Eric Tang
et5748

Assignment 3

Lecture 34

1. Because C/h would be perfect without noise - no thing will top it.

2. Even if some bits are corrupted or fail to send , some will get through.
The redudancy ensures that the pieces that get through will add up to a whole.

Lecture 35

1. -(log(1/10)) = 1

2. Inequality in individual symbol probability
Some symbols more likely to follow others.

3. Zero: all symbols random and independent;
First: symbols biased, but independent;
Second: symboles biased and dependent on previo us symbol;
Third: symboles biased and depend on previous 2 symbols

Lecture 36

1. Entropy can be dependent relative to the observer.

2. Messages mean as much as the uncertainty of the receiver.

3. The more redudancy, the further from entropy (t he ideal).

Lecture 37

1. Message most likely treasure due to nature of the environment
    Most likely English given the puns used.
    Most likely substitution.

2. Sometimes the keys lie in the plaintext
    (simply adding a certain value to each symbol or
    simply rearranging symbols).

3. The information should not change (integrity) - it just shouldn't
    be readily visible.

4. The redudancy may leak info on language or patterns which
    leak info on message.

Lecture 38

1. P

2. (E(P,KE),KE)

3. Patterns can gives hints to message (language, words, etc.)

4. Probability of certain symbols: redundancy.
    Number of different symbols: which language.

Lecture 39

1. The time it would take to break it is longer than most life times.

2. Only so many combinations - given enough time,

it can be figured out.

3. Helps confuse and diffuse

4. Confusion: different identities.
   Diffuse: spreads the info accross entire document.

5. Both are important.

Lecture 40

1. Mono: uniform substitution
   Poly: substitution based on position

2. A simple mapping of symbols.

3. Only so many symbols as are in the plaintext

4. 2 alphabet offset

5. 26!

6. No

7. The reverse of the encryption

Lecture 41

1. 26^3 (possible 26 symbols per spot)

2. There is redundancy

3. Yes - one-time pad. No information given the algorthm and ciphertext.

Lecture 42

1. No information leaked given algorthm and cipher

text.

2. Otherwise given enough ciphertexts, the key
can be figured.

3. If channel is secure, why not use channel.
If not, how to securely distribute key.

Lecture 43

1. Offers no confusion.

Lecture 44

1. Symmetric

2. Distribution: how to securely share key
Management: how to keep track of keys (and text
pairs).

3. If symmetric, yes (same key to encrypt and decr
ypt).
If asymmetric, no (different keys)

4. Both have they're advantages.
Symmetric have more efficient algorthms.
Asymmetric have better distribution.

Lecture 45

1. To accommodate for memory sizes (efficiency and
diffusion)

2. Changing the ciphertext with noticible changes
to plaintext
leaks information of the key.

3. Increases the difficulty to cracking key with c
iphertext.

Lecture 46

1. subBytes: for each byte in the array, use its value as an index into a 256-element lookup table, and replace byte by the value stored at that location in the table.

2. shiftRows: Let Ri denote the i th row in state. Shift R0 in the state left 0 bytes (i.e., no change); shift R1 left 1 byte; shift R2 left 2 bytes; shift R3 left 3 bytes.

3. The instructions fun faster on cpu for encrption.

4. Blocks used to diffuse, rounds used to recursively confuse.

5. For larger bit keys

Lecture 47

1. Ciphertext not diffused.

2. Cipher Block Chaining (XOR successive blocks)

3. Observed changes (first change in text) + content leaks (identicle blocks)

4. Ciphertext vs PRNG

Lecture 48

1. Decryption key

2. Privacy

3. The encryption key is public and doesn't have to be distributed.
The decrypt key is local to the user.

4. {{P}Kâˆ’1

5. Better efficiency on symmetric due to faster instruction
execution on CPUs (mult, shifts compared to mods, etc).

Lecture 49

1. Yes, RSA works both ways.

2. Consistency

3. Yes

4. An interceptor would have to factor M to recover the plaintext.
The legitimate receiver knows d and merely computes
(M)ka mod n = P, which is much easier.

5. Privacy without authentication

6. Only be has the key.

7. Can't

8. Encrypt it's sender's private key along with receiver's
public key.

Lecture 50

1. Reuse

2. Weak = possible but not likely
Strong = almost impossisble (just to find one collision)

3. A function f is preimage resistant if, given h, it is hard to find any
m such that h = f (m).

A function f is second preimage resistant if, given an input m1, it
is hard to find m2 6= m1 such that f (m1) = f (m2). This is
sometimes called weak collision resistance.

4. Limited values to hash to, arbitrary numbers of unhashed values -
it's going to happen

5. 1.25 sqrt(2^128)
1.25 sqrt(2^160)

6. It's one way cannot recover

7. One-way

8. Hash key

Lecture 51

1. No needs to be {{K}KS}KRâˆ’1

2. No, the receiver can't decrypt

3. No, can't decrypt

4. Confidentiality and authentication

Lecture 52

1. Still can't decrypt - doesn't know a and b
2. Can decrypt from a's side.
3. Can decrypt from a's side.