0	100.0

OVERALL REPORT

	NO.OF STUDENTS
Distinction	05
1 st class	22

CLASS TOPPERS

S.NO	REG.NO	STUDENT NAME	%
1	13BD1A1226	MANDAVALLI DURGA LAVANYA	72.8
2	13BD1A1236	PEDDI SOUMYA	72.7
3	13BD1A1201	A THODESHWARI	72.1
4	13BD1A1231	NADARGULU SAI RAGHAVA	71.3
5	13BD1A1211	DODLAPATI SHAILAJA	71.2

Signature of the Commitee **HOD** **Principal**

Keshav Memorial Institute of Technology

Department of Information Technology

Course Outcome Attainment

Name of the faculty : **G BALAKRISHNA** Academic Year: **2016-17**

Branch & Section: IT Exam: **MID I**

Subject: **IS** Year: **IV** Semester: **I**

S.No	HT No.	Question No.												Obj1	A1
		1A	1B	1C	2A	2B	2C	3A	3B	3C	4A	4B	4C		
Max. Marks ==>		5			5			5			5			10	5
1	13BD1A1201	5			5									9	5
2	13BD1A1202	5			4									8	5
3	13BD1A1203	3						3						8	5
4	13BD1A1204	5			4									8	5
5	13BD1A1205	4						3						7	5
6	13BD1A1206	5									4			7	5
7	13BD1A1207	4						4						6	5
8	13BD1A1208	5			4									5	5
9	13BD1A1209	5												6	5
10	13BD1A1210	4			4									6	5
11	13BD1A1211	5			5									8	5
12	13BD1A1212	5			5									7	5
13	13BD1A1213	4						3						6	5
14	13BD1A1214	4			3									6	5
15	13BD1A1215	2												5	5
16	13BD1A1216	5						3						6	5
17	13BD1A1217				4			4						7	5
18	13BD1A1218	5												6	5
19	13BD1A1219	5						2						5	5
20	13BD1A1220	5			4									7	5
21	13BD1A1222	4						2						6	5
22	13BD1A1223	5						2						7	5
23	13BD1A1224	4						3						6	5
24	13BD1A1225	5						1						6	5
25	13BD1A1226	5			4									8	5
26	13BD1A1227	5			3									8	5
27	13BD1A1228	5			5									8	5
28	13BD1A1229	4						4						8	5
29	13BD1A1230	5			4									8	5
30	13BD1A1231	5						5						8	5
31	13BD1A1232	5			5									5	5
32	13BD1A1234	5			5									6	5
33	13BD1A1235	5						3						7	5
34	13BD1A1236	5			5									8	5
35	13BD1A1237	5												5	5
36	13BD1A1238	5			4									6	5
37	13BD1A1239	4												6	5
38	13BD1A1240	3			3									5	5
39	13BD1A1241	5			5									8	5

40	13BD1A1242	5									2			6	5
41	13BD1A1243	5			5									8	5
42	13BD1A1244	5			5									6	5
43	13BD1A1245	5			5									6	5
44	13BD1A1246	5			4									7	5
45	13BD1A1247	5						3						5	5
46	13BD1A1248	5						3						7	5
47	13BD1A1249	5						3						7	5
48	13BD1A1250	4									3			5	5
49	13BD1A1251	5						4						9	5
50	13BD1A1252	5						4						6	5
51	13BD1A1253	5			3									6	5
52	13BD1A1254	3						4						6	5
53	13BD1A1255	5			5									8	5
54	13BD1A1256	5												5	5
55	13BD1A1257	5												5	5
56	13BD1A1258	5			3									5	5
57	13BD1A1259	5						3						7	5
58	13BD1A1260	4									2			6	5
	SUM	265	0	0	115	0	0	66	0	0	11	0	0	382	290
	COUNT	57	0	0	27	0	0	21	0	0	4	0	0	58	58
	AVERAGE	4.649			4.26			3.143			2.75			6.5862	5

