Cryptography

- **history**
  - Caesar cipher, rot13
  - substitution ciphers, etc.
  - Enigma (Turing)
- **modern secret key cryptography**
  - DES, AES
- **public key cryptography**
  - RSA, PGP, PKI
- **crypto politics**

Cryptography basics

- **Alice & Bob want to exchange messages**
  - keeping the content secret
  - though not the fact that they are communicating
- **they need some kind of secret that scrambles messages**
  - makes them unintelligible to bad guys but intelligible to good guys
- **the secret is a "key" (like a password)**
  - known only to the communicating parties
  - that is used to do the scrambling and unscrambling
  - for Caesar cipher, the "key" is the amount of the shift (A => D, etc.)
  - for substitution ciphers, the key is the permutation of the alphabet
  - for Enigma, key is wiring and position of wheels plus settings of patches
  - for modern ciphers, the key is a large integer used as part of an intricate algorithmic operation on the bits of the message

Modern secret key cryptography

- **messages encrypted and decrypted with a shared secret key**
  - usually the same key for both operations ("symmetric")
- **encryption/decryption algorithm is known to adversaries**
  - "security by obscurity" *does not work*
- **attacks**
  - decrypt specific message(s) by analysis
    - various combinations of known or chosen plaintext and ciphertext
  - determine key by "brute force" (try all possible keys)
- **if key is compromised, all past and future messages are compromised**
- **big problem: key distribution**
  - need a secure way to get the key to both/all parties
    - diplomatic pouches, secret agents, ...
  - doesn't work when the parties don't know each other
  - or have no possible channel for exchanging a secret key
  - or when want to exchange secret messages with many different parties
    - e.g., credit card numbers on Internet

DES and AES

- **Data Encryption Standard (DES)**
  - developed ~1977 by IBM, with NSA involvement
  - widely used, though lingering concerns about trap doors
  - 56-bit key is now too short:
    - can exhaustively test all keys in a few hours
    - with comparatively cheap special-purpose hardware
  - "triple DES" uses 3 DES encryptions to increase effective key length
- **Advanced Encryption Standard (AES)**
  - result of an international competition run by NIST (www.nist.gov/aes)
  - completely open: algorithms and analyses in public domain
  - Rijndael: winning algorithm selected October 2000
    - approved as official US government standard
  - 128, 192, 256-bit keys
  - fast in both hardware and software implementations

Advanced Encryption Standard (AES)

- **Rijndael**
  - Joan Daemen & Vincent Rijmen, Belgium
  - 128, 192, 256-bit keys

Public key cryptography

- **fundamentally new idea**
  - Diffie & Hellman (USA, 1976); earlier in England but kept secret
- **each person has a public key and a private key**
  - the keys are mathematically related
  - a message encrypted with one can only be decrypted with the other
- **public keys are published, visible to everyone**
- **private keys are secret, known only to owner**

- **Alice sends a secret message to Bob by**
  - encrypting it with Bob's public key
  - only Bob can decrypt it, using his private key
- **Bob sends a secret reply to Alice by**
  - encrypting it with Alice's public key
  - only Alice can decrypt it, using her private key

Digital signatures

- **can use public key cryptography for digital signatures**
  - if Alice encrypts a message with her <u>private</u> key
  - and it decodes properly with her public key
  - it had to be Alice who encoded it
- **signature can be attached to a message**
  - Alice encrypts a message with her private key
  - Alice encrypts the result with Bob's public key
  - only Bob can decrypt this (with his private key)
    but it won't make any sense yet
  - Bob then decrypts it with Alice's public key
  - if it decodes properly, it had to be Alice who encrypted it originally
- **necessary properties of digital signatures**
  - can only be done by the right person: can't be forged
  - can't re-use a signature to sign something else
  - signature attached to a document: signs specific contents
  - signature can't be repudiated

Digital signatures, continued

- **usually done by signing a "cryptographic hash" of a document, not the document itself**
- **secure hash is computed by an algorithm**
- **reduces any data to a comparatively short number such that**
  - can't deduce the original document from the number
  - any change to the original document produces a completely different hash
  - can't find another document that has the same hash
- **current secure hash algorithms**
  - MD5 (Rivest, MIT): 128 bits
  - SHA-1 (US government standard): 160 bits
- **new international competition to create a new version of SHA-1**
  - analogous to AES competition (again run by NIST)
  - first round submissions 10/31/08
  - e.g., http://www.securityfocus.com/news/11536?ref=rss

