CMSC 426
Principles of Computer Security

Cryptanalysis

Last Class We Covered

- Man in the Middle Attacks

- MAC
- Hashing
- HMAC

- Public Key Infrastructure
- Certificates
- Digital signatures

Any Questions from Last Time?

Today's Topics

- Cryptanalytic attacks
  - Attack methods
  - Attack types
- Attack types
  - Ciphertext only
  - Known plaintext
  - Chosen plaintext
- Pseudorandom numbers

Cryptanalytic Attacks

Cryptanalysis

- ***Cryptanalysis*** is the process a cryptanalyst uses in order to discover the plaintext and/or a secret key
  - (Cryptanalyst may also just be an attacker)

- Strategy of attack type used depends on
  - Cryptanalyst’s knowledge and access
  - Nature of the encryption scheme

Cryptanalysis Attack Methods

- Brute force
- Abusing and using primes
- Analyzing ciphertext (and sometimes plaintext)
- Use ~*~math~*~ to exploit weaknesses of algorithms

- “Non-traditional” attacks
  - Timing attacks
  - Glitch attacks
  - Social engineering/non-technical attacks

Brute Force Attack

- Attack attempts to “brute force” its way through the problem space to a solution

- Systematically check each and every possible key, encryption method, etc. until the correct one is found
  - Requires some way to automatically check results

- Simplest defense…
  - Make the problem space large – too large to thoroughly test
  - One reason why key size keeps increasing

Abusing Primes

- Many crypto algorithms use prime numbers as a key component
- Diffie-Hellman uses a publicly transmitted prime p and a publicly transmitted primitive root g
  - Alice and Bob each secretly choose a number, and using ~*~math~*~ transmit more public numbers, then combine that info to form the key
- What if a lot of Diffie-Hellman implementations used the same p?
  - This would allow an attacker to pre-compute discrete logs for that p
  - (This would give an attacker a very small set to brute force from)

Information taken from https://weakdh.org/

Using (Factoring) Primes

- Many cryptographic algorithms rely on the product of two prime numbers as a key security component

- RSA uses the product of two secret prime numbers, *p* and *q*
  - Public and private exponents, *e* and *d*, are chosen within certain constraints relating to each other and these primes

- If *p* and *q* can be factored out of *n*, and *e* is already public…
  - The problem space of possible *d* values shrinks significantly
  - With quantum computing and Shor's algorithm, factoring is polynomial

Math Example #1: DES Round Analysis

- An m-round characteristic of a Feistel-type cryptosystem is a sequence $(\Omega_{in}, \delta_1, \Delta_1, \cdots, \delta_m, \Delta_m, \Omega_{out}) = (\Omega_{in}, \Omega_{\Delta}, \Omega_{out})$
  - Where Ω_{in} and Ω_{out} are input and output differences. The pairs (δ_i, Δ_i); $i = 1, \cdots, m$, are consecutive input and output difference for the round f_k.
- For example, if the input difference $\Omega_{in} = (\delta_A, 60\ 00\ 00\ 00_x)$
  - The pair of difference (C_x, E_x) happens with probability 14/64
  - And then we get the output $\Omega_{out} = (\delta_A \oplus 00\ 80\ 82\ 00_X, 60\ 00\ 00\ 00_X)$
- Etc…

Information taken from Dr. Jennifer Seberry’s slides: https://www.uow.edu.au/~/jennie/CSCI971/Cs47104.ppt

Math Example #2: AES Differentials

- AES: each non-zero byte in delta input to a round contributes 2^{-6} or 2^{-7} to probability of output difference.
  - If difference input to a round is 0 except in one byte, probability specific difference occurs in output of the round is $\leq 2^{-6}$
  - If difference input to a round is 0 except in two bytes, probability specific difference occurs in output of the round is $\leq 2^{-12}$

- Entirely due to the S-Box – other steps in round do not impact differential probability