CO Mapping with Exam Questions:

CO - 1	**Y**									**Y**			Y	Y
CO - 2				Y									Y	Y
CO - 3							**Y**							
CO - 4														

Students Scored >Target %	56	65	65	31	65	65	17	65	65	3	65	65	58	58
% Students Scored >Target %	86%			48%			26%			5%			89%	89%

CO Attainment based on Exam Questions:

CO - 1	86%									5%			89%	89%
CO - 2				48%									89%	89%
CO - 3							26%							
CO - 4														

CO	Subj	obj	Asgn	Overall	Level
CO-1	45%	89%	89%	75%	3
CO-2	48%	89%	89%	75%	3
CO-3	26%			26%	1
CO-4					

Attainment Level	
1	<40%
2	40-60%
3	>60%

Overall Course Attainment = 2.33333

Keshav Memorial Institute of Technology

Department of IT

Course Outcome Attainment

Name of the faculty : G BALAKRISHNA Academic Year: 2016-17

Branch & Section: IT Exam: MID II

Subject: **IS** Year: IV Semester: I

S.No	HT No.	Question No.												Obj1	A1
		1A	1B	1C	2A	2B	2C	3A	3B	3C	4A	4B	4C		
Max. Marks ==>		5			5			5			5			10	5
1	13BD1A1201	5	3		5									8	5
2	13BD1A1202	5	3		4									8	5
3	13BD1A1203	3						3			2	1	2	AB	5
4	13BD1A1204	5	3		4							1	2	6	5
5	13BD1A1205	4			4			3			1	2	1	5	5
6	13BD1A1206	5	2		4						4			5	5
7	13BD1A1207	4	2		3			4						5	5
8	13BD1A1208	5			4						1	1	2	6	5
9	13BD1A1209	5	3								1	1		6	5
10	13BD1A1210	4			4						2	2	1	6	5
11	13BD1A1211	5	2		5									9	5
12	13BD1A1212	5	2		5									9	5
13	13BD1A1213	4	2		2			3						7	5
14	13BD1A1214	4			3						2	2	1	8	5
15	13BD1A1215	2			3			3						AB	5
16	13BD1A1216	5			3			3				2	1	7	5
17	13BD1A1217	3			4			4			2	2		6	5
18	13BD1A1218	5												AB	5
19	13BD1A1219	5						2						5	5
20	13BD1A1220	5	2		4									8	5
21	13BD1A1222	4	2		3			2						6	5
22	13BD1A1223	5			3			2			2	2	1	6	5
23	13BD1A1224	4						3			2	1	1	6	5
24	13BD1A1225	5						1						7	5
25	13BD1A1226	5	2		4						2	1		9	5
26	13BD1A1227	5	3		3									7	5
27	13BD1A1228	5	2		5									7	5
28	13BD1A1229	4	2					4			2	1	1	7	5
29	13BD1A1230	5	2		4									7	5
30	13BD1A1231	5	2					5						7	5
31	13BD1A1232	5	2		5						1	1	1	8	5
32	13BD1A1234	5			5						2	2	1	7	5
33	13BD1A1235	5	2					3						5	5

