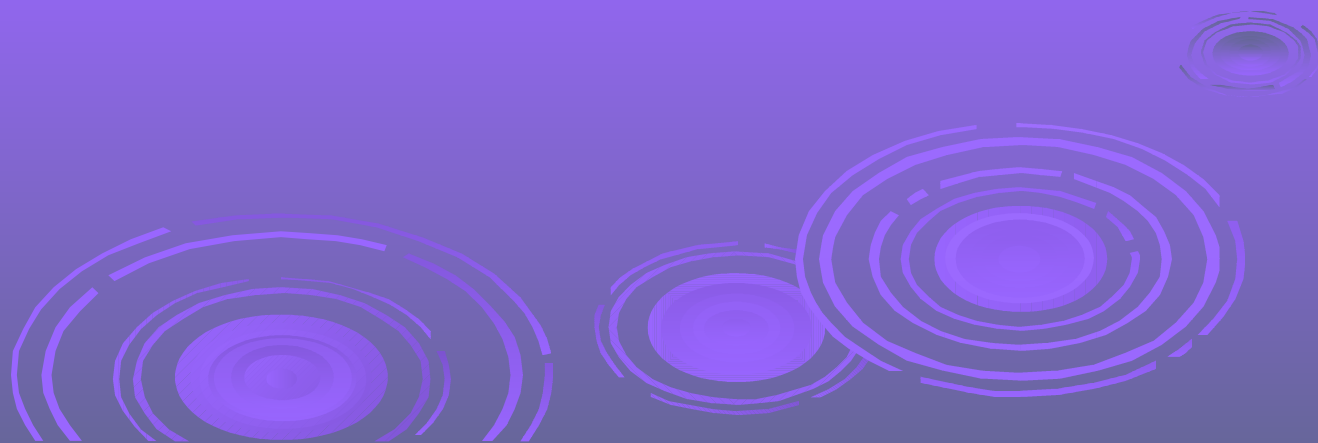


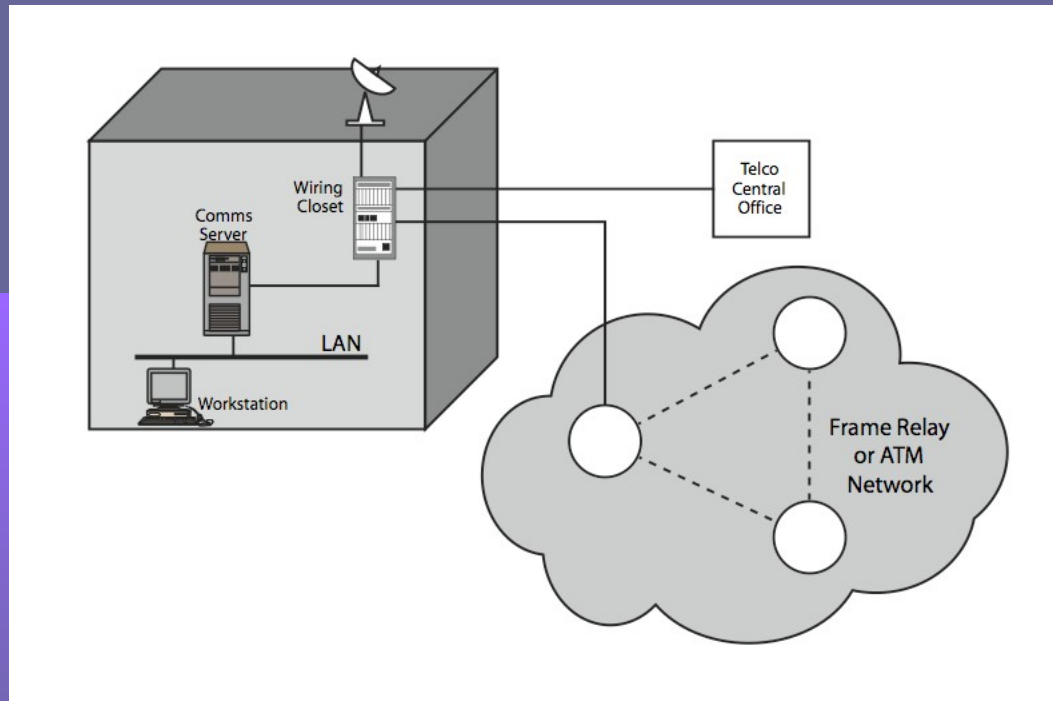
Cryptography and Network Security

Chapter 7



Confidentiality using Symmetric Encryption

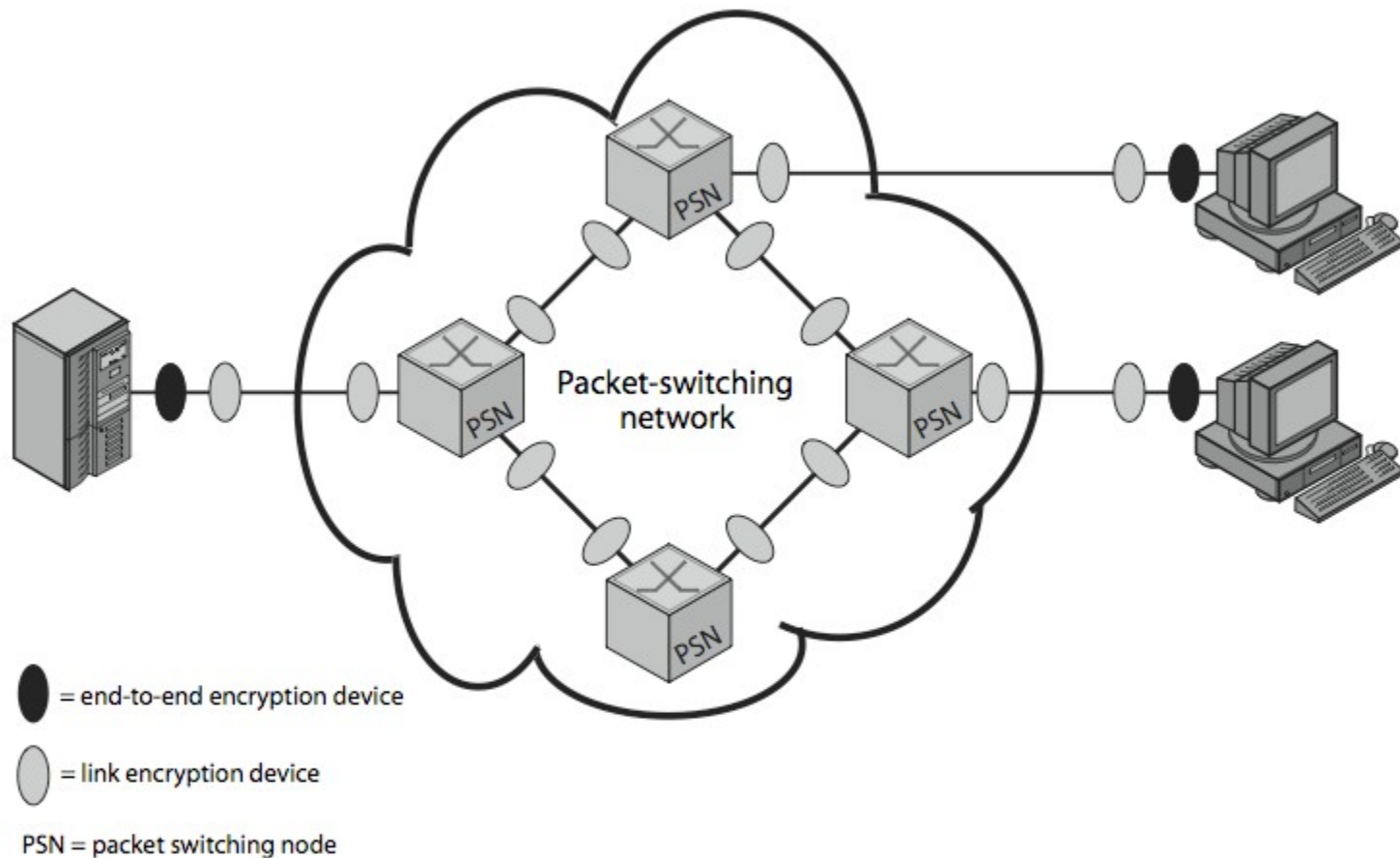
- traditionally symmetric encryption is used to provide message confidentiality



Placement of Encryption

- have two major placement alternatives
- **link encryption**
 - encryption occurs independently on every link
 - implies must decrypt traffic between links
 - requires many devices, but paired keys
- **end-to-end encryption**
 - encryption occurs between original source and final destination
 - need devices at each end with shared keys

