Outline

- Chapter 18: Protection
- Chapter 19: Security
- Paper: *A Method for Obtaining Digital Signatures and Public-Key Cryptosystems* Ronald L. Rivest, Adi Shamir, and Leonard M. Adleman. Communications of the ACM 21,2 (Feb. 1978)
  - RSA Algorithm – First practical public key crypto system

Protection

- Protect computer resources from being accessed by processes that should not have access
- Access Matrix defines protection
  - Global table
  - Access list for objects: easier to program
  - Capability list for domains/users:
  - Hybrid: lock-key mechanism
- Revocation of rights:
  - Immediate vs delayed, selective vs general, partial vs total, temporary vs permanent

Compiler/language based mechanism

- Compiler based enforcement
- Java VM
  - Multiple threads within a single JVM have different access rights
  - Enforced using stack inspection

Security

- User authentication:
  - Passwords
    - Encrypted passwords
    - One-time passwords
    - Biometrics
- Threats:
  - Trojan horse
  - Trap door/stack and buffer overflow
  - Worms/viruses
  - Denial of service
  - Intrusion and detection

Risk analysis

- Important to understand threat and perform risk analysis
  - No system is “secure”, systems usually trade security for performance, ease of use etc.
  - If information is worth **x** and it costs **y** to break into system and if (**x** < **y**), then not worth encryption
  - Wasteful to build a system that is more secure than is necessary
    - Ssh in CSE dept – good
  - Palm pilots may not require powerful encryption systems if they are expected to be physically secure

Security Attacks

- Social engineering attacks
  - Preys on people gullibility (good nature), hardest to defend
    - E.g. I once got an unlisted number from a telephone operator because I sounded desperate (I was, but that was not the point)
    - E.g. Anna kour*va virus
    - E.g. If I walk in with coupla heavy looking boxes into the elevator to go to Fitz 3rd floor (at night) would you let me in? You can go into “secure” companies by looking like you “belong” there
- Denial of service attacks
  - Network flooding, Distributed DOS, holding resources, viruses

Common technology - firewalls

- Firewalls are used to restrict the kinds of network traffic in/out of companies
  - Application level proxies
  - Packet level firewalls
- Does not prevent end-to-end security violations
  - People sometimes email list of internal computer users outside firewall to scrupulous “researchers”
  - Emails viruses exploit certain vulnerabilities in VBS to get around firewalls

Encryption methods

- Symmetric cryptography
  - Sender and receiver know the secret key (apriori)
    - Fast encryption, but key exchange should happen outside the system
- Asymmetric cryptography
  - Each person maintains two keys, public and private
    - M ≡ PrivateKey(PublicKey(M))
    - M ≡ PublicKey (PrivateKey(M))
  - Public part is available to anyone, private part is only known to the sender
  - E.g. Pretty Good Privacy (PGP), RSA