Information taken from Professor Debbie Cook's slides: https://www.csd.uoc.gr/~hy590-82/lect5+6-cryptanalysis.ppt

Non-Traditional: Timing Attacks

- Side channel attack, where time taken to execute cryptographic algorithms is analyzed
  - Every logical operation takes time to execute, and time taken will often differ based on the input provided
  - Some versions of this attack may also measure power consumption
- For example, modular exponentiation (*e.g.*, $a = g^A \% p$) has a run time that depends linearly on the number of ‘1’ bits
- Effectiveness depends on knowledge of the hardware implementation and the crypto system in use

Non-Traditional: Glitch Attacks

- Side channel attack, which requires physical access to the hardware, and is often performed on things like smart cards
- Essentially, by introducing specific glitches, the CPU can be made to execute completely incorrect instructions
  - Glitch example: replacing a 5 MHz clock with a 20 MHz one
  - Result example: dump contents of memory to output
- Can even be used to reverse engineer unknown block ciphers

Information taken from https://www.cl.cam.ac.uk/~mgk25/tamper2.pdf

Cryptanalysis Attack Types

Cryptanalysis Attack Types

Type of Attack	Known to Attacker/Cryptanalyst (assume the algorithm is always known)
Ciphertext only	Ciphertext they want decoded
Known plaintext	Ciphertext they want decoded One or more plaintext-ciphertext pairs
Chosen plaintext	Ciphertext they want decoded At least one plaintext-ciphertext pair, where plaintext was <u>chosen</u>
Chosen ciphertext	Ciphertext they want decoded At least one plaintext-ciphertext pair, where ciphertext was <u>chosen</u>
Chosen text	Ciphertext they want decoded At least one plaintext-ciphertext pair, where plaintext was <u>chosen</u> At least one plaintext-ciphertext pair, where ciphertext was <u>chosen</u>

Information taken from Computer Security (Stallings & Brown)

Ciphertext Only Attack

- Most difficult attack/analysis to pull off
  - Analyst may not even know the encryption algorithm used
- Assume that the analyst still has <u>some</u> knowledge of plaintext
  - What language or format it exists in
  - Some plaintext messages may even be in a standard format
- Every modern cryptographic algorithm has been vetted to not be susceptible to this attack
  - But coding up your own version of the algorithm hasn't been vetted!

Known Plaintext Attack

- Analyst has access to at least one plaintext-ciphertext pair

- Idea is that analyst uses information about the plaintext to begin to make sense of the ciphertext
  - Patterns and repeated words or phrases in the plaintext may have matching output in the ciphertext
  - If that output is spotted in new ciphertext, the plaintext can be assumed to be known, at least for that piece

- Integral to breaking the Engima machine during WWII

Chosen Plaintext Attack

- This attack requires that the analyst has some way of requesting or obtaining the ciphertext for some given plaintexts
  - May be achieved with social engineering if not directly
- A variation on this is CPA2 (Adaptive Chosen-Plaintext Attack), where the analyst can request ciphertexts in multiple batches
  - Normally, only one batch of plaintexts is “allowed” to be encrypted
- Based on the information gleaned, analyst’s goal is to extract the key used for the encryption

Chosen Ciphertext Attack

- Like chosen plaintext attack, may be adaptive (multiple “batches”) or non-adaptive (single “batch”)
  - Adaptive is called CCA2 (adaptive chosen-ciphertext attack)

Chosen Text Attack

- Combination of Chosen Plaintext Attack and Chosen Ciphertext Attack

- Neither are used very commonly

Pseudorandom Numbers

- See additional slides for RNGs

Announcements

- Lab 3 is scheduled to come out tomorrow
  - (Earlier than it says on the website schedule)
  - Topic will be cryptanalysis, and no VM will be involved

- HW1 is being graded, HW2 and Lab 2 will be graded following
- Also working on getting Paper 1 (and Paper 2&3) graded
  - Paper 2&3 is due tomorrow night at midnight