34	13BD1A1236	5	2		5						2	1		8	5
35	13BD1A1237	5	2		4									7	5
36	13BD1A1238	5			4						2		1	7	5
37	13BD1A1239	4			3			2						5	5
38	13BD1A1240	3	2		3						2		1	5	5
39	13BD1A1241	5			5			3			2	1		5	5
40	13BD1A1242	5	2					2			2			5	5
41	13BD1A1243	5	2		5									7	5
42	13BD1A1244	5	2		5						2	1	1	6	5
43	13BD1A1245	5	2		5						2	2	1	7	5
44	13BD1A1246	5	2		4						2	1		7	5
45	13BD1A1247	5			4			3			1	1		6	5
46	13BD1A1248	5	2					3			2	1	1	6	5
47	13BD1A1249	5						3				2	2	6	5
48	13BD1A1250	4			4						3	2	2	6	5
49	13BD1A1251	5	2					4			2	1		8	5
50	13BD1A1252	5						4			2	1		7	5
51	13BD1A1253	5			3			2						6	5
52	13BD1A1254	3			2			4			2	2	1	6	5
53	13BD1A1255	5	3		5									8	5
54	13BD1A1256	5		3							2	1		7	5
55	13BD1A1257	5												5	5
56	13BD1A1258	5		4	3						2	2		8	5
57	13BD1A1259	5		3				3			1	1		7	5
58	13BD1A1260	4	1		3						2			6	5
	SUM	268	67	10	160	0	0	78	0	0	61	42	25	364	290
	COUNT	58	31	3	41	0	0	26	0	0	32	30	20	55	58
	AVERAGE	4.621	2.161	3.3	3.9024			3			1.91	1.4	1.25	6.6182	5

CO Mapping with Exam Questions:

CO - 1														
CO - 2	**Y**	**Y**		Y			**Y**							
CO - 3													Y	Y
CO - 4										**Y**	**Y**		Y	Y

Students Scored >Target %	57	57	57	43	57	57	19	57	57	3	57	57	58	58
% Students Scored >Target %	98%	98%	98%	74%			33%			5%	98%		100%	100%

CO Attainment based on Exam Questions:

CO - 1														
CO - 2	98%	98%		74%			33%							
CO - 3													100%	100%
CO - 4										5%	98%		100%	100%

CO	Subj	obj	Asgn	Overall	Level
CO-1					
CO-2	76%			76%	3
CO-3		100%	###	100%	3
CO-4	52%	100%	###	84%	3

Attainment Level	
1	<40%
2	40-60%
3	>60%

Overall Course Attainment = 3

KESHAV MEMORIAL INSTITUTE OF TECHNOLOGY

Department of Information Technology

Course Outcome Attainment

Name of the faculty : **G BALAKRISHNA** ACADEMIC YEAR **2016-17**

Branch & Section: **IT** Exam: **University**

Subject **INFORMATION SECURITY** Year: IV Semester: I

SL.No	REG. NO	NAME OF THE STUDENT	TOTAL
		Max Marks	75.00
1	13BD1A1201	A THODESHWARI	57
2	13BD1A1202	AVADHANAM SUBHAKEERTHI	40
3	13BD1A1203	B SRAVYA	1
4	13BD1A1204	B SRINIVAS	26
5	13BD1A1205	BADHURI NIKHIL SAI	47
6	13BD1A1206	BODLA SAI KRISHNA	39
7	13BD1A1207	BYALYA PHANI RAJ	13
8	13BD1A1208	CHAITANYA THAKUR	36
9	13BD1A1209	CHIMALAPATI SRIKAR	16
10	13BD1A1210	D AKHIL	39
11	13BD1A1211	DODLAPATI SHAILAJA	67
12	13BD1A1212	G NIDHI RAO	50
13	13BD1A1213	G SANJAY	26
14	13BD1A1214	GOPAL HULSURE	42
15	13BD1A1215	GOUNI TEJASWINI	13
16	13BD1A1216	GUNDETI ANITHA	12
17	13BD1A1217	HEMA NEEHARIKA P	10
18	13BD1A1218	JANARDHAN DESAI	10
19	13BD1A1219	K KESAVA KAUSHIK	26
20	13BD1A1220	K SIMON JOSEPH	32
21	13BD1A1222	KASHI SRAVAN	31
22	13BD1A1223	KOTAMARTHY ABIJITH KISHAN	35
23	13BD1A1224	M D ABDUL HAFEEZ SHAREEF	10
24	13BD1A1225	M SANDEEP	26
25	13BD1A1226	MANDAVALLI DURGA LAVANYA	55
26	13BD1A1227	MANGU TEJASWINI	32
27	13BD1A1228	MEELA PAVAN KUMAR	55
28	13BD1A1229	MOYYA ANEESHA	26
29	13BD1A1230	N SANDHYA	49
30	13BD1A1231	NADARGULU SAI RAGHAVA	59
31	13BD1A1232	NIYATI SHAH	52
32	13BD1A1234	PANDILLA DIVYA	51
33	13BD1A1235	PATLURI SUSMITHA PRIYADARSHINI	18
34	13BD1A1236	PEDDI SOUMYA	54
35	13BD1A1237	PIYUSHI V V	48