RSA public key cryptographic algorithm

- **most widely used public key system**
- **invented by Ron Rivest, Adi Shamir, Len Adleman, 1977?**
  - patent expired Sept 2000, now in public domain

- **based on (apparent) difficulty of factoring very large integers**
  - "large" >= 1024 bits ~ 300 digits
  - public key based on product of two large (secret) primes
  - encrypting and decrypting require knowledge of the factors
- **slow, so usually use RSA to exchange a secret "session key"**
  - session key used for secret key encryption with AES
  - used by SSH for secure login
  - used by browsers for secure exchange of credit card numbers
    https: http with encryption
  - SSL (Secure Sockets Layer) or TLS (Transport Layer Security)
    used to encrypt TCP/IP

Public key personae

- **Martin Hellman,**
  Whitfield Diffie

- **Adi Shamir,**
  Ron Rivest,
  Len Adleman

How does RSA work? (you are not expected to remember this)

- **choose two big primes p and q (~100 digits each)**
- **compute $N = p \times q$ (~200 digits)**
- **select e, relatively prime to $(p-1) \times (q-1)$**
- **compute d such that $e \times d = 1 \bmod (p-1) \times (q-1)$**
- **public key is (e, N), private key is d**

- **to encode message m, $c = m^e \bmod N$**
- **to decode message c, $m = c^d \bmod N$**
- **decoding is easy if you know d, but hard if you don't:**
  - you have to figure out p-1 and q-1
  - so you have to figure out p and q
  - so you have to factor N
  - and that's too hard

A tiny example

- **$p = 17$, $q = 11$ (primes)**
- **$N = 17 \times 11 = 187$**
- **$e = 7$ (relatively prime to $p-1 \times q-1 = 16 \times 10 = 160$)**
- **$d = 23$ ($7 \times 23 = 161 = 1 \bmod 160$)**
- **public key is (7, 187); private key is 23**

- **to encode "X" (88 in ascii),**
  compute $88^7 \bmod 187 = 11$

- **to decrypt 11,**
  compute $11^{23} \bmod 187 = 88$

Properties of public/private keys

- **can't deduce the public key from the private, or vice versa**
- **can't find another encryption key that works with the decryption key**
- **keys are long enough that brute force search is infeasible**

- **nasty problems:**
  - if a key is lost, all messages and signatures are lost
  - if a key is compromised, all messages and signatures are compromised
  - it's hard to revoke a key
  - it's hard to repudiate a key (and hard to distinguish that from revoking)

- **authentication**
  - how do you know who you are talking to? is that really Alice's public key?
  - public key infrastructure, web of trust, digital certificates

Crypto politics

- **cryptographic techniques as weapons of war?**
  - until recently, (strong) cryptography was classified as "munitions" in USA
  - falls under International Traffic in Arms Regulations and follow-ons
- **export control laws prohibited export of cryptographic code**
  - though it was ok to export books and T-shirts with code
    and everyone else in the world had it anyway
  - changed during 2000, but there are still restrictions
- **does the government have the right/duty ...**
  - to control cryptographic algorithms and programs?
  - to require trapdoors, key escrow, or similar mechanisms?
  - to prevent reverse-engineering of cryptographic devices?
  - to prevent research in cryptographic techniques?
- **do corporations have the right ...**
  - to prevent publication of cryptographic techniques?
  - to prevent reverse-engineering of cryptographic devices?
- **how do we balance individual rights, property rights, & societal rights?**

Summary of crypto

- **secret/symmetric key algorithms: DES, AES**
  - key distribution problem: everyone has to have the key
- **public key algorithms: RSA, ...**
  - solves key distribution problem, but authentication is still important
  - also permits digital signatures
  - much slower than secret key, so used mainly for key exchange
- **security is entirely in the key**
  - "security by obscurity" does not work: bad guys know everything
  - brute force attacks work if keys are too short or easy
- **good cryptography is hard**
  - you can't invent your own methods
  - you can't trust "secret" or proprietary methods
- **people are the weak link**
  - complicated or awkward systems will be subverted, ignored or misused
  - social engineering attacks are effective
    ignorance, incompetence, misguided helpfulness
- **if all else fails, try bribery, burglary, blackmail, brutality**