Cryptography

(Handout November 19, 2008)

Symmetric Key Cryptography

One of the chief aims of cryptography is to provide means by which one party may communicate privately over an open channel with a second party. All messages which we intend to transmit securely is assumed to consist of bitstrings. (This condition is naturally met for text messages, which are already implemented as binary files by computers and other digital devices. Other messages, such as analogue audio/video files, are typically first discretized as streams of bits before encrypting.)

Assume Bob (B) has a message bitstring m of length n which he desires to send to Alice (A) over the internet, a cellular telephone connection, or other open channel which is accessible to any observer who has sufficient interest in eavesdropping. Bob wants to ensure that no one who intercepts the message can understand it. A very simple example of a scheme for achieving this goal is the *Vernam cipher,* which we now explain. Assume Alice and Bob have previously met and have agreed on a shared secret bitstring e, also of length n, known only to the two of them. In order to encrypt the 'plaintext' message string m, Bob will first perform bitwise addition mod 2 to obtain the 'ciphertext' $m' = m + e$. (This operation consists of simply adding m and e as vectors in $\mathbb{F}_2^n$). He then transmits the ciphertext m' to Alice over the open channel. When Alice receives the ciphertext m', she recovers the plaintext message by performing another bitwise addition $m' + e$, which results in the original plaintext message m; this is because

$$m' + e = (m + e) + e = m + 2e = m$$

since the scalar $2 = 0$.

For example, suppose Alice and Bob agree beforehand on the secret 10-bit key encryption key

$$e = 11011\,01000.$$

At a later time Bob wants to send the message

$$m = 01110\,11100$$

to Alice. He encrypts the message as

$$m' = m + e = 10101\,10100$$

which he sends to Alice, who recovers the original message as

$$m' + e = 01110\,11100.$$

Assuming Alice and Bob are careful to keep their encryption key a secret, this scheme is perfectly secure. If e is chosen completely at random (from among the $2^{10} = 1024$ bitstrings of length 10), then the ciphertext m' appears completely random to the eavesdropper, who has no more chance of deciphering the original message m than he/she does without knowing m', i.e. intercepting m' offers no advantage in determining the contents of the message (over simply guessing). This is an example of a *symmetric key encryption scheme,* because the key that Bob uses to encrypt messages is the same key that Alice uses to decrypt messages.

One apparent limitations of the Vernam cipher scheme is that the secret key e cannot be any shorter than the message Bob intends to send to Alice. Another is that in order to maintain security, the secret key an only be used for one transmission; if it is re-used to encrypt future messages, then an eavesdropper will be able to detect patterns that will allow the scheme to be broken (this may not be immediately apparent but is not hard to verify.) Both of these obstacles may be overcome with some cleverly chosen schemes, provided that the encryption key is chosen to be about 16 characters (128 bits) long, and sufficiently random. For example, Microsoft Word currently allows text files to be encrypted and protected by an encryption key: select Tools → Options → Security, then enter the encryption key as a password. The file may be later accessed by anyone who possesses both the encrypted file and the password. Note that the encryption used by MS Word is another symmetric key scheme (since the same key as was used to lock the document, may be used to unlock the document). Of course if the encryption key is poorly chosen (too short or not sufficiently random) then it is not hard for the intercepted file to be opened by an unauthorized party. This may be easily accomplished by readily available software which automatically tries passwords until it finds the right one. If a purely random bitstring of length 128 is chosen as the encryption key, then anyone trying to break the encryption will, on average, need to try about half of the

$$2^{128} = 340,282,366,920,938,463,463,374,607,431,768,211,456$$

possible keys before successfully opening the document... a rather prohibitive task!

Many good symmetric key encryption schemes are known. The only real difficulty in their use, is the task of agreeing on the secret key to be used. If Bob wants to communicate privately with Alice, how (other then meeting privately beforehand) do they agree on the secret key? It cannot be sent over the open channel, since an eavesdropper who access both the ciphertext and the encryption key would then be able to decipher the message. Perhaps Alice and Bob possess some secret shared information (Bob might suggest Alice

use his social security number, if Alice happens to already know this) but this won't work if Bob and Alice are in reality two parties who have never met (such as Amazon.com and one of its online customers).

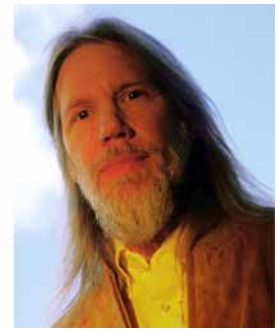
Until the 1970's, it was generally thought to be *impossible* to agree on a secret key by communicating over an unsecure channel. The credit for daring to believe otherwise, goes to Whitfield Diffie, who in the 1970's began the search for *public key cryptosystems*, which would provide the opportunity for two parties communicating over an open channel to agree on a secret key that could be used for encryption (using a symmetric key encryption scheme).