36	13BD1A1238	PRAHARSHITA KRISHNA	41
37	13BD1A1239	PURAM HARITHKUMAR	27
38	13BD1A1240	R SUSHEEL	29
39	13BD1A1241	RAHUL PATIL	54
40	13BD1A1242	RANGU SUPRIYA	39
41	13BD1A1243	REGONDA PRIYANKA	41
42	13BD1A1244	S PRIYANKA	50
43	13BD1A1245	S SACHIT REDDY	27
44	13BD1A1246	SAGI KAUMUDI	50
45	13BD1A1247	SAINI DIVYA SREE	41
46	13BD1A1248	SAMUDRALA LAXMI SINDHU	35
47	13BD1A1249	SODARI SHIREESHA	29
48	13BD1A1250	SONIA JAISWAL	35
49	13BD1A1251	SRI RAM SAI TEJA	49
50	13BD1A1252	SRIYASRI A	48
51	13BD1A1253	TEDDU POOJA	12
52	13BD1A1254	UPPALAPATI HARIKA	19
53	13BD1A1255	V ANUSHA	55
54	13BD1A1256	V GREESHMA	13
55	13BD1A1257	VANAM SRAVAN PARASAR	17
56	13BD1A1258	VINAYAK MAITREYEE	58
57	13BD1A1259	YADLA YESHASWI	33
58	13BD1A1260	YALLA AKHIL	41
59			
60			
		sum	2046
		Average	37.2

no. of students scored more than target %	28
no. of students present	58
Percentage of students scored more than target %	48%
Attainment level	**2**

Attainment Lev	Percentage
1	<40%
2	40-60%
3	>60%

KESHAV MEMORIAL INSTITUTE OF TECHNOLOGY

Department of Information Technology

Course Outcome Attainment

Name of the faculty : **G BALAKRISHNA** Academic Year: **2016-17**

Branch & Section: **IT** Year: IV Semester: I

Subject: **IS**

Course Outcomes	1st Internal Exam	2nd Internal Exam	University Exam	OVERALL
CO1	3		2	2.25
CO2	3	3	2	2.25
CO3	1	3	2	2
CO4		3	2	2.25

Attainment level of Course Outcomes

	Course Outcomes	Attainment Level
CO1	Understand mathematical foundation required for variouscryptographic algorithms	2.3
CO2	Acquire knowledge in security issues, goals, mechanisms and algorithms.	2.25
CO3	Identify and classify computer and security threats and develop a security model to prevent, detect and recover from attacks.	2
CO4	Independently discover and identify abnormalities within the network caused by worms, viruses and other network related security threats.	2.25

Average 2.2

Overall course attainment level **2**

KESHAV MEMORIAL INSTITUTE OF TECHNOLOGY

Department of Information Technology

Program Outcome Attainment

Name of Faculty: **G BALA KRISHNA**
Branch & Section: **IT**
Subject: **INFORMATION SECURITY**

Academic Year: **2016-17**
Year: **IV** Semester: **I**

Course outcome attainment

CO	Mid-1	Mid-2	AVG	Univ	DIRECT	INDIRECT	OVERALL
CO1	3		3	2	2.25		1.8
CO2	3	3	3	2	2.25		1.8
CO3	1	3	2	2	2		1.6
CO4		3	3	2	2.25		1.8
CO5							
CO6							
ATTAINMENT			2.75	2	2.1875	#DIV/0!	1.75