Placement of Encryption



Placement of Encryption

- when using end-to-end encryption must leave headers in clear
 - so network can correctly route information
- hence although contents protected, traffic pattern flows are not
- ideally want both at once
 - end-to-end protects data contents over entire path and provides authentication
 - link protects traffic flows from monitoring

Placement of Encryption

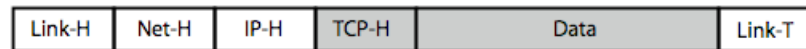
- can place encryption function at various layers in OSI Reference Model
 - link encryption occurs at layers 1 or 2
 - end-to-end can occur at layers 3, 4, 6, 7
 - as move higher less information is encrypted but it is more secure though more complex with more entities and keys



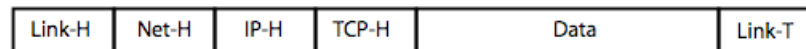
Encryption vs Protocol Level



(a) Application-Level Encryption (on links and at routers and gateways)

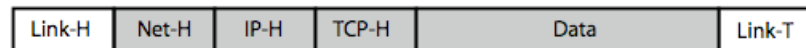


On links and at routers

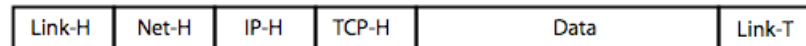


In gateways

(b) TCP-Level Encryption



On links



In routers and gateways

(c) Link-Level Encryption

Shading indicates encryption.

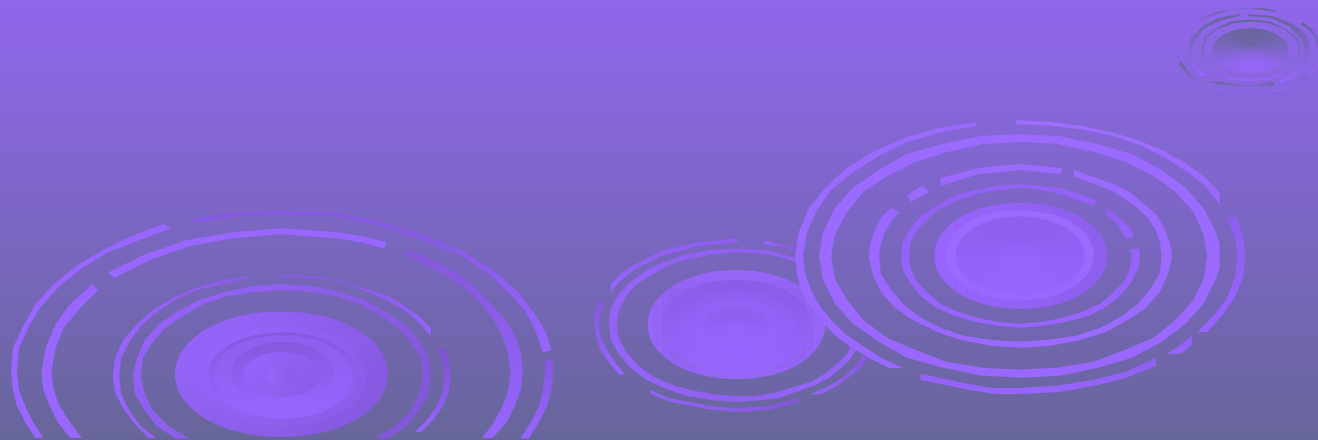
TCP-H = TCP header
IP-H = IP header
Net-H = Network-level header(e.g., X.25 packet header, LLC header)
Link-H = Data link control protocol header
Link-T = Data link control protocol trailer

Traffic Analysis

- is monitoring of communications flows between parties
 - useful both in military & commercial spheres
 - can also be used to create a covert channel
- link encryption obscures header details
 - but overall traffic volumes in networks and at end-points is still visible
- traffic padding can further obscure flows
 - but at cost of continuous traffic

Key Distribution

- symmetric schemes require both parties to share a common secret key
- issue is how to securely distribute this key
- often secure system failure due to a break in the key distribution scheme