My Public Key

```
-----BEGIN PGP PUBLIC KEY BLOCK-----
Version: PGPfreeware 7.0.3 for non-commercial use <http://www.pgp.com>

mQGiBDqtLPwRBADnG0+9IkDvI8t/3wdL3CSO4DytEH0NjrNwAYYIaewp3MklsxkP
p6iVblwiiCH4T4Nqkaru+kaEQ1hSTa7E/F9yQCWN5J0u1U7mtgTKFyt7VG0txAVx
tV7TuyxNogJkpm2BqoKqqUdCdbm+GurX/G2ynbINjEOvhcy0i1ttxgyDrwCg/8HZ
tM0i06VVNcR/QCmA+JdHGwMEAIjXLVV97huEtpuWDiq4J53ecV3HXQm6XoUZq4Sc
n+nsvXe4UD+6ldub/riOqBy22fBBAKhUsM3lGFgr7h19X3RGdw/yBVox+BLajpW+
F+ddjJAVSFeTvNanhnXL9a3nwCThb4aEUTdD6lkgoUWJ12BnsK1DUSo2X6AsZYo+
GknOA/92dUNYUzspPLkXvPjOo+uJErZA4aN+UYsJwD3AlYugVLkc3nQBQySO4bAR
XitjnN0DA6Kz/j6e+cqReCyEuBnPtaY/Nn/dAn1lgUlJ/EtKQ9J4krI3+RxRmlpY
UtWyTaakV/QCXkB/yB9i6iAfsCprlcRSpmZAGuNXr+pHTHB0ILQmU3VyZW5kYXIg
Q2hhbmRyYSA8c3VyZW5kYXJAY3MudWdhLmVkdT6JAFgEEBECABgFAjqtLPwICwMJ
CAcCAQoCGQEFGwMAAAAACgkQlU7dFVWfeisqTACfXxU9a1mbouW2nbWdx6MHatQ6
TOgAoM9W1PBRW8Iz3BIgcnSsZ2UPNJHDuQINBDqtLPwQCAD2Qle3CH8IF3Kiutap
QvMF6PlTETlPtvFuuUs4INoBplajFOmPQFXz0AfGy0OplK33TGSGSfgMg7Il6RfU
odNQ+PVZX9x2Uk89PY3bzpnhV5JZzf24rnRPxfx2vIPFRzBhznzJZv8V+bv9kV7H
AarTW56NoKVyOtQa8L9GAFgr5fSI/VhOSdvNILSd5JEHNmszbDgNRR0PfIizHHxb
LY7288kjwEPwpVsYjY67VYy4XTjTNP18F1dDox0YbN4zISylKv884bEpQBgRjXyE
pwpylobEAxnIByl6ypUM2Zafq9AKUJsCRtMIPWakXUGfnHy9iUsiGSa6q6Jew1Xp
Mgs7AAICCACLxNC3Vth553Y90JCVyM9mPWzvrkjfEGfBiCFDZ0HONW81ywUyV6jT
O/lsUsgR7jGB26XBsnIY96a9WTpUoI+20YstFLRj8sXOVXuaP/YTmgSLv82O6SWd
Bze1S0YJcU31/zdCftsz67UWT8vg39yeGyQ5KQP83p9DKpi4Z5K4M29p8eCt9BY+
kid94h9+16ZT8JLF0iEwGapZvpaTucCNoC8t6CKPto0dGpkYp7uBYoSzLgNvUh2n
BjGVEmLuioabqbOaomDErITY2iNcW3CCgjjYvgg/Hnu7HB2xKzuVUN1NTGogcuNI
Yx88mi+d/HxTY6YNr9xNW0f0pWkZDVB0iQBMBBgRAgAMBQI6rSz8BRsMAAAAAAoJ
EJVO3RVVn3orYhIAoIQPxGvHmX8c6kaAZgkolzYCeixcAJ9tp5h/KQZrIN/BpyTW
9Xgv4qxKEA==
=Pv5O
-----END PGP PUBLIC KEY BLOCK-----
```

RSA

- Named after Rivest, Shamir and Adleman
  - Only receiver receives message:
    - Encode message using receivers public key
  - Only sender could've sent the message
    - Encode message using sender's private key
  - Only sender could've sent the message and only receiver can read the message
    - Encode message using receivers public key and then encode using our private key

Strength

- Strength of crypto system depends on the strengths of the keys
- Computers get faster – keys have to become harder to keep up
- If it takes more effort to break a code than is worth, it is okay
  - Transferring money from my bank to my credit card and Citibank transferring billions of dollars with another bank should not have the same key strength

Public Key Infrastructure (PKI)

- Process of issuing, delivering, managing and revoking public keys
- E.g. Secure Sockey Layer (SSL)
  - Client C connects to Server S
    1. C requests server certificate from S
    2. S sends server certificate with Spublic to C
    3. C verifies validity of Spublic
    4. C generate symmetric key for session
    5. C encrypts Csymmetric using Spublic
    6. C transmits Csymmetric(data) and Spublic(Csymmetric) to S

