Name: Ridwan Hoq
EID: rmh2376
CS Login: ridwan
Email: ridwanhoq@gmail.com

Lecture 34

1. This is stated by the Fundamental Theorum of the Noiseless Channel. Since a language cannot achieve a perfect encoding, you cannot transmit at a rate greater than C/h.
2. Increasing the redundancy would allow the receiver to have a greater chance to reconstruct the message.

Lecture 35

1. h = - (log 1/10)
2. Sometimes letters follow other letters frequently, some letters are just more frequently used, etc
3. Zero order uses 1 letter symbols, first order uses 2 letter symbols, third order uses 3 letter symbols

Lecture 36

1. Prior probabilities are hard to compute because there are all sorts of factors that can't really be accounted for reasonably.
2. It is relative to the observer because if the observer already knows the information then no information is conveyed.
3. The less redundant the message, the closer the encoding is to entropy.

Lecture 37

1. There's only numbers and punctuation used by the code. Perhaps numbers represent one type of information and punctuation is another. It would also be useful to find what patterns of symbols are repeated so you could break down what a sequence of symbols might mean.
2. A key is optional because not every encryption algorithm requires a key.
3. Encrypting a file hides the information content from a layman; you must know how to decrypt the file to extract the information content.
4. If there redundancies in the source text, then those redundancies will also exist in the encrypted text which may be used to decrypt the text.

Lecture 38

1. P
2. D(E(C, KE), KD)
3. Finding patterns might allow for the deciphering of the message because as more patterns are observed, it could give clues as to what information is being conveyed.
4. Knowing the properties of a language could be useful since the properties of a language might be preserved in the encryption of the language which could provide clues about decryption.

Lecture 39

1. While it may be possible to break, it might not be reasonable to break since it would take hundreds or thousands of years which wouldn't be useful.

2. Because usually in a linear search, you'll find what you're looking for in the first half so it will take 2^(n-1) operations rather than 2^(n).
3. They are important in ciphers so that information gets encrypted and switched up so that information isn't transmitted in plain text.
4. Confusion is transforming information in plaintext so that an interceptor cannot readily extract it. Diffusion is spreading the information from a region of plaintext widely of the ciphertext.
5. They are both necessary to ensure information is not easily accessible.

Lecture 40

1. Monoalphabetic substitution substitutes symbols uniformly. Polyalphabetic substitution makes different substitutions depending on where in the plaintext the symbol occurs.
2. The key is the 1-1 mapping of the alphabet into itself or the another alphabet. This is usually a table.
3. There are k! mappings at most for a substitution algorithm because you have to try every combination of 1-1 mappings.
4. How many positions you shift
5. 26
6. Not really
7. Create a reverse look up table based on the plaintext and key

Lecture 41

1. Because xyy is a 3 letter string and each letter can have 26 different possibilities therefore its 26 * 26 * 26.
2. With simple substitution all you need to know is the shift distance for one letter to decrypt an entire string. So if you match up one letter with every other letter, you'll get 26 * 25.
3. No because knowing the encryption algorithm would mean that you would be able to deduce something about the ciphertext.

Lecture 42

1. Since the entire key is non repeating and completely random, only the knowledge of the key can yield a decryption. Otherwise, there is no way to reduce the search space even with the knowledge of the algorithm and the ciphertext.
2. It must be random to ensure that it cannot be replicated.
3. How does one securely transmit keys.

Lecture 43

1. Transposition only accomplishes diffusion not information confusion.

Lecture 44

1. Symmetric algorithm
2. Key distribution is solving the problem of giving keys to those who need them where as key management is solving the problem of preserving the safety of keys and making them available as needed
3. No because Ks is used to encrypt the message. Only the holder of Ks-1 can decrypt the message.
4. I think it depends on the situation

Lecture 45

1. Block encryption has high diffusion and is difficult to tamper with.
2. Changing the ciphertext makes meaningful changes in the plaintext.
3. Specific types of operations when carried out on ciphertext has the same result on the plaintext

Lecture 46

1. subBytes, mixColumns
2. shiftRows
3. Inverting the mixColumns step requires a large amount of matrix multiplication
4. Block ciphers take input in fixed size amounts. Rounds are the operations performed repeatedly on those blocks.
5. Increasing the number of rounds further confuses and diffuses the plaintext,

Lecture 47

1. When encrypting blocks, identical blocks in the plaintext are identical blocks in the ciphertext.
2. Uses Cipher Block Chaining which is XORing each successive plaintext block with the previous ciphertext block and then encrypt
3. CBC can be observed over time which could let an attack spot the first block changed. Also if an attacker finds two identical ciphertext blocks, he can derive information about two plaintext blocks.
4. Key stream is different from block encryption in that the ciphertext is used as a random number generator to output a stream of bits that is used as a one-time pad.

Lecture 48

1. A private key must be kept private so that only one actor can decrypt messages.
2. One way functions are critical because the encryption is must be difficult to invert without additional information. In other words, encrypting information with a publicly available key must only be decryptable with a singular key that is kept private from the rest of the world.
3. Public key systems solve the problem of key distribution in the sense that both parties don’t need to share a key that’s private. Only one party (who’s doing the decrypting) needs a private key.
4. {P}K-1
5. Symmetric algorithms are much more efficient than asymmetric algorithms

Lecture 49

1. Yes it would still work since {{P}d}e = P = {{P}e}d
2. Prime numbers are used to ensure that keys are unencryptable since prime number division is difficult without knowing both prime factors.
3. Yes
4. Because it can only be decrypted with A’s private key
5. Because it is signed with A’s public key, which anyone has access to
6. Because it is signed with B’s private key, which only B has access to
7. By using B’s public key
8. By ensuring that it was sent from the correct place.

Lecture 50

1. So that it can be computed quickly and efficiently

2. Weak collision resistance ensures that the messages are different
3. Preimage resistance is when you can't recover m from h. Second preimage is where you can't find 2 hashes that are the same with different messages.
4. 14.1421 arguments
5. 15.811 arguments
6. Because unhashing a file is possible
7. They are collision free
8. Encrypt the message then send it. Hash the encrypted message. Send the message and the hash. Hash the encrypted message to see if the hash matches. Decrypt the message.

Lecture 51

1. Yes since S can first encrypt the message then send it.
2. No because S doesn't have R's private key
3. No
4. Key exchange requires both confidentiality and authentication

Lecture 52

1. Nothing since the algorithm would still take forever due to the discrete logarithm problem
2. An eavesdropper can send a fake msg on behalf of A.
3. An eavesdropper can send a fake msg on behalf of B.