CO-PO mapping

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	2	2	2									
CO2			1			2						
CO3		3	3		2	2						2
CO4	1	2		3	3							2
CO5												
CO6												
AVERAGE	1.5	2.3333	2	3	2.5	2						2
ATTAINMENT												

Faculty

HEAD OF DEPARTMENT

KESHAV MEMORIAL INSTITUTE OF TECHNOLOGY

Department of Information Technology

Program Outcome Attainment

Name of Faculty: **G BALA KRISHNA** Academic Year: **2016-17**

Branch & Section: **IT** Year: **IV** Semester: **I**

Subject: **INFORMATION SECURITY**

Course outcome attainment

CO	Mid-1	Mid-2	AVG	Univ	DIRECT	INDIRECT	OVERALL
CO1	3		3	2	2.25		1.8
CO2	3	3	3	2	2.25		1.8
CO3	1	3	2	2	2		1.6
CO4		3	3	2	2.25		1.8
ATTAINMENT			2.75	2	2.1875	#DIV/0!	1.75

CO-PO mapping

	PSO1	PSO2
CO1	2	1
CO2	2	1
CO3	3	2
CO4	2	2
AVERAGE	2.25	1.5
ATTAINMENT		

Faculty HEAD OF DEPARTMENT

KESHAV MEMORIAL INSTITUTE OF TECHNOLOGY

3-5-1026, NARAYANAGUDA, HYDERABAD - 500 029

IV YEAR I SEMESTER - I & II MID Marks

BRANCH		INFORMATION TECHNOLOGY					SECTION			--	
SUBJECT NAME		INFORMATION SECURITY									IS
HallTicket No	DATE	8-08-16 - FN				DATE				5-11-16 - FN	
	Name of the Student	I MID				II MID				AVG	Student Sign.
		ASS	SUB	OBJ	Total	ASS	SUB	OBJ	Total		
13BD1A1201	A THODESHWARI	5	10	9	24	5	9	8	22	23	
13BD1A1202	AVADHANAM SUBHAKEERTHI	5	9	8	22	5	8	8	21	22	
13BD1A1203	B SRAVYA	5	6	8	19	5	5	AB	10	15	
13BD1A1204	B SRINIVAS	5	9	8	22	5	8	6	19	21	
13BD1A1205	BADHURI NIKHIL SAI	5	7	7	19	5	8	5	18	19	
13BD1A1206	BODLA SAI KRISHNA	5	9	7	21	5	8	5	18	20	
13BD1A1207	BYALYA PHANI RAJ	5	8	6	19	5	7	5	17	18	
13BD1A1208	CHAITANYA THAKUR	5	9	5	19	5	8	6	19	19	
13BD1A1209	CHIMALAPATI SRIKAR	5	5	6	16	5	5	6	16	16	
13BD1A1210	D AKHIL	5	8	6	19	5	8	6	19	19	
13BD1A1211	DODLAPATI SHAILAJA	5	10	8	23	5	10	9	24	24	
13BD1A1212	G NIDHI RAO	5	10	7	22	5	9	9	23	23	
13BD1A1213	G SANJAY	5	7	6	18	5	7	7	19	19	
13BD1A1214	GOPAL HULSURE	5	7	6	18	5	9	8	22	20	
13BD1A1215	GOUNI TEJASWINI	5	2	5	12	5	6	AB	11	12	
13BD1A1216	GUNDETI ANITHA	5	8	6	19	5	6	7	18	19	
13BD1A1217	HEMA NEEHARIKA P	5	8	7	20	5	7	6	18	19	
13BD1A1218	JANARDHAN DESAI	5	5	6	16	5	AB	AB	5	11	
13BD1A1219	K KESAVA KAUSHIK	5	7	5	17	5	5	5	15	16	
13BD1A1220	K SIMON JOSEPH	5	9	7	21	5	8	8	21	21	
13BD1A1222	KASHI SRAVAN	5	6	6	17	5	8	6	19	18	
13BD1A1223	KOTAMARTHY ABIJITH KISHAN	5	7	7	19	5	7	6	18	19	
13BD1A1224	M D ABDUL HAFEEZ SHAREEF	5	7	6	18	5	6	6	17	18	
13BD1A1225	M SANDEEP	5	6	6	17	5	6	7	18	18	
13BD1A1226	MANDAVALLI DURGA LAVANYA	5	9	8	22	5	7	9	21	22	
13BD1A1227	MANGU TEJASWINI	5	8	8	21	5	5	7	17	19	
13BD1A1228	MEELA PAVAN KUMAR	5	10	8	23	5	9	7	21	22	
13BD1A1229	MOYYA ANEESHA	5	8	8	21	5	9	7	21	21	
13BD1A1230	N SANDHYA	5	9	8	22	5	9	7	21	22	
13BD1A1231	NADARGULU SAI RAGHAVA	5	10	8	23	5	9	7	21	22	
13BD1A1232	NIYATI SHAH	5	10	5	20	5	8	8	21	21	
13BD1A1234	PANDILLA DIVYA	5	10	6	21	5	9	7	21	21	
13BD1A1235	PATLURI SUSMITHA PRIYADARSI	5	8	7	20	5	5	5	15	18	
13BD1A1236	PEDDI SOUMYA	5	10	8	23	5	8	8	21	22	
13BD1A1237	PIYUSHI V V	5	5	5	15	5	8	7	20	18	
13BD1A1238	PRAHARSHITA KRISHNA	5	9	6	20	5	7	7	19	20	
13BD1A1239	PURAM HARITHKUMAR	5	4	6	15	5	5	5	15	15	
13BD1A1240	R SUSHEEL	5	6	5	16	5	7	5	17	17	
13BD1A1241	RAHUL PATIL	5	10	8	23	5	6	5	16	20	
13BD1A1242	RANGU SUPRIYA	5	7	6	18	5	7	5	17	18	
13BD1A1243	REGONDA PRIYANKA	5	10	8	23	5	7	7	19	21	
13BD1A1244	S PRIYANKA	5	10	6	21	5	8	6	19	20	