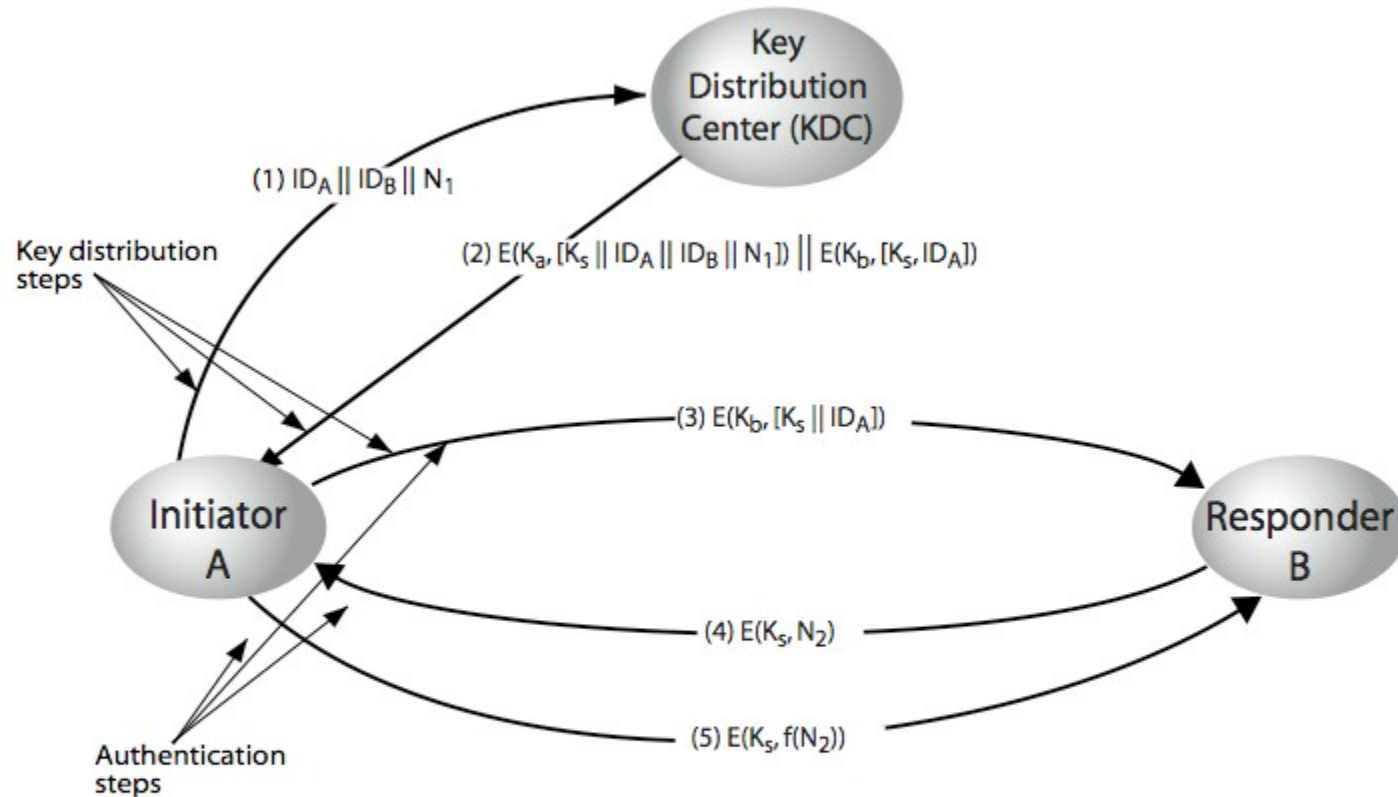
Key Distribution

- given parties A and B have various **key distribution** alternatives:
 1. A can select key and physically deliver to B
 2. third party can select & deliver key to A & B
 3. if A & B have communicated previously can use previous key to encrypt a new key
 4. if A & B have secure communications with a third party C, C can relay key between A & B

