Private Key Encryption

Information-Theoretic Security: Eve doesn't have enough information to crack the message, even with unlimited computation power

Computation Security: Eve is computationally limited (e.g. can't factor quickly)

Kerkhoff's Principle: We assume that

- Eve knows the encryption scheme
- Eve knows the alphabet and language
- Eve doesn't know the key
- Key is chosen randomly

Cracking General Sub Cipher

Let σ be a permutation. We look at the frequncy vectors of n-grams (length 26^n).

Then over some redos and lots of iterations (swapping $j, k \in \{0, \ldots, 25\}$), we find the best candidate for σ_r

Cracking Vigenere Cipher

First we find the length of the key L. We assume that a word that appears frequently will likely appear in the same position $i \pmod L$

- For example is "aiq" appears in the slots (57, 58, 59), (87, 88, 89), (102, 103, 104), (162, 163, 164), we can deduce that the length of the key is a divisor of the gaps between these sequences, $L = \{1, 3, 5, 15\}$

This will create a stream of every Lth character. We can do shift analysis on these streams

Linear Cong Gen

Use a recurrence $x_{i+1} = AX_i + B \pmod M$ to find random-looking bits. Need $\gcd(A, M) = 1$

For this example we have $x_0 = 2134, A = 4381, B = 7364, M = 8397$

$$x_{n+1} = 4381x_n + 7364 \pmod{8397}$$

We decode $x_0 = 2134$ into $(21, 34)$ and view letters as 2 digit numbers $\pmod{20}$ and do column addition $\pmod{10}$

The first few values of x_n are

- $x_0 = 2134$
- $x_1 = 2160$
- $x_2 = 6905$
- $x_3 = 3778$

Text-Letter	S	E	C	R	E	T
Text-Digits	19	05	03	18	05	20
Key-Digits	21	60	69	05	37	78
Ciphertext	30	65	62	13	32	98

To decode

Bob Wants	$m_{1,1}m_{1,2}$	$m_{2,1}m_{2,2}$	$m_{3,1}m_{3,2}$
Bob Knows Key	21	60	62
Bob Sees	30	65	62

Thus Bob can deduce $m_{i,j}$

- $m_{1,1} + 2 \equiv 3 \implies m_{1,1} \equiv 3 - 2 \equiv 1$
- $m_{1,2} + 1 \equiv 0 \implies m_{1,2} \equiv -1 \equiv 9$
- Thus the first letter is $19 = S$

Cracking LCG

Assume that Eve knows that A, B, M are all 4 digits and that the document contains the word "Pakistan". So Eve looks at each 8 sequence of letters and tests it. Suppose Eve tests the sequence $(24, 66, 87, 47, 17, 45, 26, 96)$

Text-Letter	P	A	K	I	S	T	A	N
Text-Digits	16	01	11	09	19	20	01	14
Key-Digits	$k_{11}k_{12}$	$k_{21}k_{22}$	$k_{31}k_{32}$	$k_{41}k_{42}$	$k_{51}k_{52}$	$k_{61}k_{62}$	$k_{71}k_{72}$	$k_{81}k_{82}$
Ciphertext	24	66	87	47	17	45	26	96

Eve guesses that the key digits are $(18, 65, 76, 48, 08, 25, 25, 82)$ and is able to create the formulas

$$7648 = 1865A + B \pmod M$$
$$825 \equiv 7648A + B \pmod M$$
$$2582 \equiv 825A + B \pmod M$$

Using some arithmetic, we can find the values of A, B, M

- **Note** $7649 \le M \le 9999$ since M is 4 digits long and $\gcd(A, M) = 1$

After finding A, B, M, Eve can recursively solve for x_0

Finally, after finding x_0, A, B, M still needs to recover the entire plaintext and test IS-ENGLISH. If it fails, then Eve needs to test the next sequence

Matrix Cipher

Brute force takes $O(26^{n^2})$ and row-by-row takes $O(n26^n)$

Let $T = t_1 t_2 \dots t_N$ where $t_i = t_i^1 \dots t_i^8$

Note that $Mt_i = m_i \implies R_j t_i = m_i^j$

```
for i = 1 to 8
  for r in Z^{8}_{26}
    T' = (r * t_1, ... r *t_N)
    if IS-ENGLISH(T')
      r_i = r
      goto next i
```

For an $n \times n$ matrix, each PT-CT pair gives n equations, resulting in n^2 variables and n^2 equations. Thus we need n pairs

Randomized Shift Cipher

Determinstic ciphers map message to the same ciphertext

Randomized shift sends $((r_1; m_1 + f(r_1)), \dots)$ and decodes $(c_1 - f(r_1), \dots)$

Math for Public Key Encryption

Exponentiation

Given a, n, p, calculate $a^n \pmod p$ by converting the exponent into binary and using repeated squaring. Then we have

$$a^n = a^{n_1} * a^{n_2} * \cdots \text{ where } n_1, n_2, \ldots \text{are powers of 2}$$