Diffie-Hellman Scheme

In 1976 Whitfield Diffie and Martin Hellman proposed the following scheme, one of the first public key cryptosystems invented. The steps in this process are as follows:

1. Alice and Bob agree on a large prime power q and on a nonzero element $\alpha \in \mathbb{F}_q$ chosen at random. In order to be secure, one must choose q to be very large (at least 50 digits long, preferably hundreds of digits long) and α should be different from 1 and some other poor choices of field element; but this is easily managed since α is chosen at random from a very large field. The choice of q and *alpha* can be communicated freely over the unsecure channel.
2. Alice randomly chooses a value $a \in \{2, 3, \ldots, q-2\}$ known only to herself. She computes $A = \alpha^a$, which she sends to Bob over the unsecure channel.
3. Bob randomly chooses a value $b \in \{2, 3, \ldots, q-2\}$ known only to himself. He computes $B = \alpha^b$, which he sends to Alice over the unsecure channel.
4. Alice computes $e = B^a$ using her secret value of a, and the value of B from Bob.
5. Bob computes $e = A^b$ using his secret value of b, and the value of A from Alice.

Martin Hellman
1945–



Whitfield Diffie
1944–

Note that Alice and Bob arrive at the same shared secret (steps 4 and 5) since Alice computes $e = B^a = (\alpha^b)^a = \alpha^{ab}$ whereas Bob computes $e = A^b = (\alpha^a)^b = \alpha^{ab}$. An

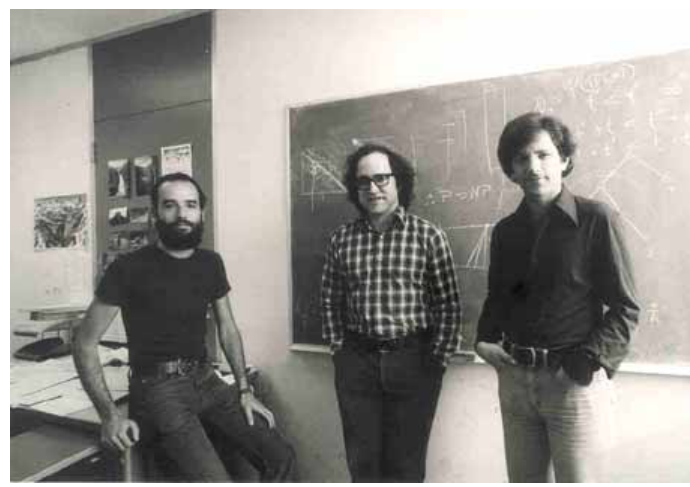
eavesdropper 'Eve' will presumably be unable to deduce this shared secret value of e since the only information she has access to are the values of q, α, $A = \alpha^a$ and $B = \alpha^b$. If Eve can solve the discrete logarithm problem, she can deduce the value of a given α and α^a; and from this and the value of B which she also intercepted, she can then deduce the value of $e = B^a$. However, the discrete logarithm problem is apparently too difficult and so we are confident that Eve is unable to deduce the value of the secret key e.

The Maple demonstration of the Diffie-Hellman scheme from class on Friday, November 14, 2008, can be viewed at

http://www.uwyo.edu/moorhouse/courses/3500/DiffieHellman.html

RSA Scheme

In 1977, an even more versatile public key encryption scheme was discovered by Rivest, Shamir and Adleman. Let's say Alice wants to enable parties everywhere to send her messages that only she can decipher. She randomly and secretly chooses two large primes p and q. (Choosing them to be 120 digits long or more should suffice. In the interests of security, other considerations apply in choosing p and q; however in the interest of brevity we'll overlook this.) She computes $n = pq$ and $\phi(n) = (p-1)(q-1)$. She also randomly chooses an integer e between 1 and $\phi(n)$ which is relatively prime to $\phi(n)$, a condition which is easily verified using the Euclidean algorithm. Actually by using the extended Euclidean algorithm in her verification of $\gcd(e, \phi(n)) = 1$, she obtains a positive integer d such that $de \equiv 1 \mod \phi(n)$. Now Alice publishes the pair of integers (n, e) (her *public key*) and keeps the other integers secret (p, q, $\phi(n)$, d).



Adi Shamir 1952– Ronald Rivest 1947– Leonard Adleman 1945–

It is now possible for anyone to send Alice a secret message which only she can decrypt. For example, suppose Bob wants to send Alice a message m. We suppose that m is an

integer between 1 and n. (If Bob's message is longer than this, he should first break it down into blocks consisting of integers less than n, and encrypt each block separately.) Now Bob looks up Alice's public key (n, e) and he encrypts he plaintext message m as the ciphertext

$$m' \equiv m^e \mod n$$

which he sends to Alice over the unsecure channel. Alice receives the ciphertext m' and decrypts it by computing

$$m \equiv (m')^d \mod n.$$

Note that both encryption and decryption may be efficiently performed using binary exponentiation mod n, but with different exponents: e to encrypt, d to decrypt. The reason Alice recovers the original message is that

$$(m')^d \equiv (m^e)^d \equiv m^{de} \equiv m \mod n.$$

Here we have used the fact that $de \equiv 1 \mod \phi(n)$. We are also assuming that $\gcd(m, n) = 1$, but this is a very safe assumption! (If $\gcd(m, n) \neq 1$, then $\gcd(m, n) = p$ or q; but then Bob could simply compute this gcd by Euclid's algorithm and he would have successfully factored n! However n is so large that factoring it is prohibitively difficult.)