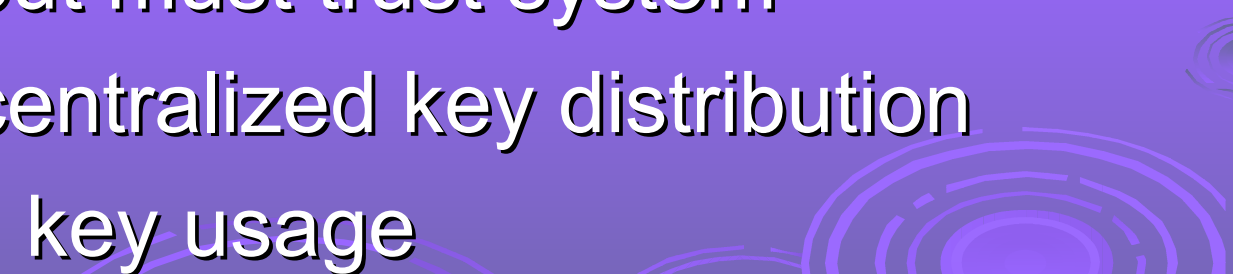
Key Hierarchy

- typically have a hierarchy of keys
- session key
 - temporary key
 - used for encryption of data between users
 - for one logical session then discarded
- master key
 - used to encrypt session keys
 - shared by user & key distribution center

Key Distribution Scenario

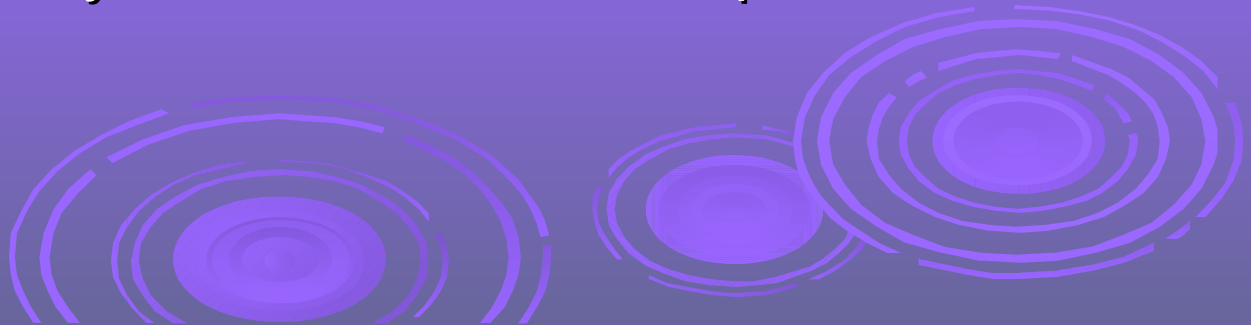


Key Distribution Issues

- hierarchies of KDC's required for large networks, but must trust each other
 - session key lifetimes should be limited for greater security
 - use of automatic key distribution on behalf of users, but must trust system
 - use of decentralized key distribution
 - controlling key usage
- 
- A decorative graphic in the bottom right corner of the slide, consisting of several concentric circles of varying shades of purple and blue, resembling ripples in water or a stylized sunburst.

Random Numbers

- many uses of **random numbers** in cryptography
 - nonces in authentication protocols to prevent replay
 - session keys
 - public key generation
 - keystream for a one-time pad
- in all cases its critical that these values be
 - statistically random, uniform distribution, independent
 - unpredictability of future values from previous values



Pseudorandom Number Generators (PRNGs)

- often use deterministic algorithmic techniques to create “random numbers”
 - although are not truly random
 - can pass many tests of “randomness”
- known as “pseudorandom numbers”
- created by “Pseudorandom Number Generators (PRNGs)”