13BD1A1245	S SACHIT REDDY	5	10	6	**21**	5	9	7	**21**	**21**	
13BD1A1246	SAGI KAUMUDI	5	9	7	**21**	5	8	7	**20**	**21**	
13BD1A1247	SAINI DIVYA SREE	5	8	5	**18**	5	6	6	**17**	**18**	
13BD1A1248	SAMUDRALA LAXMI SINDHU	5	8	7	**20**	5	9	6	**20**	**20**	
13BD1A1249	SODARI SHIREESHA	5	8	7	**20**	5	4	6	**15**	**18**	
13BD1A1250	SONIA JAISWAL	5	7	5	**17**	5	8	6	**19**	**18**	
13BD1A1251	SRI RAM SAI TEJA	5	9	9	**23**	5	7	8	**20**	**22**	
13BD1A1252	SRIYASRI A	5	9	6	**20**	5	6	7	**18**	**19**	
13BD1A1253	TEDDU POOJA	5	8	6	**19**	5	5	6	**16**	**18**	
13BD1A1254	UPPALAPATI HARIKA	5	7	6	**18**	5	7	6	**18**	**18**	
13BD1A1255	V ANUSHA	5	10	8	**23**	5	9	8	**22**	**23**	
13BD1A1256	V GREESHMA	5	5	5	**15**	5	6	7	**18**	**17**	
13BD1A1257	VANAM SRAVAN PARASAR	5	5	5	**15**	5	4	5	**14**	**15**	
13BD1A1258	VINAYAK MAITREYEE	5	8	5	**18**	5	8	8	**21**	**20**	
13BD1A1259	YADLA YESHASWI	5	8	7	**20**	5	5	7	**17**	**19**	
13BD1A1260	YALLA AKHIL	5	6	6	**17**	5	7	6	**18**	**18**	