Consider the challenge of trying to break this encrypted message. An eavesdropper, Eve, who intercepts the ciphertext m' also has access to Alice's public key (n, e). However the only known way to recover the plaintext m involves first finding d; and this evidently requires knowing $\phi(n)$, which in turn is not known unless both p and q are known. It seems that Eve's job of breaking the encryption, is as hard as trying to factor n, which itself is believed to be a prohibitively difficult task.

The Maple demonstration of the RSA scheme from class on Monday, November 17, 2008, can be viewed at

http://www.uwyo.edu/moorhouse/courses/3500/RSA.html

We remark that while the RSA scheme can be used to encrypt documents of any length, simply by first splitting it into blocks consisting of positive integers less than n, in practice this process becomes impractical for large documents. Instead, practical encryption algorithms use the RSA scheme only for encrypting a key (of length perhaps 128 bits or 256 bits), and then this key is used with a symmetric key encryption algorithm to encrypt the main body of the document. This approach allows for faster encryption.

We should also emphasize that for both the Diffie-Hellman scheme and the RSA scheme, the selection of random integers, generation of primes, and modular exponentiation, are details that can be readily implemented using simple software on a typical computer. The average user, who likely has not taken a course in applied algebra, will

not understand the details of the encryption; he/she only needs our assurance that the encryption works and is secure.

Authentication

We have described how Alice can publish a public key allowing everyone to send her private messages, which only she can decrypt. However, when Alice decrypts a message and finds that it ends with "From Bob", how can she be assured that the message really came from Bob? After all, if everyone has access to Alice's public key, isn't it possible that someone else could be sending this message, and pretending to be Bob? Another goal of public key cryptography is to provide some means of *authentication,* i.e. evidence that certifies the identity of the true author of a message. We describe how this is possible in the context of the RSA scheme.

Suppose all participants in our exchange generate their own public and private keys, and they publish their public keys. For example

- Alice publishes her public key (n_A, e_A) and keeps secret her private key d_A satisfying $d_A e_A \equiv 1 \mod n_A$.
- Bob publishes his public key (n_B, e_B) and keeps secret his private key d_B satisfying $d_B e_B \equiv 1 \mod n_B$.
- Charles publishes his public key (n_C, e_C) and keeps secret his private key d_C satisfying $d_C e_C \equiv 1 \mod n_C$.

etc. Now suppose Bob has a message m for Alice, ending with "From Bob". He first computes

$$m' = m^{d_B} \mod n_B$$

using his own secret key. He then looks up Alice's public key and computes

$$m'' \equiv (m')^{e_A} \mod n_A$$

and he sends this value m'' to Alice. Anyone other than Alice who intercepts m'' will be unable to decipher it. When Alice receives the value m'', she first recovers

$$m' \equiv (m'')^{d_A} \mod n_A$$

using her own private key. She then looks up Bob's public key (n_B, e_B) and computes

$$m = (m')^{e_B} \mod n_B .$$

Her assurance that this message came from Bob, is that the resulting message is intelligible, and ends with "From Bob". If the message m'' had not originated with Bob, the final value for m determined by Alice would have been complete gibberish.

HW#4 Due Mon Dec 1, 2008

1. We have seen how $\phi(n)$ can be easily computed if the factorization is known. We consider now the reverse problem of trying to factor n, if both n and $\phi(n)$ are known. Given that

$$n = 1756657589977279711771588130834069003345426037007673786850499681826023069$$

and

$$\phi(n) = 175665758997727971177158813083406898996668282459999012030538229105899664 0$$

find the prime factorization of n. Check your answer.

Hint: Assume $n = pq$ where p and q are primes. Solve for p and q. You may use MAPLE, but *do not* use the 'ifactor' command to try to factor n. Even if the 'ifactor' command were successful in this example, it would not work for 240-digit numbers such as one typically encounters in implementations of the RSA scheme.

2. Alice's public key consists of $n = 7745677884829$ and $e = 15163553103$. Bob's plaintext message is an integer m between 1 and n. His ciphertext is $m' = 1530682384576$.

   (a) Find the prime factorization of n using the 'ifactor' command in MAPLE.

   (b) Find Alice's secret value of d.

   (c) Deduce Bob's plaintext message m.