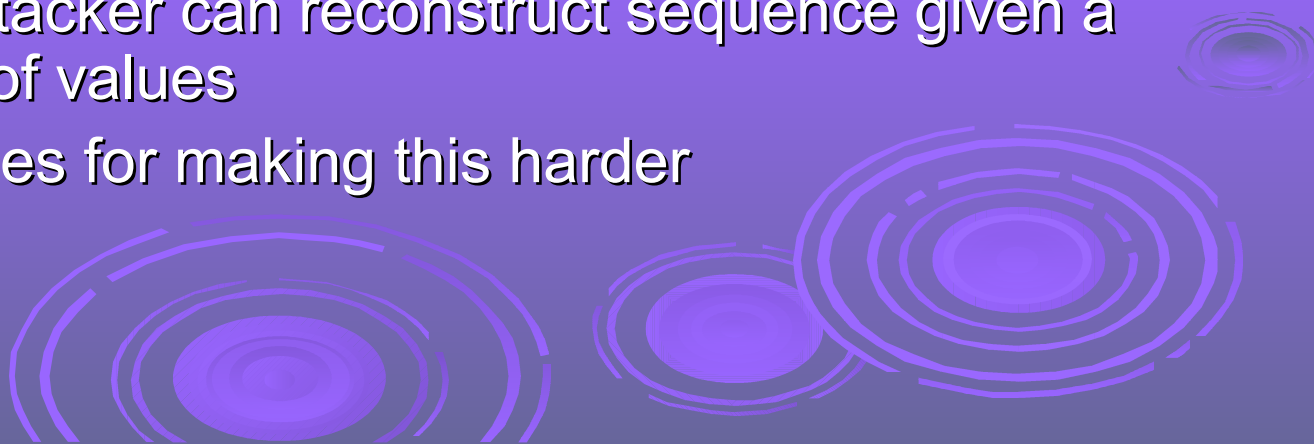


Linear Congruential Generator

- common iterative technique using:

$$X_{n+1} = (aX_n + c) \bmod m$$

- given suitable values of parameters can produce a long random-like sequence
- suitable criteria to have are:
 - function generates a full-period
 - generated sequence should appear random
 - efficient implementation with 32-bit arithmetic
- note that an attacker can reconstruct sequence given a small number of values
- have possibilities for making this harder



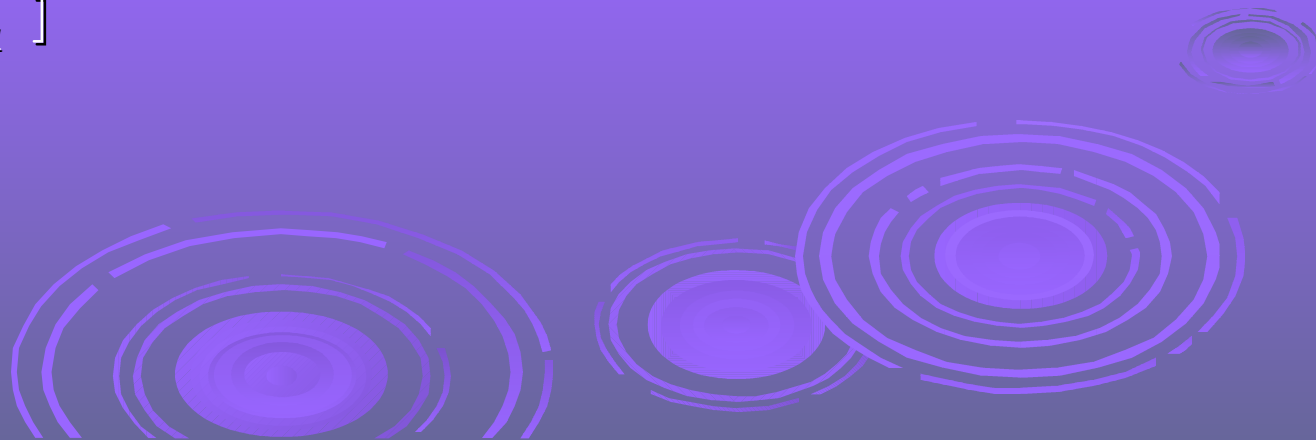
Using Block Ciphers as PRNGs

- for cryptographic applications, can use a block cipher to generate random numbers
- often for creating session keys from master key
- Counter Mode