Signature of Faculty

KESHAV MEMORIAL INSTITUTE OF TECHNOLOGY

YEAR	IV YEAR	SEM	I SEM	BRANCH	IT	SECTION	A
SUBJECT NAME			INFORMATION SECURITY				IS

FORTNIGHT ATTENDANCE		FROM	Jun 13, 2016	Jul 1, 2016	Jul 16, 2016	Aug 1, 2016	Aug 16, 2016	Sep 1, 2016	Sep 16, 2016	Oct 27, 201	TOTAL	% PER
		TO	Jun 30, 2016	Jul 15, 2016	Jul 31, 2016	Aug 15, 2016	Aug 31, 2016	Sep 15, 2016	Oct 4, 2016	Nov 3, 2016		
HT NO	ME OF THE STUDENT	CLASSES HELD	9	4	6	8	6	5	6		44	
13BD1A1201	A THODESHWARI		7	2	4	7	5	4	5		34	77%
13BD1A1202	AVADHANAM SUBHAKEERTHI		6	3	6	6	6	3	6		36	82%
13BD1A1203	B SRAVYA		6	3	4	6	5	3	5		32	73%
13BD1A1204	B SRINIVAS		9	3	6	6	6	4	6		40	91%
13BD1A1205	BADHURI NIKHIL SAI		9	2	4	5	4	3	4		31	70%
13BD1A1206	BODLA SAI KRISHNA		9	3	6	5	6	4	6		39	89%
13BD1A1207	BYALYA PHANI RAJ		6	3	4	6	5	3	5		32	73%
13BD1A1208	CHAITANYA THAKUR		5	3	5	6	5	3	5		32	73%
13BD1A1209	CHIMALAPATI SRIKAR		5	4	4	6	4	3	4		30	68%
13BD1A1210	D AKHIL		6	3	4	6	5	4	5		33	75%
13BD1A1211	DODLAPATI SHAILAJA		6	3	4	7	5	3	5		33	75%
13BD1A1212	G NIDHI RAO		7	4	4	6	4	4	4		33	75%
13BD1A1213	G SANJAY		7	2	4	6	4	4	4		31	70%
13BD1A1214	GOPAL HULSURE		6	3	4	7	5	3	5		33	75%
13BD1A1215	GOUNI TEJASWINI		8	2	4	6	4	3	4		31	70%
13BD1A1216	GUNDETI ANITHA		6	3	4	6	4	4	4		31	70%
13BD1A1217	HEMA NEEHARIKA P		6	3	4	6	5	3	5		32	73%
13BD1A1218	JANARDHAN DESAI		6	3	4	5	5	3	5		31	70%
13BD1A1219	K KESAVA KAUSHIK		5	4	4	6	4	4	4		31	70%
13BD1A1220	K SIMON JOSEPH		6	3	5	6	5	3	5		33	75%
13BD1A1222	KASHI SRAVAN		6	3	4	5	4	4	4		30	68%
13BD1A1223	KOTAMARTHY ABIJITH KISHAN		5	4	4	6	4	3	4		30	68%
13BD1A1224	M D ABDUL HAFEEZ SHAREEF		5	4	4	6	4	4	4		31	70%
13BD1A1225	M SANDEEP		7	2	4	7	5	4	5		34	77%
13BD1A1226	MANDAVALLI DURGA LAVANYA		7	4	4	7	4	3	4		33	75%
13BD1A1227	MANGU TEJASWINI		8	2	4	6	4	4	4		32	73%
13BD1A1228	MEELA PAVAN KUMAR		6	3	4	7	5	4	5		34	77%
13BD1A1229	MOYYA ANEESHA		7	2	4	6	4	3	4		30	68%
13BD1A1230	N SANDHYA		7	4	5	6	5	3	5		35	80%
13BD1A1231	NADARGULU SAI RAGHAVA		7	4	6	7	6	3	6		39	89%
13BD1A1232	NIYATI SHAH		7	2	4	6	4	4	4		31	70%
13BD1A1234	PANDILLA DIVYA		5	4	6	5	6	3	6		35	80%
13BD1A1235	PATLURI SUSMITHA PRIYADARSHINI		5	4	6	6	6	4	6		37	84%