Example: $17^{265} \pmod{101}$

$265 = 2^8 + 2^3 + 2^0 \implies 17^{265} = 17^{2^8} * 17^{2^3} * 17^{2^0} \equiv 84 * 36 * 17 \equiv 100 \pmod{101}$

Discrete Log

Given g, y, p output x such that $g^x \equiv a \pmod{p}$. Represented as $DL_{p,g}(y) = x$

This problem is suspected to be hard for $g \in \{p/3, \ldots, 2p/3\}$. Although there are some tricks

- If g is a generator of Z_p^* then $g^{(p-1)/2} \equiv p \equiv -1$
- **Example**: $3^x \equiv 92 \pmod{101} \implies 92 \equiv 101 - 9 \equiv (-1)3^2 \equiv 3^{50} * 3^2 \equiv 3^{52}$

Generator for Z_p^*

Theorem: if g is NOT a generator, then exists x such that

- $x \mid p - 1$
- $x \neq p - 1$
- $g^x \equiv 1 \pmod{p}$

We also want **safe primes** such that $p - 1 = 2q$ is prime

Let F be the set of factors, except $p-1$, of $p-1$. Then $F = \{2, q\}$

Thus we loop through $g \in \{p/3, \ldots, 2p/3\}$ and compute g^x for each $x \in F$. If any $= 1$ then g is NOT a generator

Primality Testing

Fermat's Little Theorem: $a^p \equiv a \pmod{p}$

Thus we can take a random subset of $R = \{2, \ldots, p-1\}$ and for each $a \in R$\$, if $a^p \not\equiv a$ then p is NOT a prime

Generating Primes

Return an L-bit prime

Idea is to pick a random $y \in \{0,1\}^{L-1}$ and let $x = 1y$, then test if x is a safe prime

Diffie-Hellman

Given a security param L

1. Alice finds (p, g) such that $\text{len}(p) = L$
2. Alice sends (p, g) to Bob (Eve sees this)
3. Alice picks random a and sends $g^a \pmod{p}$ to Bob (Eve sees this)
4. Bob picks random b and sends $g^b \pmod{p}$ to Alice (Eve see this)
5. Alice computes $(g^b)^a = g^{ab}$
6. Bob computes $(g^a)^b = g^{ab}$

g^{ab} is the **shared secret** and it believed that it is hard for Eve to find g^{ab}

El Gamal

1. Alice and Bob do Diffie Hellman
2. Alice and Bob share $s = g^{ab} \pmod{p}$
3. Alice and Bob compute $s^{-1} \pmod{p}$
4. $\text{Enc}(m) = c = ms \pmod{p}$
5. $\text{Dec}(c) = cs^1 = mss^{-1} = m \pmod{p}$

RSA

Fermat-Euler Theorem: $a^m \equiv a^{m \pmod{\phi(n)}} \pmod{n}$ for a rel prime to n

Example: $14^{999,999} \pmod{393}$

$\phi(393) = \phi(3 * 131) = 2 * 130 = 260$

Then $14^{999,999} = 14^{199,999 \pmod{260}} \pmod{393} \equiv 14^{39} \pmod{393}$

Algorithm:

1. Alice picks 2 primes p, q of length L and computes $N = pq$
2. Alice computes $R = \phi(N) = \phi(pq) = (p-1)(q-1)$
3. Alice picks $e \in \{R/3, \ldots, 2R/3\}$ that is relatively prime to R
4. Alice finds d such that $ed \equiv 1 \pmod{R}$
5. Alice broadcasts (N, e) so that both Bob and Eve can see it
6. Bob wants to send $m \in \{1, \ldots, N-1\}$ and broadcasts $m^e \pmod{N}$
7. Alice receives $m^e \pmod{N}$ and computes

$$(m^e)^d \equiv m^{ed} \equiv m^{ed \pmod{R}} \equiv m \pmod{N}$$

RSA issues

NY, NY problem solved by having Bob concatenate a random r and sending $(rm)^e$

- Alice knows that r takes up the first L_1 bits and m takes up the last L_2 bits
- RSA is **malleable**, so if Eve sees a message, she can figure out a way to send a similar one

Pollard-Rho

Idea is to find a factor p of N. We find x, y such that $x \equiv y \pmod{p} \implies \gcd(x-y, N)$ is a nontrivial factor since p divides both

Let $x_{i+1} = f(x) = x_i^2 + c$. Then for each x_i we check if $\gcd(x_i - x_j, N) \neq 1$ for $j < i$

Pollard $p-1$

Idea is that $p \mid n \implies \gcd(2^{p-1} - 1 \pmod{n}, n) \neq 1$ (Fermat's Little Theorem)

- Since p is unknown, we take $2^{k(p-1)} - 1 \pmod{n}$ for any k
- Idea is that we raise 2 to a power and hope it has $p-1$ as a divisor

TODO GO OVER THIS