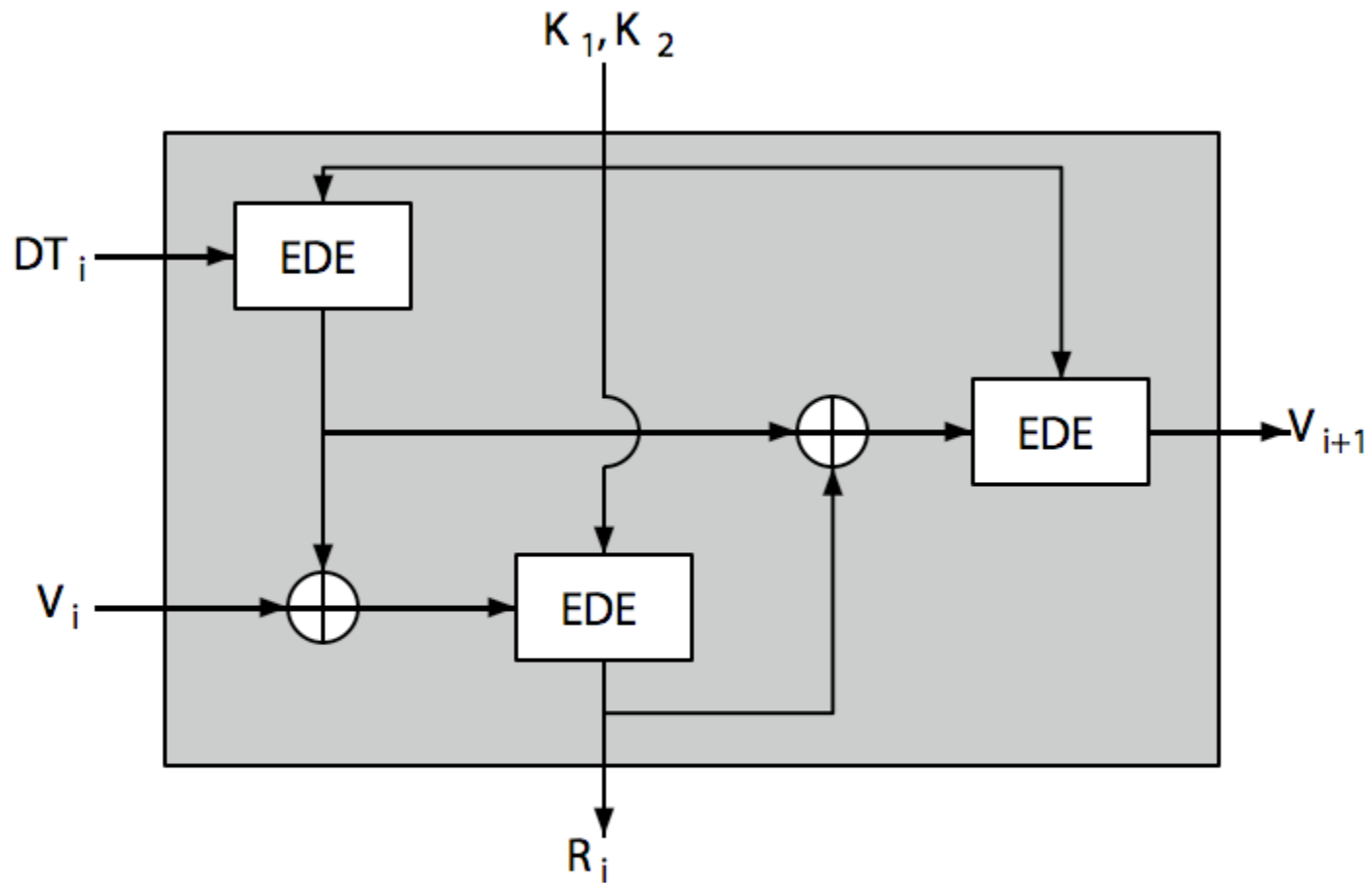
$$X_i = E_{K_m}[i]$$

- Output Feedback Mode

$$X_i = E_{K_m}[X_{i-1}]$$



ANSI X9.17 PRG



Blum Blum Shub Generator

- based on public key algorithms
- use least significant bit from iterative equation:
 - $x_i = x_{i-1}^2 \bmod n$
 - where $n = p \cdot q$, and primes $p, q \equiv 3 \bmod 4$
- unpredictable, passes **next-bit** test
- security rests on difficulty of factoring N
- is unpredictable given any run of bits
- slow, since very large numbers must be used
- too slow for cipher use, good for key generation

Natural Random Noise

- best source is natural randomness in real world
- find a regular but random event and monitor
- do generally need special h/w to do this
 - eg. radiation counters, radio noise, audio noise, thermal noise in diodes, leaky capacitors, mercury discharge tubes etc
- starting to see such h/w in new CPU's
- problems of **bias** or uneven distribution in signal
 - have to compensate for this when sample and use
 - best to only use a few noisiest bits from each sample

Published Sources

- a few published collections of random numbers
- Rand Co, in 1955, published 1 million numbers
 - generated using an electronic roulette wheel
 - has been used in some cipher designs of Khafre
- earlier Tippett in 1927 published a collection
- issues are that:
 - these are limited
 - too well-known for most uses



Summary

➤ have considered:

- use and placement of symmetric encryption to protect confidentiality
- need for good key distribution
- use of trusted third party KDC's
- random number generation issues