Authentication

- Identification verification process
  - E.g. kerberos certificates, digital certificates, smart cards
- Used to grant resources to authorized users

Practical Public Key Cryptosystem

1. Decrypt(Encrypt(Message)) = Message
2. Encrypt() and Decrypt() are easy to compute
3. Encrypt() does not reveal Decrypt()
4. Encrypt(Decrypt(Message)) = Message

- Function satisfying 1-3: Trap-door one-way function
  - One way: easy to compute in one direction, difficult in the other direction
  - Trap-door: Inverse functions are easy to compute once certain private “trap-door” information is known.
- 1-4: permutation

Signature

- Encrypt using private key of sender. Anyone can decrypt using the public key of sender to verify signature

-----BEGIN PGP SIGNED MESSAGE-----
Hash: SHA1

Hello world!!
-----BEGIN PGP SIGNATURE-----
Version: PGPfreeware 7.0.3 for non-commercial use
<http://www.pgp.com>

iQA/AwUBOq8LO5VO3RVVn3orEQLFZwCdGi9AWvlhollaYmr9TPvtdbK
oe20AoLLr
vbJ8SgkIZ73lCy6SXDi91osd
=L3Sh
-----END PGP SIGNATURE-----

Privacy

- Encrypt with receivers public key

-----BEGIN PGP MESSAGE-----
Version: PGPfreeware 7.0.3 for non-commercial use <http://www.pgp.com>

qANQR1DBwU4D30m79rqmjHMQB/4q1mu3IP8AsMBYSUW6udXZnF0/LVL51eYzVnAW
IxgbxhHmBoZf9YEltoXw82gkgVebz+3Xfj6T5mLNy5FA6cgKKw57AY9Bl3aEKlJK
/nV5qR8E/VZOhaPoog8dtV1Hpi5Z0vNCI7s5Ibp3C2tlrgYtvyYfe86bqCNe3yAI
btTUT+bA9HL3pXqhOoWIRB+N58T9ybn/9FyonYYfGuPdMTj+ZciK37R+ezWg5YmZ
jdDMf/CxgllMF/Tv2jQ8KgmrKIyi6gWQmEtWzFUlAPgdpOC7TQC3sQqVjK4GyOY6
WnrXiWqO3895ukBGyHzqyssUTJFe5qnclkrmCvA3tph+uc7pCACKrYaGLSWWoQSB
L6zch2GnhG4+JpDCVKF/poJ1URkB2Odd9/OCReR0sFXZFvW14IJQznu3HOhjtA+y
g7Nn736fqMD9jpBZFfUtKv/v4JMyWcRdp3R3icm03zi24n+244r1DQj+cVlFYPfd
zRAGTLORVjXH2amGqilKyxqMU7ZYXIMI43bFlviu4tabKYnZJxpM8keUKA3u+vPs
X9ksSoBSiT6Kow3Lac2t3Qo5TimYIS5ODFnC6Pp9aRZzNcBOKmiYO4IlbdFH2jta
RbcmesEjH5RpbDV4BfcOMdm2UGWZe6kAaKkSdxHlUVZAJnesbT+lQf4AZjXkmsOM
8qnBKi5xyS/wrhS4zamV/Mp+5qIGNASXUHPsp3rukovaZANdZ/Y6zNQQVim0kphd
5ECybmVrHQ==
=S9ph
-----END PGP MESSAGE-----

Algorithm

- To break their algorithm requires that you factor a large prime
  - Computationally very hard. Can't be "proven" yet
  - With present technology, 512 bit key takes a few months to factor using "super computers", 1024 takes a long time and 2048 takes a very long time
  - Takes 2 seconds to generate a 2048 bit key on a 933 Mhz Pentium
  - Algorithm has remained secure for the past 17 years
  - One of the most successful public key system