13BD1A1236	PEDDI SOUMYA		5	4	6	7	6	3	6		37	84%
13BD1A1237	PIYUSHI V V		6	3	4	6	5	4	5		33	75%
13BD1A1238	PRAHARSHITA KRISHNA		7	2	4	7	4	3	4		31	70%
13BD1A1239	PURAM HARITHKUMAR		7	2	4	5	5	3	5		31	70%
13BD1A1240	R SUSHEEL		5	4	4	6	4	4	4		31	70%
13BD1A1241	RAHUL PATIL		7	2	4	7	4	4	4		32	73%
13BD1A1242	RANGU SUPRIYA		9	4	6	6	6	3	6		40	91%
13BD1A1243	REGONDA PRIYANKA		9	4	6	6	6	3	6		40	91%
13BD1A1244	S PRIYANKA		9	4	6	6	6	4	6		41	93%
13BD1A1245	S SACHIT REDDY		7	2	4	6	4	3	4		30	68%
13BD1A1246	SAGI KAUMUDI		6	3	4	7	5	4	5		34	77%
13BD1A1247	SAINI DIVYA SREE		7	3	4	7	4	3	4		32	73%
13BD1A1248	SAMUDRALA LAXMI SINDHU		6	3	5	7	5	4	5		35	80%
13BD1A1249	SODARI SHIREESHA		7	2	4	6	4	4	4		31	70%
13BD1A1250	SONIA JAISWAL		7	2	4	6	5	4	5		33	75%
13BD1A1251	SRI RAM SAI TEJA		8	1	4	6	4	3	4		30	68%
13BD1A1252	SRIYASRI A		6	4	4	7	4	3	4		32	73%
13BD1A1253	TEDDU POOJA		7	2	4	6	5	4	5		33	75%
13BD1A1254	UPPALAPATI HARIKA		6	3	5	6	5	3	5		33	75%
13BD1A1255	V ANUSHA		9	4	6	7	6	4	6		42	95%
13BD1A1256	V GREESHMA		6	4	4	6	5	4	5		34	77%
13BD1A1257	VANAM SRAVAN PARASAR		6	3	4	6	4	3	4		30	68%
13BD1A1258	VINAYAK MAITREYEE		6	4	6	6	6	4	6		38	86%
13BD1A1259	YADLA YESHASWI		7	4	4	6	4	3	4		32	73%
13BD1A1260	YALLA AKHIL		9	4	6	6	6	4	6		41	93%
											0	0%
											0	0%
											0	0%
											0	0%
											0	0%

Grams: "TECHNOLOGY"
E Mail: dapjntuh@gmail.com

Phone: Off: +91–40–23156115
Fax: +91–40–23158665

JAWAHARLAL NEHRU TECHNOLOGICAL UNIVERSITY HYDERABAD

(Established by Andhra Pradesh Act No.30 of 2008)
Kukatpally, Hyderabad – 500 085, Andhra Pradesh (India)

B.TECH. (BME, CSE, EEE, ECE, ECM, EIE, ETM, IT/CST, ICE)

R13 COURSE STRUCTURE AND SYLLABUS

I YEAR

Code	Subject	L	T/P/D	C
	English	2	-	4
	Mathematics – I	3	1	6
	Mathematical Methods	3	-	6
	Engineering Physics	3	-	6
	Engineering Chemistry	3	-	6
	Computer Programming	3	-	6
	Engineering Drawing	2	3	6
	Computer Programming Lab.	-	3	4
	Engineering Physics & Engineering Chemistry Lab	-	3	4
	English Language Communication Skills Lab.	-	3	4
	Engineering Workshop / IT Workshop	-	3	4
	Total	**19**	**16**	**56**